i.MX 6Dual/6Quad Applications Processor Reference Manual

Document Number: IMX6DQRM
Rev 2, 06/2014

freescale™

Contents

Section number **Title** **Page**

Chapter 1
Introduction

Chapter 2
Memory Maps

Chapter 3
Interrupts and DMA Events

Chapter 4
External Signals and Pin Multiplexing

Chapter 5
Fusemap

Chapter 6
External Memory Controllers

Chapter 7
System Debug

Chapter 8
System Boot

Chapter 9
Multimedia

Chapter 10
Clock and Power Management

Chapter 11
System Security

Chapter 12
ARM Cortex A9 MPCore Platform (ARM)

Chapter 13
AHB to IP Bridge (AIPSTZ)

Chapter 14
AHB-to-APBH Bridge with DMA (APBH-Bridge-DMA)

Chapter 15
Asynchronous Sample Rate Converter (ASRC)

Chapter 16
Digital Audio Multiplexer (AUDMUX)

Chapter 17
40-BIT Correcting ECC Accelerator (BCH)

Chapter 18
Clock Controller Module (CCM)

Chapter 19
MIPI CSI to IPU Gasket (CSI2IPU)

Chapter 20
Display Content Integrity Checker (DCIC)

Chapter 21
Enhanced Configurable SPI (ECSPI)

Chapter 22
External Interface Module (EIM)

Chapter 23
10/100/1000-Mbps Ethernet MAC (ENET)

Chapter 24
Enhanced Periodic Interrupt Timer (EPIT)

Chapter 25
Enhanced Serial Audio Interface (ESAI)

Chapter 26
Flexible Controller Area Network (FLEXCAN)

Chapter 27
General Power Controller (GPC)

Chapter 28
General Purpose Input/Output (GPIO)

Chapter 29
General Purpose Media Interface (GPMI)

Chapter 30
General Purpose Timer (GPT)

Chapter 31
2D Graphics Processing Unit (GPU2D)

Chapter 32
3D Graphics Processing Unit (GPU3D)

Chapter 33
HDMI Transmitter (HDMI)

Chapter 34
HDMI 3D Tx PHY (HDMI_PHY)

Chapter 35
I2C Controller (I2C)

Chapter 36
IOMUX Controller (IOMUXC)

Chapter 37
Image Processing Unit (IPU)

Chapter 38
Keypad Port (KPP)

Chapter 39
LVDS Display Bridge (LDB)

Chapter 40
MIPI - Camera Serial Interface Host Controller (MIPI_CSI)

Chapter 41
MIPI DSI Host Controller (MIPI_DSI)

Chapter 42
MIPI HSI Host Controller (MIPI_HSI)

Chapter 43
MediaLB (MLB)

Chapter 44
Multi Mode DDR Controller (MMDC)

Chapter 45
Network Interconnect Bus System (NIC-301)

Chapter 46
On-Chip OTP Controller (OCOTP_CTRL)

Chapter 47
On-Chip RAM Memory Controller (OCRAM)

Chapter 48
PCI Express (PCIe)

Chapter 49
PCI Express PHY (PCIe_PHY)

Chapter 50
Power Management Unit (PMU)

Chapter 51
Pulse Width Modulation (PWM)

Chapter 52
ROM Controller with Patch (ROMC)

Chapter 53
Serial Advanced Technology Attachment Controller (SATA)

Chapter 54
Serial Advanced Technology Attachment PHY (SATA PHY)

Chapter 55
Smart Direct Memory Access Controller (SDMA)

Chapter 56
System JTAG Controller (SJC)

Chapter 57
Secure Non-Volatile Storage (SNVS)

Chapter 58
Shared Peripheral Bus Arbiter (SPBA)

Chapter 59
Sony/Philips Digital Interface (SPDIF)

Chapter 60
System Reset Controller (SRC)

Chapter 61
Synchronous Serial Interface (SSI)

Chapter 62
Temperature Monitor (TEMPMON)

Chapter 63
TrustZone Address Space Controller (TZASC)

Chapter 64
Universal Asynchronous Receiver/Transmitter (UART)

Chapter 65
Universal Serial Bus Controller (USB)

Chapter 66
Universal Serial Bus 2.0 Integrated PHY (USB-PHY)

Chapter 67
Ultra Secured Digital Host Controller (uSDHC)

Chapter 68
Video Data Order Adapter (VDOA)

Chapter 69
Video Processing Unit (VPU)

Chapter 70
Watchdog Timer (WDOG)

Chapter 71
Crystal Oscillator (XTALOSC)

Chapter 1
Introduction

1.1 About This Document

The i.MX 6Dual/6Quad application processors are Freescale Semiconductor's latest additions to a growing family of multimedia-focused products offering high performance processing optimized for lowest power consumption.

The i.MX 6Dual/6Quad processors feature Freescale's advanced implementation of the ARM®Cortex®-A9 core, which can be interfaced with DDR3-1066, LV-DDR3-1066 and LPDDR2-1066(single and dual channel) DRAM memory devices.

These products are suitable for applications such as:

- Automotive navigation and entertainment
- High-end Mobile Internet Devices and high-end PDAs
- Netbooks
- Nettops
- High-end portable media players with HD video capability
- Portable navigation devices
- Gaming Consoles

1.1.1 Audience

This manual is intended to be used by board-level product designers and product software developers. This manual assumes that the reader has a background in computer engineering and/or software engineering and understands concepts of digital system design, microprocessor architecture, Input / Output (I/O) devices, industry standard communication and device interface protocols.

1.1.2 Organization

This document covers the i.MX 6Dual/6Quad at a system level and provides an architectural overview. Also covered are system memory map, system-level interrupt events, external pins and pin multiplexing, external memory, system debug, system boot, multimedia subsystem, power management, and system security.

1.1.3 Suggested Reading

This section lists additional reading materials that provide background for the information in this manual, as well as general information about the architecture.

1.1.3.1 General Information

The following documentation provides useful background information about the ARM Cortex-A9 processor.

For information about the ARM Cortex-A9 processor see:

- **http://infocenter.arm.com**

1.1.3.2 Related Documentation

Freescale documentation is available from the sources listed on the back cover of this manual; the document order numbers are included in parentheses for ease in ordering.

For a current list of documentation, refer to **http://www.freescale.com**.

1.1.4 Conventions

This document uses the following notational conventions:

cleared / set
: When a bit takes the value zero, it is said to be cleared; when it takes a value of one, it is said to be set.

mnemonics
: Instruction mnemonics are shown in lowercase bold

italics
: Italics indicate variable command parameters, for example, **bcctr***x*

: Book titles in text are set in italics

15
An integer in decimal

0x
Prefix to denote hexadecimal number

0b
Prefix to denote binary number. Binary 0 and 1 are written without the prefix.

n'H4000CA00
n-bit Hexadecimal number

BLK_REG_NAME
Register names are all uppercase. The block mnemonic is prepended with an underscore delimiter (_).

BLK_REG[FIELD]
Fields within registers appear in brackets. For example, ESR[RLS] refers to the Receive Last Slot field of the ESAI Status Register.

BLK_REG[*n*]
Bit number *n* within register BLK.REG.

BLK_REG[*l:r*]
Register bit ranges. Ranges are indicated by the left-most bit number *l* and the right-most bit number *r* separated by a colon (:). For example, ESR[15:0] refers to the lower half word in the ESAI Status Register.

x, U
In some contexts, such as signal encodings, an unitalicized x indicates a don't care or uninitialized. The binary value could be 1 or 0.

x
An italicized *x* indicates an alphanumeric variable

n, m
Italicized *n or m* represent integer variables

!
Binary logic operator NOT

&&
Binary logic operator AND

||
Binary logic operator OR

^ or <O+>
Binary logic operator XOR

|
Bit-wise OR. For example, 0b0001 | 0b1000 yields the value 0b1001.

&
Bit-wise AND. For example, 0b0001 & 0b1000 yields the value 0b0000.

{A,B}

Concatenation, where the *n*-bit value A is prepended to the *m*-bit value B to form an (*n* +*m*)-bit value. For example, {0, REG*m* [14:0]} yeilds a 16-bit value with 0 in the most significant bit.

- or grey fill

Indicates a reserved bit field in an register. Although these bits can be written to as ones or zeros, they are always read as zeros.

>>

Shift right logical one position

<<

Shift left logical one posiiton

= <left arrow>

Assignment

==

Compare equal

!=

Compare not equal

>

Greater than

<

Less than

1.1.5 Register Access

1.1.5.1 Register Diagram Field Access Type Legend

The following figure provides the interpretation of the notation used in register diagrams for a number of common field access types.

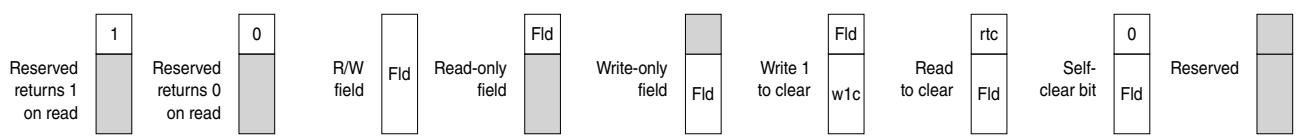


Figure 1-1. Register Field Conventions

NOTE

For reserved register fields, software should mask off the data in the field after read (software can not rely on the contents of data read from a reserved field) and always write all zeros.

1.1.5.2 Register Macro Usage

A common operation is to update one field without disturbing the contents of the remaining fields in the register. Normally, this requires a read-modify-write (RMW) operation, where the CPU reads the register, modifies the target field, then writes the results back to the register. This is an expensive operation in terms of CPU cycles, because of the initial register read.

To address this issue, some hardware registers are implemented as a group, including registers that can be used to either set, clear, or toggle (SCT) individual bits of the primary register. When writing to an SCT register, all bits set to 1 perform the associated operation on the primary register, while all bits set to 0 are not affected. The SCT registers always read back 0, and should be considered write-only. The SCT registers are not implemented if the primary register is read-only.

With this architecture, it is possible to update one or more fields using only register writes. First, all bits of the target fields are cleared by a write to the associated clear register, then the desired value of the target fields is written to the set register. This sequence of two writes is referred to as a clear-set (CS) operation.

A CS operation does have one potential drawback. Whenever a field is modified, the hardware sees a value of 0 before the final value is written. For most fields, passing through the 0 state is not a problem. Nonetheless, this behavior is something to consider when using a CS operation.

Also, a CS operation is not required for fields that are one bit wide. While the CS operation works in this case, it is more efficient to simply set or clear the target bit (that is, one write instead of two). A simple set or clear operation is also atomic, while a CS operation is not.

Note that not all macros for set, clear, or toggle (SCT) are atomic. For registers that do not provide hardware support for this functionality, these macros are implemented as a sequence of read/modify/write operations. When atomic operation is required, the developer should pay attention to this detail, because unexpected behavior might result if an interrupt occurs in the middle of the critical section comprising the update sequence.

1.1.6 Signal Conventions

_b, _B

When appended to a signal name, indicates that a signal is active-low

$\overline{\text{NEG\_ACTIVE}}$

Overbar also denotes a negative active signal

UPPERCASE

Package pin names, Block I/O signals

lowercase

Lowercase is used to indicate internal signals

1.1.7 Acronyms and Abbreviations

The table below contains acronyms and abbreviations used in this document.

Acronyms and Abbreviated Terms

Term	Meaning
AHB	Advanced High-performance Bus
ALU	Arithmetic Logic Unit
AMBA	Advanced Microcontroller Bus Architecture
APB	Advanced Peripheral Bus
AXI	Advanced eXtensible Interface
BIST	Built-in self test
DDR	Double data rate
DMA	Direct memory access
DPLL	Digital phase-locked loop
DRAM	Dynamic random access memory
ECC	Error correcting codes
EPROM	Erasable programmable read-only memory
FIFO	First-in-first-out
GPIO	General-purpose I/O
GPR	General-purpose register
GPS	Global Positioning System
GPU	Graphics Processing Unit
HAB	High Assurance Boot
I2C or I^2C	Inter-integrated circuit
IEEE	Institute of Electrical and Electronics Engineers
IrDA	Infrared Data Association
JTAG	Joint Test Action Group (a serial bus protocol usually used for test purposes)
LCD	Liquid Crystal Display
LCDIF	Liquid Crystal Display Interface
LDO	Low-Dropout
LIFO	Last-in-first-out
LRU	Least recently used
LSB	Least-significant byte
LUT	Lookup Table
LVDS	Low Voltage Differential Signaling
MAC	Medium Access Control

Table continues on the next page...

Term	Meaning
MMC	MultiMedia Card
MSB	Most-significant byte
MT/s	Mega transfers per second
OCRAM	On-Chip Random Access Memory
PCI	Peripheral Component Interconnect
PCIe	PCI enhanced
PCMCIA	Personal Computer Memory Card International Association
PGC	Power Gating Controller
PIC	Programmable interrupt controller
POR	Power-on reset
PSRAM	Pseudo-Static Random Access Memory
QoS	Quality of Service
R2D	Radians to Degrees
RISC	Reduced instruction set computing
ROM	Read-Only Memory
RTOS	Real-time operating system
Rx	Receive
SCU	Snoop Control Unit
SD	Secure Digital
SDIO	Secure Digital Input/Output
SDLC	Synchronous data link control
SDMA	Smart DMA
SoC	System-On-Chip
SPDIF	Sony Phillips Digital Interface
SPI	Serial peripheral interface
SRAM	Static random access memory
TFT	Thin Film Transistor
Tx	Transmit
UART	Universal asynchronous receiver/transmitter
USB	Universal serial bus
WLAN	Wireless Local Area Network
WXGA	Wide Extended Graphics Array

1.2 Introduction

This chapter introduces the architecture of the i.MX 6Dual/6Quad Multimedia Applications Processor.

The i.MX 6Dual/6Quad processor represents Freescale Semiconductor's latest achievement in integrated multimedia applications processors.

It is part of a growing family of multimedia-focused products, offering high performance processing optimized for the lowest power consumption.

1.3 Target Applications

The i.MX 6Dual/6Quad applications processor is tailored for use in multimedia-centric smart mobile devices, driver information systems including infotainment and graphical instrument clusters, and portable medical devices.

The architecture's flexibility also allows for use in a wide variety of general embedded applications. The heart of the application chipset, the i.MX 6Dual/6Quad processor provides all of the interfaces necessary for connecting peripherals such as WLAN, Bluetooth™, GPS, camera sensors, and multiple displays.

1.4 Features

The i.MX 6Dual/6Quad Application Processor (AP) is based on the ARM Cortex A9 MPCore™ Platform, which has the following features:

- ARM Cortex A9 MPCore™Dual or Quad core CPU configurations (with TrustZone)
- Symmetric CPU configuration where each CPU includes:
  - 32 Kbyte L1 Instruction Cache
  - 32 Kbyte L1 Data Cache
  - Private Timer and Watchdog
  - Cortex-A9 NEON MPE (Media Processing Engine) Co-processor.

The ARM Cortex A9 MPCore™ complex includes:

- General Interrupt Controller (GIC) with 128 interrupt support
- Global Timer
- Snoop Control Unit (SCU)
- 1 Megabyte unified L2 cache shared by all CPU cores (Dual or Quad)
- Two Master AXI (64-bit) bus interfaces output of L2 cache
- NEON MPE coprocessor
  - SIMD Media Processing Architecture
  - NEON register file with 32x64-bit general-purpose registers
  - NEON Integer execute pipeline (ALU, Shift, MAC)

- NEON dual, single-precision floating point execute pipeline (FADD, FMUL)
- NEON load/store and permute pipeline
- Supports single and double-precision add, subtract, multiply, divide, multiply and accumulate, and square root operations as described in the ARM VFPv3 architecture.
- Provides conversions between 16-bit, 32-bit and 64-bit floating-point formats and ARM integer word formats

The i.MX 6Dual/6Quad processor makes use of dedicated HW accelerators in order to meet the targeted multimedia performance. The use of HW accelerators is a key factor in obtaining high performance at low power consumption numbers, while keeping the CPU core relatively free to perform other tasks.

The i.MX 6Dual/6Quad processor incorporates the following hardware accelerators:

- VPU -Video Processing Unit
- Two IPUv3H -Image Processing Unit (version 3H)
- GPU3Dv4 - 3D/2D Graphics Processing Unit (OpenGL ES 2.0), version 4
- GPU2Dv2 - 2D Graphics Processing Unit (BitBlt).
- GPU VG - OpenVG 1.1 Graphics Processing Unit
- ASRC - Asynchronous Sample Rate Converter

Security functions are enabled and accelerated by the following hardware:

- ARM TrustZone including the TZ architecture (separation of interrupts, memory mapping, etc.)
- SJC- System JTAG Controller. Protecting JTAG from debug port attacks by regulating or blocking the access to the system debug features
- CAAM - Cryptographic Acceleration and Assurance Module, incorporating 16KB secure RAM
- SNVS - Secure Non-Volatile Storage, including Secure Real Time Clock
- CSU - Central Security Unit. Configured during boot and by e-fuses, and will determine the security level operation mode as well as the TZ policy
- A-HAB-Advanced High Assurance Boot - HABv4 with the next embedded enhancements: SHA-256, 2048 bit RSA key, version control mechanism, warm boot, CSU and TZ initialization

The memory system consists of the following levels:

- Level 1 Cache - 32KB Instruction, 32KB Data cache per each core.
- Level 2 Cache - Unified instruction and data (1MByte)
- On-Chip Memory
    - Boot ROM, including HAB (96 KB)

- Internal multimedia / shared, fast access RAM (256KByte)
- Secure/non-secure RAM (16 KB)

The chip enables the following interfaces to external devices (some of them are muxed and not available simultaneously):

- Hard Disk Drives
  - SATA II, 3.0 Gbps
- Displays
  - Five interfaces available. Total raw pixel rate of all interfaces is up to 532 Mpixels/sec, 24 bpp (266 Mpixels/sec per each IPU). Up to four interfaces may be active at once.
  - Two Parallel 24-bit display ports - up to 220 Mpixels/sec each (e.g. WUXGA+ @ 60Hz), or dual HD1080 + WXGA @ 60Hz.
  - LVDS serial ports: One port up to 165 Mpixels/sec or Two ports up to 85 MP/sec (e.g. WUXGA+ @ 60Hz) each
  - HDMI 1.4 port
  - MIPI/DSI, 2 lanes @ 1 Gbps
- Camera sensors
  - Up to three interfaces may be active at once
  - Two Parallel Camera ports (up to 20-bit, up to 240MHz peak[1] each).
  - MIPI CSI-2 / Parallel port, supporting from 80 Mbps up to 1 Gbps speed per data lane. The CSI-2 Receiver core can manage one clock lane and up to 4 data lanes. There are 4 lanes in i.MX 6Dual/6Quad.
- Expansion cards
  - Four SD/MMC/SDIO Controllers:
  - Fully compliant with MMC command/response sets and Physical Layer as defined in the Multimedia Card System Specification, v4.2/4.3/4.4/4.41 including high-capacity (size > 2GB) cards HC MMC. HW reset as specified for eMMC cards is supported at ports #3 and #4 only.
  - Fully compliant with SD command/response sets and Physical Layer as defined in the SD Memory Card Specifications, v3.0 including extended-capacity SDHC cards up to 32 GB.
  - Fully compliant with SDIO command/response sets and interrupt/read-wait mode as defined in the SDIO Card Specification, Part E1, v1.10
  - Fully compliant with SD Card Specification, Part A2, SD Host Controller Standard Specification, v2.00
  - Supports:

1. The peak value of 240MHz is based on IPU capability and applies to internal sources such as CSI-2. Using parallel input is subject to IO cell timing capability and board design signal timing constraints. The actual frequency is expected to be lower. Final frequency will be determined by IC characterization.

        - 1-bit or 4-bit transfer mode specifications for SD and SDIO cards up to UHS-I SDR104 mode (104MB/s max)
        - 1-bit, 4-bit, or 8-bit transfer mode specifications for MMC cards up to 52MHz in both SDR and DDR modes (104MB/s max)
    - Instances #1 and #2 are primarily intended to serve as external slots or interfaces to on-board SDIO devices
    - Instances #3 and #4 are primarily intended to serve as interfaces to embedded MMC memory or interfaces to on-board SDIO devices
    - All ports can work with 1.8V and 3.3V cards. There are two completely independent IO power domains for Ports #1 and #2.
- External memory interfaces
    - 16/32/64-bit DDR3-1066, LV-DDR3-1066 and 1 / 2 LPDDR2-1066 channels
    - 8-bit NAND-Flash, including support for Raw MLC/SLC, 2,4,8KB page size (up to 4x devices), BA-NAND, PBA-NAND, LBA-NAND, OneNAND™ and others. BCH ECC up to 40-bit.
    - 16-bit Nor Flash. All WEIMv2 pins are muxed on other interfaces.
    - 16-bit PSRAM, Cellular RAM.
- USB
    - High Speed (HS) USB 2.0 OTG (Up to 480 Mbps), with integrated HS USB Phy
    - Three USB 2.0 (480 Mbps) hosts:
        - HS host, with integrated HS Phy.
        - Two HS host, with integrated HS-IC USB (High Speed Inter-Chip USB) Phy.
- Low Power modes

  The i.MX 6Dual/6Quad integrates advanced power management unit and controllers:

    - PMU, including LDO supplies for on-chip resources
    - Temperature Sensor for monitoring the die temperature
    - Supporting DVFS techniques for low power modes
    - Uses SW State Retention, and Power Gating for ARM and MPE
    - Support for various levels of system power modes
    - Flexible clock gating control scheme
- Expansion PCI Express port (PCIe) v2.0 one lane
    - PCI express (Gen 2.0) dual mode complex supporting Root complex operations and Endpoint operations. x1 PHY configuration.
- Miscellaneous IPs and interfaces:
    - Three SSI's that support I2S/AC97, up to 1.4 Mbps each
    - Enhanced Serial Audio Interface (ESAI), up to 1.4 Mbps each channel
    - Five UART, up to 4.0 Mbps each
        - Providing RS232 interface

  - Supporting 9-bit RS485 multidrop mode
  - One of the five supports 8-wire (uart1) while the other four support 4-wire. (Due to SoC IOMUX limitation, as all UART IP are identical).
- Five eCSPI (Enhanced CSPI). Using chip selects to support multiple peripherals.
- Three I2C, supports 400 Kbps
- Gigabit Ethernet Controller (IEEE1588 compliant), 1Gbps/10/100 Mbps
- Four Pulse Width Modulators (PWM)
- System JTAG Controller (SJC)
- GPIO with interrupt capabilities
- 8x8 Key Pad Port (KPP)
- Sony Philips Digital Interface (SPDIF), Rx and Tx
- Two Controller Area Network (FlexCAN), 1 Mbps each
- Two Watchdog timers (WDOG)
- Audio MUX (AUDMUX)
- MLB (MediaLB) provides interface to MOST Networks (MOST25，MOST50，MOST150) with DTCP cipher accelerator

1.5 Architectural Overview

This section contains i.MX 6Dual/6Quad architectural details.

1.5.1 Block Diagram

A simplified block diagram is provided in the following section.

1.5.1.1 Simplified Block Diagram

A high level block diagram is shown in the figure below. This diagram provides a view of the i.MX 6Dual/6Quad's major sub-systems (processor domains, shared peripherals domain, memories, and so on) and logical connectivity.

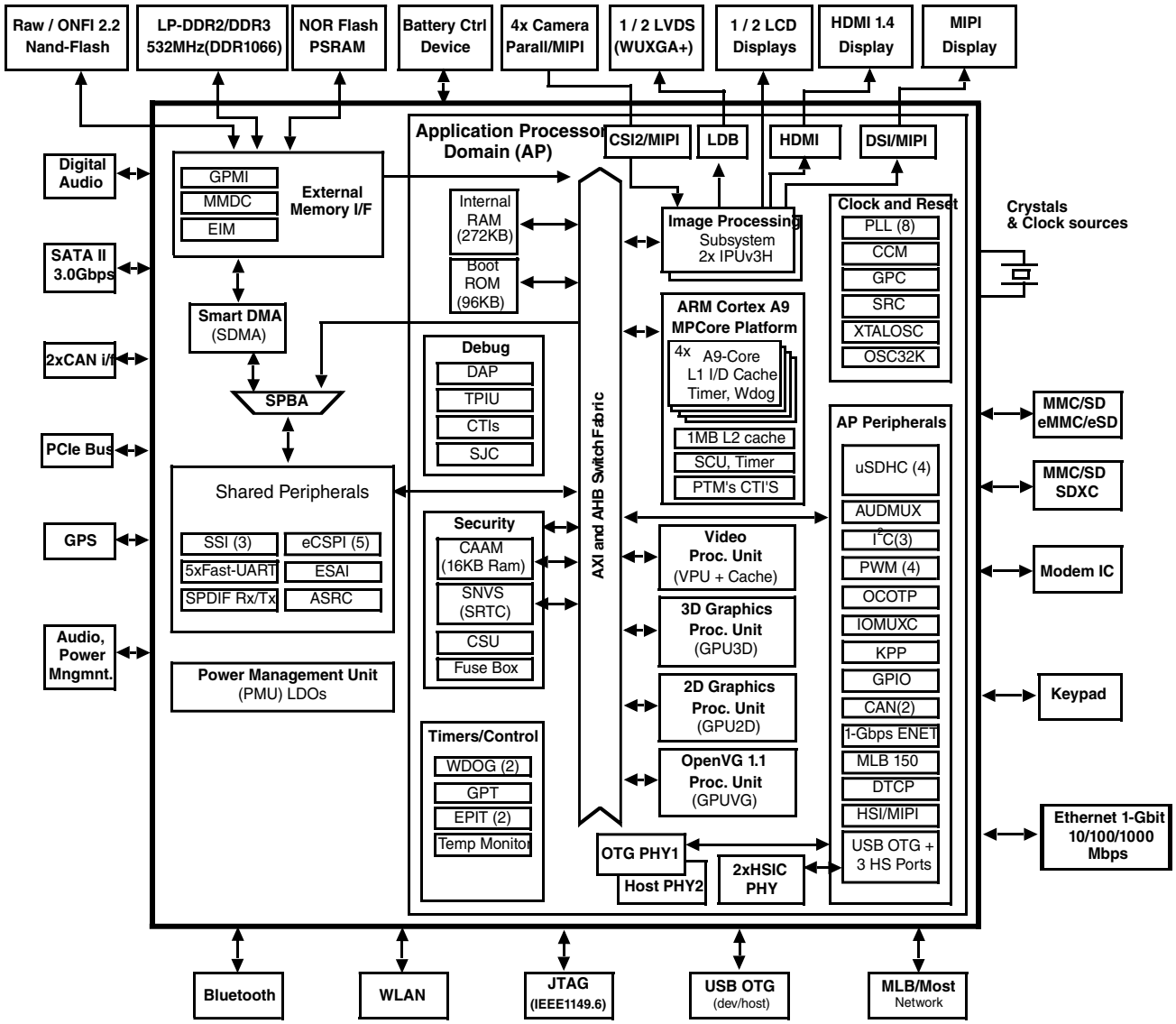


Figure 1-2. Simplified Block Diagram

1.5.2 Architectural Partitioning

Architecture supports processing-intensive tasks in the following ways:

- ARM Cortex A9 MPCore™ Platform is responsible for:
  - Operating System
  - User applications (including control over hardware accelerators and non-accelerated functions)
  - TrustZone applications

- Smart DMA enables data transfer between non-mastering peripherals and external or internal memories
- System Control is supported via:
  - Clock Control Module (CCM)
  - Eight PLLs and two PFDs
  - XTALOSC- 24MHz Crystal oscillator source support
  - OSC32KHz - 32.768Hz Crystal oscillator source support
  - System Reset Controller (SRC)
  - General Power Controller (GPC)
  - Temperature Sensor for monitoring and alarming on high temperature situations.
- Multimedia is supported with:
  - Two independent Image Processing Units-IPUv3H
    - Connectivity to displays and controllers (Parallel, LVDS, MIPI, HDMI), cameras (parallel, MIPI).
    - Display Processing: video/graphics combining, image enhancement
    - Image conversions: resizing, rotation/inversion, color conversion, de-interlacing
    - Synchronization and control capabilities, allowing autonomous operation
  - Video Processing Unit (VPU), supports various decoding/encoding formats in HW, up to 1080p plus SD 30fps decoding (H.264, VC1, RV10, DivX, etc.), and up to 1080p 30fps encoding (H.264, etc.).
  - Graphics Processing Unit (GPU3Dv4) graphics processing compliant with the following:
    - OpenGL ES 1.1 and 2.0 including extensions.
    - OpenVG 1.1
    - Windows Direct3D
    - OpenCL EP
  - Graphics Processing Unit (GPU2Dv2)
  - Vector graphics processing unit (GPUVGv2):
    - Real-time hardware curve tesselation of lines, quadratic and cubic Bezier curves
    - 16x Line Anti-aliasing
    - OpenVG 1.1 support
    - Vector Drawing functions
  - Audio
    - Audio codecs are provided by SW, which runs on ARM core, supporting (but not limited to) MP3, WMA, AAC, HE-AAC and Pro10
    - 3x SSIs
    - ESAI
    - SPDIF Tx/Rx

        - Audio Mux
        - ASRC (Audio sample rate conversion accelerator)
- Security is supported by:
    - High Assurance Boot (HAB4) System
    - ARM TrustZone (TZ) Trusted Execution environment
    - DDR Memory secure region protection by TrustZone Address Space Controller
    - On-chip RAM (OCRAM) single region TrustZone protection
    - Peripheral access policy control, using Central Security Unit (CSU)
    - One-Time Programmable (OTP) electrical fuse array (Total of 3840-bit e-fuses) via the On-chip electrical fuse controller (OCOTP_CTRL)
    - CAAM and SNVS security architecture, providing:
        - 16KB Secure RAM
        - Secure Real Time Counters
        - Security State Controller
        - Encryption and Hashing functions, Random Number Generator (RNG)
        - Security Violation/Tamper Detection & Reporting
    - System JTAG controller (SJC)
    - Secure Real Time Clock (SRTC)
    - TrustZone Watchdog (TZ WDOG)
- Connectivity peripherals, timers and External Memory Interfaces:
    - Embedded DMAs
    - 3.3V IO voltage for seamless integration
    - Four USB 2.0 ports, including four PHYs: 2x HS-USB (OTG + HOST) and 2x HS-IC USB integrated PHYs
    - Memory, 1066 MT/s per Line, supports LP-DDR2 (2x16, 2x32, 2x32 interleaved mode - x64), DDR3 (x16/x32/x64) and DDR3L (x16/x32/x64)
    - Nand-Flash (MLC up to 40-bit ECC) and NOR Flash memory interface via GPMI Nand-Flash controller
    - Timers: 2xEPIT, GPT and two Watch Dog timers (one of which is used for TZ), in addition to the timers and watchdog timers integrated within the ARM Cortex A9 MPCore™ platform.
    - Miscellaneous connectivity support - SATA, PCIe, FLEXCAN, MLB, MMC/SD, I2C, SPI, UART, PWM and Keypad interface

1.5.3 Endianness Support

i.MX 6Dual/6Quad supports Little Endian mode only.

1.5.4 Memory Interfaces

i.MX 6Dual/6QuadMulti-Port DRAM/DDR controller (MMDC) supports the following memory interfaces:

- LP-DDR2-1066, 1 or 2 channels x32-bit, 532MHz clock.
- DDR3 / LV-DDR3 32/64-bit, 532MHz clock.

NAND Flash controller supports the following:

- x8 NAND Flash bus.
- Configurable page size of 2KB, 4KB, 8KB.
- Configurable spare area per page of up to 512B
- Supports interleaved accesses to up to 8 NAND devices
- Legacy raw SLC, MLC type device
- Supports up to 40-bit ECC by HW BCH ECC accelerator, capable of 200MB/s encode/decode rate.
- ONFI2.2 compliant
    - Timing modes 0-5 for both asynchronous and synchronous I/F. Synchronous clock rate of up to 100MHz with data rate of up to 200MB/s.
- Vendor specific devices: BA-NAND (Micron), PBA-NAND and LBA-NAND (Toshiba), E2-NAND (Hynix), EF-NAND (Samsung), Samsung's "Toggle-mode" NAND (clock rate of up to 66MHZ and 80Mz, with data rate of up to 133MB/s and 160MB/s respectively)

NOR Flash, SRAM and PSRAM, 16/8-bit NOR Flash, interface is supported by the EIM block. All EIM pins are muxed on other interfaces.

Chapter 2
Memory Maps

2.1 Memory system overview

This chapter introduces the memory architecture of the i.MX 6Dual/6Quad chip.

2.2 ARM platform memory map

The i.MX 6Dual/6Quad memory map has been provided in the following tables. The mapping of the DDR memory address space can be configured by software or hardware fuses, depending on the memory device type (LPDDR2/DDR2/DDR3) and the selected interleave/non-interleaving scheme.

A combined memory map (both 2x32 and x64) is shown in the following table.

Table 2-1. System memory map

Start address	End address	Size	Description
1000_0000	FFFF_FFFF	3840 MB	MMDC - DDR Controller. See DDR mapping to MMDC controller ports
0800_0000	0FFF_FFFF	128 MB	EIM - CSO (NOR/SRAM)
02A0_0000	02DF_FFFF	4 MB	IPU-2
0260_0000	029F_FFFF	4 MB	IPU-1
0220_C000	023F_FFFF	2 MB	Reserved
0220_8000	0220_BFFF	16 KB	MIPI_HSI
0220_4000	0220_7FFF	16 KB	OpenVG (GC355)
0220_0000	0220_3FFF	16 KB	SATA
0210_0000	021F_FFFF	1 MB	Table 2-3 Peripheral IPs via AIPS-2.
0200_0000	020F_FFFF	1 MB	Table 2-2 Peripheral IPs via AIPS-1.
01FF_C000	01FF_FFFF	16 KB	PCIe registers
0100_0000	01FF_BFFF	16368 KB	PCIe
00D0_0000	00FF_FFFF	3072 KB	Reserved

Table continues on the next page...

Table 2-1. System memory map (continued)

Start address	End address	Size	Description
00C0_0000	00CF_FFFF	1 MB	"fast1" configuration port
00B0_0000	00BF_FFFF	1 MB	"fast2" configuration port
00A0_3000	00AF_FFFF	1012 KB	Reserved
00A0_2000	00A0_2FFF	4 KB	PL310 (L2 Cache controller)
00A0_0000	00A0_1FFF	8 KB	ARM MP 0000 - 00FCh SCU registers 0100 - 01FFh Interrupt controller interfaces 0200 - 02FFh Global timer 0300 - 05FFh Reserved 0600 - 06FFh Private timers and watchdogs 0700 - 0FFFh Reserved 1000 - 1FFFh Interrupt distributor
0094_0000	009F_FFFF	0.75 MB	OCRAM aliased
0090_0000	0093_FFFF	0.25 MB	OCRAM 256 KB
0080_0000	008F_FFFF	1 MB	GPV_4 PL301 configuration port
0040_0000	007F_FFFF	4 MB	Reserved
0030_0000	003F_FFFF	1 MB	GPV_3 PL301 configuration port
0020_0000	002F_FFFF	1 MB	GPV_2 PL301 configuration port
0013_C000	001F_FFFF	784 KB	Reserved
0013_8000	0013_BFFF	16 KB	DTCP
0013_4000	0013_7FFF	16 KB	GPU 2D (GC320)
0013_0000	0013_3FFF	16 KB	GPU 3D
0012_9000	0012_FFFF	28 KB	Reserved
0012_0000	0012_8FFF	36 KB	HDMI
0011_8000	0011_FFFF	32 KB	Reserved
0011_4000	0011_7FFF	16 KB	BCH
0011_2000	0011_3FFF	8 KB	GPMI
0011_0000	0011_1FFF	8 KB	APBH-Bridge-DMA
0010_4000	0010_FFFF	48 KB	Reserved
0010_0000	0010_3FFF	16 KB	CAAM (16K secure RAM)
0001_8000	000F_FFFF	928 KB	Reserved
0000_0000	0001_7FFF	96KB	Boot ROM (ROMCP)

Table 2-2 shows the AIPS-1 detailed memory map.

Table 2-2. AIPS-1 memory map

Start Address	End Address	Region	NIC Port	Size
020F_C000	020F_FFFF	AIPS-1	Reserved	16 KB

Table continues on the next page...

Table 2-2. AIPS-1 memory map (continued)

Start Address	End Address	Region	NIC Port	Size
020F_8000	020F_BFFF		Reserved	16 KB
020F_4000	020F_7FFF		Reserved	16 KB
020F_0000	020F_3FFF		Reserved	16 KB
020E_C000	020E_FFFF		SDMA	16 KB
020E_8000	020E_BFFF		DCIC2	16 KB
020E_4000	020E_7FFF		DCIC1	16 KB
020E_0000	020E_3FFF		IOMUXC	16 KB
020D_C2C0	020D_FFFF		Reserved	15680 B
020D_C2A0	020D_C2BF		PGC_ARM	32 B
020D_C280	020D_C29F		Reserved	32 B
020D_C260	020D_C27F		PGC_PU	32 B
020D_C000	020D_C25F		GPC	608 B
020D_8000	020D_BFFF		SRC	16 KB
020D_4000	020D_7FFF		EPIT2	16 KB
020D_0000	020D_3FFF		EPIT1	16 KB
020C_C000	020C_FFFF		SNVS_HP	16 KB
020C_B000	020C_BFFF		Reserved	4 KB
020C_A000	020C_AFFF		USBPHY2	4 KB
020C_9000	020C_9FFF		USBPHY1	4 KB
020C_8000	020C_8FFF		ANALOG: (PLLs, PFDs, Regulators, LDOs, Temp Sensor)	4 KB
020C_4000	020C_7FFF		CCM	16 KB
020C_0000	020C_3FFF		WDOG2	16 KB
020B_C000	020B_FFFF		WDOG1	16 KB
020B_8000	020B_BFFF		KPP	16 KB
020B_4000	020B_7FFF		GPIO7	16 KB
020B_0000	020B_3FFF		GPIO6	16 KB
020A_C000	020A_FFFF		GPIO5	16 KB
020A_8000	020A_BFFF		GPIO4	16 KB
020A_4000	020A_7FFF		GPIO3	16 KB
020A_0000	020A_3FFF		GPIO2	16 KB
0209_C000	0209_FFFF		GPIO1	16 KB
0209_8000	0209_BFFF		GPT	16 KB
0209_4000	0209_7FFF		CAN2	16 KB
0209_0000	0209_3FFF		CAN1	16 KB
0208_C000	0208_FFFF		PWM4	16 KB
0208_8000	0208_BFFF		PWM3	16 KB
0208_4000	0208_7FFF		PWM2	16 KB
0208_0000	0208_3FFF		PWM1	16 KB

Table continues on the next page...

Table 2-2. AIPS-1 memory map (continued)

Start Address	End Address	Region	NIC Port	Size
0207_C000	0207_FFFF		AIPS-1 Configuration	16 KB
0204_0000	0207_BFFF	AIPS-1	VPU	240 KB
0203_C000	0203_FFFF	AIPS-1 (s_b_0,via SPBA) Glob,Module ENABLE	SPBA	16 KB
0203_8000	0203_BFFF		Reserved for SDMA internal registers	16 KB
0203_4000	0203_7FFF		ASRC	16 KB
0203_0000	0203_3FFF		SSI3	16 KB
0202_C000	0202_FFFF		SSI2	16 KB
0202_8000	0202_BFFF		SSI1	16 KB
0202_4000	0202_7FFF		ESAI	16 KB
0202_0000	0202_3FFF		UART1	16 KB
0201_C000	0201_FFFF		Reserved for SDMA internal registers	16 KB
0201_8000	0201_BFFF		eCSPI5	16 KB
0201_4000	0201_7FFF		eCSPI4	16 KB
0201_0000	0201_3FFF		eCSPI3	16 KB
0200_C000	0200_FFFF		eCSPI2	16KB
0200_8000	0200_BFFF		eCSPI1	16 KB
0200_4000	0200_7FFF		SPDIF	16 KB
0200_0000	0200_3FFF		Reserved for SDMA internal registers	16 KB

Table 2-3 shows the AIPS-2 detailed memory map.

Table 2-3. AIPS-2 memory map

Start Address	End Address	Region	Allocation	Size
021F_C000	021F_FFFF	AIPS-2	Reserved	16 KB
021F_8000	021F_BFFF		Reserved	16 KB
021F_4000	021F_7FFF		UART5	16 KB
021F_0000	021F_3FFF		UART4	16 KB
021E_C000	021E_FFFF		UART3	16 KB
021E_8000	021E_BFFF		UART2	16 KB
021E_4000	021E_7FFF		VDOA	16 KB
021E_0000	021E_3FFF		MIPI (DSI port)	16 KB
021D_C000	021D_FFFF		MIPI (CSI port)	16 KB
021D_8000	021D_BFFF		AUDMUX	16 KB
021D_4000	021D_7FFF		TZASC2	16 KB
021D_0000	021D_3FFF		TZASC1	16 KB
021C_C000	021C_FFFF		Reserved	16 KB
021C_8000	021C_BFFF		Reserved	16 KB

Table continues on the next page...

Table 2-3. AIPS-2 memory map (continued)

Start Address	End Address	Region	Allocation	Size
021C_4000	021C_7FFF		Reserved	16 KB
021C_0000	021C_3FFF		CSU	16 KB
021B_C000	021B_FFFF		OCOTP_CTRL	16 KB
021B_8000	021B_BFFF		EIM - Registers	16 KB
021B_4000	021B_7FFF		MMDC (port 1)	16 KB
021B_0000	021B_3FFF	AIPS-2	MMDC	16 KB
021A_C000	021A_FFFF		ROMCP	16 KB
021A_8000	021A_BFFF		I2C3	16 KB
021A_4000	021A_7FFF		I2C2	16 KB
021A_0000	021A_3FFF		I2C1	16 KB
0219_C000	0219_FFFF		uSDHC4	16 KB
0219_8000	0219_BFFF		uSDHC3	16 KB
0219_4000	0219_7FFF		uSDHC2	16 KB
0219_0000	0219_3FFF		uSDHC1	16 KB
0218_C000	0218_FFFF		MLB150	16 KB
0218_8000	0218_BFFF		ENET	16 KB
0218_4000	0218_7FFF		USBOH3 (USB)	16 KB
0218_0000	0218_3FFF		Reserved	16 KB
0217_C000	0217_FFFF		AIPS-2 configuration	16 KB
0216_1000	0217_BFFF		ARM Cortex A9 MPCore Platform - Reserved	108 KB
0214_0000	0216_0FFF		ARM Cortex A9 MPCore Platform / DAP	132 KB (See Table 2-4)
0211_0000	0213_FFFF		Reserved	192 KB
0210_0000	0210_FFFF		CAAM	64 KB

Table 2-4 shows the DAP detailed memory map.

Table 2-4. DAP memory map

Start Address	End Address	Region	NIC Port	Size	Allocation
0216_0000	0216_0FFF	ARM Cortex A9 MPCore Platform, DAP	AIPS-2, Global Map	4 KB	Platform Control
0215_F000	0215_FFFF			4 KB	PTM3
0215_E000	0215_EFFF			4 KB	PTM2
0215_D000	0215_DFFF			4 KB	PTM1
0215_C000	0215_CFFF			4 KB	PTM0
0215_B000	0215_BFFF			4 KB	CTI3
0215_A000	0215_AFFF			4 KB	CTI2
0215_9000	0215_9FFF			4 KB	CTI1
0215_8000	0215_8FFF			4 KB	CTI0

Table continues on the next page...

Table 2-4. DAP memory map (continued)

Start Address	End Address	Region	NIC Port	Size	Allocation
0215_7000	0215_7FFF			4 KB	CPU3 PMU
0215_6000	0215_6FFF			4 KB	CPU3 Debug i/f
0215_5000	0215_5FFF			4 KB	CPU2 PMU
0215_4000	0215_4FFF			4 KB	CPU2 Debug i/f
0215_3000	0215_3FFF			4 KB	CPU1 PMU
0215_2000	0215_2FFF			4 KB	CPU1 Debug i/f
0215_1000	0215_1FFF			4 KB	CPU0 PMU
0215_0000	0215_0FFF			4 KB	CPU0 Debug i/f
0214_F000	0214_FFFF			4 KB	CA9-INTEG
0214_5000	0214_EFFF			40 KB	Reserved
0214_4000	0214_4FFF			4 KB	FUNNEL
0214_3000	0214_3FFF			4 KB	TPIU
0214_2000	0214_2FFF			4 KB	ext. CTI
0214_1000	0214_1FFF			4 KB	ETB
0214_0000	0214_0FFF			4 KB	DAP ROM Table

NOTE

User should not address reserved memory regions. Access to reserved memory regions can cause unpredictable behavior.

2.3 DDR mapping to MMDC controller ports

The following table lists the various DDR configuration modes and how each is mapped to the DDR channels.

Memory Mapping Mode	DDR Memory Map Config[1:0] fuse value	Start Address	End Address	Total Size	MMDC channel / ports
X32 / X64 bit, fixed mapping	‘00’	1000_0000	FFFF_FFFF	3840MB	MMDC #0
Dual channel (2x 32-bit), Fixed mapping (LPDDR2)	‘01’	8000_0000	FFFF_FFFF	2048MB	MMDC #0
		1000_0000	7FFF_FFFF	1792MB	MMDC #1
Dual channel (2x 32-bit), Interleaved mapping (LPDDR2) [1]	‘10’	1000_0000	FFFF_FFFF	3840MB	MMDC #0 , MMDC #1 (interleaved data)
Illegal	'11'				

1. In the 4KB interleaving mode the system bus maps each consecutive 4KB region to a 4KB region in the other MMDC port, such that "even" 4KB spaces are mapped to MMDC0, and "odd" 4KB regions mapped to MMDC1.

2.4 DMA memory map

The Smart DMA memory map can be found in the following table.

Table 2-6. SDMA peripheral memory map

Peripheral	Base address	Size
Reserved for SDMA internal memory	0x0000	4KB
SPDIF	0x1000	4KB
eCSPI1	0x2000	4KB
eCSPI2	0x3000	4KB
eCSPI3	0x4000	4KB
eCSPI4	0x5000	4KB
eCSPI5	0x6000	4KB
Reserved for SDMA internal registers	0x7000	4KB
UART1	0x8000	4KB
ESAI	0x9000	4KB
SSI1	0xA000	4KB
SSI2	0xB000	4KB
SSI3	0xC000	4KB
ASRC	0xD000	4KB
Reserved	0xE000	4KB
SPBA Registers	0xF000	4KB

NOTE

User should not address reserved memory regions. Access to reserved memory regions can cause unpredictable behavior.

Chapter 3
Interrupts and DMA Events

3.1 Overview

This chapter discusses the assignments of interrupts from the ARM domain in A9 interrupts and from DMA events in SDMA event mapping

3.2 A9 interrupts

The Global Interrupt Controller (GIC) collects up to 128 interrupt requests from all i.MX 6Dual/6Quad sources and provides an interface to each of the CPU cores.

The first 32 interrupts are used for interrupts that are private to the CPUs interface. These interrupts are not included in the table below. All interrupts besides the private CPU are also hooked up to the GPC in the same order.

Each interrupt can be configured as a normal or a secure interrupt. Software force registers and software priority masking are also supported. The following table describes the ARM interrupt sources.

Table 3-1. ARM domain interrupt summary

IRQ	Interrupt Source	Interrupt Description
32	IOMUXC	General Purpose Register 1 from IOMUXC. Used to notify cores on exception condition while boot.
33	DAP	Debug Access Port interrupt request.
34	SDMA	SDMA interrupt request from all channels.
35	VPU	JPEG codec interrupt request.
36	SNVS	PMIC power off request.
37	IPU	IPU error interrupt request.
38	IPU1	IPU1 sync interrupt request.

Table continues on the next page...

Table 3-1. ARM domain interrupt summary (continued)

IRQ	Interrupt Source	Interrupt Description
39	IPU2	IPU2 error interrupt request.
40	IPU2	IPU2 sync interrupt request.
41	GPU3D	GPU3D interrupt request.
42	R2D GPU2D	R2D GPU2D general interrupt request.
43	V2D GPU2D	V2D GPU2D(OpenVG) general interrupt request.
44	VPU	VPU interrupt request.
45	APBH-Bridge-DMA	Logical OR of APBH-Bridge-DMA channels 0-3 completion and error interrupts.
46	EIM	EIM interrupt request.
47	BCH	BCH operation complete interrupt.
48	GPMI	GPMI operation timeout error interrupt.
49	DTCP	DTCP interrupt request.
50	VDOA	Logical OR of VDOA interrupt requests.
51	SNVS	SNVS consolidated interrupt.
52	SNVS	SNVS security interrupt.
53	CSU	CSU interrupt request 1. Indicates to the processor that one or more alarm inputs were asserted.
54	uSDHC1	uSDHC1 (Enhanced SDHC) interrupt request.
55	uSDHC2	uSDHC2 (Enhanced SDHC) interrupt request.
56	uSDHC3	uSDHC3 (Enhanced SDHC) interrupt request.
57	uSDHC4	uSDHC4 (Enhanced SDHC) interrupt request.
58	UART1	UART1 interrupt request.
59	UART2	UART2 interrupt request.
60	UART3	UART3 interrupt request.
61	UART4	UART4 interrupt request.
62	UART5	UART5 interrupt request.
63	eCSPI1	eCSPI1 interrupt request.
64	eCSPI2	eCSPI2 interrupt request.
65	eCSPI3	eCSPI3 interrupt request.
66	eCSPI4	eCSPI4 interrupt request.
67	eCSPI5	eCSPI5 interrupt request.
68	I2C1	I2C1 interrupt request.
69	I2C2	I2C2 interrupt request.
70	I2C3	I2C3 interrupt request.
71	SATA	SATA interrupt request.
72	USB	USB Host 1 interrupt request.
73	USB	USB Host 2 interrupt request.
74	USB	USB Host 3 interrupt request.
75	USB	USB OTG interrupt request.
76	USB_PHY	UTMI0 interrupt request.

Table continues on the next page...

Table 3-1. ARM domain interrupt summary (continued)

IRQ	Interrupt Source	Interrupt Description
77	USB_PHY	UTMI1 interrupt request.
78	SSI1	SSI1 interrupt request.
79	SSI2	SSI2 interrupt request.
80	SSI3	SSI3 interrupt request.
81	Reserved	Reserved
82	ASRC	ASRC interrupt request.
83	ESAI	ESAI interrupt request.
84	SPDIF	SPDIF interrupt.
85	MLB150	MLB error interrupt request.
86	PMU	Brown out of 1.1, 2.5 and 3.0 analog regulators occurred.
87	GPT	Logical OR of GPT rollover interrupt line, input capture 1 & 2 lines, output compare 1, 2 & 3 interrupt lines.
88	EPIT1	EPIT1 output compare interrupt.
89	EPIT2	EPIT2 output compare interrupt.
90	GPIO1	INT7 interrupt request.
91	GPIO1	INT6 interrupt request.
92	GPIO1	INT5 interrupt request.
93	GPIO1	INT4 interrupt request.
94	GPIO1	INT3 interrupt request.
95	GPIO1	INT2 interrupt request.
96	GPIO1	INT1 interrupt request.
97	GPIO1	INT0 interrupt request.
98	GPIO1	Combined interrupt indication for GPIO1 signals 0 - 15.
99	GPIO1	Combined interrupt indication for GPIO1 signals 16 - 31.
100	GPIO2	Combined interrupt indication for GPIO2 signals 0 - 15.
101	GPIO2	Combined interrupt indication for GPIO2 signals 16 - 31.
102	GPIO3	Combined interrupt indication for GPIO3 signals 0 - 15.
103	GPIO3	Combined interrupt indication for GPIO3 signals 16 - 31.
104	GPIO4	Combined interrupt indication for GPIO4 signals 0 - 15.
105	GPIO4	Combined interrupt indication for GPIO4 signals 16 - 31.
106	GPIO5	Combined interrupt indication for GPIO5 signals 0 - 15.
107	GPIO5	Combined interrupt indication for GPIO5 signals 16 - 31.
108	GPIO6	Combined interrupt indication for GPIO6 signals 0 - 15.
109	GPIO6	Combined interrupt indication for GPIO6 signals 16 - 31.
110	GPIO7	Combined interrupt indication for GPIO7 signals 0 - 15.
111	GPIO7	Combined interrupt indication for GPIO7 signals 16 - 31.
112	WDOG1	WDOG1 timer reset interrupt request.
113	WDOG2	WDOG2 timer reset interrupt request.
114	KPP	Key Pad interrupt request.

Table continues on the next page...

Table 3-1. ARM domain interrupt summary (continued)

IRQ	Interrupt Source	Interrupt Description
115	PWM1	Cumulative interrupt line for PWM1. Logical OR of rollover, compare, and FIFO waterlevel crossing interrupts.
116	PWM2	Cumulative interrupt line for PWM2. Logical OR of rollover, compare, and FIFO waterlevel crossing interrupts.
117	PWM3	Cumulative interrupt line for PWM3. Logical OR of rollover, compare, and FIFO waterlevel crossing interrupts.
118	PWM4	Cumulative interrupt line for PWM4. Logical OR of rollover, compare, and FIFO waterlevel crossing interrupts.
119	CCM	CCM interrupt request 1.
120	CCM	CCM interrupt request 2.
121	GPC	GPC interrupt request 1.
122	Reserved	Reserved
123	SRC	SRC interrupt request.
124	CPU	L2 interrupt request.
125	CPU	Parity Check error interrupt request.
126	CPU	Performance Unit interrupt.
127	CPU	CTI trigger outputs interrupt.
128	SRC	Combined CPU wdog interrupts (4x) out of SRC.
129	Reserved	Reserved.
130	Reserved	Reserved
131	Reserved	Reserved
132	MIPI_CSI	CSI interrupt request 1.
133	MIPI_CSI	CSI interrupt request 2.
134	MIPI_DSI	DSI interrupt request.
135	MIPI_HSI	HSI interrupt request.
136	SJC	SJC interrupt from General Purpose register.
137	CAAM	CAAM job ring 0 interrupt.
138	CAAM	CAAM job ring 1 interrupt.
139	Reserved	Reserved
140	ASC1	ASC1 interrupt request.
141	ASC2	ASC2 interrupt request.
142	FLEXCAN1	FLEXCAN1 combined interrupt. Logical OR of ini_int_busoff, ini_int_error, ipi_int_mbor, ipi_int_rxwarning, ipi_int_txwarning and ipi_int_wakein.
143	FLEXCAN2	FLEXCAN2 combined interrupt. Logical OR of ini_int_busoff, ini_int_error, ipi_int_mbor, ipi_int_rxwarning, ipi_int_txwarning and ipi_int_wakein.
144	Reserved	Reserved
145	Reserved	Reserved
146	Reserved	Reserved
147	HDMI	HDMI master interrupt request.
148	HDMI	HDMI CEC engine dedicated interrupt signal raised by a wake-up event.

Table continues on the next page...

Table 3-1. ARM domain interrupt summary (continued)

IRQ	Interrupt Source	Interrupt Description
149	MLB150	Channels [31:0] interrupt requests. Channels [63:32] interrupt requests are available on IRQ #158, unless the MLB150_ACTL[SMX] bit is set, in which case those channels are muxed into this IRQ.
150	ENET	MAC 0 IRQ, Logical OR of: MAC 0 Periodic Timer Overflow MAC 0 Time Stamp Available MAC 0 Time Stamp Available MAC 0 Time Stamp Available MAC 0 Payload Receive Error MAC 0 Transmit FIFO Underrun MAC 0 Collision Retry Limit MAC 0 Late Collision MAC 0 Ethernet Bus Error MAC 0 MII Data Transfer Done MAC 0 Receive Buffer Done MAC 0 Receive Frame Done MAC 0 Transmit Buffer Done MAC 0 Transmit Frame Done MAC 0 Graceful Stop MAC 0 Babbling Transmit Error MAC 0 Babbling Receive Error MAC 0 Wakeup Request [synchronous]
151	ENET	MAC 0 1588 Timer interrupt [synchronous] request.
152	PCIe	PCIe interrupt request 1 (intd/msi_ctrl_int).
153	PCIe	PCIe interrupt request 2 (intc).
154	PCIe	PCIe interrupt request 3 (intb).
155	PCIe	PCIe interrupt request 4 (inta).
156	DCIC1	Logical OR of DCIC1 interrupt requests.
157	DCIC2	Logical OR of DCIC2 interrupt requests.
158	MLB150	Logical OR of channel[63:32] interrupt requests.
159	PMU	Brown out of core, gpu, and chip digital regulators occurred.

3.3 SDMA event mapping

The following table shows the DMA request signals for peripherals in i.MX 6Dual/ 6Quad.

Table 3-2. SDMA event mapping

Event Number	DMA Source	Description
0	VPU	VPU DMA request
1	IPU2	IPU2 DMA event.
2	IPU1 / HDMI audio done	IPU1 DMA Event; Muxed with HDMI audio done event, controlled by GPR0[0], IPU1 event - default.
3	eCSPI1 / I2C3	eCSPI1 Rx request; Muxed with I2C3 controlled by IOMUXC register GPR0[1].
4	eCSPI1 / I2C2	eCSPI1 Tx request; Muxed with I2C2 controlled by IOMUXC register GPR0[2].
5	eCSPI2 / I2C1	eCSPI2 Rx request; Muxed with I2C1 controlled by IOMUXC register GPR0[3].
6	eCSPI2	eCSPI2 Tx request
7	eCSPI3	eCSPI3 Rx request
8	eCSPI3	eCSPI3 Tx request
9	eCSPI4 / EPIT2	eCSPI4 Rx request; Muxed with EPIT2 DMA request controlled by IOMUXC register GPR0[5].
10	eCSPI4/ I2C1	eCSPI4 Tx request; Muxed with I2C1 controlled by IOMUXC register GPR0[4].
11	eCSPI5	eCSPI5 Rx request
12	eCSPI5	
13	GPT	GPT counter event
14	SPDIF / IOMUX	SPDIFRX DMA request; Muxed with external DMA pad #2 controlled by IOMUXC register GPR0[7].
15	SPDIF	SPDIF TX DMA request
16	EPIT1	EPIT1 request.
17	ASRC	ASRC DMA1 request (Pair A input Request)
18	ASRC	ASRC DMA2 request (Pair B input Request)
19	ASRC	ASRC DMA3 request (Pair C input Request)
20	ASRC	ASRC DMA4 request (Pair A output Request)
21	ASRC	ASRC DMA5 request (Pair B output Request)
22	ASRC	ASRC DMA6 request (Pair C output Request)
23	ESAI / I2C3	ESAI Rx FIFO DMA request; Muxed with I2C3 controlled by IOMUXC register GPR0[6].
24	ESAI	ESAI Tx FIFO DMA request
25	UART1	UART1 Rx FIFO
26	UART1	UART1 Tx FIFO
27	UART2	UART2 Rx FIFO
28	UART2	UART2 Tx FIFO
29	UART3 / QSPI1	UART3 Rx FIFO; Muxed with QSPI1 DMA Tx request controlled by IOMUXC register GPR0[21]
30	UART3 / QSPI2	UART3 Tx FIFO; Muxed with QSPI2 DMA Tx request controlled by IOMUXC register GPR0[22]

Table continues on the next page...

Table 3-2. SDMA event mapping (continued)

Event Number	DMA Source	Description
31	UART4	UART4 Rx FIFO
32	UART4	UART4 Tx FIFO
33	UART5	UART5 Rx FIFO
34	UART5	UART5 Tx FIFO
35	SSI1	SSI1 receive 1 DMA request
36	SSI1	SSI1 transmit 1 DMA request
37	SSI1	SSI1 receive 0 DMA request
38	SSI1	SSI1 transmit 0 DMA request
39	SSI2	SSI2 receive 1 DMA request
40	SSI2	SSI2 transmit 1 DMA request
41	SSI2	SSI2 receive 0 DMA request
42	SSI2	SSI2 transmit 0 DMA request
43	SSI3	SSI3 receive 1 DMA request
44	SSI3	SSI3 transmit 1 DMA request
45	SSI3	SSI3 receive 0 DMA request
46	SSI3	SSI3 transmit 0 DMA request
47	Reserved	Reserved

As shown in the table, some of the events are an output of a mux of two signals or triggers. The select of this mux is controlled by the general purpose registers in IOMUXC.

Chapter 4
External Signals and Pin Multiplexing

4.1 Overview

The i.MX 6Dual/6Quad contains a limited number of pins, most of which have multiple signal options. These signal to pin and pin to signal options are selected by the input-output multiplexer called IOMUX. The IOMUX is also used to configure other pin characteristics, such as voltage level, drive strength, and hysteresis.

Pin Assignments lists the pad names of the chip, the various signals that can be assigned to each of the pads, and the default settings for each pad. Muxing Options lists the external signals grouped by module instance, the muxing options for each signal, and the registers used to route the signal to the chosen pad.

4.1.1 Pin Assignments

Table 4-1. Pin Assignments

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
BOOT_MODE0		SRC_BOOT_MODE0		
BOOT_MODE1		SRC_BOOT_MODE1		
CLK1_N		XTALOSC_CLK1_N		
CLK1_P		XTALOSC_CLK1_P		
CLK2_N		XTALOSC_CLK2_N		
CLK2_P		XTALOSC_CLK2_P		
CSI0_DAT4	ALT0	IPU1_CSI0_DATA04	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA04
	ALT1	EIM_DATA02	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_SCLK	PUE - PULL	
	ALT3	KEY_COL5	PKE - ENABLED	
	ALT4	AUD3_TXC	ODE - DISABLED	
	ALT5	GPIO5_IO22	SPEED - MEDIUM	
	ALT7	ARM_TRACE01	DSE - 40_OHM	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
CSI0_DAT5	ALT0	IPU1_CSI0_DATA05	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA05
	ALT1	EIM_DATA03	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_MOSI	PUE - PULL	
	ALT3	KEY_ROW5	PKE - ENABLED	
	ALT4	AUD3_TXD	ODE - DISABLED	
	ALT5	GPIO5_IO23	SPEED - MEDIUM	
	ALT7	ARM_TRACE02	DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT6	ALT0	IPU1_CSI0_DATA06	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA06
	ALT1	EIM_DATA04	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_MISO	PUE - PULL	
	ALT3	KEY_COL6	PKE - ENABLED	
	ALT4	AUD3_TXFS	ODE - DISABLED	
	ALT5	GPIO5_IO24	SPEED - MEDIUM	
	ALT7	ARM_TRACE03	DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT7	ALT0	IPU1_CSI0_DATA07	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA07
	ALT1	EIM_DATA05	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_SS0	PUE - PULL	
	ALT3	KEY_ROW6	PKE - ENABLED	
	ALT4	AUD3_RXD	ODE - DISABLED	
	ALT5	GPIO5_IO25	SPEED - MEDIUM	
	ALT7	ARM_TRACE04	DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT8	ALT0	IPU1_CSI0_DATA08	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA08
	ALT1	EIM_DATA06	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_SCLK	PUE - PULL	
	ALT3	KEY_COL7	PKE - ENABLED	
	ALT4	I2C1_SDA	ODE - DISABLED	
	ALT5	GPIO5_IO26	SPEED - MEDIUM	
	ALT7	ARM_TRACE05	DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT9	ALT0	IPU1_CSI0_DATA09	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA09
	ALT1	EIM_DATA07	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_MOSI	PUE - PULL	
	ALT3	KEY_ROW7	PKE - ENABLED	
	ALT4	I2C1_SCL	ODE - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO5_IO27	SPEED - MEDIUM	
	ALT7	ARM_TRACE06	DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT10	ALT0	IPU1_CSI0_DATA10	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA10
	ALT1	AUD3_RXC	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_MISO	PUE - PULL	
	ALT3	UART1_TX_DATA	PKE - ENABLED	
	ALT5	GPIO5_IO28	ODE - DISABLED	
	ALT7	ARM_TRACE07	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT11	ALT0	IPU1_CSI0_DATA11	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA11
	ALT1	AUD3_RXFS	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_SS0	PUE - PULL	
	ALT3	UART1_RX_DATA	PKE - ENABLED	
	ALT5	GPIO5_IO29	ODE - DISABLED	
	ALT7	ARM_TRACE08	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT12	ALT0	IPU1_CSI0_DATA12	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA12
	ALT1	EIM_DATA08	PUS - 100K_OHM_PU	
	ALT3	UART4_TX_DATA	PUE - PULL	
	ALT5	GPIO5_IO30	PKE - ENABLED	
	ALT7	ARM_TRACE09	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT13	ALT0	IPU1_CSI0_DATA13	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA13
	ALT1	EIM_DATA09	PUS - 100K_OHM_PU	
	ALT3	UART4_RX_DATA	PUE - PULL	
	ALT5	GPIO5_IO31	PKE - ENABLED	
	ALT7	ARM_TRACE10	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
CSI0_DAT14	ALT0	IPU1_CSI0_DATA14	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA14
	ALT1	EIM_DATA10	PUS - 100K_OHM_PU	
	ALT3	UART5_TX_DATA	PUE - PULL	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO6_IO00	PKE - ENABLED	
	ALT7	ARM_TRACE11	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
CSI0_DAT15	ALT0	IPU1_CSI0_DATA15	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA15
	ALT1	EIM_DATA11	PUS - 100K_OHM_PU	
	ALT3	UART5_RX_DATA	PUE - PULL	
	ALT5	GPIO6_IO01	PKE - ENABLED	
	ALT7	ARM_TRACE12	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
CSI0_DAT16	ALT0	IPU1_CSI0_DATA16	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA16
	ALT1	EIM_DATA12	PUS - 100K_OHM_PU	
	ALT3	UART4_RTS_B	PUE - PULL	
	ALT5	GPIO6_IO02	PKE - ENABLED	
	ALT7	ARM_TRACE13	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
CSI0_DAT17	ALT0	IPU1_CSI0_DATA17	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA17
	ALT1	EIM_DATA13	PUS - 100K_OHM_PU	
	ALT3	UART4_CTS_B	PUE - PULL	
	ALT5	GPIO6_IO03	PKE - ENABLED	
	ALT7	ARM_TRACE14	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
CSI0_DAT18	ALT0	IPU1_CSI0_DATA18	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA18
	ALT1	EIM_DATA14	PUS - 100K_OHM_PU	
	ALT3	UART5_RTS_B	PUE - PULL	
	ALT5	GPIO6_IO04	PKE - ENABLED	
	ALT7	ARM_TRACE15	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
CSI0_DAT19	ALT0	IPU1_CSI0_DATA19	HYS - ENABLED	SW_PAD_CTL_PAD_CSI0_DATA19

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	EIM_DATA15	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT3	UART5_CTS_B		
	ALT5	GPIO6_IO05		
CSI0_DATA_EN	ALT0	IPU1_CSI0_DATA_EN	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_CSI0_DATA_EN
	ALT1	EIM_DATA00		
	ALT5	GPIO5_IO20		
	ALT7	ARM_TRACE_CLK		
CSI0_MCLK	ALT0	IPU1_CSI0_HSYNC	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_CSI0_HSYNC
	ALT3	CCM_CLKO1		
	ALT5	GPIO5_IO19		
	ALT7	ARM_TRACE_CTL		
CSI0_PIXCLK	ALT0	IPU1_CSI0_PIXCLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_CSI0_PIXCLK
	ALT5	GPIO5_IO18		
	ALT7	ARM_EVENTO		
CSI0_VSYNC	ALT0	IPU1_CSI0_VSYNC	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_CSI0_VSYNC
	ALT1	EIM_DATA01		
	ALT5	GPIO5_IO21		
	ALT7	ARM_TRACE00		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
CSI_CLK0M		CSI_CLK0_N		
CSI_CLK0P		CSI_CLK0_P		
CSI_D0M		CSI_DATA0_N		
CSI_D0P		CSI_DATA0_P		
CSI_D1M		CSI_DATA1_N		
CSI_D1P		CSI_DATA1_P		
CSI_D2M		CSI_DATA2_N		
CSI_D2P		CSI_DATA2_P		
CSI_D3M		CSI_DATA3_N		
CSI_D3P		CSI_DATA3_P		
DI0_DISP_CLK	ALT0	IPU1_DI0_DISP_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DI0_DISP_CLK
	ALT1	IPU2_DI0_DISP_CLK		
	ALT5	GPIO4_IO16		
DI0_PIN2	ALT0	IPU1_DI0_PIN02	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DI0_PIN02
	ALT1	IPU2_DI0_PIN02		
	ALT2	AUD6_TXD		
	ALT5	GPIO4_IO18		
DI0_PIN3	ALT0	IPU1_DI0_PIN03	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DI0_PIN03
	ALT1	IPU2_DI0_PIN03		
	ALT2	AUD6_TXFS		
	ALT5	GPIO4_IO19		
DI0_PIN4	ALT0	IPU1_DI0_PIN04	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_DI0_PIN04
	ALT1	IPU2_DI0_PIN04		
	ALT2	AUD6_RXD		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT3	SD1_WP	PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT5	GPIO4_IO20		
DI0_PIN15	ALT0	IPU1_DI0_PIN15	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DI0_PIN15
	ALT1	IPU2_DI0_PIN15		
	ALT2	AUD6_TXC		
	ALT5	GPIO4_IO17		
DISP0_DAT0	ALT0	IPU1_DISP0_DATA00	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA00
	ALT1	IPU2_DISP0_DATA00		
	ALT2	ECSPI3_SCLK		
	ALT5	GPIO4_IO21		
DISP0_DAT1	ALT0	IPU1_DISP0_DATA01	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA01
	ALT1	IPU2_DISP0_DATA01		
	ALT2	ECSPI3_MOSI		
	ALT5	GPIO4_IO22		
DISP0_DAT2	ALT0	IPU1_DISP0_DATA02	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA02
	ALT1	IPU2_DISP0_DATA02		
	ALT2	ECSPI3_MISO		
	ALT5	GPIO4_IO23		
DISP0_DAT3	ALT0	IPU1_DISP0_DATA03	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA03

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	IPU2_DISP0_DATA03	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT2	ECSPI3_SS0		
	ALT5	GPIO4_IO24		
DISP0_DAT4	ALT0	IPU1_DISP0_DATA04	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA04
	ALT1	IPU2_DISP0_DATA04		
	ALT2	ECSPI3_SS1		
	ALT5	GPIO4_IO25		
DISP0_DAT5	ALT0	IPU1_DISP0_DATA05	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA05
	ALT1	IPU2_DISP0_DATA05		
	ALT2	ECSPI3_SS2		
	ALT3	AUD6_RXFS		
	ALT5	GPIO4_IO26		
DISP0_DAT6	ALT0	IPU1_DISP0_DATA06	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA06
	ALT1	IPU2_DISP0_DATA06		
	ALT2	ECSPI3_SS3		
	ALT3	AUD6_RXC		
	ALT5	GPIO4_IO27		
DISP0_DAT7	ALT0	IPU1_DISP0_DATA07	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_DISP0_DATA07
	ALT1	IPU2_DISP0_DATA07		
	ALT2	ECSPI3_RDY		
	ALT5	GPIO4_IO28		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
DISP0_DAT8	ALT0	IPU1_DISP0_DATA08	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA08
	ALT1	IPU2_DISP0_DATA08		
	ALT2	PWM1_OUT		
	ALT3	WDOG1_B		
	ALT5	GPIO4_IO29		
DISP0_DAT9	ALT0	IPU1_DISP0_DATA09	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA09
	ALT1	IPU2_DISP0_DATA09		
	ALT2	PWM2_OUT		
	ALT3	WDOG2_B		
	ALT5	GPIO4_IO30		
DISP0_DAT10	ALT0	IPU1_DISP0_DATA10	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA10
	ALT1	IPU2_DISP0_DATA10		
	ALT5	GPIO4_IO31		
DISP0_DAT11	ALT0	IPU1_DISP0_DATA11	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA11
	ALT1	IPU2_DISP0_DATA11		
	ALT5	GPIO5_IO05		
DISP0_DAT12	ALT0	IPU1_DISP0_DATA12	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED	SW_PAD_CTL_PAD_DISP0_DATA12
	ALT1	IPU2_DISP0_DATA12		
	ALT5	GPIO5_IO06		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
DISP0_DAT13	ALT0	IPU1_DISP0_DATA13	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA13
	ALT1	IPU2_DISP0_DATA13		
	ALT3	AUD5_RXFS		
	ALT5	GPIO5_IO07		
DISP0_DAT14	ALT0	IPU1_DISP0_DATA14	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA14
	ALT1	IPU2_DISP0_DATA14		
	ALT3	AUD5_RXC		
	ALT5	GPIO5_IO08		
DISP0_DAT15	ALT0	IPU1_DISP0_DATA15	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA15
	ALT1	IPU2_DISP0_DATA15		
	ALT2	ECSPI1_SS1		
	ALT3	ECSPI2_SS1		
	ALT5	GPIO5_IO09		
DISP0_DAT16	ALT0	IPU1_DISP0_DATA16	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_DISP0_DATA16
	ALT1	IPU2_DISP0_DATA16		
	ALT2	ECSPI2_MOSI		
	ALT3	AUD5_TXC		
	ALT4	SDMA_EXT_EVENT0		
	ALT5	GPIO5_IO10		
DISP0_DAT17	ALT0	IPU1_DISP0_DATA17	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_DISP0_DATA17
	ALT1	IPU2_DISP0_DATA17		
	ALT2	ECSPI2_MISO		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT3	AUD5_TXD	PKE - ENABLED	
	ALT4	SDMA_EXT_EVENT1	ODE - DISABLED	
	ALT5	GPIO5_IO11	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT18	ALT0	IPU1_DISP0_DATA18	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA18
	ALT1	IPU2_DISP0_DATA18	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_SS0	PUE - PULL	
	ALT3	AUD5_TXFS	PKE - ENABLED	
	ALT4	AUD4_RXFS	ODE - DISABLED	
	ALT5	GPIO5_IO12	SPEED - MEDIUM	
	ALT7	EIM_CS2_B	DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT19	ALT0	IPU1_DISP0_DATA19	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA19
	ALT1	IPU2_DISP0_DATA19	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_SCLK	PUE - PULL	
	ALT3	AUD5_RXD	PKE - ENABLED	
	ALT4	AUD4_RXC	ODE - DISABLED	
	ALT5	GPIO5_IO13	SPEED - MEDIUM	
	ALT7	EIM_CS3_B	DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT20	ALT0	IPU1_DISP0_DATA20	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA20
	ALT1	IPU2_DISP0_DATA20	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_SCLK	PUE - PULL	
	ALT3	AUD4_TXC	PKE - ENABLED	
	ALT5	GPIO5_IO14	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT21	ALT0	IPU1_DISP0_DATA21	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA21
	ALT1	IPU2_DISP0_DATA21	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_MOSI	PUE - PULL	
	ALT3	AUD4_TXD	PKE - ENABLED	
	ALT5	GPIO5_IO15	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT22	ALT0	IPU1_DISP0_DATA22	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA22

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	IPU2_DISP0_DATA22	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_MISO	PUE - PULL	
	ALT3	AUD4_TXFS	PKE - ENABLED	
	ALT5	GPIO5_IO16	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
DISP0_DAT23	ALT0	IPU1_DISP0_DATA23	HYS - ENABLED	SW_PAD_CTL_PAD_DISP0_DATA23
	ALT1	IPU2_DISP0_DATA23	PUS - 100K_OHM_PU	
	ALT2	ECSPI1_SS0	PUE - PULL	
	ALT3	AUD4_RXD	PKE - ENABLED	
	ALT5	GPIO5_IO17	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
DRAM_A0		DRAM_ADDR00	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR00
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A1		DRAM_ADDR01	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR01
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A2		DRAM_ADDR02	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR02
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A3		DRAM_ADDR03	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR03
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A4		DRAM_ADDR04	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR04
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A5		DRAM_ADDR05	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR05
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A6		DRAM_ADDR06	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR06
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A7		DRAM_ADDR07	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR07
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A8		DRAM_ADDR08	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR08
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A9		DRAM_ADDR09	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR09
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A10		DRAM_ADDR10	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR10
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A11		DRAM_ADDR11	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR11
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A12		DRAM_ADDR12	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR12

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A13		DRAM_ADDR13	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR13
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A14		DRAM_ADDR14	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR14
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_A15		DRAM_ADDR15	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_ADDR15
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_ADDDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_CAS		DRAM_CAS_B	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_CAS_B
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_CS0		DRAM_CS0_B	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_CS0_B
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_CTLDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_CS1		DRAM_CS1_B	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_CS1_B
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_CTLDS
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_D0		DRAM_DATA00	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D1		DRAM_DATA01	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D2		DRAM_DATA02	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D3		DRAM_DATA03	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D4		DRAM_DATA04	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D5		DRAM_DATA05	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
DRAM_D6		DRAM_DATA06	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D7		DRAM_DATA07	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL0
			DSE - 40_OHM	SW_PAD_CTL_GRP_B0DS
DRAM_D8		DRAM_DATA08	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D9		DRAM_DATA09	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D10		DRAM_DATA10	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D11		DRAM_DATA11	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
				SW_PAD_CTL_GRP_DDRHYS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRPK
			PUE - PULL	SW_PAD_CTL_GRP_DDRPKE
			PKE - ENABLED	SW_PAD_CTL_GRP_TERM_CTL1
			ODT - DISABLED	SW_PAD_CTL_GRP_B1DS
			DSE - 40_OHM	
DRAM_D12		DRAM_DATA12	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D13		DRAM_DATA13	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D14		DRAM_DATA14	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D15		DRAM_DATA15	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL1
			DSE - 40_OHM	SW_PAD_CTL_GRP_B1DS
DRAM_D16		DRAM_DATA16	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D17		DRAM_DATA17	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D18		DRAM_DATA18	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D19		DRAM_DATA19	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D20		DRAM_DATA20	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D21		DRAM_DATA21	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D22		DRAM_DATA22	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D23		DRAM_DATA23	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL2
			DSE - 40_OHM	SW_PAD_CTL_GRP_B2DS
DRAM_D24		DRAM_DATA24	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D25		DRAM_DATA25	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D26		DRAM_DATA26	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D27		DRAM_DATA27	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D28		DRAM_DATA28	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D29		DRAM_DATA29	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D30		DRAM_DATA30	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D31		DRAM_DATA31	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL3
			DSE - 40_OHM	SW_PAD_CTL_GRP_B3DS
DRAM_D32		DRAM_DATA32	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
DRAM_D33		DRAM_DATA33	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D34		DRAM_DATA34	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D35		DRAM_DATA35	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D36		DRAM_DATA36	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D37		DRAM_DATA37	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D38		DRAM_DATA38	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
				SW_PAD_CTL_GRP_DDRHYS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRPK
			PUE - PULL	SW_PAD_CTL_GRP_DDRPKE
			PKE - ENABLED	SW_PAD_CTL_GRP_TERM_CTL4
			ODT - DISABLED	SW_PAD_CTL_GRP_B4DS
			DSE - 40_OHM	
DRAM_D39		DRAM_DATA39	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL4
			DSE - 40_OHM	SW_PAD_CTL_GRP_B4DS
DRAM_D40		DRAM_DATA40	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D41		DRAM_DATA41	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D42		DRAM_DATA42	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D43		DRAM_DATA43	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D44		DRAM_DATA44	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D45		DRAM_DATA45	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D46		DRAM_DATA46	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D47		DRAM_DATA47	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL5
			DSE - 40_OHM	SW_PAD_CTL_GRP_B5DS
DRAM_D48		DRAM_DATA48	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D49		DRAM_DATA49	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D50		DRAM_DATA50	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D51		DRAM_DATA51	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D52		DRAM_DATA52	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D53		DRAM_DATA53	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D54		DRAM_DATA54	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D55		DRAM_DATA55	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL6
			DSE - 40_OHM	SW_PAD_CTL_GRP_B6DS
DRAM_D56		DRAM_DATA56	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D57		DRAM_DATA57	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D58		DRAM_DATA58	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D59		DRAM_DATA59	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
DRAM_D60		DRAM_DATA60	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D61		DRAM_DATA61	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D62		DRAM_DATA62	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_D63		DRAM_DATA63	DDR_SEL - LPDDR2	SW_PAD_CTL_GRP_DDR_TYPE
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDRMODE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	SW_PAD_CTL_GRP_DDRPK
			PKE - ENABLED	SW_PAD_CTL_GRP_DDRPKE
			ODT - DISABLED	SW_PAD_CTL_GRP_TERM_CTL7
			DSE - 40_OHM	SW_PAD_CTL_GRP_B7DS
DRAM_DQM0		DRAM_DQM0	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_DQM0
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_DQM1		DRAM_DQM1	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_DQM1
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	
			ODT - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DSE - 40_OHM	
DRAM_DQM2		DRAM_DQM2	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM2 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_DQM3		DRAM_DQM3	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM3 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_DQM4		DRAM_DQM4	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM4 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_DQM5		DRAM_DQM5	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM5 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_DQM6		DRAM_DQM6	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM6 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_DQM7		DRAM_DQM7	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_DQM7 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_SDODT0		DRAM_ODT0	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_ODT0 SW_PAD_CTL_GRP_DDR_TYPE
DRAM_SDODT1		DRAM_ODT1	DDR_SEL - LPDDR2 DDR_INPUT - CMOS	SW_PAD_CTL_PAD_DRAM_ODT1 SW_PAD_CTL_GRP_DDR_TYPE

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			HYS - DISABLED ODT - DISABLED DSE - 40_OHM	
DRAM_RAS		DRAM_RAS_B	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_RAS_B SW_PAD_CTL_GRP_DDR_TYPE
DRAM_RESET		DRAM_RESET	DDR_SEL - RESERVED2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_RESET
DRAM_SDBA0		DRAM_SDBA0	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDBA0 SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_ADDDS
DRAM_SDBA1		DRAM_SDBA1	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDBA1 SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_ADDDS
DRAM_SDBA2		DRAM_SDBA2	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDBA2 SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_CTLDS
DRAM_SDCKE0		DRAM_SDCKE0	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDCKE0 SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_CTLDS
DRAM_SDCKE1		DRAM_SDCKE1	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED	SW_PAD_CTL_PAD_DRAM_SDCKE1 SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_CTLDS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DSE - 40_OHM	
DRAM_SDCLK_0_B		DRAM_SDCLK0_N		
DRAM_SDCLK_0		DRAM_SDCLK0_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDCLK0_P SW_PAD_CTL_GRP_DDR_TYPE
DRAM_SDCLK_1_B		DRAM_SDCLK1_N		
DRAM_SDCLK_1		DRAM_SDCLK1_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDCLK1_P SW_PAD_CTL_GRP_DDR_TYPE
DRAM_SDQS0_B		DRAM_SDQS0_N		
DRAM_SDQS0		DRAM_SDQS0_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED PUS - 100K_OHM_PD PUE - PULL PKE - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDQS0_P SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_DDRMODE_CTL SW_PAD_CTL_GRP_DDRHYS
DRAM_SDQS1_B		DRAM_SDQS1_N		
DRAM_SDQS1		DRAM_SDQS1_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED PUS - 100K_OHM_PD PUE - PULL PKE - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDQS1_P SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_DDRMODE_CTL SW_PAD_CTL_GRP_DDRHYS
DRAM_SDQS2_B		DRAM_SDQS2_N		
DRAM_SDQS2		DRAM_SDQS2_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED PUS - 100K_OHM_PD PUE - PULL	SW_PAD_CTL_PAD_DRAM_SDQS2_P SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_DDRMODE_CTL SW_PAD_CTL_GRP_DDRHYS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			PKE - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_SDQS3_B		DRAM_SDQS3_N		
DRAM_SDQS3		DRAM_SDQS3_P	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_SDQS3_P
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRMODE_CTL
			PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	
			PKE - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_SDQS4_B		DRAM_SDQS4_N		
DRAM_SDQS4		DRAM_SDQS4_P	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_SDQS4_P
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRMODE_CTL
			PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	
			PKE - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_SDQS5_B		DRAM_SDQS5_N		
DRAM_SDQS5		DRAM_SDQS5_P	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_SDQS5_P
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRMODE_CTL
			PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	
			PKE - DISABLED	
			ODT - DISABLED	
			DSE - 40_OHM	
DRAM_SDQS6_B		DRAM_SDQS6_N		
DRAM_SDQS6		DRAM_SDQS6_P	DDR_SEL - LPDDR2	SW_PAD_CTL_PAD_DRAM_SDQS6_P
			DDR_INPUT - CMOS	SW_PAD_CTL_GRP_DDR_TYPE
			HYS - DISABLED	SW_PAD_CTL_GRP_DDRMODE_CTL
			PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_DDRHYS
			PUE - PULL	
			PKE - DISABLED	
			ODT - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DSE - 40_OHM	
DRAM_SDQS7_B		DRAM_SDQS7_N		
DRAM_SDQS7		DRAM_SDQS7_P	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED PUS - 100K_OHM_PD PUE - PULL PKE - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDQS7_P SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_DDRMODE_CTL SW_PAD_CTL_GRP_DDRHYS
DRAM_SDWE		DRAM_SDWE_B	DDR_SEL - LPDDR2 DDR_INPUT - CMOS HYS - DISABLED ODT - DISABLED DSE - 40_OHM	SW_PAD_CTL_PAD_DRAM_SDWE_B SW_PAD_CTL_GRP_DDR_TYPE SW_PAD_CTL_GRP_CTLDS
DSI_CLK0M		DSI_CLK0_N		
DSI_CLK0P		DSI_CLK0_P		
DSI_D0M		DSI_DATA0_N		
DSI_D0P		DSI_DATA0_P		
DSI_D1M		DSI_DATA1_N		
DSI_D1P		DSI_DATA1_P		
EIM_DA0	ALT0	EIM_AD00	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD00
	ALT1	IPU1_DISP1_DATA09		
	ALT2	IPU2_CSI1_DATA09		
	ALT5	GPIO3_IO00		
	ALT7	SRC_BOOT_CFG00		
EIM_DA1	ALT0	EIM_AD01	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD01
	ALT1	IPU1_DISP1_DATA08		
	ALT2	IPU2_CSI1_DATA08		
	ALT5	GPIO3_IO01		
	ALT7	SRC_BOOT_CFG01		
EIM_DA2	ALT0	EIM_AD02	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD02

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	IPU1_DISP1_DATA07	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA07	PUE - PULL	
	ALT5	GPIO3_IO02	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG02	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA3	ALT0	EIM_AD03	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD03
	ALT1	IPU1_DISP1_DATA06	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA06	PUE - PULL	
	ALT5	GPIO3_IO03	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG03	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA4	ALT0	EIM_AD04	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD04
	ALT1	IPU1_DISP1_DATA05	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA05	PUE - PULL	
	ALT5	GPIO3_IO04	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG04	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA5	ALT0	EIM_AD05	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD05
	ALT1	IPU1_DISP1_DATA04	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA04	PUE - PULL	
	ALT5	GPIO3_IO05	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG05	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA6	ALT0	EIM_AD06	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD06
	ALT1	IPU1_DISP1_DATA03	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA03	PUE - PULL	
	ALT5	GPIO3_IO06	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG06	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - FAST	
EIM_DA7	ALT0	EIM_AD07	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD07
	ALT1	IPU1_DISP1_DATA02	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA02	PUE - PULL	
	ALT5	GPIO3_IO07	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG07	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA8	ALT0	EIM_AD08	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD08
	ALT1	IPU1_DISP1_DATA01	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA01	PUE - PULL	
	ALT5	GPIO3_IO08	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG08	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA9	ALT0	EIM_AD09	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD09
	ALT1	IPU1_DISP1_DATA00	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA00	PUE - PULL	
	ALT5	GPIO3_IO09	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG09	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA10	ALT0	EIM_AD10	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD10
	ALT1	IPU1_DI1_PIN15	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA_EN	PUE - PULL	
	ALT5	GPIO3_IO10	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG10	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_DA11	ALT0	EIM_AD11	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_AD11
	ALT1	IPU1_DI1_PIN02	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_HSYNC	PUE - PULL	
	ALT5	GPIO3_IO11	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG11	ODE - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_DA12	ALT0	EIM_AD12	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD12
	ALT1	IPU1_DI1_PIN03		
	ALT2	IPU2_CSI1_VSYNC		
	ALT5	GPIO3_IO12		
	ALT7	SRC_BOOT_CFG12		
EIM_DA13	ALT0	EIM_AD13	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD13
	ALT1	IPU1_DI1_D0_CS		
	ALT5	GPIO3_IO13		
	ALT7	SRC_BOOT_CFG13		
EIM_DA14	ALT0	EIM_AD14	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD14
	ALT1	IPU1_DI1_D1_CS		
	ALT5	GPIO3_IO14		
	ALT7	SRC_BOOT_CFG14		
EIM_DA15	ALT0	EIM_AD15	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_AD15
	ALT1	IPU1_DI1_PIN01		
	ALT2	IPU1_DI1_PIN04		
	ALT5	GPIO3_IO15		
	ALT7	SRC_BOOT_CFG15		
EIM_A16	ALT0	EIM_ADDR16	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_EIM_ADDR16
	ALT1	IPU1_DI1_DISP_CLK		
	ALT2	IPU2_CSI1_PIXCLK		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO2_IO22	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG16	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_A17	ALT0	EIM_ADDR17	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_ADDR17
	ALT1	IPU1_DISP1_DATA12	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA12	PUE - PULL	
	ALT5	GPIO2_IO21	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG17	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_A18	ALT0	EIM_ADDR18	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_ADDR18
	ALT1	IPU1_DISP1_DATA13	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA13	PUE - PULL	
	ALT5	GPIO2_IO20	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG18	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_A19	ALT0	EIM_ADDR19	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_ADDR19
	ALT1	IPU1_DISP1_DATA14	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA14	PUE - PULL	
	ALT5	GPIO2_IO19	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG19	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_A20	ALT0	EIM_ADDR20	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_ADDR20
	ALT1	IPU1_DISP1_DATA15	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA15	PUE - PULL	
	ALT5	GPIO2_IO18	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG20	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_A21	ALT0	EIM_ADDR21	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_ADDR21

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	IPU1_DISP1_DATA16	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
	ALT2	IPU2_CSI1_DATA16		
	ALT5	GPIO2_IO17		
	ALT7	SRC_BOOT_CFG21		
EIM_A22	ALT0	EIM_ADDR22	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_ADDR22
	ALT1	IPU1_DISP1_DATA17		
	ALT2	IPU2_CSI1_DATA17		
	ALT5	GPIO2_IO16		
	ALT7	SRC_BOOT_CFG22		
EIM_A23	ALT0	EIM_ADDR23	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_ADDR23
	ALT1	IPU1_DISP1_DATA18		
	ALT2	IPU2_CSI1_DATA18		
	ALT3	IPU2_SISG3		
	ALT4	IPU1_SISG3		
	ALT5	GPIO6_IO06		
	ALT7	SRC_BOOT_CFG23		
EIM_A24	ALT0	EIM_ADDR24	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_ADDR24
	ALT1	IPU1_DISP1_DATA19		
	ALT2	IPU2_CSI1_DATA19		
	ALT3	IPU2_SISG2		
	ALT4	IPU1_SISG2		
	ALT5	GPIO5_IO04		
	ALT7	SRC_BOOT_CFG24		
EIM_A25	ALT0	EIM_ADDR25	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_EIM_ADDR25
	ALT1	ECSPI4_SS1		
	ALT2	ECSPI2_RDY		
	ALT3	IPU1_DI1_PIN12		
	ALT4	IPU1_DI0_D1_CS		
	ALT5	GPIO5_IO02		
	ALT6	HDMI_TX_CEC_LINE		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - FAST	
EIM_BCLK	ALT0	EIM_BCLK	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_BCLK
	ALT1	IPU1_DI1_PIN16		
	ALT5	GPIO6_IO31		
EIM_CS0	ALT0	EIM_CS0_B	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_CS0_B
	ALT1	IPU1_DI1_PIN05		
	ALT2	ECSPI2_SCLK		
	ALT5	GPIO2_IO23		
EIM_CS1	ALT0	EIM_CS1_B	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_CS1_B
	ALT1	IPU1_DI1_PIN06		
	ALT2	ECSPI2_MOSI		
	ALT5	GPIO2_IO24		
EIM_D16	ALT0	EIM_DATA16	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_EIM_DATA16
	ALT1	ECSPI1_SCLK		
	ALT2	IPU1_DI0_PIN05		
	ALT3	IPU2_CSI1_DATA18		
	ALT4	HDMI_TX_DDC_SDA		
	ALT5	GPIO3_IO16		
	ALT6	I2C2_SDA		
EIM_D17	ALT0	EIM_DATA17	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED	SW_PAD_CTL_PAD_EIM_DATA17
	ALT1	ECSPI1_MISO		
	ALT2	IPU1_DI0_PIN06		
	ALT3	IPU2_CSI1_PIXCLK		
	ALT4	DCIC1_OUT		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO3_IO17	SPEED - MEDIUM	
	ALT6	I2C3_SCL	DSE - 40_OHM	
			SRE - SLOW	
EIM_D18	ALT0	EIM_DATA18	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA18
	ALT1	ECSPI1_MOSI	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN07	PUE - PULL	
	ALT3	IPU2_CSI1_DATA17	PKE - ENABLED	
	ALT4	IPU1_DI1_D0_CS	ODE - DISABLED	
	ALT5	GPIO3_IO18	SPEED - MEDIUM	
	ALT6	I2C3_SDA	DSE - 40_OHM	
			SRE - SLOW	
EIM_D19	ALT0	EIM_DATA19	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA19
	ALT1	ECSPI1_SS1	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN08	PUE - PULL	
	ALT3	IPU2_CSI1_DATA16	PKE - ENABLED	
	ALT4	UART1_CTS_B	ODE - DISABLED	
	ALT5	GPIO3_IO19	SPEED - MEDIUM	
	ALT6	EPIT1_OUT	DSE - 40_OHM	
			SRE - SLOW	
EIM_D20	ALT0	EIM_DATA20	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA20
	ALT1	ECSPI4_SS0	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN16	PUE - PULL	
	ALT3	IPU2_CSI1_DATA15	PKE - ENABLED	
	ALT4	UART1_RTS_B	ODE - DISABLED	
	ALT5	GPIO3_IO20	SPEED - MEDIUM	
	ALT6	EPIT2_OUT	DSE - 40_OHM	
			SRE - SLOW	
EIM_D21	ALT0	EIM_DATA21	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA21
	ALT1	ECSPI4_SCLK	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN17	PUE - PULL	
	ALT3	IPU2_CSI1_DATA11	PKE - ENABLED	
	ALT4	USB_OTG_OC	ODE - DISABLED	
	ALT5	GPIO3_IO21	SPEED - MEDIUM	
	ALT6	I2C1_SCL	DSE - 40_OHM	
	ALT7	SPDIF_IN	SRE - SLOW	
EIM_D22	ALT0	EIM_DATA22	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA22
	ALT1	ECSPI4_MISO	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN01	PUE - PULL	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT3	IPU2_CSI1_DATA10	PKE - ENABLED	
	ALT4	USB_OTG_PWR	ODE - DISABLED	
	ALT5	GPIO3_IO22	SPEED - MEDIUM	
	ALT6	SPDIF_OUT	DSE - 40_OHM	
			SRE - SLOW	
EIM_D23	ALT0	EIM_DATA23	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA23
	ALT1	IPU1_DI0_D0_CS	PUS - 100K_OHM_PU	
	ALT2	UART3_CTS_B	PUE - PULL	
	ALT3	UART1_DCD_B	PKE - ENABLED	
	ALT4	IPU2_CSI1_DATA_EN	ODE - DISABLED	
	ALT5	GPIO3_IO23	SPEED - MEDIUM	
	ALT6	IPU1_DI1_PIN02	DSE - 40_OHM	
	ALT7	IPU1_DI1_PIN14	SRE - SLOW	
EIM_D24	ALT0	EIM_DATA24	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA24
	ALT1	ECSPI4_SS2	PUS - 100K_OHM_PU	
	ALT2	UART3_TX_DATA	PUE - PULL	
	ALT3	ECSPI1_SS2	PKE - ENABLED	
	ALT4	ECSPI2_SS2	ODE - DISABLED	
	ALT5	GPIO3_IO24	SPEED - MEDIUM	
	ALT6	AUD5_RXFS	DSE - 40_OHM	
	ALT7	UART1_DTR_B	SRE - SLOW	
EIM_D25	ALT0	EIM_DATA25	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA25
	ALT1	ECSPI4_SS3	PUS - 100K_OHM_PU	
	ALT2	UART3_RX_DATA	PUE - PULL	
	ALT3	ECSPI1_SS3	PKE - ENABLED	
	ALT4	ECSPI2_SS3	ODE - DISABLED	
	ALT5	GPIO3_IO25	SPEED - MEDIUM	
	ALT6	AUD5_RXC	DSE - 40_OHM	
	ALT7	UART1_DSR_B	SRE - SLOW	
EIM_D26	ALT0	EIM_DATA26	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA26
	ALT1	IPU1_DI1_PIN11	PUS - 100K_OHM_PU	
	ALT2	IPU1_CSI0_DATA01	PUE - PULL	
	ALT3	IPU2_CSI1_DATA14	PKE - ENABLED	
	ALT4	UART2_TX_DATA	ODE - DISABLED	
	ALT5	GPIO3_IO26	SPEED - MEDIUM	
	ALT6	IPU1_SISG2	DSE - 40_OHM	
	ALT7	IPU1_DISP1_DATA22	SRE - SLOW	
EIM_D27	ALT0	EIM_DATA27	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA27

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	IPU1_DI1_PIN13	PUS - 100K_OHM_PU	
	ALT2	IPU1_CSI0_DATA00	PUE - PULL	
	ALT3	IPU2_CSI1_DATA13	PKE - ENABLED	
	ALT4	UART2_RX_DATA	ODE - DISABLED	
	ALT5	GPIO3_IO27	SPEED - MEDIUM	
	ALT6	IPU1_SISG3	DSE - 40_OHM	
	ALT7	IPU1_DISP1_DATA23	SRE - SLOW	
EIM_D28	ALT0	EIM_DATA28	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA28
	ALT1	I2C1_SDA	PUS - 100K_OHM_PU	
	ALT2	ECSPI4_MOSI	PUE - PULL	
	ALT3	IPU2_CSI1_DATA12	PKE - ENABLED	
	ALT4	UART2_CTS_B	ODE - DISABLED	
	ALT5	GPIO3_IO28	SPEED - MEDIUM	
	ALT6	IPU1_EXT_TRIG	DSE - 40_OHM	
	ALT7	IPU1_DI0_PIN13	SRE - SLOW	
EIM_D29	ALT0	EIM_DATA29	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA29
	ALT1	IPU1_DI1_PIN15	PUS - 100K_OHM_PU	
	ALT2	ECSPI4_SS0	PUE - PULL	
	ALT4	UART2_RTS_B	PKE - ENABLED	
	ALT5	GPIO3_IO29	ODE - DISABLED	
	ALT6	IPU2_CSI1_VSYNC	SPEED - MEDIUM	
	ALT7	IPU1_DI0_PIN14	DSE - 40_OHM	
			SRE - SLOW	
EIM_D30	ALT0	EIM_DATA30	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA30
	ALT1	IPU1_DISP1_DATA21	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN11	PUE - PULL	
	ALT3	IPU1_CSI0_DATA03	PKE - ENABLED	
	ALT4	UART3_CTS_B	ODE - DISABLED	
	ALT5	GPIO3_IO30	SPEED - MEDIUM	
	ALT6	USB_H1_OC	DSE - 40_OHM	
			SRE - SLOW	
EIM_D31	ALT0	EIM_DATA31	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_DATA31
	ALT1	IPU1_DISP1_DATA20	PUS - 100K_OHM_PU	
	ALT2	IPU1_DI0_PIN12	PUE - PULL	
	ALT3	IPU1_CSI0_DATA02	PKE - ENABLED	
	ALT4	UART3_RTS_B	ODE - DISABLED	
	ALT5	GPIO3_IO31	SPEED - MEDIUM	
	ALT6	USB_H1_PWR	DSE - 40_OHM	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
EIM_EB0	ALT0	EIM_EB0_B	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_EB0_B
	ALT1	IPU1_DISP1_DATA11	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA11	PUE - PULL	
	ALT4	CCM_PMIC_READY	PKE - ENABLED	
	ALT5	GPIO2_IO28	ODE - DISABLED	
	ALT7	SRC_BOOT_CFG27	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_EB1	ALT0	EIM_EB1_B	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_EB1_B
	ALT1	IPU1_DISP1_DATA10	PUS - 100K_OHM_PU	
	ALT2	IPU2_CSI1_DATA10	PUE - PULL	
	ALT5	GPIO2_IO29	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG28	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - FAST	
EIM_EB2	ALT0	EIM_EB2_B	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_EB2_B
	ALT1	ECSPI1_SS0	PUS - 100K_OHM_PU	
	ALT3	IPU2_CSI1_DATA19	PUE - PULL	
	ALT4	HDMI_TX_DDC_SCL	PKE - ENABLED	
	ALT5	GPIO2_IO30	ODE - DISABLED	
	ALT6	I2C2_SCL	SPEED - MEDIUM	
	ALT7	SRC_BOOT_CFG30	DSE - 40_OHM	
			SRE - SLOW	
EIM_EB3	ALT0	EIM_EB3_B	HYS - ENABLED	SW_PAD_CTL_PAD_EIM_EB3_B
	ALT1	ECSPI4_RDY	PUS - 100K_OHM_PU	
	ALT2	UART3_RTS_B	PUE - PULL	
	ALT3	UART1_RI_B	PKE - ENABLED	
	ALT4	IPU2_CSI1_HSYNC	ODE - DISABLED	
	ALT5	GPIO2_IO31	SPEED - MEDIUM	
	ALT6	IPU1_DI1_PIN03	DSE - 40_OHM	
	ALT7	SRC_BOOT_CFG31	SRE - SLOW	
EIM_LBA	ALT0	EIM_LBA_B	HYS - DISABLED	SW_PAD_CTL_PAD_EIM_LBA_B
	ALT1	IPU1_DI1_PIN17	PUS - 100K_OHM_PU	
	ALT2	ECSPI2_SS1	PUE - PULL	
	ALT5	GPIO2_IO27	PKE - ENABLED	
	ALT7	SRC_BOOT_CFG26	ODE - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SPEED - MEDIUM DSE - 40_OHM SRE - FAST	
EIM_OE	ALT0	EIM_OE_B	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_OE_B
	ALT1	IPU1_DI1_PIN07		
	ALT2	ECSPI2_MISO		
	ALT5	GPIO2_IO25		
EIM_RW	ALT0	EIM_RW	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_EIM_RW
	ALT1	IPU1_DI1_PIN08		
	ALT2	ECSPI2_SS0		
	ALT5	GPIO2_IO26		
	ALT7	SRC_BOOT_CFG29		
EIM_WAIT	ALT0	EIM_WAIT_B	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 60_OHM SRE - SLOW	SW_PAD_CTL_PAD_EIM_WAIT_B
	ALT1	EIM_DTACK_B		
	ALT5	GPIO5_IO00		
	ALT7	SRC_BOOT_CFG25		
ENET_CRS_DV	ALT1	ENET_RX_EN	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_ENET_CRS_DV
	ALT2	ESAI_TX_CLK		
	ALT3	SPDIF_EXT_CLK		
	ALT5	GPIO1_IO25		
ENET_MDC	ALT0	MLB_DATA	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_ENET_MDC
	ALT1	ENET_MDC		
	ALT2	ESAI_TX5_RX0		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT4	ENET_1588_EVENT1_IN	PKE - ENABLED	
	ALT5	GPIO1_IO31	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
ENET_MDIO	ALT1	ENET_MDIO	HYS - ENABLED	SW_PAD_CTL_PAD_ENET_MDIO
	ALT2	ESAI_RX_CLK	PUS - 100K_OHM_PU	
	ALT4	ENET_1588_EVENT1_OUT	PUE - PULL PKE - ENABLED	
	ALT5	GPIO1_IO22	ODE - DISABLED	
	ALT6	SPDIF_LOCK	SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
ENET_REF_CLK	ALT1	ENET_TX_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_ENET_REF_CLK
	ALT2	ESAI_RX_FS	PUS - 100K_OHM_PU	
	ALT5	GPIO1_IO23	PUE - PULL	
	ALT6	SPDIF_SR_CLK	PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
ENET_RXD0	ALT0	XTALOSC_OSC32K_32K_OUT	HYS - ENABLED	SW_PAD_CTL_PAD_ENET_RX_DATA0
	ALT1	ENET_RX_DATA0	PUS - 100K_OHM_PU	
	ALT2	ESAI_TX_HF_CLK	PUE - PULL	
	ALT3	SPDIF_OUT	PKE - ENABLED	
	ALT5	GPIO1_IO27	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
ENET_RXD1	ALT0	MLB_SIG	HYS - ENABLED	SW_PAD_CTL_PAD_ENET_RX_DATA1
	ALT1	ENET_RX_DATA1	PUS - 100K_OHM_PU	
	ALT2	ESAI_TX_FS	PUE - PULL	
	ALT4	ENET_1588_EVENT3_OUT	PKE - ENABLED	
	ALT5	GPIO1_IO26	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
ENET_RX_ER	ALT0	USB_OTG_ID	HYS - ENABLED	SW_PAD_CTL_PAD_ENET_RX_ER

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	ENET_RX_ER	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT2	ESAI_RX_HF_CLK		
	ALT3	SPDIF_IN		
	ALT4	ENET_1588_EVENT2_OUT		
	ALT5	GPIO1_IO24		
ENET_TXD0	ALT1	ENET_TX_DATA0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_ENET_TX_DATA0
	ALT2	ESAI_TX4_RX1		
	ALT5	GPIO1_IO30		
ENET_TXD1	ALT0	MLB_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_ENET_TX_DATA1
	ALT1	ENET_TX_DATA1		
	ALT2	ESAI_TX2_RX3		
	ALT4	ENET_1588_EVENT0_IN		
	ALT5	GPIO1_IO29		
ENET_TX_EN	ALT1	ENET_TX_EN	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_ENET_TX_EN
	ALT2	ESAI_TX3_RX2		
	ALT5	GPIO1_IO28		
GPIO_0	ALT0	CCM_CLKO1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_GPIO00
	ALT2	KEY_COL5		
	ALT3	ASRC_EXT_CLK		
	ALT4	EPIT1_OUT		
	ALT5	GPIO1_IO00		
	ALT6	USB_H1_PWR		
	ALT7	SNVS_VIO_5		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
GPIO_1	ALT0	ESAI_RX_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO01
	ALT1	WDOG2_B	PUS - 100K_OHM_PU	
	ALT2	KEY_ROW5	PUE - PULL	
	ALT3	USB_OTG_ID	PKE - ENABLED	
	ALT4	PWM2_OUT	ODE - DISABLED	
	ALT5	GPIO1_IO01	SPEED - MEDIUM	
	ALT6	SD1_CD_B	DSE - 40_OHM	
			SRE - SLOW	
GPIO_2	ALT0	ESAI_TX_FS	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO02
	ALT2	KEY_ROW6	PUS - 100K_OHM_PU	
	ALT5	GPIO1_IO02	PUE - PULL	
	ALT6	SD2_WP	PKE - ENABLED	
	ALT7	MLB_DATA	ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
GPIO_3	ALT0	ESAI_RX_HF_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO03
	ALT2	I2C3_SCL	PUS - 100K_OHM_PU	
	ALT3	XTALOSC_REF_CLK_24M	PUE - PULL	
	ALT4	CCM_CLKO2	PKE - ENABLED	
	ALT5	GPIO1_IO03	ODE - DISABLED	
	ALT6	USB_H1_OC	SPEED - MEDIUM	
	ALT7	MLB_CLK	DSE - 40_OHM	
			SRE - SLOW	
GPIO_4	ALT0	ESAI_TX_HF_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO04
	ALT2	KEY_COL7	PUS - 100K_OHM_PU	
	ALT5	GPIO1_IO04	PUE - PULL	
	ALT6	SD2_CD_B	PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
GPIO_5	ALT0	ESAI_TX2_RX3	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO05
	ALT2	KEY_ROW7	PUS - 100K_OHM_PU	
	ALT3	CCM_CLKO1	PUE - PULL	
	ALT5	GPIO1_IO05	PKE - ENABLED	
	ALT6	I2C3_SCL	ODE - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT7	ARM_EVENTI	SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
GPIO_6	ALT0	ESAI_TX_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO06
	ALT2	I2C3_SDA		
	ALT5	GPIO1_IO06		
	ALT6	SD2_LCTL		
	ALT7	MLB_SIG		
GPIO_7	ALT0	ESAI_TX4_RX1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO07
	ALT1	ECSPI5_RDY		
	ALT2	EPIT1_OUT		
	ALT3	FLEXCAN1_TX		
	ALT4	UART2_TX_DATA		
	ALT5	GPIO1_IO07		
	ALT6	SPDIF_LOCK		
	ALT7	USB_OTG_HOST_MODE		
GPIO_8	ALT0	ESAI_TX5_RX0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO08
	ALT1	XTALOSC_REF_CLK_32K		
	ALT2	EPIT2_OUT		
	ALT3	FLEXCAN1_RX		
	ALT4	UART2_RX_DATA		
	ALT5	GPIO1_IO08		
	ALT6	SPDIF_SR_CLK		
	ALT7	USB_OTG_PWR_CTL_WAKE		
GPIO_9	ALT0	ESAI_RX_FS	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO09
	ALT1	WDOG1_B		
	ALT2	KEY_COL6		
	ALT3	CCM_REF_EN_B		
	ALT4	PWM1_OUT		
	ALT5	GPIO1_IO09		
	ALT6	SD1_WP		
GPIO_16	ALT0	ESAI_TX3_RX2	HYS - ENABLED PUS - 100K_OHM_PU	SW_PAD_CTL_PAD_GPIO16
	ALT1	ENET_1588_EVENT2_IN		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT2	ENET_REF_CLK	PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT3	SD1_LCTL		
	ALT4	SPDIF_IN		
	ALT5	GPIO7_IO11		
	ALT6	I2C3_SDA		
	ALT7	JTAG_DE_B		
GPIO_17	ALT0	ESAI_TX0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO17
	ALT1	ENET_1588_EVENT3_IN		
	ALT2	CCM_PMIC_READY		
	ALT3	SDMA_EXT_EVENT0		
	ALT4	SPDIF_OUT		
	ALT5	GPIO7_IO12		
GPIO_18	ALT0	ESAI_TX1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO18
	ALT1	ENET_RX_CLK		
	ALT2	SD3_VSELECT		
	ALT3	SDMA_EXT_EVENT1		
	ALT4	ASRC_EXT_CLK		
	ALT5	GPIO7_IO13		
	ALT6	SNVS_VIO_5_CTL		
GPIO_19	ALT0	KEY_COL5	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_GPIO19
	ALT1	ENET_1588_EVENT0_OUT		
	ALT2	SPDIF_OUT		
	ALT3	CCM_CLKO1		
	ALT4	ECSPI1_RDY		
	ALT5	GPIO4_IO05		
	ALT6	ENET_TX_ER		
HDMI_CLKM		HDMI_TX_CLK_N		
HDMI_CLKP		HDMI_TX_CLK_P		
HDMI_D0M		HDMI_TX_DATA0_N		
HDMI_D0P		HDMI_TX_DATA0_P		
HDMI_D1M		HDMI_TX_DATA1_N		
HDMI_D1P		HDMI_TX_DATA1_P		
HDMI_D2M		HDMI_TX_DATA2_N		
HDMI_D2P		HDMI_TX_DATA2_P		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
HDMI_DDCCEC		HDMI_TX_DDC_CEC		
HDMI_HPD		HDMI_TX_HPD		
JTAG_MOD		JTAG_MOD	HYS - DISABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - 50MHZ DSE - 60_OHM SRE - SLOW	SW_PAD_CTL_PAD_JTAG_MOD
JTAG_TCK		JTAG_TCK	HYS - DISABLED PUS - 47K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - 50MHZ DSE - 60_OHM SRE - SLOW	SW_PAD_CTL_PAD_JTAG_TCK
JTAG_TDI		JTAG_TDI	HYS - DISABLED PUS - 47K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - 50MHZ DSE - 60_OHM SRE - SLOW	SW_PAD_CTL_PAD_JTAG_TDI
JTAG_TDO		JTAG_TDO	HYS - DISABLED PUS - 100K_OHM_PU PUE - KEEP PKE - ENABLED ODE - DISABLED SPEED - 100MHZ DSE - 40_OHM SRE - FAST	SW_PAD_CTL_PAD_JTAG_TDO
JTAG_TMS		JTAG_TMS	HYS - DISABLED PUS - 47K_OHM_PU PUE - PULL PKE - ENABLED	SW_PAD_CTL_PAD_JTAG_TMS

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			ODE - DISABLED SPEED - 50MHZ DSE - 60_OHM SRE - SLOW	
JTAG_TRSTB		JTAG_TRSTB	HYS - DISABLED PUS - 47K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - 50MHZ DSE - 60_OHM SRE - SLOW	SW_PAD_CTL_PAD_JTAG_TRSTB
KEY_COL0	ALT0	ECSPI1_SCLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_COL0
	ALT1	ENET_RX_DATA3		
	ALT2	AUD5_TXC		
	ALT3	KEY_COL0		
	ALT4	UART4_TX_DATA		
	ALT5	GPIO4_IO06		
	ALT6	DCIC1_OUT		
KEY_COL1	ALT0	ECSPI1_MISO	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_COL1
	ALT1	ENET_MDIO		
	ALT2	AUD5_TXFS		
	ALT3	KEY_COL1		
	ALT4	UART5_TX_DATA		
	ALT5	GPIO4_IO08		
	ALT6	SD1_VSELECT		
KEY_COL2	ALT0	ECSPI1_SS1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_COL2
	ALT1	ENET_RX_DATA2		
	ALT2	FLEXCAN1_TX		
	ALT3	KEY_COL2		
	ALT4	ENET_MDC		
	ALT5	GPIO4_IO10		
	ALT6	USB_H1_PWR_CTL_WAKE		
KEY_COL3	ALT0	ECSPI1_SS3	HYS - ENABLED PUS - 100K_OHM_PU	SW_PAD_CTL_PAD_KEY_COL3
	ALT1	ENET_CRS		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT2	HDMI_TX_DDC_SCL	PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT3	KEY_COL3		
	ALT4	I2C2_SCL		
	ALT5	GPIO4_IO12		
	ALT6	SPDIF_IN		
KEY_COL4	ALT0	FLEXCAN2_TX	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_COL4
	ALT1	IPU1_SISG4		
	ALT2	USB_OTG_OC		
	ALT3	KEY_COL4		
	ALT4	UART5_RTS_B		
	ALT5	GPIO4_IO14		
KEY_ROW0	ALT0	ECSPI1_MOSI	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_ROW0
	ALT1	ENET_TX_DATA3		
	ALT2	AUD5_TXD		
	ALT3	KEY_ROW0		
	ALT4	UART4_RX_DATA		
	ALT5	GPIO4_IO07		
	ALT6	DCIC2_OUT		
KEY_ROW1	ALT0	ECSPI1_SS0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_ROW1
	ALT1	ENET_COL		
	ALT2	AUD5_RXD		
	ALT3	KEY_ROW1		
	ALT4	UART5_RX_DATA		
	ALT5	GPIO4_IO09		
	ALT6	SD2_VSELECT		
KEY_ROW2	ALT0	ECSPI1_SS2	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_ROW2
	ALT1	ENET_TX_DATA2		
	ALT2	FLEXCAN1_RX		
	ALT3	KEY_ROW2		
	ALT4	SD2_VSELECT		
	ALT5	GPIO4_IO11		
	ALT6	HDMI_TX_CEC_LINE		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
KEY_ROW3	ALT0	XTALOSC_OSC32K_32K_OUT	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_ROW3
	ALT1	ASRC_EXT_CLK		
	ALT2	HDMI_TX_DDC_SDA		
	ALT3	KEY_ROW3		
	ALT4	I2C2_SDA		
	ALT5	GPIO4_IO13		
	ALT6	SD1_VSELECT		
KEY_ROW4	ALT0	FLEXCAN2_RX	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_KEY_ROW4
	ALT1	IPU1_SISG5		
	ALT2	USB_OTG_PWR		
	ALT3	KEY_ROW4		
	ALT4	UART5_CTS_B		
	ALT5	GPIO4_IO15		
LVDS0_CLK_N		LVDS0_CLK_N		
LVDS0_CLK_P		LVDS0_CLK_P		
LVDS0_TX0_N		LVDS0_DATA0_N		
LVDS0_TX0_P		LVDS0_DATA0_P		
LVDS0_TX1_N		LVDS0_DATA1_N		
LVDS0_TX1_P		LVDS0_DATA1_P		
LVDS0_TX2_N		LVDS0_DATA2_N		
LVDS0_TX2_P		LVDS0_DATA2_P		
LVDS0_TX3_N		LVDS0_DATA3_N		
LVDS0_TX3_P		LVDS0_DATA3_P		
LVDS1_CLK_N		LVDS1_CLK_N		
LVDS1_CLK_P		LVDS1_CLK_P		
LVDS1_TX0_N		LVDS1_DATA0_N		
LVDS1_TX0_P		LVDS1_DATA0_P		
LVDS1_TX1_N		LVDS1_DATA1_N		
LVDS1_TX1_P		LVDS1_DATA1_P		
LVDS1_TX2_N		LVDS1_DATA2_N		
LVDS1_TX2_P		LVDS1_DATA2_P		
LVDS1_TX3_N		LVDS1_DATA3_N		
LVDS1_TX3_P		LVDS1_DATA3_P		
MLB_CN		MLB_CLK_N		
MLB_CP		MLB_CLK_P		
MLB_DN		MLB_DATA_N		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
MLB_DP		MLB_DATA_P		
MLB_SN		MLB_SIG_N		
MLB_SP		MLB_SIG_P		
NANDF_ALE	ALT0	NAND_ALE	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_ALE
	ALT1	SD4_RESET		
	ALT5	GPIO6_IO08		
NANDF_CLE	ALT0	NAND_CLE	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_CLE
	ALT1	IPU2_SISG4		
	ALT5	GPIO6_IO07		
NANDF_CS0	ALT0	NAND_CE0_B	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_CS0_B
	ALT5	GPIO6_IO11		
NANDF_CS1	ALT0	NAND_CE1_B	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_CS1_B
	ALT1	SD4_VSELECT		
	ALT2	SD3_VSELECT		
	ALT5	GPIO6_IO14		
NANDF_CS2	ALT0	NAND_CE2_B	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_NAND_CS2_B
	ALT1	IPU1_SISG0		
	ALT2	ESAI_TX0		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT3	EIM_CRE	PKE - ENABLED	
	ALT4	CCM_CLKO2	ODE - DISABLED	
	ALT5	GPIO6_IO15	SPEED - MEDIUM	
	ALT6	IPU2_SISG0	DSE - 40_OHM	
			SRE - SLOW	
NANDF_CS3	ALT0	NAND_CE3_B	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_CS3_B
	ALT1	IPU1_SISG1	PUS - 100K_OHM_PU	
	ALT2	ESAI_TX1	PUE - PULL	
	ALT3	EIM_ADDR26	PKE - ENABLED	
	ALT5	GPIO6_IO16	ODE - DISABLED	
	ALT6	IPU2_SISG1	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
NANDF_D0	ALT0	NAND_DATA00	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_DATA00
	ALT1	SD1_DATA4	PUS - 100K_OHM_PU	
	ALT5	GPIO2_IO00	PUE - PULL	
			PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
NANDF_D1	ALT0	NAND_DATA01	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_DATA01
	ALT1	SD1_DATA5	PUS - 100K_OHM_PU	
	ALT5	GPIO2_IO01	PUE - PULL	
			PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
NANDF_D2	ALT0	NAND_DATA02	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_DATA02
	ALT1	SD1_DATA6	PUS - 100K_OHM_PU	
	ALT5	GPIO2_IO02	PUE - PULL	
			PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
NANDF_D3	ALT0	NAND_DATA03	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_DATA03

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT1	SD1_DATA7	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT5	GPIO2_IO03		
NANDF_D4	ALT0	NAND_DATA04	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_DATA04
	ALT1	SD2_DATA4		
	ALT5	GPIO2_IO04		
NANDF_D5	ALT0	NAND_DATA05	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_DATA05
	ALT1	SD2_DATA5		
	ALT5	GPIO2_IO05		
NANDF_D6	ALT0	NAND_DATA06	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_NAND_DATA06
	ALT1	SD2_DATA6		
	ALT5	GPIO2_IO06		
NANDF_D7	ALT0	NAND_DATA07	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_NAND_DATA07
	ALT1	SD2_DATA7		
	ALT5	GPIO2_IO07		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
NANDF_RB0	ALT0	NAND_READY_B	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_READY_B
	ALT1	IPU2_DI0_PIN01	PUS - 100K_OHM_PU	
	ALT5	GPIO6_IO10	PUE - PULL	
			PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
NANDF_WP_B	ALT0	NAND_WP_B	HYS - ENABLED	SW_PAD_CTL_PAD_NAND_WP_B
	ALT1	IPU2_SISG5	PUS - 100K_OHM_PU	
	ALT5	GPIO6_IO09	PUE - PULL	
			PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	
ONOFF		SRC_ONOFF		
PCIE_RXM		PCIE_RX_N		
PCIE_RXP		PCIE_RX_P		
PCIE_TXM		PCIE_TX_N		
PCIE_TXP		PCIE_TX_P		
PMIC_ON_REQ		SNVS_PMIC_ON_REQ		
PMIC_STBY_REQ		CCM_PMIC_STBY_REQ		
POR_B		SRC_POR_B		
RGMII_RD0	ALT0	HSI_RX_READY	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RD0
	ALT1	RGMII_RD0	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO25	PUS - 100K_OHM_PU	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_RD1	ALT0	HSI_TX_FLAG	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RD1
	ALT1	RGMII_RD1	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO27	PUS - 100K_OHM_PU	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			DSE - 37_OHM	
RGMII_RD2	ALT0	HSI_TX_DATA	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RD2
	ALT1	RGMII_RD2	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO28	PUS - 100K_OHM_PU	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_RD3	ALT0	HSI_TX_WAKE	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RD3
	ALT1	RGMII_RD3	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO29	PUS - 100K_OHM_PU	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_RXC	ALT0	USB_H3_STROBE	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RXC
	ALT1	RGMII_RXC	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO30	PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_RX_CTL	ALT0	USB_H3_DATA	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_RX_CTL
	ALT1	RGMII_RX_CTL	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO24	PUS - 100K_OHM_PD	SW_PAD_CTL_GRP_RGMII_TERM
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_TD0	ALT0	HSI_TX_READY	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_TD0
	ALT1	RGMII_TD0	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT5	GPIO6_IO20	PUS - 100K_OHM_PU	
			PUE - PULL	
			PKE - ENABLED	
			ODT - DISABLED	
			DSE - 37_OHM	
RGMII_TD1	ALT0	HSI_RX_FLAG	DDR_SEL - 1P2V_IO	SW_PAD_CTL_PAD_RGMII_TD1
	ALT1	RGMII_TD1	HYS - ENABLED	SW_PAD_CTL_GRP_DDR_TYPE_RGMII

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO6_IO21	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODT - DISABLED DSE - 37_OHM	
RGMII_TD2	ALT0	HSI_RX_DATA	DDR_SEL - 1P2V_IO HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODT - DISABLED DSE - 37_OHM	SW_PAD_CTL_PAD_RGMII_TD2 SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT1	RGMII_TD2		
	ALT5	GPIO6_IO22		
RGMII_TD3	ALT0	HSI_RX_WAKE	DDR_SEL - 1P2V_IO HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODT - DISABLED DSE - 37_OHM	SW_PAD_CTL_PAD_RGMII_TD3 SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT1	RGMII_TD3		
	ALT5	GPIO6_IO23		
RGMII_TXC	ALT0	USB_H2_DATA	DDR_SEL - 1P2V_IO HYS - ENABLED PUS - 100K_OHM_PD PUE - PULL PKE - ENABLED ODT - DISABLED DSE - 37_OHM	SW_PAD_CTL_PAD_RGMII_TXC SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT1	RGMII_TXC		
	ALT2	SPDIF_EXT_CLK		
	ALT5	GPIO6_IO19		
	ALT7	XTALOSC_REF_CLK_24M		
RGMII_TX_CTL	ALT0	USB_H2_STROBE	DDR_SEL - 1P2V_IO HYS - ENABLED PUS - 100K_OHM_PD PUE - PULL PKE - ENABLED ODT - DISABLED DSE - 37_OHM	SW_PAD_CTL_PAD_RGMII_TX_CTL SW_PAD_CTL_GRP_DDR_TYPE_RGMII
	ALT1	RGMII_TX_CTL		
	ALT5	GPIO6_IO26		
	ALT7	ENET_REF_CLK		
RTC_XTALI		XTALOSC_RTC_XTALI		
RTC_XTALO		XTALOSC_RTC_XTALO		
SATA_RXM		SATA_PHY_RX_N		
SATA_RXP		SATA_PHY_RX_P		
SATA_TXM		SATA_PHY_TX_N		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
SATA_TXP		SATA_PHY_TX_P		
SD1_CLK	ALT0	SD1_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD1_CLK
	ALT1	ECSPI5_SCLK		
	ALT2	XTALOSC_OSC32K_32K_OUT		
	ALT3	GPT_CLKIN		
	ALT5	GPIO1_IO20		
SD1_CMD	ALT0	SD1_CMD	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD1_CMD
	ALT1	ECSPI5_MOSI		
	ALT2	PWM4_OUT		
	ALT3	GPT_COMPARE1		
	ALT5	GPIO1_IO18		
SD1_DAT0	ALT0	SD1_DATA0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD1_DATA0
	ALT1	ECSPI5_MISO		
	ALT3	GPT_CAPTURE1		
	ALT5	GPIO1_IO16		
SD1_DAT1	ALT0	SD1_DATA1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD1_DATA1
	ALT1	ECSPI5_SS0		
	ALT2	PWM3_OUT		
	ALT3	GPT_CAPTURE2		
	ALT5	GPIO1_IO17		
SD1_DAT2	ALT0	SD1_DATA2	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED	SW_PAD_CTL_PAD_SD1_DATA2
	ALT1	ECSPI5_SS1		
	ALT2	GPT_COMPARE2		
	ALT3	PWM2_OUT		
	ALT4	WDOG1_B		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO1_IO19	ODE - DISABLED	
	ALT6	WDOG1_RESET_B_DEB	SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD1_DAT3	ALT0	SD1_DATA3	HYS - ENABLED	SW_PAD_CTL_PAD_SD1_DATA3
	ALT1	ECSPI5_SS2	PUS - 100K_OHM_PU	
	ALT2	GPT_COMPARE3	PUE - PULL	
	ALT3	PWM1_OUT	PKE - ENABLED	
	ALT4	WDOG2_B	ODE - DISABLED	
	ALT5	GPIO1_IO21	SPEED - MEDIUM	
	ALT6	WDOG2_RESET_B_DEB	DSE - 40_OHM SRE - SLOW	
SD2_CLK	ALT0	SD2_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_SD2_CLK
	ALT1	ECSPI5_SCLK	PUS - 100K_OHM_PU	
	ALT2	KEY_COL5	PUE - PULL	
	ALT3	AUD4_RXFS	PKE - ENABLED	
	ALT5	GPIO1_IO10	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD2_CMD	ALT0	SD2_CMD	HYS - ENABLED	SW_PAD_CTL_PAD_SD2_CMD
	ALT1	ECSPI5_MOSI	PUS - 100K_OHM_PU	
	ALT2	KEY_ROW5	PUE - PULL	
	ALT3	AUD4_RXC	PKE - ENABLED	
	ALT5	GPIO1_IO11	ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD2_DAT0	ALT0	SD2_DATA0	HYS - ENABLED	SW_PAD_CTL_PAD_SD2_DATA0
	ALT1	ECSPI5_MISO	PUS - 100K_OHM_PU	
	ALT3	AUD4_RXD	PUE - PULL	
	ALT4	KEY_ROW7	PKE - ENABLED	
	ALT5	GPIO1_IO15	ODE - DISABLED	
	ALT6	DCIC2_OUT	SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD2_DAT1	ALT0	SD2_DATA1	HYS - ENABLED	SW_PAD_CTL_PAD_SD2_DATA1
	ALT1	ECSPI5_SS0	PUS - 100K_OHM_PU	

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT2	EIM_CS2_B	PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
	ALT3	AUD4_TXFS		
	ALT4	KEY_COL7		
	ALT5	GPIO1_IO14		
SD2_DAT2	ALT0	SD2_DATA2	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD2_DATA2
	ALT1	ECSPI5_SS1		
	ALT2	EIM_CS3_B		
	ALT3	AUD4_TXD		
	ALT4	KEY_ROW6		
	ALT5	GPIO1_IO13		
SD2_DAT3	ALT0	SD2_DATA3	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD2_DATA3
	ALT1	ECSPI5_SS3		
	ALT2	KEY_COL6		
	ALT3	AUD4_TXC		
	ALT5	GPIO1_IO12		
SD3_CLK	ALT0	SD3_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_CLK
	ALT1	UART2_RTS_B		
	ALT2	FLEXCAN1_RX		
	ALT5	GPIO7_IO03		
SD3_CMD	ALT0	SD3_CMD	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_CMD
	ALT1	UART2_CTS_B		
	ALT2	FLEXCAN1_TX		
	ALT5	GPIO7_IO02		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
SD3_DAT0	ALT0	SD3_DATA0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA0
	ALT1	UART1_CTS_B		
	ALT2	FLEXCAN2_TX		
	ALT5	GPIO7_IO04		
SD3_DAT1	ALT0	SD3_DATA1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA1
	ALT1	UART1_RTS_B		
	ALT2	FLEXCAN2_RX		
	ALT5	GPIO7_IO05		
SD3_DAT2	ALT0	SD3_DATA2	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA2
	ALT5	GPIO7_IO06		
SD3_DAT3	ALT0	SD3_DATA3	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA3
	ALT1	UART3_CTS_B		
	ALT5	GPIO7_IO07		
SD3_DAT4	ALT0	SD3_DATA4	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED	SW_PAD_CTL_PAD_SD3_DATA4
	ALT1	UART2_RX_DATA		
	ALT5	GPIO7_IO01		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD3_DAT5	ALT0	SD3_DATA5	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA5
	ALT1	UART2_TX_DATA		
	ALT5	GPIO7_IO00		
SD3_DAT6	ALT0	SD3_DATA6	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA6
	ALT1	UART1_RX_DATA		
	ALT5	GPIO6_IO18		
SD3_DAT7	ALT0	SD3_DATA7	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_DATA7
	ALT1	UART1_TX_DATA		
	ALT5	GPIO6_IO17		
SD3_RST	ALT0	SD3_RESET	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD3_RESET
	ALT1	UART3_RTS_B		
	ALT5	GPIO7_IO08		
SD4_CLK	ALT0	SD4_CLK	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL	SW_PAD_CTL_PAD_SD4_CLK
	ALT1	NAND_WE_B		
	ALT2	UART3_RX_DATA		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO7_IO10	PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD4_CMD	ALT0	SD4_CMD	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_CMD
	ALT1	NAND_RE_B		
	ALT2	UART3_TX_DATA		
	ALT5	GPIO7_IO09		
SD4_DAT0	ALT1	SD4_DATA0	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA0
	ALT2	NAND_DQS		
	ALT5	GPIO2_IO08		
SD4_DAT1	ALT1	SD4_DATA1	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA1
	ALT2	PWM3_OUT		
	ALT5	GPIO2_IO09		
SD4_DAT2	ALT1	SD4_DATA2	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA2
	ALT2	PWM4_OUT		
	ALT5	GPIO2_IO10		
SD4_DAT3	ALT1	SD4_DATA3	HYS - ENABLED	SW_PAD_CTL_PAD_SD4_DATA3

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO2_IO11	PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	
SD4_DAT4	ALT1	SD4_DATA4	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA4
	ALT2	UART2_RX_DATA		
	ALT5	GPIO2_IO12		
SD4_DAT5	ALT1	SD4_DATA5	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA5
	ALT2	UART2_RTS_B		
	ALT5	GPIO2_IO13		
SD4_DAT6	ALT1	SD4_DATA6	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	SW_PAD_CTL_PAD_SD4_DATA6
	ALT2	UART2_CTS_B		
	ALT5	GPIO2_IO14		
SD4_DAT7	ALT1	SD4_DATA7	HYS - ENABLED PUS - 100K_OHM_PU PUE - PULL PKE - ENABLED ODE - DISABLED SPEED - MEDIUM DSE - 40_OHM	SW_PAD_CTL_PAD_SD4_DATA7
	ALT2	UART2_TX_DATA		
	ALT5	GPIO2_IO15		

Table continues on the next page...

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
			SRE - SLOW	
TAMPER		SNVS_TAMPER		
TEST_MODE		TCU_TEST_MODE		
USB_H1_DN		USB_H1_DN		
USB_H1_DP		USB_H1_DP		
USB_H1_VBUS		USB_H1_VBUS		
USB_OTG_CHD_B		USB_OTG_CHD_B		
USB_OTG_DN		USB_OTG_DN		
USB_OTG_DP		USB_OTG_DP		
USB_OTG_VBUS		USB_OTG_VBUS		
XTALI		XTALOSC_XTALI		
XTALO		XTALOSC_XTALO		

4.1.2 Muxing Options

An additional view of external signals muxing is shown by the presentation of the muxing options per block/instance.

Table 4-2. Muxing Options

Signal	Pad (Mode)	Mux/Input Select Registers
ARM - ARM Platform		
ARM_EVENTI	GPIO_5 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO05
ARM_EVENTO	CSI0_PIXCLK (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK
ARM_TRACE00	CSI0_VSYNC (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC
ARM_TRACE01	CSI0_DAT4 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04
ARM_TRACE02	CSI0_DAT5 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05
ARM_TRACE03	CSI0_DAT6 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06
ARM_TRACE04	CSI0_DAT7 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
ARM_TRACE05	CSI0_DAT8 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
ARM_TRACE06	CSI0_DAT9 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
ARM_TRACE07	CSI0_DAT10 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10
ARM_TRACE08	CSI0_DAT11 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11
ARM_TRACE09	CSI0_DAT12 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12
ARM_TRACE10	CSI0_DAT13 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13
ARM_TRACE11	CSI0_DAT14 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14
ARM_TRACE12	CSI0_DAT15 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15
ARM_TRACE13	CSI0_DAT16 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
ARM_TRACE14	CSI0_DAT17 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17
ARM_TRACE15	CSI0_DAT18 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18
ARM_TRACE_CLK	CSI0_DATA_EN (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN
ARM_TRACE_CTL	CSI0_MCLK (ALT7)	IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC
ASRC - Asynchronous Sample Rate Converter		
ASRC_EXT_CLK	GPIO_0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO00 IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT
	GPIO_18 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO18 IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT
	KEY_ROW3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3 IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT
AUDMUX - Digital Audio Multiplexer		
AUD3_RXC	CSI0_DAT10 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10
AUD3_RXD	CSI0_DAT7 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
AUD3_RXFS	CSI0_DAT11 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11
AUD3_TXC	CSI0_DAT4 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04
AUD3_TXD	CSI0_DAT5 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05
AUD3_TXFS	CSI0_DAT6 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06
AUD4_RXC	DISP0_DAT19 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19 IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT
	SD2_CMD (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_CMD IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT
AUD4_RXD	DISP0_DAT23 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23 IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT
	SD2_DAT0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0 IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT
AUD4_RXFS	DISP0_DAT18 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18 IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT
	SD2_CLK (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_CLK IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT
AUD4_TXC	DISP0_DAT20 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20 IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT
	SD2_DAT3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3 IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT
AUD4_TXD	DISP0_DAT21 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21 IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT
	SD2_DAT2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2 IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT
AUD4_TXFS	DISP0_DAT22 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT
	SD2_DAT1 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1
		IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT
AUD5_RXC	DISP0_DAT14 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14
		IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT
	EIM_D25 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
		IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT
AUD5_RXD	DISP0_DAT19 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
		IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT
	KEY_ROW1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
		IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT
AUD5_RXFS	DISP0_DAT13 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13
		IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT
	EIM_D24 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
		IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT
AUD5_TXC	DISP0_DAT16 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16
		IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT
	KEY_COL0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
		IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT
AUD5_TXD	DISP0_DAT17 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17
		IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT
	KEY_ROW0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
		IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT
AUD5_TXFS	DISP0_DAT18 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
		IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT
	KEY_COL1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1
		IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT
AUD6_RXC	DISP0_DAT6 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06
AUD6_RXD	DI0_PIN4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04
AUD6_RXFS	DISP0_DAT5 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05
AUD6_TXC	DI0_PIN15 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15
AUD6_TXD	DI0_PIN2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02
AUD6_TXFS	DI0_PIN3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03
CCM - Clock Controller Module		
CCM_CLKO1	CSI0_MCLK (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC
	GPIO_0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO00
	GPIO_5 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO05
	GPIO_19 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
CCM_CLKO2	GPIO_3 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO03

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
	NANDF_CS2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
CCM_PMIC_READY	EIM_EB0 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B IOMUXC_CCM_PMIC_READY_SELECT_INPUT
	GPIO_17 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO17 IOMUXC_CCM_PMIC_READY_SELECT_INPUT
CCM_PMIC_STBY_REQ	PMIC_STBY_REQ	Not multiplexed.
CCM_REF_EN_B	GPIO_9 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO09
DCIC1 - Display Content Integrity Checker		
DCIC1_OUT	EIM_D17 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
	KEY_COL0 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
DCIC2 - Display Content Integrity Checker		
DCIC2_OUT	KEY_ROW0 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
	SD2_DAT0 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0
ECSPI1 - Enhanced Configurable SPI		
ECSPI1_MISO	CSI0_DAT6 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06 IOMUXC_ECSPI1_MISO_SELECT_INPUT
	DISP0_DAT22 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22 IOMUXC_ECSPI1_MISO_SELECT_INPUT
	EIM_D17 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17 IOMUXC_ECSPI1_MISO_SELECT_INPUT
	KEY_COL1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1 IOMUXC_ECSPI1_MISO_SELECT_INPUT
ECSPI1_MOSI	CSI0_DAT5 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05 IOMUXC_ECSPI1_MOSI_SELECT_INPUT
	DISP0_DAT21 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21 IOMUXC_ECSPI1_MOSI_SELECT_INPUT
	EIM_D18 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18 IOMUXC_ECSPI1_MOSI_SELECT_INPUT
	KEY_ROW0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0 IOMUXC_ECSPI1_MOSI_SELECT_INPUT
ECSPI1_RDY	GPIO_19 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
ECSPI1_SCLK	CSI0_DAT4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04 IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT
	DISP0_DAT20 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20 IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT
	EIM_D16 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16 IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT
	KEY_COL0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0 IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
ECSPI1_SS0	CSI0_DAT7 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
		IOMUXC_ECSPI1_SS0_SELECT_INPUT
	DISP0_DAT23 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23
		IOMUXC_ECSPI1_SS0_SELECT_INPUT
	EIM_EB2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
		IOMUXC_ECSPI1_SS0_SELECT_INPUT
	KEY_ROW1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
		IOMUXC_ECSPI1_SS0_SELECT_INPUT
ECSPI1_SS1	DISP0_DAT15 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15
		IOMUXC_ECSPI1_SS1_SELECT_INPUT
	EIM_D19 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19
		IOMUXC_ECSPI1_SS1_SELECT_INPUT
	KEY_COL2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
		IOMUXC_ECSPI1_SS1_SELECT_INPUT
ECSPI1_SS2	EIM_D24 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
		IOMUXC_ECSPI1_SS2_SELECT_INPUT
	KEY_ROW2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
		IOMUXC_ECSPI1_SS2_SELECT_INPUT
ECSPI1_SS3	EIM_D25 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
		IOMUXC_ECSPI1_SS3_SELECT_INPUT
	KEY_COL3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
		IOMUXC_ECSPI1_SS3_SELECT_INPUT
ECSPI2 - Enhanced Configurable SPI		
ECSPI2_MISO	CSI0_DAT10 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10
		IOMUXC_ECSPI2_MISO_SELECT_INPUT
	DISP0_DAT17 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17
		IOMUXC_ECSPI2_MISO_SELECT_INPUT
	EIM_OE (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B
		IOMUXC_ECSPI2_MISO_SELECT_INPUT
ECSPI2_MOSI	CSI0_DAT9 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
		IOMUXC_ECSPI2_MOSI_SELECT_INPUT
	DISP0_DAT16 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16
		IOMUXC_ECSPI2_MOSI_SELECT_INPUT
	EIM_CS1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B
		IOMUXC_ECSPI2_MOSI_SELECT_INPUT
ECSPI2_RDY	EIM_A25 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
ECSPI2_SCLK	CSI0_DAT8 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
		IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
	DISP0_DAT19 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
		IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT
	EIM_CS0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B
		IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT
ECSPI2_SS0	CSI0_DAT11 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11
		IOMUXC_ECSPI2_SS0_SELECT_INPUT
	DISP0_DAT18 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
		IOMUXC_ECSPI2_SS0_SELECT_INPUT
	EIM_RW (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_RW
		IOMUXC_ECSPI2_SS0_SELECT_INPUT
ECSPI2_SS1	DISP0_DAT15 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15
		IOMUXC_ECSPI2_SS1_SELECT_INPUT
	EIM_LBA (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B
		IOMUXC_ECSPI2_SS1_SELECT_INPUT
ECSPI2_SS2	EIM_D24 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
ECSPI2_SS3	EIM_D25 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
ECSPI3 - Enhanced Configurable SPI		
ECSPI3_MISO	DISP0_DAT2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02
ECSPI3_MOSI	DISP0_DAT1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01
ECSPI3_RDY	DISP0_DAT7 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07
ECSPI3_SCLK	DISP0_DAT0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00
ECSPI3_SS0	DISP0_DAT3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03
ECSPI3_SS1	DISP0_DAT4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04
ECSPI3_SS2	DISP0_DAT5 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05
ECSPI3_SS3	DISP0_DAT6 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06
ECSPI4 - Enhanced Configurable SPI		
ECSPI4_MISO	EIM_D22 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
ECSPI4_MOSI	EIM_D28 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
ECSPI4_RDY	EIM_EB3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
ECSPI4_SCLK	EIM_D21 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21
ECSPI4_SS0	EIM_D20 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
		IOMUXC_ECSPI4_SS0_SELECT_INPUT
	EIM_D29 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
		IOMUXC_ECSPI4_SS0_SELECT_INPUT
ECSPI4_SS1	EIM_A25 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
ECSPI4_SS2	EIM_D24 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
ECSPI4_SS3	EIM_D25 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
ECSPI5 - Enhanced Configurable SPI		
ECSPI5_MISO	SD1_DAT0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_ECSPI5_MISO_SELECT_INPUT
	SD2_DAT0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0
		IOMUXC_ECSPI5_MISO_SELECT_INPUT
ECSPI5_MOSI	SD1_CMD (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_CMD
		IOMUXC_ECSPI5_MOSI_SELECT_INPUT
	SD2_CMD (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_CMD
		IOMUXC_ECSPI5_MOSI_SELECT_INPUT
ECSPI5_RDY	GPIO_7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
ECSPI5_SCLK	SD1_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_CLK
		IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT
	SD2_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_CLK
		IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT
ECSPI5_SS0	SD1_DAT1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1
		IOMUXC_ECSPI5_SS0_SELECT_INPUT
	SD2_DAT1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1
		IOMUXC_ECSPI5_SS0_SELECT_INPUT
ECSPI5_SS1	SD1_DAT2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
		IOMUXC_ECSPI5_SS1_SELECT_INPUT
	SD2_DAT2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2
		IOMUXC_ECSPI5_SS1_SELECT_INPUT
ECSPI5_SS2	SD1_DAT3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
ECSPI5_SS3	SD2_DAT3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3
EIM - External Interface Module		
EIM_AD00	EIM_DA0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD00
EIM_AD01	EIM_DA1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD01
EIM_AD02	EIM_DA2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD02
EIM_AD03	EIM_DA3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD03
EIM_AD04	EIM_DA4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD04
EIM_AD05	EIM_DA5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD05
EIM_AD06	EIM_DA6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD06
EIM_AD07	EIM_DA7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD07
EIM_AD08	EIM_DA8 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD08
EIM_AD09	EIM_DA9 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD09
EIM_AD10	EIM_DA10 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD10
EIM_AD11	EIM_DA11 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD11
EIM_AD12	EIM_DA12 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD12
EIM_AD13	EIM_DA13 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD13
EIM_AD14	EIM_DA14 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD14
EIM_AD15	EIM_DA15 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD15

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
EIM_ADDR16	EIM_A16 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16
EIM_ADDR17	EIM_A17 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17
EIM_ADDR18	EIM_A18 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18
EIM_ADDR19	EIM_A19 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19
EIM_ADDR20	EIM_A20 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20
EIM_ADDR21	EIM_A21 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21
EIM_ADDR22	EIM_A22 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22
EIM_ADDR23	EIM_A23 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
EIM_ADDR24	EIM_A24 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
EIM_ADDR25	EIM_A25 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
EIM_ADDR26	NANDF_CS3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B
EIM_BCLK	EIM_BCLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK
EIM_CRE	NANDF_CS2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
EIM_CS0_B	EIM_CS0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B
EIM_CS1_B	EIM_CS1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B
EIM_CS2_B	DISP0_DAT18 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
	SD2_DAT1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1
EIM_CS3_B	DISP0_DAT19 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
	SD2_DAT2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2
EIM_DATA00	CSI0_DATA_EN (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN
EIM_DATA01	CSI0_VSYNC (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC
EIM_DATA02	CSI0_DAT4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04
EIM_DATA03	CSI0_DAT5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05
EIM_DATA04	CSI0_DAT6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06
EIM_DATA05	CSI0_DAT7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
EIM_DATA06	CSI0_DAT8 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
EIM_DATA07	CSI0_DAT9 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
EIM_DATA08	CSI0_DAT12 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12
EIM_DATA09	CSI0_DAT13 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13
EIM_DATA10	CSI0_DAT14 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14
EIM_DATA11	CSI0_DAT15 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15
EIM_DATA12	CSI0_DAT16 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16
EIM_DATA13	CSI0_DAT17 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17
EIM_DATA14	CSI0_DAT18 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18
EIM_DATA15	CSI0_DAT19 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19
EIM_DATA16	EIM_D16 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
EIM_DATA17	EIM_D17 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
EIM_DATA18	EIM_D18 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
EIM_DATA19	EIM_D19 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
EIM_DATA20	EIM_D20 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
EIM_DATA21	EIM_D21 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21
EIM_DATA22	EIM_D22 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
EIM_DATA23	EIM_D23 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
EIM_DATA24	EIM_D24 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
EIM_DATA25	EIM_D25 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
EIM_DATA26	EIM_D26 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
EIM_DATA27	EIM_D27 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
EIM_DATA28	EIM_D28 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
EIM_DATA29	EIM_D29 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
EIM_DATA30	EIM_D30 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
EIM_DATA31	EIM_D31 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
EIM_DTACK_B	EIM_WAIT (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B
EIM_EB0_B	EIM_EB0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B
EIM_EB1_B	EIM_EB1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B
EIM_EB2_B	EIM_EB2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
EIM_EB3_B	EIM_EB3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
EIM_LBA_B	EIM_LBA (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B
EIM_OE_B	EIM_OE (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B
EIM_RW	EIM_RW (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_RW
EIM_WAIT_B	EIM_WAIT (ALT0)	IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B
ENET - 10/100/1000-Mbps Ethernet MAC		
ENET_1588_EVENT0_IN	ENET_TXD1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1
ENET_1588_EVENT0_OUT	GPIO_19 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
ENET_1588_EVENT1_IN	ENET_MDC (ALT4)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDC
ENET_1588_EVENT1_OUT	ENET_MDIO (ALT4)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO
ENET_1588_EVENT2_IN	GPIO_16 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO16
ENET_1588_EVENT2_OUT	ENET_RX_ER (ALT4)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER
ENET_1588_EVENT3_IN	GPIO_17 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO17
ENET_1588_EVENT3_OUT	ENET_RXD1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1
ENET_COL	KEY_ROW1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
ENET_CRS	KEY_COL3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
ENET_MDC	ENET_MDC (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDC
	KEY_COL2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
ENET_MDIO	ENET_MDIO (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT
	KEY_COL1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1 IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT
ENET_REF_CLK	GPIO_16 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO16

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_ENET_REF_CLK_SELECT_INPUT
	RGMII_TX_CTL (ALT7)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL
		IOMUXC_ENET_REF_CLK_SELECT_INPUT
ENET_RX_CLK	GPIO_18 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO18
		IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT
ENET_RX_DATA0	ENET_RXD0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0
		IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT
ENET_RX_DATA1	ENET_RXD1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1
		IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT
ENET_RX_DATA2	KEY_COL2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
		IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT
ENET_RX_DATA3	KEY_COL0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
		IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT
ENET_RX_EN	ENET_CRS_DV (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV
		IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT
ENET_RX_ER	ENET_RX_ER (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER
ENET_TX_CLK	ENET_REF_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK
ENET_TX_DATA0	ENET_TXD0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0
ENET_TX_DATA1	ENET_TXD1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1
ENET_TX_DATA2	KEY_ROW2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
ENET_TX_DATA3	KEY_ROW0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
ENET_TX_EN	ENET_TX_EN (ALT1)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN
ENET_TX_ER	GPIO_19 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
RGMII_RD0	RGMII_RD0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0
		IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT
RGMII_RD1	RGMII_RD1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1
		IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT
RGMII_RD2	RGMII_RD2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2
		IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT
RGMII_RD3	RGMII_RD3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3
		IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT
RGMII_RXC	RGMII_RXC (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC
		IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT
RGMII_RX_CTL	RGMII_RX_CTL (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL
		IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT
RGMII_TD0	RGMII_TD0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0
RGMII_TD1	RGMII_TD1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1
RGMII_TD2	RGMII_TD2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2
RGMII_TD3	RGMII_TD3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
RGMII_TXC	RGMII_TXC (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC
RGMII_TX_CTL	RGMII_TX_CTL (ALT1)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL
EPIT1 - Enhanced Periodic Interrupt Timer		
EPIT1_OUT	EIM_D19 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19
	GPIO_0 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO00
	GPIO_7 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
EPIT2 - Enhanced Periodic Interrupt Timer		
EPIT2_OUT	EIM_D20 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
	GPIO_8 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
ESAI - Enhanced Serial Audio Interface		
ESAI_RX_CLK	ENET_MDIO (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO IOMUXC_ESAI_RX_CLK_SELECT_INPUT
	GPIO_1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO01 IOMUXC_ESAI_RX_CLK_SELECT_INPUT
ESAI_RX_FS	ENET_REF_CLK (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK IOMUXC_ESAI_RX_FS_SELECT_INPUT
	GPIO_9 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO09 IOMUXC_ESAI_RX_FS_SELECT_INPUT
ESAI_RX_HF_CLK	ENET_RX_ER (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT
	GPIO_3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO03 IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT
ESAI_TX0	GPIO_17 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO17 IOMUXC_ESAI_SDO0_SELECT_INPUT
	NANDF_CS2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B IOMUXC_ESAI_SDO0_SELECT_INPUT
ESAI_TX1	GPIO_18 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO18 IOMUXC_ESAI_SDO1_SELECT_INPUT
	NANDF_CS3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B IOMUXC_ESAI_SDO1_SELECT_INPUT
ESAI_TX2_RX3	ENET_TXD1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1 IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT
	GPIO_5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO05 IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT
ESAI_TX3_RX2	ENET_TX_EN (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT
	GPIO_16 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO16 IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT
ESAI_TX4_RX1	ENET_TXD0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT
	GPIO_7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
		IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT
ESAI_TX5_RX0	ENET_MDC (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDC
		IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT
	GPIO_8 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
		IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT
ESAI_TX_CLK	ENET_CRS_DV (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV
		IOMUXC_ESAI_TX_CLK_SELECT_INPUT
	GPIO_6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO06
		IOMUXC_ESAI_TX_CLK_SELECT_INPUT
ESAI_TX_FS	ENET_RXD1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1
		IOMUXC_ESAI_TX_FS_SELECT_INPUT
	GPIO_2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO02
		IOMUXC_ESAI_TX_FS_SELECT_INPUT
ESAI_TX_HF_CLK	ENET_RXD0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0
		IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT
	GPIO_4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO04
		IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT
FLEXCAN1 - Flexible Controller Area Network		
FLEXCAN1_RX	GPIO_8 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
		IOMUXC_FLEXCAN1_RX_SELECT_INPUT
	KEY_ROW2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
		IOMUXC_FLEXCAN1_RX_SELECT_INPUT
	SD3_CLK (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD3_CLK
		IOMUXC_FLEXCAN1_RX_SELECT_INPUT
FLEXCAN1_TX	GPIO_7 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
	KEY_COL2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
	SD3_CMD (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD3_CMD
FLEXCAN2 - Flexible Controller Area Network		
FLEXCAN2_RX	KEY_ROW4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
		IOMUXC_FLEXCAN2_RX_SELECT_INPUT
	SD3_DAT1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1
		IOMUXC_FLEXCAN2_RX_SELECT_INPUT
FLEXCAN2_TX	KEY_COL4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4
	SD3_DAT0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0
GPIO1 - General Purpose Input/Output		
GPIO1_IO00	GPIO_0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO00
GPIO1_IO01	GPIO_1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO01

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO1_IO02	GPIO_2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO02
GPIO1_IO03	GPIO_3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO03
GPIO1_IO04	GPIO_4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO04
GPIO1_IO05	GPIO_5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO05
GPIO1_IO06	GPIO_6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO06
GPIO1_IO07	GPIO_7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
GPIO1_IO08	GPIO_8 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
GPIO1_IO09	GPIO_9 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO09
GPIO1_IO10	SD2_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_CLK
GPIO1_IO11	SD2_CMD (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_CMD
GPIO1_IO12	SD2_DAT3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3
GPIO1_IO13	SD2_DAT2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2
GPIO1_IO14	SD2_DAT1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1
GPIO1_IO15	SD2_DAT0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0
GPIO1_IO16	SD1_DAT0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0
GPIO1_IO17	SD1_DAT1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1
GPIO1_IO18	SD1_CMD (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_CMD
GPIO1_IO19	SD1_DAT2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
GPIO1_IO20	SD1_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_CLK
GPIO1_IO21	SD1_DAT3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
GPIO1_IO22	ENET_MDIO (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO
GPIO1_IO23	ENET_REF_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK
GPIO1_IO24	ENET_RX_ER (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER
GPIO1_IO25	ENET_CRS_DV (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV
GPIO1_IO26	ENET_RXD1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1
GPIO1_IO27	ENET_RXD0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0
GPIO1_IO28	ENET_TX_EN (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN
GPIO1_IO29	ENET_TXD1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1
GPIO1_IO30	ENET_TXD0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0
GPIO1_IO31	ENET_MDC (ALT5)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDC
GPIO2 - General Purpose Input/Output		
GPIO2_IO00	NANDF_D0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00
GPIO2_IO01	NANDF_D1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01
GPIO2_IO02	NANDF_D2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02
GPIO2_IO03	NANDF_D3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03
GPIO2_IO04	NANDF_D4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04
GPIO2_IO05	NANDF_D5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05
GPIO2_IO06	NANDF_D6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06
GPIO2_IO07	NANDF_D7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO2_IO08	SD4_DAT0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0
GPIO2_IO09	SD4_DAT1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1
GPIO2_IO10	SD4_DAT2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2
GPIO2_IO11	SD4_DAT3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA3
GPIO2_IO12	SD4_DAT4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4
GPIO2_IO13	SD4_DAT5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5
GPIO2_IO14	SD4_DAT6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6
GPIO2_IO15	SD4_DAT7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7
GPIO2_IO16	EIM_A22 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22
GPIO2_IO17	EIM_A21 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21
GPIO2_IO18	EIM_A20 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20
GPIO2_IO19	EIM_A19 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19
GPIO2_IO20	EIM_A18 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18
GPIO2_IO21	EIM_A17 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17
GPIO2_IO22	EIM_A16 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16
GPIO2_IO23	EIM_CS0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B
GPIO2_IO24	EIM_CS1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B
GPIO2_IO25	EIM_OE (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B
GPIO2_IO26	EIM_RW (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_RW
GPIO2_IO27	EIM_LBA (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B
GPIO2_IO28	EIM_EB0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B
GPIO2_IO29	EIM_EB1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B
GPIO2_IO30	EIM_EB2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
GPIO2_IO31	EIM_EB3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
GPIO3 - General Purpose Input/Output		
GPIO3_IO00	EIM_DA0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD00
GPIO3_IO01	EIM_DA1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD01
GPIO3_IO02	EIM_DA2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD02
GPIO3_IO03	EIM_DA3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD03
GPIO3_IO04	EIM_DA4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD04
GPIO3_IO05	EIM_DA5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD05
GPIO3_IO06	EIM_DA6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD06
GPIO3_IO07	EIM_DA7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD07
GPIO3_IO08	EIM_DA8 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD08
GPIO3_IO09	EIM_DA9 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD09
GPIO3_IO10	EIM_DA10 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD10
GPIO3_IO11	EIM_DA11 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD11
GPIO3_IO12	EIM_DA12 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD12
GPIO3_IO13	EIM_DA13 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD13

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO3_IO14	EIM_DA14 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD14
GPIO3_IO15	EIM_DA15 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD15
GPIO3_IO16	EIM_D16 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
GPIO3_IO17	EIM_D17 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
GPIO3_IO18	EIM_D18 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
GPIO3_IO19	EIM_D19 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19
GPIO3_IO20	EIM_D20 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
GPIO3_IO21	EIM_D21 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21
GPIO3_IO22	EIM_D22 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
GPIO3_IO23	EIM_D23 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
GPIO3_IO24	EIM_D24 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
GPIO3_IO25	EIM_D25 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
GPIO3_IO26	EIM_D26 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
GPIO3_IO27	EIM_D27 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
GPIO3_IO28	EIM_D28 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
GPIO3_IO29	EIM_D29 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
GPIO3_IO30	EIM_D30 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
GPIO3_IO31	EIM_D31 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
GPIO4 - General Purpose Input/Output		
GPIO4_IO05	GPIO_19 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
GPIO4_IO06	KEY_COL0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
GPIO4_IO07	KEY_ROW0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
GPIO4_IO08	KEY_COL1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1
GPIO4_IO09	KEY_ROW1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
GPIO4_IO10	KEY_COL2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
GPIO4_IO11	KEY_ROW2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
GPIO4_IO12	KEY_COL3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
GPIO4_IO13	KEY_ROW3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
GPIO4_IO14	KEY_COL4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4
GPIO4_IO15	KEY_ROW4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
GPIO4_IO16	DI0_DISP_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK
GPIO4_IO17	DI0_PIN15 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15
GPIO4_IO18	DI0_PIN2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02
GPIO4_IO19	DI0_PIN3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03
GPIO4_IO20	DI0_PIN4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04
GPIO4_IO21	DISP0_DAT0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00
GPIO4_IO22	DISP0_DAT1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01
GPIO4_IO23	DISP0_DAT2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02
GPIO4_IO24	DISP0_DAT3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO4_IO25	DISP0_DAT4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04
GPIO4_IO26	DISP0_DAT5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05
GPIO4_IO27	DISP0_DAT6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06
GPIO4_IO28	DISP0_DAT7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07
GPIO4_IO29	DISP0_DAT8 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08
GPIO4_IO30	DISP0_DAT9 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09
GPIO4_IO31	DISP0_DAT10 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10
GPIO5 - General Purpose Input/Output		
GPIO5_IO00	EIM_WAIT (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B
GPIO5_IO02	EIM_A25 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
GPIO5_IO04	EIM_A24 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
GPIO5_IO05	DISP0_DAT11 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11
GPIO5_IO06	DISP0_DAT12 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12
GPIO5_IO07	DISP0_DAT13 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13
GPIO5_IO08	DISP0_DAT14 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14
GPIO5_IO09	DISP0_DAT15 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15
GPIO5_IO10	DISP0_DAT16 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16
GPIO5_IO11	DISP0_DAT17 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17
GPIO5_IO12	DISP0_DAT18 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
GPIO5_IO13	DISP0_DAT19 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
GPIO5_IO14	DISP0_DAT20 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20
GPIO5_IO15	DISP0_DAT21 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21
GPIO5_IO16	DISP0_DAT22 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22
GPIO5_IO17	DISP0_DAT23 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23
GPIO5_IO18	CSI0_PIXCLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK
GPIO5_IO19	CSI0_MCLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC
GPIO5_IO20	CSI0_DATA_EN (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN
GPIO5_IO21	CSI0_VSYNC (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC
GPIO5_IO22	CSI0_DAT4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04
GPIO5_IO23	CSI0_DAT5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05
GPIO5_IO24	CSI0_DAT6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06
GPIO5_IO25	CSI0_DAT7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
GPIO5_IO26	CSI0_DAT8 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
GPIO5_IO27	CSI0_DAT9 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
GPIO5_IO28	CSI0_DAT10 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10
GPIO5_IO29	CSI0_DAT11 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11
GPIO5_IO30	CSI0_DAT12 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12
GPIO5_IO31	CSI0_DAT13 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13
GPIO6 - General Purpose Input/Output		

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO6_IO00	CSI0_DAT14 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14
GPIO6_IO01	CSI0_DAT15 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15
GPIO6_IO02	CSI0_DAT16 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16
GPIO6_IO03	CSI0_DAT17 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17
GPIO6_IO04	CSI0_DAT18 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18
GPIO6_IO05	CSI0_DAT19 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19
GPIO6_IO06	EIM_A23 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
GPIO6_IO07	NANDF_CLE (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_CLE
GPIO6_IO08	NANDF_ALE (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_ALE
GPIO6_IO09	NANDF_WP_B (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B
GPIO6_IO10	NANDF_RB0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B
GPIO6_IO11	NANDF_CS0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS0_B
GPIO6_IO14	NANDF_CS1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B
GPIO6_IO15	NANDF_CS2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
GPIO6_IO16	NANDF_CS3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B
GPIO6_IO17	SD3_DAT7 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7
GPIO6_IO18	SD3_DAT6 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6
GPIO6_IO19	RGMII_TXC (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC
GPIO6_IO20	RGMII_TD0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0
GPIO6_IO21	RGMII_TD1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1
GPIO6_IO22	RGMII_TD2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2
GPIO6_IO23	RGMII_TD3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3
GPIO6_IO24	RGMII_RX_CTL (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL
GPIO6_IO25	RGMII_RD0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0
GPIO6_IO26	RGMII_TX_CTL (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL
GPIO6_IO27	RGMII_RD1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1
GPIO6_IO28	RGMII_RD2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2
GPIO6_IO29	RGMII_RD3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3
GPIO6_IO30	RGMII_RXC (ALT5)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC
GPIO6_IO31	EIM_BCLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK
GPIO7 - General Purpose Input/Output		
GPIO7_IO00	SD3_DAT5 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5
GPIO7_IO01	SD3_DAT4 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4
GPIO7_IO02	SD3_CMD (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_CMD
GPIO7_IO03	SD3_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_CLK
GPIO7_IO04	SD3_DAT0 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0
GPIO7_IO05	SD3_DAT1 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1
GPIO7_IO06	SD3_DAT2 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA2
GPIO7_IO07	SD3_DAT3 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
GPIO7_IO08	SD3_RST (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD3_RESET
GPIO7_IO09	SD4_CMD (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_CMD
GPIO7_IO10	SD4_CLK (ALT5)	IOMUXC_SW_MUX_CTL_PAD_SD4_CLK
GPIO7_IO11	GPIO_16 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO16
GPIO7_IO12	GPIO_17 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO17
GPIO7_IO13	GPIO_18 (ALT5)	IOMUXC_SW_MUX_CTL_PAD_GPIO18
GPMI - General Purpose Media Interface		
NAND_ALE	NANDF_ALE (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_ALE
NAND_CE0_B	NANDF_CS0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS0_B
NAND_CE1_B	NANDF_CS1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B
NAND_CE2_B	NANDF_CS2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
NAND_CE3_B	NANDF_CS3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B
NAND_CLE	NANDF_CLE (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_CLE
NAND_DATA00	NANDF_D0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00
NAND_DATA01	NANDF_D1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01
NAND_DATA02	NANDF_D2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02
NAND_DATA03	NANDF_D3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03
NAND_DATA04	NANDF_D4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04
NAND_DATA05	NANDF_D5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05
NAND_DATA06	NANDF_D6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06
NAND_DATA07	NANDF_D7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07
NAND_DQS	SD4_DAT0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0
NAND_READY_B	NANDF_RB0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B
NAND_RE_B	SD4_CMD (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_CMD
NAND_WE_B	SD4_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_CLK
NAND_WP_B	NANDF_WP_B (ALT0)	IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B
GPT - General Purpose Timer		
GPT_CAPTURE1	SD1_DAT0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0
GPT_CAPTURE2	SD1_DAT1 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1
GPT_CLKIN	SD1_CLK (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_CLK
GPT_COMPARE1	SD1_CMD (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_CMD
GPT_COMPARE2	SD1_DAT2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
GPT_COMPARE3	SD1_DAT3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
HDMI - High Definition Multimedia Interface		
HDMI_TX_CEC_LINE	EIM_A25 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25 IOMUXC_HDMI_ICECIN_SELECT_INPUT
	KEY_ROW2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2 IOMUXC_HDMI_ICECIN_SELECT_INPUT
HDMI_TX_CLK_N	HDMI_CLKM	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
HDMI_TX_CLK_P	HDMI_CLKP	Not multiplexed.
HDMI_TX_DATA0_N	HDMI_D0M	Not multiplexed.
HDMI_TX_DATA0_P	HDMI_D0P	Not multiplexed.
HDMI_TX_DATA1_N	HDMI_D1M	Not multiplexed.
HDMI_TX_DATA1_P	HDMI_D1P	Not multiplexed.
HDMI_TX_DATA2_N	HDMI_D2M	Not multiplexed.
HDMI_TX_DATA2_P	HDMI_D2P	Not multiplexed.
HDMI_TX_DDC_CEC	HDMI_DDCCEC	Not multiplexed.
HDMI_TX_DDC_SCL	EIM_EB2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
		IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT
	KEY_COL3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
		IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT
HDMI_TX_DDC_SDA	EIM_D16 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
		IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT
	KEY_ROW3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
		IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT
HDMI_TX_HPD	HDMI_HPD	Not multiplexed.
I2C1 - I2C Controller		
I2C1_SCL	CSI0_DAT9 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
		IOMUXC_I2C1_SCL_IN_SELECT_INPUT
	EIM_D21 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21
		IOMUXC_I2C1_SCL_IN_SELECT_INPUT
I2C1_SDA	CSI0_DAT8 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
		IOMUXC_I2C1_SDA_IN_SELECT_INPUT
	EIM_D28 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
		IOMUXC_I2C1_SDA_IN_SELECT_INPUT
I2C2 - I2C Controller		
I2C2_SCL	EIM_EB2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
		IOMUXC_I2C2_SCL_IN_SELECT_INPUT
	KEY_COL3 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
		IOMUXC_I2C2_SCL_IN_SELECT_INPUT
I2C2_SDA	EIM_D16 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
		IOMUXC_I2C2_SDA_IN_SELECT_INPUT
	KEY_ROW3 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
		IOMUXC_I2C2_SDA_IN_SELECT_INPUT
I2C3 - I2C Controller		
I2C3_SCL	EIM_D17 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
		IOMUXC_I2C3_SCL_IN_SELECT_INPUT
	GPIO_3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO03

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_I2C3_SCL_IN_SELECT_INPUT
	GPIO_5 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO05
		IOMUXC_I2C3_SCL_IN_SELECT_INPUT
I2C3_SDA	EIM_D18 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
		IOMUXC_I2C3_SDA_IN_SELECT_INPUT
	GPIO_6 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO06
		IOMUXC_I2C3_SDA_IN_SELECT_INPUT
	GPIO_16 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO16
		IOMUXC_I2C3_SDA_IN_SELECT_INPUT
IPU1 - Image Processing Unit		
IPU1_CSI0_DATA00	EIM_D27 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
IPU1_CSI0_DATA01	EIM_D26 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
IPU1_CSI0_DATA02	EIM_D31 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
IPU1_CSI0_DATA03	EIM_D30 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
IPU1_CSI0_DATA04	CSI0_DAT4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04
IPU1_CSI0_DATA05	CSI0_DAT5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05
IPU1_CSI0_DATA06	CSI0_DAT6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06
IPU1_CSI0_DATA07	CSI0_DAT7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07
IPU1_CSI0_DATA08	CSI0_DAT8 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08
IPU1_CSI0_DATA09	CSI0_DAT9 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
IPU1_CSI0_DATA10	CSI0_DAT10 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10
IPU1_CSI0_DATA11	CSI0_DAT11 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11
IPU1_CSI0_DATA12	CSI0_DAT12 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12
IPU1_CSI0_DATA13	CSI0_DAT13 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13
IPU1_CSI0_DATA14	CSI0_DAT14 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14
IPU1_CSI0_DATA15	CSI0_DAT15 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15
IPU1_CSI0_DATA16	CSI0_DAT16 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16
IPU1_CSI0_DATA17	CSI0_DAT17 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17
IPU1_CSI0_DATA18	CSI0_DAT18 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18
IPU1_CSI0_DATA19	CSI0_DAT19 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19
IPU1_CSI0_DATA_EN	CSI0_DATA_EN (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN
IPU1_CSI0_HSYNC	CSI0_MCLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC
IPU1_CSI0_PIXCLK	CSI0_PIXCLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK
IPU1_CSI0_VSYNC	CSI0_VSYNC (ALT0)	IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC
IPU1_DI0_D0_CS	EIM_D23 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
IPU1_DI0_D1_CS	EIM_A25 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
IPU1_DI0_DISP_CLK	DI0_DISP_CLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK
IPU1_DI0_PIN01	EIM_D22 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
IPU1_DI0_PIN02	DI0_PIN2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
IPU1_DI0_PIN03	DI0_PIN3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03
IPU1_DI0_PIN04	DI0_PIN4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04
IPU1_DI0_PIN05	EIM_D16 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
IPU1_DI0_PIN06	EIM_D17 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
IPU1_DI0_PIN07	EIM_D18 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
IPU1_DI0_PIN08	EIM_D19 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19
IPU1_DI0_PIN11	EIM_D30 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
IPU1_DI0_PIN12	EIM_D31 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
IPU1_DI0_PIN13	EIM_D28 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
IPU1_DI0_PIN14	EIM_D29 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
IPU1_DI0_PIN15	DI0_PIN15 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15
IPU1_DI0_PIN16	EIM_D20 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
IPU1_DI0_PIN17	EIM_D21 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21
IPU1_DI1_D0_CS	EIM_DA13 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD13
	EIM_D18 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
IPU1_DI1_D1_CS	EIM_DA14 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD14
IPU1_DI1_DISP_CLK	EIM_A16 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16
IPU1_DI1_PIN01	EIM_DA15 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD15
IPU1_DI1_PIN02	EIM_DA11 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD11
	EIM_D23 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
IPU1_DI1_PIN03	EIM_DA12 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD12
	EIM_EB3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
IPU1_DI1_PIN04	EIM_DA15 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD15
IPU1_DI1_PIN05	EIM_CS0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B
IPU1_DI1_PIN06	EIM_CS1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B
IPU1_DI1_PIN07	EIM_OE (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B
IPU1_DI1_PIN08	EIM_RW (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_RW
IPU1_DI1_PIN11	EIM_D26 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
IPU1_DI1_PIN12	EIM_A25 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25
IPU1_DI1_PIN13	EIM_D27 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
IPU1_DI1_PIN14	EIM_D23 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
IPU1_DI1_PIN15	EIM_DA10 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD10
	EIM_D29 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
IPU1_DI1_PIN16	EIM_BCLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK
IPU1_DI1_PIN17	EIM_LBA (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B
IPU1_DISP0_DATA00	DISP0_DAT0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00
IPU1_DISP0_DATA01	DISP0_DAT1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01
IPU1_DISP0_DATA02	DISP0_DAT2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02
IPU1_DISP0_DATA03	DISP0_DAT3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
IPU1_DISP0_DATA04	DISP0_DAT4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04
IPU1_DISP0_DATA05	DISP0_DAT5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05
IPU1_DISP0_DATA06	DISP0_DAT6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06
IPU1_DISP0_DATA07	DISP0_DAT7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07
IPU1_DISP0_DATA08	DISP0_DAT8 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08
IPU1_DISP0_DATA09	DISP0_DAT9 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09
IPU1_DISP0_DATA10	DISP0_DAT10 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10
IPU1_DISP0_DATA11	DISP0_DAT11 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11
IPU1_DISP0_DATA12	DISP0_DAT12 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12
IPU1_DISP0_DATA13	DISP0_DAT13 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13
IPU1_DISP0_DATA14	DISP0_DAT14 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14
IPU1_DISP0_DATA15	DISP0_DAT15 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15
IPU1_DISP0_DATA16	DISP0_DAT16 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16
IPU1_DISP0_DATA17	DISP0_DAT17 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17
IPU1_DISP0_DATA18	DISP0_DAT18 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
IPU1_DISP0_DATA19	DISP0_DAT19 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
IPU1_DISP0_DATA20	DISP0_DAT20 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20
IPU1_DISP0_DATA21	DISP0_DAT21 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21
IPU1_DISP0_DATA22	DISP0_DAT22 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22
IPU1_DISP0_DATA23	DISP0_DAT23 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23
IPU1_DISP1_DATA00	EIM_DA9 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD09
IPU1_DISP1_DATA01	EIM_DA8 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD08
IPU1_DISP1_DATA02	EIM_DA7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD07
IPU1_DISP1_DATA03	EIM_DA6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD06
IPU1_DISP1_DATA04	EIM_DA5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD05
IPU1_DISP1_DATA05	EIM_DA4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD04
IPU1_DISP1_DATA06	EIM_DA3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD03
IPU1_DISP1_DATA07	EIM_DA2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD02
IPU1_DISP1_DATA08	EIM_DA1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD01
IPU1_DISP1_DATA09	EIM_DA0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD00
IPU1_DISP1_DATA10	EIM_EB1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B
IPU1_DISP1_DATA11	EIM_EB0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B
IPU1_DISP1_DATA12	EIM_A17 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17
IPU1_DISP1_DATA13	EIM_A18 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18
IPU1_DISP1_DATA14	EIM_A19 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19
IPU1_DISP1_DATA15	EIM_A20 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20
IPU1_DISP1_DATA16	EIM_A21 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21
IPU1_DISP1_DATA17	EIM_A22 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22
IPU1_DISP1_DATA18	EIM_A23 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
IPU1_DISP1_DATA19	EIM_A24 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
IPU1_DISP1_DATA20	EIM_D31 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
IPU1_DISP1_DATA21	EIM_D30 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
IPU1_DISP1_DATA22	EIM_D26 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
IPU1_DISP1_DATA23	EIM_D27 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
IPU1_EXT_TRIG	EIM_D28 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28
IPU1_SISG0	NANDF_CS2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
IPU1_SISG1	NANDF_CS3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B
IPU1_SISG2	EIM_A24 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
	EIM_D26 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
IPU1_SISG3	EIM_A23 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
	EIM_D27 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
IPU1_SISG4	KEY_COL4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4
IPU1_SISG5	KEY_ROW4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
IPU2 - Image Processing Unit		
IPU2_CSI1_DATA00	EIM_DA9 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD09
IPU2_CSI1_DATA01	EIM_DA8 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD08
IPU2_CSI1_DATA02	EIM_DA7 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD07
IPU2_CSI1_DATA03	EIM_DA6 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD06
IPU2_CSI1_DATA04	EIM_DA5 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD05
IPU2_CSI1_DATA05	EIM_DA4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD04
IPU2_CSI1_DATA06	EIM_DA3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD03
IPU2_CSI1_DATA07	EIM_DA2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD02
IPU2_CSI1_DATA08	EIM_DA1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD01
IPU2_CSI1_DATA09	EIM_DA0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD00
IPU2_CSI1_DATA10	EIM_D22 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22 IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT
	EIM_EB1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT
IPU2_CSI1_DATA11	EIM_D21 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21 IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT
	EIM_EB0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT
IPU2_CSI1_DATA12	EIM_A17 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17 IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT
	EIM_D28 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28 IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT
IPU2_CSI1_DATA13	EIM_A18 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18 IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
	EIM_D27 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27
		IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT
IPU2_CSI1_DATA14	EIM_A19 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19
		IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT
	EIM_D26 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
		IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT
IPU2_CSI1_DATA15	EIM_A20 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20
		IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT
	EIM_D20 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20
		IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT
IPU2_CSI1_DATA16	EIM_A21 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21
		IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT
	EIM_D19 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19
		IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT
IPU2_CSI1_DATA17	EIM_A22 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22
		IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT
	EIM_D18 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18
		IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT
IPU2_CSI1_DATA18	EIM_A23 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
		IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT
	EIM_D16 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16
		IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT
IPU2_CSI1_DATA19	EIM_A24 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
		IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT
	EIM_EB2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
		IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT
IPU2_CSI1_DATA_EN	EIM_DA10 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD10
		IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT
	EIM_D23 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
		IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT
IPU2_CSI1_HSYNC	EIM_DA11 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD11
		IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT
	EIM_EB3 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
		IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT
IPU2_CSI1_PIXCLK	EIM_A16 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16
		IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT
	EIM_D17 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17
		IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
IPU2_CSI1_VSYNC	EIM_DA12 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD12
		IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT
	EIM_D29 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29
		IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT
IPU2_DI0_DISP_CLK	DI0_DISP_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK
IPU2_DI0_PIN01	NANDF_RB0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B
IPU2_DI0_PIN02	DI0_PIN2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02
IPU2_DI0_PIN03	DI0_PIN3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03
IPU2_DI0_PIN04	DI0_PIN4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04
IPU2_DI0_PIN15	DI0_PIN15 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15
IPU2_DISP0_DATA00	DISP0_DAT0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00
IPU2_DISP0_DATA01	DISP0_DAT1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01
IPU2_DISP0_DATA02	DISP0_DAT2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02
IPU2_DISP0_DATA03	DISP0_DAT3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03
IPU2_DISP0_DATA04	DISP0_DAT4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04
IPU2_DISP0_DATA05	DISP0_DAT5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05
IPU2_DISP0_DATA06	DISP0_DAT6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06
IPU2_DISP0_DATA07	DISP0_DAT7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07
IPU2_DISP0_DATA08	DISP0_DAT8 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08
IPU2_DISP0_DATA09	DISP0_DAT9 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09
IPU2_DISP0_DATA10	DISP0_DAT10 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10
IPU2_DISP0_DATA11	DISP0_DAT11 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11
IPU2_DISP0_DATA12	DISP0_DAT12 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12
IPU2_DISP0_DATA13	DISP0_DAT13 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13
IPU2_DISP0_DATA14	DISP0_DAT14 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14
IPU2_DISP0_DATA15	DISP0_DAT15 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15
IPU2_DISP0_DATA16	DISP0_DAT16 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16
IPU2_DISP0_DATA17	DISP0_DAT17 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17
IPU2_DISP0_DATA18	DISP0_DAT18 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18
IPU2_DISP0_DATA19	DISP0_DAT19 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19
IPU2_DISP0_DATA20	DISP0_DAT20 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20
IPU2_DISP0_DATA21	DISP0_DAT21 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21
IPU2_DISP0_DATA22	DISP0_DAT22 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22
IPU2_DISP0_DATA23	DISP0_DAT23 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23
IPU2_SISG0	NANDF_CS2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B
IPU2_SISG1	NANDF_CS3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B
IPU2_SISG2	EIM_A24 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
IPU2_SISG3	EIM_A23 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
IPU2_SISG4	NANDF_CLE (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_CLE

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
IPU2_SISG5	NANDF_WP_B (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B
KPP - Keypad Port		
KEY_COL0	KEY_COL0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
KEY_COL1	KEY_COL1 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1
KEY_COL2	KEY_COL2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
KEY_COL3	KEY_COL3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3
KEY_COL4	KEY_COL4 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4
KEY_COL5	CSI0_DAT4 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04 IOMUXC_KEY_COL5_SELECT_INPUT
	GPIO_0 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO00 IOMUXC_KEY_COL5_SELECT_INPUT
	GPIO_19 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_GPIO19 IOMUXC_KEY_COL5_SELECT_INPUT
	SD2_CLK (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD2_CLK IOMUXC_KEY_COL5_SELECT_INPUT
KEY_COL6	CSI0_DAT6 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06 IOMUXC_KEY_COL6_SELECT_INPUT
	GPIO_9 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO09 IOMUXC_KEY_COL6_SELECT_INPUT
	SD2_DAT3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3 IOMUXC_KEY_COL6_SELECT_INPUT
KEY_COL7	CSI0_DAT8 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08 IOMUXC_KEY_COL7_SELECT_INPUT
	GPIO_4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO04 IOMUXC_KEY_COL7_SELECT_INPUT
	SD2_DAT1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1 IOMUXC_KEY_COL7_SELECT_INPUT
KEY_ROW0	KEY_ROW0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
KEY_ROW1	KEY_ROW1 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
KEY_ROW2	KEY_ROW2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
KEY_ROW3	KEY_ROW3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
KEY_ROW4	KEY_ROW4 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
KEY_ROW5	CSI0_DAT5 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05 IOMUXC_KEY_ROW5_SELECT_INPUT
	GPIO_1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO01 IOMUXC_KEY_ROW5_SELECT_INPUT
	SD2_CMD (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD2_CMD IOMUXC_KEY_ROW5_SELECT_INPUT
KEY_ROW6	CSI0_DAT7 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
		IOMUXC_KEY_ROW6_SELECT_INPUT
	GPIO_2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO02
		IOMUXC_KEY_ROW6_SELECT_INPUT
	SD2_DAT2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2
		IOMUXC_KEY_ROW6_SELECT_INPUT
KEY_ROW7	CSI0_DAT9 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09
		IOMUXC_KEY_ROW7_SELECT_INPUT
	GPIO_5 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO05
		IOMUXC_KEY_ROW7_SELECT_INPUT
	SD2_DAT0 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0
		IOMUXC_KEY_ROW7_SELECT_INPUT
LDB - LVDS Display Bridge		
LVDS0_CLK_N	LVDS0_CLK_N	Not multiplexed.
LVDS0_CLK_P	LVDS0_CLK_P	Not multiplexed.
LVDS0_DATA0_N	LVDS0_TX0_N	Not multiplexed.
LVDS0_DATA0_P	LVDS0_TX0_P	Not multiplexed.
LVDS0_DATA1_N	LVDS0_TX1_N	Not multiplexed.
LVDS0_DATA1_P	LVDS0_TX1_P	Not multiplexed.
LVDS0_DATA2_N	LVDS0_TX2_N	Not multiplexed.
LVDS0_DATA2_P	LVDS0_TX2_P	Not multiplexed.
LVDS0_DATA3_N	LVDS0_TX3_N	Not multiplexed.
LVDS0_DATA3_P	LVDS0_TX3_P	Not multiplexed.
LVDS1_CLK_N	LVDS1_CLK_N	Not multiplexed.
LVDS1_CLK_P	LVDS1_CLK_P	Not multiplexed.
LVDS1_DATA0_N	LVDS1_TX0_N	Not multiplexed.
LVDS1_DATA0_P	LVDS1_TX0_P	Not multiplexed.
LVDS1_DATA1_N	LVDS1_TX1_N	Not multiplexed.
LVDS1_DATA1_P	LVDS1_TX1_P	Not multiplexed.
LVDS1_DATA2_N	LVDS1_TX2_N	Not multiplexed.
LVDS1_DATA2_P	LVDS1_TX2_P	Not multiplexed.
LVDS1_DATA3_N	LVDS1_TX3_N	Not multiplexed.
LVDS1_DATA3_P	LVDS1_TX3_P	Not multiplexed.
MIPI_CSI - MIPI Camera Serial Interface		
CSI_CLK0_N	CSI_CLK0M	Not multiplexed.
CSI_CLK0_P	CSI_CLK0P	Not multiplexed.
CSI_DATA0_N	CSI_D0M	Not multiplexed.
CSI_DATA0_P	CSI_D0P	Not multiplexed.
CSI_DATA1_N	CSI_D1M	Not multiplexed.
CSI_DATA1_P	CSI_D1P	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
CSI_DATA2_N	CSI_D2M	Not multiplexed.
CSI_DATA2_P	CSI_D2P	Not multiplexed.
CSI_DATA3_N	CSI_D3M	Not multiplexed.
CSI_DATA3_P	CSI_D3P	Not multiplexed.
MIPI_DSI - MIPI Display Serial Interface		
DSI_CLK0_N	DSI_CLK0M	Not multiplexed.
DSI_CLK0_P	DSI_CLK0P	Not multiplexed.
DSI_DATA0_N	DSI_D0M	Not multiplexed.
DSI_DATA0_P	DSI_D0P	Not multiplexed.
DSI_DATA1_N	DSI_D1M	Not multiplexed.
DSI_DATA1_P	DSI_D1P	Not multiplexed.
MIPI_HSI - MIPI High Speed Synchronous Interface		
HSI_RX_DATA	RGMII_TD2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2
HSI_RX_FLAG	RGMII_TD1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1
HSI_RX_READY	RGMII_RD0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0
HSI_RX_WAKE	RGMII_TD3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3
HSI_TX_DATA	RGMII_RD2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2
HSI_TX_FLAG	RGMII_RD1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1
HSI_TX_READY	RGMII_TD0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0
HSI_TX_WAKE	RGMII_RD3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3
MLB - Media Local Bus		
MLB_CLK	ENET_TXD1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1 IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT
	GPIO_3 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO03 IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT
MLB_CLK_N	MLB_CN	Not multiplexed.
MLB_CLK_P	MLB_CP	Not multiplexed.
MLB_DATA	ENET_MDC (ALT0)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDC IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT
	GPIO_2 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO02 IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT
MLB_DATA_N	MLB_DN	Not multiplexed.
MLB_DATA_P	MLB_DP	Not multiplexed.
MLB_SIG	ENET_RXD1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1 IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT
	GPIO_6 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO06 IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT
MLB_SIG_N	MLB_SN	Not multiplexed.
MLB_SIG_P	MLB_SP	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
MMDC - Multi Mode DDR Controller		
DRAM_ADDR00	DRAM_A0	Not multiplexed.
DRAM_ADDR01	DRAM_A1	Not multiplexed.
DRAM_ADDR02	DRAM_A2	Not multiplexed.
DRAM_ADDR03	DRAM_A3	Not multiplexed.
DRAM_ADDR04	DRAM_A4	Not multiplexed.
DRAM_ADDR05	DRAM_A5	Not multiplexed.
DRAM_ADDR06	DRAM_A6	Not multiplexed.
DRAM_ADDR07	DRAM_A7	Not multiplexed.
DRAM_ADDR08	DRAM_A8	Not multiplexed.
DRAM_ADDR09	DRAM_A9	Not multiplexed.
DRAM_ADDR10	DRAM_A10	Not multiplexed.
DRAM_ADDR11	DRAM_A11	Not multiplexed.
DRAM_ADDR12	DRAM_A12	Not multiplexed.
DRAM_ADDR13	DRAM_A13	Not multiplexed.
DRAM_ADDR14	DRAM_A14	Not multiplexed.
DRAM_ADDR15	DRAM_A15	Not multiplexed.
DRAM_CAS_B	DRAM_CAS	Not multiplexed.
DRAM_CS0_B	DRAM_CS0	Not multiplexed.
DRAM_CS1_B	DRAM_CS1	Not multiplexed.
DRAM_DATA00	DRAM_D0	Not multiplexed.
DRAM_DATA01	DRAM_D1	Not multiplexed.
DRAM_DATA02	DRAM_D2	Not multiplexed.
DRAM_DATA03	DRAM_D3	Not multiplexed.
DRAM_DATA04	DRAM_D4	Not multiplexed.
DRAM_DATA05	DRAM_D5	Not multiplexed.
DRAM_DATA06	DRAM_D6	Not multiplexed.
DRAM_DATA07	DRAM_D7	Not multiplexed.
DRAM_DATA08	DRAM_D8	Not multiplexed.
DRAM_DATA09	DRAM_D9	Not multiplexed.
DRAM_DATA10	DRAM_D10	Not multiplexed.
DRAM_DATA11	DRAM_D11	Not multiplexed.
DRAM_DATA12	DRAM_D12	Not multiplexed.
DRAM_DATA13	DRAM_D13	Not multiplexed.
DRAM_DATA14	DRAM_D14	Not multiplexed.
DRAM_DATA15	DRAM_D15	Not multiplexed.
DRAM_DATA16	DRAM_D16	Not multiplexed.
DRAM_DATA17	DRAM_D17	Not multiplexed.
DRAM_DATA18	DRAM_D18	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
DRAM_DATA19	DRAM_D19	Not multiplexed.
DRAM_DATA20	DRAM_D20	Not multiplexed.
DRAM_DATA21	DRAM_D21	Not multiplexed.
DRAM_DATA22	DRAM_D22	Not multiplexed.
DRAM_DATA23	DRAM_D23	Not multiplexed.
DRAM_DATA24	DRAM_D24	Not multiplexed.
DRAM_DATA25	DRAM_D25	Not multiplexed.
DRAM_DATA26	DRAM_D26	Not multiplexed.
DRAM_DATA27	DRAM_D27	Not multiplexed.
DRAM_DATA28	DRAM_D28	Not multiplexed.
DRAM_DATA29	DRAM_D29	Not multiplexed.
DRAM_DATA30	DRAM_D30	Not multiplexed.
DRAM_DATA31	DRAM_D31	Not multiplexed.
DRAM_DATA32	DRAM_D32	Not multiplexed.
DRAM_DATA33	DRAM_D33	Not multiplexed.
DRAM_DATA34	DRAM_D34	Not multiplexed.
DRAM_DATA35	DRAM_D35	Not multiplexed.
DRAM_DATA36	DRAM_D36	Not multiplexed.
DRAM_DATA37	DRAM_D37	Not multiplexed.
DRAM_DATA38	DRAM_D38	Not multiplexed.
DRAM_DATA39	DRAM_D39	Not multiplexed.
DRAM_DATA40	DRAM_D40	Not multiplexed.
DRAM_DATA41	DRAM_D41	Not multiplexed.
DRAM_DATA42	DRAM_D42	Not multiplexed.
DRAM_DATA43	DRAM_D43	Not multiplexed.
DRAM_DATA44	DRAM_D44	Not multiplexed.
DRAM_DATA45	DRAM_D45	Not multiplexed.
DRAM_DATA46	DRAM_D46	Not multiplexed.
DRAM_DATA47	DRAM_D47	Not multiplexed.
DRAM_DATA48	DRAM_D48	Not multiplexed.
DRAM_DATA49	DRAM_D49	Not multiplexed.
DRAM_DATA50	DRAM_D50	Not multiplexed.
DRAM_DATA51	DRAM_D51	Not multiplexed.
DRAM_DATA52	DRAM_D52	Not multiplexed.
DRAM_DATA53	DRAM_D53	Not multiplexed.
DRAM_DATA54	DRAM_D54	Not multiplexed.
DRAM_DATA55	DRAM_D55	Not multiplexed.
DRAM_DATA56	DRAM_D56	Not multiplexed.
DRAM_DATA57	DRAM_D57	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
DRAM_DATA58	DRAM_D58	Not multiplexed.
DRAM_DATA59	DRAM_D59	Not multiplexed.
DRAM_DATA60	DRAM_D60	Not multiplexed.
DRAM_DATA61	DRAM_D61	Not multiplexed.
DRAM_DATA62	DRAM_D62	Not multiplexed.
DRAM_DATA63	DRAM_D63	Not multiplexed.
DRAM_DQM0	DRAM_DQM0	Not multiplexed.
DRAM_DQM1	DRAM_DQM1	Not multiplexed.
DRAM_DQM2	DRAM_DQM2	Not multiplexed.
DRAM_DQM3	DRAM_DQM3	Not multiplexed.
DRAM_DQM4	DRAM_DQM4	Not multiplexed.
DRAM_DQM5	DRAM_DQM5	Not multiplexed.
DRAM_DQM6	DRAM_DQM6	Not multiplexed.
DRAM_DQM7	DRAM_DQM7	Not multiplexed.
DRAM_ODT0	DRAM_SDODT0	Not multiplexed.
DRAM_ODT1	DRAM_SDODT1	Not multiplexed.
DRAM_RAS_B	DRAM_RAS	Not multiplexed.
DRAM_RESET	DRAM_RESET	Not multiplexed.
DRAM_SDBA0	DRAM_SDBA0	Not multiplexed.
DRAM_SDBA1	DRAM_SDBA1	Not multiplexed.
DRAM_SDBA2	DRAM_SDBA2	Not multiplexed.
DRAM_SDCKE0	DRAM_SDCKE0	Not multiplexed.
DRAM_SDCKE1	DRAM_SDCKE1	Not multiplexed.
DRAM_SDCLK0_N	DRAM_SDCLK_0_B	Not multiplexed.
DRAM_SDCLK0_P	DRAM_SDCLK_0	Not multiplexed.
DRAM_SDCLK1_N	DRAM_SDCLK_1_B	Not multiplexed.
DRAM_SDCLK1_P	DRAM_SDCLK_1	Not multiplexed.
DRAM_SDQS0_N	DRAM_SDQS0_B	Not multiplexed.
DRAM_SDQS0_P	DRAM_SDQS0	Not multiplexed.
DRAM_SDQS1_N	DRAM_SDQS1_B	Not multiplexed.
DRAM_SDQS1_P	DRAM_SDQS1	Not multiplexed.
DRAM_SDQS2_N	DRAM_SDQS2_B	Not multiplexed.
DRAM_SDQS2_P	DRAM_SDQS2	Not multiplexed.
DRAM_SDQS3_N	DRAM_SDQS3_B	Not multiplexed.
DRAM_SDQS3_P	DRAM_SDQS3	Not multiplexed.
DRAM_SDQS4_N	DRAM_SDQS4_B	Not multiplexed.
DRAM_SDQS4_P	DRAM_SDQS4	Not multiplexed.
DRAM_SDQS5_N	DRAM_SDQS5_B	Not multiplexed.
DRAM_SDQS5_P	DRAM_SDQS5	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
DRAM_SDQS6_N	DRAM_SDQS6_B	Not multiplexed.
DRAM_SDQS6_P	DRAM_SDQS6	Not multiplexed.
DRAM_SDQS7_N	DRAM_SDQS7_B	Not multiplexed.
DRAM_SDQS7_P	DRAM_SDQS7	Not multiplexed.
DRAM_SDWE_B	DRAM_SDWE	Not multiplexed.
PCIE - PCI Express		
PCIE_RX_N	PCIE_RXM	Not multiplexed.
PCIE_RX_P	PCIE_RXP	Not multiplexed.
PCIE_TX_N	PCIE_TXM	Not multiplexed.
PCIE_TX_P	PCIE_TXP	Not multiplexed.
PMU - Power Management Unit		
CSI_REXT	CSI_REXT	Not multiplexed.
DRAM_VREF	DRAM_VREF	Not multiplexed.
DRAM_ZQPAD	ZQPAD	Not multiplexed.
DSI_REXT	DSI_REXT	Not multiplexed.
GND	GND	Not multiplexed.
HDMI_REF	HDMI_REF	Not multiplexed.
HDMI_VP	HDMI_VP	Not multiplexed.
HDMI_VPH	HDMI_VPH	Not multiplexed.
NC	NC	Not multiplexed.
NVCC_CSI	NVCC_CSI	Not multiplexed.
NVCC_DRAM	NVCC_DRAM	Not multiplexed.
NVCC_EIM0	NVCC_EIM0	Not multiplexed.
NVCC_EIM1	NVCC_EIM1	Not multiplexed.
NVCC_EIM2	NVCC_EIM2	Not multiplexed.
NVCC_ENET	NVCC_ENET	Not multiplexed.
NVCC_GPIO	NVCC_GPIO	Not multiplexed.
NVCC_JTAG	NVCC_JTAG	Not multiplexed.
NVCC_LCD	NVCC_LCD	Not multiplexed.
NVCC_LVDS_2P5	NVCC_LVDS2P5	Not multiplexed.
NVCC_MIPI	NVCC_MIPI	Not multiplexed.
NVCC_NAND	NVCC_NANDF	Not multiplexed.
NVCC_PLL	NVCC_PLL_OUT	Not multiplexed.
NVCC_RGMII	NVCC_RGMII	Not multiplexed.
NVCC_SD1	NVCC_SD1	Not multiplexed.
NVCC_SD2	NVCC_SD2	Not multiplexed.
NVCC_SD3	NVCC_SD3	Not multiplexed.
PCIE_REXT	PCIE_REXT	Not multiplexed.
PCIE_VP	PCIE_VP	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
PCIE_VPH	PCIE_VPH	Not multiplexed.
PCIE_VPTX	PCIE_VPTX	Not multiplexed.
SATA_REXT	SATA_REXT	Not multiplexed.
SATA_VP	SATA_VP	Not multiplexed.
SATA_VPH	SATA_VPH	Not multiplexed.
USB_H1_VBUS	USB_H1_VBUS	Not multiplexed.
USB_OTG_VBUS	USB_OTG_VBUS	Not multiplexed.
VDD_ARM23_CAP	VDDARM23_CAP	Not multiplexed.
VDD_ARM23_IN	VDDARM23_IN	Not multiplexed.
VDD_ARM_CAP	VDDARM_CAP	Not multiplexed.
VDD_ARM_IN	VDDARM_IN	Not multiplexed.
VDD_CACHE_CAP	VDD_CACHE_CAP	Not multiplexed.
VDD_HIGH_CAP	VDDHIGH_CAP	Not multiplexed.
VDD_HIGH_IN	VDDHIGH_IN	Not multiplexed.
VDD_PU_CAP	VDDPU_CAP	Not multiplexed.
VDD_SNVS_CAP	VDD_SNVS_CAP	Not multiplexed.
VDD_SNVS_IN	VDD_SNVS_IN	Not multiplexed.
VDD_SOC_CAP	VDDSOC_CAP	Not multiplexed.
VDD_SOC_IN	VDDSOC_IN	Not multiplexed.
VDD_USB_CAP	VDDUSB_CAP	Not multiplexed.
PWM1 - Pulse Width Modulation		
PWM1_OUT	DISP0_DAT8 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08
	GPIO_9 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO09
	SD1_DAT3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
PWM2 - Pulse Width Modulation		
PWM2_OUT	DISP0_DAT9 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09
	GPIO_1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO01
	SD1_DAT2 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
PWM3 - Pulse Width Modulation		
PWM3_OUT	SD1_DAT1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1
	SD4_DAT1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1
PWM4 - Pulse Width Modulation		
PWM4_OUT	SD1_CMD (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD1_CMD
	SD4_DAT2 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2
SATA_PHY - Serial Advanced Technology Attachment PHY		
SATA_PHY_RX_N	SATA_RXM	Not multiplexed.
SATA_PHY_RX_P	SATA_RXP	Not multiplexed.
SATA_PHY_TX_N	SATA_TXM	Not multiplexed.
SATA_PHY_TX_P	SATA_TXP	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
SDMA - Smart Direct Memory Access Controller		
SDMA_EXT_EVENT0	DISP0_DAT16 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16 IOMUXC_SDMA_EVENTS14_SELECT_INPUT
	GPIO_17 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO17 IOMUXC_SDMA_EVENTS14_SELECT_INPUT
SDMA_EXT_EVENT1	DISP0_DAT17 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17 IOMUXC_SDMA_EVENTS15_SELECT_INPUT
	GPIO_18 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO18 IOMUXC_SDMA_EVENTS15_SELECT_INPUT
SJC - System JTAG Controller		
JTAG_DE_B	GPIO_16 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO16
JTAG_MOD	JTAG_MOD	Not multiplexed.
JTAG_TCK	JTAG_TCK	Not multiplexed.
JTAG_TDI	JTAG_TDI	Not multiplexed.
JTAG_TDO	JTAG_TDO	Not multiplexed.
JTAG_TMS	JTAG_TMS	Not multiplexed.
JTAG_TRSTB	JTAG_TRSTB	Not multiplexed.
SNVS - Secure Non-Volatile Storage		
SNVS_PMIC_ON_REQ	PMIC_ON_REQ	Not multiplexed.
SNVS_TAMPER	TAMPER	Not multiplexed.
SNVS_VIO_5	GPIO_0 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO00
SNVS_VIO_5_CTL	GPIO_18 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO18
SPDIF - Sony/Philips Digital Interface		
SPDIF_EXT_CLK	ENET_CRS_DV (ALT3)	IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT
	RGMII_TXC (ALT2)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT
SPDIF_IN	EIM_D21 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21 IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT
	ENET_RX_ER (ALT3)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT
	GPIO_16 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO16 IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT
	KEY_COL3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL3 IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT
SPDIF_LOCK	ENET_MDIO (ALT6)	IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO
	GPIO_7 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
SPDIF_OUT	EIM_D22 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
	ENET_RXD0 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
	GPIO_17 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO17
	GPIO_19 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO19
SPDIF_SR_CLK	ENET_REF_CLK (ALT6)	IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK
	GPIO_8 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
SRC - System Reset Controller		
SRC_BOOT_CFG00	EIM_DA0 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD00
SRC_BOOT_CFG01	EIM_DA1 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD01
SRC_BOOT_CFG02	EIM_DA2 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD02
SRC_BOOT_CFG03	EIM_DA3 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD03
SRC_BOOT_CFG04	EIM_DA4 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD04
SRC_BOOT_CFG05	EIM_DA5 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD05
SRC_BOOT_CFG06	EIM_DA6 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD06
SRC_BOOT_CFG07	EIM_DA7 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD07
SRC_BOOT_CFG08	EIM_DA8 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD08
SRC_BOOT_CFG09	EIM_DA9 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD09
SRC_BOOT_CFG10	EIM_DA10 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD10
SRC_BOOT_CFG11	EIM_DA11 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD11
SRC_BOOT_CFG12	EIM_DA12 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD12
SRC_BOOT_CFG13	EIM_DA13 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD13
SRC_BOOT_CFG14	EIM_DA14 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD14
SRC_BOOT_CFG15	EIM_DA15 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_AD15
SRC_BOOT_CFG16	EIM_A16 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16
SRC_BOOT_CFG17	EIM_A17 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17
SRC_BOOT_CFG18	EIM_A18 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18
SRC_BOOT_CFG19	EIM_A19 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19
SRC_BOOT_CFG20	EIM_A20 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20
SRC_BOOT_CFG21	EIM_A21 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21
SRC_BOOT_CFG22	EIM_A22 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22
SRC_BOOT_CFG23	EIM_A23 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23
SRC_BOOT_CFG24	EIM_A24 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24
SRC_BOOT_CFG25	EIM_WAIT (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B
SRC_BOOT_CFG26	EIM_LBA (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B
SRC_BOOT_CFG27	EIM_EB0 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B
SRC_BOOT_CFG28	EIM_EB1 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B
SRC_BOOT_CFG29	EIM_RW (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_RW
SRC_BOOT_CFG30	EIM_EB2 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B
SRC_BOOT_CFG31	EIM_EB3 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
SRC_BOOT_MODE0	BOOT_MODE0	Not multiplexed.
SRC_BOOT_MODE1	BOOT_MODE1	Not multiplexed.

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
SRC_ONOFF	ONOFF	Not multiplexed.
SRC_POR_B	POR_B	Not multiplexed.
UART1 - Universal Asynchronous Receiver/Transmitter		
UART1_CTS_B	EIM_D19 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19 IOMUXC_UART1_UART_RTS_B_SELECT_INPUT
	SD3_DAT0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0 IOMUXC_UART1_UART_RTS_B_SELECT_INPUT
UART1_DCD_B	EIM_D23 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
UART1_DSR_B	EIM_D25 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
UART1_DTR_B	EIM_D24 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
UART1_RI_B	EIM_EB3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
UART1_RTS_B	EIM_D20 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20 IOMUXC_UART1_UART_RTS_B_SELECT_INPUT
	SD3_DAT1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1 IOMUXC_UART1_UART_RTS_B_SELECT_INPUT
UART1_RX_DATA	CSI0_DAT11 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11 IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT
	SD3_DAT6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6 IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT
UART1_TX_DATA	CSI0_DAT10 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10 IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT
	SD3_DAT7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7 IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT
UART2 - Universal Asynchronous Receiver/Transmitter		
UART2_CTS_B	EIM_D28 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28 IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
	SD3_CMD (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_CMD IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
	SD4_DAT6 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6 IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
UART2_RTS_B	EIM_D29 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29 IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
	SD3_CLK (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_CLK IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
	SD4_DAT5 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5 IOMUXC_UART2_UART_RTS_B_SELECT_INPUT
UART2_RX_DATA	EIM_D27 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27 IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
	GPIO_8 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
	SD3_DAT4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
	SD4_DAT4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
UART2_TX_DATA	EIM_D26 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
	GPIO_7 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
	SD3_DAT5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
	SD4_DAT7 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7
		IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT
UART3 - Universal Asynchronous Receiver/Transmitter		
UART3_CTS_B	EIM_D23 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
	EIM_D30 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
	SD3_DAT3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
UART3_RTS_B	EIM_D31 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
	EIM_EB3 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
	SD3_RST (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD3_RESET
		IOMUXC_UART3_UART_RTS_B_SELECT_INPUT
UART3_RX_DATA	EIM_D25 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25
		IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT
	SD4_CLK (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_CLK
		IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT
UART3_TX_DATA	EIM_D24 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24
		IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT
	SD4_CMD (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD4_CMD
		IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT
UART4 - Universal Asynchronous Receiver/Transmitter		
UART4_CTS_B	CSI0_DAT17 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17
		IOMUXC_UART4_UART_RTS_B_SELECT_INPUT

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
UART4_RTS_B	CSI0_DAT16 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16
		IOMUXC_UART4_UART_RTS_B_SELECT_INPUT
UART4_RX_DATA	CSI0_DAT13 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13
		IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT
	KEY_ROW0 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0
		IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT
UART4_TX_DATA	CSI0_DAT12 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12
		IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT
	KEY_COL0 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL0
		IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT
UART5 - Universal Asynchronous Receiver/Transmitter		
UART5_CTS_B	CSI0_DAT19 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19
		IOMUXC_UART5_UART_RTS_B_SELECT_INPUT
	KEY_ROW4 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
		IOMUXC_UART5_UART_RTS_B_SELECT_INPUT
UART5_RTS_B	CSI0_DAT18 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18
		IOMUXC_UART5_UART_RTS_B_SELECT_INPUT
	KEY_COL4 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4
		IOMUXC_UART5_UART_RTS_B_SELECT_INPUT
UART5_RX_DATA	CSI0_DAT15 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15
		IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT
	KEY_ROW1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
		IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT
UART5_TX_DATA	CSI0_DAT14 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14
		IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT
	KEY_COL1 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1
		IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT
USB - Universal Serial Bus Controller		
USB_H1_DN	USB_H1_DN	Not multiplexed.
USB_H1_DP	USB_H1_DP	Not multiplexed.
USB_H1_OC	EIM_D30 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30
		IOMUXC_USB_H1_OC_SELECT_INPUT
	GPIO_3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO03
		IOMUXC_USB_H1_OC_SELECT_INPUT
USB_H1_PWR	EIM_D31 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31
	GPIO_0 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO00
USB_H1_PWR_CTL_WAKE	KEY_COL2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL2
USB_H2_DATA	RGMII_TXC (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
USB_H2_STROBE	RGMII_TX_CTL (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL
USB_H3_DATA	RGMII_RX_CTL (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL
USB_H3_STROBE	RGMII_RXC (ALT0)	IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC
USB_OTG_CHD_B	USB_OTG_CHD_B	Not multiplexed.
USB_OTG_DN	USB_OTG_DN	Not multiplexed.
USB_OTG_DP	USB_OTG_DP	Not multiplexed.
USB_OTG_HOST_MODE	GPIO_7 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO07
USB_OTG_ID	ENET_RX_ER (ALT0)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER
	GPIO_1 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO01
USB_OTG_OC	EIM_D21 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21 IOMUXC_USB_OTG_OC_SELECT_INPUT
	KEY_COL4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL4 IOMUXC_USB_OTG_OC_SELECT_INPUT
USB_OTG_PWR	EIM_D22 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22
	KEY_ROW4 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4
USB_OTG_PWR_CTL_WAKE	GPIO_8 (ALT7)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
USDHC1 - Ultra Secured Digital Host Controller		
SD1_CD_B	GPIO_1 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO01
SD1_CLK	SD1_CLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_CLK
SD1_CMD	SD1_CMD (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_CMD
SD1_DATA0	SD1_DAT0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0
SD1_DATA1	SD1_DAT1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1
SD1_DATA2	SD1_DAT2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
SD1_DATA3	SD1_DAT3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
SD1_DATA4	NANDF_D0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00
SD1_DATA5	NANDF_D1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01
SD1_DATA6	NANDF_D2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02
SD1_DATA7	NANDF_D3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03
SD1_LCTL	GPIO_16 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO16
SD1_VSELECT	KEY_COL1 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_COL1
	KEY_ROW3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
SD1_WP	DI0_PIN4 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04 IOMUXC_USDHC1_WP_ON_SELECT_INPUT
	GPIO_9 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO09 IOMUXC_USDHC1_WP_ON_SELECT_INPUT
USDHC2 - Ultra Secured Digital Host Controller		
SD2_CD_B	GPIO_4 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO04
SD2_CLK	SD2_CLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_CLK
SD2_CMD	SD2_CMD (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_CMD

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
SD2_DATA0	SD2_DAT0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0
SD2_DATA1	SD2_DAT1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1
SD2_DATA2	SD2_DAT2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2
SD2_DATA3	SD2_DAT3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3
SD2_DATA4	NANDF_D4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04
SD2_DATA5	NANDF_D5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05
SD2_DATA6	NANDF_D6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06
SD2_DATA7	NANDF_D7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07
SD2_LCTL	GPIO_6 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO06
SD2_VSELECT	KEY_ROW1 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1
	KEY_ROW2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2
SD2_WP	GPIO_2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_GPIO02
USDHC3 - Ultra Secured Digital Host Controller		
SD3_CLK	SD3_CLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_CLK
SD3_CMD	SD3_CMD (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_CMD
SD3_DATA0	SD3_DAT0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0
SD3_DATA1	SD3_DAT1 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1
SD3_DATA2	SD3_DAT2 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA2
SD3_DATA3	SD3_DAT3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3
SD3_DATA4	SD3_DAT4 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4
SD3_DATA5	SD3_DAT5 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5
SD3_DATA6	SD3_DAT6 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6
SD3_DATA7	SD3_DAT7 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7
SD3_RESET	SD3_RST (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD3_RESET
SD3_VSELECT	GPIO_18 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_GPIO18
	NANDF_CS1 (ALT2)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B
USDHC4 - Ultra Secured Digital Host Controller		
SD4_CLK	SD4_CLK (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD4_CLK
SD4_CMD	SD4_CMD (ALT0)	IOMUXC_SW_MUX_CTL_PAD_SD4_CMD
SD4_DATA0	SD4_DAT0 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0
SD4_DATA1	SD4_DAT1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1
SD4_DATA2	SD4_DAT2 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2
SD4_DATA3	SD4_DAT3 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA3
SD4_DATA4	SD4_DAT4 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4
SD4_DATA5	SD4_DAT5 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5
SD4_DATA6	SD4_DAT6 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6
SD4_DATA7	SD4_DAT7 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7
SD4_RESET	NANDF_ALE (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_ALE
SD4_VSELECT	NANDF_CS1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B

Table continues on the next page...

Table 4-2. Muxing Options (continued)

Signal	Pad (Mode)	Mux/Input Select Registers
WDOG1 - Watchdog Timer		
WDOG1_B	DISP0_DAT8 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08
	GPIO_9 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO09
	SD1_DAT2 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
WDOG1_RESET_B_DEB	SD1_DAT2 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2
WDOG2 - Watchdog Timer		
WDOG2_B	DISP0_DAT9 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09
	GPIO_1 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO01
	SD1_DAT3 (ALT4)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
WDOG2_RESET_B_DEB	SD1_DAT3 (ALT6)	IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3
XTALOSC - Crystal Oscillator		
XTALOSC_CLK1_N	CLK1_N	Not multiplexed.
XTALOSC_CLK1_P	CLK1_P	Not multiplexed.
XTALOSC_CLK2_N	CLK2_N	Not multiplexed.
XTALOSC_CLK2_P	CLK2_P	Not multiplexed.
XTALOSC_OSC32K_32K_OUT	ENET_RXD0 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0
	KEY_ROW3 (ALT0)	IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3
	SD1_CLK (ALT2)	IOMUXC_SW_MUX_CTL_PAD_SD1_CLK
XTALOSC_REF_CLK_24M	GPIO_3 (ALT3)	IOMUXC_SW_MUX_CTL_PAD_GPIO03
	RGMII_TXC (ALT7)	IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC
XTALOSC_REF_CLK_32K	GPIO_8 (ALT1)	IOMUXC_SW_MUX_CTL_PAD_GPIO08
XTALOSC_RTC_XTALI	RTC_XTALI	Not multiplexed.
XTALOSC_RTC_XTALO	RTC_XTALO	Not multiplexed.
XTALOSC_XTALI	XTALI	Not multiplexed.
XTALOSC_XTALO	XTALO	Not multiplexed.

Chapter 5
Fusemap

5.1 Fusemap

This section details the various modes and selection of the required boot devices. A separate map is given for each and every boot device.

The device select is specified by BOOT_CFG1[7:3] fuses listed in Table 5-1.

Table 5-1. Boot Device Select

Boot Device	BOOT _CFG1[7]	BOOT_CFG1[6]	BOOT_CFG1[5]	BOOT_CFG1[4]	BOOT_CFG1[3]
EIM (Table 5-4)	0	0	0	0	Memory Type: 0 - NOR Flash 1 - OneNAND
SATA (Table 5-5)	0	0	1	0	X
Serial ROM (Table 5-6)	0	0	1	1	X
SD/eSD (Table 5-7)	0	1	0	X	X
MMC/eMMC (Table 5-8)	0	1	1	X	X
NAND Flash (Table 5-9)	1	X	X	X	X

NOTE

Fuses marked as “Reserved” are reserved for Freescale internal (and future) use only. Customers should not attempt to burn these, as the IC behavior may be unpredictable. The reserved fuses can be read as either ‘0’ or ‘1’.

Table 5-2. Lock Fuses

Addr	7	6	5	4	3	2	1	0
0x400[7:0]	Reserved	SJC_RESP _LOCK	MEM_TRIM_LOCK		BOOT_CFG_LOCK		TESTER_LOCK	

Table continues on the next page...

Table 5-2. Lock Fuses (continued)

Addr	7	6	5	4	3	2	1	0
		WRP,OP,RDP	1x - OP x1 - WP		(Locking rows: 0x450-0x470) 1x - OP x1 - WP		1x - OP x1 - WP	
00x400[15:8]	Reserved	SRK_LOCK RD,WR,OP	GP2_LOCK 1x - OP x1 - WP		GP1_LOCK 1x - OP x1 - WP		MAC_ADDR_LOCK at 9:8 1x - OP x1 - WP	
0x400[23:16]	Reserved	MISC_CONF_LOCK 1 - WP + OP of MISC_CONF	Reserved		ANALOG_LOCK 1x - OP x1 - WP		Reserved	
0x400[31:24]	Reserved							

Table 5-3. General Fuses

Addr	7	6	5	4	3	2	1	0
0x430 [23:16]	Reserved				SI_REV[3:0] 0000 - Revisions 1.0, 1.1, 1.2 0010 - Revision 1.3			

Table 5-4. EIM Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	0	0	0	Memory Type: 0 - NOR Flash 1 - OneNAND	Reserved	Reserved	Reserved
0x450[15:8] (BOOT_CFG2)	Muxing Scheme: 00 - A/D16 (HW Default in external boot) 01 - A+DH 10 - A+DL 11 - Reserved		OneNand Page Size: 00 - 1KB 01 - 2KB 10 - 4KB 11 - Reserved		Reserved	Reserved	Reserved	Reserved
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map		Reserved	Boot Frequencies (ARM/DDR)	Reserved	Reserved

Table continues on the next page...

Table 5-4. EIM Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
			10 - 4KB Interleaving Enabled 11 - Illegal			0 - 792 / 528 MHz 1 - 396 / 352MHz		
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved for 'Serial-ROM' Boot mode	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3		eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5 101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved			BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable 01 - 1 GPIO (def freq) 10 - Div by2 11 - Div by 4		BT_LPB_POLARITY (GPIO polarity) 0 - Active High 1 -Active Low	Reserved			
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)		PAD_SETTINGS[5:0]					

Table 5-5. SATA Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	0	1	0	Reserved	Reserved	Reserved	Reserved

Table continues on the next page...

Table 5-5. SATA Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
0x450[15:8] (BOOT_CFG2)	Reserved	Reserved	Reserved	Tx Spread Spectrum 0 - Disabled 1 - Enabled	Rx Spread Spectrum 0 - Enabled 1 - Disabled	SATA_SPEED 0 - Gen2 (3.0Gbps) 1 - Gen1 (1.5Gbps)	SATA Type: 00 - i 01 - m 10 - x 11 - Reserved	
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map 10 - 4KB Interleaving Enabled 11 - Illegal		Reserved	Boot Frequencies (ARM/DDR) 0 - 792 / 528 MHz 1 - 396 / 352 MHz	Reserved	Reserved
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved for 'Serial-ROM' Boot mode	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3		eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5 101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved			BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable 01 - 1 GPIO (def freq) 10 - Div by2 11 - Div by 4		BT_LPB_POLARITY (GPIO polarity) 0 - Active High 1 -Active Low	Reserved			
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						

Table continues on the next page...

Table 5-5. SATA Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)		PAD_SETTINGS[5:0]					

Table 5-6. Serial-ROM Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	0	1	1	Reserved	Reserved	Reserved	Reserved
0x450[15:8] (BOOT_CFG2)	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map 10 - 4KB Interleaving Enabled 11 - Illegal		Reserved	Boot Frequencies (ARM/DDR) 0 - 792 / 528 MHz 1 - 396 / 352MHz	Reserved	Reserved
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved for 'Serial-ROM' Boot mode	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3		eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5 101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved			TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable		BT_LPB_POLARITY (GPIO polarity)	Reserved			

Table continues on the next page...

Table 5-6. Serial-ROM Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
		01 - 1 GPIO (def freq) 10 - Div by2 11 - Div by 4		0 - Active High 1 -Active Low				
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)	PAD_SETTINGS[5:0]						

Table 5-7. SD/eSD Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	1	0	Fast Boot: 0 - Regular 1 - Fast Boot	SD/SDXC Speed 00 - Normal/SDR12 01 - High/SDR25 10 - SDR50 11 - SDR104		SD Power Cycle Enable 0 - No power cycle 1 - Enabled via USDHC_RST pad (uSDHC3 & 4 only)	SD Loopback Clock Source Sel (for SDR50 and SDR104 only) 0 - through SD pad 1 - direct
0x450[15:8] (BOOT_CFG2)	SD Calibration Step 00 - 1 delay cell 01 - 1 delay cell 10 - 2 delay cell 11 - 3 delay cell		Bus Width: 0 - 1-bit 1 - 4-bit	Port Select: 00 - USDHC1 01 - USDHC2 10 - USDHC3 11 - USDHC4		DLL Override: 0 - Boot ROM default 1 - Apply value per fuse field MMC_DLL_DLY[3:0]	Boot Acknowledge Disable / Pull-Down 0 - Use default SD pad settings during power cycle 1 - Set pull-down on SD pads during power cycle (used only when "SD Power Cycle Enable" enabled)	Override Pad Settings (using PAD_SETTINGS value)
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map 10 - 4KB Interleaving Enabled		Reserved	Boot Frequencies (ARM/DDR) 0 - 792 / 528 MHz 1 - 396 / 352MHz	Reserved	Reserved

Table continues on the next page...

Table 5-7. SD/eSD Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
			11 - Illegal					
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved for 'Serial-ROM' boot mode	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3		eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5 101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved			BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable 01 - 1 GPIO (def freq) 10 - Div by2 11 - Div by 4		BT_LPB_POLARITY (GPIO polarity) 0 - Active High 1 -Active Low	Reserved			
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)		PAD_SETTINGS[5:0]					

Table 5-8. MMC/eMMC Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	1	1	Fast Boot: 0 - Regular 1 - Fast Boot	SD/MMC Speed 0 - High 1- Normal	Reserved	Reserved	Reserved

Table continues on the next page...

Table 5-8. MMC/eMMC Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
0x450[15:8] (BOOT_CFG2)	Bus Width: 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - Reserved			Port Select: 00 - uSDHC1 01 - uSDHC2 10 - uSDHC3 11 - uSDHC4		DLL Override: 0 - Boot ROM default 1 - Apply value per fuse field MMC_DLL_DLY[3:0]	Fast Boot Acknowledge Disable: 0 - Boot Ack Enabled 1 - Boot Ack Disabled	Override Pad Settings (using PAD_SETTINGS value)
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map 10 - 4KB Interleaving Enabled 11 - Illegal		Reserved	Boot Frequencies (ARM/DDR) 0 - 792 / 528 MHz 1 - 396 / 352MHz	Reserved	Reserved
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved for 'Serial-ROM' boot mode	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3		eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5 101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved			BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable 01 - 1 GPIO (def freq) 10 - Div by2		BT_LPB_POLARITY (GPIO polarity)	Reserved			

Table continues on the next page...

Table 5-8. MMC/eMMC Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
		11 - Div by 4		0 - Active High 1 -Active Low				
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)	PAD_SETTINGS[5:0]						

Table 5-9. NAND Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	1	Reserved	BT_TOGGLEMODE 0 - Raw Nand 1 - Toggle-Mode NAND	Override Pad Settings (using PAD_SETTINGS value)	Nand Number Of Devices: 00 - 1 01 - 2 10 - 4 11 - Reserved		Nand_Row_address_bytes: 00 - 3 01 - 4 10 - 5 11 - 6	
0x450[15:8] (BOOT_CFG2)	Toggle Mode 33MHz Preamble Delay, Read Latency: 000 - 16 GPMICLK cycles 001 - 1 GPMICLK cycles 010 - 2 GPMICLK cycles 011 - 3 GPMICLK cycles 100 - 4 GPMICLK cycles 101 - 5 GPMICLK cycles 110 - 6 GPMICLK cycles 111 - 7 GPMICLK cycles			BOOT_SEARCH_COUNT: 00 - 2 01 - 2 10 - 4 11 - 8		Pages In Block: 00 - 128 01 - 64 10 - 32 11 - 256		Reset time 0 - 12ms 1 - 22ms (LBA Nand)
0x450[23:16] (BOOT_CFG3)	L1 I-Cache DISABLE	BT_MMU_DISABLE	DDR Memory Map default config 00 - Single DDR channel 01 - Fixed 2x32 map 10 - 4KB Interleaving Enabled 11 - Illegal	Reserved	Boot Frequencies (ARM/DDR) 0 - 792 / 528 MHz 1 - 396 / 352MHz	Reserved	Reserved	
0x450[31:24] (BOOT_CFG4)	Infinitie Loop (for debug) 0 - Disabled 1 - Enabled	EEPROM Recovery Enable 0 - Disabled 1 - Enabled Fuse is Reserved	eCSPI chip select: 00 - ECSPIx_SS0 (default) 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3	eCSPI Addressing: 0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)	Port Select: 000 - eCSPI1 001 - eCSPI2 010 - eCSPI3 011 - eCSPI4 100 - eCSPI5			

Table continues on the next page...

Table 5-9. NAND Boot Fusemap (continued)

Addr	7	6	5	4	3	2	1	0
		for 'Serial-ROM' boot mode				101 - I2C1 110 - I2C2 111 - I2C3		
0x460[7:0]	Reserved	eMMC_RESET_EN	SDMMC_HYS_EN	BT_FUSE_SEL	DIR_BT_DIS	Reserved	SEC_CONFIG[1]	Reserved
0x460[15:8]	Reserved							
0x460[23:16]	JTAG_SMODE[1:0]		WDOG_ENABLE 0 - Disabled 1 - Enabled	SJC_DISABLE	Reserved			
0x460[31:24]	Reserved			TZASC_ENABLE	JTAG_HEO	KTE	Reserved	
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]							
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]							
0x470[23:16]	Reserved	LPB_BOOT (Core / DDR-Bus) 00 - LPB Disable 01 - 1 GPIO (def freq) 10 - Div by2 11 - Div by 4		BT_LPB_POLARITY (GPIO polarity) 0 - Active High 1 -Active Low	Reserved			
0x470[31:24]	Reserved	MMC_DLL_DLY[6:0]						
0x6D0[7:0]	Reserved (locked by MISC_CONF_LOCK)		PAD_SETTINGS[5:0]					

5.2 Fusemap Description Table

This section covers the fusemap descriptions of the chip.

Table 5-10. Fusemap Descriptions

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
0x400[1:0]	TESTER_LOCK	2	Provides Locking of various fuse bits.	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded 1x - Override Protect (OP) x1 - Write Protect (WP)	N/A

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
				11 - Both OP and WP	
0x400[3:2]	BOOT_CFG_LOCK	2	Perform lock on BOOT related fuses.	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[5:4]	MEM_TRIM_LOCK	2	Trimming fuses. Burned on the tester or by customer before the final product shipment.	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded. 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[6]	SJC_RESP_LOCK	1	Lock bit for JTAG response fuses. (SJC_RESP)	00 - Unlock 1 - Read Protect + Write Protect + Override Protect and Program protect lock	N/A
0x400[9:8]	MAC_ADDR_LOCK	2	Lock MAC_ADDR fuses.	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded. 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[11:10]	GP1_LOCK	2	Lock for General Purpose fuse register #1 (GP1)	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded. 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[13:12]	GP2_LOCK	2	Lock for General Purpose fuse register #2 (GP2)	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded. 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[14]	SRK_LOCK	1	Locking SRK_HASH[255:0]	0 - Unlock	N/A

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
				1 - Write Protect + Override Protect	
0x400[19:18]	ANALOG_LOCK	2	Lock bit for various fuses (Addresses: 0x4D0-0x4F0)	00 - Unlock (The controlled field can be read, sensed, burned or overridden in the corresponded. 1x - Override Protect (OP) x1 - Write Protect (WP) 11 - Both OP and WP	N/A
0x400[22]	MISC_CONF_LOCK	1	Locking of MISC_CONFIG fuses (address 0x6D0)	0 - Unlock 1 - WP + OP of MISC_CONF	N/A
0x410[31:0]	SJC_CHALL[31:0] / UNIQUE_ID[31:0]	32	Device Unique ID, also be used for SJC Challenge phrase. (bits 31-0)	Random, by test program.	TESTER_LOCK
0x420[31:0]	SJC_CHALL/ UNIQUE_ID[42:32]	32	Device Unique ID, also be used for SJC Challenge phrase. (bits 63-32)	Random, by test program.	TESTER_LOCK
0x430[19:16]	SI_REV[3:0]	4	Silicon revision number	0 - Revision 1.0, 1.1, 1.2 1 - Revision 1.2.2 2 - Revision 1.2.3	TESTER_LOCK
0x430[21:20]	NUM_CORES	2	Indicates the type of device : 2x cores or 4x cores.	00 - 4x Cores. 01 - Reserved 10 - 2x Cores 11 - Reserved	TESTER_LOCK
0x430[24]	SATA_RST_SRC	1	Controls mux to select 'herset' to SATA, from either SATA reset controller, or SRC module.	SATA_RST_SRC 0 - Origin SATA reset controller 1 - SRC reset	TESTER_LOCK
0x440[17:16]	SPEED_GRADING[1:0]	2	IC core speed	Reflects part speed grading: 00 = 800 MHz 01 = 852 MHz 10 = 1.0 GHz 11 = Reserved	TESTER_LOCK
0x450[7:0]	BOOT_CFG1	8	BOOT configuration register #1, Usage varies, depending on selected boot device.	See Table 5-1 for details.	BOOT_CFG_LOCK
0x450[15:8]	BOOT_CFG2	8	BOOT configuration register #2, Usage varies, depending on selected boot device.	Refer to the respective boot fusemap tables for details.	BOOT_CFG_LOCK
0x450[23:16]	BOOT_CFG3	8	BOOT configuration register #3	Refer to the respective boot fusemap tables for details.	BOOT_CFG_LOCK

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
0x450[31:24]	BOOT_CFG4	8	BOOT configuration register #4	Refer to the respective boot fusemap tables for details.	BOOT_CFG_LOCK
0x460[1]	SEC_CONFIG[1]	1	Select polarity of GPIO pin used for LPB indication.	Out of Factory state: 0 - Open - allows any code to be flashed and executed, even if it has no valid signature. 1 - Closed (Security On)	BOOT_CFG_LOCK
0x460[3]	DIR_BT_DIS	1	Direct External Memory Boot Disable	0 - Direct boot from external memory is allowed 1 - Direct boot from external memory is not allowed	BOOT_CFG_LOCK
0x460[4]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If BOOT_MODE[1:0] = 0b10 0 - Bits of SBMR are overridden by GPIO pins. 1 - Specific bits of SBMR are controlled by eFUSE settings. If BOOT_MODE[1:0] = 0b00 0 - BOOT configuration eFuses are not yet programmed. Boot flow jumps to serial downloader. 1 - BOOT configuration eFuses have been programmed. Regular boot flow is performed.	BOOT_CFG_LOCK
0x460[15:8]	DDR3_CONFIG[7:0]	8	TBD (DDR3 config options)		BOOT_CFG_LOCK
0x460[20]	SJC_DISABLE	1	Disable/Enable the Secure JTAG Controller module. This fuse is used to create highest JTAG security level, where JTAG is totally blocked.	0 - Secure JTAG Controller is enabled 1 - Secure JTAG Controller is disabled	BOOT_CFG_LOCK
0x460[21]	WDOG_ENABLE	1	Watchdog Enable	Used to specify whether to enable / not watchdog at boot. 0 - Watch-Dog is disabled 1 - Watch-Dog is enabled	BOOT_CFG_LOCK
0x460[23:22]	JTAG_SMODE[1:0]	2	JTAG Security Mode. Controls the security mode of the JTAG debug interface	00 - JTAG enable mode 01 - Secure JTAG mode 11 - No debug mode	BOOT_CFG_LOCK

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
0x460[26]	KTE	1	Kill Trace Enable. Enables tracing capability on ETM, and other modules.	0 - Bus tracing is allowed 1 - Bus tracing is allowed in case security state as defined by Secure JTAG allows it (for example, JTAG_ENABLE or NO_DEBUG)	BOOT_CFG_LOCK
0x460[27]	JTAG_HEO	1	JTAG HAB Enable Override. Disallows HAB JTAG enabling. The HAB may normally enable JTAG debugging by means of the HAB_JDE-bit in the OCOTP SCS register. The JTAG_HEO-bit can override this behavior.	0 - HAB may enable JTAG debug access 1 - HAB JTAG enable is overridden (HAB may not enable JTAG debug access)	BOOT_CFG_LOCK
0x460[28]	TZASC_ENABLE	1	TZASC1,2 enable fuse.	0 - TZASC1, 2 modules left in disable and bypass state. 1 - TZASC1, 2 modules, associated clocks and muxing are enabled by Boot ROM code.	BOOT_CFG_LOCK
0x460[29]	SDMMC_HYS_EN	1	Override HYS bit for SD/MMC pads	0 - Override HYS bit for SD/MMC pads disabled. 1 - Once the fuse blown, the [HYS] bit of IOMUXC_SW_PAD_CTL_PAD_SDx_CLK, IOMUXC_SW_PAD_CTL_PAD_SDx_CMD, IOMUXC_SW_PAD_CTL_PAD_SDx_DAT0-n will be set.	BOOT_CFG_LOCK
0x460[30]	eMMC_RESET_EN	1	eMMC 4.4 - RESET TO PRE-IDLE STATE	0 - The CMD0 with argument 0xf0f0f0f0 will not be sent to put the eMMC card into pre-IDLE state. 1 - The CMD0 with argument 0xf0f0f0f0 will be sent to put the eMMCcard into pre-IDLE state so that eMMC card's fast boot can work properly.	BOOT_CFG_LOCK
0x470[7:0]	NAND_READ_CMD_CODE1[7:0]	8	NAND_READ_CMD_CODE1	First command word to be used for Nand read.	BOOT_CFG_LOCK
0x470[15:8]	NAND_READ_CMD_CODE2[7:0]	8	NAND_READ_CMD_CODE2	Second command word to be used for Nand read.	BOOT_CFG_LOCK
0x470[20]	BT_LPB_POLARITY	1	Define GPIO3 polarity, for determining LPB boot mode.	0 - Active High 1 -Active Low	BOOT_CFG_LOCK

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
0x470[22:21]	LPB_BOOT	2	Defined the LowPower Boot options	(Core / DDR- Bus) 00 - LPB Disable 01 - 1 GPIO (default freq) 01 - Div by2 11 - Div by 4	BOOT_CFG_LOCK
0x470[30:24]	MMC_DLL_DLY[6:0]	7	eMMC 4.4/4.41 delay line default value (set by boot rom), used in conjunction with "DLL Override" = 1 (BOOT_CFG3[3])	Delay value required by USDHC for eMMC 4.4/4.41 to work	BOOT_CFG_LOCK
0x480[7:6]	Temperature Grade	2	Indicates the device temperature grade	00 - Commercial (0 to 95C) 01 – Extended Commercial (-20 to 105C) 10 - Industrial (-40 to 105C) 11 - Automotive (-40 to 125C)	MEM_TRIM_LOCK
0x4D0[31]	Power-Gate cores 2,3	1		0 - All quad-core devices: 4 cores powered on. 1 - All dual-core devices: cores 2 and 3 power-gated.	ANALOG_LOCK
0x4F0[15:0]	USB_VID[31:0]	16	USB VID value,(to be used by USB software driver).		ANALOG_LOCK
0x4F0[31:16]	USB_PID[31:0]	16	USB PID value (to be used by USB software driver).		ANALOG_LOCK
0x580-0x5F0	SRK_HASH[255:0]	256	SRK key	0x580-SRK0 0x590-SRK1 0x5F0-SRK2 0x5F0-SRK3 0x5F0-SRK4 0x5F0-SRK5 0x5F0-SRK6 0x5F0-SRK7	SRK_LOCK
0x600[31:0]	SJC_RESP[31:0]	32	Response reference value for the secure JTAG controller	Customer / OEM use.	SJC_RESP_LOCK (locks also for read and explicit sense)
0x610[23:0]	SJC_RESP[55:32]	24	Response reference value for the secure JTAG controller	Customer / OEM use.	SJC_RESP_LOCK (locks also for read and explicit sense)
0x620[31:0]	MAC_ADDR[31:0]	32	Reserved for customers/ software	Customer / OEM use.	MAC_ADDR_LOCK

Table continues on the next page...

Table 5-10. Fusemap Descriptions (continued)

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Settings	Locked by
0x630[15:0]	MAC_ADDR[47:32]	16	Reserved for customers/ software	Customer / OEM use.	MAC_ADDR_LOCK
0x660[31:0]	GP1[31:0]	32	General Purpose fuse register #1		GP1_LOCK
0x670[31:0]	GP2[31:0]	32	General Purpose fuse register #2		GP2_LOCK
0x6D0[5:0]	PAD_SETTINGS	6	Used with conjunction of MMC/SD/Nand "Override Pad Settings" fuse value, as follows: 0 - Use IO default settings for boot device IO pads 1 - Use "Override" value, as set by this register	IO pads settings of selected boot interface, are override with this fuses, as follow: [0] - Slew Rate [3:1] Drive Strength [5:4] - Speed Settings. Refer to IO PAD chapter for "Settings" fields value	MISC_CONF_LOCK
0x6E0[0]	FIELD_RETURN	1	Configure device for field return testing. Fuse burning requires CSF command.	0 - Device is in functional / secure mode. 1 - Device is open for 'field-return' testing	FIELD_RETURN_LOCK

Chapter 6
External Memory Controllers

6.1 Overview

This chip has the following external memory interfaces and controllers:

- Multi-mode DDR controller (MMDC)
- Raw NAND Flash controller, consisting of GPMI2, BCH40, and APBH_DMA components including Randomizer module.
- EIM-PSRAM/NOR Flash controller

6.2 Multi-mode DDR controller (MMDC) overview and feature summary

The MMDC module is a DDR controller that can support several types of DDR memories and two channel x32 and x64 memory widths.

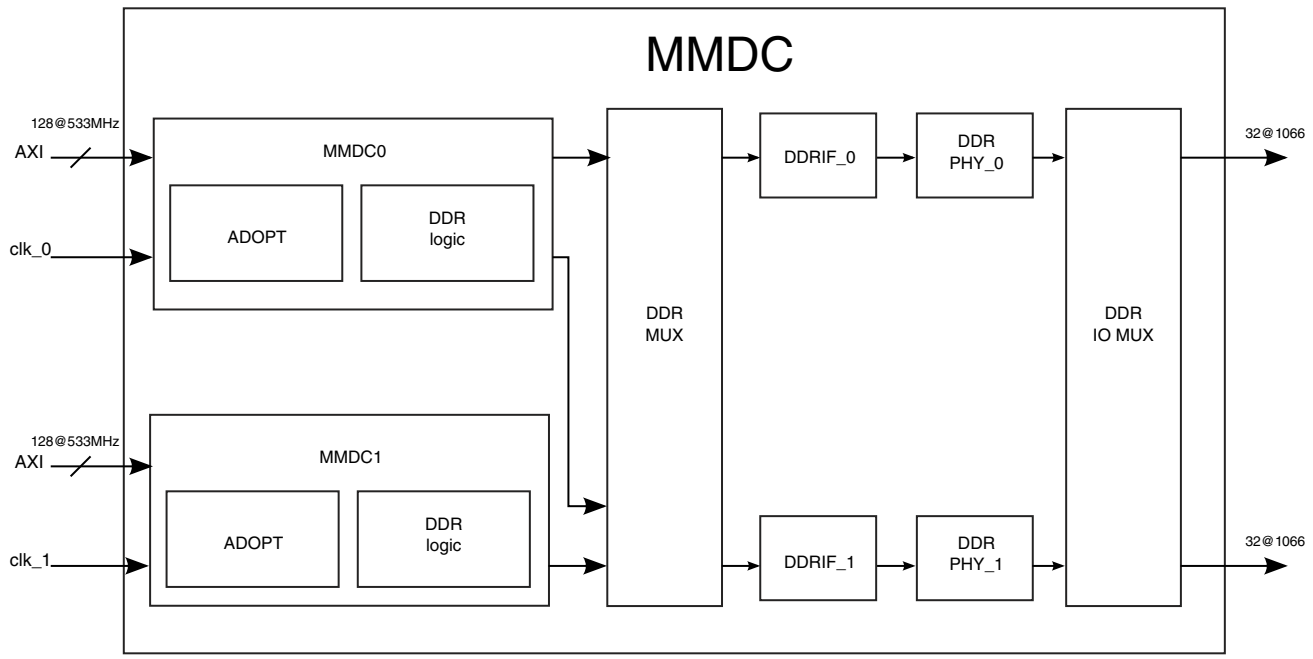


Figure 6-1. MMDC block diagram

Table 6-1. MMDC feature summary

Feature	Description
Supported standards	• LV-DDR3, DDR3 x16, x32, x64 (includes SODIMM) • LPDDR2 2ch x32, in either split map or interleaving mode • LPDDR2 1ch x32

DDR interface	• x16, x32, x64 data bus width
	• Density of 256 Mbytes-8 Gbytes
	  • Column size of 8-12 bits
	  • Row size of 11-16 bits
	• 2 CS per channel, with a separate CS allocation for LPDDR2-DRAM)
	• Up to 4 Gbyte address space and configurable address space per CS. For LPDDR2 2ch x32 up to 2 Gbytes per channel
	• Interleaved accesses of LPDDR2-DRAM towards the same DDR channel. This is supported only when using the same clock frequency for LPDDR2-DRAM
	• Supports burst length of 8 (aligned) for DDR3 and burst lengths of 4 for LPDDR2

DDR performance	• DDR3 and LPDDR2 support up to 1066MT/s transfer rate • Supports Real-Time priority by means of QoS sideband priority signals from the chip to enable different priority levels in the re-ordering mechanism • Page hit/page miss optimizations • Consecutive read/write access optimizations • Supports deep read and write requests queues to enable bank prediction • Drives back the critical word in a read transaction as soon as it is received by the DDR device (doesn't wait until the whole data phase has been completed)

Table continues on the next page...

Table 6-1. MMDC feature summary (continued)

Feature	Description
	• Can track open memory pages • Supports bank interleaving • Special optimization for non-aligned wrap accesses in burst length 8
AXI interface	• AXI bus compliant with glueless interface to PL301 AXI network interconnect • Supports bus transfers of 8,16,32, 64 and 128 bits (single accesses and bursts)

DDR calibration and delay-lines	• All calibrations can be done automatically by hardware or manually by software
	• ZQ calibration for external DDR device (in DDR3 through the ZQ calibration command and in LPDDR2 through the MRW command).
	  • Can be handled automatically for ZQ Short (periodically) and ZQ Long (at exit from self-refresh).
	  • Can be handled manually at ZQ INIT.

DDR general	• Configurable timing parameters • Configurable refresh scheme • Supports dynamic voltage, frequency change and low power mode entry through hardware negotiation with the system (req/ack handshake) • Suppors automatic self-refresh and power down entry and exit • Supports fast and slow precharge power down in DDR3 • Supports various ODT control schemes. • Assertion/Deassertion of ODT control per read or write accesses and for active or passive CS • Supports MRW and MRR commands for LPDDR2. • Software control for moving to derated timing parameters and derated refresh rate according to temperature variation. • Supports various debug and profiling modes

6.3 Raw NAND Flash controller overview

The Raw NAND Flash controller consists of three components: BCH, GPMI, and APBH_DMA

- BCH is a 40-bit error correction hardware engine with an AXI bus master and a private connection to GPMI.
- GPMI is the NAND controller pin interface.
- APBH_DMA is the DMA engine that drives the GPMI module.

The following figure shows the modules' connectivity.

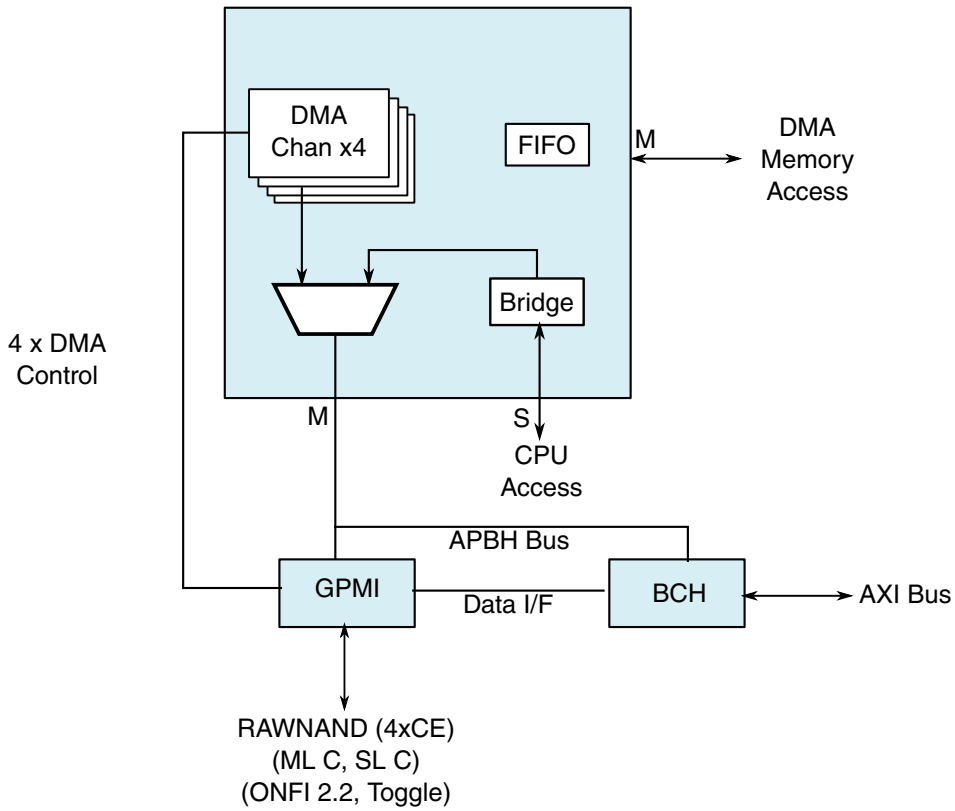


Figure 6-2. Raw NAND Flash controller sub-system

6.3.1 NAND interface features

- ONFI2.2 compliant
  - Timing modes 0-5 for both asynchronous and synchronous interface
  - Synchronous clock rate of up to 100 MHz with data rate of up to 200 MB/s
- Support for ganged ready/busy inputs
  - Support allows the use of a single package pin for all ready busy/busy input signals.
  - In ganged mode (selectable by a programmable mode bit), GPMI expects that all NAND r/b pins are hard wired to the GPMI RDY_BUSY0.
- Up to 4 NAND devices supported by 4 chip-selects and 1 ganged ready/busy.
- Legacy raw SLC, MLC type device
- BA-NAND (Micron)
- PBA-NAND and LBA-NAND (Toshiba)
- E2-NAND (Hynix)
- EF-NAND (Samsung)

- Samsung's "Toggle-mode" NAND (clock rate of up to 66 Mhz and 80 Mhz, with data rate of up to 133 MB/s and 160 MB/s respectively).
- Configurable page size of 2 Kbytes, 4 Kbytes, or 8 Kbytes.
- Configurable spare area per page of up to 512 bytes.
- Support for non-identical NAND devices not required

6.3.2 NAND control features

The NAND control supports interleaved accesses to as many as four NAND devices while hiding the busy period of each devices. However, fully-independent interleaving cannot be supported in ganged ready/busy mode because that suggests fully independent DMA channel operation on multiple paralleled-accessed chip-enables.

It also features:

- Delay line for adjusting the latching edge in case the frequency is relatively high while the delay is big
- Support for monitoring the external ready/busy signal and read status command polling to compare R/B bit.

6.3.3 Internal interface features

- AXI Master interface can access the system's memory space.
- APBH slave port for register configuration through the APBH bridge DMA.

6.3.4 APBH-DMA overview

APBH-DMA provides an APB peripheral bus system for GPMI and BCH. APBH-DMA uses a single GPMI interface to support as many as four DMA channels for four NAND devices (4 chip-enables) in interleaved mode.

6.3.5 ECC-BCH features

BCH supports:

- A data rate of up to 200 Mbytes/s
- 2 Kbyte codeword (1 Kbyte data + parity)

    - Galois Field polynomial GF(2^14)
    - Up to 40-bit ECC for 1 Kbyte of data. (2,4,6,8,10,12,14,16, 18,20,22,24,26,28,30,32,34,36,38,40 configurable)
- 1 Kbyte codeword (1/2 Kbyte data + parity)
    - Galois Field polynomial GF(2^13)
    - Up to 40-bit ECC for 1/2 Kbyte of data (2,4,6,8,10,12,14,16, 18,20,22,24,26,28,30,32,34,36,38,40 configurable)

6.4 EIM-PSRAM/NOR Flash controller overview

EIM, which is an external interface module, handles the interface to devices that are external to the chip, including the generation of chip selects, clocks, and control for external peripherals and memory.

It provides asynchronous and synchronous access to devices with an SRAM-like interface.

6.4.1 EIM features

- Up to four (software configurable) chip selects for external devices
    - Flexible address decoding; each chip select memory space is determined separately according to the GPR bits in IOMUXC.
    - 128 MByte maximum supported density by default (AUS bit is cleared). When the AUS bit is set, maximum supported density is 32MBytes.
- Selectable write protection for each chip select
- Support for multiplexed address/data bus operation x16 and x32 port size
- Programmable data port size for each chip select (x8, x16 and x32)
- Programmable wait-state generator for each chip select, for write and read accesses separately
- Asynchronous accesses with programmable setup and hold times for control signals
- Supports asynchronous page mode accesses (x16 and x32 port size)
- Independent synchronous memory burst read mode support for NOR-Flash and PSRAM memories (x16 and x32 port size)
- Independent synchronous memory burst write mode support for PSRAM and NOR-Flash like memories (CellularRAM™ from Micron, Infineon, and Cypress, OneNAND™ and utRAM™ from Samsung, and COSMORAM™from Toshiba)
- Supports NAND-Flash devices with NOR-Flash like interface - OneNAND™ (Samsung)
- Independent programmable variable/fix latency support for read and write synchronous (burst) mode

- Support for little endian operation
- ARM AXI slave interface accesses only handled in parallel for single AXI ID transactions
- External interrupt support using the RDY_INT signal function as an external interrupt pin
- Boot from external device support according to boot signals, using the RDY_INT signal
  - RDY signal support assertion after reset
  - INT signal support assertion after reset for OneNAND™ (Samsung) device
  - Supports little endian mode only

6.4.2 EIM boot scenarios

EIM allows booting from NOR Flash devices. To select NOR Flash as the boot source, use either the boot mode and configuration GPIO pins or the internal boot-related fuses.

See Sytem Boot for more information.

6.4.3 EIM boot configuration

The following table shows the EIM boot configuration.

Table 6-2. EIM boot configuration

EIM_BOOT_CFG bus	EIM affected bits	EIM register
12	NUM16_BYP_GRANT	CS0GCR2
11	DZS[2]	CS0GCR1
10	AUS	CS0GCR1
[9:8]	CSREC[2:1]	CS0GCR1
[7:5]	RWSC[4:2]	CS0GCR1
	WWSC[4:2]	CS0WCR
4	ERRST	WCR
3	RAL	CS0RCR1
	WAL	CS0WCR
2	MUM	CS0GCR1
	OEA[1]	CS0RCR1
[1:0]	DSZ[1:0]	CS0GCR1

6.4.4 OneNAND requirements

Because Ready/Busy pin is not in use, OneNAND devices require the following actions:

- Poll the device to see whether it is ready; software performs a read from the device.
- Connect the Ready/Busy signal of the device to any GPIO pin and use it as an interrupt that indicates the on ready state.

Chapter 7
System Debug

7.1 Overview

This chapter describes the hardware and software debug and application development features and resources of the chip. It discusses the following:

- Core/platform-specific resources
- Resources associated with complex IP blocks
- Chip-wide resources
- Interface to the external debug and development tools

The debug and trace architecture is designed around the following:

- ARM's CoreSight architecture, adapted to i.MX 6Dual/6Quad SoC (for 2x and 4x core debug), including a cross-trigger subsystem for cross-domain triggering of debug resources
- JTAG port used to interact with cores under debug by means of SJC, the system JTAG controller port
- DAP, the debug access port that supports the interface to the ARM RealView Debugging tools and other third party tools
- TPIU, a trace port interface unit that efficiently accesses program trace information from the system
- Various chip-wide resources, such as debug features built into the IP blocks and critical signal visibility available through alternate pin functions or observability muxes

7.2 Chip and Cortex-A9 Core Platform Debug Architecture

ARM Cortex-A9 Debug architecture is based on CoreSight architecture by ARM Ltd. The CoreSight architecture provides a system wide solution to real-time debug and trace.

The CoreSight architecture is embodied in a set of CoreSight components and compliant processors that form CoreSight systems. Its architecture maintains the traditional requirements of debug and trace:

- To access debug functionality without software interaction;
- To connect to a running system without performing a reset.

Full access to the processor debug capability is available by the ARM Cortex-A9 debug register map through the Advanced Peripheral Bus (APB) slave port. The core includes a Processor Debug Unit allow which stops program execution, examines and alters the processor and coprocessor state, examines and alters the memory and input/output peripheral state and restarts the processor core.

7.2.1 Debug Features

- CoreSight Program Trace Macrocell (PTM): trace generator for the ARM Cortex A9™ core
- Support for a TrustZone-related 3-level debug scheme:
  - Debug in Non-Secure privileged and user, and Secure user
  - Debug in Non-Secure only
- EmbeddedICE-RT logic
  - Support for both monitor-mode and halt-mode debugging:
  - Core run/halt control, debug status/control
  - Breakpoint/watchpoint control
  - Core- and memory-mapped resource examination/modification
- Data communication channel between ARM core and host debugger via JTAG, and the Debug Access Port (DAP) module.
- PMU: Performance Metrics Unit used for system profiling and debug.
- CP15 register for debugging the MMU, I & D L1 cache, and TLB
- PL310 L2 Cache: Provides an event monitoring signal routed to pins for visibility to help in system debug.

The chip includes ARM CoreSight components for multicore debug and trace solutions.

See the following figure for the ARM debug architecture scheme.

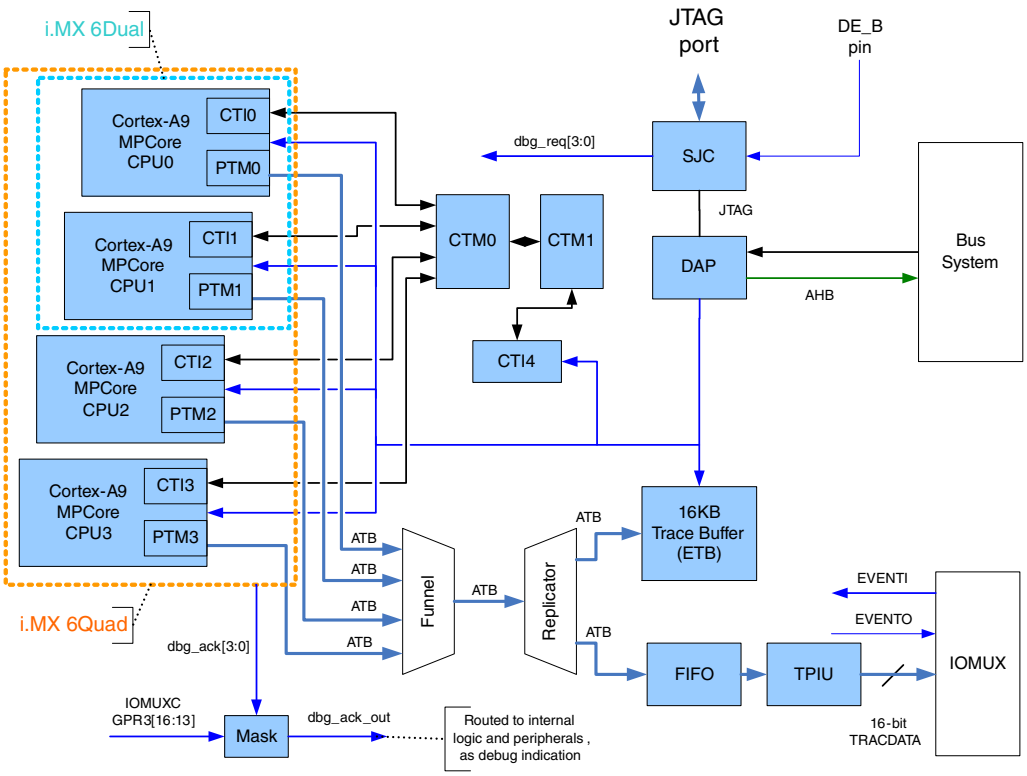


Figure 7-1. i.MX 6Dual/6Quad Cortex-A9 Core Platform Debug Architecture

7.2.2 Debug System components

CoreSight components include:

- Embedded Trace Buffer (ETB):16 Kbyte RAM array to be used for on-chip capture of trace data output from the PTM
- ATB Replicator to connect the trace data to TPIU (Trace Port Interface) and ETB (Embedded Trace Buffer)
- ATB Funnel for capturing trace data from the either two or four cores
- Cross Triggering logic for event routing, including CTIs and CTMs

Other related IPs and functionality:

- ROMPATCH module to support modification of program/data information in the MCU ROM
- Debug Visibility, which selects critical signals routed to the I/O pads as alternate outputs for external visibility

7.2.2.1 AMBA trace bus (ATB)

ATB transfers trace data though the CoreSight infrastructure in a chip. Trace sources are ATB masters and sinks are ATB slaves. The ARM (via PTM) cores are the data generators. Link components such as the Trace Funnel and Replicator provide both master and slave interfaces.

The ATB protocol supports:

- Stalling of trace sources to enable the CoreSight components to funnel and combine sources into a single trace stream.
- Association of the trace data with the generating source using trace source IDs. A CoreSight system can trace up to 111 different items at any one time.
- Capture and transfer of multiple byte bus widths, currently to 32 bits.
- A flushing mechanism to force historic trace to drain from any sources, links, or sinks up to the point that the request was initiated

7.2.2.2 ATB replicator

The ATB replicator enables two trace sinks to be wired together and to operate from the same incoming trace stream.

There are no programmable registers. This component is invisible to the user on a particular trace path, from source to sink.

- Incoming ATB Interface—The ATB replicator accepts trace data from the trace source, either directly or through a trace funnel.
- Outgoing ATB Interfaces—The ATB replicator sends identical trace data on outgoing master port interfaces.

7.2.2.3 Embedded Cross Triggering

The ECT is a modular component from ARM Limited that supports the interaction and synchronization of multiple triggering events within a chip. The main function of the ECT (CTI and CTM) is to pass debug events from one core to another. For example, the ECT can communicate debug state information from one core to another, so that program execution on both processors can be stopped at the same time if required.

The ECT consists of the following types of modules:

- Cross trigger interfaces (CTI)
- Cross trigger matrix (CTM)

Cross trigger interfaces provide the interface between a component or subsystem and the cross trigger matrix. The system requires a CTI for each subsystem that supports cross triggering. The CTI combines and maps the trigger requests, and broadcasts them to all other interfaces on the ECT as channel events. When the CTI receives a channel event it maps this onto a trigger output. This enables subsystems to cross trigger with each other. The receiving and transmitting of triggers is performed through the trigger interface.

The Cross Trigger Matrix (CTM) combines the trigger request generated from the CTIs and broadcasts them to all CTIs as channel triggers. The CTM controls the distribution of channel events. It provides Channel Interfaces (CIs) to connect to either CTIs or CTMs. This enables multiple CTIs to be linked together. The ECT is composed of three/five CTIs (Cross Trigger Interface) and two CTMs (Cross Trigger Matrix). The ECT is key in the multi-core and multi-IP debug strategy. The outcome is a SWcontrolled debug signal matrix that receives signals from various sources (i.e. cores and peripherals) and propagates/ routes them to the different debug resources of the SoC. Those debug resources can include time stamping capability, profiling capabilities, real-time trace (trace enabled or disabled), triggers, SOC level multiplexing, and debug interrupts.

NOTE

As the ECT should only be used during debug sessions, it is off (disabled) by default.

The ECT features are enabled by enabling the individual CTIs. It is only allowed in supervisor mode, after having presented a suitable key to the CTI CTILOCK register. The CTI has eight inputs (trigger inputs) and eight outputs (trigger outputs) to the core/ chip and four inputs and outputs to the CTM (channel triggers).

7.2.2.3.1 Cross-Trigger Matrix (CTM)

The CTM (Cross Trigger Matrix) is provided by ARM. A brief description is provided below. It is advised to refer to ARM documentation for more details.

The CTM is a relatively simple block with no configuration options. There are two CTM instances in i.MX6Dual/6Quad in ARM Platform.

One of them is used to route 4x Core's CTI's, while the second one, is used for additional CoreSight CTI. Each CTI has 4 channel lines, which CTI events are mapped to. The exact mapping is configured in the CTI logic.

7.2.2.3.2 Cross-Trigger Interface (CTI)

The Cross-Trigger Interface (CTI) component is provided by ARM. A brief description of the CTI is provided below.

There are 5 CTIs in the chip's ARM platform. Four are of which, dedicated to each Core, while the 5th is used for various other signals routing.

Each of these CTIs has 8 trigger inputs and 8 trigger outputs that connect to logic in the domain to be debugged or profiled. Each CTI also includes a 4 channel interface to the CTM (4 inputs and 4 outputs).

For more information, see ../../../block_guide_library/arm_a9/map_arm_a9.xml.

7.2.2.4 Debug Access Port (DAP)

The DAP enables debug access to the chip modules through APB-AP (the APB access port) and APB-Mux (the APB multiplexer).

AHB-AP provides system access. Debug tools can use JTAG to connect to the chip.

DAP has the following features:

- AMBA 3 Peripheral Bus Multiplexor access though AMBA 3 APB Access Port, providing debug peripheral access through the APB interface.
- External JTAG access using the JTAG Debug Port (JTAG-DP).
- Internal chip module access using:
  - AHB Access Port (AHB-AP)
  - APB Access Port (APB-AP)
  - JTAG Access Port (JTAG-AP)

APB-Mux enables system access to CoreSight components connected to the Debug APB.

The ROM table provides a list of memory locations of CoreSight components connected to the Debug APB. This is visible from both tools and system access and one configures it during system implementation.

External read/write access to the internal interface is provided by JTAG-DP. JTAG-DP provides a standard interface for debug access to the chip through DAP. It interfaces to the DAP internal bus.

Internal access to on-chip buses and other interfaces are provided by the access ports (APs). The available APs are:

- AHB-AP which provides an AHB-Lite master for access to a system AHB bus.
- APB-AP which provides an AMBA 3 APB master for access to the Debug APB that configures all CoreSight components.
- JTAG-AP which provides JTAG access to on-chip components and operates as a JTAG master port to drive JTAG chains throughout the chip.

7.2.3 i.MX6Dual/6Quad-Specific SJC Features

7.2.3.1 JTAG Disable Mode

In addition to four different JTAG security modes that are implemented internally in the System JTAG Controller (SJC), there is an option to disable the SJC functionality by e-fuse configuration.

This creates additional JTAG mode "JTAG Disabled" with highest level of JTAG protection. In this mode all JTAG features are disabled. Specifically, the following debug features are disabled in addition to the features that were already disabled in "No Debug" JTAG mode:

- Non-Secure JTAG control registers (PLL configuration, Deterministic Reset, PLL bypass)
- Non-Secure JTAG status registers (Core status)
- Chip Identification Code (IDCODE)

7.2.3.2 JTAG ID

Table 7-1. i.MX JTAG ID

Device	Silicon revision	JTAG ID (ID CODE)
i.MX 6Dual	Rev 1.0	0191_E01Dh[1]
i.MX 6Quad	Rev 1.0	0191_C01Dh[2]

1. In follow-on silicon revisions, the ID value is subject to change by incrementing the first nibble as follows: 1191_E01Dh for Rev 1.1, 2191_E01Dh for Rev 1.2 , etc.
2. In follow-on silicon revisions, the ID value is subject to change by incrementing the first nibble as follows: 1191_C01Dh for Rev 1.1, 2191_C01Dh for Rev 1.2 , etc.

7.2.4 System JTAG Controller - SJC

The SJC module is the bridge between external development and test instrumentation and the internal JTAG-accessible debug and test resources.

It implements and manages the daisy-chained topology consisting of its own TAP and those of the SDMA, and the ARM Debug Access Port (DAP).

The DAP itself has a JTAG serial links, which can drive the PCIe PHY and SATA PHY's JTAG logic, used for testing of the PHYs.

NOTE

Single Wire Debug (SWD) protocol is not supported.

7.2.5 System JTAG controller main features

- IEEE P1149.1, 1149.6 (standard JTAG) interface to off-chip test and development equipment
  - Includes an SJC-only mode for true IEEE P1149.1 compliance, used primarily for board-level implementation of boundary scan.
  - Supports IEEE P1149.6 extensions to the JTAG standard for AC testing of selected I/O signals.
- Debug-related control and status; puting selected cores into reset and/or debug mode and monitoring individual core status signals by means of JTAG
- System status, such as the state of the PLLs (locked or not locked)
- Four levels of security, ranging from no security to no JTAG accessibility to the chip

7.2.6 SJC TAP Port

The SJC supports the following standard JTAG pins:

- TRSTB
- TDI
- TDO
- TCK
- TMS

7.2.7 SJC main blocks

- Interface to the outside world via the standard JTAG pins
- Interface to the external Debug_Event pin
- A master TAP controller which implements the standard JTAG state machine
- Implementation of the mandatory and optional IEEE P1149.1 (JTAG) instructions
  - Mandatory: "EXTEST", "SAMPLE/PRELOAD", and "BYPASS"
  - Optional: "ID_CODE" (SOC JTAG ID register), "HIGHZ"
- Implementation IEEE P1149.6 (JTAG) mandatory instructions:
  - "EXTEST_PULSE" and "EXTEST_TRAIN". These two instructions enable edge-detecting behavior on the signal path containing AC pins.
- Supports the SDMA's DR-path-only JTAG architecture by implementing the controller portion of its TAP (including "BYPASS" as the default state) within the SJC
- The ExtraDebug registers, which implement a variety of control and status features
  - Three 32-bit insecure general purpose status registers
  - Two 32-bit secure status registers - one predefined, one general purpose.
  - Control and status registers for debug, core, charge pump, and PLL.
- Four levels of fuse-defined security, ranging from no security to no access.

Both predefined and user-defined (SOC integration team) control and status functions are supported by the SJC.

The user-defined functions will be defined and documented by the SOC integration team.

7.3 Smart DMA (SDMA) core

SDMA is a dedicated, programmable DMA engine. It is an integration of a 32-bit RISC core and DMA-specific hardware. It includes ports for the AP domain and a peripheral domain, along with a burst-capable port for direct external memory access.

The SDMA and its integration in the chip is unchanged from previous i.MX chips.

The main SDMA debug features are:

- OnCE - On Chip Emulator, provides the following capabilities:
  - SDMA core control - run/halt/single-step
  - SDMA core register/memory-map access
  - Event detection, watchpoints, and hardware breakpoints
  - Real time buffer and PC trace buffer capability
- Trace buffer
  - Contains information to identify the 32 last changes of flow detected during a program execution

- Context dump
  - Includes information about all the channel dump activity
  - Current contents of SDMA RAM
- ROMPATCH

7.3.1 SDMA On Chip Emulation Module (OnCE) Feature Summary

The SDMA debug features are primarily defined by the OnCE portion of its design.

They are summarized as follows:

- Memory And Register Access - dedicated logic enables user-access to SDMA memory and register locations. These accesses are supported only when the processor is in debug mode.
- Event Detection Unit - watches signals from the data memory bus (DMBus) which is used by the RISC core to access its RAM, ROM, and memory-mapped registers
- Watchpoints - one output signal is available to watch event matching conditions at the chip level. Match conditions are defined by programming memory-mapped registers.
- Hardware Breakpoint - a counter is decremented after an event detection. A debug request is sent to the SDMA core only when the counter reaches the value of zero. It is possible to program the initial value of the counter or to disable the use of the counter if a debug request must be generated after each event detection.
- Real Time Buffer - The Real Time Buffer Register (RTB) is a single 32-bit memory-mapped register which can be accessed as a regular memory location during program execution. It is used to store and retrieve run time information without putting the SDMA in debug mode. Each write to this register causes an event. This register is, in fact, located in the OnCE. Executing through JTAG, a buffer command exports the content of this register through the JTAG port.
- Core Control (Core Status / Single Stepping) - Commands are provided to monitor and control processor activity. The commands can halt the core, rerun the core from another address location, and get processor status.
- Trace Buffer - a 32x32 buffer which records the last 32 changes of flow during program execution. The buffer stores data in a modulo fashion (i.e. the 33rd instruction change replaces the 1st). Captured trace information is retrieved via reads to the Trace Buffer Register.

7.3.1.1 Other SDMA Debug Functionality

- Core Trace - basic core trace capability is available through debug visibility functionality only. PTM trace capability does not exist.
- ROM Patch - can be accomplished by manipulating the CHN0ADDR register through JTAG or via the MCU's ability to write to SDMA OnCE registers. This must be done right after reset and before the SDMA core is enabled to begin processing events.
- Additional debug control/status interaction with the SJC module
  - SJC-controlled Debug Request
  - SJC-readable Debug Acknowledge (in debug mode)
  - Debug clock control - allows SJC to force clocks on for debug purposes
  - Debug core state (SDMA RISC Core State) - 4 bits accessible from the SJC via JTAG

7.3.1.2 SDMA ROM Patching

After reset, the SDMA is in its IDLE_AFTER_RESET mode. A debug request also puts the SDMA in its DEBUG_IN_IDLE_AFTER_RESET mode. The new address boot must be stored in CHN0ADDR register (e.g., through the SDMA OnCE via debugger).

The user must then issue the exec_core <instruction> SDMA OnCE instruction to return to the IDLE_AFTER_RESET mode. The very first instructions of the boot code fetches the contents of this register (which is also mapped in the SDMA memory space) and jumps to the given address.

7.4 Miscellaneous

7.4.1 Clock/Reset/Power

CDBGPWRUPREQ and CDBGPWRUPACK are the handshake signals between the DAP and the clock control module to ensure debug power and clocks are turned on. If the debug components are always powered on, the handshake becomes a mechanism to turn debug clocks on. Similarly, there is a register bit in the CCM which allows internal software to turn debug clocks on as well because the CDBGPWRUPREQ is in the TCLK domain and is inaccessible to software.

The Cortex-A9 and VSP cores can receive resets from the following sources:

- Debug Reset (CDBGRSTREQ bit within the SWJ-DP CTRL/STAT register of the DAP) in the TCLK domain. This allows the debug tools to reset the debug logic.
- System POR reset

Conversely, the debug system is capable of generating a system reset via a request bit in the MDM-AP control register. This allows the debugger to hold the system in reset.

7.5 Supported tools

RealView™ ARM Debugger is supported.

The debugger is connected to the chip from the host by the RealView ICE protocol converter. Other 3rd party tools can be used via the standard JTAG interface, but may need to be adapted for individual IC. It is important to check with tool vendors for specific tool requirements, especially for on-chip IC.

Chapter 8
System Boot

8.1 Overview

The boot process begins at Power On Reset (POR) where the hardware reset logic forces the ARM core to begin execution starting from the on-chip boot ROM.

Boot ROM code uses the state of the internal register BOOT_MODE[1:0] as well as the state of various eFUSEs and/or GPIO settings to determine the boot flow behavior of the device.

The main features of the ROM include:

- Support for booting from various boot devices
- Serial downloader support (USB OTG)
- Device configuration data (DCD) and plugin
- Digital signature and encryption based High Assurance Boot (HAB)
- Wake-up from low power modes
- Wake-up secondary core

The boot ROM supports the following boot devices:

- NOR Flash
- NAND Flash
- OneNAND Flash
- SD/MMC
- Serial ATA (SATA) HDD (only i.MX 6Dual/6Quad)
- Serial (I2C/SPI) NOR Flash and EEPROM

In normal operation, the Boot ROM uses the state of BOOT_MODE and eFUSEs to determine the boot device. For development purposes, eFUSEs used to determine the boot device may be overridden by using GPIO pin inputs.

Boot ROM code also allows the downloading of programs to be run on the device. An example is a provisioning program that can make further use of the serial connection to provision a boot device with a new image. Typically the provisioning program is downloaded to internal RAM and allows the programming of boot devices, such as an SD/MMC Flash. The ROM Serial Downloader uses high speed USB in a non-stream mode connection.

Boot ROM allows waking up from low-power modes and waking up the secondary core of Cortex-A9. On reset the ROM checks the ARM core ID and power gating status register. On waking from low power mode, the primary core (ID=0) will skip loading an image from the boot device and jump to the address saved in PERSISTENT_ENTRY0. The secondary cores (ID!=0) jump to the address in PERSISTENT_ENTRY<X> where 0<X<4 in all power modes.

The device configuration data (DCD) feature allows boot ROM code to obtain SOC configuration data from an external Program Image residing on the boot device. As an example, DCD can be used to program the DDR controller for optimal settings improving the boot performance. DCD is restricted to memory areas and peripheral addresses that are considered essential for boot purposes (see Write Data Command).

A key feature of the boot ROM is the ability to perform a secure boot or High Assurance Boot (HAB). This is supported by the HAB security library which is a subcomponent of the ROM code. HAB uses a combination of hardware and software together with a Public Key Infrastructure (PKI) protocol to protect the system from executing unauthorized programs. Before the HAB allows a user's image to execute, the image must be signed. The signing process is done during the image build process by the private key holder and the signatures are then included as part of the final Program Image. If configured to do so, the ROM verifies the signatures using the public keys included in the Program Image. In addition to supporting digital signature verification to authenticate Program Images, Encrypted boot is also supported. Encrypted boot can be used to prevent cloning of the Program Image directly off the boot device. A secure boot with HAB can be performed on all boot devices supported on the chip in addition to the Serial Downloader. The HAB library in the boot ROM also provides API functions, allowing additional boot chain components (bootloaders) to extend the secure boot chain. The out-of-fab setting for SEC_CONFIG is the Open configuration in which the ROM/HAB performs image authentication, but all authentication errors are ignored and the image is still allowed to execute.

8.2 Boot modes

During reset, the chip checks ARM core ID and Power Gating Controller status register.

On normal boot, the core's behavior is defined by the Boot Mode pins settings as described in Boot mode pin settings. On waking up from low power boot mode, the core skips clock settings. Boot ROM checks that PERSISTENT_ENTRY0 (see Persistent Bits) is a pointer to valid address space (OCRAM, DDR or EIM). If PERSISTENT_ENTRY0 is a pointer to valid range, it starts execution using entry point from PERSISTENT_ENTRY0 register. If PERSISTENT_ENTRY0 is a pointer to invalid range, the core performs system reset.

For secondary cores (with ID!=0) boot ROM checks that PERSISTENT_ENTRY is a pointer to valid address space (OCRAM, DDR, or EIM). If PERSISTENT_ENTRY is a pointer to valid range, it starts execution using entry point from PERSISTENT_ENTRY register. If PERSISTENT_ENTRY is a pointer to invalid range, it sets error status registers (see Persistent Bits), sends wakeup error interrupt and performs Wait For Interrupt instruction. The interrupt service routine of the other core must reconfigure the system and reset the secondary core that failed to boot.

NOTE

Code that enables the secondary cores is not part of ROM, but it must be part of upper level software.

8.2.1 Boot mode pin settings

The device has four boot modes (one is reserved for Freescale use). Boot mode is selected based on the binary value stored in the internal BOOT_MODE register.

BOOT_MODE is initialized by sampling the BOOT_MODE0 and BOOT_MODE1 inputs on the rising edge of POR_B. After these inputs are sampled, their subsequent state does not affect the contents of the BOOT_MODE internal register. The state of the internal BOOT_MODE register may be read from the BMOD[1:0] field of the SRC Boot Mode Register (SRC_SBMR2). The available boot modes are: Boot From Fuses, serial boot via USB, and Internal Boot. See the table below for settings.

Table 8-1. Boot MODE Pin Settings

BOOT_MODE[1:0]	Boot Type
00	Boot From Fuses
01	Serial Downloader
10	Internal Boot
11	Reserved

8.2.2 High level boot sequence

The figure found here shows the high-level boot ROM code flow.

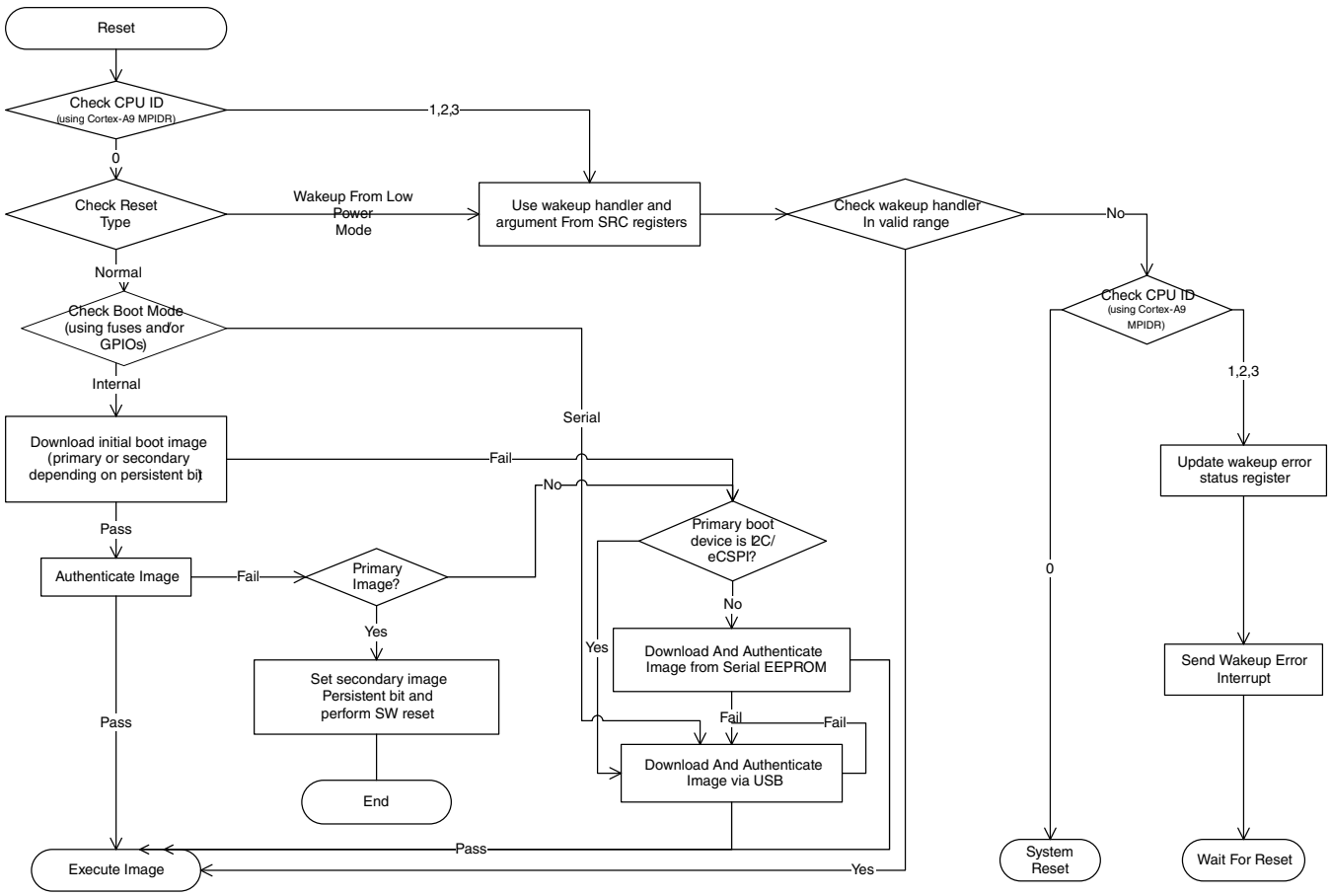


Figure 8-1. Boot Flow

8.2.3 Boot From Fuses Mode (BOOT_MODE[1:0] = 00b)

A value of 00b in the BOOT_MODE[1:0] register selects the Boot From Fuses mode.

This mode is similar to the Internal Boot mode described in Internal Boot Mode (BOOT_MODE[1:0] = 0b10)with one difference. In this mode the GPIO boot override pins are ignored. The boot ROM code uses the boot eFUSE settings only. This mode also supports a secure boot using HAB.

If set to Boot From Fuses, the boot flow is controlled by the BT_FUSE_SEL eFUSE value. If BT_FUSE_SEL = 0, indicating that the boot device (for example, Flash, SD/MMC) has not yet been programmed, the boot flow jumps directly to the Serial Downloader. If BT_FUSE_SEL = 1, the normal boot flow is followed, where the ROM attempts to boot from the selected boot device.

The first time a board is used, the default eFUSEs may be configured incorrectly for the hardware on the platform. In such a case, the Boot ROM code may try to boot from a device that does not exist. This may cause an electrical/logic violation on some pads. Using Boot From Fuses mode addresses this problem.

Setting BT_FUSE_SEL forces the ROM code to jump directly to the Serial Downloader. This allows a bootloader to be downloaded which can then provision the boot device with a Program Image and blow the BT_FUSE_SEL and the other boot configuration eFUSEs. After reset, the Boot ROM code determines that BT_FUSE_SEL is blown (BT_FUSE_SEL = 1) and the ROM code performs internal boot according to the new eFUSE settings. This allows a user to set BOOT_MODE[1:0]=00b on a production device and burn fuses on the same device (by forcing entry to the Serial Downloader), without changing the value of BOOT_MODE[1:0] or pullups/pulldowns on the BOOT_MODE pins.

8.2.4 Serial Downloader

The Serial Downloader provides a means to download a Program Image to the chip over USB serial connection.

In this mode the ROM programs WDOG-1 for a 32-second time-out if WDOG_ENABLE eFuse is 1, and then continuously polls for USB connection. If no activity is found on USB OTG1 and the watchdog timer expires, the ARM core is reset.

NOTE

The downloaded image must continue to service the watchdog timer to avoid an undesired reset from occurring.

The USB boot flow is shown in the figure below.

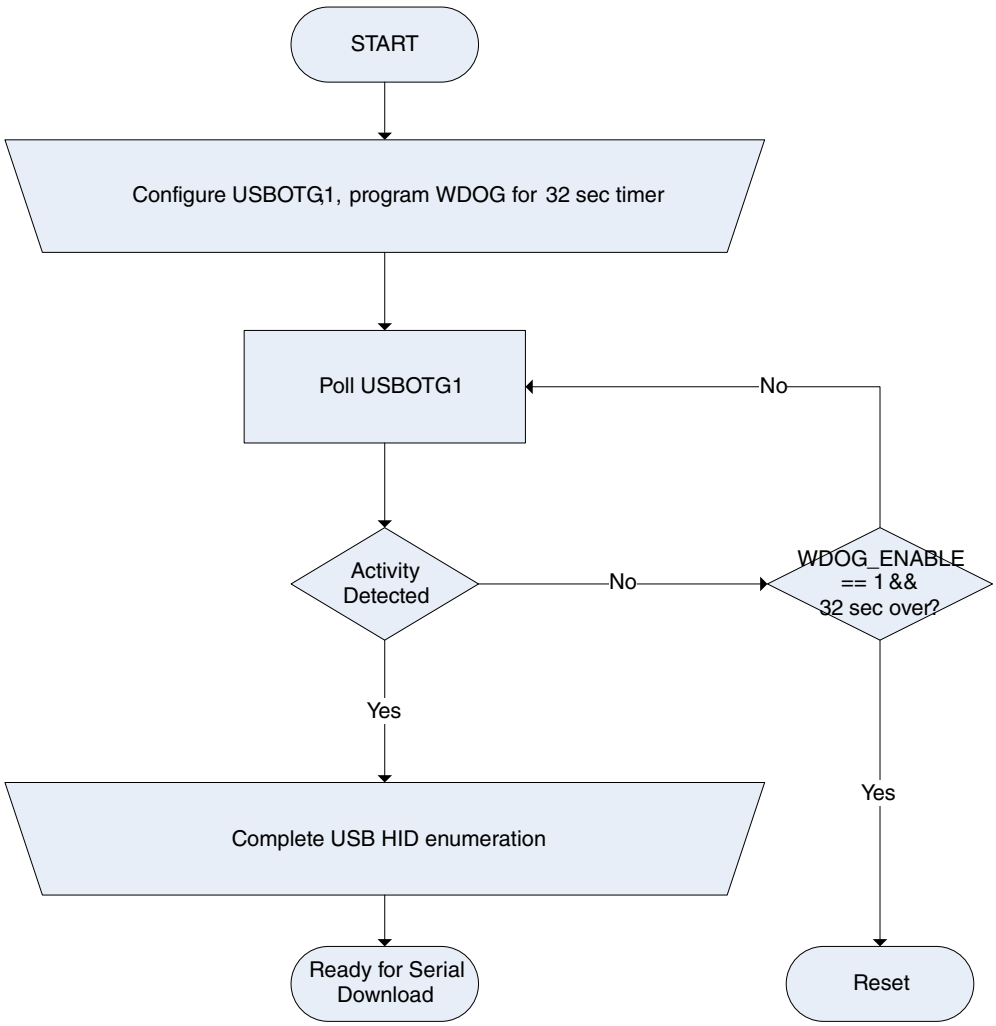


Figure 8-2. USB Boot Flow

8.2.5 Internal Boot Mode (BOOT_MODE[1:0] = 0b10)

A value of 0b10 in the BOOT_MODE[1:0] register selects the internal boot mode. In this mode, the processor continues to execute boot code from the internal boot ROM.

The boot code performs hardware initialization, loads the Program Image from the chosen boot device, performs image validation using the HAB library (see Boot security settings), and then jumps to an address derived from the Program Image. If any error occurs during internal boot, the boot code jumps to the Serial Downloader (see Serial Downloader). A secure boot using the HAB is possible in all the three boot modes.

When set to internal boot, the boot flow may be controlled by a combination of eFUSE settings with an option of overriding the fuse settings using General Purpose I/O (GPIO) pins. The GPIO Boot Select FUSE (BT_FUSE_SEL) determines whether the ROM uses GPIO pins for a select number of configuration parameters or eFUSEs in this mode. See Table 8-4 for more details.

- If BT_FUSE_SEL = 1, all boot options are controlled by the eFUSEs described in Table 8-2.
- If BT_FUSE_SEL = 0, specific boot configuration parameters may be set using GPIO pins rather than eFUSEs. The fuses that can be overridden when in this mode are indicated in the GPIO column of Table 8-2. Table 8-3 provides the details on the GPIO pins.

The use of GPIO overrides is intended for development since these pads are used for other purposes in deployed products. Freescale recommends controlling the boot configuration by eFUSEs in deployed products and reserving the use of the GPIO mode for development and testing purposes only.

8.2.6 Boot security settings

Internal boot modes use one of three security configurations:

- Closed: This level is intended for use with shipping secure products. All HAB functions are executed and security hardware is initialized (the Security Controller, or SNVS, enters Secure state), DCD is processed if present, and the program image is authenticated by HAB prior to its execution. All detected errors will be logged, and the boot flow aborted with control passing to the serial downloader. At this level, execution does not leave the internal ROM unless the target executable image has been authenticated.
- Open: This level is intended for use in non-secure products or during the development phases of a secure product. All HAB functions are executed as for a closed device. Security hardware is initialized (except the SNVS is left in Non-Secure state), DCD is processed if present, and the program image is authenticated by HAB prior to its execution. All detected errors will be logged, but have no influence on the boot flow, which continues as if the errors did not occur. This configuration is useful for secure product development, since the Program Image will run even if the authentication data is missing or incorrect, and the error log can be examined to determine the cause of authentication failure.
- Field Return: This level is intended for parts returned from shipped products.

NOTE

If the DIR_BT_DIS eFuse is not blown, authentication may be bypassed. In this case the system is not secure.

8.3 Device Configuration

This section describes the external inputs that control the behavior of the Boot ROM code.

This includes boot device selection (SPI, EIM, NOR, SD, MMC, etc.), boot device configuration (SD bus width, speed, etc), and so on. In general, the source for this configuration comes from eFUSEs embedded inside the chip. However, certain configuration parameters can be sourced from GPIO pins allowing further flexibility during the development process.

8.3.1 Boot eFUSE Descriptions

The table below is a comprehensive list of the configuration parameters that the ROM uses.

Table 8-2. Boot eFUSE Descriptions

Fuse	Configuration	Definition	GPIO[1]	Shipped Value	Settings[2]
DIR_BT_DIS	OEM	Disables Freescale reserved modes. Must be set for secure boot.	NA	0	0 Reserved Freescale modes enabled 1 Reserved Freescale modes disabled
BT_FUSE_SEL	OEM	In internal Boot mode BOOT_MODE[1:0] = 10, the BT_FUSE_SEL fuse determines whether the boot settings indicated by a Yes in the GPIO column are controlled by GPIO pins or eFUSE settings in the On-Chip OTP Controller (OCOTP). In Boot From Fuse mode BOOT_MODE[1:0] = 00, BT_FUSE_SEL fuse indicates whether bit configuration eFuses have been programmed.	NA	0	If BOOT_MODE[1:0] = 0b10 0 Bits of SBMR are overridden by GPIO pins. 1 Specific bits of SBMR are controlled by eFUSE settings. If BOOT_MODE[1:0] = 0b00 0 BOOT configuration eFuses are not yet programmed. Boot flow jumps to serial downloader. 1 BOOT configuration eFuses have been programmed. Regular boot flow is performed.

Table continues on the next page...

Table 8-2. Boot eFUSE Descriptions (continued)

Fuse	Configuration	Definition	GPIO[1]	Shipped Value	Settings[2]
SEC_CONFIG[1:0]	SEC_CONFIG[0] - Freescale SEC_CONFIG[1] - OEM	Security Configuration as defined in Boot security settings	NA	01	00 Reserved 01 Open (allows any program image, even if authentication fails) 1x Closed (Program image executes only if authenticated)
FIELD_RETURN	OEM	Enables Freescale reserved modes			0 - Freescale reserved modes are enabled/ disabled based on DIR_BT_DIS value 1 - Freescale reserved modes are enabled
SRK_HASH[255:0]	OEM	256-bit hash value of super root key (SRK_HASH)	NA	0	Settings vary - used by HAB
DIE-X-CORDINATE[7:0] DIE-Y-CORDINATE[7:0] WAFER_NO[4:0] LOT_NO_ENC[42:40] LOT_NO_ENC[39:32] LOT_NO_ENC[31:24] LOT_NO_ENC[23:16] LOT_NO_ENC[15:8] LOT_NO_ENC[7:0]	Freescale	Device Unique ID, 64-bit UID.	NA	Unique ID	Settings vary - used by HAB
BT_MMU_DISABLE (BOOT_CFG3[6])	OEM	MMU/L1 D Cache/PL310 disable bit used by boot ROM for fast HAB processing	Yes	0	0 - MMU/L1 D Cache/PL310 is enabled by ROM during the boot 1 - MMU/L1 D Cache/PL310 is disabled by ROM during the boot
L1 I-Cache DISABLE (BOOT_CFG3[7])	OEM	L1 I Cache disable bit used by boot during entire execution	Yes	0	0 - L1 I Cache is enabled by ROM during the boot 1 - L1 I Cache is disabled by ROM during the boot
BT_FREQ (BOOT_CFG3[2])	OEM	Frequency Selection	Yes	0	0 - ARM - 792 MHz, DDR - 396 MHz, AXI - 264 MHz 1 - ARM - 396 MHz, DDR - 352 MHz, AXI - 176 MHz
BOOT_CFG1[7:0]	OEM	Boot Configuration1	Yes	0	Specific to selected boot mode
BOOT_CFG2[7:0]	OEM	Boot Configuration2	Yes	0	Specific to selected boot mode
BOOT_CFG4[6:0]	OEM	Boot Configuration4		0	Specific to selected boot mode

Table continues on the next page...

Table 8-2. Boot eFUSE Descriptions (continued)

Fuse	Configuration	Definition	GPIO[1]	Shipped Value	Settings[2]
BOOT_CFG4[7]	OEM	Infinite Loop Enable at start of boot ROM. Used for debugging purposes. Ignored if DIR_BT_DIS is 1 and FIELD_RETURN is 0.	Yes	0	0 - Disabled 1 - Enabled
LPB_BOOT	OEM	USB Low Power Boot	No	0	00 - LPB Disable 01 - 1 GPIO (default frequencies) 10 - Divide by 2 11 - Divide by 4
BT_LPB_POLARITY	OEM	USB Low Power Boot GPIO polarity	No	0	0 - low on GPIO pad indicates low power condition 1 - high on GPIO pad indicates low power condition
WDOG_ENABLE	OEM	Watchdog reset counter enable	No	0	0 - watchdog reset counter is disabled during serial downloader 1 - watchdog reset counter is enabled during serial downloader
MMC_DLL_DLY[6:0]	OEM	uSDHC Delay Line settings	No	0	uSDHC Delay Line settings
SRK_REVOKE[2:0]	OEM	SRK revocation mask	No	0	SRK revocation mask
PAD_SETTINGS	OEM	Override values for SD/MMC and NAND boot modes	No	0	Override the following IO PAD settings: PAD_SETTINGS[0] - Slew Rate PAD_SETTINGS[3:1] Drive Strength PAD_SETTINGS[5:4] - Speed Settings .
OVERRIDE_HYS_SDMMC_PADS	OEM	Overrides HYS bit for SD pads	No	0	Override the IO PAD setting HYS to 1 for SD pads
eMMC_4.4_RESET_TO_PRE-IDLE_STATE	OEM	ROM reset the boot device in pre-idle state using eMMC 4.4 feature, CMD0 with argument value 0xf0f0f0f0	No	0	Applicable for booting from eMMC 4.4 spec or greater version devices. The fuse should not be blown for eMMC 4.3 or lesser spec version devices.

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.
2. 0 = intact fuse and 1= blown fuse

8.3.2 GPIO Boot Overrides

The table below provides a list of GPIO boot overrides.

Table 8-3. GPIO Override Contact Assignments

Package Pin	Direction on reset	eFuse
BOOT_MODE1	Input	Boot Mode Selection
BOOT_MODE0	Input	
EIM_DA0	Input	BOOT_CFG1[0]
EIM_DA1	Input	BOOT_CFG1[1]
EIM_DA2	Input	BOOT_CFG1[2]
EIM_DA3	Input	BOOT_CFG1[3]
EIM_DA4	Input	BOOT_CFG1[4]
EIM_DA5	Input	BOOT_CFG1[5]
EIM_DA6	Input	BOOT_CFG1[6]
EIM_DA7	Input	BOOT_CFG1[7]
EIM_DA8	Input	BOOT_CFG2[0]
EIM_DA9	Input	BOOT_CFG2[1]
EIM_DA10	Input	BOOT_CFG2[2]
EIM_DA11	Input	BOOT_CFG2[3]
EIM_DA12	Input	BOOT_CFG2[4]
EIM_DA13	Input	BOOT_CFG2[5]
EIM_DA14	Input	BOOT_CFG2[6]
EIM_DA15	Input	BOOT_CFG2[7]
EIM_A16	Input	BOOT_CFG3[0]
EIM_A17	Input	BOOT_CFG3[1]
EIM_A18	Input	BOOT_CFG3[2]
EIM_A19	Input	BOOT_CFG3[3]
EIM_A20	Input	BOOT_CFG3[4]
EIM_A21	Input	BOOT_CFG3[5]
EIM_A22	Input	BOOT_CFG3[6]
EIM_A23	Input	BOOT_CFG3[7]
EIM_A24	Input	BOOT_CFG4[0]
EIM_WAIT	Input	BOOT_CFG4[1]
EIM_LBA	Input	BOOT_CFG4[2]
EIM_EB0	Input	BOOT_CFG4[3]
EIM_EB1	Input	BOOT_CFG4[4]
EIM_RW	Input	BOOT_CFG4[5]
EIM_EB2	Input	BOOT_CFG4[6]
EIM_EB3	Input	BOOT_CFG4[7]

The input pins provided are sampled at boot, and can be used to override corresponding eFUSE values, depending on the setting of the BT_FUSE_SEL fuse. Table below describes boot options control sampling in different boot modes.

Table 8-4. Boot Options Control Selection

BOOT_MODE[1:0]	BT_FUSE_SEL Value	Boot Options Controlled By
00	0	eFUSEs
	1	
01[1]	0	GPIO pins
	1	eFUSEs
10	0	GPIO pins
	1	eFUSEs

1. BOOT_MODE[1:0]=01 always selects serial download regardless of the setting or the fuses or GPIO pins.

8.3.3 Device Configuration Data

DCD is configuration information contained in a Program Image, external to the ROM, that the ROM interprets to configure various on-chip peripherals. See Device Configuration Data (DCD) for more details on Device Configuration Data.

8.4 Device Initialization

This section describes the details on the ROM and provides initialization details.

This includes details on:

- The ROM Memory Map
- The RAM Memory Map
- On-chip blocks that the ROM should make use of or change POR register default values
- Clock initialization
- Enabling the MMU/L2 cache
- Exception handling and interrupt handling

8.4.1 Internal ROM /RAM memory map

Figure 8-3 shows the iROM memory map for the i.MX 6Dual/6Quad.

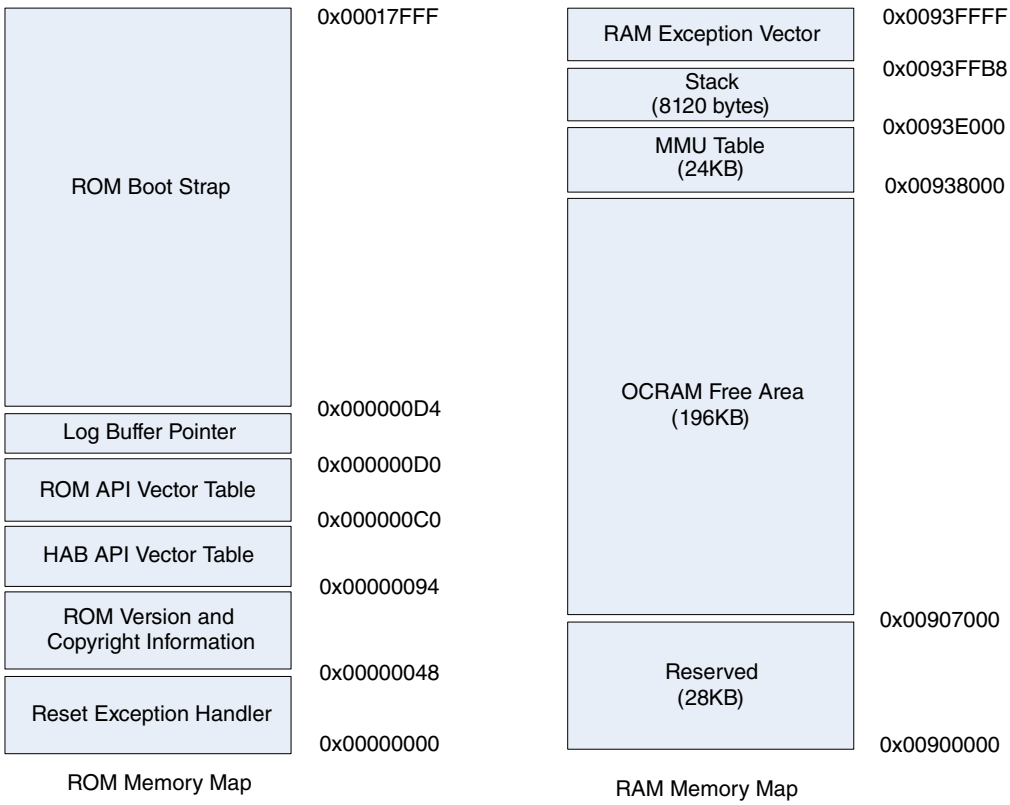


Figure 8-3. Internal ROM and RAM memory map for i.MX 6Dual/6Quad

NOTE

The entire OCRAM region can be used freely post boot.

8.4.2 Boot Block Activation

The boot ROM affects a number of different hardware blocks which are activated and play a vital role in the boot flow.

The ROM configures and uses the following blocks (listed in alphabetical order) during the boot process. Note that the blocks actually used depend on the boot mode and boot device selection:

- APBH - DMA engine to drive the GPMI module
- BCH - 40-bit error correction hardware engine with AXI bus master and private connection to GPMI
- CCM - Clock Control Module
- ECSPI - Enhanced Configurable Serial Peripheral Interface
- EIM - External Interface Module. Used for NOR and OneNAND devices

- I2C - I2C Controller
- GPMI - NAND controller pin interface
- OCOTP_CTRL - On-Chip OTP Controller. The OCOTP contains the eFUSEs.
- IOMUXC - I/O Multiplexer Control allows GPIO use to override eFUSE boot settings
- IOMUXC GPR - I/O Multiplexer Control General Purpose registers
- CAAM - Cryptographic Acceleration and Assurance Module
- SNVS - Secure Non-Volatile Storage
- SRC - System Reset Controller
- USB - Used for serial download of a boot device provisioning program
- USDHC - Ultra Secure Digital Host Controller
- WDOG-1 - Watchdog Timer
- DTCP - Digital Transmission Content Protection
- HDCP - High-bandwidth Digital Content Protection

8.4.3 Clocks at Boot Time

The table below shows the various clocks and their sources used by ROM.

Table 8-5. Normal Frequency Clocks Configuration

Clock	CCM signal	Source	Frequency(MHz) BT_FREQ=0	Frequency(MHz) BT_FREQ=1
System PLL	pll1_sw_clk		792	792
528MHz PLL	pll2_sw_clk		528	528
480MHz PLL	pll3_sw_clk		480	480
ARM core clock	arm_clk_root	System PLL	792	396
EIM	eim_slow_clk_root	528MHz PLL	132	132
AHB	ahb_clk_root	528MHz PLL/PFD352	132	88
IPG	ipg_clk_root	528MHz PLL/PFD352	66	44
AXI_A	axi_a	528MHz PLL/PFD352	528	352
AXI_B	axi_b	528MHz PLL/PFD352	264	176
USB	usboh3_clk_root	480MHz PLL	60	60
USDHC	usdhc1_clk_root usdhc2_clk_root usdhc3_clk_root usdhc4_clk_root	PFD400	198	198
ECSPI	ecspi_clk_root	480MHz PLL	60	60
I2C	per_clk_root	528MHz PLL	66	38.3

Following reset, each ARM core has access to all peripherals. The ROM code will disable the clocks listed in the following table, except for the boot devices listed in the second column.

Table 8-6. List Of Disabled Clocks

Clock Name	Enabled For Boot Device
CCGR0_APBHDMA	
CCGR1_ECSPI1	ECSPI1
CCGR1_ECSPI2	ECSPI2
CCGR1_ECSPI3	ECSPI3
CCGR1_ECSPI4	ECSPI4
CCGR1_FEC	
CCGR1_EPIT1	
CCGR1_EPIT2	
CCGR2_I2C1_SERIAL	I2C1
CCGR2_I2C2_SERIAL	I2C2
CCGR2_I2C3_SERIAL	I2C3
CCGR3_OPENVGAXICLK	
CCGR4_PWM1	
CCGR4_PWM2	
CCGR4_PWM3	
CCGR4_PWM4	
CCGR5_SDMA	
CCGR5_SPDIF	
CCGR5_SSI1	
CCGR5_SSI2	
CCGR5_SSI3	
CCGR5_UART	
CCGR5_UART_SERIAL	
CCGR6_USBOH3	USB
CCGR6_USDHC1	USDHC1
CCGR6_USDHC2	USDHC2
CCGR6_USDHC3	USDHC3
CCGR6_USDHC4	USDHC4
CCGR6_EIM_SLOW	NOR, OneNAND

8.4.4 Enabling MMU and Caches

The boot ROM includes a feature of enabling the Memory Management Unit (MMU) and caches to improve boot speed.

L1 instruction cache is enabled at the start of image download. L1 data cache, L2 cache and MMU are enabled during image authentication. Once HAB authentication completes the ROM disables the L1 data cache, L2 cache and MMU.

L1 Instruction cache, L1 data cahce, L2 cache and MMU is controlled by eFuse. By default these features are enabled.

Enabling the MMU when booting non-securely with SEC_CONFIG=Open, and setting the CSF pointer in the Image Vector Table to NULL, has no impact on the boot performance. With this configuration it is recommended to blow BT_MMU_DISABLE fuse.

8.4.5 Exception Handling

The exception vectors located at the start of ROM are used to map all the ARM exceptions (except the reset exception) to a duplicate exception vector table in internal RAM.

During the boot phase of CPU0, the RAM vectors point to the serial downloader in ROM.

During the boot phase of a secondary CPU, the internal RAM vectors point to a function that sets the error status registers (see Persistent Bits), sends a wakeup error interrupt and performs the Wait For Interrupt instruction. The interrupt service routine of primary CPU must reconfigure the system and reset the secondary CPU.

After boot the program image can overwrite the vectors as required. The code shown below is used to map the ROM exception vector table to the duplicate one in RAM.

Mapping ROM Exception Vector Table

```
;; Define linker area for ROM exception vector table
AREA IROM_VECTORS, CODE, READONLY
LDR     PC, Reset_Addr
LDR     PC, Undefined_Addr
LDR     PC, SWI_Addr
LDR     PC, Prefetch_Addr
LDR     PC, Abort_Addr
NOP                         ; Reserved vector
LDR     PC, IRQ_Addr
LDR     PC, FIQ_Addr

;; Define exception vector table
Reset_Addr      DCD     start_address
Undefined_Addr  DCD     iRAM_Undefined_Handler
SWI_Addr        DCD     iRAM_SWI_Handler
Prefetch_Addr   DCD     iRAM_Prefetch_Handler
Abort_Addr      DCD     iRAM_Abort_Handler
                DCD     0               ; Reserved vector
IRQ_Addr        DCD     iRAM_IRQ_Handler
FIQ_Addr        DCD     iRAM_FIQ_Handler
```

```
start_address DCD start ;reset handler vector
```

8.4.6 Interrupt Handling During Boot

No special interrupt handling routines are required during the boot process. Interrupts are disabled during boot ROM execution and may be enabled in a later boot stage.

8.4.7 Persistent Bits

Some modes of boot ROM require registers that keep their values after warm reset. SRC General Purpose registers are used for this purpose.

See the table below for persistent bits list and description.

Table 8-7. Persistent Bits

Bit Name	Bit Location	Description
PERSIST_SECONDARY_BOOT	SRC_GPR10[30]	This bit identifies which image must be used - primary and secondary. Used only for boot modes that support redundant boot.
PERSIST_BLOCK_REWRITE	SRC_GPR10[29]	This bit is used as warning. It identifies that there are errors in NAND blocks that hold the application image. See NAND Flash for more details.
PERSISTENT_ENTRY0[31:0]	SRC_GPR1[31:0]	Holds entry function for CPU0 for waking-up from low power mode.
PERSISTENT_ARG0[31:0]	SRC_GPR2[31:0]	Holds argument of entry function for CPU0 for waking-up from low power mode.
PERSISTENT_ENTRY1[31:0]	SRC_GPR3[31:0]	Holds entry function for CPU1.
PERSISTENT_ARG1[31:0]	SRC_GPR4[31:0]	Holds argument of entry function for CPU1.
PERSISTENT_ENTRY2[31:0]	SRC_GPR5[31:0]	Holds entry function for CPU2 (i.MX 6Quad only).
PERSISTENT_ARG2[31:0]	SRC_GPR6[31:0]	Holds argument of entry function for CPU2 (i.MX 6Quad only).
PERSISTENT_ENTRY3[31:0]	SRC_GPR7[31:0]	Holds entry function for CPU3 (i.MX 6Quad only).
PERSISTENT_ARG3[31:0]	SRC_GPR8[31:0]	Holds argument of entry function for CPU3 (i.MX 6Quad only).
CPU3_ERROR_STATUS	SRC_GPR10[27]	CPU3 error status bit (i.MX 6Quad only)
CPU2_ERROR_STATUS	SRC_GPR10[26]	CPU2 error status bit (i.MX 6Quad only)
CPU1_ERROR_STATUS	SRC_GPR10[25]	CPU1 error status bit

8.5 Boot Devices (Internal Boot)

The Chip supports the following boot Flash devices:

- NOR Flash with External Interface Module (EIM), located on CS0, 16-bit bus width
- OneNAND Flash with EIM interface, located on CS0, 16-bits bus width
- Raw NAND (MLC and SLC), and Toggle-mode NAND flash through GPMI-2 interface. Page sizes of 2 Kbyte, 4 Kbyte and 8 Kbyte. Bus widths of 8-bit with 2 through 40-bit BCH Hardware ECC (Error Correction) are supported.
- SD/MMC/eSD/SDXC/eMMC4.4 via USDHC interface, supporting high capacity cards
- EEPROM boot via SPI (serial flash) and I2C (via ECSPI and I2C blocks respectively)
- Serial ATA (SATA) boot via SATA interface

The selection of external boot device type is controlled by BOOT_CFG1[7:4] eFUSEs. See the table below for more details.

Table 8-8. Boot Device Selection

BOOT_CFG1[7:4]	Boot Device
0000	NOR/OneNAND (EIM)
0001	Reserved
0010	SSD/Hard Disk (SATA)
0011	Serial ROM (I2C/SPI)
010x	SD/eSD/SDXC
011x	MMC/eMMC
1xxx	Raw NAND

8.5.1 NOR Flash/OneNAND using EIM Interface

The External Interface Module (EIM) works in the asynchronous mode, and supports either muxed, Address/Data, or non-muxed schemes based on fuse settings.

Table 8-9. EIM Boot eFUSE Descriptions

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG1[7:4]	OEM	Boot Device Selection	Yes	0000	0000 - Boot from EIM Interface
BOOT_CFG1[3]	OEM	NOR/OneNAND Selection	Yes	0	0 - NOR 1 - OneNAND

Table continues on the next page...

Table 8-9. EIM Boot eFUSE Descriptions (continued)

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG2[7:6]	OEM	Muxing Scheme	Yes	00	00 - Muxed, 16-bit data (low half) interface 01 - Not muxed, 16-bit data (high half) interface 10 - Not muxed, 16-bit data (low half) interface 11 - Reserved
BOOT_CFG2[5:4]	OEM	OneNAND Page Size	Yes	00	00 - 1K 01 - 2K 10 - 4K 11 - Reserved

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.

8.5.1.1 NOR Flash Boot Operation

Booting from the NOR Flash is supported via EIM interface. The ROM reads Image Vector Table and Boot Data structures to determine if the image can be executed directly from EIM address space or should be copied to other memory.

The start field of Boot Data Structure specifies the final location of the image (see Image Vector Table and Boot Data).

8.5.1.2 OneNAND Flash Boot Operation

At system power-up, the OneNAND device automatically copies an Initial Load Region of 1 Kbyte from the start of the flash array (sector 0 and sector 1, page 0, block 0) to its Boot RAM (OneNAND's internal RAM).

NOTE

The OneNAND boot RAM memory containing the Initial 1K Load Region must contain the IVT, DCD and the Boot Data structures.

Next, the ROM processes the DCD and then proceeds to copy the Program Image contents to the application destination pointer (located in the start entry of Boot Data (see Image Vector Table and Boot Data). The ROM determines the size of the Program Image by the length specified by size entry in Boot Data structure (see Image Vector Table and

Boot Data). A failure loading data from the OneNAND device for any reason forces the Chip to enter the Serial Downloader, otherwise the booting from the OneNAND device continues.

The figure below illustrates the layout of the Program Image on a OneNAND boot device.

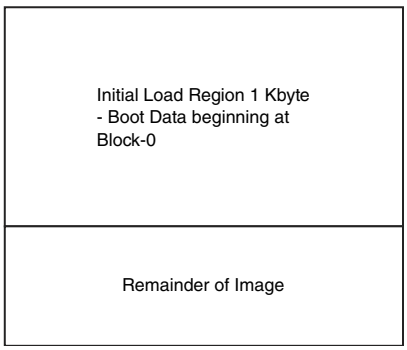


Figure 8-4. Program Image Layout on a OneNAND Flash Device

Prior to accessing the OneNAND device, the Chip waits approximately 500 µs after Power On Reset. This delay is required for the OneNAND device to become ready. After this initial 500 µs delay it can take an addition 70 µs for the OneNAND device to load the Initial Load Region of 1 Kbyte into its boot RAM. The Chip polls the OneNAND device Interrupt Status Register to confirm that the first 1 Kbytes has been loaded to the OneNAND boot RAM before continuing with the boot flow.

8.5.1.3 IOMUX Configuration for EIM Devices

The EIM interface uses dedicated contacts on the IC.

The contacts assigned to the data signals used by EIM are shown in the table below.

Table 8-10. EIM IOMUX Pin Configuration

Signal	A/D16 (Muxed, 16-bit data low half interface)	A+DH (Not muxed, 16-bit data high half interface)	A+DL (Not muxed, 16-bit data low half interface)
DATA0	EIM_DA0.alt0	EIM_D16.alt0	CSI0_DATA_EN.alt1
DATA1	EIM_DA1.alt0	EIM_D17.alt0	CSI0_VSYNC.alt1
DATA2	EIM_DA2.alt0	EIM_D18.alt0	CSI0_DAT4.alt1
DATA3	EIM_DA3.alt0	EIM_D19.alt0	CSI0_DAT5.alt1
DATA4	EIM_DA4.alt0	EIM_D20.alt0	CSI0_DAT6.alt1
DATA5	EIM_DA5.alt0	EIM_D21.alt0	CSI0_DAT7.alt1
DATA6	EIM_DA6.alt0	EIM_D22.alt0	CSI0_DAT8.alt1

Table continues on the next page...

Table 8-10. EIM IOMUX Pin Configuration (continued)

Signal	A/D16 (Muxed, 16-bit data low half interface)	A+DH (Not muxed, 16-bit data high half interface)	A+DL (Not muxed, 16-bit data low half interface)
DATA7	EIM_DA7.alt0	EIM_D23.alt0	CSI0_DAT9.alt1
DATA8	EIM_DA8.alt0	EIM_D24.alt0	CSI0_DAT12.alt1
DATA9	EIM_DA9.alt0	EIM_D25.alt0	CSI0_DAT13.alt1
DATA10	EIM_DA10.alt0	EIM_D26.alt0	CSI0_DAT14.alt1
DATA11	EIM_DA11.alt0	EIM_D27.alt0	CSI0_DAT15.alt1
DATA12	EIM_DA12.alt0	EIM_D28.alt0	CSI0_DAT16.alt1
DATA13	EIM_DA13.alt0	EIM_D29.alt0	CSI0_DAT17.alt1
DATA14	EIM_DA14.alt0	EIM_D30.alt0	CSI0_DAT18.alt1
DATA15	EIM_DA15.alt0	EIM_D31.alt0	CSI0_DAT19.alt1
ADDR0	EIM_DA0.alt0		
ADDR1	EIM_DA1.alt0		
ADDR2	EIM_DA2.alt0		
ADDR3	EIM_DA3.alt0		
ADDR4	EIM_DA4.alt0		
ADDR5	EIM_DA5.alt0		
ADDR6	EIM_DA6.alt0		
ADDR7	EIM_DA7.alt0		
ADDR8	EIM_DA8.alt0		
ADDR9	EIM_DA9.alt0		
ADDR10	EIM_DA10.alt0		
ADDR11	EIM_DA11.alt0		
ADDR12	EIM_DA12.alt0		
ADDR13	EIM_DA13.alt0		
ADDR14	EIM_DA14.alt0		
ADDR15	EIM_DA15.alt0		

8.5.2 NAND Flash

The boot ROM supports a number of MLC/SLC NAND Flash devices from different vendors and LBA NAND Flash devices. The Error Correction and Control (ECC) subblock (BCH) is used to detect the errors.

8.5.2.1 NAND eFUSE Configuration

The boot ROM determines the configuration of external the NAND flash by parameters, either provided by eFUSE, or sampled on GPIO pins, during boot.

See the table below for parameters details.

Table 8-11. NAND Boot eFUSE Descriptions

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG1[7]	OEM	Boot Device Selection	Yes	0	1 - Boot from NAND Interface
BOOT_CFG1[5]	OEM	BT_TOGGLEMODE	Yes	0	0 - raw NAND 1 - toggle mode NAND
BOOT_CFG1[4]	OEM	Override Pad Settings	Yes	0	0 - Use default values 1 - Use PAD_SETTINGS values
BOOT_CFG1[3:2]	OEM	Number of devices	Yes	00	00 - 1 device 01- 2 device 10- 4 device 11- Reserved
BOOT_CFG1[1:0]	OEM	Row Address Cycles	Yes	00	00 - 3 01 - 2 10 - 4 11 - 5
BOOT_CFG2[7:5]	OEM	Toggle Mode 33MHz Preamble Delay, Read Latency	Yes	000	000 - 16 GPMICLK cycles. 001 - 1 GPMICLK cycles 010 - 2 GPMICLK cycles 011 - 3 GPMICLK cycles 100 - 4 GPMICLK cycles 101 - 5 GPMICLK cycles 110 - 6 GPMICLK cycles 111 - 7 GPMICLK cycles
BOOT_CFG2[4:3]	OEM	Boot Search Count	Yes	00	00 - 2 01 - 2 10 - 4 11 - 8
BOOT_CFG2[2:1]	OEM	Pages In Block	Yes	00	00 - 128 01 - 64 10 - 32 11 - 256
BOOT_CFG2[0]	OEM	Reset time	Yes	0	0 - 12ms 1 - 22ms (LBA NAND)

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.

8.5.2.2 NAND Flash Boot Flow and Boot Control Blocks (BCB)

There are two BCB data structures: FCB and DBBT.

As part of the NAND media initialization, the ROM driver uses safe NAND timings to search for a Firmware Configuration Block (FCB) that contains the optimum NAND timings, page address of Discovered Bad Block Table (DBBT) Search Area and start page address of primary and secondary firmware.

The hardware ECC level to use is embedded inside FCB block. The FCB data structure is protected using software ECC (SEC-DED Hamming Codes). Driver reads raw 2112 bytes of first sector and runs through software ECC engine that determines whether FCB data is valid or not.

If the FCB is found, the optimum NAND timings are loaded for further reads. If the ECC fails, or the fingerprints do not match, the Block Search state machine increments page number to Search Stride number of pages to read for the next BCB until SearchCount pages have been read.

If search fails to find a valid FCB, the NAND driver responds with an error and the boot ROM enters into serial download mode.

The FCB contains the page address of DBBT Search Area, and the page address for primary and secondary boot images. DBBT is searched in DBBT Search Area just like how FCB is searched. After the FCB is read, the DBBT is loaded, and the primary or secondary boot image is loaded using starting page address from FCB.

The state diagram of FCB search is shown in the following figure.

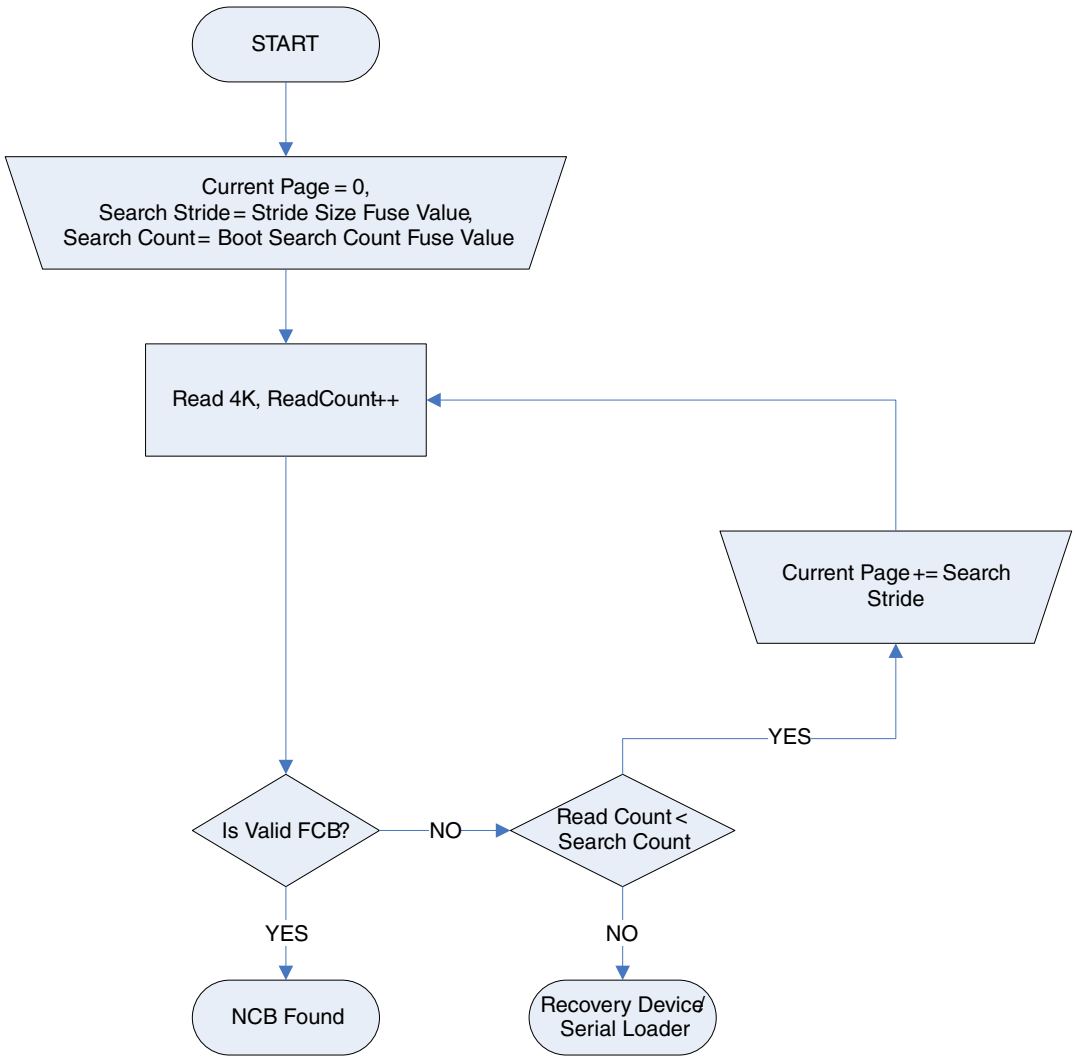


Figure 8-5. FCB Search Flow

Once FCB is found, the boot ROM searches for the Discovered Bad Blocks Table (DBBT). If DBBT Search Area is 0 in FCB, then ROM assumes that there are no bad blocks on NAND device. See the figure below for the DBBT search flow.

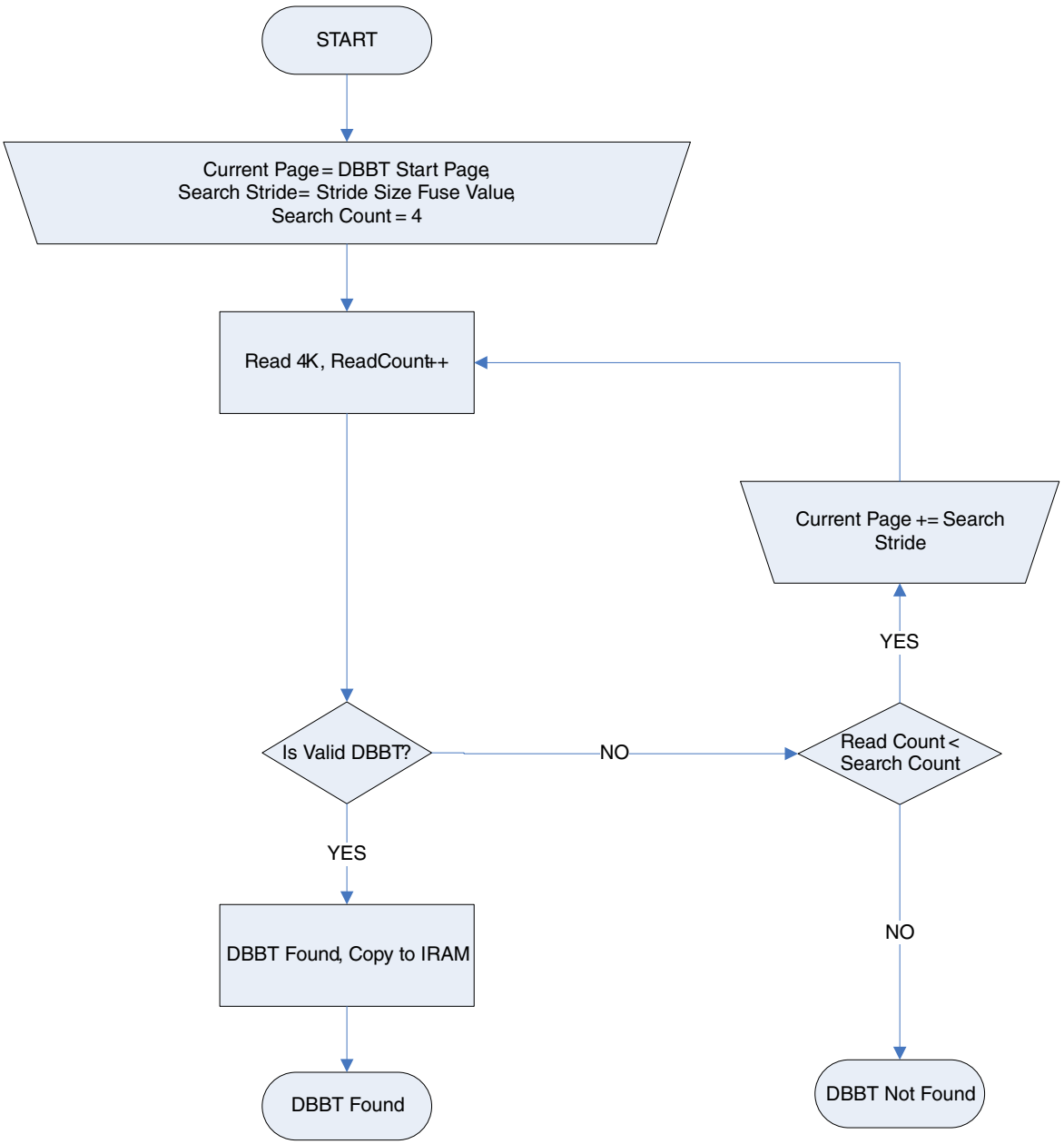


Figure 8-6. DBBT Search Flow

The BCB search and load function also monitors the ECC correction threshold and sets the PERSIST_BLOCK_REWRITE persistent bit if the threshold exceeds the maximum ECC correction ability.

If during primary image read there is a page with a number of errors higher than ECC can correct, the boot ROM will turn on PERSIST_SECONDARY_BOOT bit and perform SW reset. (After SW reset, secondary image will be used.)

If during secondary image read there is a page with number of errors higher than ECC can correct, the boot ROM will go to serial loader.

8.5.2.3 Firmware Configuration Block

The FCB is the first sector in the first good block. The FCB should be present at each search stride of the search area.

The search area contains copies of the FCB at each stride distance, so in case the first NAND block becomes corrupted, the ROM will find its copy in the next NAND block. The search area should span over at least two NAND blocks. The location information for DBBT search area, FW1, and FW2 are all specified in the FCB. Flash Control Block Structure is as shown in the table below.

Table 8-12. Flash Control Block Structure

Name	Start Byte	Size in Bytes	Description
Reserved	0	4	Reserved for Fingerprint #1(Checksum)
FingerPrint	4	4	32 bit word with a value of 0x4E434220, in ascii "FCB"
Version	8	4	32-bit version number; this version of FCB is 0x00000001
m_NANDTiming	12	8	8 bytes of data for 8 NAND Timing Parameters from NAND datasheet. The 8 parameters are: m_NandTiming[0]=data_setup, m_NandTiming[1]=data_hold, m_NandTiming[2]=address_setup, m_NandTiming[3]=dsample_time, m_NandTiming[4]=nand_timing_state, m_NandTiming[5]=REA, m_NandTiming[6]=RLOH, m_NandTiming[7]=RHOH. ROM only uses first 4 parameters but FCB provides space for other 4 parameters to be used by boot-loader or other applications.
PageDataSize	20	4	Number of bytes of data in a page. Typically, this is 2048 bytes for 2112 bytes page size or 4096 bytes for 4314/4224 bytes page size or 8192 for 8568 bytes page size
TotalPageSize	24	4	Total number of bytes in page. Typically, 2112 for 2 KB page or 4224 or 4314 for 4 KB page or 8568 for 8 KB page.
SectorsPerBlock	28	4	Number of pages per block. Typically 64 or 128 or depending on NAND device type.

Table continues on the next page...

Table 8-12. Flash Control Block Structure (continued)

Name	Start Byte	Size in Bytes	Description
NumberOfNANDs	32	4	Not used by ROM
TotalInternalDie	36	4	Not used by ROM
CellType	40	4	Not used by ROM
EccBlockNEccType	44	4	Value from 0 to 16 used to set BCH Error Corrrection level 0, 2, 4, .. or 40 for Block BN of ECC page, used in configuring BCH40 page layout registers
EccBlock0Size	48	4	Size of block B0, used in configuring BCH40 page layout registers
EccBlockNSize	52	4	Size of block BN, used in configuring BCH40 page layout registers
EccBlock0EccType	56	4	Value from 0 to 16 used to set BCH Error Corrrection level 0, 2, 4, .. or 40 for Block BN of ECC page, used in configuring BCH40 page layout registers
MetadataBytes	60	4	Size of metadata bytes used in configuring BCH40 page layout registers
NumEccBlocksPerPage	64	4	Number of ECC blocks BN not including B0. This value is used in configuring BCH40 page layout registers
EccBlockNEccLevelSDK	68	4	Not used by ROM
EccBlock0SizeSDK	72	4	Not used by ROM
EccBlockNSizeSDK	76	4	Not used by ROM
EccBlock0EccLevelSDK	80	4	Not used by ROM
NumEccBlocksPerPageSDK	84	4	Not used by ROM
MetadataBytesSDK	88	4	Not used by ROM
EraseThreshold	92	4	Not used by ROM
Firmware1_startingPage	104	4	Page number address where first copy of bootable firmware is located
Firmware2_startingPage	108	4	Page number address where second copy of bootable firmware is located
PagesInFirmware1	112	4	Size of first copy of firmware in pages
PagesInFirmware2	116	4	Size of second copy of firmware in pages
DBBTSearchAreaStartAddress	120	4	Page address for bad block table search area
BadBlockMarkerByte	124	4	This is an input offset in BCH page for ROM to swap with first byte of metadata after reading a page using BCH40. ROM supports restoration of manufacturer marked bad block markers in the page and this offset is the bad block marker offset location
BadBlockMarkerStartBit	128	4	This is an input bit offset in BadBlockMarkerByte for ROM to use when swapping 8 bits with fiirst byte of metadata.
BBMarkerPhysicalOffset	132	4	This is the offset where manufacturer leaves bad block marker on a page

Table continues on the next page...

Table 8-12. Flash Control Block Structure (continued)

Name	Start Byte	Size in Bytes	Description
BCHType	136	4	0 for BCH20 and 1 for BCH40. The Chip is backward compatible to BCH20 and this field tell ROM to use BCH20 or BCH40 block
TMTiming2_ReadLatency	140	4	Toggle mode NAND timing parameter read latency, ROM use this value to configure timing2 register of GPMI
TMTiming2_PreambleDelay	144	4	Toggle mode NAND timing parameter Preamble Delay. ROM use this value to configure timing2 register of GPMI
TMTiming2_CEDelay	148	4	Toggle mode NAND timing parameter CE Delay. ROM use this value to configure timing2 register of GPMI
TMTiming2_PostambleDelay	152	4	Toggle mode NAND timing parameter Postamble Delay. ROM use this value to configure timing2 register of GPMI
TMTiming2_CmdAddPause	156	4	Toggle mode NAND timing parameter Cmd Add Pause. ROM use this value to configure timing2 register of GPMI
TMTiming2_DataPause	160	4	Toggle mode NAND timing parameter Data Pause. ROM use this value to configure timing2 register of GPMI
TMSpeed	164	4	This is the toggle mode speed for ROM to configure gpmi clock. 0 for 33 MHz, 1 for 40 MHz and 2 for 66 MHz
TMTiming1_BusyTimeout	168	4	Toggle mode NAND timing parameter Busy Timeout. ROM use this value to configure timing1 register of GPMI
DISBBM	172	4	If 0 ROM will swap BadBlockMarkerByte with metadata[0] after reading a page using BCH40. If the value set is 1 then ROM will not do swapping
BBMark_spare_offset	176	4	The offset in mata data place which stores the data in Bad block marker place.
Onfi_sync_enable	180	4	Enable the Onfi nand sync mode support
Onfi_sync_speed	184	4	Speed for onfi nand sync mode: 0 - 24MHZ, 1 - 33MHZ, 2 - 40MHZ, 3 - 50MHZ, 4 - 66MHZ, 5 - 80MHZ, 6 - 100MHZ, 7 - 133MHZ, 8 - 160MHZ, 9 - 200MHZ
Onfi_syncNANDData	188	28	parameters for onfi nand sync mode timing. They are read latency, ce_delay, preamble_delay, postamble_delay, cmdadd_pause, data_pause, busy_timeout
DISBB_Search	216	4	Disable the badblock search function when reading the firmware, only using DBBT.

8.5.2.4 Discovered Bad Block Table

See the table below for DBBT format.

Table 8-13. DBBT Structure

Name	Start Byte	Size in Bytes	Description
reserved	0	4	-
FingerPrint	4	4	32-bit word with a value of 0x44424254,in ascii "DBBT"
Version	8	4	32-bit version number; this version of DBBT is 0x00000001
reserved	12	4	-
DBBT_NUM_OF_PAGES	16	4	Size of DBBT in pages
reserved	20	4*PageSize-20	-
reserved	4*PageSize	4	-
Number of Entries	4*PageSize + 4	4	Number of bad blocks
Bad Block Number	4*PageSize + 8	4	First bad block number
Bad Block Number	4*PageSize + 12	4	Second bad block number
...-	-	-	...next bad block number
...-	-	-	...-
Last bad block number	-	-	last bad block number

8.5.2.5 Bad Block Handling in the ROM

During firmware boot, at the block boundary, the Bad Block table is searched for a match to the next block.

If no match is found, the next block can be loaded. If a match is found, the block must be skipped and the next block checked.

If Bad Block table start page is null, check the manufactory made Bad Block marker. The location of Bad Block maker is at the first 3 or last 3 pages in every block of the NAND flash. NAND manufacturers normally use one byte in the spare area of certain pages within a block to mark a block is bad or not. 0xFF means good block, non FF means bad block.

In order to preserve the BI (bad block information), flash updater or gang programmer applications need to swap Bad Block Information (BI) data to byte 0 of metadata area for every page before programming NAND flash. ROM when loading firmware copies back the value at metadata[0] to BI offset in page data. The figure below shows how the factory bad block marker is preserved.

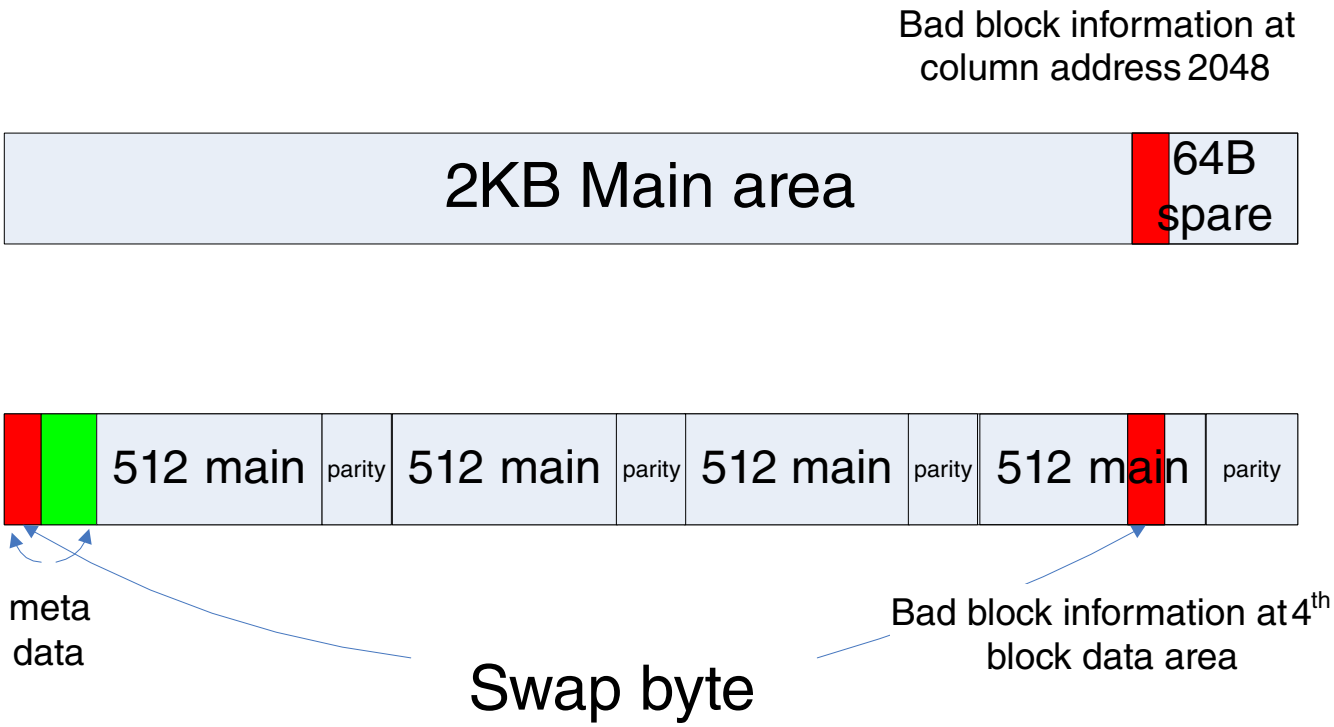


Figure 8-7. Factory Bad Block Marker Preservation

In the FCB structure, there are two elements m_u32BadBlockMarkerByte and m_u32BadBlockMarkerStartBit to indicate the byte and bit place in the page data, that manufacturer marked the bad block marker.

8.5.2.6 Toggle Mode DDR NAND Boot

If BT_TOGGLEMODE efuse is blown then ROM does the following to boot from Samsung's toggle mode DDR NAND.

8.5.2.6.1 GPMI and BCH Clocks Configuration

ROM sets the clock source and the dividers in CCM registers.

If BOOT_CFG1[5] is set(toggle mode), GPMI/BCH CLK source is PLL2PFD4, and running at 66MHZ, otherwise GPMI/ BCH CLK souce is PLL3, running at 24 Mhz. ROM sets default values to timing0, timing1 and timing2 gpmi registers for 24 MHz clock speed. It uses fuse BOOT_CFG2[7:5] to configure GPMI timing2 register parameters preamble delay and read latency, the default value for these parameters is 2 when fuses are not blown.

Default timing parameter values used by ROM for toggle-mode device:

- Timing0.ADDRESS_SETUP = 5

- Timing0.DATA_SETUP = 10
- Timing0.DATA_HOLD = 10
- Timing1.DEVICE_BUSY_TIMEOUT = 0 x 500
- Timing2.READ_LATENCY = BOOT_CFG2[7:5] if blown, otherwise 2
- Timing2.CE_DELAY = 2
- Timing2.PREAMBLE_DELAY = BOOT_CFG2[7:5] if blown, otherwise 2
- Timing2.POSTAMBLE_DELAY = 3
- Timing2.CMDADD_PAUSE = 4
- Timing2.DATA_PAUSE = 6

Default timing parameters can be overriden by TMTiming2_ReadLatency, TMTiming2_PreambleDelay, TMTiming2_CEDelay, TMTiming2_PostambleDelay, TMTiming2_CmdAddPause, TMTiming2_DataPause parameters of FCB.

8.5.2.6.2 Setup DMA for DDR Transfers

In DMA descriptors GPMI is configured to read page data at double data rate, the word length is set to 16 and transfer count to half of page size.

8.5.2.6.3 Reconfigure Timing and Speed Using Values in FCB

After reading FCB page with GPMI set to default timings and speed 33 MHz, ROM reconfigures CCM dividers to run gpmi/bch clks to desired speed specified in FCB for rest of boot process. The GPMI timing registers are also reconfigured to values specified in FCB.

The GPMI speed can be configured using FCB parameter TMSpeed: 0 - 24MHZ, 1 - 33MHZ, 2 - 40MHZ, 3 - 50MHZ, 4 - 66MHZ, 5 - 80MHZ, 6 - 100MHZ, 7 - 133MHZ, 8 - 160MHZ, 9 - 200MHZ.

The GPMI timing0 register fields data_setup, data_hold and address_setup are set to values specified for data_setup, data_hold and address_setup in FCB member m_NANDTiming.

The GPMI timing1.DEVICE_BUSY_TIMEOUT is set to value specified in FCB member TMTiming1_BusyTimeout.

The GPMI timing2 register values are set using FCB members TMTiming2.READ_LATENCY, CE_DELAY, PREAMBLE_DELAY, POSTAMBLE_DELAY, CMDADD_PAUSE and DATA_PAUSE.

8.5.2.7 Typical NAND Page Organization

8.5.2.7.1 BCH ECC Page Organization

The first data block is called block 0 and the rest of the blocks are called block N. Separate ECC level scan be used for block 0 and block N.

The metadata bytes should be located at the beginning of a page, starting at byte 0, followed by data block 0, followed by ECC bytes for data block 0, followed by block 1 and its ECC bytes, and so on until N data blocks. The ECC level for block 0 can be different from the ECC level of rest of the blocks.

For NAND boot, with page size restrictions and data block size restricted to 512 bytes, only few combinations of ECC for block 0 and block N are possible.

The figure below shows the valid layout for 2112 byte sized page.

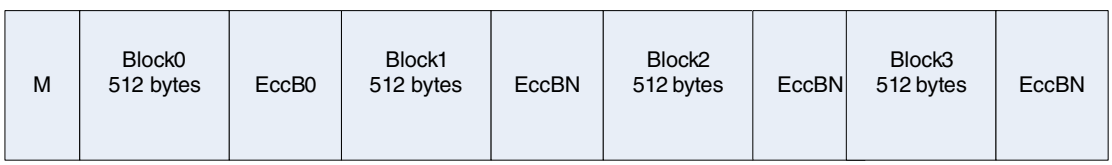


Figure 8-8. Valid Layout for 2112 bytes Sized Page

The example below is for 13 bits of parity(GF13). The number of ECC bits required for a data block is calculated using (ECC_Correction_Level * 13) bits.

In the above layout the ECC size for EccB0 and EccBN should be selected to not exceed a total page size of 2112 bytes. EccB0 and EccBN can be one of 2, 4, 6, 8, 10, 12, 14, 16, 18, 20 bits ECC correction level. The total bytes would then be:

[M + (data_block_size x 4) + ([EccB0 + (EccBN x 3)] x 13) / 8] <= 2112;

M = metadata bytes and data_block_size is 512.

There are 4 data blocks of 512 bytes each in a page of 2k page sized NAND. The values of EccB0 and EccBN should be such that the above calculation would not result in a value greater than 2112 bytes.

M	Block0 512 bytes	EccB0	Block1 512 bytes	EccBN	Block2 512 bytes	EccBN	Block3 512 bytes	EccBN
	Block4 512 bytes	EccBN	Block5 512 bytes	EccBN	Block6 512 bytes	EccBN	Block7 512 bytes	EccBN

Figure 8-9. Valid Layout for 4 Kbytes Sized Page

Different NAND manufacturers have different sizes for a 4K page; 4314 bytes is typical.

[M + (data_block_size x 8) + ([EccB0 + (EccBN x 7)] x 13) / 8] <= 4314;

M= metadata bytes and data_block_size is 512.

There are 8 data blocks of 512 bytes each in a page of a 4k page sized NAND. The values of EccB0 and EccBN should be such that above calculation should not result in a value greater than the size of a page in a 4k page NAND.

8.5.2.7.2 Metadata

The number of bytes used for metadata is specified in FCB. Metadata for BCH encoded pages will be placed at the beginning of a page. ROM only cares about the first byte of metadata to swap it with bad block marker byte in page data after each page read; it is important to have at least one byte for the metadata bytes field in FCB data structure.

8.5.2.8 IOMUX Configuration for NAND

The table below shows the RawNAND IOMUX pin configuration. NAND boot is only supported on chip select 0.

Table 8-14. NAND IOMUX Pin Configuration

Signal	Pad Name
CLE	NANDF_CLE.alt0
ALE	NANDF_ALE.alt0
WPN	NANDF_WP_B.alt0
RD_N	SD4_CMD.alt1
WRN	SD4_CLK.alt1
READY0	NANDF_RB0.alt0
DQS	SD4_DAT0.alt2
CE0N	NANDF_CS0.alt0

Table continues on the next page...

Table 8-14. NAND IOMUX Pin Configuration (continued)

CE1N	NANDF_CS1.alt0
CE2N	NANDF_CS2.alt0
CE3N	NANDF_CS3.alt0
D0	NANDF_D0.alt0
D1	NANDF_D1.alt0
D2	NANDF_D2.alt0
D3	NANDF_D3.alt0
D4	NANDF_D4.alt0
D5	NANDF_D5.alt0
D6	NANDF_D6.alt0
D7	NANDF_D7.alt0

8.5.3 Expansion Device

The ROM supports booting from MMC/eMMC and SD/eSD compliant devices.

8.5.3.1 Expansion Device eFUSE Configuration

SD/MMC/eSD/eMMC/SDXC boot can be performed using either USDHC-1, USDHC-2, USDHC-3, or USDHC-4 ports, based on setting of the BOOT_CFG2[4:3] (Port Select) fuse or it's associated GPIO input value at boot. All USDHC ports support eMMC4.3 and eMMC4.4 fast boot.

See the table below for details.

Table 8-15. USDHC Boot eFUSE Descriptions

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG1[7:6]	OEM	Boot Device Selection	Yes	00	01 - Boot from USDHC Interfaces
BOOT_CFG1[5]	OEM	SD/MMC Selection	Yes	0	0 - SD/eSD/SDXC 1 - MMC/eMMC
BOOT_CFG1[4]	OEM	Fast Boot Support	Yes	0	0 - Normal Boot 1 - Fast Boot
BOOT_CFG1[3:2]	OEM	SD/MMC Speed Mode	Yes	00	MMC 0x - High Speed Mode 1x - Normal Speed Mode x0 - eMMC Fast boot acknowledge enable

Table continues on the next page...

Table 8-15. USDHC Boot eFUSE Descriptions (continued)

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
					x1 - eMMC Fast boot acknowledge disable SD 0x - High/Normal 10 - SDR50 11 - SDR104
BOOT_CFG1[1]	OEM	SD Power Cycle Enable/ eMMC Reset Enable	Yes	0	MMC 0 - eMMC reset disabled 1 - eMMC reset enabled via SD_RST pad (on USDHC3 and USDHC4 only) SD 0 - No power cycle 1 - Power cycle enabled via SD_RST pad (on USDHC3 and USDHC4 only)
BOOT_CFG1[0]	OEM	SD Loopback Clock Source Sel(for SDR50 and SDR104 only)	Yes	00	0 - through SD pad 1 - direct
BOOT_CFG2[7:5]	OEM	Bus Width/SD Calibration Step	Yes	000	SD/eSD/SDXC (BOOT_CFG1[5]=0) Bus Width xx0 - 1-bit xx1 - 4-bit SD Calibration Step 00x - 1 delay cells 01x - 1 delay cells 10x - 2 delay cells 11x - 3 delay cells MMC/eMMC (BOOT_CFG1[5]=1) 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - reserved.
BOOT_CFG2[4:3]	OEM	Port Select	Yes	00	00 - USDHC-1 01 - USDHC-2 10 - USDHC-3 11 - USDHC-4
BOOT_CFG2[2]	OEM	Boot Frequencies	Yes	0	0 - Boot ROM default.

Table continues on the next page...

Table 8-15. USDHC Boot eFUSE Descriptions (continued)

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
					1 - Apply value per fuse field MMC_DLL_DLY[6:0]
BOOT_CFG2[1]	OEM	Boot Acknowledge Disable/ Pull Down During Power Cycle Enable	Yes	0	MMC 0 - Boot Acknowledge Enabled. 1 - Boot Acknowledge Disabled. SD 0 - Use default SD pad settings during power cycle 1 - Set pull-down on SD pads during power cycle (used only if "SD Power Cycle Enable" enabled)
BOOT_CFG2[0]	OEM	Override Pad Settings	Yes	0	0 - Use default values 1 - Use PAD_SETTINGS values
MMC_DLL_DLY[6:0]	OEM	MMC DLL Value / UHSI Calibaration Start Value	No	0000000	MMC DLL Value / UHSI Calibaration Start Value

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.

Boot code supports following standards.

- MMCv4.4 or less
- eMMCv4.4 or less
- SDv2.0 or less
- eSDv2.10 rev-0.9, with or without FAST_BOOT.
- SDXCv3.0

MMC/SD/eSD/SDXC/eMMC can be connected to any of USDHC-1,2,3,4 blocks and can be booted by copying 4Kbyte of data from MMC/SD/eSD/eMMC device to internal RAM. After checking the Image Vector Table header value (0xD1) from Program Image, the ROM code performs a DCD check. After successful DCD extraction, the ROM code extracts from Boot Data Structure the destination pointer and length of image to be copied to RAM device from where code execution occurs.

The maximum image size to load in SD/MMC boot is 32MB. This is due to the limited number of uSDHC ADMA Buffer Descriptors allocated by ROM.

NOTE

The Initial 4Kbyte of Program Image must contain the IVT, DCD and the Boot Data structures.

Table 8-16. SD/MMC Frequencies

	SD	MMC	MMC (DDR Mode)
Identification (KHz)	347.22		
Normal Speed Mode (MHz)	25	20	25
High Speed Mode (MHz)	50	40	50
UHSI SDR50 (MHz)	100		
UHSI SDR104 (MHz)	200		

NOTE

The boot ROM code reads application image length and application destination pointer from image.

8.5.3.2 MMC and eMMC Boot

The following table provides MMC and eMMC boot details.

Table 8-17. MMC and eMMC Boot Details

Normal Boot Mode	During initialization (normal boot mode) the MMC frequency is set to 347.22 KHz. When the MMC card enters the identification portion of the initialization, voltage validation is performed and the ROM boot code checks high voltage settings and card capacity. The ROM boot code supports both high capacity and low capacity MMC/eMMC cards. After initialization phase is complete, the ROM boot code switches to a higher frequency (20 MHz in Normal boot mode or 40MHz in High Speed mode). eMMC is also interfaced via USDHC and follows the same flow as MMC. The boot partition can be selected for an MMC4.x card after the card initialization is complete. The ROM code reads the BOOT_PARTITION_ENABLE field in the Ext_CSD[179] to get the boot partition to be set. If there is no boot partition mentioned in BOOT_PARTITION_ENABLE field or the user partition has been mentioned, ROM boots from the user partition.
eMMC4.3 or eMMC4.4 Device Supporting Special Boot Mode	If using an eMMC4.3 or eMMC4.4 device supporting special boot mode, it can be initiated by pulling the CMD line low. If BOOT ACK is enabled, the eMMC4.3/eMMC4.4 device sends the BOOT ACK via DATA lines and ROM can read the BOOT ACK [S010E] to identify the eMMC4.3/eMMC4.4 device. eMMC4.3/eMMC4.4 device with "Boot mode" feature can only be supported via ESDHCV3-3 and with or without BOOT ACK. If BOOT ACK is enabled ROM waits 50 ms to get the BOOT ACK and if BOOT ACK is received by ROM. If BOOT ACK is disabled ROM waits 1 second for data. If BOOT ACK or data was received then eMMC4.3/eMMC4.4 is booted in "Boot mode", otherwise eMMC4.3/eMMC4.4 boots as a

Table continues on the next page...

Table 8-17. MMC and eMMC Boot Details (continued)

	normal MMC card from the selected boot partition. This boot mode can be selected by BOOT_CFG1[4] (Fast Boot) fuse. BOOT ACK is selected by BOOT_CFG2[1].
eMMC4.4 Device	If using eMMC4.4 device, Double Data Rate (DDR) mode can be used. This mode can be selected by BOOT_CFG2[7:5] (Bus Width) fuse.

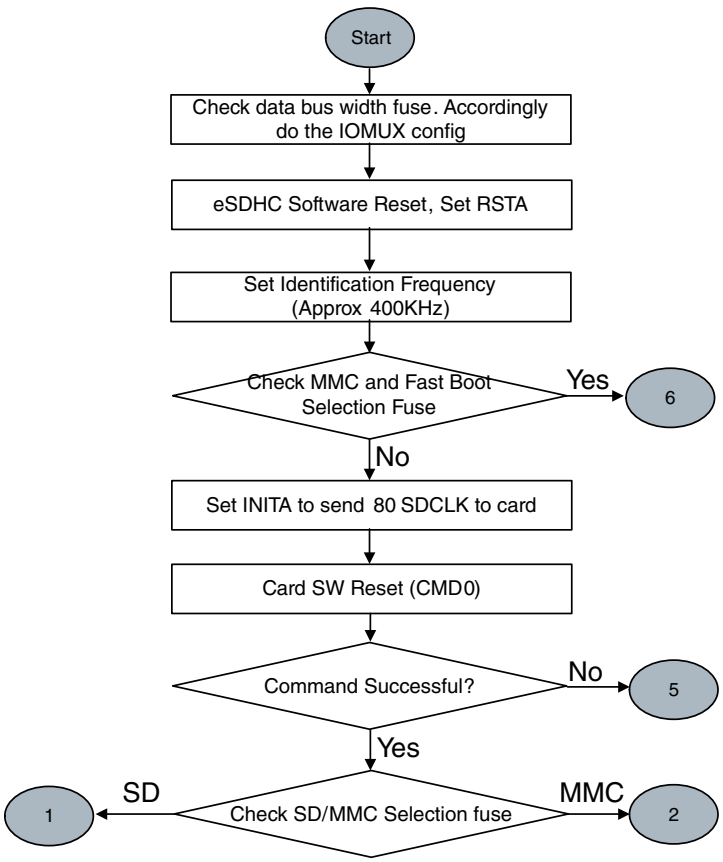


Figure 8-10. Expansion Device Boot Flow (1 of 6)

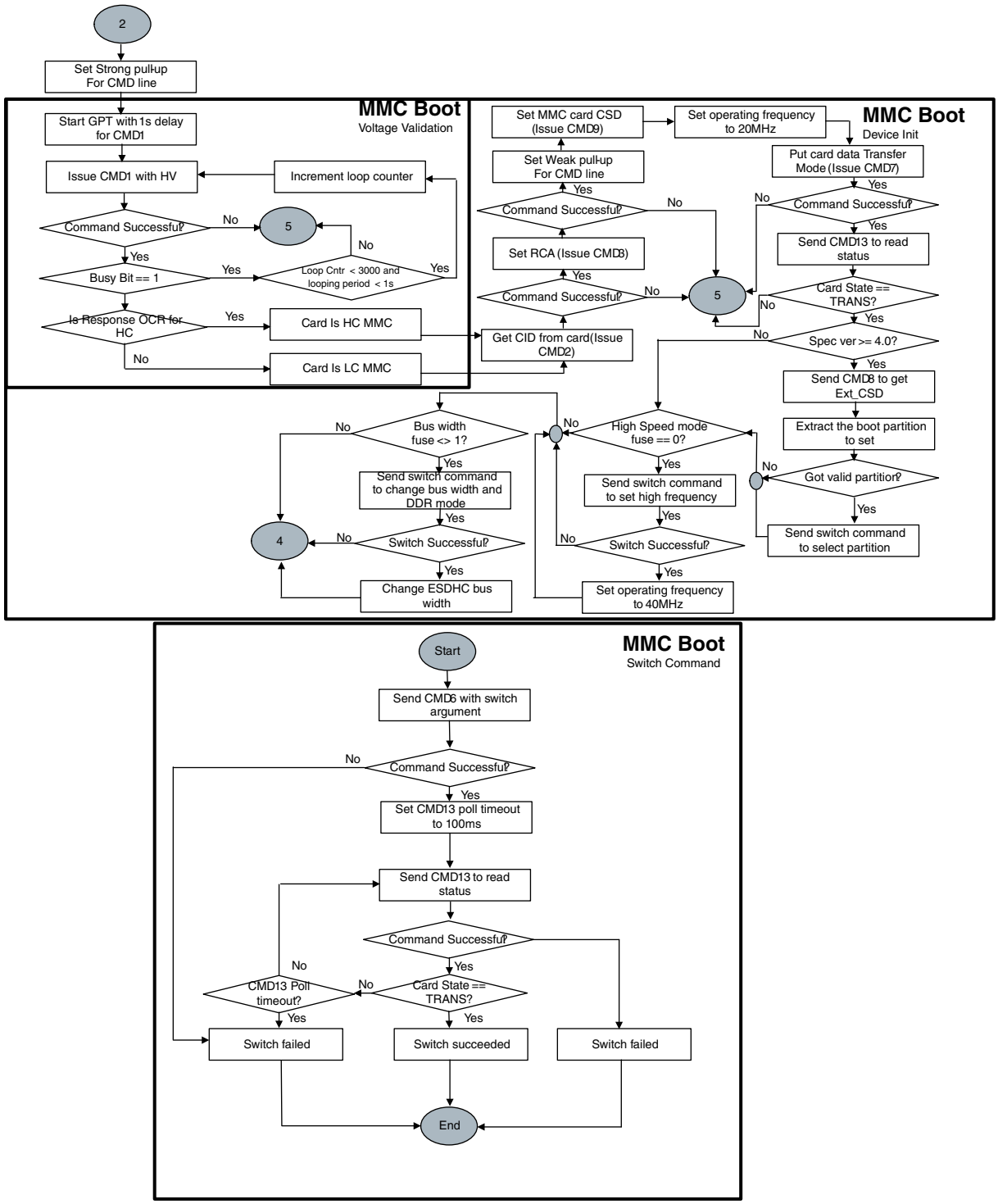


Figure 8-11. Expansion Device (MMC) Boot Flow (2 of 6)

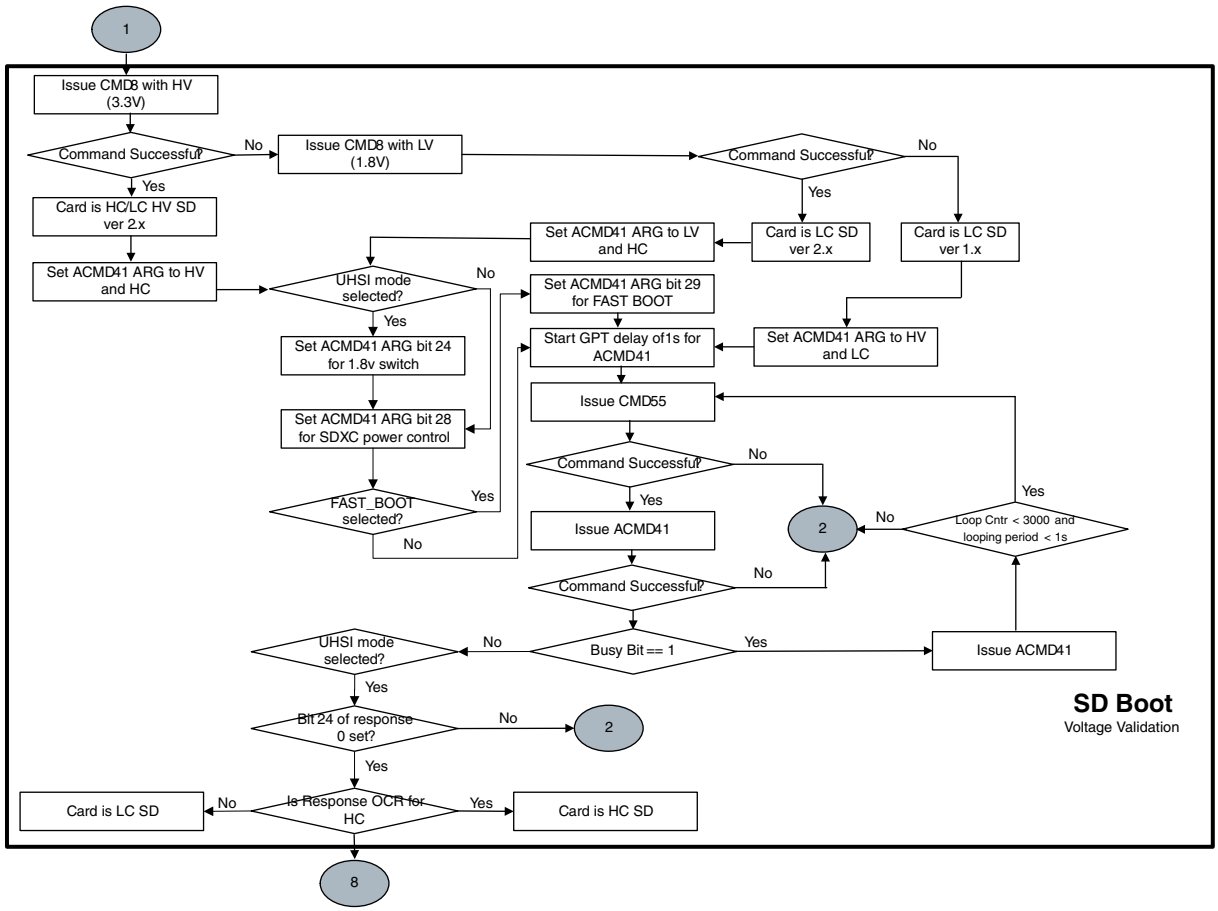


Figure 8-12. Expansion Device (SD/eSD/SDXC) Boot Flow (3 of 6) Part 1

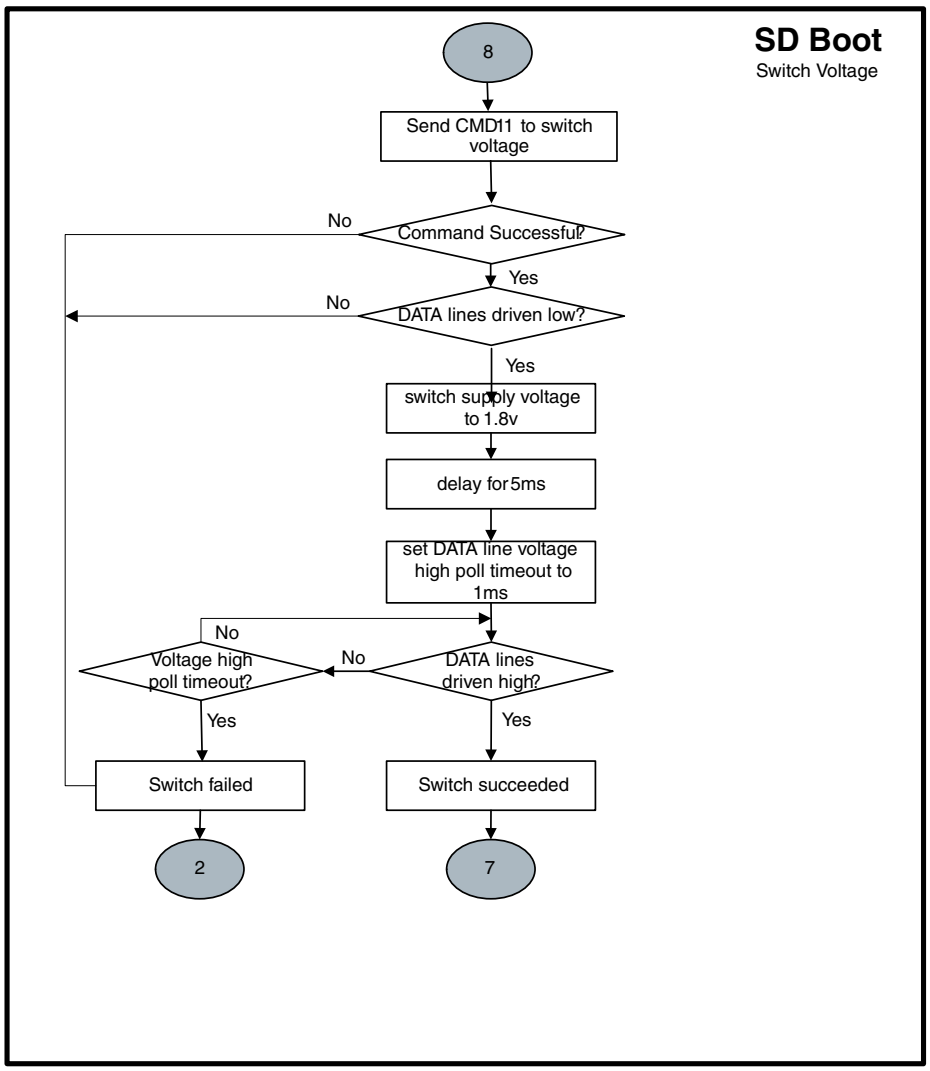
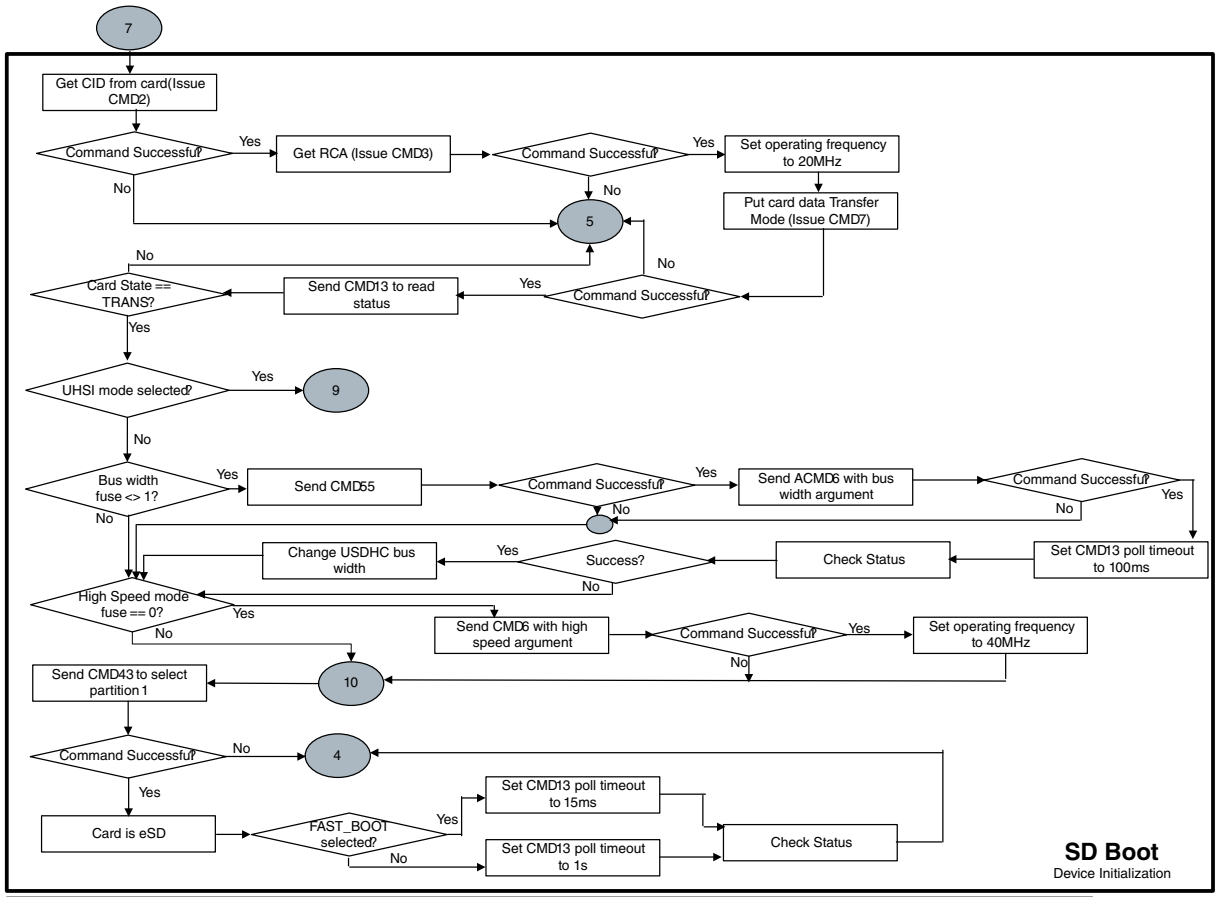


Figure 8-13. Expansion Device (SD/eSD/SDXC) Boot Flow (3 of 6) Part 2



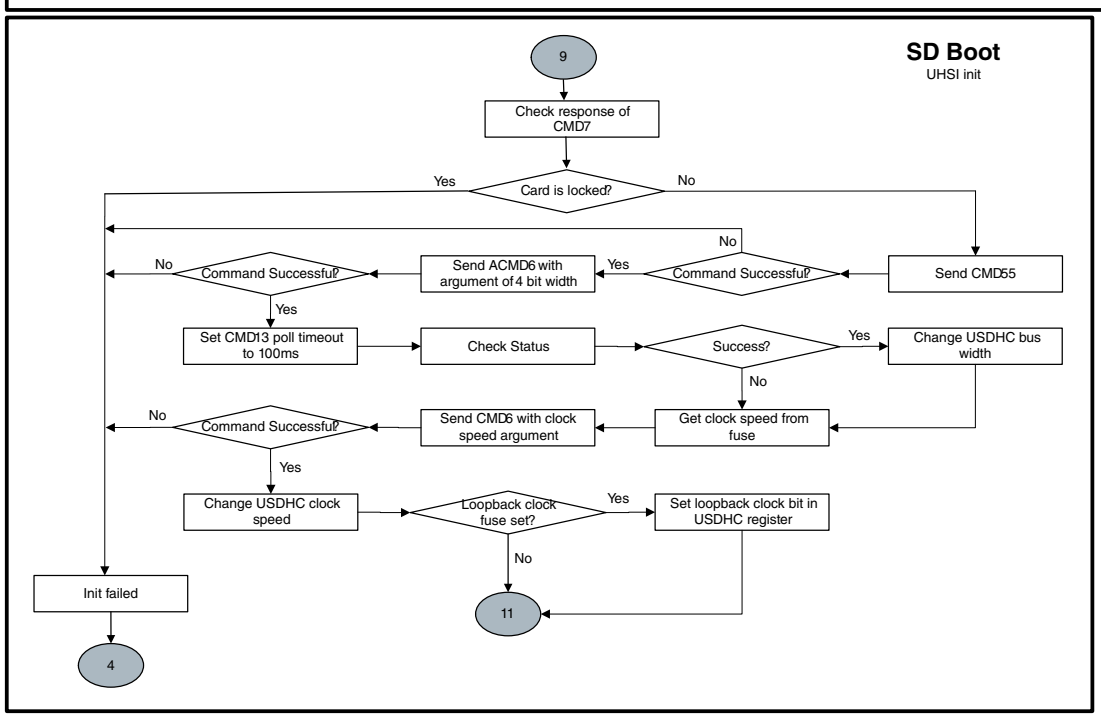


Figure 8-14. Expansion Device (MMCSD/eSD/SDXC) Boot Flow (4 of 6)

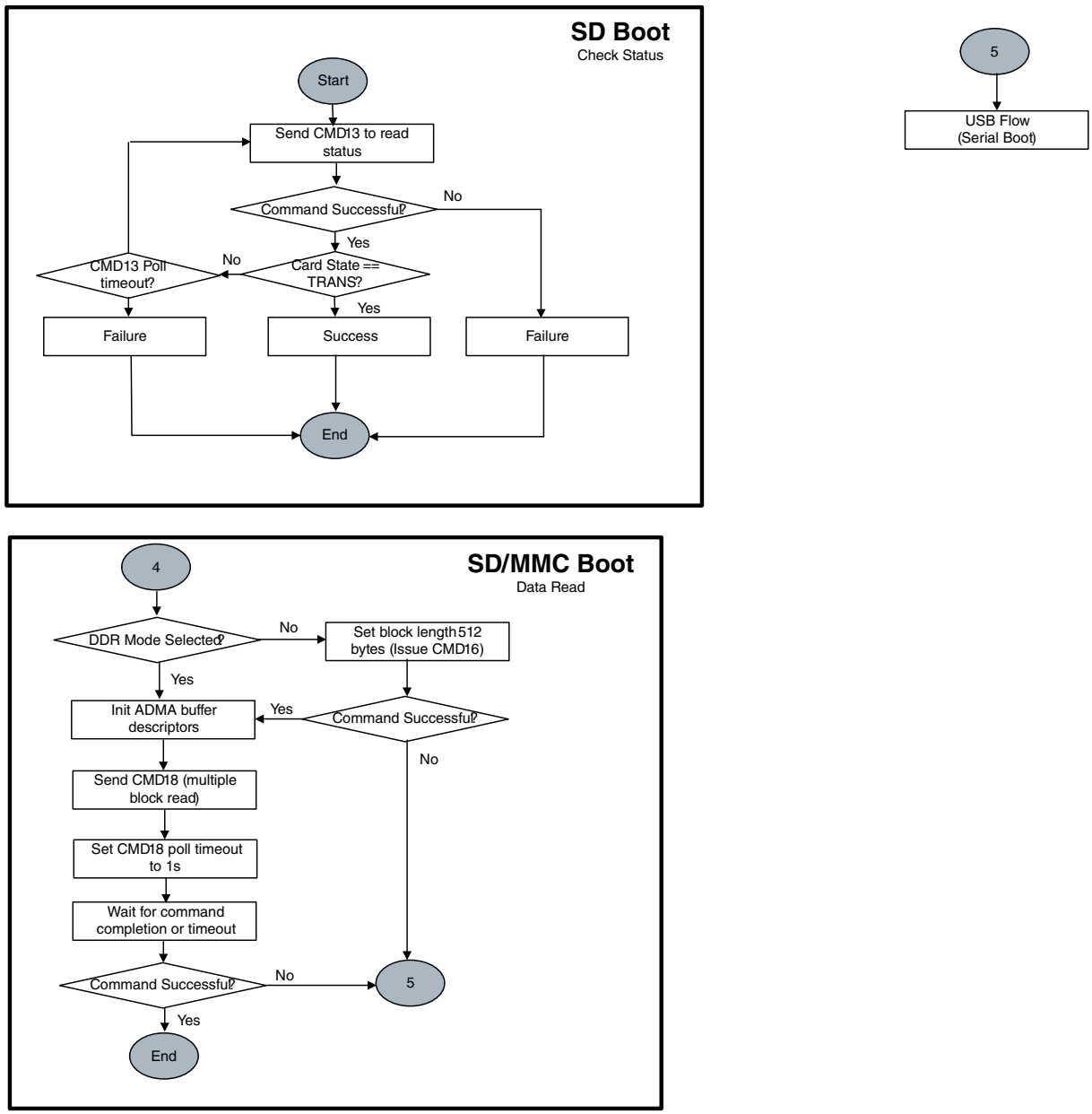
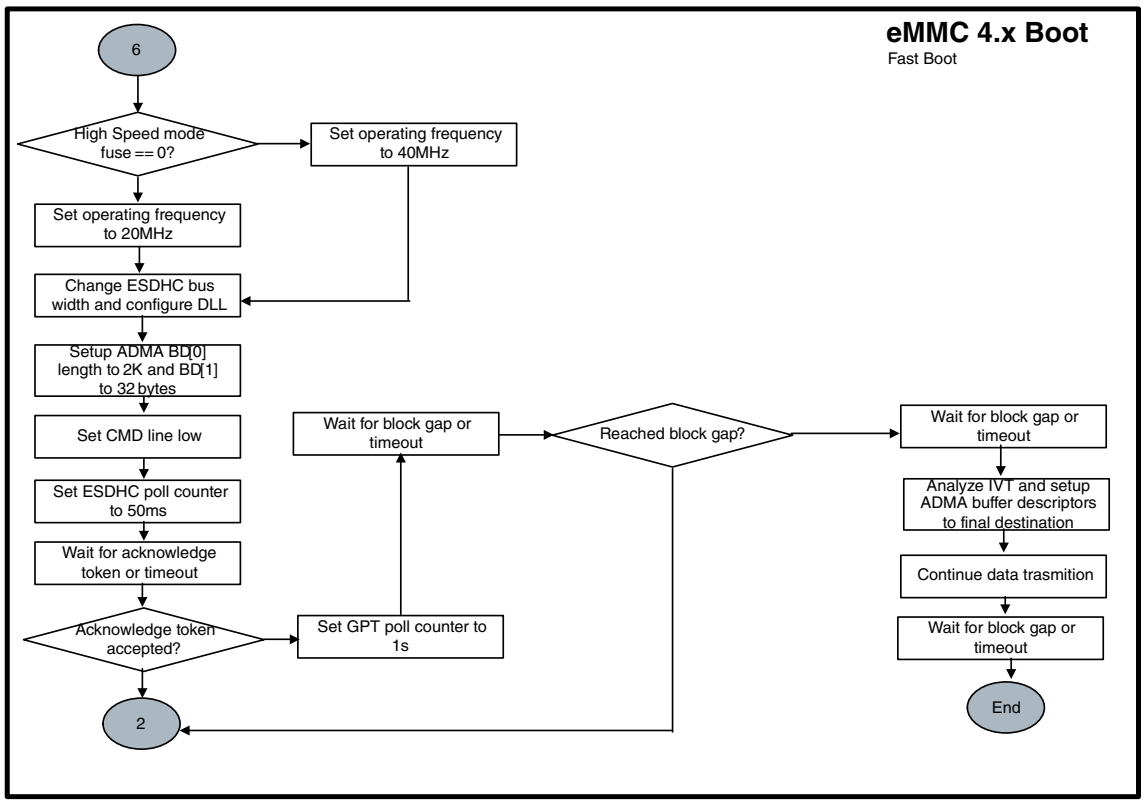


Figure 8-15. Expansion Device (SD/eSD) Boot Flow (5 of 6)



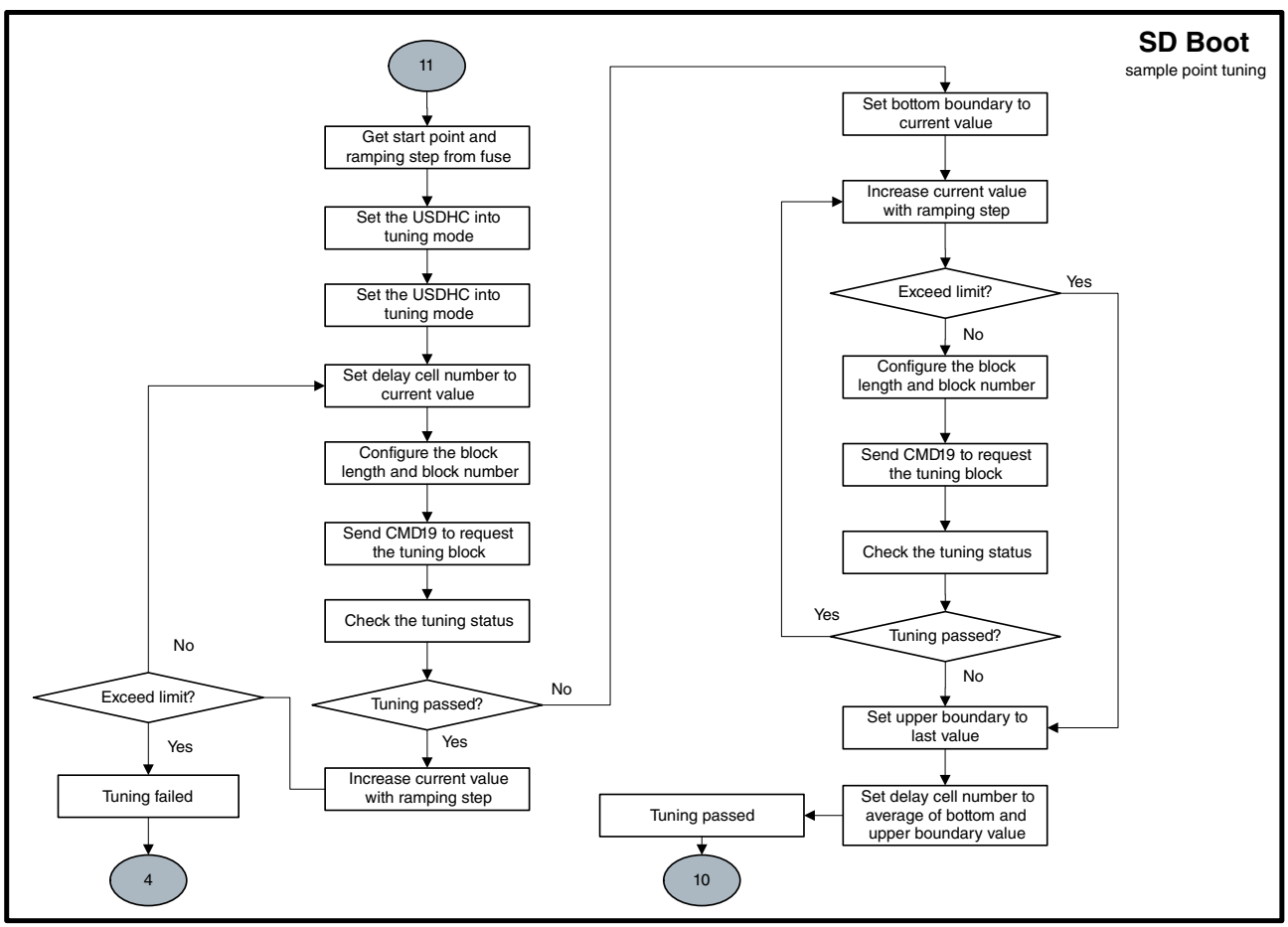


Figure 8-16. Expansion Device Boot Flow (6 of 6)

8.5.3.3 SD, eSD and SDXC

After the normal boot mode initialization begins, the SD/eSD/SDXC frequency is set to 347.22 kHz. During the identification phase, SD/eSD/SDXC card voltage validation is performed. During voltage validation, boot code first checks with high voltage settings; if that fails, it checks with low voltage settings.

The capacity of the card is also checked. Boot code supports high capacity and low capacity SD/eSD/SDXC cards after voltage validation card initialization is done.

During card initialization, the ROM boot code attempts to set the boot partition for all SD, eSD, and SDXC devices. If this fails, the boot code assumes the card is a normal SD card or SDXC card. If it does not fail, the boot code assumes it is an eSD card. After the initialization phase is over, boot code switches to a higher frequency (25 MHz in Normal Speed mode or 50 MHz in High Speed Mode). ROM also supports FAST_BOOT mode booting from eSD card. This mode can be selected by BOOT_CFG1[4] (Fast Boot) fuse described in Table 8-15.

For UHSI cards, clock speed fuses can be set to SDR50 or SDR104 on USDHC3 and USDHC4 ports. This will enable the voltage switch process to set the signaling voltage to 1.8V during voltage validation. The bus width is fixed at 4 bits wide and a sampling point tuning process is needed to calibrate the number of delay cells. If SD Loopback Clock eFuse is set, the feedback clock will come directly from the loopback SD clock, instead of the card clock (by default). The SD clock speed can be selected by BOOT_CFG1[3:2], and the SD Loopback Clock is selected by BOOT_CFG1[0].

UHSI calibration start value (MMC_DLL_DLY[6:0]) and step value (BOOT_CFG2[7:5]) can be set to optimize the sample point tuning process.

If SD Power Cycle Enable eFuse is 1, ROM will set SD_RST pad low, wait 5ms and then set SD_RST pad high. If SD_RST pad is connected to SD power supply enable logic on board, it enables power cycle of SD card. This may be crucial in case when SD logic is in 1.8V states and must be reset to 3.3V states.

8.5.3.4 IOMUX Configuration for SD/MMC

Table 8-18. SD/MMC IOMUX Pin Configuration

Signal	USDHC-1	USDHC-2	USDHC-3	USDHC-4
CLK	SD1_CLK.alt0	SD2_CLK.alt0	SD3_CLK.alt0	SD4_CLK.alt0
CMD	SD1_CMD.alt0	SD2_CMD.alt0	SD3_CMD.alt0	SD4_CMD.alt0
DAT0	SD1_DAT0.alt0	SD2_DAT0.alt0	SD3_DAT0.alt0	SD4_DAT0.alt1

Table continues on the next page...

Table 8-18. SD/MMC IOMUX Pin Configuration (continued)

Signal	USDHC-1	USDHC-2	USDHC-3	USDHC-4
DAT1	SD1_DAT1.alt0	SD2_DAT1.alt0	SD3_DAT1.alt0	SD4_DAT1.alt1
DAT2	SD1_DAT2.alt0	SD2_DAT2.alt0	SD3_DAT2.alt0	SD4_DAT2.alt1
DAT3	SD1_DAT3.alt0	SD2_DAT3.alt0	SD3_DAT3.alt0	SD4_DAT3.alt1
DAT4	NANDF_D0.alt1	NANDF_D4.alt1	SD3_DAT4.alt0	SD4_DAT4.alt1
DAT5	NANDF_D1.alt1	NANDF_D5.alt1	SD3_DAT5.alt0	SD4_DAT5.alt1
DAT6	NANDF_D2.alt1	NANDF_D6.alt1	SD3_DAT6.alt0	SD4_DAT6.alt1
DAT7	NANDF_D3.alt1	NANDF_D7.alt1	SD3_DAT7.alt0	SD4_DAT7.alt1
VSELECT			GPIO_18.alt2	NANDF_CS1.alt1
RESET [1]			SD3_RESET.alt0	NANDF_ALE.alt1
CD	GPIO_1.alt6	GPIO_4.alt6	-	-

1. Active low

8.5.3.5 Redundant Boot Support for Expansion Device

ROM supports redundant boot for expansion device. Primary or Secondary image is selected depending on PERSIST_SECONDARY_BOOT setting (see Table 8-7).

If PERSIST_SECONDARY_BOOT is 0, the boot ROM uses address 0x0 for primary image.

If PERSIST_SECONDARY_BOOT is 1, the boot ROM will read secondary image table from address 0x200 on boot media and will use address specified in the table.

Table 8-19. Secondary Image Table Format

Reserved (chipNum)
Reserved (driveType)
tag
firstSectorNumber
Reserved (sectorCount)

Where:

- tag: used as indication of valid secondary image table. Must be 0x00112233.
- firstSectorNumber is the first 512B sector number of the secondary image.

For secondary image support, the primary image must reserve space for secondary image table. See the figure below for typical structures layout on expansion device.

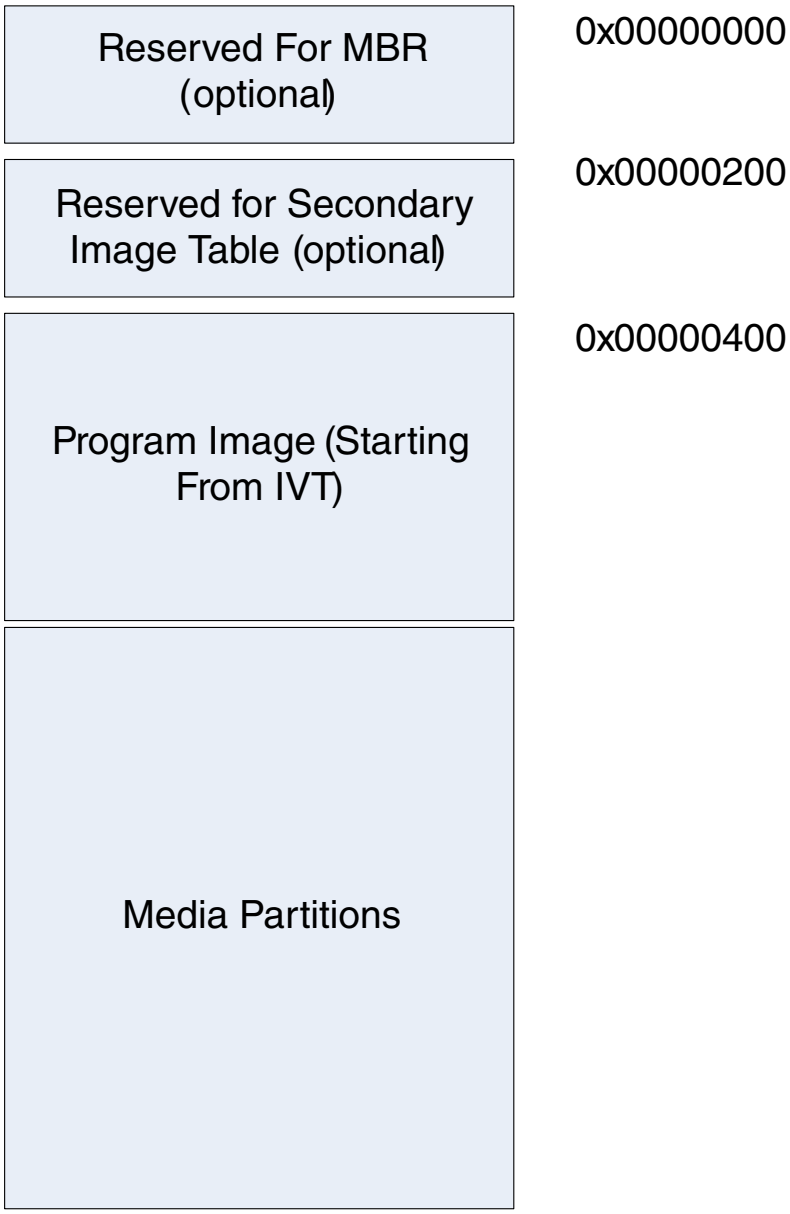


Figure 8-17. Expansion Device Structures Layout

For Closed mode, if there are failures during primary image authentication, the boot ROM will turn on PERSIST_SECONDARY_BOOT bit (see Table 8-7) and perform software reset. (After software reset, secondary image will be used.)

8.5.4 Hard Disk and SSD

The chip supports boot from Hard Disk and SSD devices using SATA interface.

8.5.4.1 Hard Disk and SSD eFUSE Configuration

The boot ROM code determines the type of device using the following parameters, either provided by eFUSE settings or sampled on the I/O pins, during boot.

Table 8-20. HDD eFUSE Descriptions

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG1[7:4]	OEM	Boot Device Selection	Yes	0000	0010 - Boot from Hard Disk
BOOT_CFG2[4]	OEM	Tx Spread Spectrum	Yes	0	0 - Disabled 1 - Enabled
BOOT_CFG2[3]	OEM	Rx Spread Spectrum	Yes	0	0 - Enabled 1 - Disabled
BOOT_CFG2[2]	OEM	SATA Speed	Yes	0	0 - Gen2 (3.0Gbps) 1 - Gen1 (1.5Gbps)
BOOT_CFG2[1:0]	OEM	SATA Type	Yes	00	00 - i (internal electrical specifications with cable length up to 1m, see Serial ATA specification) 01 - m (electrical specifications used in Short Backplane Application and External Desktop Application, see Serial ATA specification) 10 - x (external cabled applications or Long Backplane Applications, see Serial ATA specification) 11 - Reserved

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.

The boot ROM will send IDENTIFY command to hard disk during initialization. When identification block is received, boot ROM assumes that the device is ready. The boot ROM sends a separate command for each sector of 512 bytes in PIO mode.

The boot ROM will copy 4KB of data from Hard Disk or SSD device to internal RAM. After checking the Image Vector Table header value (0xD1) from Program Image, the ROM code performs a DCD check. After successful DCD extraction, the ROM code extracts the destination pointer and length of image to be copied to RAM device from the Boot Data Structure, where code execution occurs.

NOTE

The Initial 4 KB of Program Image must contain the IVT, DCD and the Boot Data structures.

8.5.4.2 IOMUX and Timing Configuration for SATA

The interface signals of the SATA PHY are not configured in the IOMUX. The SATA PHY interface uses dedicated contacts on the IC. See the Chip data sheet for details.

ROM reads the TX Spread Spectrum, RX Spread Spectrum, Speed and Type of SATA and configures timing parameters via the IOMUX GPR register.

8.5.4.3 Redundant Boot Support for Hard Disk and SSD

ROM supports redundant boot for hard disk and SSD. Primary or Secondary image is selected depending on PERSIST_SECONDARY_BOOT setting (see Table 8-7).

If PERSIST_SECONDARY_BOOT is 0, the boot ROM uses address 0x0 for primary image.

If PERSIST_SECONDARY_BOOT is 1, the boot ROM will read secondary image table from address 0x200 on boot media and will use address specified in the table.

Table 8-21. Secondary Image Table Format

Reserved (chipNum)
Reserved (driveType)
tag
firstSectorNumber
Reserved (sectorCount)

Where:

- tag: used as indication of valid secondary image table. Must be 0x00112233.
- firstSectorNumber is the first 512B sector number of the secondary image.

For secondary image support, the primary image must reserve space for secondary image table. See the figure below for typical structures layout on expansion device.

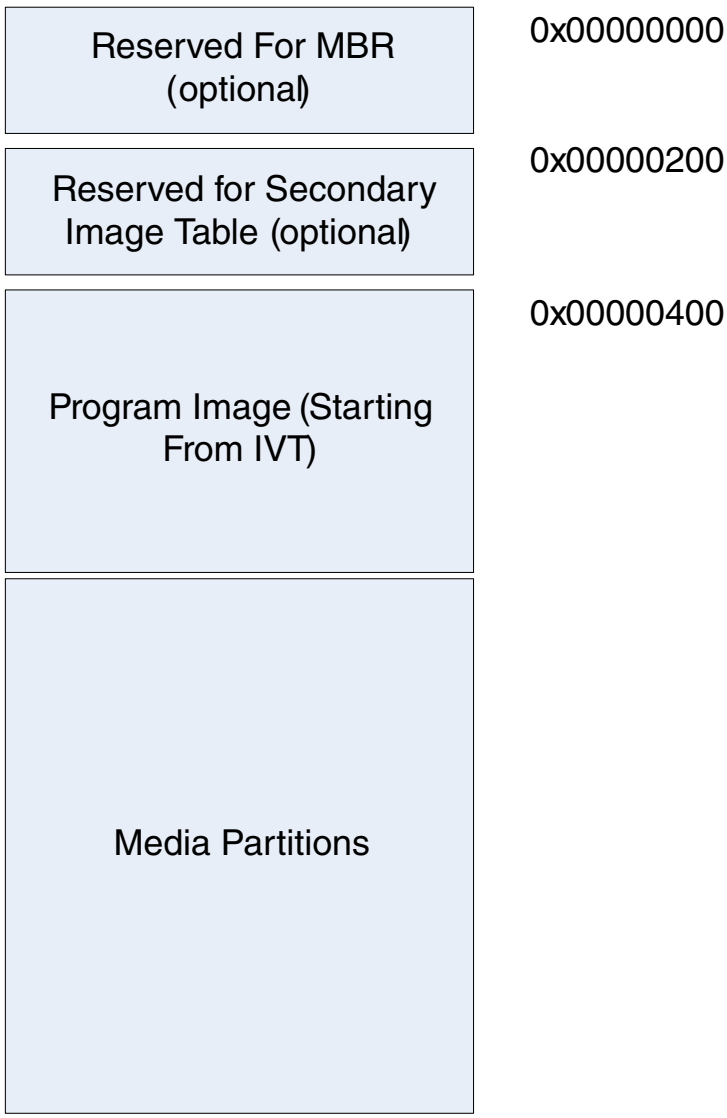


Figure 8-18. Hard Disk Structures Layout

For Closed mode, if there are failures during primary image authentication, the boot ROM will turn on PERSIST_SECONDARY_BOOT bit (see Table 8-7) and perform software reset. (After software reset secondary image will be used).

8.5.5 Serial ROM through SPI and I2C

The chip supports boot from serial memory devices, such as EEPROM and Serial Flash using the SPI.

The following ports are available for serial boot: ECSPI (ECSPI-1, ECSPI-2, ECSPI-3, ECSPI-4, ECSPI-5), and I2C Controller (I2C-1, I2C-2 and I2C-3) interfaces.

8.5.5.1 Serial ROM eFUSE Configuration

The boot ROM code determines the type of device using the following parameters, either provided by eFUSE settings or sampled on the I/O pins, during boot.

See the table below for details:

Table 8-22. Serial ROM Boot eFUSE Descriptions

Fuse	Config	Definition	GPIO[1]	Shipped Value	Settings
BOOT_CFG1[7:4]	OEM	Boot Device Selection	Yes	0000	0011 - Boot from Serial ROM
BOOT_CFG4[6]	OEM	EEPROM Recovery Enable	Yes	0	0 - Disabled EEPROM recovery 1 - Enabled EEPROM recovery
BOOT_CFG4[5:4]	OEM	CS select (SPI only)	Yes	00	00 - ECSPIx_SS0 01 - ECSPIx_SS1 10 - ECSPIx_SS2 11 - ECSPIx_SS3
BOOT_CFG4[3]	OEM	SPI Addressing (SPI only)	Yes	0	0 - 2-bytes (16-bit) 1 - 3-bytes (24-bit)
BOOT_CFG4[2:0]	OEM	Port Select	Yes	00	000 - ECSPI-1 001 - ECSPI-2 010 - ECSPI-3 011 - ECSPI-4 100 - ECSPI-5 101- I2C-1 110- I2C-2 111- I2C-3

1. Setting can be overridden by GPIO settings when BT_FUSE_SEL fuse is intact. See Table 1 for corresponding GPIO pin.

The ECPSI-1/ECPSI-2/ECPSI-3/ECPSI-4/ECSPI-5 block can be used as boot device using ECSPI interface for serial ROM boot. The SPI interface is configured to operate at 15MHz for 3-byte addressing device and 3.75MHz for 2-byte addressing devices.

The I2C-1/I2C-2/I2C-3 block can be used as boot device using I2C interface, for serial ROM boot. The I2C interface is configured to operate at 343.75 Kbps.

The boot ROM will copy 4Kbyte of data from Serial ROM device to internal RAM. After checking the Image Vector Table header value (0xD1) from Program Image, the ROM code performs a DCD check. After successful DCD extraction, the ROM code extracts from Boot Data Structure the destination pointer and length of image to be copied to RAM device from where code execution occurs.

NOTE

The Initial 4K of Program Image must contain the IVT, DCD and the Boot Data structures.

8.5.5.2 I2C Boot

The boot flow when booting from an I2C device is shown in Figure 8-19.

The boot ROM code reads the fuses BOOT_CFG1[7:4] (Boot Device Selection) and BOOT_CFG1[7:4] (Port select) to detect EEPROM device type. The ROM program copies 4K data from the EEPROM device to internal RAM. The boot ROM code next copies the initial 4Kbyte of data as well as rest of image directly to application destination extracted from application image.

The chip uses the Device Select Code/Device Address in the table below to boot from an EEPROM.

Table 8-23. EEPROM via I2C Device Select Code

Bits	Device Type Identifier				Chip Enable Address[1]			R/W
	7	6	5	4	3	2	1	0
Device Select Code	1	0	1	0	0	0	0	R/W

1. These address bits, should be configured at the memory device, to match this '000' value.

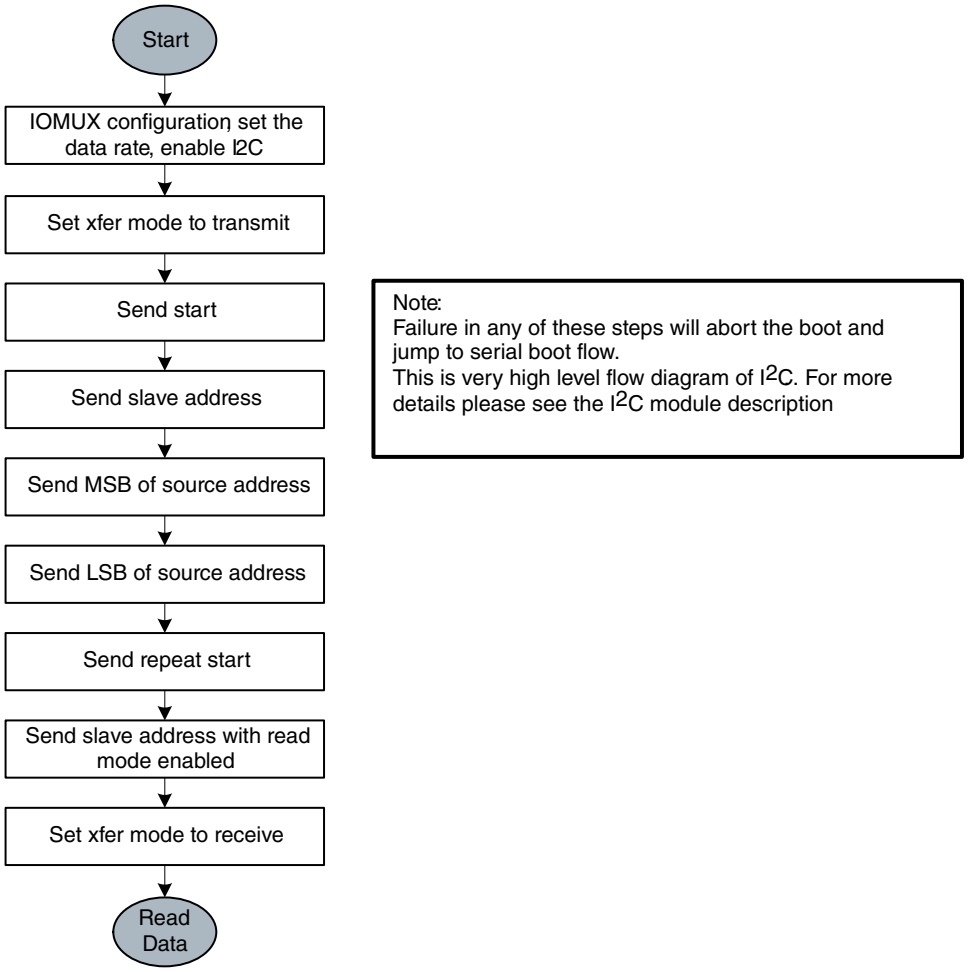


Figure 8-19. I2C Flow Chart

8.5.5.2.1 I2C IOMUX Pin Configuration

The contacts assigned to the signals used by the I2C blocks is shown in the table below.

Table 8-24. I2C IOMUX Pin Configuration

Signal	I2C-1	I2C-2	I2c-3
SDA	EIM_D28.alt1	EIM_D16.alt6	EIM_D18.alt6
SCL	EIM_D21.alt6	EIM_EB2.alt6	EIM_D17.alt6

8.5.5.3 ECSPI Boot

The Enhanced Configurable SPI (ECSPI) interface is configured in master mode and the EEPROM device is connected to ECSPI interface as a slave.

The boot ROM code copies 4 KB data from EEPROM device to the internal RAM. If DCD verification is successful, the ROM code copies the initial 4 KB data, as well as the rest of the image extracted from application image, directly to the application destination. The ECSPI can read data from EEPROM using 2 or 3 byte addressing. Its burst length is 32 bytes.

NOTE

The Serial ROM Chip Select Number is determined by BOOT_CFG4[5:4] (Chip Select) fuse.

When using the SPI as boot device, the Chip supports booting from both Serial EEPROM and Serial Flash devices. The boot code determines which device is being used by reading the appropriate eFUSE/I/O values at boot (see Table 8-22 for details).

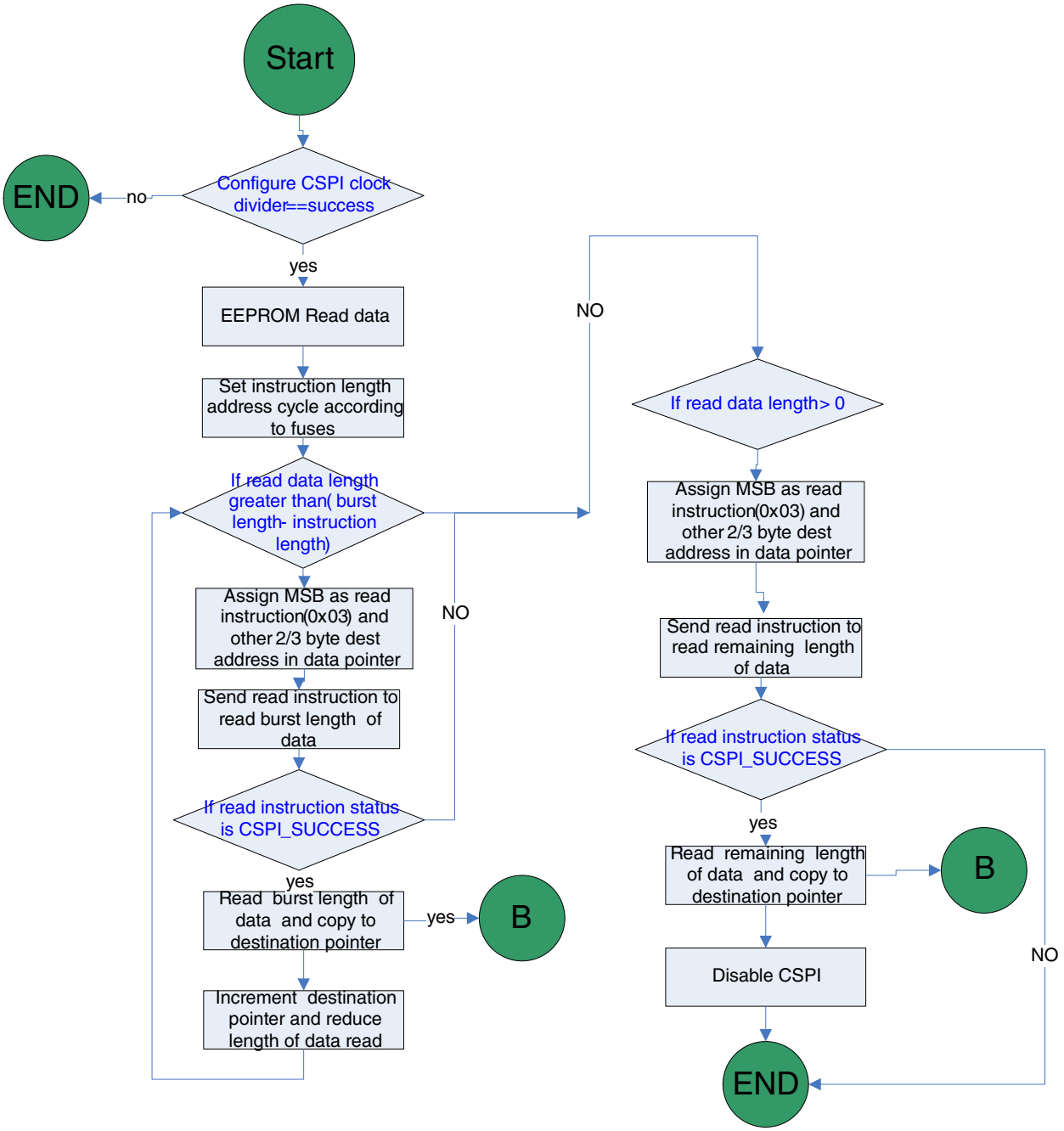


Figure 8-20. CSPI Flow chart

8.5.5.3.1 ECSPI IOMUX Pin Configuration

The contacts assigned to the signals used by the three CSPI blocks is shown in the table below.

Table 8-25. SPI IOMUX Pin Configuration

Signal	ECSPI-1	ECSPI-2	ECSPI-3	ECSPI4	ECSPI-5
MISO	EIM_D17.alt1	CSI0_DAT10.alt2	DISP0_DAT2.alt2	EIM_D22.alt1	SD1_DAT0.alt1
MOSI	EIM_D18.alt1	CSI0_DAT9.alt2	DISP0_DAT1.alt2	EIM_D28.alt2	SD1_CMD.alt1
RDY	N/A[1]	N/A	N/A	N/A	N/A
SCLK	EIM_D16.alt1	CSI0_DAT8.alt2	DISP0_DAT0.alt2	EIM_D21.alt1	SD1_CLK.alt1
SS0	EIM_EB2.alt1	CSI0_DAT11.alt2	DISP0_DAT3.alt2	EIM_D20.alt1	SD1_DAT1.alt1
SS1	EIM_D19.alt1	EIM_LBA.alt1	DISP0_DAT4.alt2	EIM_A25.alt1	SD1_DAT2.alt1
SS2	EIM_D24.alt1	EIM_D24.alt4	DISP0_DAT5.alt2	EIM_D24.alt1	SD1_DAT3.alt1
SS3	EIM_D25.alt1	EIM_D25.alt4	DISP0_DAT6.alt2	EIM_D25.alt1	SD2_DAT3.alt1

1. N/A in the ROM code indicates the pins are not available or not used.

8.6 QuadSPI Serial Flash Memory Boot

8.6.1 QuadSPI eFUSE Configuration

Table 8-26. QSPI Boot eFUSE Descriptions

Fuse	Config	Definition	GPIO	Shipped Value	Settings
BOOT_CFG1[7:4]	OEM	Boot Device Selection	Yes	0001	0001 - Boot from QuadSPI
BOOT_CFG1[3]	OEM	QuadSPI Interface Selection	Ye	0	0 - QSPI1 1 - QSPI2

8.6.2 QuadSPI Serial Flash BOOT Operation

The Boot ROM will attempt to boot from QuadSPI flash if the "BOOT_CFG1[7:4]" fuses are programmed to "0001" as shown in the QuadSPI eFUSE Configuration table. The ROM will initialize the requested the QuadSPI Interface as selected in Fuse bit BOOT_CFG1[3] in the QuadSPI eFUSE Configuration. QuadSPI interface initialization is a two step process.

The ROM expects the QuadSPI configuration parameters as explained in the QuadSPI Configuration Parameters to be present in the Serial Flash memory from offset 0x400 of serial flash of length 368 bytes. The ROM reads these configuration parameters using the default read command configured in the LUT of the QuadSPI interface with SCLOCK operating at 18 MHz.

In the second step, ROM configures the selected QuadSPI interface with the configuration parameters read from the serial flash and starts the boot procedure. Refer to Table 19-12 for details regarding QuadSPI configuration parameters and to the QuadSPI boot flow chart for detailed boot flow chart of QuadSPI.

Both booting an XIP and non XIP image is supported from serial flash. For XIP boot, the image has to be built for QuadSPI address space and for non XIP the image can be built to execute from DDR or OCRAM.

For QUAD mode boot, the Boot ROM expects the Quad Enable bit inside the QSPI Flash to be already set before booting starts. Therefore, the QUAD enable bit must be set in the non-volatile register of the flash at the time of programming.

NOTE

If the SPI flash device requires quad enable command, it can be sent via configuration structure fields: device_quad_mode_en, device_cmd, write_cmd_ipcr, write_enable_ipcr, busy_bit_offset, read_status_ipcr.

8.6.3 QuadSPI Configuration Parameters

The QuadSPI Configuration Parameters Table is built in boot image at fixed offset 0x400 from QSPI NOR A1 base address (368 bytes). Table below lists various QuadSPI Configuration Parameters.

Table 8-27. QuadSPI Configuration Parameters

Name	Offset	Size in Bytes	Description
DQS Loopback	0	4	DQS LoopBack Mode to enable Dummy Pad, 0 - Disable, 1 - Enable
Hold Delay	4	4	Hold Delay for QSPI[0,1] A/B Value / QSPI1 B / QSPI1 A/QSPI2 A 00 / Disable / Disable 01 / Disable / Enable 10 / Enable / Disable 11 / Enable / Enable
Reserved	8	4	Reserved to 0
Reserved	12	4	Reserved to 0
device_quad_mode_en	16	4	Send Quad enable command to SPI device.
device_cmd	20	4	Command to send to SPI device.

Table continues on the next page...

Table 8-27. QuadSPI Configuration Parameters (continued)

<table>
<tr><th>Name</th><th>Offset</th><th>Size in Bytes</th><th>Description</th></tr>
<tr><td>write_cmd_ipcr</td><td>24</td><td>4</td><td>IPCR register value for write command</td></tr>
<tr><td>write_enable_ipcr</td><td>28</td><td>4</td><td>IPCR register value for Enable</td></tr>
<tr><td>Chip Select hold time</td><td>32</td><td>4</td><td>This is chip select hold time in terms of Serial clock (For Example 1 serial clock cycle 0-15).</td></tr>
<tr><td>Chip Select setup time</td><td>36</td><td>4</td><td>Chip select setup time in terms of Serial clock (For example 1 serial clock).</td></tr>
<tr><td>Serial Flash A1 size</td><td>40</td><td>4</td><td>Serial Flash A1 size in units of bytes</td></tr>
<tr><td>Serial Flash A2 size</td><td>44</td><td>4</td><td>Serial Flash A2 size in units of bytes</td></tr>
<tr><td>Serial Flash B1 size</td><td>48</td><td>4</td><td>Serial Flash B1 size in units of bytes</td></tr>
<tr><td>Serial Flash B2</td><td>52</td><td>4</td><td>Serial Flash B2 size in units of bytes</td></tr>
<tr><td>Serial Clock Frequency</td><td>56</td><td>4</td><td>This is serial clock frequency select parameter.
<table><tr><th>Value</th><th>Clock</th></tr><tr><td>00</td><td>18 MHz</td></tr><tr><td>01</td><td>49 MHz</td></tr><tr><td>02</td><td>55 MHz</td></tr><tr><td>03</td><td>60 MHz</td></tr><tr><td>04</td><td>66 MHz</td></tr><tr><td>05</td><td>76 MHz</td></tr><tr><td>06</td><td>99Mhz (only SDR mode)</td></tr></table></td></tr>
<tr><td>busy_bit_offset</td><td>60</td><td>4</td><td>SPI Flash device busy bit offset in its status register, used for enabling Quad mode of SPI device</td></tr>
<tr><td>Mode of operation of serial Flash</td><td>64</td><td>4</td><td>This field describes the mode of operation of Serial flash
<table><tr><th>Value</th><th>Mode</th></tr><tr><td>01</td><td>Single</td></tr><tr><td>02</td><td>Dual</td></tr><tr><td>04</td><td>Quad</td></tr></table></td></tr>
<tr><td>Serial Flash Port B Selection</td><td>68</td><td>4</td><td>Port A is always available. This field informs the device ROM the availability of Port B.
0 – Port B is not used
1 – Port B is used</td></tr>
<tr><td>Dual Data Rate mode enable</td><td>72</td><td>4</td><td>This field enables the device ROM to enable DDR mode.
0 – DDR mode is disabled
1 – DDR mode is enabled</td></tr>
</table>

Table continues on the next page...

Table 8-27. QuadSPI Configuration Parameters (continued)

Name	Offset	Size in Bytes	Description
Data Strobe Signal enable in Serial Flash	76	4	This field enables Data Strobe signal in Serial Flash which supports it. 0 – Disable DQS 1 – Enable DQS
Parallel Mode enable	80	4	This field enables parallel mode. Data will be read from serial Flash in parallel mode. Refer to QSP chapter for detail. 0 – Disable Parallel mode in QSPI 1 – Enable Parallel Mode in QSPI
CS1 on Port A	84	4	This field enables CS1 on port A 0 – Disable CS1 on Port A 1 – Enable CS1 on Port A
CS1 on Port B	88	4	This field enables CS1 on port B 0 – Disable CS1 on Port B 1 – Enable CS1 on Port B
Full Speed Phase Selection	92	4	Select the edge of the sampling clock valid for full speed commands: 0: Select sampling at non-inverted clock 1: Select sampling at inverted clock This bit is also used to shift the dqs_enable when DQS mode is selected
Full Speed Delay Selection	96	4	Select the delay w.r.t. the reference edge for the sample point valid for full speed commands: 0: One clock cycle delay 1: Two clock cycles delay This bit is also used to shift the dqs_enable when DQS mode is selected
DDR Sampling Point	100	4	Select the sampling point for incoming data when serial flash is in DDR mode. **NOTE:** Valid Values are (b000-b111)
LUT program sequence	104	256	256 Bytes of Look up table program sequence. ROM programs the LUT of QuadSPI with this parameter supplied. It assumes that the optimize read command sequence which will be used to read data from Serial flash and fill the AHB buffer is programmed at index 0.
read_status_ipcr	360	4	IPCR value of Read Status Reg
enable_dqs_phase	364	4	Enable DQS Phase

8.6.4 QuadSPI boot flow chart

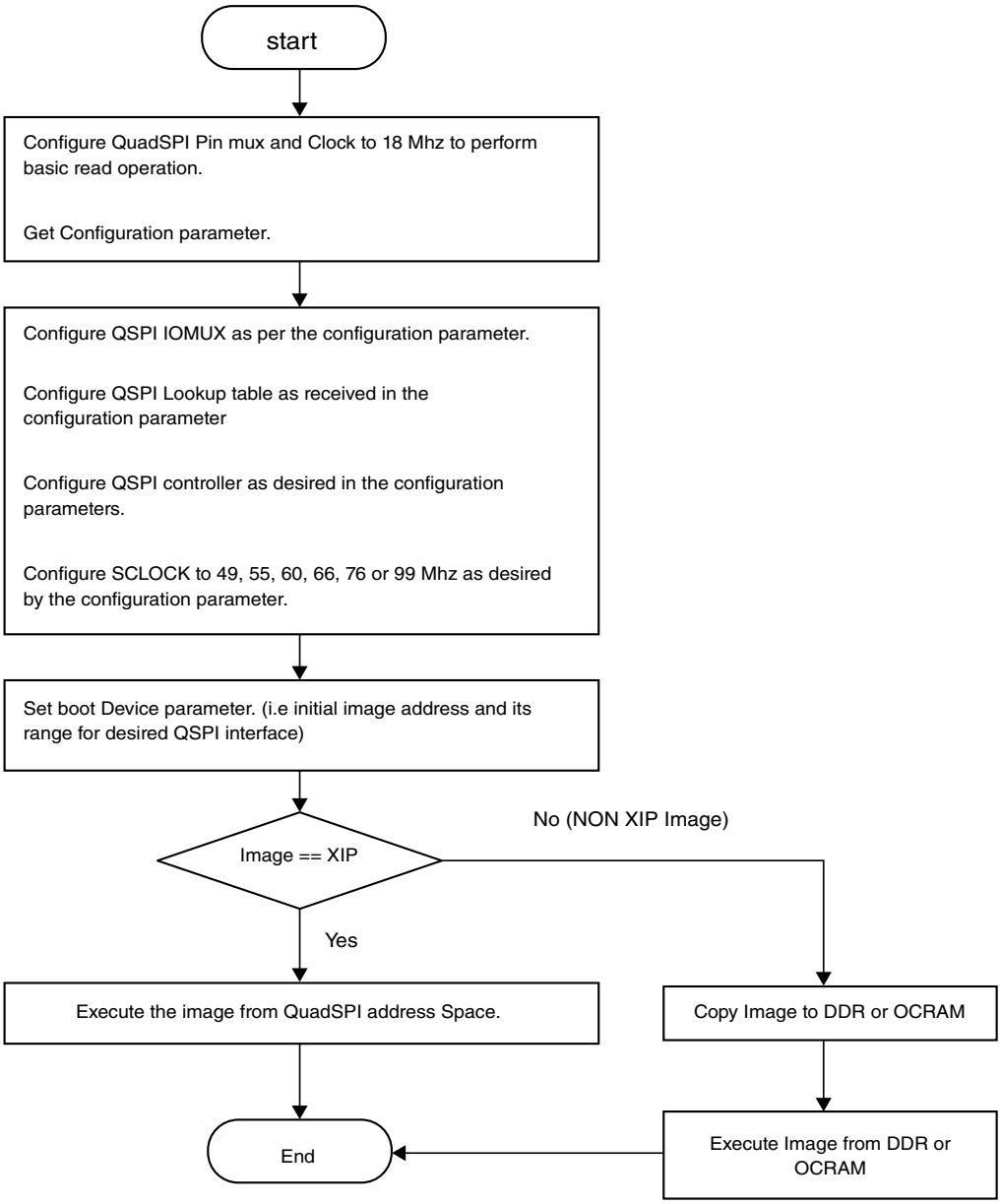


Figure 8-21. QuadSPI boot flow chart

NOTE

If flash is configured for "High performance mode (where command is generated only once)" in LUT program sequence. then external reset should be routed to flash reset to allow rebooting in case of any device reset other than Power On Reset. Also this high performance mode must be exited by application before any Low power mode entry where the device is supposed to reboot from QSPI flash on Low power mode exit. In general, any preserved configuration in external flash will not be understood by device after reset.

8.7 Program image

This section describes the data structures that are required to be included in a user's program image. A program image consists of:

- Image vector table—A list of pointers located at a fixed address that the ROM examines to determine where other components of the program image are located
- Boot data—A table indicating the program image location, program image size in bytes, and the plugin flag
- Device configuration data—IC configuration data
- User code and data

8.7.1 Image Vector Table and Boot Data

The Image Vector Table (IVT) is the data structure that the ROM reads from the boot device supplying the program image containing the required data components to perform a successful boot.

The IVT includes the program image entry point, a pointer to Device Configuration Data (DCD) and other pointers used by the ROM during the boot process.The ROM locates the IVT at a fixed address that is determined by the boot device connected to the Chip. The IVT offset from the base address and initial load region size for each boot device type is defined in the table below. The location of the IVT is the only fixed requirement by the ROM. The remainder or the image memory map is flexible and is determined by the contents of the IVT.

Table 8-28. Image Vector Table Offset and Initial Load Region Size

Boot Device Type	Image Vector Table Offset	Initial Load Region Size
NOR	4 Kbyte = 0x1000 bytes	Entire Image Size
NAND	1 Kbyte = 0x400 bytes	4 Kbyte
OneNAND	256 bytes = 0x100 bytes	1 Kbyte
SD/MMC/eSD/eMMC/SDXC	1 Kbyte = 0x400 bytes	4 Kbyte
I2C/SPI EEPROM	1 Kbyte = 0x400 bytes	4 Kbyte
SATA	1 Kbyte = 0x400 bytes	4 Kbyte

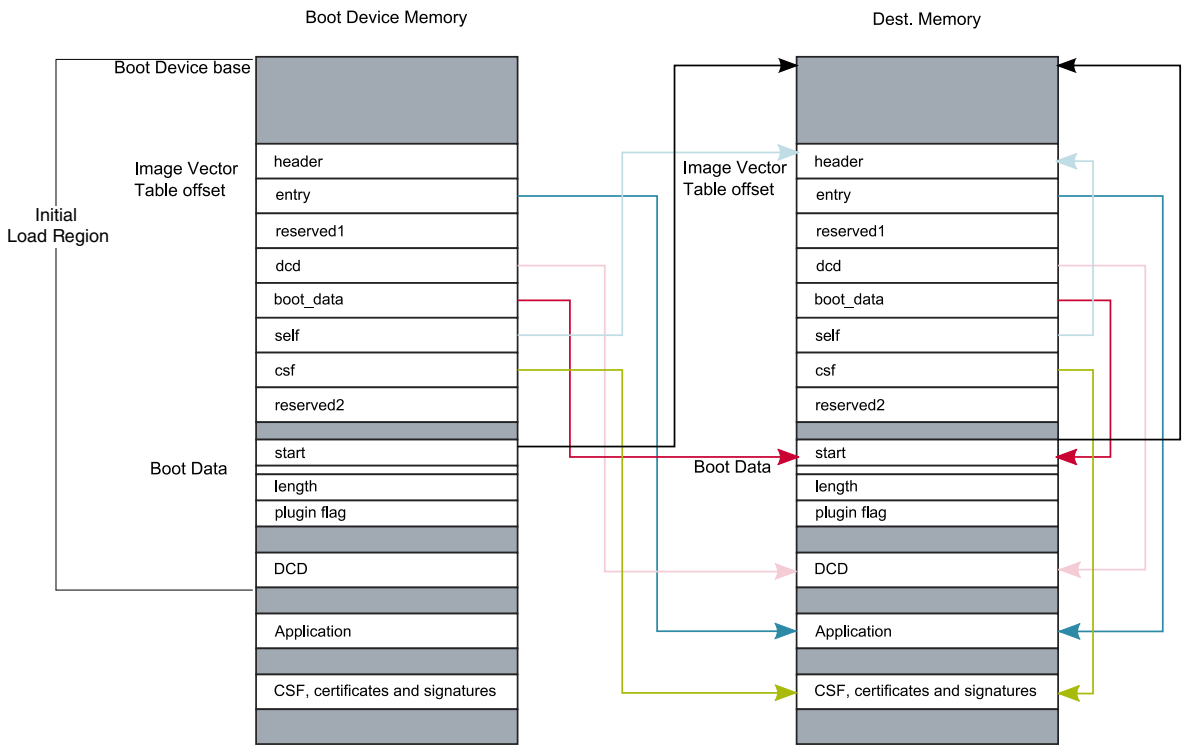


Figure 8-22. Image Vector Table

8.7.1.1 Image Vector Table Structure

The IVT has the following format where each entry is a 32 bit word:

Table 8-29. IVT Format

header
entry: Absolute address of the first instruction to execute from the image
reserved1: Reserved and should be zero
dcd: Absolute address of the image DCD. The DCD is optional so this field may be set to NULL if no DCD is required. See Device Configuration Data (DCD) for further details on DCD.
boot data: Absolute address of the Boot Data
self: Absolute address of the IVT. Used internally by the ROM
csf: Absolute address of Command Sequence File (CSF) used by the HAB library. See High Assurance Boot (HAB) for details on secure boot using HAB. This field must be set to NULL when not performing a secure boot
reserved2: Reserved and should be zero

The IVT header has the following format:

Table 8-30. IVT Header Format

Tag	Length	version

where:

```
Tag: A single byte field set to 0xD1
Length: a two byte field in big endian format containing the overall length of the IVT,
in bytes, including the header. (the length is fixed and must have a value of
32 bytes)
Version: A single byte field set to 0x40 or 0x41
```

8.7.1.2 Boot Data Structure

The Boot Data must follow the format defined in the table found here, each entry is a 32-bit word.

Table 8-31. Boot Data Format

start	Absolute address of the image
length	Size of the program image
plugin	Plugin flag (see Plugin Image)

8.7.2 Device Configuration Data (DCD)

Upon reset, the Chip uses the default register values for all peripherals in the system. However, these settings typically are not ideal for achieving optimal system performance and there are even some peripherals that must be configured before they can be used.

The DCD is configuration information contained in a Program Image, external to the ROM, that the ROM interprets to configure various peripherals on the Chip.

For example, the EIM default settings allow the core to interface to a NOR flash device immediately out of reset. This allows the Chip to interface with any NOR flash device, but has the cost of slow performance. Additionally, some components such as DDR require some sequence of register programming as part of configuration before it is ready to be used. The DCD feature can be used to program the EIM registers and MMDC registers to the optimal settings.

The ROM determines the location of the DCD table based on information located in the Image Vector Table (IVT). See Image Vector Table and Boot Data for more details. The DCD table shown below is a big endian byte array of the allowable DCD commands. The maximum size of the DCD limited to 1768 bytes.

Table 8-32. DCD Data format

Header
[CMD]
[CMD]
...

The DCD header is 4 bytes with the following format:

Table 8-33. DCD Header

Tag	Length	Version

where:

```
Tag: A single byte field set to 0xD2
Length: a two byte field in big endian format containing the overall length of the DCD,
in bytes, including the header
Version: A single byte field set to 0x41
```

8.7.2.1 Write Data Command

The Write Data Command is used to write a list of given 1-, 2- or 4-byte values or bitmasks to a corresponding list of target addresses.

The format of Write Data Command, again a big endian byte array, is shown in the table below.

Table 8-34. Write Data Command Format

Tag	Length	Parameter
Address		
Value/Mask		
[Address]		
[Value/Mask]		
...		
[Address]		
[Value/Mask]		

where:

```
Tag: A single byte field set to 0xCC
Length: A two byte field in big endian format containing the length of the Write Data
Command, in bytes, including the header
Address: target address to which data should be written
Value/Mask: data value or bitmask to be written to preceding address
```

The Parameter field is a single byte divided into bitfields as follows:

Table 8-35. Write Data Command Parameter field

7	6	5	4	3	2	1	0
flags					bytes		

where

```
bytes: width of target locations in bytes. Either 1, 2 or 4
flags: control flags for command behavior.
Data Mask = bit 3: if set, only specific bits may be overwritten at target address
(otherwise all bits may be overwritten)
Data Set = bit 4: if set, bits at the target address overwritten with this flag
(otherwise it is ignored)
```

One or more target address and value/bitmask pairs can be specified. The same bytes and flags parameters apply to all locations in the command.

When successful, this command writes to each target address in accordance with the flags as follows:

Table 8-36. Interpretation of Write Data Command Flags

"Mask"	"Set"	Action	Interpretation
0	0	*address = val_msk	Write value
0	1	*address = val_msk	Write value
1	0	*address &= ~val_msk	Clear bitmask
1	1	*address |= val_msk	Set bitmask

NOTE

If any of the target addresses does not have the same alignment as the data width indicated in the parameter field, none of the values are written.

If any of the values is larger or any of the bitmasks is wider than permitted by the data width indicated in the parameter field, none of the values are written.

If any of the target addresses do not lie within an allowed region, none of the values are written. The list of allowable blocks and target addresses for the Chip are given below.

Table 8-37. Valid DCD Address Ranges

Address range	Start address	Last Address
IOMUX Control (IOMUXC) registers	0x020E0000	0x020E3FFF
CCM register set	0x020C4000	0x020C7FFF

Table continues on the next page...

Table 8-37. Valid DCD Address Ranges (continued)

Address range	Start address	Last Address
ANADIG registers	0x020C8000	0x020C8FFF
MMDC register set	0x021B0000	0x021B7FFF
IRAM Free Space	0x00907000	0x00937FF0
EIM - Memory	0x08000000	0x0FFEFFFF
EIM - Registers	0x021B8000	0x021BBFFF
DDR	0x10000000	0xFFFFFFFF

8.7.2.2 Check Data Command

The Check Data Command is used to test for a given -1, 2- or 4-byte bitmasks from a source address.

The Check Data Command is a big endian byte array with format shown in the table below.

Table 8-38. Check Data Command Format

Tag	Length	Parameter
Address		
Mask		
[Count]		

where:

```
Tag: A single byte field set to 0xCF
Length: A two byte field in big endian format containing the length of the Check Data
Command, in bytes, including the header
Address: source address to test
Mask: bit mask to test
Count: optional poll count. If count is not specified this command will poll indefinitely
until the exit condition is met. If count = 0, this command behaves as for NOP.
```

The Parameter field is a single byte divided into bitfields as follows:

Table 8-39. Check Data Command Parameter field

7	6	5	4	3	2	1	0
flags					bytes		

where

```
bytes: width of target locations in bytes. Either 1, 2 or 4
flags: control flags for command behavior.
Data Mask = bit 3: if set, only specific bits may be overwritten at target address
(otherwise all bits may be overwritten)
```

```
Data Set = bit 4: if set, bits at the target address overwritten with this flag
(otherwise it is ignored)
```

This command polls the source address until either the exit condition is satisfied, or the poll count is reached. The exit condition is determined by the flags as follows:

Table 8-40. Interpretation of Check Data Command Flags

"Mask"	"Set"	Action	Interpretation
0	0	(*address & mask) == 0	All bits clear
0	1	(*address & mask) == mask	All bits set
1	0	(*address & mask)!= mask	Any bit clear
1	1	(*address & mask)!= 0	Any bit set

NOTE

If the source address does not have the same alignment as the data width indicated in the parameter field, the value is not read.

If the bitmask is wider than permitted by the data width indicated in the parameter field, the value is not read.

8.7.2.3 NOP Command

This command has no effect.

The format of NOP Command is a big endian four byte array as shown in the table below.

Table 8-41. NOP Command Format

Tag	Length	Undefined

where:

```
Tag: A single byte field set to 0xC0
Length: A two byte field in big endian containing the length of the NOP Command in bytes.
Fixed to a
value of 4.
Undefined: This byte is ignored and can be set to any value.
```

8.7.2.4 Unlock Command

The Unlock Command is used to prevent specific engine features being locked when exiting ROM.

The format of Unlock Command, again a big endian byte array, is shown in the table below.

Table 8-42. Unlock Command Format

Tag	Length	Eng
Value		
Value		
...		
Value		

where:

```
Tag: A single byte field set to 0xB2
Eng: Engine to be left unlocked.
Values: [optional] unlock values required by engine.
```

NOTE

This command may not be used in DCD structure if the SEC_CONFIG is configured as closed.

8.8 Plugin Image

The ROM supports a limited number of boot devices. For using other devices as boot source (for example, Ethernet, CDROM, or USB), the supported boot device must be used (typically serial ROM) for firmware with the missing boot drivers. >Additionally plugin can customize supported boot drivers. It is more flexible when doing device initialization, such as condition judging, delay assertion, applying custom settings to boot device and memory system.

>In addition to standard images, the chip also supports plugin images. Plugin images return execution to the ROM whereas a standard image does not.

The boot ROM detects the image type using the plugin flag of the boot data structure (see Boot Data Structure). If the plugin flag is 1, then the ROM uses the image as a plugin function. The function must initialize the boot device and copy the program image to the final location. At the end the plugin function must return with the program image parameters. (See High level boot sequence for details about boot flow).

The boot ROM authenticates the plugin image prior to running the plugin function and then authenticates the program image.

The plugin function must follow the API described below:

typedef unsigned char (*) plugin_download_f(void **start, size_t *bytes, UINT32 *ivt_offset)

ARGUMENTS PASSED:

- start - Image load address on exit.
- bytes - Image size on exit.
- ivt_offset - Offset in bytes of the IVT from the image start address on exit.

RETURN VALUE:

- 1 - on success
- 0 - on failure

8.9 Serial Downloader

The Serial Downloader provides a means to download a Program Image to the chip over USB serial connection.

In this mode the ROM programs WDOG-1 for a 32-second time-out if WDOG_ENABLE eFuse is 1, and then continuously polls for USB connection. If no activity is found on USB OTG1 and the watchdog timer expires, the ARM core is reset.

NOTE

The downloaded image must continue to service the watchdog timer to avoid an undesired reset from occurring.

The USB boot flow is shown in the figure below.

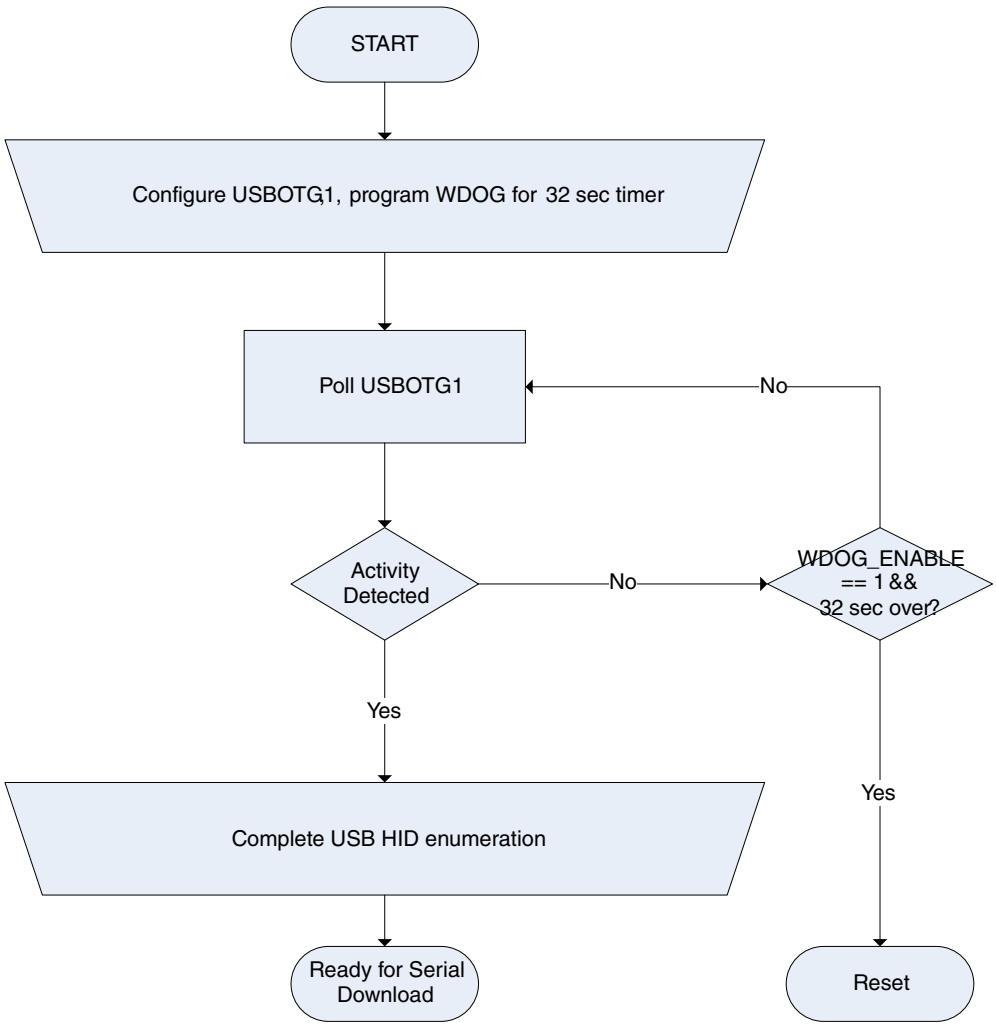


Figure 8-23. USB Boot Flow

8.9.1 USB

USB support is composed of the USBOH3 (USB OTG1 core controller, compliant with the USB 2.0 specification) and the USBPHY (HS USB transceiver).

The ROM supports the USB OTG port for boot purposes. The other USB ports on the chip are not supported for boot purposes.

The USB Driver is implemented as a USB HID class. A collection of 4 HID reports are used to implement SDP protocol for data transfers as described in Table 8-43.

Table 8-43. USB HID Reports

Report ID (first byte)	Transfer Endpoint	Direction	Length	Description
1	control OUT	Host to device	17 bytes	SDP command from host to device
2	control OUT	Host to device	Up to 1025 bytes	Data associated with report 1 SDP command
3	interrupt	Device to host	5 bytes	HAB security configuration. Device sends 0x12343412 in closed mode and 0x56787856 in open mode.
4	interrupt	Device to host	Up to 65 bytes	Data in response to SDP command in report 1

8.9.1.1 USB Configuration Details

The USB OTG function device driver supports a high speed (HS for UTMI) non-stream mode with a maximal packet size of 512 B and a low-level USB OTG function.

The VID/PID and strings for USB device driver are listed in the table below.

Table 8-44. VID/PID and Strings for USB Device Driver

Descriptor	Value
VID	0x15A2 (Freescale vendor ID)
PID[1]	0x0054
String Descriptor1 (manufacturer)	Freescale Semiconductor, Inc.
String Descriptor2 (product)	S Blank ARIK SE Blank ARIK NS Blank ARIK
String Descriptor4	Freescale Flash
String Descriptor5	Freescale Flash

1. Allocation based on BPN (Before Part Number)

8.9.1.2 IOMUX Configuration for USB

The interface signals of the UTMI PHY are not configured in the IOMUX. The UTMI PHY interface uses dedicated contacts on the IC. See the Chip data sheet for details.

8.9.2 Serial Download protocol

The 16 byte SDP command from host to device is sent using HID report 1.

The table below describes 16 byte SDP command data structure:

Table 8-45. 16 Byte SDP Command Data Structure

BYTE Offset	Size	Name	Description
0	2	COMMAND TYPE	The following commands are supported for i.MX6 Dual/6Quad ROM: • 0x0101 READ_REGISTER • 0x0202 WRITE_REGISTER • 0x0404 WRITE_FILE • 0x0505 ERROR_STATUS • 0x0A0A DCD_WRITE • 0x0B0B JUMP_ADDRESS
2	4	ADDRESS	Only relevant for following commands: READ_REGISTER, WRITE_REGISTER, WRITE_FILE, DCD_WRITE, and JUMP_ADDRESS. For READ_REGISTER and WRITE_REGISTER commands, this field is address to a register. For WRITE_FILE and JUMP_ADDRESS commands, this field is an address to internal or external memory address.
6	1	FORMAT	Format of access, 0x8 for 8-bit access, 0x10 for 16-bit and 0x20 for 32-bit access. Only relevant for READ_REGISTER and WRITE_REGISTER commands.
7	4	DATA COUNT	Size of data to read or write. Only relevant for WRITE_FILE, READ_REGISTER, WRITE_REGISTER and DCD_WRITE commands. For WRITE_FILE and DCD_WRITE commands DATA COUNT is in byte units.
11	4	DATA	Value to write. Only relevant for WRITE_REGISTER command.
15	1	RESERVED	Reserved

8.9.2.1 SDP Command

SDP commands are decribed in the following sections.

8.9.2.1.1 READ REGISTER

The transaction for command READ_REGISTER consists of following reports: Report1 for command, Report3 for security configuration and Report4 for response or register value.

The register to read is specified in ADDRESS field of SDP command. First device sends Report3 with security configuration followed by Report4 with bytes read at given address. If count is greater than 64 then multiple reports with report id 4 are sent until entire data requested by host is sent. The STATUS is either 0x12343412 for closed parts and 0x56787856 for open or field return parts.

Report1, Command, Host to Device:

1	Valid values for READ_REGISTER COMMAND, ADDRESS, FORMAT, DATA_COUNT

ID 16 byte SDP Command

Report3, Response, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

Report4, Response, Device to Host: first response report

4	Register Value

ID 4 bytes of data containing register value. If number of bytes requested is less than 4 then remaining bytes should be ignored by host.

Multiple reports of report id 4 are sent until entire data requested is sent

Report4, Response, Device to Host: Last response report

4	Register Value

ID 64 bytes of data containing register value. If number of bytes requested is less than 64 then remaining bytes should be ignored by host.

8.9.2.1.2 WRITE REGISTER

The transaction for command WRITE_REGISTER consists of the following reports: Report1 for command, Report3 for security configuration and Report4 for write status.

Host sends Report1 with WRITE_REGISTER command. The register to write is specified in ADDRESS field of SDP command of Report1, with FORMAT field set to data type (number of bits to write 8, 16 or 32) and value to write in DATA field of SDP command. Device writes the DATA to register address and returns WRITE_COMPLETE code using Report4 and security configuration using Report3 to complete the transaction.

Report1, Command, Host to Device:

1	Valid values for WRITE_REGISTER COMMAND, ADDRESS, FORMAT, DATA_COUNT and DATA

ID 16 byte SDP Command

Report3, Response, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

Report4, Response, Device to Host:

4	WRITE_COMPLETE (0x128A8A12) status

ID 64 bytes data with first 4 bytes to indicate write is completed with code 0x128A8A12. On failure device will report HAB error status.

8.9.2.1.3 WRITE_FILE

The transaction for command WRITE_FILE consists of following reports: Report1 for command-phase, Report2 for data-phase, Report3 for hab mode and Report4 to indicate data received in full.

The size of each Report2 is limited to 1024 bytes (limitation of USB HID protocol) hence multiple Report2 packets will be sent by host in data phase until entire data is transferred to device. Once entire data (DATA_COUNT bytes) is received then device sends report 3 with hab mode and report 4 with 0x88888888, indicating file download completed.

Report1, Host to Device:

1	Valid values for WRITE_FILE COMMAND, ADDRESS, DATA_COUNT

ID 16 byte SDP Command

==========Optional Begin==========

Host sends ERROR_STATUS command to query if HAB rejected the address

========== Optional End==========

Report2, Host to Device:

2	File data

ID Max 1024 bytes data per report

Report2, Host to Device:

2	File data

ID Max 1024 bytes data per report

Report3, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

Report4, Response, Device to Host:

4	COMPLETE (0x88888888) status

ID 64 bytes data with first 4 bytes to indicate file download has completed with code 0x88888888. On failure device will report HAB error status.

8.9.2.1.4 ERROR_STATUS

The transaction for SDP command ERROR_STATUS consists of three reports.

Report1 is used by host to send the command; device sends global error status in 4 bytes of Report4 after returning security configuration in Report3. When device receives ERROR_STATUS command it will return global error status that is updated for each command. This command is useful to find out if last command resulted in device error or succeeded.

Report1, Command, Host to Device:

1	ERROR_STATUS COMMAND

ID 16 byte SDP Command

Report3, Response, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

Report4, Response, Device to Host:

4	4 bytes Error status

ID first 4 bytes status in 64 bytes report 4

8.9.2.1.5 DCD WRITE

The SDP command DCD_WRITE is used by host to send multiple register writes in one shot. This command is provided to speed up the process of programming register writes such as to configure external RAM device.

The command goes with Report1 from host with COMMAND TYPE set to DCD_WRITE, ADDRESS which is used for temporary location of DCD data and DATA_COUNT to number of bytes sent in data out phase. In data phase host sends data for number of registers using Report2. Device completes the transaction with Report3 indicating security configuration and report 4 with WRITE_COMPLETE code 0x12828212.

Report1, Command, Host to Device:

1	DCD_WRITE COMMAND, ADDRESS, DATA_COUNT

ID 16 byte SDP Command

Report2, Data, Host to Device:

2	DCD binary data

ID Max 1024 bytes per report

Report3, Response, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

Report4, Response, Device to Host:

4	WRITE_COMPLETE (0x128A8A12) status

ID 64 bytes report with first 4 bytes to indicate write is completed with code 0x128A8A12. On failure device will report HAB error status.

See Device Configuration Data (DCD) for DCD format description.

8.9.2.1.6 JUMP ADDRESS

The SDP command JUMP_ADDRESS will be the last command host can send to the device, after this command device will jump to the address specified in the ADDRESS field of SDP command and start executing.

This command should typically follow after WRITE_FILE command. The command is sent by host in command-phase of transaction using Report1, there is no data phase for this command but device send status report3 to complete the transaction. And if HAB authentication fails then it will also send report 4 with HAB error status.

Report1, Command, Host to Device:

1	JUMP_ADDRESS COMMAND, ADDRESS

ID 16 byte SDP Command

Report3, Response, Device to Host:

3	4 bytes indicating security configuration

ID 4 bytes status

This report is sent by device only in case of an error jumping to the given address, device reports error in Report4, Response, Device to Host:

4	4 bytes HAB error status

ID 4 bytes status, 64 bytes report length

8.10 Recovery Devices

The Chip supports recovery devices. If primary boot device fails, boot ROM will try to boot from recovery device using one of I2C or ECSPI ports.

For enabling recovery device BOOT_CFG4[6] fuse must be set. Additionally Serial EEPROM fuses must be set as described in Serial ROM through SPI and I2C.

8.11 USB Low Power Boot

ROM supports USB Low Power Boot. This feature enables a device with dead or weak battery to power up and boot if the device is connected to a USB upstream port, no matter the upstream port is a USB charger or USB host/hub.

If a USB dedicated charger or host/hub charger are connected, as soon as the device is connected to the upstream port, a stable current (Max.1.5A) can be supplied by charger. If USB host/hub are connected, the maximal 1 00mA current is supplied to the device, the device should be able to power up to boot the image with less than 100mA.

If LPB_BOOT fuses are blown, the Chip will check if there is low power condition via GPIO_3 pad. If there is low power boot condition USB charger detection will be activated. If there is no USB charger, ROM will initialize USB as device and apply division factors on ARM, DDR, AXI and AHB root clocks based on LPB_BOOT fuses value (see the table below). Polarity of low power boot condition on GPIO_3 pad is set by BT _LPB_POLARITY fuse (see the figure below).

Table 8-68. USB Low Power Boot Frequencies

LPB_BOOT	Boot Frequencies=0	Boot Frequencies=1
00	ARM_CLK_ROOT=792MHz MMDC_CH0_AXI_CLK_ROOT=528MHz MMDC_CH1_AXI_CLK_ROOT=528MHz AXI_CLK_ROOT=264MHz AHB_CLK_ROOT=132MHz	ARM_CLK_ROOT=396MHz MMDC_CH0_AXI_CLK_ROOT=352MHz MMDC_CH1_AXI_CLK_ROOT=352MHz AXI_CLK_ROOT=176MHz AHB_CLK_ROOT=88MHz
01	ARM_CLK_ROOT=792MHz MMDC_CH0_AXI_CLK_ROOT=528MHz MMDC_CH1_AXI_CLK_ROOT=528MHz AXI_CLK_ROOT=264MHz AHB_CLK_ROOT=132MHz	ARM_CLK_ROOT=396MHz MMDC_CH0_AXI_CLK_ROOT=352MHz MMDC_CH1_AXI_CLK_ROOT=352MHz AXI_CLK_ROOT=176MHz AHB_CLK_ROOT=88MHz

Table continues on the next page...

Table 8-68. USB Low Power Boot Frequencies (continued)

LPB_BOOT	Boot Frequencies=0	Boot Frequencies=1
10	ARM_CLK_ROOT=396MHz MMDC_CH0_AXI_CLK_ROOT=264MHz MMDC_CH1_AXI_CLK_ROOT=264MHz AXI_CLK_ROOT=132MHz AHB_CLK_ROOT=66MHz	ARM_CLK_ROOT=264MHz MMDC_CH0_AXI_CLK_ROOT=176MHz MMDC_CH1_AXI_CLK_ROOT=176MHz AXI_CLK_ROOT=88MHz AHB_CLK_ROOT=44MHz
11	ARM_CLK_ROOT=264MHz MMDC_CH0_AXI_CLK_ROOT=132MHz MMDC_CH1_AXI_CLK_ROOT=132MHz AXI_CLK_ROOT=66MHz AHB_CLK_ROOT=66MHz	ARM_CLK_ROOT=132MHz MMDC_CH0_AXI_CLK_ROOT=88MHz MMDC_CH1_AXI_CLK_ROOT=88MHz AXI_CLK_ROOT=44MHz AHB_CLK_ROOT=44MHz

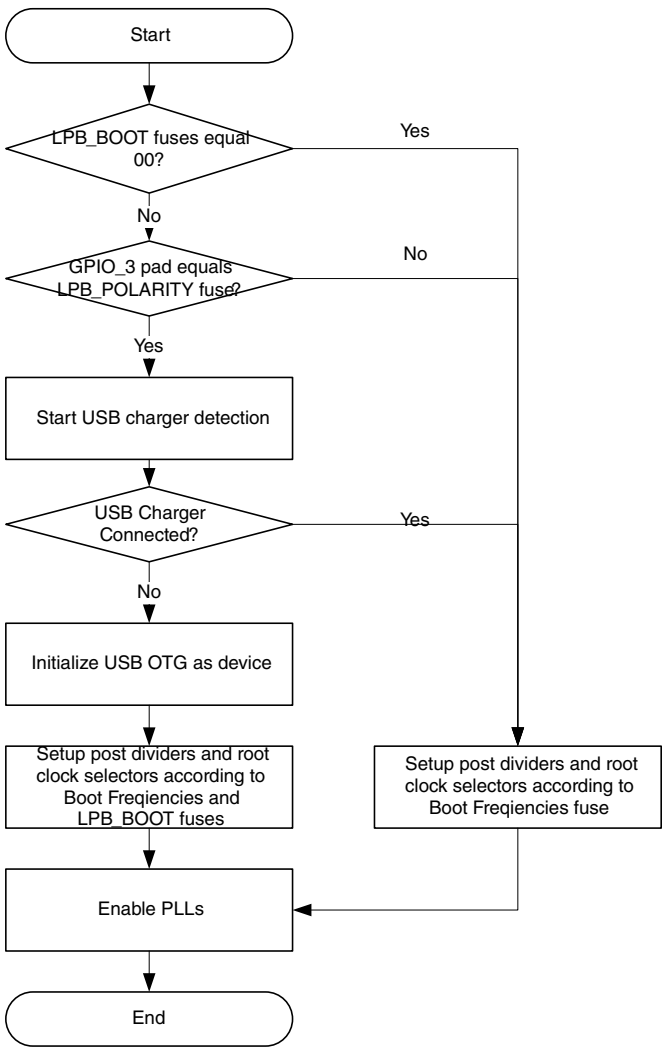


Figure 8-24. USB Low Power Boot Flow

8.12 High Assurance Boot (HAB)

The High Assurance Boot (HAB) component of the ROM protects against the potential threat of attackers modifying areas of code or data in programmable memory to make it behave in an incorrect manner. The HAB also prevents attempts to gain access to features which should not be available.

The integration of the HAB feature with the ROM code ensures that Chip does not enter an operational state if the existing hardware security blocks have detected a condition that may be a security threat or areas of memory deemed to be important have been modified. The HAB uses RSA digital signatures to enforce these policies.

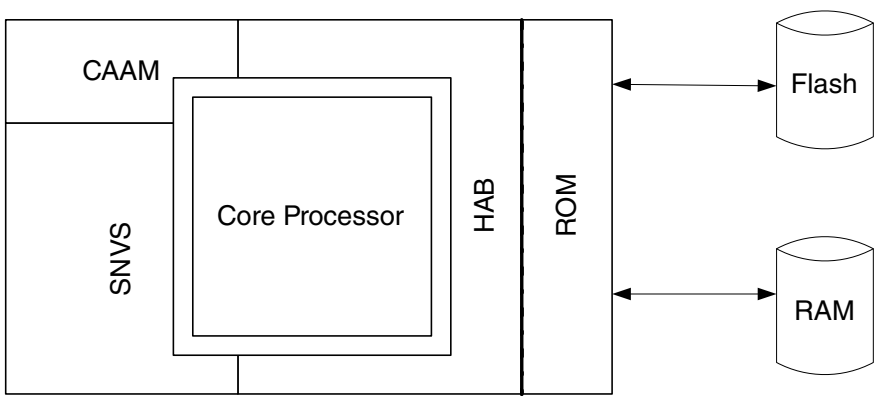


Figure 8-25. Secure Boot Components

The figure above illustrates the components used during a secure boot using HAB. The HAB interfaces with the SNVS to ensure the system security state is as expected. The HAB also makes use of CAAM hardware block to accelerate SHA-256 message digest operations performed during signature verifications and AES-128 operations for encrypted boot operations. The HAB also includes a software implementation of SHA-256 for cases where a hardware accelerator cannot be used. The RSA key sizes supported are 1024, 2048 and 3072 bits. The RSA signature verification operations are performed by a software implementation contained in the HAB library. The main features supported by HAB are:

- X.509 Public key certificate support

- CMS signature format support
- Proprietary encrypted boot support. Note that encrypted boot depends on the CAAM HW module. When CAAM is disabled (i.e. when the EXPORT_CONTROL fuse is blown) then encrypted boot is not available.

NOTE

Freescale provides a reference Code Signing Tool (CST) for key generation, certificate generation and code signing for use with the HAB library. The CST can be found by searching for "IMX_CST_TOOL" at http://www.freescale.com.

NOTE

For further details on making use of the secure boot feature using HAB contact your local Freescale representative.

8.12.1 HAB API Vector Table Addresses

For devices that perform a secure boot, the HAB library may be called by boot stages that execute after ROM code.

The RVT table contains the pointers to the HAB API functions. The address of the RTV table is shown in the table below.

Table 8-69. RVT Table Addresses

	HAB API vector table address	ROM API vector table address
Silicon revisions prior to 1.3	0x00000094	0x000000C0
Silicon revision 1.3 and later	0x00000098	0x000000C4

NOTE

For additional information on secure boot including the HAB API, contact your local Freescale representative.

Chapter 9
Multimedia

9.1 Video Graphics Sub System

The i.MX 6Dual/6Quad video graphics subsystem consists of the dedicated modules found here.

- Video Processing Unit (VPU): a multi-standard high performance video/image CODEC
- Three Graphics Processing Units (GPUs):
  - 3D GPU: accelerating the generation of 3D graphics (OpenGL/ES) and vector graphics (OpenVG)
  - 2D GPU: acceleration the generation of 2D graphics (BitBLT).
  - OpenVG: acceleration of vector graphics (OpenVG).
- Two (identical) Image Processing Units (IPUs): providing connectivity to cameras and displays, related processing, synchronization and control.
- Display interface bridges: providing optional translation from the digital display interface supported by the IPU to other interfaces:
  - LVDS bridge (LDB): providing up to two LVDS interfaces
  - HDMI transmitter
  - MIPI/DSI transmitter
- MIPI/CSI-2 receiver
- Two (identical) Display Content Integrity Checker (DCIC) are used to authenticate sensitive displayed data.
- A Video Data Order Adapter (VDOA): used to re-order video data from the "tiled" order used by the VPU to the conventional raster-scan order needed by the IPU.

High level integration scheme of the i.MX 6Dual/6Quad video/graphics system is provided by the figure below.

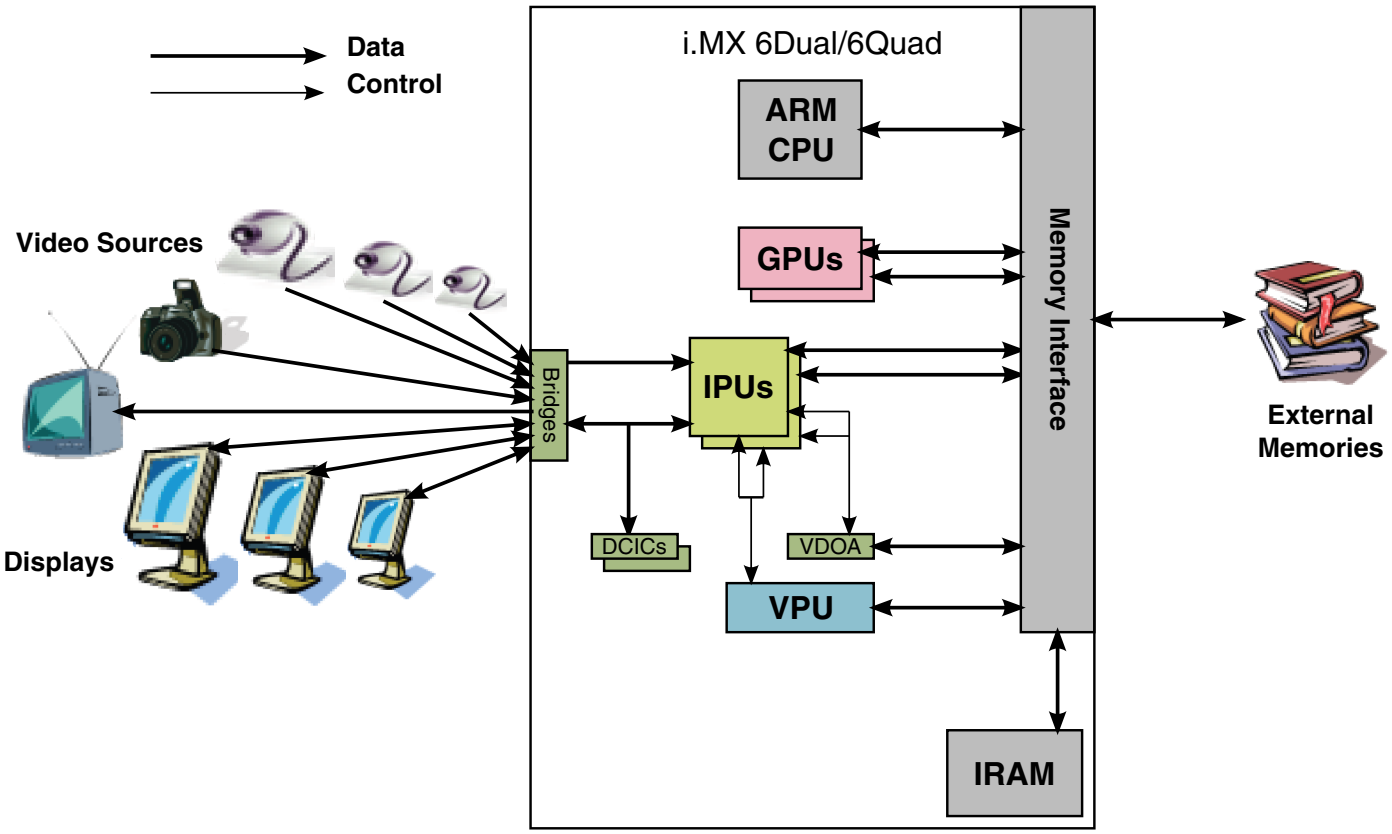


Figure 9-1. i.MX 6Dual/6Quad Video-Graphics Subsystem

9.1.1 Display outputs

The i.MX 6Dual/6Quad implements a robust muxing logic on the four display ports (2x per IPU), to the external interfaces, either direct, or via bridges (MIPI, LVDS, HDMI), per description below:

- Two parallel - driven directly by each of the IPUs; pixel clock at least up to 200 MHz (for external load of up to 10 pF).
  - Parallel interface works up to 200 MHz
  - HDMI interface works up to 240 MHz (IPU)
- Two LVDS channels, driven by the LDB; pixel clock up to 170 MHz.
- One HDMI port (ver. 1.4) - driven by the HDMI transmitter: pixel clock up to 264 MHz (gated by the IPU capabilities)
- One MIPI/DSI port - driven by the MIPI/DSI transmitter; 2 data lanes at 1 GHz
- Each IPU display port (DI) can be connected to each of the above ports

- Each IPU has 2 display ports, up to four external ports can be active at any given time. (Additional asynchronous data flows can be sent though the parallel ports and the MIPI/DSI port.)
- Read access is supported as follows
  - For the Parallel0 port: through DI00
  - For the Parallel1 port: through DI10
  - For the MIPI/DSI port: through DI01 or DI11

- Inputs to either of the DCICs are taken from one of the following buses
  - For each of the parallel interfaces: probing the I/O loopback (essentially equivalent to probing the external wires).
  - For other integrated interfaces (e.g. LVDS): probing the DI1 output of each of the IPUs (essentially equivalent to the inputs to the serializers)
  - For the data enable signal, two control signals are probed from each of the above buses.

For visual view of display signal routing, see the following figure. The chip MUX select signals are driven by configuration bits as specified in the IOMUX controller (IOMUXC) chapter in the IOMUXC_IOMUXC_GPR registers description fields.

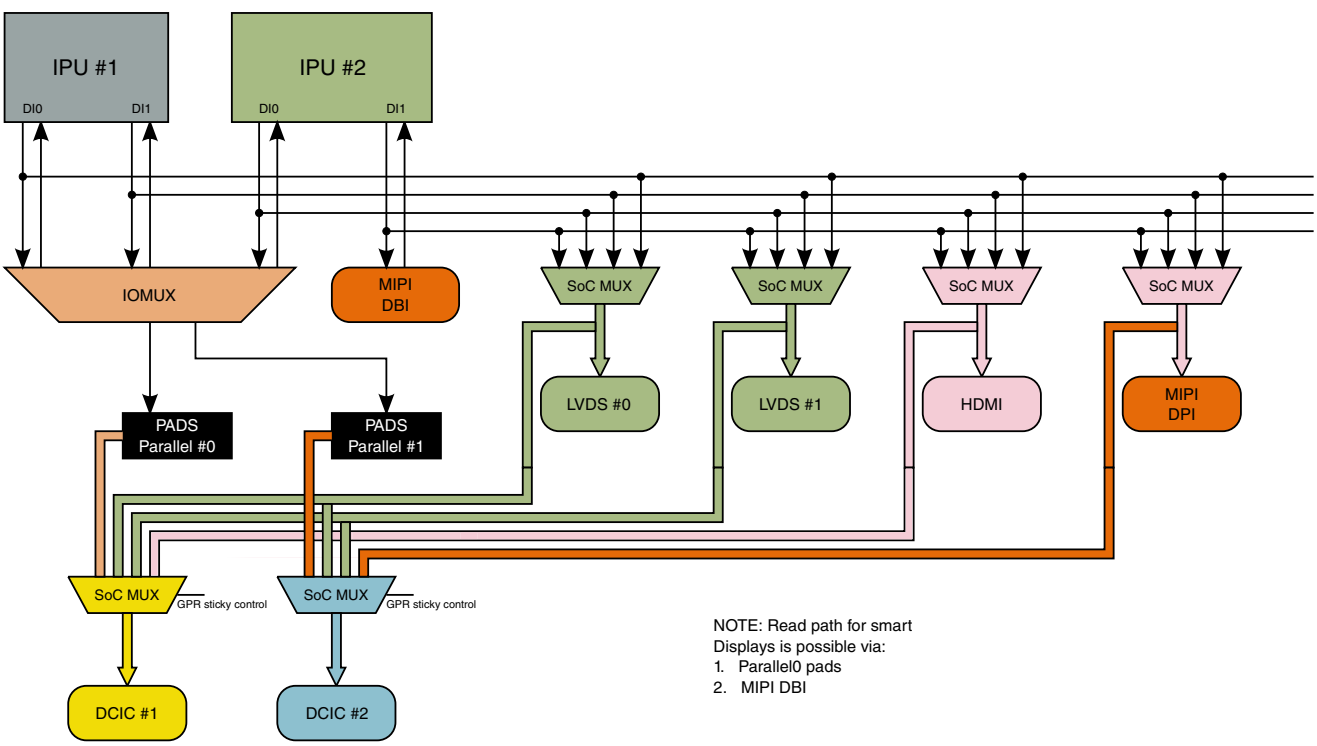


Figure 9-2. Display port muxing scheme

9.1.2 Video input

There are 3 video input ports (e.g. from image sensors).

See the figure below for details.

Video Input Ports Connectivity

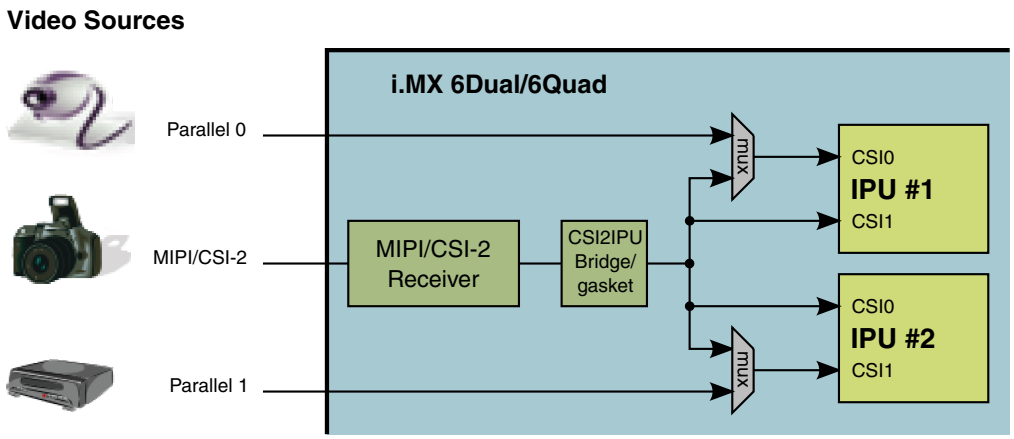


Figure 9-3. The Video Input Ports

Two parallel ports are connected directly to an IPU input port.

One MIPI/CSI-2 port- IPU receives two components per cycle from the MIPI_CSI2 interface. The maximum bandwidth of the interface is as follows:

- 400MByte/sec for four data lanes configuration (800Mbps/lane)
- 375MByte/sec for 3 data lanes configuration (1000Mbps/lane)
- 250MByte/sec for 2 data lanes configuration (1000Mbps/lane)
- 125Mbyte/sec for 1 data lanes configuration (1000Mbps/lane)

Each IPU has two input ports, CSI0 and CSI1, which can receive data concurrently and independently. At any given time, an IPU input port may receive data either from a parallel external port or from the MIPI/CSI-2 receiver.

The MIPI/CSI-2 port can receive up to 4 concurrent data channels. Each data channel is routed to a different CSI input of the IPU (2 IPUs, 2 CSIs on each IPU; a total of 4 CSI inputs). Pixel data can be further processed by the IPU. Other data types can be transferred through a CSI transparently as generic data to the system memory.

The IPUs, VPU, VDOA and the GPUs have master AXI ports, providing access to system memory.

The modules are controlled (by the ARM CPU or the SDMA) as follows:

- The LDB is controlled by signals connected to top-level registers.
- All other modules have a host interface. For the VPU, VDOA and the DCIC's, this is a slave IP port and for the MIPI bridges, GPUs, HDMI and IPUs, this is a slave AHB port.
- The data flow between VPU, VDOA, GPUs and IPUs is through system memory and it is normally controlled by the CPU. For spacial situations, a direct synchronization interface is provided:
  - An interface between the IPUs and VPU, for low-latency video record.
  - An interface between the VDOA and IPUs, for tight pipelining of data from the VDOA to the IPU, through the IRAM.

9.1.3 Synchronization Mechanisms

The i.MX 6Dual/6Quad provides HW synchronization mechanism between IPU and VPU and between IPU and VDOA to reduce core intervention and enable lock-step operation.

9.1.3.1 Synchronization between the VDOA and the IPU

The VDOA can transfer its output to the IPU through internal memory, containing a band double buffer.

This tight double-buffering synchronization is performed without CPU involvement, using dedicated signals between the VDOA and IPU and the same protocol as used between two IPU DMA channels.

This synchronization is supported for each of the IPUs, with muxing between the IPUs performed at SoC level.

Within the IPU, this synchronization is supported only for asynchronous flows .

9.1.4 Supported applications

The system described above supports a wide variety of video/graphics applications. The following table describes the main applications.

Table 9-1. Video/graphics applications

Application	Features
Display management	

Table continues on the next page...

Table 9-1. Video/graphics applications (continued)

Application	Features
Screen refresh	Up to two displays active simultaneously Pixel clock rate: • Single display - up to 264 MHz • Two displays (the sum of the two rates) - up to 240 MHz **NOTE:** The parallel interface works up to 200 MHz, and the HDMI interface works up to 240 MHz (IPU). Special features: • On-the-fly image conversion: rotation, inversion, resizing, color-space conversion and combining • On-the-fly image quality enhancement: color adjustment (including special effects) and gamut mapping, gamma correction and contrast stretching • Low-light compensation, allowing back-light reduction • Scrolling/panning
Updating the display buffer (from a background buffer)	Buffer either in system memory or in an external display controller (accessed through the display port) Special features: • On-the-fly processing - as for screen refresh • Optimized update - only modified parts are transferred • Synchronization with screen refresh, to prevent tearing • Scrolling (e.g. a running banner or a short animation)
Video	
Camera preview (displaying a view-finder window)	Input rate (from sensor): up to 240 MHz Additional features: • Window-of-interest • On-the-fly image conversion: de-interlacing, resizing, color space conversion, rotation, inversion • Combining with graphics • Synchronization, to prevent tearing
Still image capture	Rate: • Burst mode (off-line processing): up to 180 Mpixels/sec (while still leaving headroom for up to 35% blanking overhead) • Continuous: up to 160 Mpixels/sec Additional features: • Image quality enlacement before compression - as for camera preview • Image conversion before compression - as for camera preview; with independent parameters • Synchronization with a flash, a mechanical shutter and a mechanical iris
Motion Video Record	Resolution and rate: up to 1080p (1920x1080 at 48 fps in unit test without multimedia framework) Image processing before compression - as for still image capture
Motion Video Playback	Resolution and rate - up to 1080i/p (1920x1080) at 60 fps in unit test (without multimedia framework) Additional features: • Post-filtering: de-blocking and de-ringing

Table continues on the next page...

Table 9-1. Video/graphics applications (continued)

Application	Features
	• Image conversion after decompression: de-interlacing, rotation, inversion, resizing, color-space conversion and combining • Quality enhancement: color adjustment (including special effects) and gamut mapping, gamma correction and contrast stretching
Two-Way Video	Resolution and rate - up to 1080p (1920x1080) at 25 fps in unit test (without multimedia framework) Record and playback features - as above Various display options, including: • Two non-overlapping windows • Picture-in-picture

The following figure provides the sample processing flow of multimedia application.

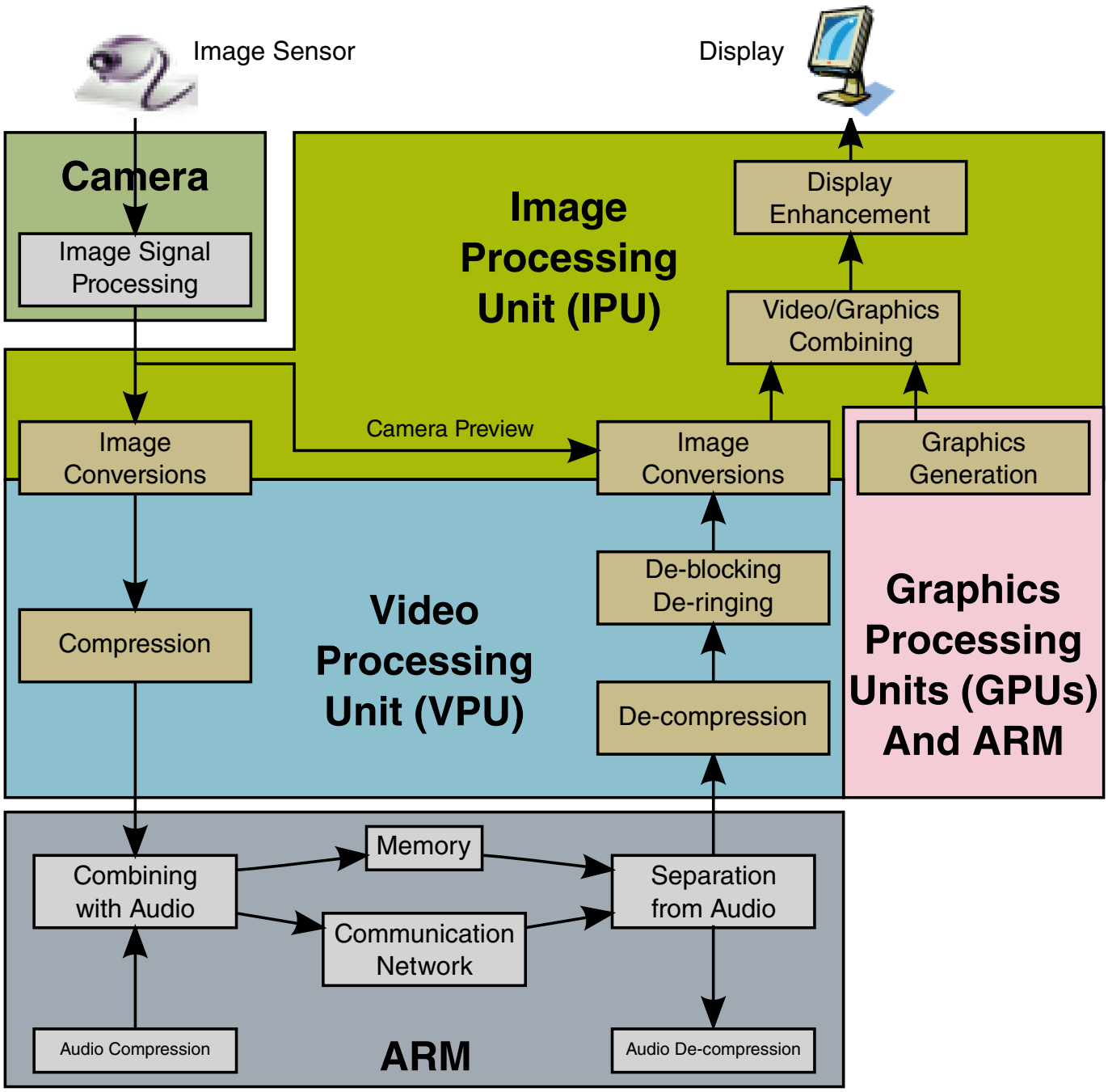


Figure 9-4. Processing flow for multimedia applications

- The camera preview mode-the upper part
- Video recording-the left part
- Video playback-the right part
- A video call-both sides

9.2 Image Processing Unit (IPU)

The following table shows the IPU IP parametric table.

Table 9-2. IPU IP Parametric Table

Name	IPU
Function	Connectivity to cameras and displays; related processing; synchronization and control
External I/O Pins Notes: This is the pinout of the IPU module At chip level, some of the pins are muxed and some are omitted. Additional GPIO pins are required to construct the connection. This is not included in this list.	Parallel Display port: 32 bit data, ~18 clocks and controls. Regular CMOS IO type, 264 MHz max. May be also connected to internal connectivity bridges. Slow Serial Display port: 2 bit data, 3 clocks and controls. Regular CMOS IO type, 120 MHz max. Parallel Sensor port: 20 bit data inputs, 5-6 clocks and controls. Regular CMOS IO type, 240 MHz max. Camera strobe port: 6 camera control outputs. Regular CMOS IO type, 120 MHz max.
SoC Buses	AXI master - for accessing the memory AHB slave - for programming, control and direct access of the MCU to the display
Interrupts	Two interrupts: functional and error
DMA Requests	Includes an integral DMA controller, with an AXI master port Also one DMA request to the SDMA
Number of instantiations	2
Clock sources and range	HSP_CLK - Internal high-speed processing clock: up to 264 MHz DI_CLK0, DI_CLK1 - Display interface clocks: up to 240 MHz

The goal of the IPU is to provide comprehensive support for the flow of data from an image sensor and/or to a display device. This support covers all aspects of these activities:

- Connectivity to relevant devices - cameras, displays, graphics coprocessors, TV encoders and decoders.

- Related image processing and manipulation: image enhancements and conversions, etc.
- Synchronization and control capabilities (to avoid tearing artifacts, for example)

This integrative approach leads to several significant advantages:

- Automation: The involvement of the MCU (Main Control Unit) in image management is minimized. In particular, display refresh/update and a camera preview (displaying the input from an image sensor) can be performed completely autonomously. The resulting benefits are reducing the overhead due to SW-HW synchronization, freeing the MCU to perform other tasks and reduced power consumption (when the MCU is idle and can be powered down).
- Optimal data path: Access to system memory is minimized. In particular, significant processing can be performed on-the-fly while receiving data from an image sensor and/or sending data to a display. System memory is used essentially only when a change in pixel order or frame rate is needed. The resulting benefits are reduced load on the system bus and further reduction of power consumption.
- Resource sharing: Maximal HW reuse for different applications, resulting with the support of a wide range of requirements with minimal hardware

The hardware reuse mentioned above is enabled by a sophisticated configurability of each hardware block. This configurability also allows the support of a wide range of external devices, data formats and operation modes.

A simplified block diagram of the IPU is given in the figure below. The role of each block is described in the table below.

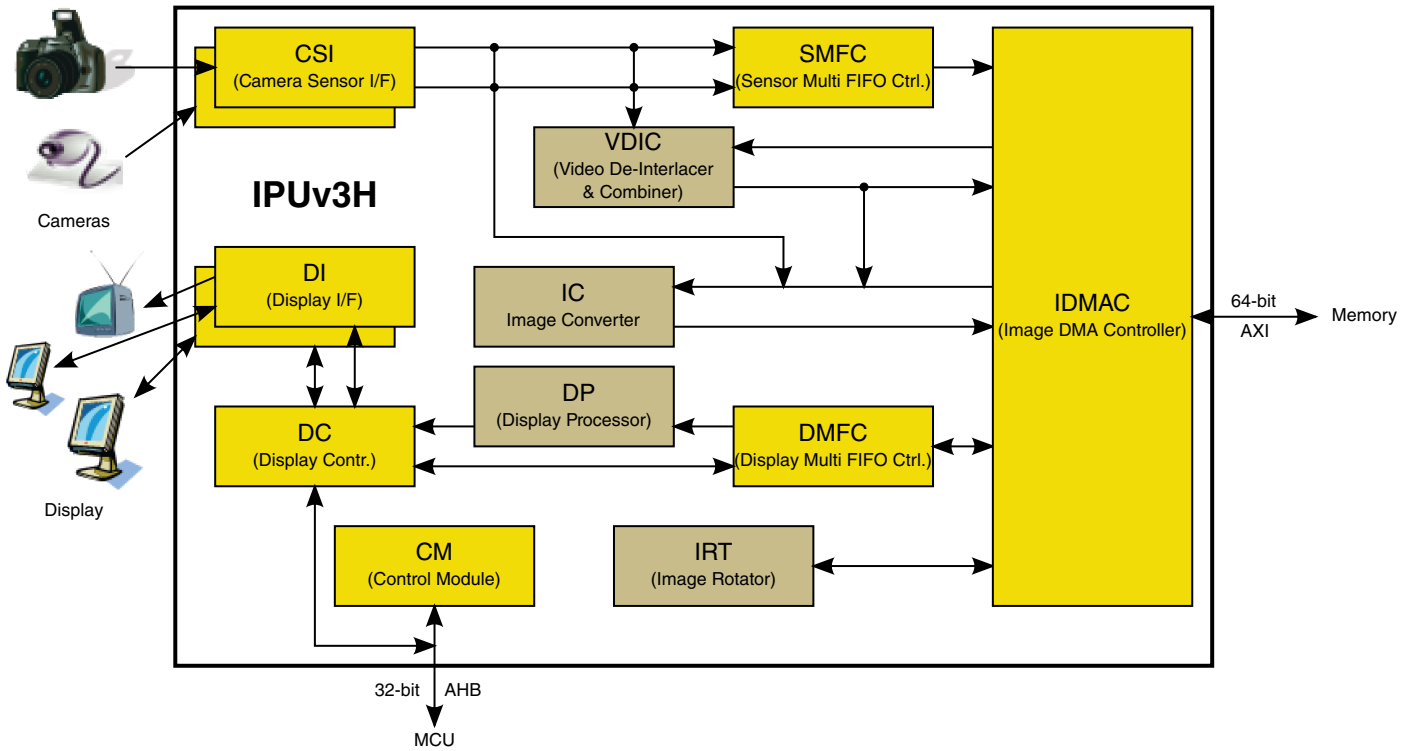


Figure 9-5. IPU - Block Diagram

Table 9-3. IPU - Block Description

Block	Description
CSI - Camera Sensor Interface	Controls a camera port; provides interface to an image sensor or a related device. IPUv3 includes 2 such blocks
DI - Display Interface	Provides interface to displays, display controllers and related devices. IPUv3 includes 2 such blocks
DC - Display Controller	Controls the display ports.
VDIC - Video De-Interlacer and Combiner	Performs de-interlacing - converting interlaced video to progressive - or combining
IC - Image Converter	Performs resizing, color conversion/correction, combining with graphics, and horizontal inversion
DP - Display Processor	Performs the processing required for data sent to display
IRT - Image Rotator	Performs rotation (90 or 180 degrees) and inversion (vertical/horizontal)
IDMAC - Image DMA Controller	Controls the memory port; transfers data to/from system memory
SMFC - Sensor Multi FIFO Controller	Controls FIFO's for output from the CSI's to system memory
DMFC - Display Multi FIFO Controller	Controls FIFO's for IDMAC channels related to the display system
CM - Control Module	Provides control and synchronization.

9.2.1 IPU External Ports

The IPU has the following ports:

- Two camera ports - controlled by a CSI module - providing a connection to image sensors and related devices.
- Two display ports - each controlled by a DI module - providing a connection to displays and related devices.
- Memory port - AXI (AHB V3.0) master, controlled by the IDMAC - providing connection to the system memory.
- AHB-lite slave port, providing connection to the ARM MCU (and to any other master connected to the ARM's cross-bar switch)

9.2.1.1 Camera Ports

The role of these ports is to receive input from video sources (e.g. image sensors) and to provide support for time-sensitive control signals to the camera. (Non-time-sensitive controls - e.g. configuration, reset - are performed by the MCU, through an I2C I/F or GPIO signals).

Each of the camera ports includes the following features:

- Direct connectivity to most relevant external devices.
- Parallel interface - up to 20-bit data bus
- Frame size: up to 8192 x 4096 pixels (including blanking intervals)
- Data formats supported include Raw (Bayer), RGB, YUV 4:4:4, YUV 4:2:2 and grayscale, up to 16 bits per value (component).
- Synchronization - video mode
  - The sensor is the master of the pixel clock (PIXCLK) and synchronization signals
  - Synchronization signals are received using either of the following methods:
    - Dedicated control signals -VSYNC, HSYNC - with programmable pulse width and polarity
    - Controls embedded in the data stream, following loosely the BT.656 protocol, with flexibility in code values and location.
- Synchronization - still image capture
  - The image capture is triggered by the MCU or by an external signal (e.g. a mechanical shutter)
  - Synchronized strobes are generated for up to 6 outputs - the sensor and camera peripherals (flash, mechanical shutter...)
- Additional features
  - Frame rate reduction, by periodic skipping of frames

- Downsizing x2, by skipping rows/columns
- Window-of-interest selection
- Pre-flash - for red-eye reduction and for measurements (e.g. focus) in low-light conditions

Several sensors can be connected to each of the CSI's. Simultaneous functionality (sending data) is supported as follows:

- Two streams can be received independently, each through a different CSI.
- A CSI can receive several interleaved streams (up to 4; e.g. as supported by the MIPI/CSI-2 I/F), The CSI can de-interleave the streams according to an ID signal received with the input.
- Unpacking capabilities are provided for a single stream in each CSI, while the other ones are treated as generic data.
- Only one of the streams can be transferred to the VDIC or IC for on-the-fly processing, while the other ones are sent directly to system memory.

The input rate supported by the camera port is as follows:

- Peak: up to 240 MHz (values/sec)
- Average (assuming 35% blanking overhead), for YUV 4:2:2
  - Pixel in one cycle (e.g. BT.1120): up to 180 MP/sec, e.g. 12M pixels @ 15 fps
  - Pixel on two cycles (e.g. BT.656): up to 90 MP/sec, e.g. 6M pixels @ 15 fps.
- On-the-fly processing may be restricted to a lower input rate - see below.

9.2.1.2 Display Ports

The role of these ports is to communicate with display devices, either directly or through a controller (e.g. graphics accelerator) or a bridge (e.g. TV encoder or an LVDS interface bridge).

9.2.1.2.1 Access Modes

Two access modes are supported:

- Synchronous Access
- Asynchronous Access

9.2.1.2.2 Synchronous Access

In this mode, the IPU transfers a two-dimensional block of pixels to the display device, in synchronization with the screen refresh cycle.

This mode has a dual role:

- For a RAM-less display or a TV screen, this mode is used to perform the screen refresh process from a display buffer in system memory.
- For a "smart" display, this mode is used to transfer a rectangular block of pixels to the display's screen and, in some cases, also to the display buffer

In both cases, the IPU sends to the display all the synchronization signals controlling the screen refresh and the block transfer is synchronized with these signals. This synchronization means that tearing effects are avoided when using this mode.

9.2.1.2.3 Asynchronous Access

This is the main mode used for communicating with an external display controller (possibly in a smart display or a graphics accelerator). In this mode, the IPU performs random access - read/write - to the memory and registers of the controller.

The following access types are provided:

- Data transfer to the external device, after on-the-fly processing in the IPU.
- Data transfer (DMA) - read/write - between the host's system memory and the external device, through the IPU's memory port (controlled by the IDMAC); e.g. transfer of a rectangular block of pixels (possibly full screen).
- Host access - read/write - to an external device, through the AHB-slave port

9.2.1.2.4 Interface details

The display interface is very flexible and supports a wide variety of devices from major manufacturers.

The following interface types are provided (in each of the two display ports):

- A parallel video interface (for synchronous access) - up to 24-bit data bus.
  - Supports BT.656 (8-bit) and BT.1120 (16-bit) protocols
  - Supports HDTV standards SMPTE274 (1080i/p) and SMPTE296 (720p)
- A parallel bidirectional bus interface (for asynchronous access) - up to 32-bit data bus.
- A serial interface - 3-wire, 4-wire and 5-wire (two flavors) (for asynchronous access)

The supported formats for pixel data are: RGB and YUV 4:2:2.

For the interface clock, there are the following options (independently for each port)

- Derived from the IPU internal clock (master mode)
- Provided by an external source (slave mode)

The transfer rate supported

- When a single port is active, the pixel clock rate is up to 264 MHz
- The both ports are active
  - Each pixel clock rate may be up to 220 MHz[1]
  - The sum of pixel clock rates is up to 240 MHz

9.2.1.2.5 Connecting To Display Devices

IPU allows the connectivity to multiple display devices.

In particular, it supports the following setup:

- Primary LCD display.
- Second LCD display.
- External display; e.g. TV.

Simultaneous functionality of the above devices is possible in each of the following ways:

- Two devices can be accessed (synchronously or asynchronously) independently, each through a different port: each using any of the available interfaces.
- Two devices can time-share asynchronous accesses through the legacy serial and parallel interfaces, using the CS signals.

9.2.2 Processing

The IPU processes rectangular blocks of pixels. The processing is performed in four modules - VDIC, IC, DP and IRT.

(see Figure 9-5 and Table 9-3). Several time-shared data flows are supported, as described in the following table.

Table 9-4. Time-Shared Data Flows Through The IPU

Name	Number	Type	Flow	Target	Restrictions
Display Refresh/ Update	5 flows (at most two of them of type DS1)	DS1	Fmem -> DP -> Display	Synchronous Access (e.g. display refresh; controlled by the DI)	
		DS2	Fmem -> DP -> Display	Asynchronous Access (e.g. display update)	

Table continues on the next page...

1. Specified pixel clocks frequencies are applicable for internal clocks, but may be limited by IO buffers speed capability. Final numbers are subjected to AC characterization.

Table 9-4. Time-Shared Data Flows Through The IPU (continued)

Name	Number	Type	Flow	Target	Restrictions
		DS3	Fmem <-> Display	Generic Data Transfer	
	1 flow	DS4	MCU <-> Display	Direct Access	
Video Playback	1 flow	PL1	Bmem -> VDIC -> IC -> Bmem -> IRT -> Fmem + DSx	Main option	
		PL2	Fmem -> IRT -> Bmem -> IC -> DP	Low power (branching to DSx, as a video plane)	Progressive source Large enough window No other video flows
		PL3	Fmem -> VDIC -> IC -> DP	Low power (branching to DSx, as a video plane)	Interlaced source Large enough window No other video flows
Camera Preview	2 flows (VF2 may be used also as a playback flow)	VF1	Sensor -> IC -> Bmem -> IRT -> Fmem + DSx	main option	Single progressive input
		VF2	Sensor -> Fmem -> VDIC -> IC -> Bmem -> IRT -> Fmem + DSx	two inputs and/or interlaced input	When the VDIC is used for de-interlacing, one of the three input fields can go directly from the sensor to the VDIC
		VF4	Sensor -> IC -> Fmem + DS1	Low power RAM-less Display Single Display Buffer (in internal memory) Tearing-less	Single progressive input Refresh rate = 2x sensor frame rate Large enough window No other video flows
Video Record	2 flows	RCx	IC -> Bmem -> IRT -> Fmem	(branching from VFx)	
Graphic Overlays	2 flows	GF1	Fmem -> IC	(combining with the main flow	
	2 flows	GF2	Fmem -> DP		

9.2.2.1 Display Processor (DP)

The Display Processor performs all the processing required for data sent to a display.

- Combining 2 video/graphics planes
- Overlaying a simple HW cursor 32 x 32 pixels, uniform color; may be combined logically with the background.
- Color conversion/correction - linear (multiplicative and additive) Programmable; including:

    - YUV <-> RGB, YUV<->YUV conversions where YUV stands for any one of the color formats defined in the MPEG-4 standard
    - Adjustments: brightness, contrast, color saturation...
    - Special effects: gray-scale, color inversion, sephia, blue-tone...
    - Hue-preserving gamut mapping - for minimal color distortion
    - Applied to the output of combining or to one of the inputs
- Gamma correction and contrast stretching - programmable piecewise-linear map

The DP processes a single data flow at any given time, but it supports up to three data flows, by time sharing, one of them may be synchronous.

The data rate is up to 264M pixels/sec

9.2.2.2 Video de-interlacer (VDIC)

VDIC, the video de-interlacer and combiner module, has two operation modes

- De-interlacing: converts an interlaced video stream to progressive order.
- Combining: combines two video/graphics planes and a background color

The input and output to/from the VDIC is as follows:

- Input for de-interlacing: three consecutive fields
    - Source
        - The most recent field may come from the CSI or from system memory
        - The other two fields are read from memory
    - Field size: Supports up to 1080p
    - Pixel format: YUV 4:2:2/4:2:0, 8 bits/value
    - Typical video sources - SDTV: 480i30 (720x480 at 30 fps) or 576i25 (720x576 at 25 fps) and HDTV: 1080i30 (1920x1080 at 30 fps)
- Input for combining: two progressive video/graphics planes
    - Source: system memory
    - Plane size: up to 1920x1200 pixels.
    - Pixel format: RGB/YUV 4:2:2, 8 bits/value
- Output: progressive frame
    - Destination: to system memory or to the Image Converter.
    - Frame size: up to 1920x1200 pixels.
    - Rate: up to 240M pixels/sec.
    - Format: same as input format.

De-interlacing is performed using a high-quality 3-field filter, which is motion adaptive:

- For slow motion - retains the full resolution (of both top and bottom fields)
- For fast motion - prevents motion artifacts

VDIC supports a single video stream at any given time.

9.2.2.3 Image Converter (IC)

The Image Converter performs various operations on a video stream.

The operations performed are:

- Resizing:
  - Fully flexible resizing ratio Maximal downsizing ratio: 8:1. Subject to this limitation, any N->M resizing can be performed
  - Independent horizontal and vertical resizing ratios.
- Color conversion/correction - linear (multiplicative and additive) Programmable; including:
  - YUV <-> RGB, YUV<->YUV conversions where YUV stands for any one of the color formats defined in the MPEG-4 standard
  - Adjustments: brightness, contrast, color saturation...
  - Special effects: gray-scale, color inversion, sephia, blue-tone
- Combining with a graphics plane (e.g. application-specific overlay)
- Horizontal inversion

The IC supports three time-shared data flows: record, camera preview and playback (the first two share a common input).

The frame resolution supported is up to 4096x4096 for input and up to 1024x1024 for output. Wider frames can be processed by the IC by splitting them to vertical stripes.

Data throughput:

- For a single active task: up to 240M pixels/sec for input and up to 120M pixels/sec for output.
- For several active tasks: up to 200M pixels/sec for input and up to 100M pixels/sec for output.

9.2.2.4 Image Rotator (IRT)

The Image Rotator performs any combination of the following:

- 90-degree rotation
- Horizontal inversion
- Vertical inversion

The data throughput

- When a single task is active: up to 120M pixels/sec.
- When more tasks are active: up to 100M pixels/sec.

9.2.3 Automatic Procedures

The IPU is equipped with powerful control and synchronization capabilities to perform its tasks with minimal involvement of ARM and minimal use of memory.

In particular, it includes:

- An integrated DMA controller with an AXI master port, allowing autonomous access to the system memory.
- An integrated display controller, performing screen refresh of a RAM-less display.
- A page-flip double buffering mechanism, synchronizing read and write access to system memory, to prevent tearing effects.
- A double/triple buffer synchronization mechanism with a video/graphics source.
- Internal synchronization, e.g., between input from sensor and output to display.

As a result, in most cases, the MCU is involved only when it also performs part of the processing (e.g. video coding). In particular, the following procedures are performed by the IPU completely autonomously:

- Screen refresh for RAM-less displays
- Camera preview (displaying a view finder - a video-stream from the image sensor to the display).

NOTE

Direct camera to display flow is possible when the frame rate of the source and the destination is the same (typically the target display will be asynchronous display, where the display is updated at the rate of the source).

Typically, there are extended periods of time in which there is no other activity in the system. The MCU - being idle - can be put to a low-power mode, reducing the power consumption and extending significantly the battery life.

The IPU supports several techniques to reduce further the power consumption of the display system:

- Dynamic backlight control, with low-light compensation by image enhancement

Further features and capabilities of the automatic procedures include:

- Automatic display of a changing image (animation) or moving image (scrolling).

- The timing of the display update can be adjusted to avoid tearing.
- The video stream from an image sensor can be sent directly to the display buffer used for screen refresh, while avoiding tearing.

The clock sources received by the IPU are listed in the following table.

Table 9-5. IPU Clock Sources

Name	Symbol	Source	Rate	Comments
High-Speed Processing Clock	HSP_CLK	Clock control Module	Up to 264 MHz	
Display Interface Clocks	DI_CLK0 DI_CLK1	Clock control Module or an external PLL	Up to 220 MHz	Optional; needed for synchronization with interface bridges

9.2.4 Further Changes in IPUv3H vs. IPUv3M

- The sizes of the input/output FIFOs in the DMFC, SMFC and VDIC have been increased, to support an increased data rate
- The memory address space has been reduced to 32 MB. Direct access to an external display device is canceled, but low-level access is retained.
- Synchronization with the VDOA has been added - see Synchronization between the VDOA and the IPU

9.3 LVDS Display Bridge (LDB)

LVDS Display Bridge (LDB) will be used to connect the IPU (Image Processing Unit) to External LVDS Display Interface.

9.3.1 LDB Overview

Table 9-6. IP Parametric Table

Name	IPUv3EX
Function	Connectivity to displays with LVDS interface
External I/O Pins Notes: Those are LVDS IO pads	LVDS Display port: 2 channels. Each channel consists of: • 1 clock pair • 4 data pairs Each pair contains - LVDS special differential pad (PadP, PadM).

Table continues on the next page...

Table 9-6. IP Parametric Table (continued)

Name	IPUv3EX
	total of 20 IO pads.
SoC Buses	None. Only configuration signals.
Interrupts	None
DMA Requests	None
Number of instantiations	1
Clock sources and range	DI0_CLK, DI1_CLK- Display interface clock: 20-170 MHz DI0_SER_CLK, DI1_SER_CLK - Serializer clock: 140-595 MHz

The purpose of the LDB is to support flow of synchronous RGB data from the IPU to external display devices through LVDS interface. This support covers all aspects of these activities:

- Connectivity to relevant devices - Displays with LVDS receivers.
- Arranging the data as required by the external display receiver and by LVDS display standards.
- Synchronization and control capabilities.

9.3.2 LDB External Ports

The LDB has the following ports:

- Two input parallel display ports.
- Two Output LVDS channels - Each channel consisting of 4 data pair, and 1 clock pair (pair=LVDS pad contains PadP, PadM).
- Control signals - to configure LDB parameters and operations.
- Clocks from SOC PLLs.

9.3.3 Input Parallel Display Ports

One or Two (DI0, DI1) parallel RGB input ports are supported (configurable). Only synchronous access mode is supported.

Each RGB data interface contains the following:

- RGB Data of 18 or 24 bits
- Pixel clock
- Control signals: HSYNC, VSYNC, DE, and 1 additional optional general purpose control.

Total of up to 28 bits per data interface are transferred per pixel clock cycle.

Rates supported:

- For dual-channel output: Up to 170 MHz pixel clock (e.g. UXGA - 1600x1200 @ 60 Hz + 35% blanking)
- For single-channel output: Up to 85 MHz per interface. (e.g. WXGA - 1366x768 @ 60 Hz + 35% blanking).

9.3.3.1 Output LVDS Ports

There are 2 LVDS channels. These inputs are used to communicate RGB data and controls to external LCD displays with LVDS interface, or through LVDS receivers.

The LVDS ports may be used as follows:

- One single-channel output
- One dual channel output: single input, split to two output channels
- Two identical outputs: single input sent to both output channels
- Two independent outputs: two inputs sent, each, to a different output channel

9.4 Video Data Order Adapter (VDOA)

The goal of the VDOA is to re-order video data from the "tiled" order used by the VPU to the conventional raster-scan order needed by the IPU.

9.4.1 VDOA Interfaces

The VDOA has the following interfaces

- Memory interface: 64-bit AXI master port (no AXI ID output)
- Host interface: IP slave port
- Synchronization with IPU (see below)

9.4.2 VDOA Data Path

9.4.2.1 Input

- Source: a frame buffer in system memory

- Addressing: tile-based
- Order: raster-scan of tiles
- Maximal bursts (up to 64 bytes)

9.4.2.2 Output

- Destination: a frame buffer or band buffer in system memory (band = row of tiles; to be used when the buffering is in internal memory)
- Addressing: raster-scan of rows
- Order: full bursts of 64 bytes (requires data from several tiles)

Input/output - additional aspects

- Pixel format: YUV 4:2:0 partially-interleaved
- Pixel format: YUV 4:2:2 interleaved
- Total rate: at least up to 220 MP/sec (e.g. 1080i @ 60 fps + 480i @ 60 fps, three fields per frame)
- Frame width: even; up to 4096 pixels
- Frame height: even; up to 2048 pixels
- Addresses [bytes]
  - Base address: 32-bit integer, multiple of 8
  - Offset from the Y buffer to the UV buffer: 23-bit integer, multiple of 8
  - Line stride (same for both buffers): 12-bit integer, multiple of 8

9.4.3 Control

Each VDOA task is activated individually by the CPU and consists of

- Reordering a single buffer of pixels;

  or

- Reordering concurrently three buffers or pixels (needed for the 3 input fields used for de-interlacing in the IPU).

When reordering concurrently three buffers

- The buffers differ only in their base addresses.
- The switching between them is performed at the end of a band (= row of tiles), in a fixed order: buffer 0 band 0 -> buffer 1 band 0 -> buffer 2 band 0 -> buffer 0 band 1...

Multi-tasking

- All task parameters are double buffered, to allow SW-controlled frame-by-frame task-switching without re-configuration overhead.

Synchronization with IPU - see Synchronization between the VDOA and the IPU

End-of-task interrupt

- The VDOA can issue a functional interrupt after completing the task.
- The interrupt is triggered by the acknowledgement received on the AXI bus for the last write access.
- For this to indicate that the data indeed reached its destination, this last write access is non-bufferable.

9.5 Display Content Integrity Checker (DCIC)

The goal of the DCIC is to verify that a safety-critical information sent to a display is not corrupted.

Such a verification is mandatory for warning icons in the instrument cluster of a car, to comply with the ASIL B (Automotive Safety Integrity Level B) specification. It is also required in other safety-sensitive systems.

9.5.1 DCIC Interfaces

The DCIC has the following interfaces

- Data input port - snooping one of the display ports
- Host interface: IP slave port

9.5.2 DCIC Data Path

See Figure 9-2 for DCIC integration in i.MX 6Dual/6Quad SoC.

Input - snooped display bus

- Pixel clock signal; up to 264 MHz
- Data enable signal: marking a cycle with valid data (pixel)
- Data: 24-bit bus, assumed to contain a single full pixel
- Vsync & Hsync signals: indicating frame & row boundaries respectively
- Mask signal (see below)

Actions

- Identify pixels belonging to a ROI (Region Of Interest)
  - Up to 16 ROIs are supported
  - Each is confined to a rectangle, with a configurable location and size
  - A refined shape of the ROI is characterized through the mask input signal, when enabled
  - Overlap between the regions is not allowed (each pixel contributes to at most one region)
- Calculate an integrity signature - CRC32 - independently for each ROI
- Compare the calculated signature to the reference signature - provided by the host CPU
- This check is triggered by the Vsync signal

Results

- Registered values - accessible by the CPU through the host interface
  - Calculated signatures - one for each ROI - from the last frame (the intermediate values from the current frame are stored separately)
  - Match status bits - one for each ROI - from the last frame
- Interrupts - to the host CPU (maskable)
  - Match results ready (functional)
  - Mismatch (error)
  - Both interrupts are generated immediately after completing the signature match check
- Signal to an external CPU (through a GPIO pin)
  - Continuously oscillating at a rate which is an integer division of the input clock
  - When the status bit of the mismatch interrupt is set - i.e. from mismatch detection until the CPU clears the bit: division x16
  - Otherwise: division x4

9.5.3 Configuration parameters

- For each of the 16 ROIs
  - Enable bit; double-buffered; changes take effect at frame boundaries.
  - Rectangle offset [pixels] = location of the upper-left pixel
    - Horizontal: 0 .. 2^13-1
    - Vertical: 0 .. 2^12-1
  - Rectangle size [pixels]
    - Horizontal: 1 .. 2^13
    - Vertical: 1.. 2^12
  - A 32-bit "reference signature" - to be compared with the calculated signature
  - Freeze bit

    - "Sticky" control bit: zero at reset; can be set but cannot be reset
    - Once set (typically at boot), the only parameters from the above list that are allowed to be changed are the reference signatures
- Mask input settings: enable, polarity
- Interrupt settings
    - A mask bit for each of the two interrupts
    - Freeze bit: the masks can be changed only before the freeze bit is set.

9.5.4 System Considerations

- The DCIC always assumes a 24-bit pixel.
- For proper functionality with lower color depth, one must ensure that:
    - When the CPU calculates the reference signature, it applies the same mapping of a pixel to a 24-bit field as that performed by the IPU.
    - The display is connected to the appropriate pins.
    - The simplest mapping would be
        - Map the pixel to the data bus in the same way as for 24 bpp
        - Set the values at the extra LSBs to zero.
- Parameter updates
    - The reference signature can be freely updated from frame to frame, in coordination with the changing content (since it is used only once per frame, just before the interrupt)
    - Each ROI can be freely and independently enabled/disabled between frames (since the enable bit is double buffered)
    - The size and location of a ROI can be modified while it is disabled.

9.6 Video Processing Unit (VPUv6)

The VPU is a multi-standard video codec (encoder/decoder) capable of handling multiple streams simultaneously by time multiplexing.

It is a very flexible block consisting of hardware accelerators surrounding a programmable core. The VPU presents to the system a register mapped interface that is controlled by the embedded processor. End users should only interface with the VPU using the API that is provided with each BSP Programmers User Guide. This API isolates the user from possible changes in the register level interface.

In the following table we can see a summary of the VPU specs. The VPU has its own DMA driven AXI masters that allow it to retrieve the required data directly from system memory (DDR and iRAM). The load in the host ARM platform is negligible because it only needs to interact with the VPU at the frame level.

Table 9-7. VPU Spec Summary

Name	VPU
Function	Decode video streams including optional video processing such as rotation, deringing and mirroring.
Supported encoders	MPEG-4 SP H.263 V2 + Annex J, K (RS=0 and ASO=0), and T H.264 BP, CBP MJPEG Baseline
Supported decoders	MPEG-2 MP, HP VC-1 SP, MP, AP MPEG-4 SP, ASP H.263 V2 + Annex J, K (RS=0 and ASO=0), and T H.264 BP, MP, HP H.264-MVC BP, MP, HP DivX v3,4,5 Real Video 8, 9, 10 MJPEG Baseline On2 VP8 AVS Jizhun
External I/O Pins (List, Type, Schmidt Trigger, Speed)	No external I/O pins are needed
SoC Buses (List, Type, Bandwidth)	64 bit AXI master for accessing the system memory and the optional secondary AXI but to iRAM IPBus slave for host control
Interrupts	Two interrupts
DMA Requests	Integrated DMA controller on the AXI master port
Endianness	64 and 32 bit BE/LE
Number of instantiations	1
Clock sources and range	Core clock: up to 264 MHz AXI bus clock: up to 264 MHz IP bus clock: up to 66 MHz

NOTE

RealNetworks video codec is disabled by default on i.MX 6 series processors. Please contact your FSL sales representative for more details.

The i.MX 6Dual/6Quad processor has a high-performance video processing unit (VPU), which covers many standard and high definition video decoders and encoders as a multi-standard video codec engine as well as several important video processing such as rotation and mirroring.

9.6.1 Basic Structure

The following Figure 9-6 shows the basic structure of the VPU. The VPU is self contained except for memory accesses. It is connected to the memory subsystem through two AXI master ports. The ARM configures the VPU operation through the IP bus that is converted to APB in a gasket.

A new feature is added for allowing VPU encoding a frame even the IPU writes only a small portion of source video (from camera sensor) into the frame buffer. This feature is called IPU-VPU sub-frame synchronization. Four additional signals are added for implementing this feature with three of them from IPU to VPU, and one of them from VPU to IPU, as shown in Figure 9-6 for the signal category of "IPU VPU sub-frame sync"

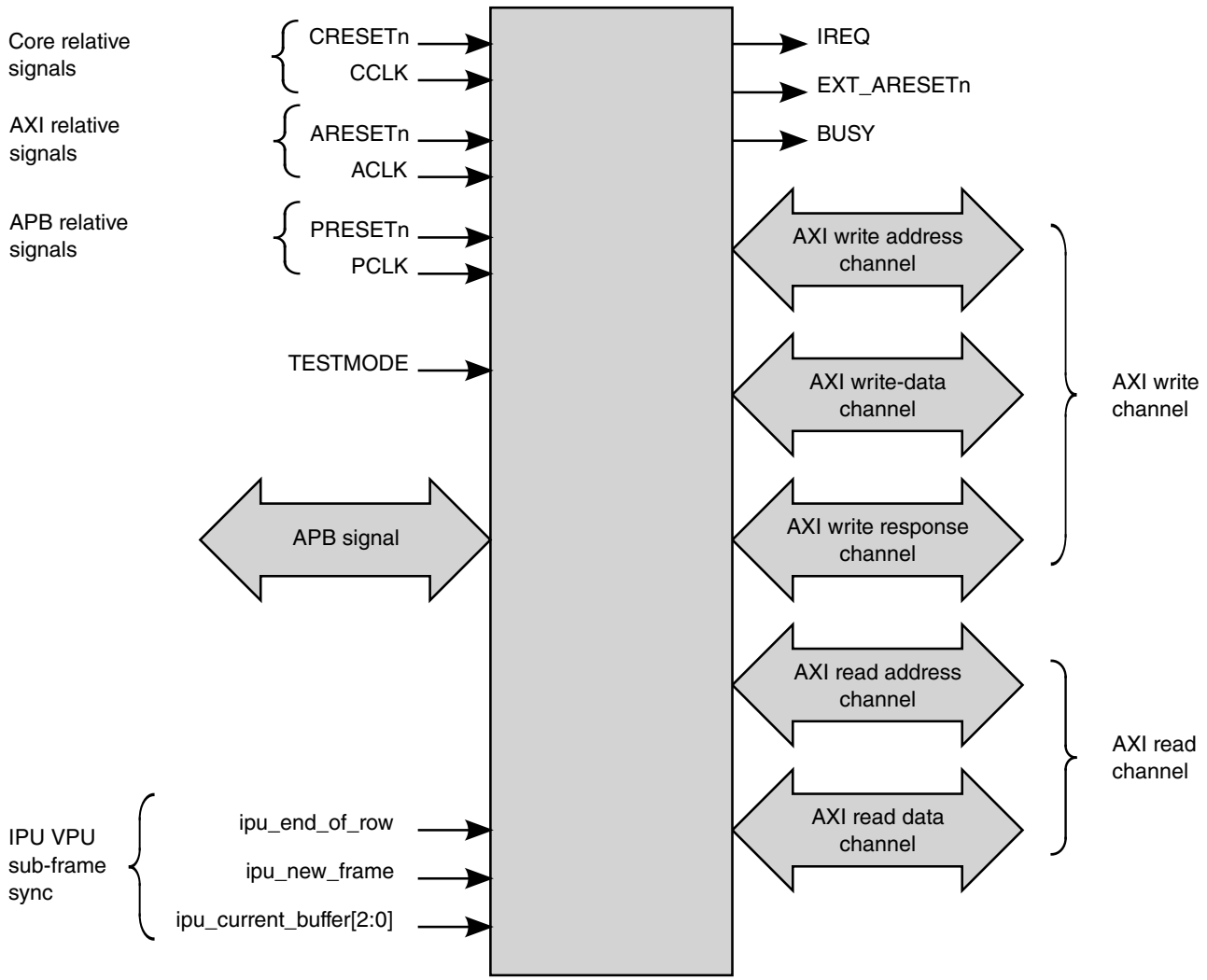


Figure 9-6. The Block Diagram of the VPU

9.6.2 Feature summary

The following table describes the VPU's encoding / decoding capabilities:

Table 9-8. VPU decoding/encoding capabilities

Dec/Enc	Standard	Profile	Resolution	Bitrate	Comments
HW Decoder	MPEG-2	Main-High	1080 i/p, 30fps	50 Mbps	1080p+SD at 30fps, 720p60
	MPEG4/XviD	SP/ASP	1080 i/p, 30fps	40 Mbps	—
	H.263	P0/P3	16CIF, 30 fps	20 Mbps	—
	H.264	BP/CBP/MP/HP	1080 i/p, 30 fps	50 Mbps	1080p+SD at 30 fps, 720p60
	H.264-MVC	BP/MP/HP	720p, 30 fps	—	—

Table continues on the next page...

Table 9-8. VPU decoding/encoding capabilities (continued)

Dec/Enc	Standard	Profile	Resolution	Bitrate	Comments
	VC1	SP/MP/AP	1080 i/p, 30 fps	45 Mbps	1080p+SD at 30 fps, 720p60
	RV	8/9/10	1080 p, 30 fps	40 Mbps	—
	DivX	3/4/5/6	1080 i/p, 30 fps	40 Mbps	—
	On2 VP6	—	720p, 30 fps	20 Mbps	—
	On2 VP8	—	720p, 30 fps	20 Mbps	—
	Theora	—	720p, 30 fps	20 Mbps	—
AVS	Jizhun	1080 i/p, 30fp	40 Mbps	—	—
MJPEG	Baseline	8192x8192	120 Mpixel/sec	Perf shown at 4:4:4 format	—
Hardware encoder	MPEG4	Simple	720p, 30 fps	12 Mbps	VPU can generate higher bitrate than the maximum specified by the corresponding standard.
	H.263	P0/P3	4CIF, 30 fps	8 Mbps	
	H.264	BP/CBP	1080p, 30 fps	14 Mbps	
	MJPEG	Baseline	8192 x 8192	160 Mpixel/sec	Perf shown at 4:2:2 format

VPU can also perform the following:

- On-the-fly (90 x n) degree simultaneous rotation and mirroring (n = 0,1,2,3).
- Post-processing
  - De-blocking filtering for MPEG-4
  - De-ringing filtering for MPEG-4 and H.264 decoder

9.7 OpenGL ES 3D Graphics Processing Unit (GPU3Dv4)

9.7.1 OpenGL Overview

9.7.2 OpenGL Features

Summary of features in the GPU3D includes:

- OpenGL ES 2.0 compliance, including extensions; OpenGL ES 1.1; OpenVG 1.1
- IEEE 32-bit floating-point pipeline
- Ultra-threaded, unified vertex and fragment shaders
- Low bandwidth at both high and low data rates

- Low CPU loading
- Up to 12 programmable elements per vertex
- Dependent texture operation with high-performance
- Alpha blending
- Depth and stencil compare
- Support for 8 fragment shader simultaneous textures
- Support for 4 vertex shader simultaneous textures
- Point sampling, bi-linear sampling, tri-linear filtering, and cubic textures
- Resolve and fast clear
- 8k x 8k texture size and 8k x 8k rendering target

9.7.3 OpenGL Block Diagram

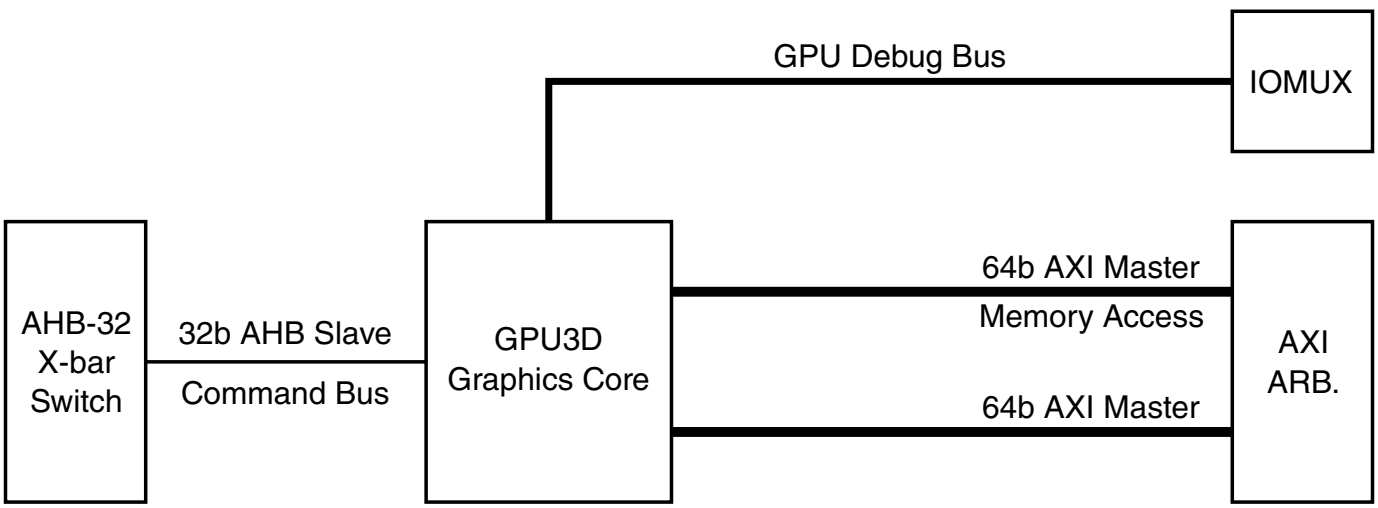


Figure 9-7. GPU System Integration Diagram

9.7.4 OpenGL Performance

- Geometry Rate: 88M Triangles/sec
- Pixel Rate: 1.066G pixels/sec

9.7.5 OpenGL Software

API / Driver Support

- OpenGL ES 1.1, 2.0
- OpenVG 1.1
- EGL 1.4
- OpenGL 2.1
- OpenCL 1.1 EP
- OpenVG 1.1

Operating Systems

- Windows CE
- Linux Embedded and X11
- Android

9.8 2D Graphics Processing Unit (GPU2Dv2)

9.8.1 2D feature summary

GPU2D has the following features:

- Bit BLT & stretch BLT
- Rectangle fill and clear
- Line drawing
- High performance stretch and shrink
- Mono expansion for text rendering
- ROP2, ROP3, and ROP4
- Alpha blending including Java 2 Porter-Duff compositing blending rules
- Support rendering size of 32Kx32K
- 90/180/270 degree rotation
- Transparency by monochrome mask, chroma key, or pattern mask
- Color space conversion between YUV and RGB
- High quality image scaling, using up to 9x9 separable filter
- Bit-Blit Formats
    - A1R5G5B5 (source/destination)
    - A4R4G4B4 (source/destination)
    - X1R5G5B5 (source/destination)
    - X4R4G4B4 (source/destination)
    - R5G6B5 (source/destination)
    - X8R8G8B8 (source/destination)
    - A8R8G8B8 (source/destination)

- 8-bit color index (source only)
- A8 (source/destination)
- 1-bit monochrome (source only)

- Filter Blit Formats
  - A1R5G5B5 (source/destination)
  - A4R4G4B4 (source/destination)
  - A8R8G8B8 (source/destination)
  - R5G6B5 (source/destination)
  - X1R5G5B5 (source/destination)
  - X4R4G4B4 (source/destination)
  - X8R8G8B8 (source/destination)
  - YUV (source only):
    - NV12 (4:2:0, 2 planes)
    - NV16 (4:2:2, 2 planes)
    - UYVY (4:2:2, interleave)
    - YUY2 (4:2:2, interleave)
    - YV12 (4:2:0, 3 planes)
  - 8-bit color index(source only)

9.8.2 2D Block Diagram

Figure 9-8 below presents the integration of the GPU2D core in i.MX 6Dual/6Quad SoC.

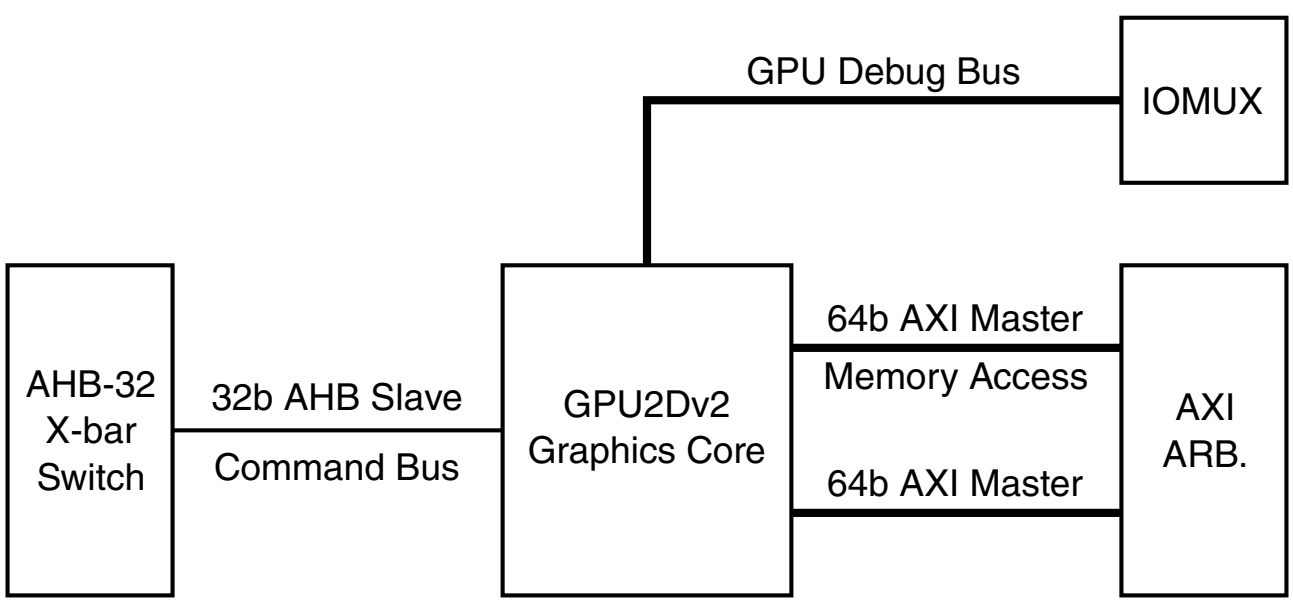


Figure 9-8. GPU2Dv2 (GPU2D) Interface Diagram

9.8.3 2D Performance

- Geometry Rate: 26.6M Triangles/sec
- Pixel Rate: 256M pixels/sec

9.8.4 2D Software

API / Driver Support

- GDI/DirectDraw
- DirectFB
- X11 EXA

Operating Systems

- Windows CE
- Linux Embedded and X11
- Android

9.9 Vector Graphics Processing Unit (GPUVGv2)

9.9.1 Vector Graphics Overview

The vector graphics processing unit (version 2) is based on the GC355 IP core.

9.9.2 Vector Graphics Features

Featureset Overview:

- Real-time hardware curve tesselation of lines, quadratic and cubic Bezier curves
- 16x Line Anti-aliasing
- OpenVG 1.1 support
- Vector Drawing:
  - Coordinate Systems and Transformations
  - Viewport Clipping, Scissoring and Alpha Masking
  - Pathsand stroke generation
  - Image interpolation
  - Image Filters

- Paint (gradient and pattern)
- Blending

Figure 9-9 below presents the integration of OpenVG core, in i.MX 6Dual/6Quad SoC.

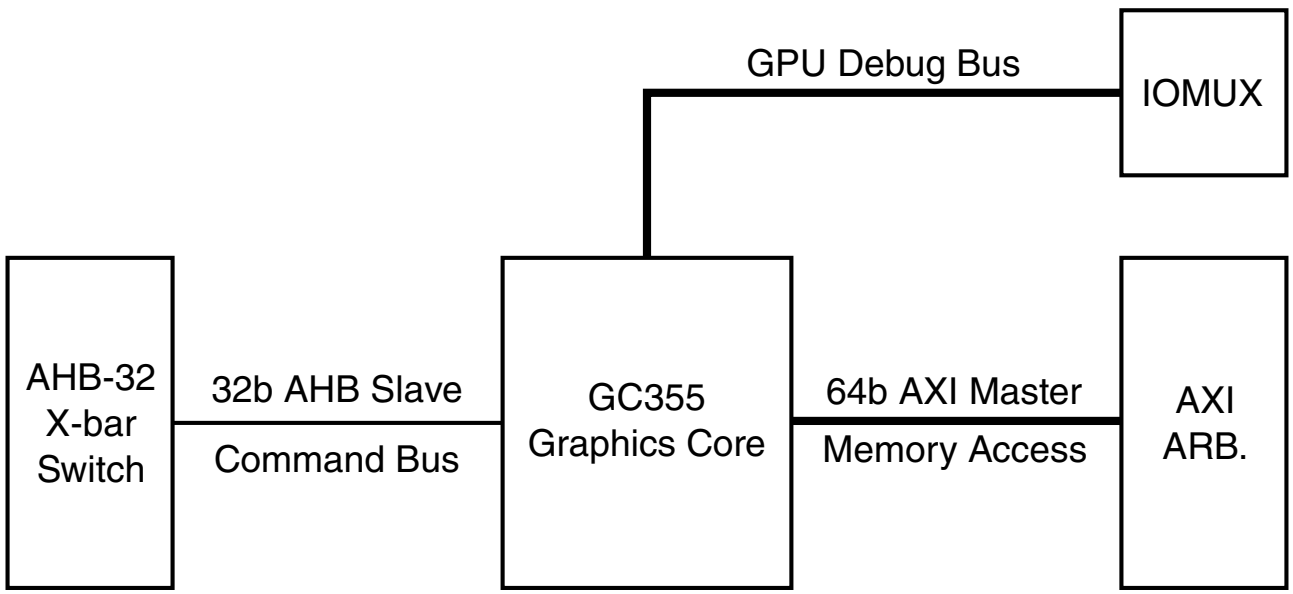


Figure 9-9. GPU2Dv2 (GC355) Interface Diagram

9.9.3 Vector Graphics Performance

- GPU-VG @ 533MHz
- Full OpenVG 1.1 Khronos Conformance
- 533 M pixels / sec raw performance (1 pixel / clock)

9.9.4 Vector Graphics Software

API / Driver Support

- OpenVG 1.1

Operating Systems

- Windows CE
- Linux Embedded and X11

9.10 HDMI TX - HD Multimedia Interface transmitter

9.10.1 HDMI Introduction

HDMI (High-Definition Multimedia Interface) is a compact audio/video interface for transmitting uncompressed digital video data and uncompressed/compressed digital audio data. HDMI connects digital audio/video sources-such as set-top boxes, Blu-ray Disc players, personal computers (PCs), video game consoles, and AV receivers to compatible digital audio devices, computer monitors, and digital televisions.

HDMI supports, on a single cable, any TV or PC video format, including standard, enhanced, and high-definition video, up to 8 channels of digital audio, and a Consumer Electronics Control (CEC) connection. The CEC allows HDMI devices to control each other when necessary and allows the user to operate multiple devices with one remote control handset.

Because HDMI is electrically compatible with the signals used by Digital Visual Interface (DVI), no signal conversion is necessary, nor is there a loss of video quality when a DVI-to-HDMI adapter is used.

The HDMI Transmitter (HDMI TX) consists of two parts:

- HDMI TX Controller
- HDMI TX PHY

Figure 9-10 depicts the HDMI TX integration scheme into the i.MX 6Dual/6Quad.

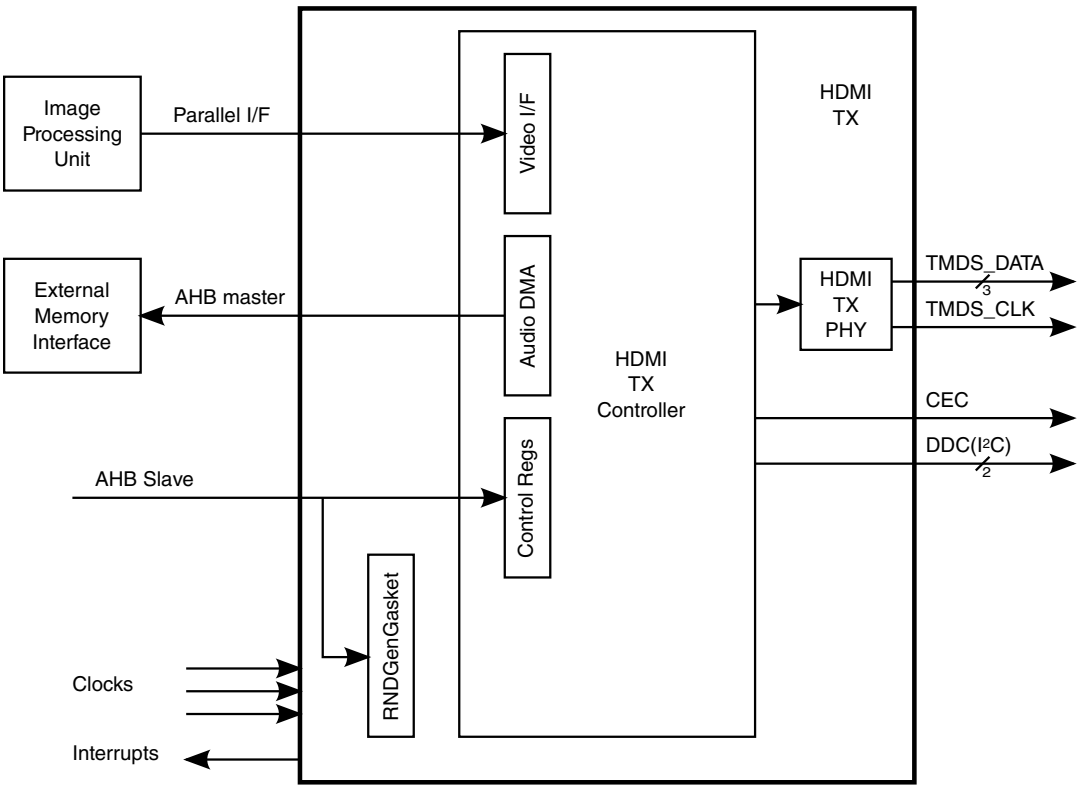


Figure 9-10. HDMI TX integration into the i.MX 6Dual/6Quad

The video interface is parallel with 24 bit/pixel data and separate signals for vertical (VSYNC) and horizontal (HSYNC) synchronization. The video data is supplied by the Image Processing Unit (IPU).

The audio data is read from an external memory by the internal Audio DMA via the master AHB interface. The HDMI TX is controlled via the AHB interface.

9.10.2 Features

The HDMI TX features are listed in Table 9-9.

Table 9-9. HDMI TX Features

Specification item	Requirement
General Features	
Standard Compliance	HDMI 1.4a

Table continues on the next page...

Table 9-9. HDMI TX Features (continued)

	HDMI CTS 1.4a DVI 1.0 HDCP 1.4
Consumer Electronic Control	Supported
Monitor Detection	Hot plug/unplug detection and link status monitor supported
Video Features	
Video Standard Compliance	EIA/CEA-861D
Supported Video Resolutions	Up to 1080p@120Hz HDTV display, up to QXGA graphics display, HDMI 1.4a 4K x 2K video formats HDMI 1.4a 3D video modes with up to 340MHz TMDS clock
Pixel Clock Frequency	From 25 MHz to 340 MHz
Input Data Formats	Parallel YCbCr 4:4:4 and parallel RGB 4:4:4 and parallel YCbCr 4:2:2
Input Color Depth	24/30/36/48 bits/pixel
Input Syncs Format	Separate HSYNC and VSYNC plus data enable control
Internal Video Processing	Interpolation YCbCr 4:2:2 to 4:4:4 Color space conversion YCbCr to RGB and vice versa
Audio Features	
Audio Standard Compliance	IEC60958, IEC61937
Supported Audio Formats	All audio formats as specified by the HDMI Specification Version 1.4a supported.
Audio Input Interfaces	Audio DMA
Audio Sampling Rate	Up to 192 kHz

9.11 Display / Sensor MIPI interfaces

9.11.1 Introduction

i.MX 6Dual/6Quad Application Processor features MIPI DSI/CSI-2 interfaces that includes:

- MIPI DSI Host controller with associated MIPI D-PHY Tx (One clock lane, two Data lanes)
- MIPI CSI-2 Host controller with associated MIPI D-PHY Rx (one clock lane, four Data Lanes)

9.11.2 DSI

DSI is a high performance serial interconnect bus for mobile applications connecting display system to the host system.

Table 9-10. DSI Parametric Table

Name	DSI
Function	High speed serial interface controller for MIPI Display interface
Supported standard version	MIPI DSI Compliant DSI Version 1.01 DPI Version 2.0 DBI Version 2.0 DCS Version 1.02 PPI for D-PHY MIPI D-PHY Version 1.0
External I/O Pins (List, Type, Schmidt Trigger, Speed)	Analog at chip boundary: DATAP[1:0] DATAN[1:0] CLKP CLKN Supplies AVDD -2.5V general analog supply AVDDREF - 2.5V analog supply for D-PHY reference source (low noise) should be directly connected to package ball VDD - 1.1V digital supply GND
SoC Buses (List, Type, Bandwidth)	Synchronous Pixel Interface (DPI) Display Bus Interface (DBI) IPB slave interface x 32 bit PPI to D-PHY
Interrupts	1
DMA Requests	N/A
Num of instantiations	1
Clock sources and range	AHB clock (132MHz) Reference clock for Tx D-PHY - 6 ... 27 MHz

There is one instance of DSI port in the i.MX 6Dual/6Quad application processor. This interface support from 80 Mbps up to 1 Gbps speed per data lane. The DSI Receiver core can mange one clock lane and up to 4 data lanes through the lane management, however two data lanes are implemented in the transmitter D-PHY, therefore the maximum throughput of display port is 2Gbps.

DSI host core is capable of supporting a variety of resolutions and formats:

- Resolution:
  - QQVGA
  - QCIF, QVGA
  - CIF
  - VGA
  - WVGA
  - SVGA
  - XVGA
- Pixel format:
  - RGB565
  - LRGB565
  - RGB666
  - RGB888

The DSI can support both command and video modes and up to four virtual channels to accommodate multiple displays.

- Command and video mode support (type 1, 2, 3, and 4 display architecture)
- Mode switching: low power and ultra low power
- Burst mode: dual video channel
- Non-burst mode: single video channel
- Bus turnaround
- Fault error recovery scheme

DPI and DBI could coexists in the system but only one of them could be active in a certain time moment

9.11.3 CSI-2

CSI-2 is a high performance serial interconnect bus for mobile applications connecting camera sensors to the host system.

Table 9-11. CSI Parametric Table

Name	CSI-2
Function	High speed serial interface controller for MIPI sensor interface
Supported standard version	MIPI CSI-2 Version 1.0 MIPI PPI interface for D-PHY MIPI D-PHY Version 1.0
External I/O Pins (List, Type, Schmidt Trigger, Speed)	Analog at chip boundary: DATAP[3:0] DATAN[3:0] CLKP CLKN Supplies AVDD -2.5V general analog supply AVDDREF - 2.5V analog supply for D-PHY reference source (low noise) should be directly connected to package ball VDD - 1.1V digital supply GND
SoC Buses (List, Type, Bandwidth)	Synchronous Pixel Interface - 32 bit, data formatting compliant to CSI-2 specification IPS slave interface x 32 bit
Interrupts	2
DMA Requests	N/A
Num of instantiations	1
Clock sources and range	AHB clock (132 MHz)

There is one instance of CSI-2 port in the i.MX 6Dual/6Quad application processor. This interface support from 80 Mbps up to 1 Gbps speed per data lane. The CSI-2 Receiver core can mange one clock lane and up to four data lanes through the lane management and de-packetization, providing a maximum throughput of 4 Gbps transfer rate.

9.11.4 D - PHY

D-PHY serves as physical layer for both MIPI DSI and CSI-2 interfaces.

D-PHY Transceiver unit is responsible for transmission and reception of data in High speed (HS) or Low Power (LP) mode. High speed mode is used for high-speed data transmission while the low power mode used for the control purpose. Point-to-point lane interconnect can be used for either data or clock signal transmission. High speed receiver is a differential line receiver while low- Power receiver is an un-terminated, single-ended receiver circuit.

Table 9-12. D-PHY Parametric Table

Name	D-PHY TX, D-PHY RX
Function	MIPI D-PHY dual mode transceiver
Supported standard version	D-PHY specification v1.0
External I/O Pins (List, Type, Schmidt Trigger, Speed)	One differential pair per lane. See CSI-2 and DSI parametric tables
SoC Buses (List, Type, Bandwidth)	PPI i/f
Interrupts	
DMA Requests	N/A
Num of instantiations	8 - 3 in DSI interface and 5 in CSI-2 interface
Clock sources and range	Reference clock - 6...27MHz

A single lane module is shown in the figure below.

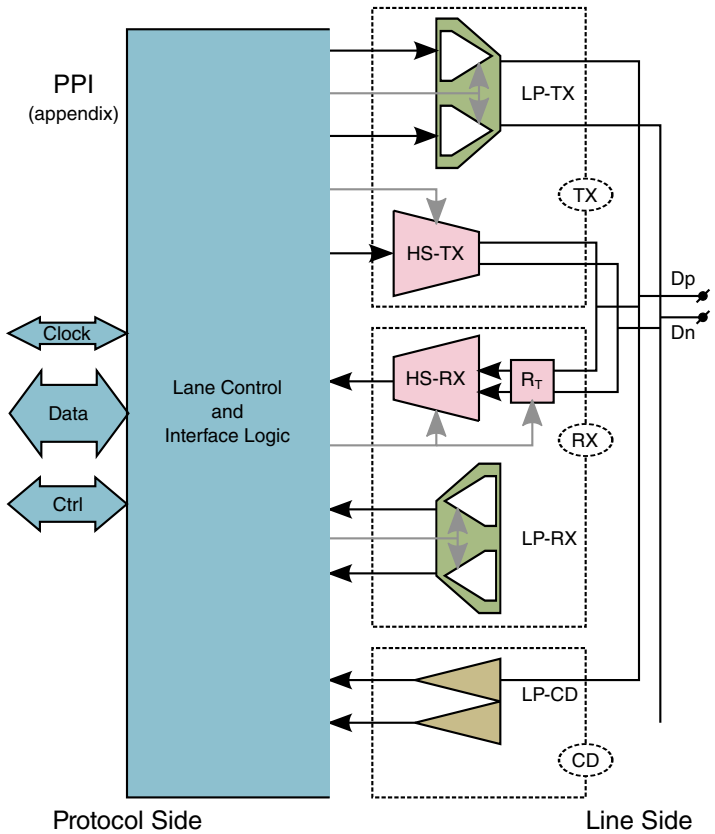


Figure 9-11. Single lane D-PHY- Block Diagram.

There is one instance of D-PHY TX (hard macro; two data lanes, one clock lane, PLL) and one instance of D-PHY RX (hard macro: four data lanes, one clock lane, no PLL) in i.MX 6Dual/6Quad.

9.12 Audio subsystem

The audio subsystem consists of the following modules: SSI-1, SSI-2, SSI-3, AUDMUX, ESAI, SPDIF, ASRC, and MLB. In addition, the IOMUX must be appropriately configured to get signals in and out of the chip.

Audio subsystem module overview provides an overview of each of the audio susbsystem component modules, followed by a module-specific section.

9.12.1 Audio subsystem module overview

The following figure shows a high level block diagram of the audio subsystem.

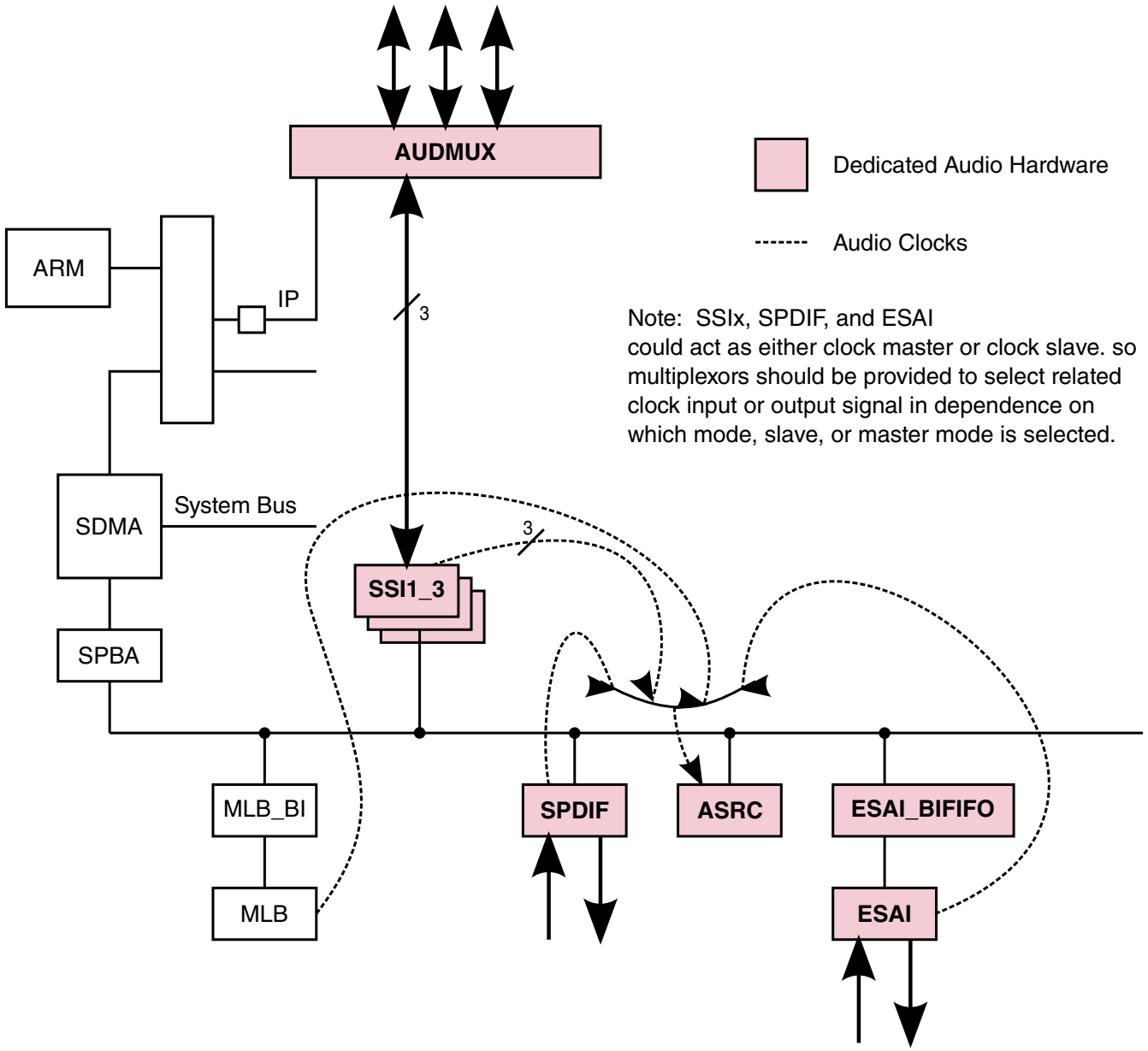


Figure 9-12. Audio subsystem block diagram

SSI1–3 are synchronous serial interfaces used to transfer audio data. SSI1–3 are on the shared peripheral bus. Instead of connecting to the IOMUX directly, their serial lines connect to the digital audio mux (AUDMUX).

AUDMUX provides flexible, programmable routing of the serial interfaces (SSI1, SSI2, or SSI3) to and from off-chip devices. AUDMUX routes audio data (and even splices together multiple time-multiplexed audio streams) but does not decode or process audio data itself. The ARM controls AUDMUX, but AUDMUX can route data even when the ARM is in a low-power mode.

ESAI (enhanced serial audio interface) provides a full-duplex serial port for serial communication with a variety of serial devices, including industry-standard codecs, SPDIF transceivers, and other processors. ESAI consists of independent transmitter and receiver sections, and each section has its own clock generator. ESAI is connected to the IOMUX and to the ESAI_BIFIFO module.

ESAI_BIFIFO (ESAI bus interface and FIFO) is the interface between the ESAI module and the shared peripheral bus. It contains the FIFOs used to buffer data to and from ESAI. It also provides the data word alignment and padding necessary to match the 24-bit data bus of the ESAI to the 32-bit data bus of the shared peripheral bus.

The SPDIF (Sony/Philips digital interface) audio module is a stereo transceiver that allows the processor to receive and transmit digital audio over it. The SPDIF receiver section includes a frequency measurement block that allows the precise measurement of an incoming sampling frequency. A recovered clock is provided by the SPDIF receiver section and may be used to drive both internal and external components in the system. SPDIF is connected to the shared peripheral bus.

ASRC (asynchronous sample rate converter) converts the sampling rate of a signal associated to an input clock into a signal associated to a different output clock. ASRC supports concurrent sample rate conversion of up to 10 channels of over 120dB THD+N. The sample rate conversion of each channel is associated to a pair of incoming and outgoing sampling rates. ASRC supports up to 3 sampling rate pairs. ASRC is connected to the shared peripheral bus.

9.12.2 Synchronous Serial Interface (SSI)

The Synchronous Serial Interface (SSI) is a full-duplex serial port that allows communication with external devices using a variety of serial protocols. The SSI supports a wide variety of protocols (SSI normal, SSI network, I2S, and AC-97), bit depths (up to 24 bits per word), and clock/frame sync options.

The SSI has two pairs of 15x32 FIFOs and hardware support for an external DMA controller in order to minimize its impact on system performance. The second pair of FIFOs provides hardware interleaving of a second audio stream which reduces CPU overhead in use cases where two timeslots are being used simultaneously.

The three SSIs may support three audio streams (possibly at different sample rates) simultaneously. SSI1, SSI2 and SSI3 are located on the Shared Peripheral Bus. Since the SDMA can directly access SSI1...SSI3 (being on the Shared Peripheral Bus), they can be used for high-bandwidth data transfers in order to optimize bus bandwidth consumption.

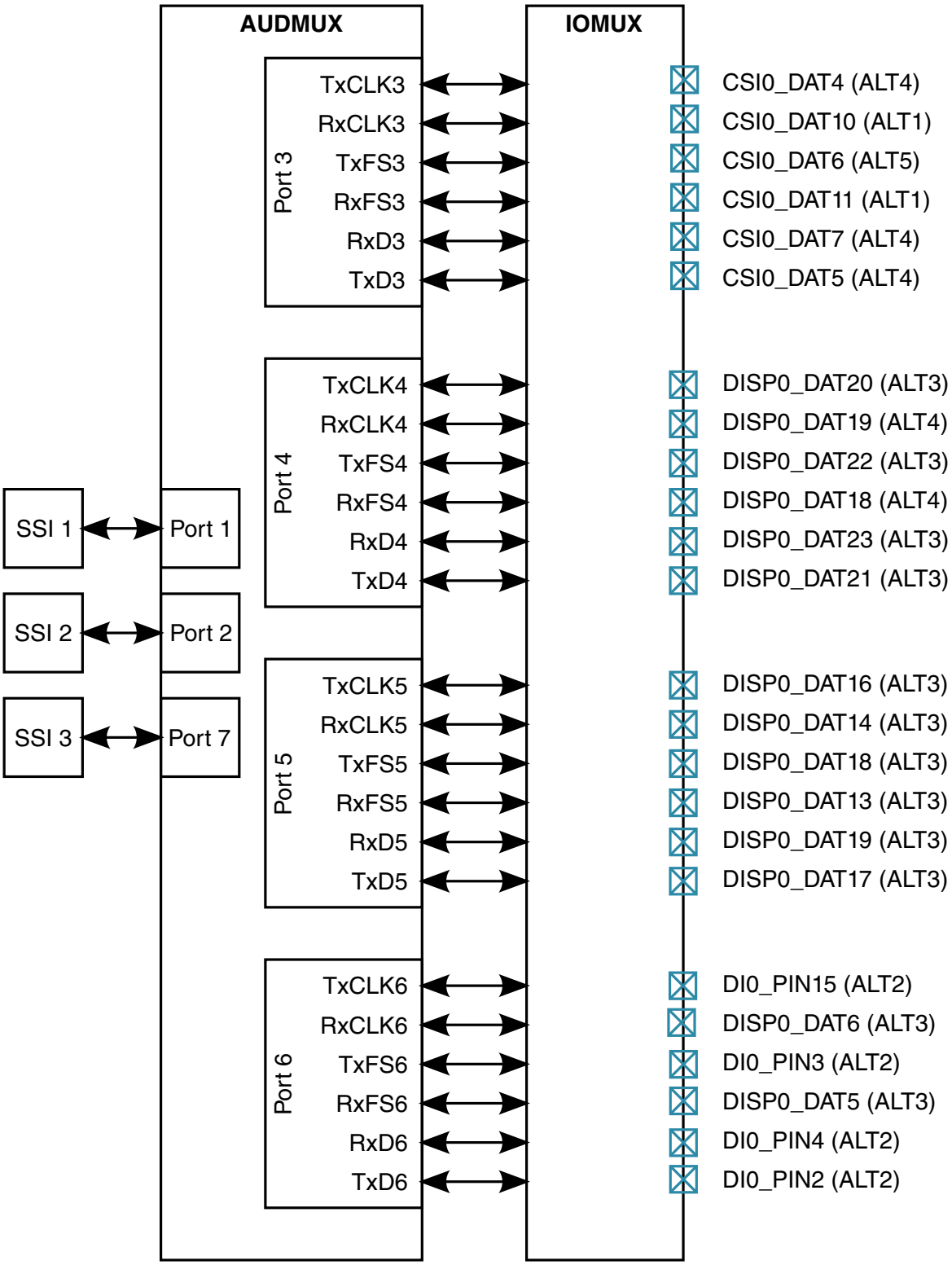
9.12.3 Digital Audio MUX (AUDMUX)

The Digital Audio Mux (AUDMUX) provides a programmable interconnect fabric for voice, audio, and synchronous data routing between host serial interfaces, such as SSI, and peripheral serial interfaces-that is, audio and voice codecs.

The AUDMUX includes two types of interfaces. Internal ports connect to the processor serial interfaces, and External ports connect to off-chip audio devices. A desired connectivity is achieved by configuring the appropriate host and peripheral ports.

The AUDMUX provides flexible, programmable routing of the on-chip serial interfaces to and from off-chip audio devices. The AUDMUX routes audio data (and even splices together multiple time-multiplexed audio streams) but does not decode or process audio data itself.

The following figure illustrates how the AUDMUX is connected in the system.



☒ - Daisy Chain, i.e. more than one PAD is available (only one option shown above).

Figure 9-13. AUDMUX System Block Diagram

9.12.4 Enhanced Serial Audio Interface (ESAI)

The Enhanced Serial Audio Interface (ESAI) provides a full-duplex serial port for serial communication with a variety of serial devices, including industry-standard codecs, SPDIF transceivers, and other processors.

The ESAI consists of independent transmitter and receiver sections, each section with its own clock generator. All serial transfers are synchronized to a clock. Additional synchronization signals are used to delineate the word frames. The normal mode of operation is used to transfer data at a periodic rate, one word per period. The network mode is also intended for periodic transfers; however, it supports up to 32 words (time slots) per period. This mode can be used to build time division multiplexed (TDM) networks. In contrast, the on-demand mode is intended for non-periodic transfers of data and to transfer data serially at high speed when the data becomes available.

The ESAI has 12 pins for data and clocking connection to external devices. The ESAI is internally connected to the ESAI_BIFIFO, and does not connect directly to the shared peripheral bus. The ESAI interface is designed for a 24-bit data bus, while the shared peripheral data bus is 32-bit wide. Also, the ESAI data paths are only double buffered, not allowing efficient DMA service in the applications processor environment. The ESAI_BIFIFO allows increasing the data buffering and data width matching to the shared peripheral bus.

9.12.5 Sony/Philips Digital Interface (SPDIF)

The Sony/Philips Digital Interface (SPDIF) module is a stereo that allows the processor transmit digital audio over it using the IEC60958 standard, consumer format. i.MX 6Dual/6Quad provides one SPDIF transmitter with one output and one SPDIF receiver with one input.

The SPDIF allows the handling of both SPDIF channel status (CS) and User (U) data.

For the SPDIF transmitter, the audio data is provided by the processor via the SPDIFTxLeft and SPDIFTxRight registers, and the data is stored in two 16-word-deep FIFOs, one for the right channel, the other for the left channel. The FIFOs support programmable watermark levels so that FIFO Empty service request can be triggered when the combined number of empty data words locations in both FIFOs is 8, 16, 24 or 32 words. It is recommended to program the watermark level to trigger a FIFO Empty service request when 16 word locations are empty. For optimal performance when servicing the FIFO Empty service request, the FIFOs should be written alternately,

starting with the left channel FIFO. The Channel Status bits are also provided via the corresponding registers. The SPDIF transmitter generates an SPDIF output bitstream in the biphase mark format (IEC 60958), which consists of audio data, channel status and user bits.

The data handled by the SPDIF module is 24-bit wide. The 24-bit SPDIF data is aligned in the 24 least significant bits of the 32-bit shared peripheral bus data word. The 8 most significant bits of the 32-bit word are ignored by the SPDIF Transmitter when data is being stored in the Transmit FIFOs from the peripheral bus. The 8 most significant bits of the 32-bit word are zeroed by the SPDIF Receiver module when the data is being read from the Receiver FIFOs to the peripheral bus.

Note that 16-bit data is left-aligned in the 24-bit word format of the SPDIF. When 16-bit data is to be transmitted, the 32-bit word to be written to the SPDIF Transmit FIFOs should be created as follows: the 16-bit data should be located in the middle two bytes of the 32-bit data word and the 8 bits of the LSB must be set to zero, while the 8 bits of the MSB will be ignored.

The SPDIF Transmit clock is generated by the SPDIF internal clock generator module and the clock sources are from outside of the SPDIF block. The ESAI, SSI's and MLB clock sources should provide a clock that is at least 64 x Fs, where Fs is the sampling frequency. The external clock source should provide at least 128 x Fs. Clocks of higher frequency may be provided as long as the multiplication factor is a power of 2 (for example, 128x, 256x or 512x). Also, clock frequency precision of 100ppm or better should be provided.

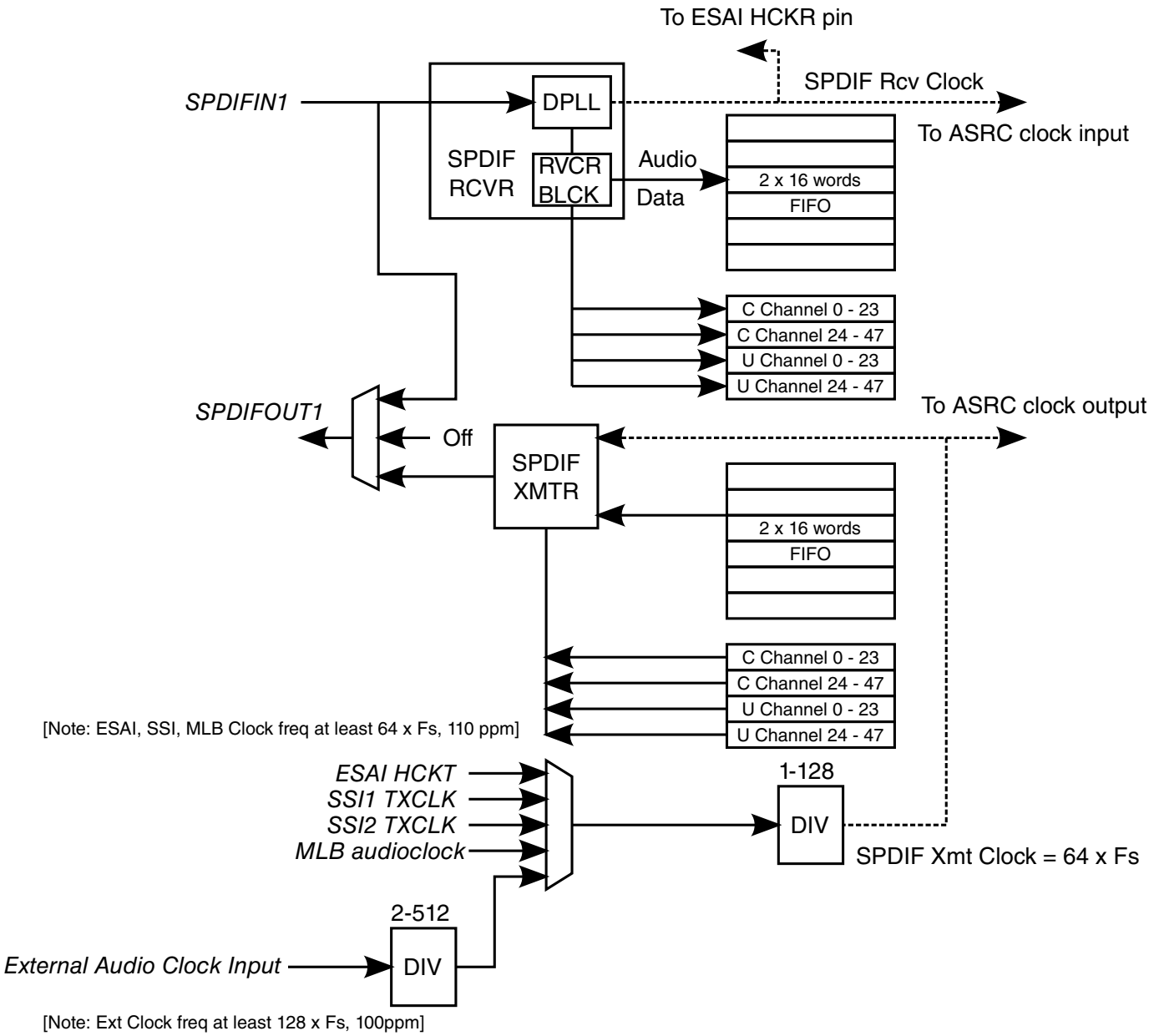


Figure 9-14. SPDIF Transceiver Clock Diagram

9.12.6 Asynchronous Sample Rate Converter (ASRC)

The incoming audio data may be received from various sources at different sampling rates. The outgoing audio data may have different sampling rates, and it can also be associated to output clocks that are asynchronous to the input clocks.

ASRC converts the sampling rate of a signal associated to an input clock into a signal associated to a different output clock. ASRC supports concurrent sample rate conversion of up to 10 channels of about -120dB THD+N. The sampling rate conversion of each channel is associated to a pair of incoming and outgoing sampling rates. ASRC supports up to 3 sampling rate pairs.

In the real-time audio use case, both input/output sampling rate clocks are activated. Both sampling rate clocks are directly connected to ASRC, and the ratio estimation of the input clocks to output clocks is used to perform the sample rate conversion in ASRC hardware.

The SSI1, SSI2, SSI3, S/PDIF, ESAI, and MLB audio modules can act as either a clock master or a clock slave. Multiplexors are provided on a chip level to select which of the module's clock signal, input or output, will be connected to ASRC input in dependence of the module's operational mode as it is shown in Figure 9-15 and Table 9-13. Figure 9-15 presents the multiplexor cell used for all ASRC input clocks. Table 9-13 shows the detailed clocks, multiplexor control and data bits for each ASRC input clock. For the audio blocks clocks that are directly conected to ASRC (without the multiplexor scheme) see Table 9-14.

In the non-real time streaming audio use case, the input sampling rate clock does not need to be provided. Instead the ideal-ratio value conversion is set in the ASRC interface registers. In this case only the output sampling rate clock must be provided, and the fixed ratio of the input to the output in the register is used to perform the sample rate conversion.

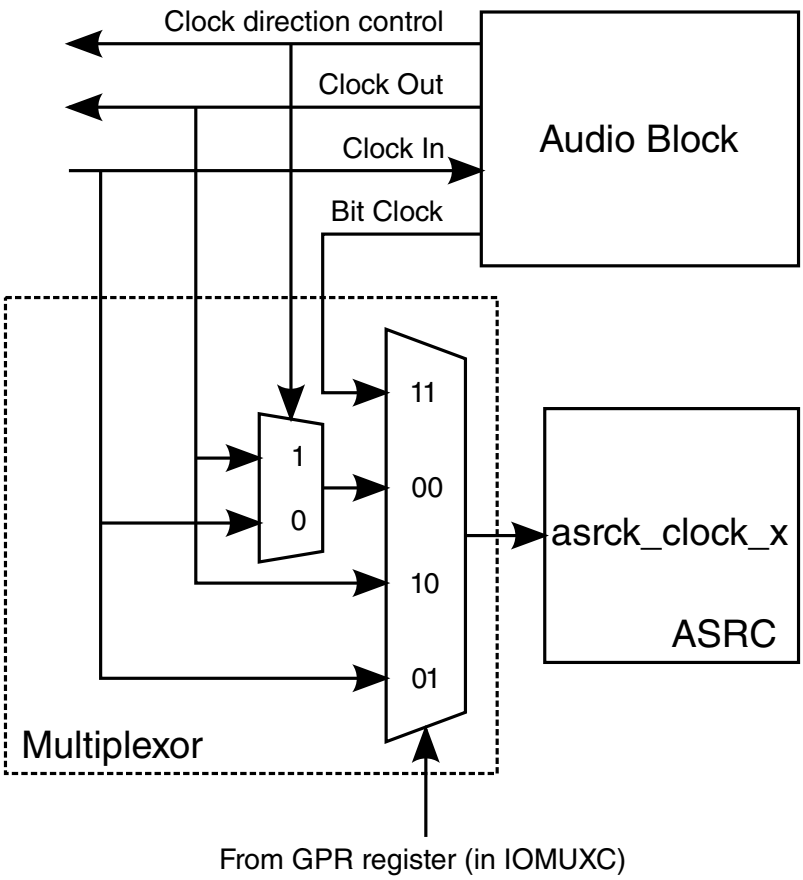


Figure 9-15. Muxing scheme template for ASRC input clocks

Table 9-13. ASRC Muxed Input Clocks

ASRC Clock Input	Audio block source	MUX2:1 Control	MUX2:1 Input 0	MUX2:1 Input 1	MUX4:1 Control	MUX4:1 Input 00	MUX4:1 Input 01	MUX4:1 Input 10	MUX4:1 Input 11
asrck_clock_1	SSI-1 (Rx)	SRCR[RXDIR][1]	External SRCK	InternalSRCK	GPR0[17:16][2]	MUX2:1 output	External SRCK	InternalSRCK	RX bit clock
asrck_clock_9	SSI-1 (Tx)	STCR[TXDIR]	External STCK	InternalSTCK	GPR0[17:16]	MUX2:1 output	External STCK	InternalSTCK	TX bit clock
asrck_clock_2	SSI-2 (Rx)	SRCR[RXDIR]	External SRCK	InternalSRCK	GPR0[19:18]	MUX2:1 output	External SRCK	InternalSRCK	RX bit clock
asrck_clock_a	SSI-2 (Tx)	STCR[TXDIR]	External STCK	InternalSTCK	GPR0[19:18]	MUX2:1 output	External STCK	InternalSTCK	TX bit clock
asrck_clock_3	SSI-3 (Rx)	SRCR[RXDIR]	External SRCK	InternalSRCK	GPR0[21:20]	MUX2:1 output	External SRCK	InternalSRCK	RX bit clock
asrck_clock_b	SSI-3 (Tx)	STCR[TXDIR]	External STCK	InternalSTCK	GPR0[21:20]	MUX2:1 output	External STCK	InternalSTCK	TX bit clock
asrck_clock_0	ESAI (Rx)	RCCR[RCKD]	External SCKR	InternalSCKR	GPR0[23:22]	MUX2:1 output	External SCKR	InternalSCKR	N/A[3]
asrck_clock_8	ESAI (Tx)	TCCR[TCKD]	External SCKT	InternalSCKT	GPR0[23:22]	MUX2:1 output	External SCKT	InternalSCKT	N/A

1. Register[bit] format, SRCR is the regsiter name in the block specified in the "Audio block source" column, while RXDIR is name field/bit in the register.
2. GPR0 is the General Register 0 in the IOMUX controller. See IOMUXC chapter for more details.
3. Not in used, the clock value is "0".

Table 9-14. ASRC Direct Clocks

ASRC Clock Input	Block driving clock source	Block output clock
asrck_clock_4	SPDIF (Rx)	SPDIF Rx Clock
asrck_clock_c	SPDIF (Tx)	SPDIF Tx Clock
asrck_clock_5	MLB	MLB Input clock (from device)
asrck_clock_6	External (from PAD)	Via IOMUX (from PAD) KEY_ROW3 (ALT1) [default] GPIO_0 (ALT3) GPIO_18 (ALT4)
asrck_clock_7	Reserved	Reserved[1]
asrck_clock_d	CCM	SPDIF1 clock root

1. Not in used, the clock value is "0".

Chapter 10
Clock and Power Management

10.1 Introduction

This chapter describes the Clock and Power Management architecture of the SoC.

The i.MX 6Dual/6Quad device targets many applications where low power consumption, long battery life, always-on and instant-on capabilities, and no need for active cooling are paramount. For this reason, the i.MX 6Dual/6Quad design constantly focuses on reducing current consumption as much as possible, while simultaneously enabling the maximum level of peak performance and a balanced level of sustained performance for target applications. To achieve this, the i.MX 6Dual/6Quad architecture uses a wide range of power-management techniques and their combinations for maximum system design flexibility:

This introduction contains information about:

- Structural components of the power and clock management systems of the i.MX 6Dual/6Quad device
- Power, clock and thermal management techniques supported by the i.MX 6Dual/6Quad device

All the numerical values are typical or examples, for accurate values one should use the datasheet.

10.2 Device Power Management Architecture Components

To provide a clean and versatile architecture supporting a wide range of power-management techniques, the Clock and Power rails are considered managed resources.

For each rail, two levels of management are defined: the first level is centralized or SoC-level resource management, and the second is a local or "module level" resource management.

The high level architectural view of clock, power and thermal management system of the chip is presented in the figure below.

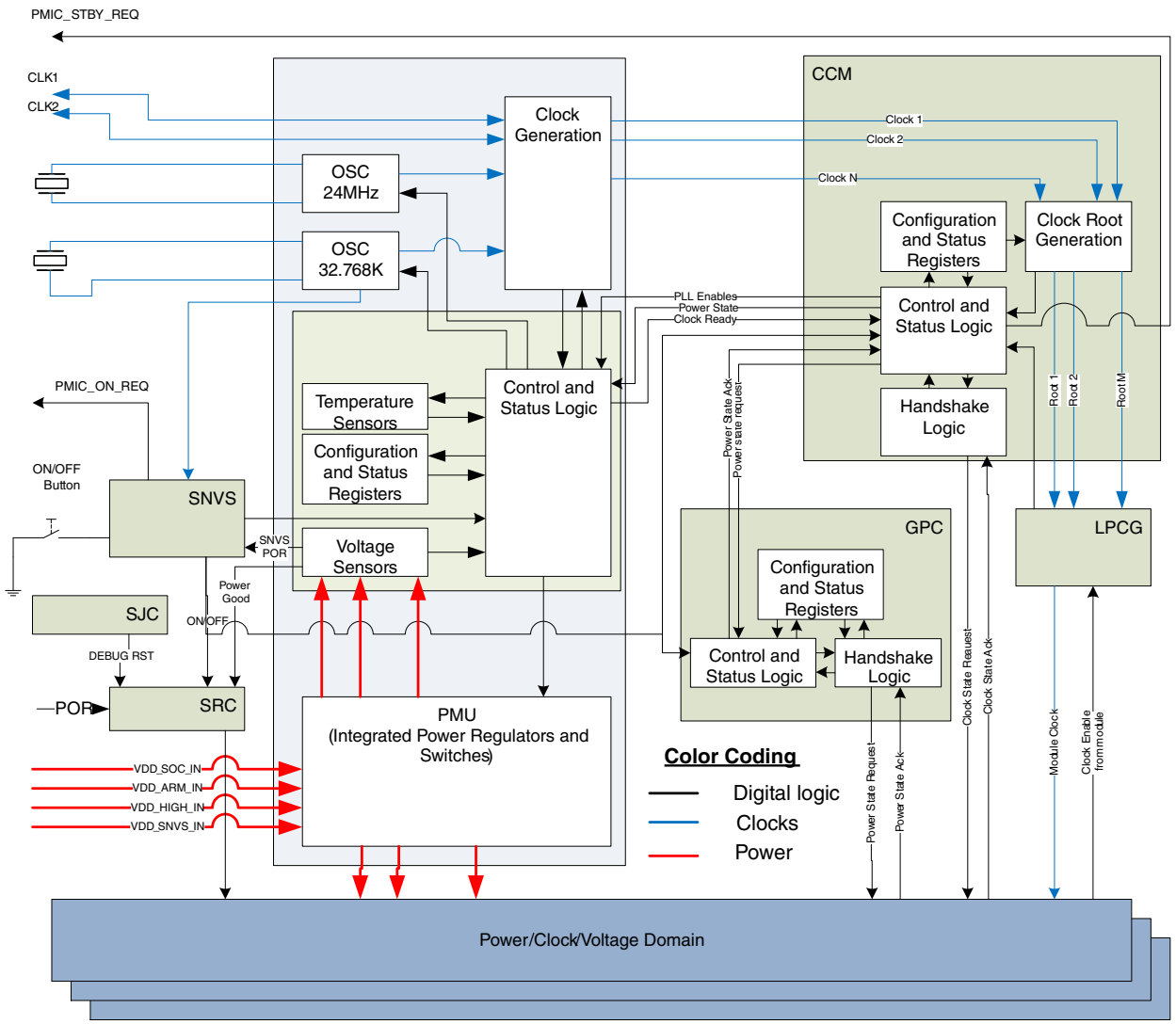


Figure 10-1. Power and clock management framework

10.2.1 Centralized components of clock generation and management

Centralized components of clock generation and management sub-system are implemented in the following blocks:

- CCM (Clock Control Module): The CCM module provides control for primary (source level) and secondary (root level) clock generation, division, distribution, synchronization, and coarse-level gating.

- See Clock Controller Module (CCM)) for information on the CCM architecture, functional description and programming model.
- LPCG (Low Power Clock Gating): This module distributes the clocks to all blocks in the SoC and handles block level software-controllable and automated clock gating. See Clock Controller Module (CCM) for information on the LPCG architecture and functional description.

10.2.2 Centralized components of power generation, distribution and management

Centralized components of power generation, distribution and management sub-system are implemented in the following blocks:

- PMU (Integrated Power Management Unit). See Power Management Unit (PMU) for information on the PMU architecture, functional description and programming model.
- GPC (General Power Controller). See General Power Controller (GPC) for information on the GPC architecture, functional description and programming model.

10.2.3 Reset generation and distribution system

Power and clock management are accompanied with an appropriate reset generation and distribution system, centralized functions of which are implemented System Reset Controller (SRC). See General Power Controller (GPC) for information on the GPC architecture, functional description and programming model.

10.2.4 Power and clock management framework

Together, the modules listed above provide enhanced power-management features with the centralized control for the clock, reset, and power-management signals on the SoC.

The centralized management defines the minimal managed components of the power-management architecture. These components are called the clock, power, and voltage domains.

NOTE

A domain is a group of modules or functional blocks that share a common resource entity (for example, common clock root, common power source, or a common power switch). The software component managing shared resources should take

into account the joint constraints of all the modules belonging to that resource domain.

10.3 Clock Management

10.3.1 Centralized components of clock management system

The clock generation and management system is built around the CCM and LPCG blocks.

A high level block diagram of the clock management system in the SoC environment is shown in figure below.

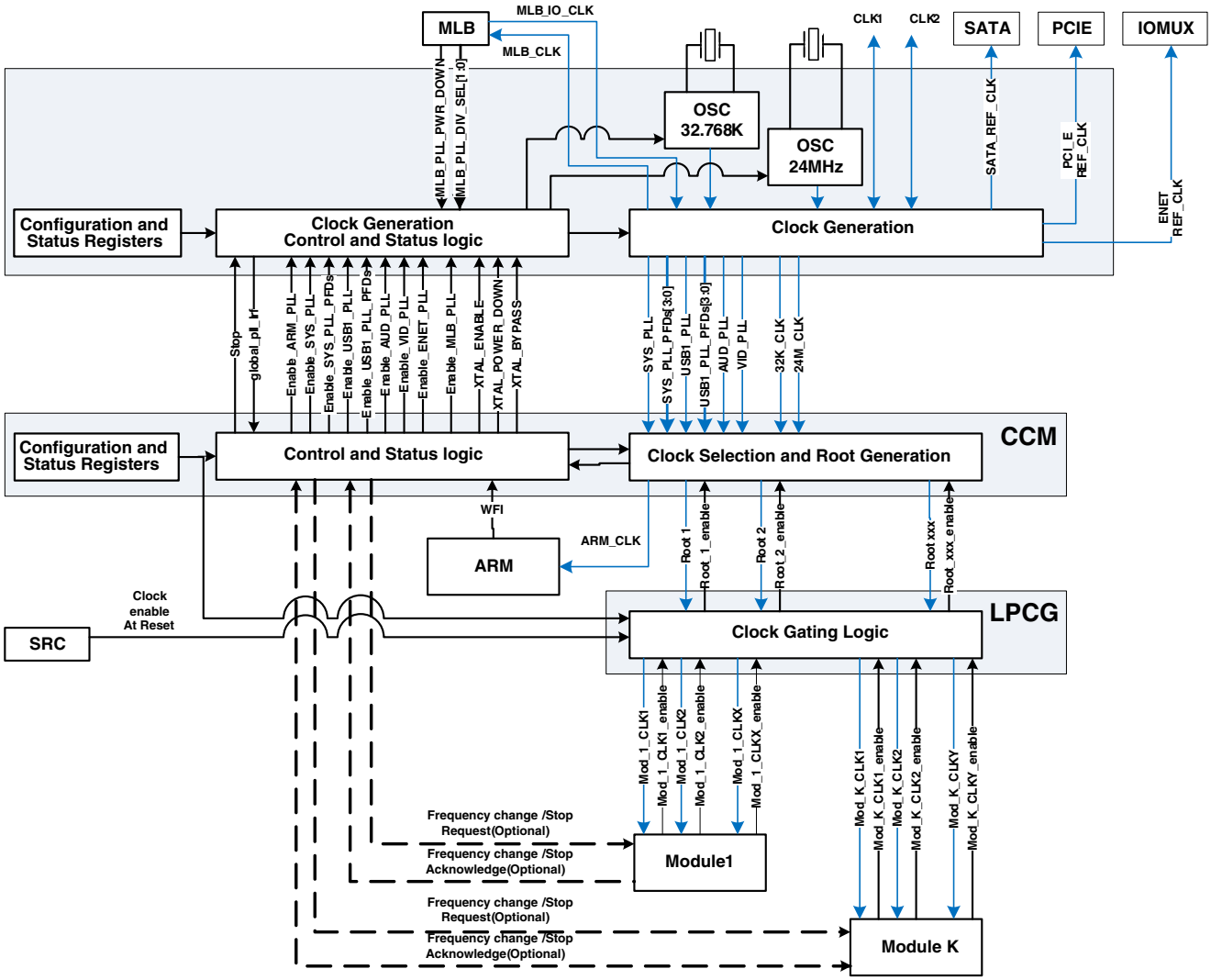


Figure 10-2. Clock Management System

A high level block diagram of the clock generation is shown in the figure below.

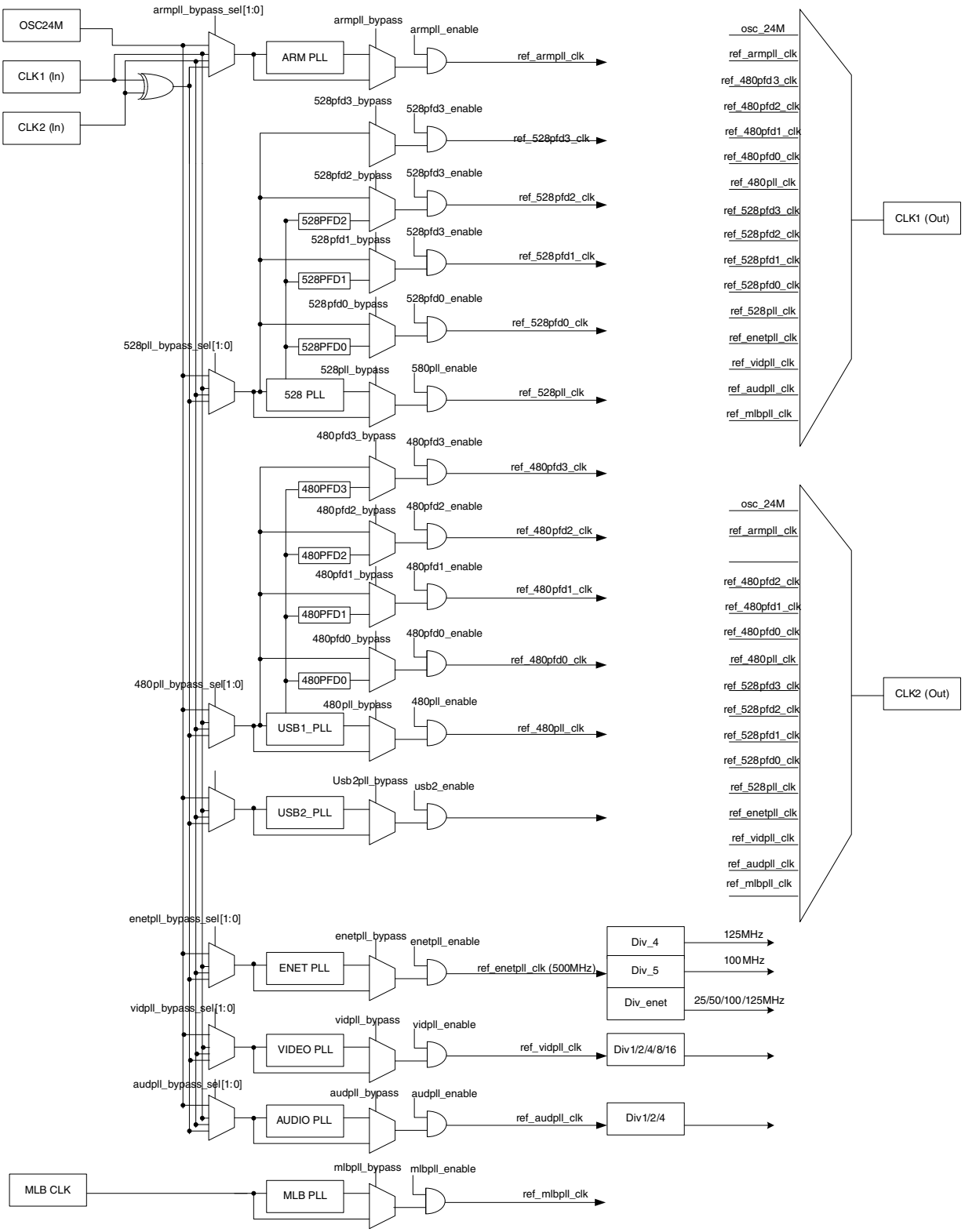


Figure 10-3. Primary Clock Generation

10.3.2 Clock generation

The clock generation section includes the components detailed in the following sections.

10.3.2.1 Crystal Oscillator (XTALOSC)

The Crystal Oscillator block is comprised of both the high frequency oscillator (typical frequency is 24 MHz) and the low frequency real time clock oscillator (typical frequency of 32.768 KHz). Each of these oscillators is implemented as a biased amplifier that, when combined with a suitable external quartz crystal and external load capacitors, implements an oscillator. See Crystal Oscillator (XTALOSC) for details of the XTALOSC block.

10.3.2.2 LVDS I/O ports

There is two LVDS I/O ports used for clock generation. The low jitter differential I/O ports are provided to input and output clocks. They can take input clocks from outside of the SoC and provide them to the PLLs or to the other modules, or they can take the outputs of the PLLs and provide them outside of the SoC as a functional or reference clock.

10.3.2.3 PLLs

Eight PLLs are included in the clock generation section. Two of these PLLs are each equipped with four Phase Fractional Dividers (PFDs) in order to generate additional frequencies.

NOTE

Each PFD works independently by interpolating the VCO of the PLL to which it is connected. It effectively takes the PLL VCO frequency and produces 18/N x Fvco at its output where N ranges from 12 to 35. PFD is a completely digital design with no analog components or feedback loops. The frequency switch time is much faster than a PLL because keeping the base PLL locked and changing the integer N only changes the logical combination of the interpolated outputs of the VCO. Note that the PFD not only enables faster frequency changes than a PLL, but also allows the configuration to be safely changed "on-the-

fly" without going through the output clock disabling/enabling process.

The eight PLLs are listed below:

- PLL1 (also referred to as ARM_PLL) - This is the PLL clocking the ARM core complex. It is a programmable integer frequency multiplier capable of output frequency of up to 1.3GHz (may exceed chip capabilities, see datasheet for more information).
- PLL2 (also referred tp as System_PLL or 528_PLL) - PLL2 generally runs at a fixed multiplier of 22, producing 528MHz output frequency with 24MHz reference from XTALOSC. It has the capability to spread spectrum the generated signal. Besides the main output, this PLL drives four PFDs (528_PFD0...528_PFD3). The main PLL output and its PFD outputs are used as inputs for many clock roots. These do not require exact/constant frequency and can be changed as a part of dynamic frequency scaling procedure and/or can be spread-spectrum modulated. Typically, this PLL is a clock source for internal system buses, internal processing logic, DDR interface, NAND/NOR interface modules, etc.
- PLL3 (also referred to as USB1_PLL or 480 PLL) - PLL3 is used in conjunction with the first instance of USB PHY (USB0 PHY, also known as OTG PHY). This PLL also drives four PFDs (480_PFD0...480_PFD3) and runs at a fixed multiplier of 20. This results in a VCO frequency of 480MHz with a 24MHz oscillator. The main PLL output and its PFD outputs are used as inputs for many clock roots that require constant frequency, such as UART, CAN and other serial interfaces, audio interfaces, etc.
- 480_PLL2 (also referred to as USB2_PLL) - This PLL provides clock exclusively to USB2 PHY (also known as HOST PHY). It runs at a fixed multiplier of 20, resulting in a VCO frequency of 480MHz with a 24MHz oscillator.
- PLL4 (also referred to as an Audio PLL) - This is a fractional multiplier PLL used for generating a low jitter and high precision audio clock with standardized audio frequencies. The PLLs oscillator frequency range is from 650MHz to 1300MHz, and the frequency resolution is better than 1Hz. This clock is mainly used as a clock for serial audio interfaces and as a reference clock for external audio codecs. It is equipped with a divider on its output and can generate divided by 1, 2 or 4 from the PLL VCO frequency.
- PLL5 (also referred to as a Video PLL) - This is a fractional multiplier PLL used for generating a low jitter and high precision video clock with standardized video frequencies. The PLLs oscillator frequency range is from 650MHz to 1300MHz, and the frequency resolution is better than 1Hz. This clock is mainly used as a clock for display and video interfaces. It is equipped with dividers on its output and can generate clock divided by 1, 2, 4, 8 or16 from the PLL VCO frequency

- PLL6 (also referred to as PLL_ENET) - This PLL implements a fixed 20+(5/6) multiplier. With a 24MHz input, it has a VCO frequency of 500MHz. This PLL is used to generate:
- 125MHz for the PCIe serial interface and reduced gigabit ethernet interface.
- 100MHz for the SATA serial interface
- 50 or 25 MHz for the external ethernet interface.
- MLB_PLL - This PLL takes the Media Link Bus (MLB) interface clock, multiplies it up by 1, 2, or 4, and delay compensates so that the MLB data stream can be captured.

10.3.2.3.1 General PLL Control and Status Functions

PLLs configuration and control functions are accessible via individual per PLL and PFDs and global configuration and status registers.

Reference input clock for any of the PLLs except the MLB PLL could be selected individually by the BYPASS_CLK_SRC field of the PLL control register. See CCM Analog Memory Map/Register Definition for more information.

Each of the PLLs could be individually configured to "Bypass", "Output disabled" and "Power Down" modes.

When configured in "Bypass" PLL pass directly its input reference clocks to the PLL output. Bypassing the PLL is done by setting the BYPASS bit in the control register. For the PLL equipped with PFDs the input reference clock is also bypassed to all PFDs outputs.

When configured in output disabled mode (ENABLE=0), the PLL's output is completely gated and there is neither a bypass clock nor PLL generated clock that propagates to PLL output. Each PLL output has an individual "Output Enable" control bit. The PFDs are gated by the ENABLE bit of their associated PLL. Each PFD does have an associated clock gate bit that can be used to turn it off individually.

When configured in "Power Down mode" most of the PLL circuitry is switched off. Neither main PLL output nor PFD outputs are available in this mode.

When the related PLL is powered up from the power down state or made to go through a relock cycle due to PLL regrogramming, it is required that the related PFDx_CLKGATE bit in CCM_ANALOG_PFD_480n or CCM_ANALOG_PFD_528n, be cycled on and off (1 to 0) after PLL lock. The PFDs can be in the clock gated state during PLL relock but must be un-clock gated only after lock is achieved. See the engineering bulletin, Configuration of Phase Fractional Dividers (EB790) at www.freescale.com for procedure details.

Individual PLL status is reflected in "PLL Lock" bits of the PLL control registers. PLL enable logic which monitors the register value change is implemented to gate off the PLL outputs during the "lock in" period.

Outputs are generated to be sent out by monitoring the individual PLL lock flags and filtering out any random initial edges.

Individual PLL Lock ready flags are are first "ORED" with "enables" and then "ANDED" together to generate the global PLL lock ready flag that reflects status of all PLLs enabled in certain moment.

CCM Memory Map/Register Definition and CCM Analog Memory Map/Register Definition contains detailed descriptions of the memory mapped registers and control functions of the clock generation sub-module.

10.3.2.4 CCM

CCM includes:

- Clock root generation logic - This sub-block provides the registers that control most of the secondary clock source programming, including both the primary clock source selection and the clock dividers. The clock roots are each individual clocks to the core, system buses (AXI, AHB, IPG) and all other SoC peripherals, among those are serial clocks, baud clocks, and special functional clocks. Most of clock roots are specific per module.
- CCM, in coordination with GPC, PMU and SRC, manages the Power modes, namely RUN, WAIT and STOP modes. The gating of the peripheral clocks is programmable in RUN and WAIT modes.

CCM manages the frequency scaling procedure for:

- ARM core clock - "on the fly" without clock interruption, by either shifting between PLL sources PLL clock change or by changing the divider ratio.
- Graceful changing of the DDR memory controller clock. See MMDC handshake and Self refresh and Frequency change entry/exit for more details.
- Peripheral root clock - by using programmable divider. The division factor can change on the fly without loss of clocks.

NOTE

Note: on-the-fly frequency changing for synchronous interfaces like serial audio interfaces (SSI, ESAI), Audio Sample Rate Converter (ASRC), video and display interfaces (MIPI, IPU), or general purpose serial interfaces (UART, CAN) in general causes synchronization loss and should not be done.

10.3.2.5 Low Power Clock Gating unit (LPCG)

The LPCG block receives the root clocks from CCM and splits them to clock branches for each block. The clock branches are individually gated clocks.

The enables for those gates can come from four sources:

- Clock enable signal from CCM - This signal is generated depending on the power mode the system is in. For each power mode, it is defined in the software using the configuration of the CGR bits in CCM.
- Clock enable signal from the block - This signal is generated by the block based on its internal logic. Not every enable signal from the block is used. Each clock enable signal from the block can be overridden based on the programmable bit in CCM.
- Clock enable signal from the reset controller (SRC) - This signal will enable the clock during the reset procedure.

10.3.3 Peripheral components of clock management system

The figure found here shows the clock interface of a functional module in the i.MX 6Dual/6Quad system.

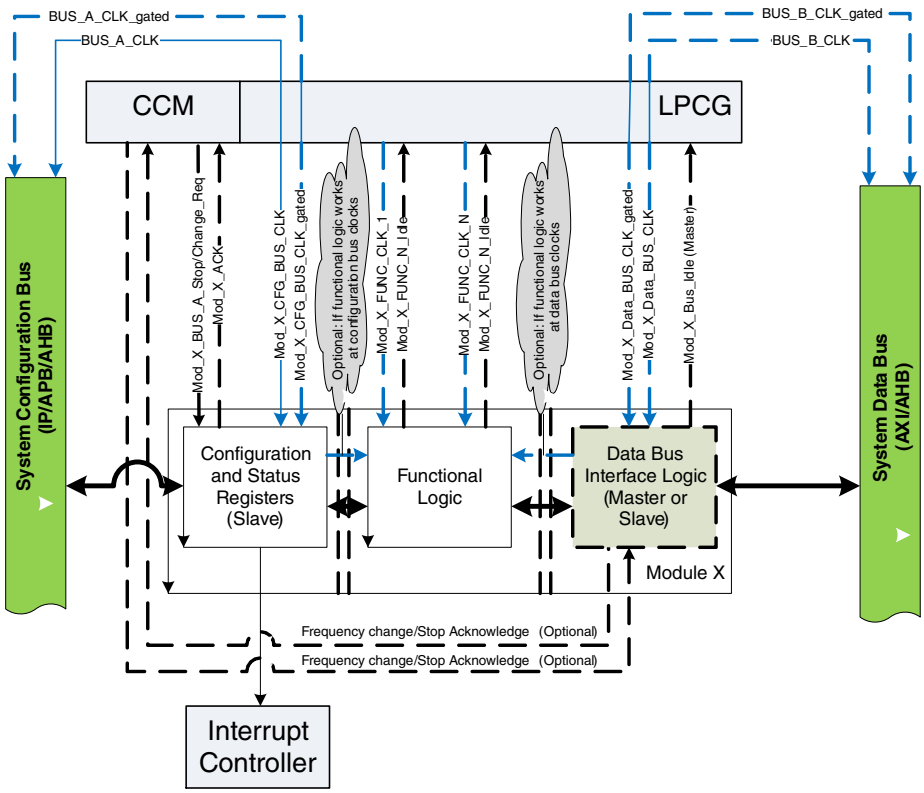


Figure 10-4. Clock interface of the functional module in i.MX 6Dual/6Quad system

10.3.3.1 Interface and functional clock

Each block within the SoC has specific clock input characteristic requirements. Based on the characteristics of the clocks delivered to modules, the clocks are divided into two categories: bus interface clocks and functional clocks.

The bus interface clocks have the following characteristics:

- They ensure proper communication between any block/subsystem and the system buses.
- In most cases, they supply the system interface and configuration registers of the block.
- A typical block has one system bus clock, but blocks with multiple interface clocks may also exist (that is, when a block is connected to multiple buses).

- The bus interface clocks are always fed by the outputs of the CCM/LPCG.
- Clock management for this type of clock is always implemented at the system level because it requires coordinated clock management between the block and system buses.

Functional clocks have the following characteristics:

- They supply the functional part of a block or a subsystem.
- Typically, these clocks are completely asynchronous and independent from the bus interface clock of the same block.
- A block can have one or more functional clocks. Some functional clocks are mandatory, while others are optional for its functioning. A block needs its mandatory clock(s) to be operational. The optional clocks are used for specific features and can be shut down without stopping the block activity (in the case of IPU, that could be the clock for a display interface or a camera sensor interface).
- The functional clocks are fed either by a CCM/LPCG block functional clock output, or by some other clock source, such as a clock output of another block or an external signal coming from IOMUX.

10.3.3.2 Block level clock management

Each block in the system may also have specific clock requirements. Certain module clocks must be active when operating in some specific modes, or may be gated in some others. Generally, the activation and gating of the module clocks are managed by LPCG. Hence, the LPCG block must be programmed properly and, in case of hardware controllable clock gating, peripheral module should provide signals indicating when to activate and when to gate the module clocks.

The LPCG block differentiates the clock-management behavior for device modules based on whether the block can initiate transactions on the device interconnect (called master module), or if it cannot initiate transactions and only responds to the transactions initiated by the master (called slave module). Thus, two hardware-based clock-management protocols are used:

- Master protocol - Clock-management protocol between the CCM/LPCG and blocks that can be bus master
- Slave protocol - Clock-management protocol between the CCM/LPCG and slave modules

10.3.3.2.1 Master clock protocol

This protocol is used to indicate that a master module is ready to initiate a transaction on the device interconnect and requests specific (both functional and interface) clocks. The CCM/LPCG block ensures that the required clocks are active when the master module requests that the CCM/LPCG enable them. The module is said to be functional after the required clocks are activated.

Similarly, when the master module no longer requires the clocks, it informs the LPCG/CCM block and the LPCG/CCM can then gate the clocks to the module and all the clock precedents that are not used by other blocks. The master module is then said to be in clock-gated or partially clock gated mode.

Examples of modules supporting master clock protocol are GPU3D, VPU, GPU2D, VDOA and USDHC. Please see details in chapters describing these modules and in the CCM enable override register (CCM_CMEOR).

10.3.3.2.2 Slave clock protocol

This hardware protocol allows CCM to control the state of a slave module. CCM informs the slave module, through assertion of a stop/change request, when its clocks (both interface and functional) can be changed or gated. The slave acknowledges the request and CCM is then allowed to gate or change the clocks to the block.

Similarly, a clock-gated slave module may need to be woken up because of some event or a service request from a master module. In this situation, CCM enables the clocks to the module and then de-asserts the stop request to signal the module to wake up.

Examples of modules supporting slave clock protocol are CAN, EPIT and GPT. Please see details in chapters describing these modules and in the CCM Module Enable Overide Register (CCM_CMEOR). See CCM Memory Map/Register Definition for more details.

The protocol in both "master" and "slave" cases is completely hardware-controlled, but software should configure the clock management behavior for the module in two places: in the CCM registers associated with the block and in the block configuration registers.

10.3.3.3 Clock Domain(s)

A clock domain is a group of blocks fed by clock signals controlled by the same clock controls in CCM. By gating the clocks in a clock domain, the clocks to all the blocks belonging to that clock domain can be gated/activated, either by software control or by hardware control associated with block activity. Thus, a clock domain allows efficient control of the dynamic power consumption of the domain.

The device is partitioned into multiple clock domains and each clock domain is controlled by an associated group of clock gating cells within the LPCG block. This allows the CCM/LPCG to individually activate and gate each clock domain of the system.

Examples of clock domains are: Main AXI bus clock domain, VPU clock domain, GPU3D clock domain, etc.

10.3.3.4 Domain level clock management

The domain clock manager can automatically (based on hardware conditions) and manage the bus interface clocks within the clock domain. The functional clocks within the clock domain are managed through software settings.

10.3.3.5 Domain dependencies

A domain dependency is a hierarchical relationship between two clock domains. Clock domain "X" is said to depend on a clock domain "Y" when a block in clock domain "Y" provides services (or even just a clock) to a block in clock domain "X". As a result, clock domain "Y" must be active whenever clock domain "X" is active.

The dependency between two clock domains may also exist if one clock domain serves to ensure communication between two blocks (for example, the clock domain of the device interconnect).

10.4 Power management

10.4.1 Centralized Components of Power Management System

The power generation and management system is built around the PMU and GPC blocks.

A high level block diagram of the power management system in the SoC environment is shown in the figure below.

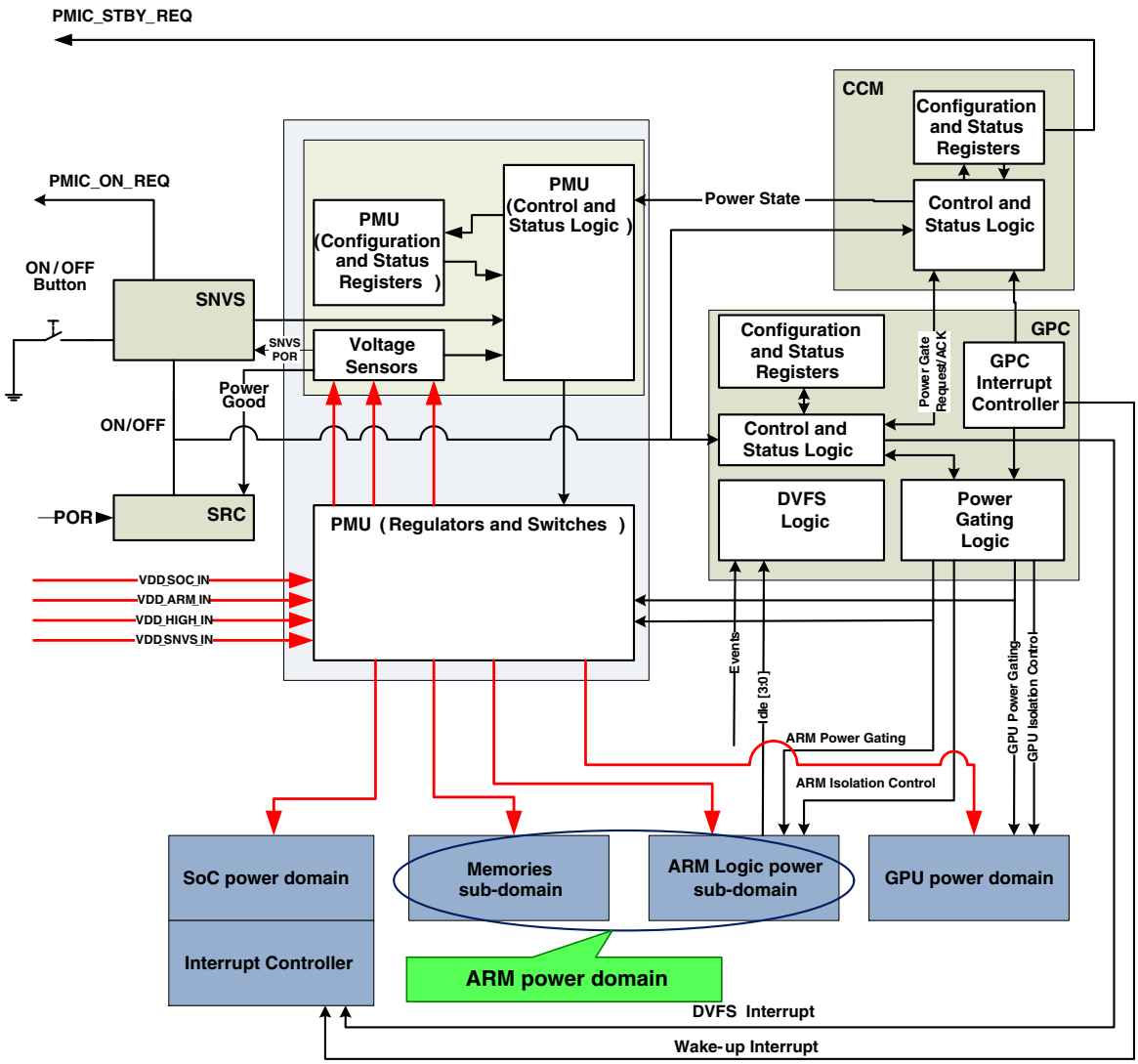


Figure 10-5. Power Management System

10.4.1.1 Integrated PMU

The first component of the power management system, referred to as the integrated PMU, is designed to simplify the external power interface.

It consists of a set of secondary power supplies that enable SoC operations from just two or three primary supplies. The high level block diagram of the power tree, utilizing the integrated PMU, is shown below.

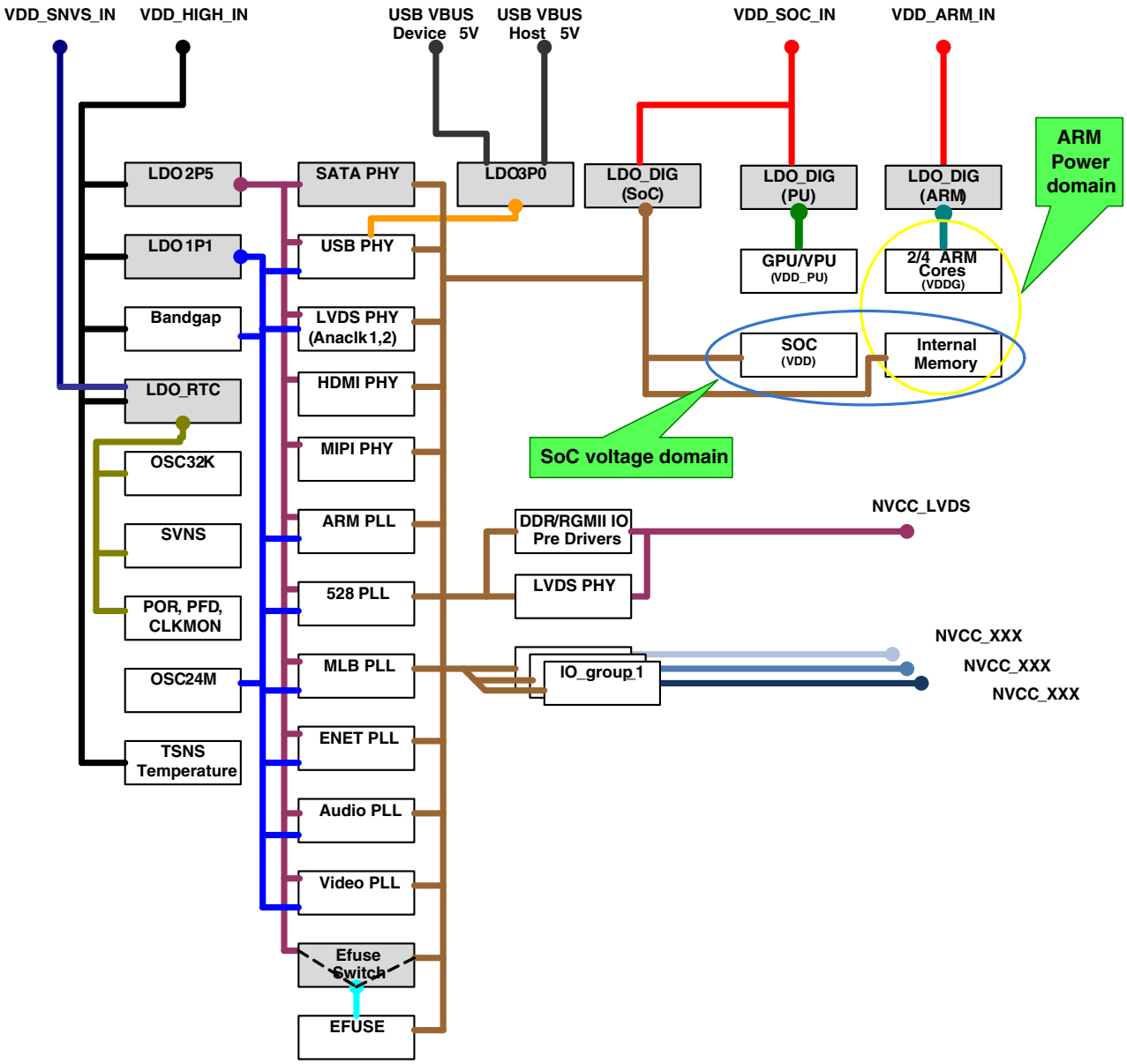


Figure 10-6. i.MX 6Dual/6Quad Power Tree

The integrated PMU includes the following components:

- Three Digital LDO regulators
- Two Analog LDO regulators
- USB LDO
- SNVS regulator
- Reverse well biasing

See Power Management Unit (PMU) for further details on integrated PMU functional description and programmability.

10.4.1.1.1 Digital LDO Regulators

The integrated PMU includes three digital LDO regulators: LDO_ARM, LDO_PU, and LDO_SOC. These regulators provide power to the ARM_Core power domain, the combined VPU, IPU and GPU power domain, and the rest of the SoC logic (except always-ON SNVS domain).

NOTE

The name "digital" only refers to the type of load. It is not related to the LDO design or feature set.

The digital LDO regulators can operate in the following modes:

- Internal Bypass - The regulation pass device (FET) is switched fully on, passing the external input voltage to the load unaltered. The analog part of the regulator is powered down in this state, removing any loss other than the IR drop through the power grid and the FET. Be aware that a period of time (see datasheet) is required to switch from the internal digital bypass mode to the analog regulation mode. Typically it takes less than 100us. Please refer to PMU for further details on bypass and power gate configuration.
- External Bypass - The input and output of the regulator are shorted externally to the SoC. If operating in this configuration, enable the internal bypass early in the startup sequence before attempting high frequency/high power operation. Be aware that internal power gating is not available in this mode.
- Power Gate - The regulation FET is switched fully off, limiting the current draw from the supply. The analog part of the regulator is powered down, limiting the power consumption. The output voltage will fall to a level where the residual leakage of the power FET balances with the leakage of the load. Power gating is applicable to ARM and PU power domains.
- Analog regulation mode - The regulation FET is controlled such that the output voltage of the regulator equals the programmed target voltage. The target voltage is fully programmable in 25mV steps.

These modes allow the regulators to implement voltage scaling and power gating, and allow bypass when an external high power efficient regulator is used as a direct source for some of the SoC loads.

These digital regulators also feature brownout detection, which is helpful to sense when supplies are starting to collapse. Note that the core will be interrupted on a brownout. Please see details in Miscellaneous Control Register (PMU_MISC2*n*).

For further details of LDO programming and configuration please refer to Digital Regulator Core Register (PMU_REG_CORE).

The power management system is built under assumption that in typical applications the single (and simple) shared power supply will be used for ARM core domain and SoC domain. The combined load gains some efficiency, especially in low power modes and saves BoM significantly.

The DVFS in a typical cost/complexity optimized application is considered by mean of internal LDO. In"full speed" modes LDO bypass is considered in both domains. The dynamic voltage scaling to low load workpoints for ARM domain is implemented by programming associated LDO.

In highly power-optimized systems, it is possible to use multiple external DCDC buck converters and bypass internal LDO for high power domains. The obvious trade-off is in the increased complexity of the external power supply components and the associated increase in the BoM and board design complexity.

10.4.1.1.2 Analog LDO regulators

There are two analog LDO regulators used for general system purposes:

- LDO_1P1 - The LDO_1P1 (VDD_HIGH_IN, NVCC_PLL) linearly regulates down a higher supply voltage (2.8V-3.3V) to produce a nominal 1.1V output voltage. This regulator supplies digital portions of USB PHYs, PLLs, and the internal 24MHz oscillator.
- LDO_2P5 - The LDO_2P5 (VDD_HIGH_IN, VDD_HIGH_CAP) linearly regulates down a higher supply voltage (2.8V-3.3V) to produce a nominal 2.5V output voltage. The regular 2.5V LDO is combined with an alternate self-biased low-precision weak regulator which can be enabled for applications that need to keep the 2.5V output voltage alive during low power modes, where the main regulator and its associated global bandgap reference module are disabled to save power. The output of this weak-regulator is not programmable and is a function of its input power supply as well as its load current. Typically with a 3V input power supply, the weak-regulator output is 2.525V and its output impedance is approximately 40Ohm. Special procedure is recommended to move load back and forth between the main and low power regulators. This regulator supplies most of the analog circuitry of the integrated PHYs, special I/Os (LVDS I/O, DDR I/O), and other analog and mix signal components integrated into the SoC.

10.4.1.1.3 USB LDO

The USB_LDO linearly regulates down the USB VBUS input voltages (typically 5V) to produce a nominal 3.0V output voltage. This regulator has a built in power-mux that allows the user to run the regulator from either one of the VBUS supplies when both are present. If only one of the VBUS voltages is present, the regulator automatically selects that supply. Current limit is also included to help the system keep the in-rush current within limits as required in USB 2.0 specification. This regulator supplies only low speed and full speed transceivers of USB PHYs.

10.4.1.1.4 SNVS regulator

The SNVS regulator takes the SNVS_IN supply and generates the SNVS_CAP supply, which in turn powers the real time clock and low power section of the SNVS blocks. If VDDHIGH_IN is present, then the SNVS_IN supply is internally shorted to the VDDHIGH_IN supply to allow coin cell recharging if necessary.

10.4.1.1.5 Reverse well biasing

The reverse well biasing module on the SoC consists of a self-clocked/self-regulating charge-pump circuit, used to generate a negative bias voltage for the floating PWELL, and a low-power regulator. The low-power regulator is used to generate a positive bias voltage for the NWELL of the digital logic cells on the SOC power domain. Static leakage reduction can be achieved during SoC low-power modes through the use of these reverse well bias voltages. Please refer to the CCM_CLPCR register in CCM Memory Map/Register Definition and PMU_MISC0 in PMU Memory Map/Register Definition for details about well bias control and Static " for functional description of the reverse well biasing.

10.4.1.2 GPC - General Power Controller

The GPC block provides hardware assistance to Dynamic Voltage Frequency Scaling (DVFS) and power gating, and includes the sub-blocks listed here.

- DVFS load tracking block - This block allows hardware tracking on the core load and generates an interrupt when a frequency change is requested. It does not generate any request for voltage and/or frequency changes made by a hardware signal. The frequency/voltage changing process requires interaction with the CCM block, as well as either the integrated PMU modules or the external programmable regulator. This process should be completed by either the CPU interrupt routine or a DMA transaction.
- Power Gating Controller (PGC) - This sub-block of GPC has the following functions:

- Provides the user with the ability to switch off power to a target subsystem.
- Generates power-up and power-down control sequences. This includes interaction with CCM/LPCG and SRC, and control for clock and reset generation for power domains affected by power gating.
- Provides programmable registers that adjust the timing of the power control signals.
- Controls the CPU power domain and the combined GPU/VPU power domain.
- Wake-up interrupt controller - This controller initiates the system wake-up from low power modes when only low frequency real time clock remains active, and thus the Generic Interrupt Controller (GIC) can not handle synchronous interrupt signals. Additional features are as follows:
- Supports up to 128 interrupts
- Provides an option to mask/unmask each interrupt
- Detects interrupts and generates the wake up signal

See General Power Controller (GPC) for further details on GPC, its sub-blocks, and information on its functional description and programmability.

10.4.1.3 SRC - System reset Controller

The reset controller is responsible for the generation of all reset signals and boot configuration decoding.

It determines the source and the type of reset, such as POR, WARM, and COLD, and performs the necessary reset signal qualifications. SRC is capable of generating reset sequences in the following conditions:

- in interaction with external PMIC, based on external POR_B signal and "power ready" signals generated by the integrated PMU
- or in interaction with the integrated PMU only, based on its "power ready" signal.

Based on the type of reset, the reset logic generates the reset sequence for either the entire SoC or for the blocks that are power-gated.

See System Reset Controller (SRC) for further details on SRC functional description and programmability.

10.4.1.4 Power domain(s)

A power domain is a group of blocks or sub-blocks fed by power sources controlled by the same power controls in GPC.

Some power domains can be split into a logic sub-domain and a memory sub-domain. The memory sub-domain in such case may contain two entities:

- Memory array(s) - Powered by a dedicated voltage rail enabling memory retention while core is OFF.
- Memory interface logic - Powered by the same voltage source as the logic sub-domain of the power domain.

Signals crossing power domain boundaries or sub-domain boundaries are passed through proper isolation and/or level-shifting cells to ensure robust operations of the SoC when some of domains are power gated or working at a reduced voltage.

Figure below shows the power domain interface in the i.MX 6Dual/6Quad system

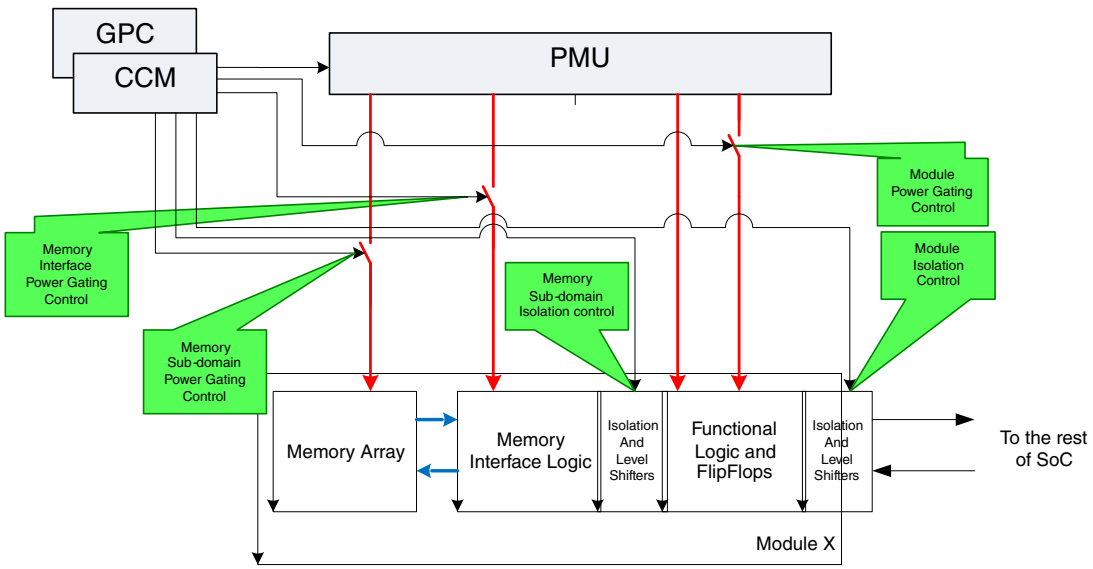


Figure 10-7. Power domain interface in i.MX 6Dual/6Quad system

10.4.1.4.1 Power distribution

The i.MX 6Dual/6Quad power distribution tree is comprised of multiple power domains. The main power domains are:

- ARM - The ARM domain contains the ARM Core platform (except for memory arrays and interface logic). This domain can be supplied either from an integrated power supply or from an external controllable regulator, preferably high efficiency DCDC converter.
- ARM Memory array - Memory arrays are connected to a separate and dedicated power domain (separated from the Main logic and ARM domain). In normal operation mode (functional, non-DVFS mode), the memory arrays domain voltage level should be kept equal to (same as) the rest of the core logic domains (Main,

ARM). Please refer to the Datasheet for further information about voltage level difference between domains allowed in different power modes.
- Combined GPU and VPU domain - The combined GPU/VPU domain contains all GPU3D and GPU2D engines and VPU engine.
- SNVS/RTC low power domain - The SRTC domain contains only counter, comparator and compared data of the on-chip RTC. This domain should be supplied from an external single cell LiION battery and/or an external pre-regulated power supply.
- Analog domain - The analog domain contains the PLLs, LDOs and USB PHY. The domain supplies should be constant to allow continuous clock during any dynamic voltage scaling techniques. The digital supply should be provided from an internal regulator, and can be combined with the memory array supply. The analog supply should be provided from internal low noise regulator.
- Main SoC logic - The main SoC logic domain contains the rest of the logic of the SoC.

From a DVFS and Power Gating standpoint, the following digital logic domains are affected:

- Cortex-A9 Core Platform - DVFS and power gating.
- ARM Cortex-A9 memories - Power gating only.
- GPU3D and VPU - Power gating only.

See table below for details of the i.MX 6Dual/6Quad system power domains layout and dependencies.

10.4.1.4.2 Domain Memory and domain logic state retention in case of Power Gating

The following is the list of relevant memories and logic domains with the description of their state-retention support:

- Cortex-A9 Core Platform is sub-divided into three sub-domains listed below:
- Cortex -A9 Core Platform logic: The software state retention for all logic is implemented in this domain. That means that the content of relevant registers should be be stored in some memory retaining its state (L2 cache for example) while the logic domain is power-gated. Details on how to implement the software retention can be found in the Cortex-A9 Core Platform TRM.
- Cortex-A9 Core L1 memories - No retention. The L1 memories have a dedicated supply on the package (VDD_CACHE_CAP) which should be connected to the Cortex-A9 Core Platform supply. The L1 cache should be flushed prior to power

gating in order to allow powering up of the CPU at the same state as before power gating.
- Cortex-A9 Core L2 memories - hardware state-retention since its supplies are driven by the SoC supplies.
- GPU3D, GPU2D and VPU: These three modules can be power gated (together) independently of Cortex-A9 power gating, because it resides on a separate power domain. State retention is supported neither for GPU3D, GPU2D and VPU logic nor for their internal memories.
- ANALOG PHY IPs - PCIe, SATA and HDMI - hardware state-retention since its supplies are driven by non-gated supplies.
- SoC - hardware state-retention in Standby mode. Reverse Well Biasing is applicable for this domain.
- SNVS_LP - hardware state-retention even when SoC supplies are removed.

10.4.1.4.3 Power Gating Domain Management

The following bullets provide the sequence required for power-gating the relevant power-domains:

10.4.1.4.3.1 Cortex-A9 Core Platform

1. Copy through software all the Core configuration registers to a powered-on memory
2. Configure the GPC/PGC CPU registers in PGC Memory Map/Register Definition as follows to power-down the core on the next "WFI" instruction:
   - Configure the GPC/PGC PGC_CPU_PDNSCR Register ISO and ISO2SW bits. These bits determine the delay between the power-down request to enabling the platform isolation and the platform isolation to the actual power-off switch to the supplies accordingly.
   - Configure the GPC/PGC PGC_CPU_PUPSCR Register SW and SW2ISO bits. These bits determine the delay between the power-up request to the actual power-up of the supplies and the last to the platform isolation disabling.
   - Configure the GPC PGC PGC_CPU_CTRL PCR bit to allow the power down of the platform
   - Cortex-A9 Core Platform should execute a "WFI" instruction.

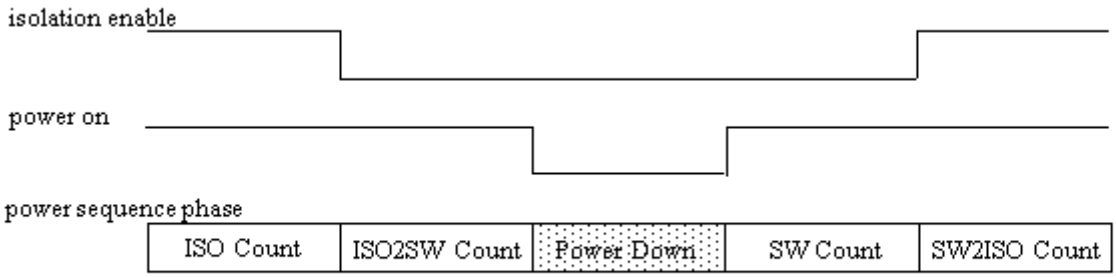


Figure 10-8. Cortex-A9 Core Platform isolation and power on switch flow

10.4.1.4.3.2 GPU3D, GPU2D and VPU

1. Configure the CCM CGR bits (CCM Memory Map/Register Definition) to disable the GPU3D, GPU2D and VPU clocks.
2. Configure the GPC/PGC Registers (GPC Memory Map/Register Definition) as follows to power-down isolate the GPU3D, GPU2D and VPU logic from the rest of the SoC logic:

Configure the GPC/PGC PDNSCR Register ISO bits. These bits determine the delay between the power-down request to enabling the LDO domain isolation.

Configure the GPC/PGC PUPSCR Register SW2ISO bits. These bits determine the power-up request to the LDO domain isolation disabling.

Configure the GPC/PGC CTRL[PCR] bit to allow the power down of the block.

Configure the GPC/PGC GPC_CNTR to power down GPU3D, GPU2D and VPU

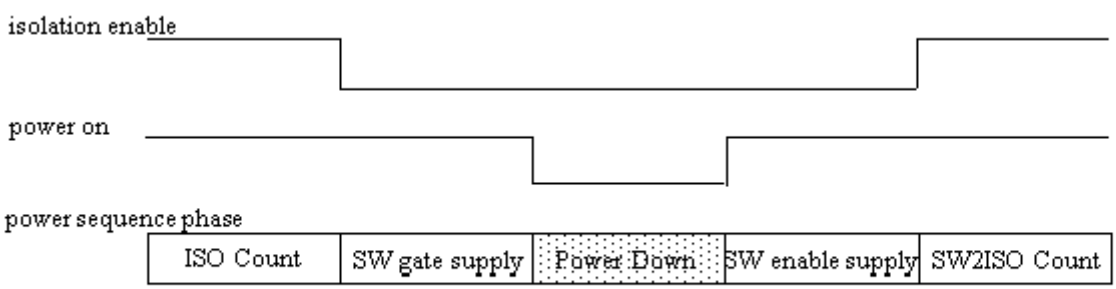


Figure 10-9. GPUs and VPU isolation and power on switch flow

10.4.1.4.3.3 SoC

For additional power reduction it is possible to do the following:

- Power-down the internal oscillator by configuring the following bits CCM_CCR[COSC_EN] (CCM Memory Map/Register Definition). This can be done only in case there is no dependency on 24MHz XTAL for wake-up.
- Enable reverse well biasing by configuring the CCM_CLPCR[WB_PER_AT_LPM] bit (CCM Memory Map/Register Definition).
- It is possible to turn off and turn on the PMIC supplies to the SoC even when the SoC supplies are off. Since SNVS_LP is powered through an "always on" supply, configuring the SNVS_LP DP_EN to "1" allows changing the PMIC_ON_REQ pad (SoC on/off supply indication to the PMIC) through the ONOFF pad.

10.4.1.4.4 Power Gating domain dependencies

There are 3 power domains that need to be isolated in different power-down cases:

- Cortex-A9 Core Platform - Isolation needs to be enabled before power-down. This is taken care of automatically once CCM and PGC are configured and the Cortex-A9 Core Platform executes the "WFI" instruction.
- GPU3D, GPU2D and VPU - Isolation needs to be enabled before power-down. This should be taken care through software configuration of the PGC.
- SNVS_LP - Different from the 2 cases above the SNVS_LP isolation isolates the signals coming from the SoC to the SNVS_LP. This is required for saving the contents of the SNVS_LP. (such as the real-time clock) The isolation is activated in 2 ways:
- Automatically through the power-fail detector in the PMU
- Through software configuration

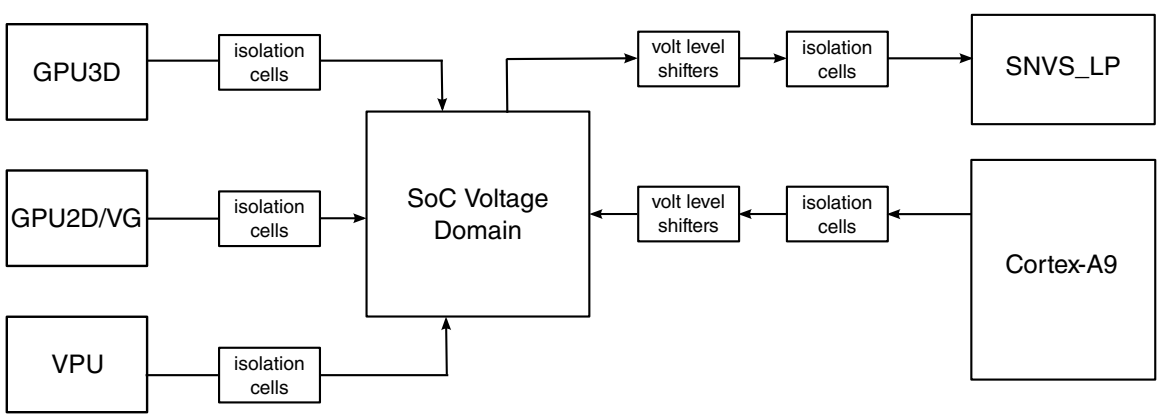


Figure 10-10. Isolation cells and Voltage level shifters placing

10.4.1.5 Voltage domains

The list found here states the different voltage domains and their scalability in regarding to power-saving in dynamic and static scenarios.

- ARM - Cortex-A9 Core Platform including L1 cache - Scalable voltage in both dynamic and static scenarios
- SoC and PU LDO Domains - Scalable voltage only in static scenarios
- ANALOG components including SATA, PCIe, HDMI, MIPI, LVDS and PLLs - Fixed voltage
- I/O - Fixed voltage
- SNVS_LP - Fixed voltage

10.4.1.6 Voltage domain management

10.4.1.6.1 Dynamic

10.4.1.6.1.1 DVFS

Dynamic Voltage and Frequency Scaling (DVFS) is a well-known technique to reduce power consumption in mobile devices. In order to improve power saving efficiency, DVFS is applied on the ARM core voltage domain.

In this scenario the Dynamic Voltage and Frequency Scaling (DVFS) block is used. More details can be found in DVFS-CORE (DVFS). The DVFS block can monitor separately the IDLE indication of each of the Cortex-A9 Cores. The DVFS performs the following:

- Simple, non-overlapping averaging providing a level-based average index of the tracked CPU load
- Sums the CPU load and the load detected from additional load indicators weighted according to software configurations
- Calculates an exponential moving average of the tracked load
- Provides up, down and "panic" threshold comparators and counters for generating interrupts to the Cortex-A9 Core Platform
- Frequency pattern generator is able to manage the frequency update requests periodically

The DVFS interrupts are then forwarded to the Cortex-A9 Core Platform. The Cortex-A9 Core Platform reacts according to the DVFS interrupt types. The following are 2 examples for handling different DVFS interrupt types:

- Upper threshold is reached, meaning that the Core is heavily loaded and the core frequency needs to be increased according to the following flow:

- Configure PMU to raise the core voltage
- Wait until the voltage is stable
- Configure CCM to raise the frequency.
- Lower threshold is reached, meaning that the Core is "IDLE" most of the time and the core frequency can be decreased according to the following flow:
- Configure CCM to reduce the frequency.
- Configure PMU to lower the core voltage.

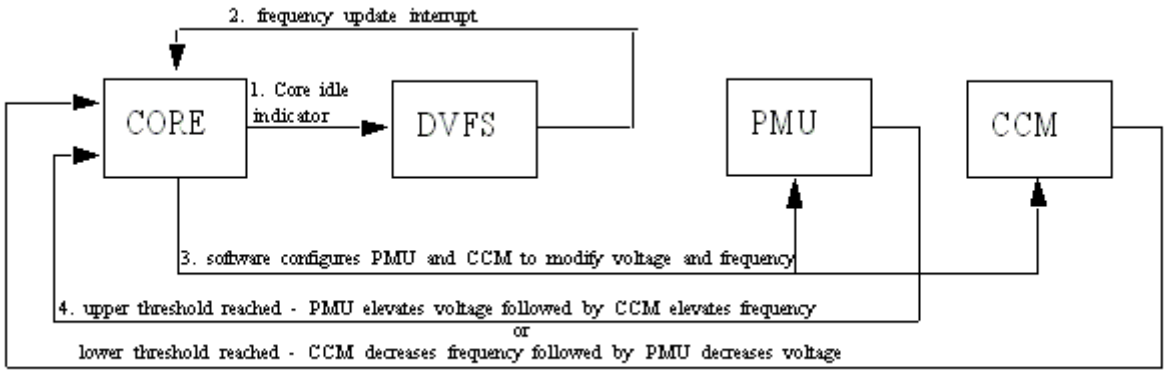


Figure 10-11. High Level DVFS working flow

NOTE

PMU can not be used to change the voltage in case the LDO is bypassed.

10.4.1.6.1.2 Voltage Scaling

A simplistic way to reduce power consumption in dynamic scenarios is to scale down the ARM, SoC and PU LDOs voltage according to the allowed voltage points and corresponding frequencies specified in datasheet.

10.4.1.6.2 Static

10.4.1.6.2.1 Standby Leakage reduction (SLR)

Standby leakage reduction is a power-management technique utilizing:

- Reduced supply voltage for relevant domains
- Reverse well-biasing in STOP, WAIT, or Deep Sleep Mode (DSM) modes

With SLR, the device switches into low-power active system modes automatically or in response to user requests during system Stop, Wait, or DSM modes (that is, in situations when no application is started and no system activity is presented).

When applying SLR, the system remains in the lowest static power mode while retaining logic and memory states. This technique trades static power consumption for wake-up latency while maintaining fast system response time suitable for most applications.

See CCM Control Register (CCM_CCR), CCM Low Power Control Register (CCM_CLPCR) and PMU Miscellaneous Register 0 (PMU_MISC0) for further details on SLR programmability options.

The following describes the flow for applying standby voltage and reverse well bias to the SoC:

- Configure the external PMIC standby voltage, refer to chip datasheet.
- Configure CCM_CCR[RBC_EN] bits to bypass and disable PMU regulators in the next ARM "WFI" execution.
- Configure CCM_CCR[REG_BYP_COUNT] bits to allow proper voltage restoration by the external PMIC when exiting standby.
- Cortex-A9 Core Platform executes the "WFI" instruction that completes the software sequence putting the SoC into low power mode
- After that, the reverse well bias will be applied automatically (if enabled) with appropriate delay. Please refer to CCM_CLPCR and CCM_CCR Registers in CCM for further information on reverse bias enabling and counters configuration.

10.4.1.6.2.2 ANALOG PHYs IPs -

The PCIe, SATA and HDMI analog PHYs can also be configured to consume less power when the are in a non-active state. Details on how to put the IPs in low power mode can be found in each of the IP documentation. Further reduction can be achieved by settings the ENABLE_WEAK_LINREG in the PMU PMU_REG_2P5 register.

10.4.1.6.3 Voltage domain dependencies

When applying voltage changes for power saving the following voltage domain limitations should be taken into account:

- In dynamic scenarios the SoC and PU voltage supplies should not differ
- In dynamic scenario's the ARM supply should not be higher than the SoC supply by more the 50mV.
- In dynamic scenarios VDDCACHE_CAP should not be above VDDARM_CAP by more than 200mV and VDDARM_CAP should not be above VDDCACH_CAP by more than 50mV

The above limitations are also mentioned in datasheet.

10.4.1.6.4 IO voltage

10.4.1.7 System domains layout

The following table describes the different power modes.

NOTE

Wakeup time is the hardware perspective, and doesn't reflect the time it takes software to resume drivers and perform system operations.

Table 10-1. Aimed power modes

Mode	Description (Status of main power domains)	Wake-up capability	Wakeup time	Applicable use case
RUN	Power supplies are on, all clocks are on.	N/A	N/A	All
WAIT	Power supplies are on. ARM executes WFI command, Option to gate off clocks for specific modules based on programmable bits.	Based on interrupt from any module	Immediate (ARM exit WFI)	IDLE process of operating system.
STOP	Power supplies are on. ARM executes WFI command. VPU and GPU are in power gating PLLs are off, hence all system clocks are off.	Based on interrupt from modules that can track operation with low frequency clock- like Keypad press, GPIO, Timer, CAN activity, etc.	~50us	Suspend state of operating system - system is on but is waiting for operation.
Standby	- PLLs disabled - Low voltage could be applied to ARM and SoC power inputs - Chip regulator bypass if reduce supply voltage, - Well Bias could be activated - CPU regulator disabled for power gating - PU regulator is disabled in software - xtal enabled	Same as above	~220 us	Low power standby mode with limited reaction on external events
Deep Sleep	Same as above but with OSC24M disabled	Interrupts from modules those could generate asynchronous interrupts (Keypad, GPIO, SNVS)	~1400us	

Table continues on the next page...

Table 10-1. Aimed power modes (continued)

Mode	Description (Status of main power domains)	Wake-up capability	Wakeup time	Applicable use case
SRTC	Only SNVS domain and OSC32K keep alive	SRTC security alert / SRTC timer expire. Based on each one of those, SRTC module will generate request to PMIC to perform POR sequence and walk-up the system. The system will go through boot process. PMIC can also receive press on "ON" button to walk-up the system. This should be connected directly to PMIC.	~1700us to WARM boot COLD boot, depends on boot procedure and boot device	

There is a single hardware signal coming into PMU which sets the PMU in either of two "STOP" states. The STOP state is implemented is controlled by the PMU_MISC0[STOP_MODE_CONFIG] bit (See PMU Memory Map/Register Definition). It is recommended that the blocks be configured for safe powerdown/up through the registers before asserting the stop_mode signal. Blocks not described in the section below are unaffected by stop_mode.

If the stop_mode_config is set to zero, thus in the STOP mode all blocks powered down in minimum power configuration.

If the stop_mode_config is set to one, thus in the STOP mode some of the blocks remain powered and in different states as defined in the table below.

Table 10-2. STOP mode configuration

Block	STOP_MODE_CONFIG=0	STOP_MODE_CONFIG=1
reg1p1	off	on
reg2p5	off	on
reg3p0	off/on depending on vbus. Uses crude local reference if vbus is present	off/on depending on vbus . Uses analog central bandgap if VBUS is present.
reg_core	bypassed if not power gated.	bypassed if not power gated
reg_pu	bypassed if not power gated.	bypassed if not power gated.
reg_soc	bypassed	bypassed
bandgap	off	functional
temp_sensor	off	off
well_bias	hardware controlled	hardware controlled
All PLLs	off	off
OSC24M	off	Controlled by CCM configuration
LVDS clock I/O CLK1 and CLK2	off	off

10.4.2 Power management techniques

The device supports the power-management techniques with the features found here.

- Partitioning of the device into voltage, power, clock, and reset domains
- Domain isolation that allows flexible configurations of domains on/off states to form use cases targeting various applications
- Clock tree with selective clock-gating conditions and almost independent clock roots
- Power, reset, and clock control hardware mechanism to manage sleep and wake-up dependencies of power domains
- Software-controllable and hardware-controllable clock gating for functional modules and buses
- Memory retention and state retention capability (Software State Retention for ARM A9) for preserving memory contents and device state in low-power modes
- Dynamic Voltage and Frequency Scaling (DVFS) support for the ARM A9 processor cluster
- Support for low-power device modes input/output (I/O) pad configuration for minimum power
- Variety of operating modes to optimize device performance and wake-up times
- Thermal monitoring and thermal aware performance management

Many of the low power features are fully or partially software controllable and can be configured for the specific requirements of a target system.

Combining these techniques, the system designer may meet tight requirements of low-power standby and operational modes while maintaining high performance for time-critical tasks.

10.4.2.1 Power saving techniques

The table below lists power saving techniques supported by the SoC in their connection to different components of power consumption.

Table 10-3. Power saving design/architecture and power saving techniques

Techniques	Active SoC Power	Standby SoC Power	System Power
Temperature Monitoring, and active frequency throttling	√		
Cortex-A9 Core Platform DVFS	√		
Cortex-A9 Core Platform SRPG (Software)		√	

Table continues on the next page...

Table 10-3. Power saving design/architecture and power saving techniques (continued)

Techniques	Active SoC Power	Standby SoC Power	System Power
Cortex-A9 Core Platform Power Gating		√	
Cortex-M4 Asymmetric multicore	√	√	√
VPU and GPU3D Power Gating	√[1]	√	
Clock gating (automatic dynamic and forced)	√		
Integrated PMU (IR drop, efficiency, accuracy)	√		√
C4 package (IR drop, thermal)	√		
Display Backlight optimization (IPU, SW)			√
Architecture: L2 cache, Video / Audio / Graphics acceleration	√		
Architecture: SATA, PCIe, HDMI, USB, LVDS integration			√
Low Power DDR: LPDDR2, LV-DDR3			√

1. Applicable to use cases where GPU3D and VPU operation is not required.

10.4.2.2 Thermal-aware power management

The temperature sensor block (TEMPMON) implements a temperature sensor/conversion function. The block features an alarm function that can raise an interrupt signal if the temperature is above a specified threshold.

Software may implement temperature aware DVFS for the ARM domain and the GPU domain, as well as temperature aware frequency scaling for other system components to ensure that both the frequency and voltage is lowered when the die temperature is above the specified limit.

Software may also implement temperature aware task scheduling to ensure that non-critical tasks are suspended when the die temperature is above the specified limit.

See Temperature Monitor (TEMPMON) for further details on temperature monitor functions and programmability options.

10.4.2.3 Peripheral Power management

10.4.2.3.1 Main memory power management

Main system memory, DDR3, and LPDDR2 are some of the most power-hungry system components, but the SoC provides several options to manage DDR power.

Automated power saving modes are supported by the MMDC hardware. This feature allows the DDR memory to automatically enter self-refresh mode when there are no DDR accesses for a configurable time. The default setting is 1024 clock cycles which can be optimized based on the customer use case and application.

See Power Saving and Clock Frequency Change modes for further details on MMDC power saving features and programmability options.

Software may support DDR frequency scaling. Automated frequency changing procedure is supported by MMDC and CCM modules.

NOTE

DDR frequency changes cost extra time and power. Slowing requestors while keeping DDR at full speed may increase total system power. Software may also implement Cooperative Dynamic Frequency Scaling in order to keep the system balanced, (that is, keep the system in balance when DDR throughput is equal or slightly higher than total amount of requests generated by all requestors).

Reducing the DDR frequency while in DLL-ON mode may be not efficient because:

- Reduction in DDR frequency will cause bus duty cycle to increase and thus reduces chance of automatic MMDC power saving (place memory into SR).
- Total amount of read/write operation does not change (power is per-operation).
- The termination is active longer, though, lowering frequency from 528 MHz to 400 MHz or below may enable lowering drive strengths and termination.

When possible at lower performance use cases, software may switch DDR3 to DLL-off mode. This allows it to greatly reduce DDR3 frequency and thus disable or reduce termination and drive strength, which significantly reduces the power consumption of the DDR3 interface.

A good strategy for many types of workloads is to combine most activity in bursts (natively possible, for example, for typical multimedia applications, communication, etc.) and run this segment at maximal speed, then switch to DLL-OFF mode to support background activity (communication, display refresh, housekeeping).

The DRAM Interface power dissipation depends on many variables, however, proper termination and drive strength is key for power and thermal performance. Memory and controllers provide a host of programmable options for the drive strength of the output buffers and for the on-die termination impedance.

The ideal settings for drive strength and ODT also depend on the clock frequency to ensure that inter-symbol interference (ISI) effects are not introduced.

DDR PHY power is proportional to the amount and type of bus activity. For more data on ODT savings, please refer to the Application Note, AN4509 i.MX 6Dual/6Quad Power Consumption.

In cases where the DDR is placed into self-refresh, software can configure DDR I/O to be floated or lowered to the minimum drive allowed by JEDEC.

Modifying the DDR drive strength must be done by code that is executing from a memory region other than DDR (for example, IRAM). No access to DDR (including page table walks, cache misses, alternate bus master accesses, etc.) is allowed while the DDR I/O pads are being re-configured.

10.4.2.3.2 Video-Graphics system power management

10.4.2.3.3 IO power reduction

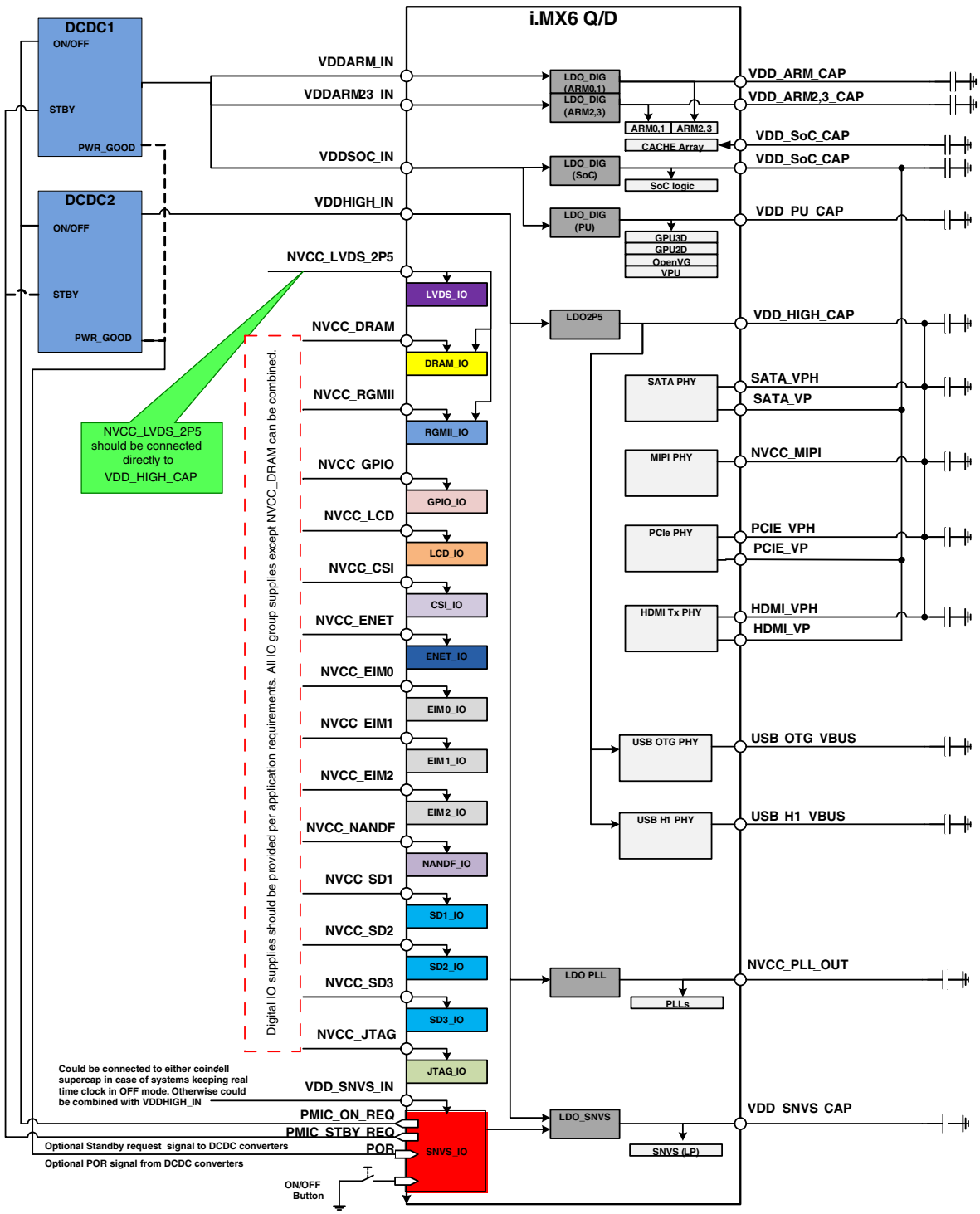
SW should configure IO to low power modes:

- PHYs - make sure that all unused PHYs are placed to lowest power state. Please refer relevant chapter for further information about different PHYs
- Digital IOs - Make sure all unnecessary PU/PD are disabled and IO are switched to either minimal drive strength or to input mode (when applicable)
- Set DDR type IO to CMOS mode if possible, specifically RGMII segment
- Disable all unused LVDS pads (ANALOG, MLB PHY, LDB)

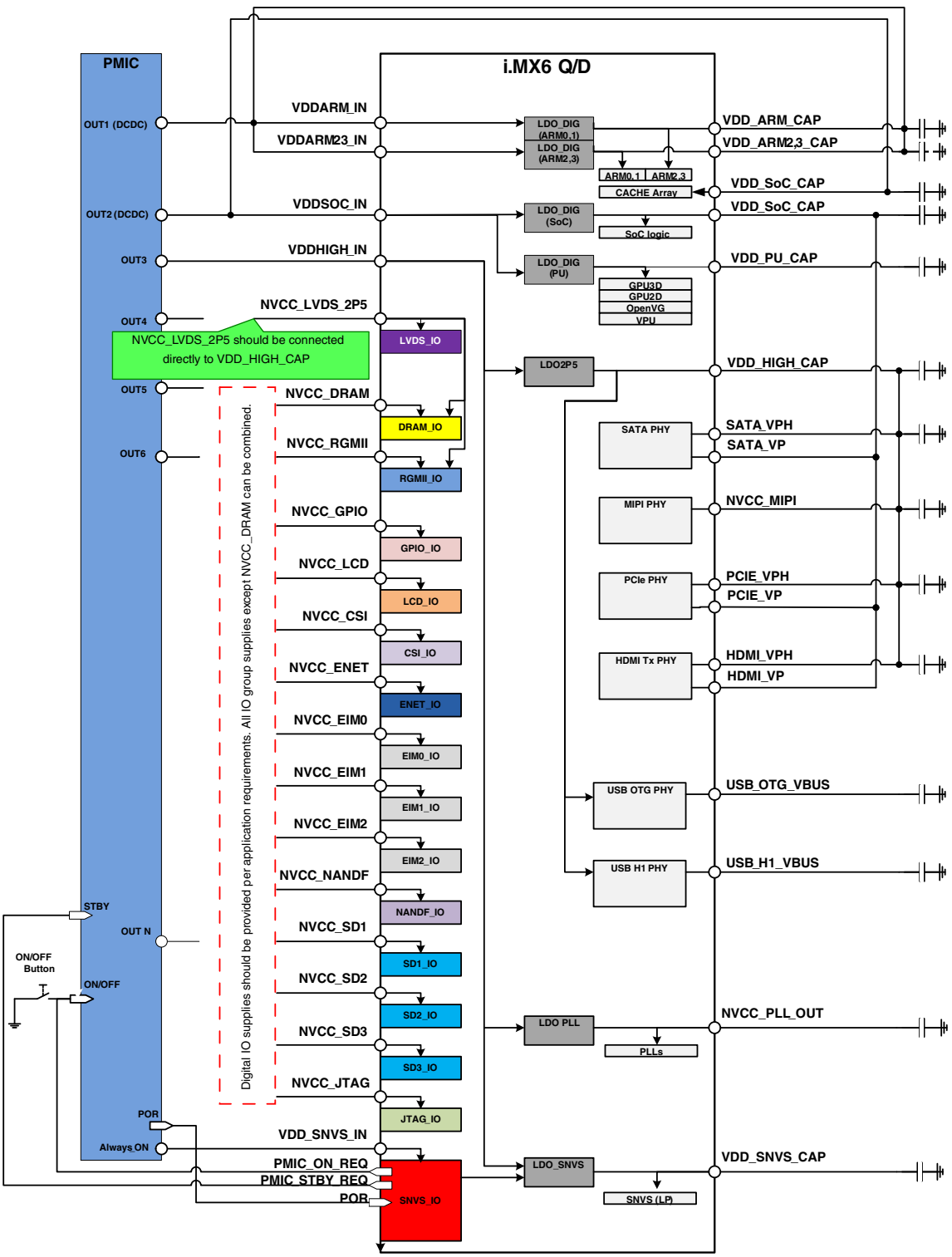
10.4.3 Examples of External Power Supply Interfacing in the i.MX 6Dual/6Quad based systems

This section presents the examples of external power supply interfacing to the chip.

The scenario based on integrated PMU system is presented below. This scenario minimizes BoM and board design complexity.


i.MX6 Q/D
DCDC1
ON/OFF
STBY
PWR_GOOD
DCDC2
ON/OFF
STBY
PWR_GOOD
VDDARM_IN
VDDARM23_IN
VDDSOC_IN
VDDHIGH_IN
NVCC_LVDS_2P5
NVCC_DRAM
NVCC_RGMII
NVCC_GPIO
NVCC_LCD
NVCC_CSI
NVCC_ENET
NVCC_EIM0
NVCC_EIM1
NVCC_EIM2
NVCC_NANDF
NVCC_SD1
NVCC_SD2
NVCC_SD3
NVCC_JTAG
VDD_SNVS_IN
PMIC_ON_REQ
PMIC_STBY_REQ
POR
ON/OFF Button
LVDS_IO
DRAM_IO
RGMII_IO
GPIO_IO
LCD_IO
CSI_IO
ENET_IO
EIM0_IO
EIM1_IO
EIM2_IO
NANDF_IO
SD1_IO
SD2_IO
SD3_IO
JTAG_IO
SNVS_IO
LDO_DIG (ARM0,1)
LDO_DIG (ARM2,3)
ARM0,1
ARM2, 3
CACHE Array
LDO_DIG (SoC)
SoC logic
LDO_DIG (PU)
GPU3D
GPU2D
OpenVG
VPU
LDO2P5
LDO PLL
PLLs
LDO_SNVS
SNVS (LP)
SATA PHY
MIPI PHY
PCIe PHY
HDMI Tx PHY
USB OTG PHY
USB H1 PHY
VDD_ARM_CAP
VDD_ARM2,3_CAP
VDD_SoC_CAP
VDD_SoC_CAP
VDD_PU_CAP
VDD_HIGH_CAP
SATA_VPH
SATA_VP
NVCC_MIPI
PCIE_VPH
PCIE_VP
HDMI_VPH
HDMI_VP
USB_OTG_VBUS
USB_H1_VBUS
NVCC_PLL_OUT
VDD_SNVS_CAP
NVCC_LVDS_2P5 should be connected directly to VDD_HIGH_CAP
Digital IO supplies should be provided per application requirements. All IO group supplies except NVCC_DRAM can be combined.
Could be connected to either coindell supercap in case of systems keeping real time clock in OFF mode. Otherwise could be combined with VDDHIGH_IN
Optional Standby request signal to DCDC converters
Optional POR signal from DCDC converters

The scenario based on external programmable regulators is presented below: This scenario could be used in highly power optimized systems.


PMIC
i.MX6 Q/D
OUT1 (DCDC)
OUT2 (DCDC)
OUT3
OUT4
OUT5
OUT6
OUT N
STBY
ON/OFF Button
ON/OFF
POR
Always ON
VDDARM_IN
VDDARM23_IN
VDDSOC_IN
VDDHIGH_IN
NVCC_LVDS_2P5
NVCC_LVDS_2P5 should be connected directly to VDD_HIGH_CAP
NVCC_DRAM
NVCC_RGMII
NVCC_GPIO
NVCC_LCD
NVCC_CSI
NVCC_ENET
NVCC_EIM0
NVCC_EIM1
NVCC_EIM2
NVCC_NANDF
NVCC_SD1
NVCC_SD2
NVCC_SD3
NVCC_JTAG
VDD_SNVS_IN
PMIC_ON_REQ
PMIC_STBY_REQ
POR
Digital IO supplies should be provided per application requirements. All IO group supplies except NVCC_DRAM can be combined.
LVDS_IO
DRAM_IO
RGMII_IO
GPIO_IO
LCD_IO
CSI_IO
ENET_IO
EIM0_IO
EIM1_IO
EIM2_IO
NANDF_IO
SD1_IO
SD2_IO
SD3_IO
JTAG_IO
SNVS_IO
LDO_DIG (ARM0,1)
LDO_DIG (ARM2,3)
ARM0,1
ARM2,3
CACHE Array
LDO_DIG (SoC)
SoC logic
LDO_DIG (PU)
GPU3D
GPU2D
OpenVG
VPU
LDO2P5
LDO PLL
PLLs
LDO_SNVS
SNVS (LP)
SATA PHY
MIPI PHY
PCIe PHY
HDMI Tx PHY
USB OTG PHY
USB H1 PHY
VDD_ARM_CAP
VDD_ARM2,3_CAP
VDD_SoC_CAP
VDD_SoC_CAP
VDD_PU_CAP
VDD_HIGH_CAP
SATA_VPH
SATA_VP
NVCC_MIPI
PCIE_VPH
PCIE_VP
HDMI_VPH
HDMI_VP
USB_OTG_VBUS
USB_H1_VBUS
NVCC_PLL_OUT
VDD_SNVS_CAP

10.5 ONOFF (Button)

The chip supports the use of a button input signal to request main SoC power state changes (i.e. On or Off) from the PMU.

The button is used to power On and Off the SoC using an always-on (e.g., coincell battery-backed) power domain.

- When the chip main power supply is Off, a button press greater in duration than 750 ms asserts an output signal to request power from a power IC to power up the SoC.
- When the chip main power supply is On, a button press between 750 ms and 5 seconds will send an interrupt to the core to request that software bring down the SoC safely. Software may respond to the interrupt by saving the processor state and then setting a control bit that requests to the power IC the removal of the main power supply.
- Button presses greater than 5 seconds, when the SoC is powered, results in a direct hardware power down request signal to the power IC without providing a software interrupt first. This long button press initiates a hardware-enforced mode which is applicable when software is unable to power Off the device.

The button is connected to input ONOFF and the power IC is connected the chip output PMIC_ON_REQ. The chip must have TEST_MODE deasserted. An always-On supply (e.g. coincell) is needed in the system for this feature.

Chapter 11
System Security

11.1 Overview

Security is a common requirement for platforms built using the i.MX 6Dual/6Quad, although the specific needs vary greatly depending on the platform and market. The type and cost of assets to be protected on a portable consumer device are very different from those to be protected on automotive or industrial platforms, and the same applies to the kind of attacks and level of resources threatening those assets. The platform designer must select an appropriate set of counter measures to meet the relevant platform security needs.

The following figure shows a simplified diagram of the security subsystem.

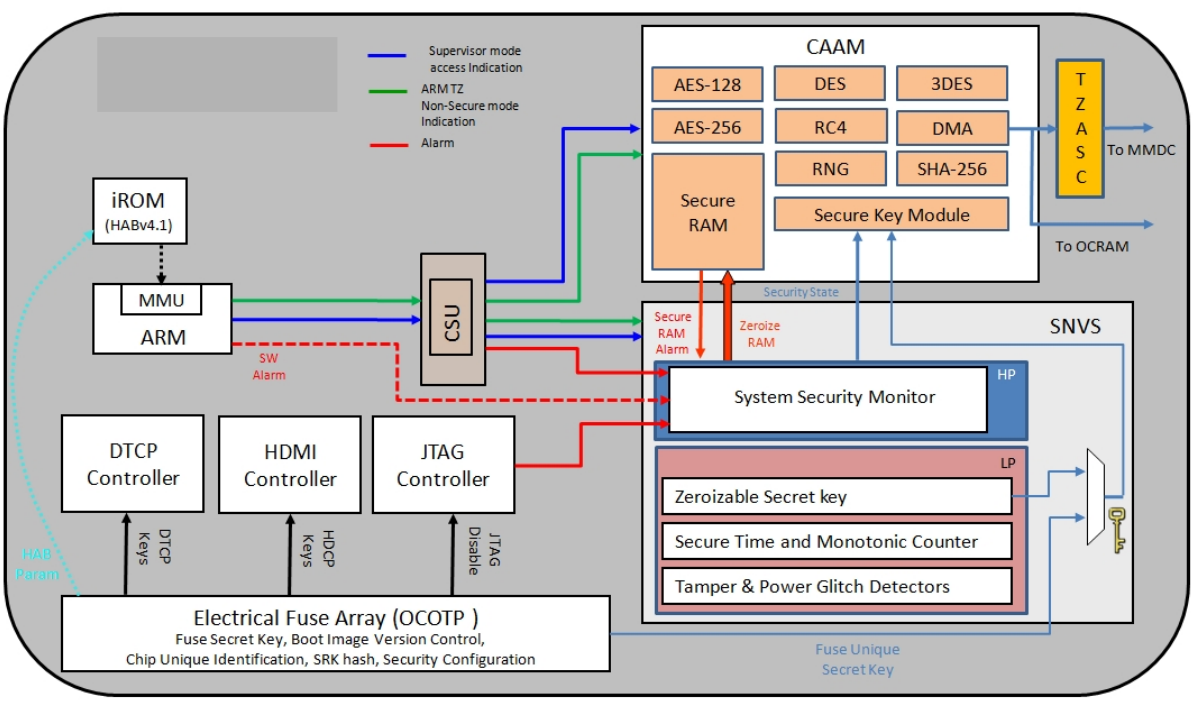


Figure 11-1. Security subsystem (simplified)

For the platform designer to meet the requirements for each market, the i.MX 6Dual/6Quad incorporates a range of security features which can be used individually or in concert to underpin the platform security architecture. Most of the i.MX 6Dual/6Quad security features provide protection against particular kinds of attack and can be configured at various levels according to the required degree of protection. These features are designed to work together and can be integrated with appropriate software to create defensive layers. In addition to protection features, the i.MX 6Dual/6Quad includes a general purpose accelerator to enhance the performance of selected industry standard cryptographic algorithms.

The following is an introduction to the i.MX 6Dual/6Quad security components.

- TrustZone (TZ) Architecture in the ARM Cortex A9 Platform, TrustZone aware Interrupt Controller (GIC) and TrustZone Watchdog Timer (WDOG-2)
- TrustZone Address Space Controller (TZC-380) - providing security address region control functions on DDR memory space.
- On-chip RAM (OCRAM) secure region protection using OCRAM controller.
- High Assurance Boot (HAB) feature in the System Boot
- 16 Kbyte of on-chip Secure RAM
- Encryption algorithms, stream cipher algorithms, hashing algorithms, hardware random number generator and run time integrity checking are all included in the new Cryptographic Acceleration and Assurance Module (CAAM)
- Secure Non Volatile Storage (SNVS)
- On chip OTP (OCOTP) with on-chip electrical fuses
- Central Security Unit (CSU)
- Resource Domain Controller (RDC)
- Secure JTAG Controller (SJC)
- Locked mode in the Smart Direct Memory Access (SDMA) controller
- For DTCP (Digital Transmission Content Protection) adopters, DTCP functionality to support DTCP-MOST and DTCP-IP.

Detailed descriptions of the components are provided in the *Multimedia Applications Processor Security Reference Manual for i.MX 6Dual/6Quad and i.MX 6Solo/6DualLite*.

11.2 Central Security Unit (CSU)

11.2.1 CSU Overview

The CSU manages the system security policy for peripheral access on the SoC. The CSU allows trusted code to set individual security access privilges on each of the peripherals, using one of eight security access privilege levels. Also, according to programmed policy, the CSU may assign bus master security privileges during bus transactions.

11.2.2 CSU Features

The Central Security Unit (CSU) sets access control policies between bus masters and bus slaves, allowing peripherals to be separated into distinct security domains. This protects against unauthorized access to data e.g. when software programs a DMA bus master to access addresses that the software itself is prohibited from accessing directly. Configuring DMA bus master privileges in the CSU consistent with software privileges defends against such attempted accesses.

CSU has the following security related features:

- Peripheral access policy - Appropriate bus master privileges and identity are required to access each peripheral.
- Masters privilege policy - CSU overrides bus master privilege signals, i.e. user/supervisor secure/non-secure, according to access control policy.

11.2.3 CSU Functional Description

The CSU enables secure software to set bus privilege security policy within the platform.

Security policies may be set, and optionally locked in the CSU registers. These privilege values may originate in the command sequence file (CSF) which is processed by the High Assurance Boot (HAB) itself or by an HAB authenticated image which executes after the initial boot ROM phase.

11.2.3.1 CSU Peripheral Access Policy

According to its programmed policy, the CSU determines the bus master privileges and the masters that are allowed to access each of the slave peripherals.

There are four security modes of operation (i.e. bus privileges) in the system distinguished by security (TrustZone/non-TrustZone) and privilege (Supervisor/User) setting of the module. Below is the list of these security modes from the highest security level to the lowest:

- TrustZone (Secure) Privilege (Supervisor) Mode - Highest Security Level
- TrustZone (Secure) non-Privilege (User) Mode - Medium Security Level
- non-TrustZone (Regular) Privilege (Supervisor) Mode - Medium Security Level
- non-TrustZone (Regular) non-Privilege (User) Mode - Lowest Security Level

This functionality is implemented as follows:

The Configure Slave Level (CSL) Register value for a specified peripheral resource defines the output signal -- csu_sec_level for that peripheral. The value of this signal determines by what master privileges a peripheral is accessible. The relationship between the value of the csu_sec_level signal and security operation mode is shown in the table below. The CSL registers reside in the CSU module. Details, describing CSL register fields and how they are programmed to control access privileges for specific peripherals, can be found in the Security Reference Manual (see System Security Overview section).

Table 11-1. Permission Access Table

CSU_SEC_LEVEL[2:0]	Non-Secure User Mode	Non-Secure Spvr Mode	Secure (TZ) User Mode	Secure (TZ) Spvr Mode	CSL register value
(0) 000	RD+WR	RD+WR	RD+WR	RD+WR	8'b1111_1111
(1) 001	None	RD+WR	RD+WR	RD+WR	8'b1011_1011
(2) 010	RD	RD	RD+WR	RD+WR	8'b0011_1111
(3) 011	None	RD	RD+WR	RD+WR	8'b0011_1011
(4) 100	None	None	RD+WR	RD+WR	8'b0011_0011
(5) 101	None	None	None	RD+WR	8'b0010_0010
(6) 110	None	None	RD	RD	8'b0000_0011
(7) 111	None	None	None	None	Any other value

11.3 Cryptographic Acceleration and Assurance Module (CAAM)

11.3.1 CAAM Overview

CAAM is a cryptographic acceleration device that accelerates block encryption algorithms, stream cipher algorithms, hashing algorithms, and random number generation. It has an integral DMA engine that allows CAAM to fetch its command programs, read input data and write the resulting output.

CAAM works with the SNVS to provide platform assurance features, including support for High Assurance Boot, detection of and response to potential tamper events, and shortterm and long-term protection of secret data such as public keypairs, Digital Rights Management keys and proprietary software.

CAAM provides the following features:

- Cryptographic acceleration of hashing, encryption and decryption
- Offloading of cryptographic functions via programmable command descriptor language
- Two independent queues for commands and results
- Register Bus Interface used for configuration, control and status
- DMA, including scatter/gather support for data
- Automatic short-term encryption/decryption of cryptographic keys
- Long-term protection of secret data via device-specific encryption keys
- Secure Memory for access-controlled protection of critical data, and automatic zeroization in the case of tampering
- Random Number Generation using a true random number generator and a NISTcompliant pseudo random number generator

11.4 Secure Non-Volatile Storage (SNVS)

11.4.1 SNVS Overview

SNVS is a hardware device that includes a security state machine and security violation detection circuits that, together with High Assurance Boot software, determine whether the chip is currently in a secure state.

When the security state machine indicates a secure state, the SNVS allows CAAM use of special cryptographic keys to decrypt long-term secrets such as public/private keypairs, Digital Rights Management keys and proprietary software. When the SNVS detects a potential security violation, such as a tamper alert, the SNVS signals CAAM to erase sensitive data such as cryptographic keys, and sends an interrupt to alert the Operating System of the event. The SNVS also includes a general purpose real-time counter.

The SNVS includes the following features:

- Security State Machine driven by High Assurance Boot software and tamper detection circuits

- Master Key Control that protects the integrity and secrecy of the Master Key (OTPMK) stored in fuses
- Tamper detection circuits that detect JTAG events, power glitches, Master Key ECC check failure, and software-reported and hardware-reported security violations
- 256-bit Zeroizable Master Key that can be automatically erased in the event of a security breach
- Tamper-protected Secure Realtime Counter that continues running when the chip is powered off
- Non-volatile Monotonic Counter used to protect against “roll-back” attacks
- Non-volatile General Purpose Register can be used to store a 32-bit value across power cycles
- Non-Secure Real Time Counter with programmable alarm and periodic interrupt

11.4.2 Tamper Detection

Tamper Detection is a special mechanism provided through a chip pin to signal when the device encounters unauthorized opening or tampering.

When not in use, the Tamper Detection signal is pulled-down internally. In case of use, it should be connected to a Tamper Detection contact in a target system (Normally closed, pulled-up to the VDD_SNVS_IN).

An always-ON power supply (coincell battery) should be present in the system. If the tamper detection feature is enabled by software then opening of the tamper contact:

- Switches system power ON with a Tamper Detection alarm interrupt asserted (for software reaction)
- Activate security related hardware (e.g. automatic and immediate erasure of the Zeroizable Master Key and deny access and erase secure memory contents)

11.5 High Assurance Boot (HAB)

HAB, which is the high assurance boot feature in the system boot ROM, detects and prevents execution of unauthorized software (malware) during the boot sequence.

When unauthorized software is permitted to gain control of the boot sequence, it can be exploited for a variety of goals, such as exposing stored secrets; circumventing access controls to sensitive data, services, or networks; or repurposing the platform. Unauthorized software can enter the platform during upgrades or reprovisioning, or when booting from USB connections or removable devices.

HAB protects against unauthorized software by:

- Using digital signatures to recognize authentic software. This allows the user to boot the device to a known initial state, running software signed by the device manufacturer.

11.6 System JTAG Controller (SJC)

The JTAG port provides debug access to hardware blocks, including the ARM processor and the system bus. This allows program control and manipulation as well as visibility to the chip peripherals and memory.

The JTAG port must be accessible during initial platform development, manufacturing tests, and general troubleshooting. Given its capabilities, JTAG manipulation is a known attack vector for accessing sensitive data and gaining control over software execution. System JTAG Controller (SJC) protects against the whole range of attacks based on unauthorized JTAG manipulation. It also provides a JTAG port that conforms to IEEE 1149.1 and IEEE 1149.6 (AC) standards for BSR (boundary scan) testing.

SJC provides the following security levels:

- JTAG Disabled-JTAG use is permanently blocked.
- No-Debug-All security sensitive JTAG features are permanently blocked.
- Secure JTAG-JTAG use is restricted (as in the No-Debug level) unless a secret-key challenge/response protocol is successfully executed.
- JTAG Enabled-JTAG use is unrestricted.

Security levels are selected via e-fuse configuration.

Chapter 12
ARM Cortex A9 MPCore Platform (ARM)

12.1 Overview

The Cortex-A9 Core Platform consists of an ARM®Cortex®-A9 MPCore processor, which includes a Neon co-processor, a private timer and watch-dog, and 32 KB + 32 KB L1 data and instruction caches per core.

The Cortex-A9 Core Platform consists of a unified 1 MB L2 cache, SCU (Snoop Control Unit), and Generic Interrupt Controller (GIC). In addition, the Cortex-A9 Core Platform includes various components composing the ARM CoreSight debug/Trace system, including PTM and a 3 / 5[1] x CTI, 2xCTM, and 16KB ETB.

The Cortex-A9 processor utilizes two AXI-64 master ports connected from the SCU to the Level 2 Cache. The L2 cache also utilizes 2x AXI-64 to access the L3 memory or other SoC peripherals in a symmetric way.

The core supports debug through real-time trace via PTM, and static debug via JTAG.

The core platform supports static debug through the debug logic to SOC. This includes the capability of real time trace via ARM's CoreSight PTM, ETB and TPIU modules. The CTI and CTM modules allow cross-triggering of internal and external trigger sources.

12.2 External Signals

The following table describes the external signals of ARM:

Table 12-1. ARM External Signals

Signal	Description	Pad	Mode	Direction
ARM_EVENTI	Input event signal	GPIO_5	ALT7	I

Table continues on the next page...

1. Depending on specific part: i.MX 6Dual, i.MX 6Quad.

Table 12-1. ARM External Signals (continued)

Signal	Description	Pad	Mode	Direction
ARM_EVENTO	Output event signal	CSI0_PIXCLK	ALT7	O
ARM_TRACE00	Trace signal	CSI0_VSYNC	ALT7	O
ARM_TRACE01	Trace signal	CSI0_DAT4	ALT7	O
ARM_TRACE02	Trace signal	CSI0_DAT5	ALT7	O
ARM_TRACE03	Trace signal	CSI0_DAT6	ALT7	O
ARM_TRACE04	Trace signal	CSI0_DAT7	ALT7	O
ARM_TRACE05	Trace signal	CSI0_DAT8	ALT7	O
ARM_TRACE06	Trace signal	CSI0_DAT9	ALT7	O
ARM_TRACE07	Trace signal	CSI0_DAT10	ALT7	O
ARM_TRACE08	Trace signal	CSI0_DAT11	ALT7	O
ARM_TRACE09	Trace signal	CSI0_DAT12	ALT7	O
ARM_TRACE10	Trace signal	CSI0_DAT13	ALT7	O
ARM_TRACE11	Trace signal	CSI0_DAT14	ALT7	O
ARM_TRACE12	Trace signal	CSI0_DAT15	ALT7	O
ARM_TRACE13	Trace signal	CSI0_DAT16	ALT7	O
ARM_TRACE14	Trace signal	CSI0_DAT17	ALT7	O
ARM_TRACE15	Trace signal	CSI0_DAT18	ALT7	O
ARM_TRACE_CLK	Clock signal	CSI0_DATA_EN	ALT7	O
ARM_TRACE_CTL	Control signal	CSI0_MCLK	ALT7	O

12.3 Platform configuration

The Bus, Cortex A9 Core, and L2 Cache configuration options are contained in the following subsections.

Table 12-2. Cortex-A9 revision

Core	MP004-BU-50000-r2p10-0rel0
Neon	AT397-BU-50001- r2p0-00rel0
PL310	PL310-BU-00000-r3p1-50rel0

12.3.1 Platform and SCU configuration

Table 12-3. Cortex-A9 configuration

Option	Selected Value	Comments
MP_MODE	Yes	Multi-Processor mode
POWER_DOMAIN_WRAPPER	No	Wrappers to support power off of individual cores.
PTM_INTERFACE_PRESENT	Yes	Use PTM as part of Trace/Debug logic.
PARITY	Yes	Using RAM arrays which support parity.
CORE_NUM	2/4[1]	Number of cores
INT_NUM	128	Number of interrupts (SPIs) in GIC
ACP_PRESENT	No	Accelerator Coherency Port (ACP)
MASTER_NUM	2	Number of 64-bit AXI output master ports.

1. Depending on type of part - i.MX 6Dual, i.MX 6Quad.

12.3.2 Core configuration

Table 12-4. Cortex-A9 Core configuration

Option	Selected Value	Comments
DCACHESIZE	32	L1 Data cache size
ICACHESIZE	32	L1 Instruction cache size
TLBSIZE	128	
JAZELLE_PRESENT	Yes	Providing ARM's Jazelle technology hardware extensions.
FPU_PRESENT	No	The FPU functions are provided by NEON, thus additional FPU cannot be used.
NEON_PRESENT	Yes	Use MPE, NEON Co-Processor and FPU
PRELOAD_ENGINE_PRESENT	No	May only be beneficial in Video processing.

12.3.3 PL310 L2 Cache configuration

Table 12-5. PL310 L2 Cache configuration

Option	Selected Value	Comments
Cache way size	64 KB	(For total of 1 MB L2 size)
Number of cache ways	16	Performance enhancement versus 8 ways
RAM latencies	4[1]	
Data RAM banking	Yes	Significantly improves cache throughput

Table continues on the next page...

Table 12-5. PL310 L2 Cache configuration (continued)

Option	Selected Value	Comments
Slave port 1 present	Yes	
Master port 1 present	Yes	
Parity logic	Yes	For military / surveillance applications, and side ease the process of identify memory related issues.
Lockdown by master	Yes	Increase L2 optimization
Lockdown by line	Yes	Increase L2 optimization
AXI ID width	5	
Address filtering	No	Help in timing closure, not required for symmetric AXI bus connectivity scheme.
Speculative read	Yes	Performance boost, when used with CortexA9.
Size of L2 cache	1 MB	Size is implied by Cache-Size times cache-ways (i.e. 1 MB)

1. Preliminary estimate, final value TBD.

12.3.4 Endian Modes

The Cortex-A9 Core Platform supports little endian mode only. Big Endian is not supported even though both modes are supported by the Cortex-A9 processor.

12.3.5 Memory Parity error support

The i.MX 6Dual/6Quad ARM Cortex A9 MPCore ™ platform supports Parity Fail signals for several of the RAM arrays.

Parity fail indication is provided by "PARITYFAILn[7:0]" and "PARITYFAILSCU[N:0]" buses for system notification. On parity error event, these signals will be ORed together to provide a single event (interrupt) to the core, in case of any parity fail event.

12.4 Performance and Power

This section will discuss the operational conditions and performance goals for the Cortex-A9 Core Platform.

12.4.1 Low-Power design

The Cortex-A9 Core Platform low-power design is based on these characteristics:

- Symmetric processing and ARM design by using the same clock frequency on cores
- Low leakage of LP process
- C4 package

As a result, the proposed power modes and power management scheme is as follows:

- The same voltage level must be used for the logic part of the whole platform.
- Separate voltage for memories array is required to allow DVFS.
- On-chip power switches are not utilized by gating the power of the platform directly from the regulator.

12.4.1.1 SRPG (State Retention Power Gating)

ARM core SRPG is implemented by software save & restore of essential configuration registers prior to the complete power-down of the entire ARM platform.

Save & restore utilizes "Dormant mode" for the primary core (L1 cache flushed, L2 preserved), and power down mode of all other cores, as follows:

Power Down flow, [2]

- Power down request
- Save cores' essential registers and platform registers to L2/DDR memory by Dormant mode routines.
- Perform L1 cache clean operation on all cores
- Enter WFI state
- Power Gating of all cores' and platform logic

Power Up flow[2]

- Power up request to GPC external controller (interrupt)
- Supply power
- Reset
- Restore registers of platform and cores from memory by Dormant mode routines.

12.4.1.2 Dynamic Voltage and Frequency Scaling (DVFS)

The Cortex-A9 Core Platform has been designed, in conjunction with external control logic and software, to support dynamic scaling of voltage and frequency.

2. Dormant mode implementation information, provided by ARM, is preliminary.

12.4.2 Clocks, frequency goals

12.4.2.1 ARM Clock

For ARM clock please see the product data sheet.

12.4.2.2 Bus Clocks

The AXI master ports are designed to run at half the frequency of the Cortex A9 core clock.

The on-platform debug components clock is asynchronous to the ARM core clock.

12.4.2.3 Debug Clocks

The ARM platform contains several debug components.

Their clock frequencies are as follows:

- Trace buffer (ETB) clock - 133 MHz
- Trace port (TPIU) clock - 133 MHz
- AHB clock - 133 MHz

12.5 Core Platform Sub-Blocks details

12.5.1 ARM Cortex A9 MPCore ™ Processor

The information presented in this section focuses on design aspects of the ARM Cortex A9 MPCore ™ in the AP subsystem.

The ARM Cortex A9 is a high-performance, low-power, synthesisable processor with an L1 cache subsystem that provides full virtual memory capabilities. The Cortex A9 processor implements the ARMv7-A architecture and runs 32-bit ARM instructions, 16-bit and 32-bit Thumb2 instructions, and 8-bit Java™ byte-codes in Jazelle state.

The ARM Cortex A9 MPCore ™ processor in the chip consists of:

- Four (4) Cortex A9 processors in a cluster and a Snoop Control Unit (SCU) that can be used to ensure coherency within the cluster.

- A set of private memory-mapped peripherals, including a global timer and a watchdog and private timer for each Cortex-A9 processor present in the cluster.
- An integrated Interrupt Controller is an implementation of the Generic Interrupt Controller architecture. The integrated Interrupt Controller registers sit beside the timers and watchdog control registers in the private memory region of the Cortex-A9 MPCore.

Individual Cortex-A9 processors in the Cortex-A9 MPCore cluster are symmetrically implemented with hardware configurations as specified in Platform configuration.

12.5.2 Media Processing Engine (MPE - NEON)

The Media Processing Engine (MPE) implements ARM NEON technology, a media and signal processing architecture that adds instructions targeted at audio, video, 3-D graphics, image, and speech processing.

Advanced SIMD instructions are available in both ARM and Thumb states. The MPE also implements a VFPv3-D32 Floating-Point Unit.

The ARM Cortex A9 MPCore ™ Platform includes MPE per core.

12.5.3 Generic Interrupt Controller (GIC)

The Cortex-A9 MPCore contains an integrated interrupt controller that shares the same programmer's model as the PL390 (GIC), although there are implementation-specific differences.

12.5.3.1 Interrupt Controller Features

- 128 interrupt sources
- The Cortex-A9 multiprocessor contains the following types of interrupts:
  - Up to 16 Software Generated Interrupts (SGIs)
  - Private Peripheral Interrupt (PPI) - an interrupt generated by a peripheral that is specific to a single Cortex-A9 processor (there are 5 PPIs for each Cortex-A9 processor interface)

- Shared Peripheral Interrupt (SPI) - an interrupt generated by a peripheral which the Interrupt Controller can route to any or all Cortex-A9 processor interfaces. The Interrupt Controller supports a maximum of 224 SPIs (the i.MX 6Dual/ 6Quad controller is configured for 128 interrupts).
- Lockable Shared Peripheral Interrupts (LSPI) - there are 31 LSPIs, interrupts 32-62. The user can configure and then lock these interrupts against further change using CFGSDISABLE. The LSPIs are present only if the SPIs are present.

For more informtation, see ARM MPCORE Technical Reference Manual.

12.5.3.2 About the Interrupt Controller

The Interrupt Controller is a single functional unit that is located in a Cortex-A9 multiprocessor design.

The Interrupt Controller is memory-mapped. The Cortex-A9 processors access it by using a private interface through the SCU.

12.5.3.3 Interrupt Controller Clock frequency

The interrupt controller's clock period is 2x multiple of the main clock period.

The watchdogs and timers use the same clock as the interrupt controller.

12.5.3.4 TrustZone support

The Interrupt Controller permits all implemented interrupts to be individually defined as Secure or Non-secure.

The user can program Secure interrupts to use either the IRQ or FIQ interrupt mechanism of a Cortex-A9 processor through the FIQen bit in the ICPICR Register.

Non-secure interrupts are always signalled using the IRQ mechanism of a Cortex-A9 processor.

12.5.4 Instruction and data caches (L1)

The Cortex-A9 processor is configured with a 32 Kbyte Instruction Cache and a 32 Kbyte Data Cache.

12.5.4.1 L1 features

- Four-way set associative cache
- Virtually indexed and physically addressed
- Capable of providing two words per cycle for all requesting sources
- Eight 32-bit words per cache line
- 128 indexes per tag RAM

12.5.5 L2 Cache and controller (PL310)

The ARM Cortex A9 MPCore ™ platform includes 1MB unified (data / instruction) L2 cache unit, based on the PL310 cache controller IP by ARM.

The Cortex-A9 processor utilizes 2x AXI-64 master ports connected from the SCU to the Level 2 Cache. The L2 cache also utilizes 2x AXI-64 to access the L3 memory or other SoC peripherals in a symmetric way.

See Table 12-5 for more information.

12.6 Debug and Trace Sub-blocks (CoreSight components)

This section gives a brief overview of the modules that are implemented within the Cortex-A9 Core Platform.

The Cortex-A9 Core Platform debug blocks are part of the overall CoreSight debug system which include the 16KB ETB, 2 x CTM's, 3 / 5[3] x CTI's, ATB replicator, DAP, TPIU and APB address decode.

The CoreSight™ compatible Program Flow Trace Macrocell (PTM) provides control for ARM software tracing and debug. The Cross Trigger Interface (CTI) is included in the Cortex-A9 platform to provide a common programming model for use by the debug tools, control the trigger sources, and interface to the Cross Trigger Matrix (CTM). The debug is controlled via an ARM Debug Access Port (DAP).

For details of the full CoreSight debug subsystem, see the System Debug chapter.

3. Depending on specific part: i.MX 6Dual, i.MX 6Quad.

12.6.1 Debug Access Port (DAP)

The Debug Access Port (DAP) is an implementation of an ARM Debug Interface version 5.1 (ADIv5.1) comprised of a number of components supplied in a single configuration.

All the supplied components fit into the various architectural components for Debug Ports (DPs), which are used to access the DAP from an external debugger and Access Ports (APs), to access on-chip system resources.

The debug port and access ports together are referred to as the DAP. The DAP provides real-time access for the debugger without halting the processor to:

- AMBA system memory and peripheral registers
- All debug configuration registers

The DAP also provides debugger access to JTAG scan chains of system components, (to non-CoreSight compliant processors, for example).

12.6.2 Program Trace Macrocell (PTM)

The PTM unit is a nonintrusive trace macrocell that filters and compresses instruction trace for use in system debugging and system profiling.

The PTM unit has an external interface outside of the processor called the *Advanced Trace Bus* (ATB) interface. The PTM is an evolvement of the ETM, designed for the Cortex A9 cores, handling program trace only.

The Cortex A9 PTM provides real time instruction trace for the Cortex A9. It's designed to be used with the CoreSight Design Kit.

Real time tracing is controlled by specifying a set of filtering and triggering resources which include address and data comparators, counters and sequencers.

Two main schemes can be used for connecting PTMs:

1. Single PTM - shared by all cores and resources.
2. Nx PTMs, where N is numbers of cores in the system.

12.6.2.1 Program Flow Trace (PFT)

The CoreSight Program Flow Trace Macrocell (PTM) is based on the Program Flow Trace (PFT) architecture. The PTM generates information that trace tools use to reconstruct the execution of all or part of a program.

The PFT architecture assumes the trace tools can access a copy of the code being traced. For this reason, the PTM generates trace only at certain points in program execution, called waypoints. This reduces the amount of trace data generated by the PTM compared to the ETM protocol. Waypoints are changes in the program flow or events, such as an exception. The trace tools use waypoints to follow the flow of program execution.

For full reconstruction of the program flow, the PTM traces:

- Indirect branches, with target address and condition code
- Direct branches with only the condition code
- Instruction barrier instructions
- Exceptions, with indication of where the exception occurred
- Changes in processor instruction set state
- Changes in processor security state
- Context-ID changes
- Entry to and return from Debug state when Halting Debug-mode is enabled

You can also configure the PTM to trace:

- Cycle count between traced waypoints
- Global system timestamps
- Target addresses for taken direct branches

PTM components include the following main components:

- Processor Interface to monitor the behavior of the processor.
- Trace Generation to create a real-time trace stream.
- Filtering and Triggering Resources used to affect when trace is generated and to control the capturing of trace by the trace tools.
- Main FIFO (72 bytes) flattens out any bursts in the trace stream, and signals an overflow in the trace when it becomes full, halting trace generation until the FIFO empties.
- ATB interface for PTM output.
- APB interface to access the PTM registers.

12.6.3 Cross Trigger Interface (CTI)

This block controls the Trigger Interface (TI). The CTI combines and maps the trigger requests, and broadcasts them to all other interfaces on the ECT as channel events.

When the CTI receives a channel event, it maps this onto a trigger output. This enables subsystems to cross trigger with each other. The receiving and transmitting of triggers is performed through the TI. Each CTI has 3/5 trigger inputs.

See System Debug for more information on the CTI.

12.6.4 Embedded Trace Buffer (ETB)

The ETB provides on-chip storage of trace data using 32-bit, 16KB RAM.

The ETB accepts trace data from the Cortex-A9 PTM via an ATB port (passing through a replicator in between). Providing an on-chip buffer alleviates the pin count, bandwidth, and pad design requirements associated with sending trace data to a debugger directly through package pins in near real-time.

Features:

- 16 KB compiled memory for the trace buffer and can be used as general purpose memory
- AMBA Peripheral Bus programming interface for configuration and memory access

12.6.4.1 AMBA Trace Bus (ATB) Replicator

The ATB Replicator enables two trace sinks (ETB and an off platform port generally connected to a Trace Port Interface Unit-TPIU) to be wired together and receive ATB trace data from the same trace source (PTM).

There are no programmable registers. It takes incoming trace data from a single source (PTM) and replicates it as multiple masters.

Chapter 13
AHB to IP Bridge (AIPSTZ)

13.1 Overview

This section provides an overview of the AHB to IP Bridge (AIPSTZ). The peripheral bridge acts as an interface between the system bus and lower bandwidth IP Slave (IPS) bus peripherals.

13.1.1 Features

The following list summarizes the key features of the bridge:

- The bridge supports the IPS slave bus signals. This interface is only meant for slave peripherals.
- The bridge supports 8-, 16-, and 32-bit IPS peripherals. (Accesses larger than the size of a peripheral are not supported, except to 32-bit memory.)
- The bridge supports a pair of IPS accesses for 64-bit and certain misaligned AHB transfers to 32-bit memory in 64-bit platforms.
- The bridge directly supports up to 32 16-Kbyte external IPS peripherals, and 2 global external IPS peripheral spaces. The bridge occupies 1 MBytes of total address space.
- The bridge provides configurable per-block and per-master access protections. More details on the protection features and configuration can be found in the Security Reference Manual
- Peripheral read transactions require a minimum of 2 hclk clocks, and unbuffered write transactions require a minimum of 3 hclk clocks.
- The bridge uses one single asynchronous reset and one global clock.

13.2 Clocks

The following table describes the clock sources for AIPSTZ. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 13-1. AIPSTZ Clocks

Clock name	Clock Root	Description
hclk	ahb_clk_root	Module clock

13.3 General Operation

The AHB to IP bridge is the interface between the AHB and on-chip IPS peripherals, which are sub-blocks containing readable/writable control and status registers.

The AHB master reads and writes these registers through the AIPSTZ. The bridge generates block enables, the block address, transfer attributes, byte enables and write data as inputs to the IPS peripherals. The bridge captures read data from the IPS interface and drives it on the AHB.

It occupies a 1-Mbyte portion of the address space. The register maps of the IPS peripherals are located on 16-Kbyte boundaries. Each IPS peripheral is allocated one 16K-byte block of the memory map, and is activated by one of the block enables from the bridge. Up to thirty-two 16-Kbyte external IPS peripherals may be implemented, occupying contiguous blocks of 16 Kbytes. Two global external IPS block enables are available for the remaining address space to allow for customization and expansion of addressed peripheral devices. In addition, a single "non-global" block enable is also asserted whenever any of the thirty-two non-global block enables is asserted.

The bridge is responsible for indicating to IPS peripherals if an access is in supervisor or user mode. It may block user mode accesses to certain IPS peripherals or it may allow the individual IPS peripherals to determine if user mode accesses are allowed. In addition, peripherals may be designated as write-protected.

The bridge supports the notion of "trusted" masters for security purposes. Masters may be individually designated as trusted for reads, trusted for writes, or trusted for both reads and writes, as well as being forced to look as though all accesses from a master are in user-mode privilege level. Refer to AIPSTZ Memory Map/Register Definition for more information.

All peripheral devices are expected to only require aligned accesses equal to or smaller in size than the peripheral size. An exception to this rule is supported for 32-bit peripherals to allow memory to be placed on the IPS.

13.4 Functional Description

The AIPS bridge serves as a protocol translator between the AHB system bus and the IP bus.

Support is provided for generating a pair of 32-bit IP bus accesses when targeted by a 64-bit system bus access, or a misaligned access which crosses a 32-bit boundary. No other bus-sizing access support is provided.

13.5 Access Protections

The AIPSTZ bridge provides programmable access protections for both masters and peripherals. It allows the privilege level of a master to be overridden, forcing it to user-mode privilege, and allows masters to be designated as trusted or untrusted.

Peripherals may require supervisor privilege level for access, may restrict access to a trusted master only, and may be write-protected. IP bus peripherals are subject to access control policies set in both CSU registers and AIPSTZ registers. An access is blocked if it is denied by either policy.

13.6 Access Support

Aligned 64-bit accesses, aligned and misaligned word and half word accesses, as well as byte accesses are supported for 32-bit peripherals. Misaligned accesses are supported to allow memory to be placed on the IPS.

Peripheral registers must not be misaligned, although no explicit checking is performed by the AIPS bridge. The bridge will perform two IPS transfers for 64-bit accesses, word accesses with byte offsets of 1, 2, or 3, and for half word accesses with a byte offset of 3. All other accesses will be performed with a single IPS transfer.

Only aligned half word and byte accesses are supported for 16-bit peripherals. All other accesses types are unsupported, and results of such accesses are undefined. They are not terminated with an error response.

Only byte accesses are supported for 8-bit peripherals. All other accesses types are unsupported, and results of such accesses are undefined. They are not terminated with an error response.

13.7 Initialization Information

The AIPS bridge should be programmed before use.

The following registers should be initialized: The Master Privilege Registers (AIPSTZ_MPRs), the Peripheral Access Control registers (AIPSTZ_PACRs), and the Off-platform Peripheral Access Control registers (AIPSTZ_OPACRs) described in AIPSTZ Memory Map/Register Definition.

13.7.1 Security Block

The AIPSTZ contains a security block that is connected to each off-platform peripheral. This block filters accesses based on write/read, non-secure, and supervisor signals.

Each peripheral can be individually configured to allow or deny each of the following transactions as described in the table below:

Table 13-2. Peripheral Access Configuration options

Config Bit	Write	Non-Secure	Supervisor	Meaning
0	0	0	0	Secure User Read
1	0	0	1	Secure Supervisor Read
2	0	1	0	Non-Secure User Read
3	0	1	1	Non-Secure Supervisor Read
4	1	0	0	Secure User Write
5	1	0	1	Secure Supervisor Write
6	1	1	0	Non-Secure User Write
7	1	1	1	Non-Secure Supervisor Write

Each peripheral has a security configuration (sec_config_X) input for determining whether to allow or deny a given access type. These are 8-bit vectors, with each bit corresponding to one of the transactions above as listed in the Config Bit column of Table 13-2. If the bit is asserted (1'b1), the transaction is allowed. If the bit is negated (1'b0), the transaction is not allowed.

For example, if peripheral 0 is configured as follows:

sec_config_0 [7:0] = 8'b0011_0011

This peripheral can only be accessed by secure transactions. Bits 0, 1, 4, and 5 are asserted and these bits refer to the four types of secure transactions. If an insecure transaction is attempted to this peripheral, it will result in an error.

Eight bits per peripheral across an entire system can result in a large number of configuration bits that must be assigned and controlled, most likely in a series of registers in another block. To reduce the number of register bits required predefined sets of security profiles can be defined and encapsulated in an external security translation block. The table below describes one set of security profiles that has been proposed for use with the AIPSTZ.

Table 13-3. Security Levels

CSU_SEC_LEVEL	Non-Secure User	Non-Secure Supervisor	Secure User	Secure Supervisor
0	RD+WR	RD+WR	RD+WR	RD+WR
1	NOT ALLOWED	RD+WR	RD+WR	RD+WR
2	Read Only	Read Only	RD+WR	RD+WR
3	NOT ALLOWED	Read Only	RD+WR	RD+WR
4	NOT ALLOWED	NOT ALLOWED	RD+WR	RD+WR
5	NOT ALLOWED	NOT ALLOWED	NOT ALLOWED	RD+WR
6	NOT ALLOWED	NOT ALLOWED	Read Only	Read Only
7	NOT ALLOWED	NOT ALLOWED	NOT ALLOWED	NOT ALLOWED

Information regarding CSU is provided in the Security Reference Manual. Contact your Freescale representative for information about obtaining this document.

A 3-bit input, 8-bit output translation block can be used such that only three register bits are required to set the security profile and the translation block will drive the correct 8-bit configuration vector. Each peripheral connected to the AIPSTZ would require this translation block. The top level AIPSTZ has this three bit input line `csu_sec_level[2:0]' corresponding to each peripheral X.

13.8 AIPSTZ Memory Map/Register Definition

The memory map for the AIPS SW-visible registers is shown in the table below.

The MPROT and OPACR fields are 4 bits in width. Some bits may be reserved depending on device.

AIPSTZ memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
207_C000	Master Priviledge Registers (AIPSTZ1_MPR)	32	R/W	7700_0000h	13.8.1/572
207_C040	Off-Platform Peripheral Access Control Registers (AIPSTZ1_OPACR)	32	R/W	4444_4444h	13.8.2/574
207_C044	Off-Platform Peripheral Access Control Registers (AIPSTZ1_OPACR1)	32	R/W	4444_4444h	13.8.3/578
207_C048	Off-Platform Peripheral Access Control Registers (AIPSTZ1_OPACR2)	32	R/W	4444_4444h	13.8.4/581
207_C04C	Off-Platform Peripheral Access Control Registers (AIPSTZ1_OPACR3)	32	R/W	4444_4444h	13.8.5/584
207_C050	Off-Platform Peripheral Access Control Registers (AIPSTZ1_OPACR4)	32	R/W	4444_4444h	13.8.6/587
217_C000	Master Priviledge Registers (AIPSTZ2_MPR)	32	R/W	7700_0000h	13.8.1/572
217_C040	Off-Platform Peripheral Access Control Registers (AIPSTZ2_OPACR)	32	R/W	4444_4444h	13.8.2/574
217_C044	Off-Platform Peripheral Access Control Registers (AIPSTZ2_OPACR1)	32	R/W	4444_4444h	13.8.3/578
217_C048	Off-Platform Peripheral Access Control Registers (AIPSTZ2_OPACR2)	32	R/W	4444_4444h	13.8.4/581
217_C04C	Off-Platform Peripheral Access Control Registers (AIPSTZ2_OPACR3)	32	R/W	4444_4444h	13.8.5/584
217_C050	Off-Platform Peripheral Access Control Registers (AIPSTZ2_OPACR4)	32	R/W	4444_4444h	13.8.6/587

13.8.1 Master Priviledge Registers (AIPSTZx_MPR)

Each AIPSTZ_MPR specifies 16 4-bit fields defining the access privilege level associated with a bus master in the platform, as well as specifying whether write accesses from this master are bufferable shown in Table 13-6

The registers provide one field per bus master, where field 15 corresponds to master 15, field 14 to master 14,... field 0 to master 0 (typically the processor core). The master index allocation is shown in Table 13-7.

Table 13-16. MPROT Field

Bit	Field	Description
3	MBW	**Master Buffer Writes** - This bit determines whether the AIPSTZ is enabled to buffer writes from this master.
2	MTR	**Master Trusted for Reads** - This bit determines whether the master is trusted for read accesses.
1	MTW	**Master Trusted for Writes** - This bit determines whether the master is trusted for write accesses.

Table continues on the next page...

Table 13-16. MPROT Field (continued)

Bit	Field	Description
0	MPL	**Master Privilege Level** - This bit determines how the privilege level of the master is determined.

NOTE

The reset value is set to 0000_0000_7700_0000, which makes master 0 and master 1 (ARM CORE) the trusted masters. Trusted software can change the settings after reset.

Table 13-17. Master Index Allocation

Master Index	Master Name	Comments
Master 0	All masters excluding ARM core, SDMA and CAAM	Share the same number allocation.
Master 1	ARM CORE	
Master 2	CAAM	
Master 3	SDMA	
Master 4-15	Reserved	

Address: Base address + 0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MPROT0				MPROT1				MPROT2				MPROT3				Reserved															
Reset	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AIPSTZ*x*_MPR field descriptions

Field	Description
31–28 MPROT0	Master 0 Priviledge, Buffer, Read, Write Control xxx0 **MPL** — Accesses from this master are forced to user-mode (ips_supervisor_access is forced to zero) regardless of the hprot[1] access attribute. xxx1 **MPL** — Accesses from this master are not forced to user-mode. The hprot[1] access attribute is used directly to determine ips_supervisor_access. xx0x **MTW** — This master is not trusted for write accesses. xx1x **MTW** — This master is trusted for write accesses. x0xx **MTR** — This master is not trusted for read accesses. x1xx **MTR** — This master is trusted for read accesses. 0xxx **MBW** — Write accesses from this master are not bufferable 1xxx **MBW** — Write accesses from this master are allowed to be buffered
27–24 MPROT1	Master 1 Priviledge, Buffer, Read, Write Control xxx0 **MPL** — Accesses from this master are forced to user-mode (ips_supervisor_access is forced to zero) regardless of the hprot[1] access attribute. xxx1 **MPL** — Accesses from this master are not forced to user-mode. The hprot[1] access attribute is used directly to determine ips_supervisor_access. xx0x **MTW** — This master is not trusted for write accesses.

Table continues on the next page...

AIPSTZx_MPR field descriptions (continued)

Field	Description
	xx1x **MTW** — This master is trusted for write accesses. x0xx **MTR** — This master is not trusted for read accesses. x1xx **MTR** — This master is trusted for read accesses. 0xxx **MBW** — Write accesses from this master are not bufferable 1xxx **MBW** — Write accesses from this master are allowed to be buffered
23–20 MPROT2	Master 2 Priviledge, Buffer, Read, Write Control xxx0 **MPL** — Accesses from this master are forced to user-mode (ips_supervisor_access is forced to zero) regardless of the hprot[1] access attribute. xxx1 **MPL** — Accesses from this master are not forced to user-mode. The hprot[1] access attribute is used directly to determine ips_supervisor_access. xx0x **MTW** — This master is not trusted for write accesses. xx1x **MTW** — This master is trusted for write accesses. x0xx **MTR** — This master is not trusted for read accesses. x1xx **MTR** — This master is trusted for read accesses. 0xxx **MBW** — Write accesses from this master are not bufferable 1xxx **MBW** — Write accesses from this master are allowed to be buffered
19–16 MPROT3	Master 3 Priviledge, Buffer, Read, Write Control. xxx0 **MPL** — Accesses from this master are forced to user-mode (ips_supervisor_access is forced to zero) regardless of the hprot[1] access attribute. xxx1 **MPL** — Accesses from this master are not forced to user-mode. The hprot[1] access attribute is used directly to determine ips_supervisor_access. xx0x **MTW** — This master is not trusted for write accesses. xx1x **MTW** — This master is trusted for write accesses. x0xx **MTR** — This master is not trusted for read accesses. x1xx **MTR** — This master is trusted for read accesses. 0xxx **MBW** — Write accesses from this master are not bufferable 1xxx **MBW** — Write accesses from this master are allowed to be buffered
-	This field is reserved. Reserved

13.8.2 Off-Platform Peripheral Access Control Registers (AIPSTZx_OPACR)

Each of the off-platform peripherals have an Off-platform Peripheral Access Control Register (AIPSTZ_OPACR) which defines the access levels supported by the given block.

Each AIPSTZ_OPACR has the following format shown in Table 13-9

Table 13-19. OPAC Field

Bit	Field	Description
3	BW	**Buffer Writes** - This bit determines whether write accesses to this peripheral are allowed to be buffered.[1]
2	SP	**Supervisor Protect** - This bit determines whether the peripheral requires supervisor privilege level for access.
1	WP	**Write Protect** - This bit determines whether the peripheral allows write accesses.
0	TP	**Trusted Protect** - This bit determines whether the peripheral allows accesses from an untrusted master.

1. Buffered writes are not available for AIPSTZ. This bit should be set to '0'.

Address: Base address + 40h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OPAC0				OPAC1				OPAC2				OPAC3				OPAC4				OPAC5				OPAC6				OPAC7			
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

AIPSTZx_OPACR field descriptions

Field	Description
31–28 OPAC0	Off-platform Peripheral Access Control 0 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
27–24 OPAC1	Off-platform Peripheral Access Control 1 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus.

Table continues on the next page...

AIPSTZx_OPACR field descriptions (continued)

Field	Description
	0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
23–20 OPAC2	Off-platform Peripheral Access Control 2 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
19–16 OPAC3	Off-platform Peripheral Access Control 3 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
15–12 OPAC4	Off-platform Peripheral Access Control 4 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

Table continues on the next page...

AIPSTZx_OPACR field descriptions (continued)

Field	Description
11–8 OPAC5	Off-platform Peripheral Access Control 5 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
7–4 OPAC6	Off-platform Peripheral Access Control 6 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
OPAC7	Off-platform Peripheral Access Control 7 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

13.8.3 Off-Platform Peripheral Access Control Registers (AIPSTZx_OPACR1)

Each of the off-platform peripherals have an Off-platform Peripheral Access Control Register (AIPSTZ_OPACR) which defines the access levels supported by the given block.

Each AIPSTZ_OPACR has the following format shown in Table 13-9

Address: Base address + 44h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OPAC8				OPAC9				OPAC10				OPAC11				OPAC12				OPAC13				OPAC14				OPAC15			
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

AIPSTZx_OPACR1 field descriptions

Field	Description
31–28 OPAC8	Off-platform Peripheral Access Control 8 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
27–24 OPAC9	Off-platform Peripheral Access Control 9 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

Table continues on the next page...

AIPSTZx_OPACR1 field descriptions (continued)

Field	Description
23–20 OPAC10	Off-platform Peripheral Access Control 10 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
19–16 OPAC11	Off-platform Peripheral Access Control 11 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
15–12 OPAC12	Off-platform Peripheral Access Control 12 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
11–8 OPAC13	Off-platform Peripheral Access Control 13

Table continues on the next page...

AIPSTZx_OPACR1 field descriptions (continued)

Field	Description
	xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
7–4 OPAC14	Off-platform Peripheral Access Control 14 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
OPAC15	Off-platform Peripheral Access Control 15 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

13.8.4 Off-Platform Peripheral Access Control Registers (AIPSTZx_OPACR2)

Each of the off-platform peripherals have an Off-platform Peripheral Access Control Register (AIPSTZ_OPACR) which defines the access levels supported by the given block.

Each AIPSTZ_OPACR has the following format shown in Table 13-9

Address: Base address + 48h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OPAC16				OPAC17				OPAC18				OPAC19				OPAC20				OPAC21				OPAC22				OPAC23			
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

AIPSTZx_OPACR2 field descriptions

Field	Description
31–28 OPAC16	Off-platform Peripheral Access Control 16 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
27–24 OPAC17	Off-platform Peripheral Access Control 17 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

Table continues on the next page...

AIPSTZx_OPACR2 field descriptions (continued)

Field	Description
23–20 OPAC18	Off-platform Peripheral Access Control 18 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
19–16 OPAC19	Off-platform Peripheral Access Control 19 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
15–12 OPAC20	Off-platform Peripheral Access Control 20 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
11–8 OPAC21	Off-platform Peripheral Access Control 21

Table continues on the next page...

AIPSTZx_OPACR2 field descriptions (continued)

Field	Description
	xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
7–4 OPAC22	Off-platform Peripheral Access Control 22 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
OPAC23	Off-platform Peripheral Access Control 23 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

13.8.5 Off-Platform Peripheral Access Control Registers (AIPSTZx_OPACR3)

Each of the off-platform peripherals have an Off-platform Peripheral Access Control Register (AIPSTZ_OPACR) which defines the access levels supported by the given block.

Each AIPSTZ_OPACR has the following format shown in Table 13-9

Address: Base address + 4Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OPAC24				OPAC25				OPAC26				OPAC27				OPAC28				OPAC29				OPAC30				OPAC31			
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

AIPSTZx_OPACR3 field descriptions

Field	Description
31–28 OPAC24	Off-platform Peripheral Access Control 24 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
27–24 OPAC25	Off-platform Peripheral Access Control 25 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

Table continues on the next page...

AIPSTZx_OPACR3 field descriptions (continued)

Field	Description
23–20 OPAC26	Off-platform Peripheral Access Control 26 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
19–16 OPAC27	Off-platform Peripheral Access Control 27 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
15–12 OPAC28	Off-platform Peripheral Access Control 28 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
11–8 OPAC29	Off-platform Peripheral Access Control 29

Table continues on the next page...

AIPSTZx_OPACR3 field descriptions (continued)

Field	Description
	xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
7–4 OPAC30	Off-platform Peripheral Access Control 30 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
OPAC31	Off-platform Peripheral Access Control 31 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

13.8.6 Off-Platform Peripheral Access Control Registers (AIPSTZx_OPACR4)

Each of the off-platform peripherals have an Off-platform Peripheral Access Control Register (AIPSTZ_OPACR) which defines the access levels supported by the given block.

Each AIPSTZ_OPACR has the following format shown in Table 13-9

Address: Base address + 50h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OPAC32				OPAC33				Reserved																							
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

AIPSTZx_OPACR4 field descriptions

Field	Description
31–28 OPAC32	Off-platform Peripheral Access Control 32 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.
27–24 OPAC33	Off-platform Peripheral Access Control 33 xxx0 **TP** — Accesses from an untrusted master are allowed. xxx1 **TP** — Accesses from an untrusted master are not allowed. If an access is attempted by an untrusted master, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. xx0x **WP** — This peripheral allows write accesses. xx1x **WP** — This peripheral is write protected. If a write access is attempted, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. x0xx **SP** — This peripheral does not require supervisor privilege level for accesses. x1xx **SP** — This peripheral requires supervisor privilege level for accesses. The master privilege level must indicate supervisor via the hprot[1] access attribute, and the MPROTx[MPL] control bit for the master must be set. If not, the access is terminated with an error response and no peripheral access is initiated on the IPS bus. 0xxx **BW** — Write accesses to this peripheral are not bufferable by the AIPSTZ. 1xxx **BW** — Write accesses to this peripheral are allowed to be buffered by the AIPSTZ.

Table continues on the next page...

AIPSTZx_OPACR4 field descriptions (continued)

Field	Description
-	This field is reserved. Reserved

Chapter 14
AHB-to-APBH Bridge with DMA (APBH-Bridge-DMA)

14.1 Overview

The AHB-to-APBH bridge provides the i.MX 6Dual/6Quad with an inexpensive peripheral attachment bus running on the AHB's HCLK.

(The H in APBH denotes that the APB*H* is synchronous to HCLK.)

As shown in the figure below, the AHB-to-APBH bridge includes the AHB-to-APB PIO bridge for a memory-mapped I/O to the APB devices, as well as a central DMA facility for devices on this bus and a vectored interrupt controller for the ARM core. Each one of the APB peripherals, including the vectored interrupt controller, is documented in their respective chapters.

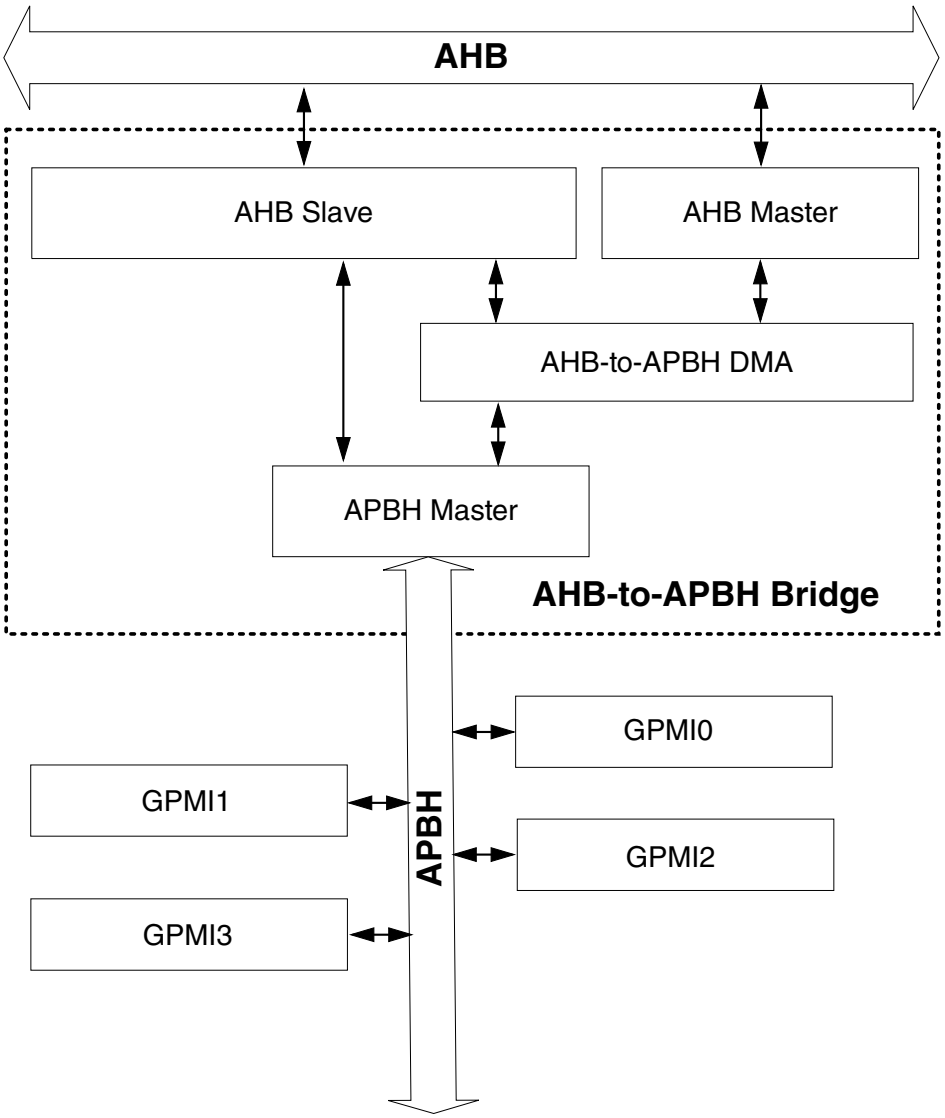


Figure 14-1. AHB-to-APBH Bridge DMA Block Diagram

There is no separate DMA bus for these devices. Contention between the DMA's use of the APBH bus and the AHB-to-APB bridge functions' use of the APBH is mediated by an internal arbitration logic. For contention between these two units, the DMA is favored and the AHB slave will report "not ready" through its HREADY output until the bridge transfer can complete. The arbiter tracks repeated lockouts and inverts the priority, guaranteeing the ARM platform every fourth transfer on the APB.

14.2 Clocks

The table found here describes the clock sources for APBH.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 14-1. APBH Clocks

Clock name	Clock Root	Description
hclk	usdhc3_clk_root	Module clock

14.3 APBH DMA

The DMA supports four channels of DMA services, as shown in the following table. The shared DMA resource allows each independent channel to follow a simple chained command list.

Command chains are built up using the general structure, as shown in Figure 14-2.

Table 14-2. APBH DMA channel assignments

APBH DMA Channel #	Usage
0	GPMI0
1	GPMI1
2	GPMI2
3	GPMI3

A single command structure or channel command word specifies a number of operations to be performed by the DMA in support of a given device. Thus, the ARM platform can set up large units of work, chaining together many DMA channel command words, pass them off to the DMA, and have no further concern for the device until the DMA completion interrupt occurs. The goal is to have enough intelligence in the DMA and the devices to keep the interrupt frequency from any device below 1 KHz (arrival intervals longer than 1 ms).

A single command structure can issue 32-bit PIO write operations to key registers in the associated device using the same APB bus and controls that it uses to write DMA data bytes to the device. For example, this allows a chain of operations to be issued to the GPMI controller to send NAND command bytes, address bytes, and data transfers where the command and the address structure is completely under software control, but the administration of that transfer is handled autonomously by the DMA. Each DMA structure can have 0–15 PIO words appended to it. The CMDPIOWORDS field, if non-zero, instructs the DMA engine to copy these words to the APB, beginning at the first register address offset for the peripheral and and incrementing the register offset each cycle.

The DMA master generates only normal read/write transfers to the APBH. It does *not* generate set, clear, or toggle (SCT) transfers.

After any requested PIO words have been transferred to the peripheral, the DMA examines the two-bit command field in the channel command structure. Table 14-3 shows the four commands implemented by the DMA.

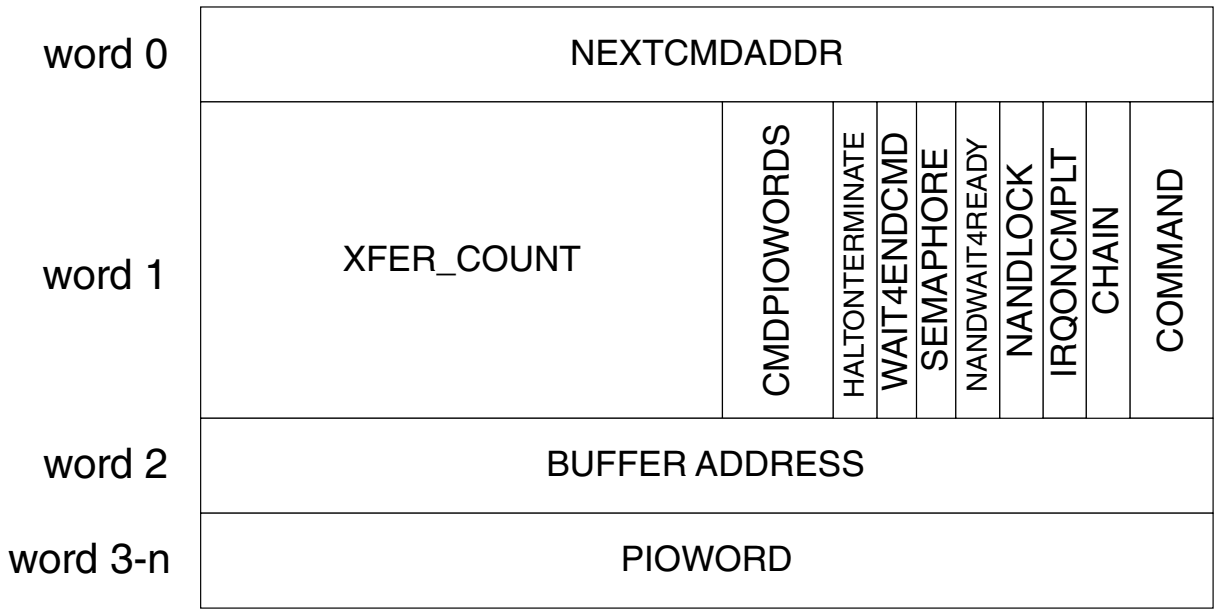


Figure 14-2. AHB-to-APBH Bridge DMA channel command structure

Table 14-3. APBH DMA commands

DMA Command	Usage
00	NO_DMA_XFER. Perform any requested PIO word transfers, but terminate the command before any DMA transfer.
01	DMA_WRITE. Perform any requested PIO word transfers, then perform a DMA transfer from the peripheral for the specified number of bytes.
10	DMA_READ. Perform any requested PIO word transfers and then perform a DMA transfer to the peripheral for the specified number of bytes.
11	DMA_SENSE. Perform any requested PIO word transfers, then perform a conditional branch to the next chained device. Follow the NEXTCMD_ADDR pointer if the peripheral sense is false. Follow the BUFFER_ADDRESS as a chain pointer if the peripheral sense line is true. This command becomes a no-operation for any channel other than a GPMI channel.

DMA_WRITE operations copy data bytes to the system memory (on-chip RAM or SDRAM) from the associated peripheral.

DMA_READ operations copy data bytes to the APB peripheral from the system memory. The DMA engine contains a shared byte aligner that aligns bytes from system memory to or from the peripherals. Peripherals always assume little-endian-aligned data arrives or

departs on their 32-bit APB. The DMA_READ transfer uses the BUFFER_ADDRESS word in the command structure to point to the DMA data buffer to be read by the DMA_READ command.

The NO_DMA_XFER command is used to write PIO words to a device without performing any DMA data byte transfers. This command is useful in such applications as activating the NAND devices CHECKSTATUS operation. The check status command reads a status byte from the NAND device, performs an XOR and MASK against an expected value supplied as part of the PIO transfer. Once the read check completes (see NAND Read Status Polling Example), the NO_DMA_XFER command completes. The result in the peripheral is that its sense line is driven by the results of the comparison. The sense flip-flop is only updated by CHECKSTATUS for the device that is executed. At some future point, the chain contains a DMA command structure with the fourth and final command value, that is, the DMA_SENSE command.

As each DMA command completes, it triggers the DMA to load the next DMA command structure in the chain. The normal flow list of DMA commands is found by following the NEXTCMD_ADDR pointer in the DMA command structure. The DMA_SENSE command uses the DMA buffer pointer word of the command structure to point to an alternate DMA command structure chain or list. The DMA_SENSE command examines the sense line of the associated peripheral. If the sense line is false, then the DMA follows the standard list found whose next command is found from the pointer in the NEXTCMD_ADDR word of the command structure. If the sense line is true, then the DMA follows the alternate list whose next command is found from the pointer in the DMA Buffer Pointer word of the DMA_SENSE command structure (see Figure 14-2). The sense command ignores the CHAIN bit, so that both pointers must be valid when the DMA comes to a sense command.

If the wait-for-end-command bit (WAIT4ENDCMD) is set in a command structure, the DMA channel waits for the device to signal completion of a command by toggling the endcmd signal before proceeding to load and execute the next command structure. Then, if DECREMENT_SEMAPHORE is set, the semaphore is decremented after the end command is seen.

A detailed bit-field view of the DMA command structure is shown in the following table, which shows a field that specifies the number of bytes to be transferred by this DMA command. The transfer-count mechanism is duplicated in the associated peripheral, either as an implied or as a specified count in the peripheral.

Table 14-4. DMA channel command word in system memory

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
NEXT_COMMAND_ADDRESS																															

Table continues on the next page...

Table 14-4. DMA channel command word in system memory (continued)

Number DMA Bytes to Transfer	Number PIO Words to Write				HALTONTERMINATE	WAIT4ENDCMD	DECREMENT SEMAPHORE	NANDwAIT4READY	NANDLOCK	IRQ_COMPLETE	CHAIN	COMMAND
DMA Buffer or Alternate CCW												
Zero or More PIO Words to Write to the Associated Peripheral Starting at its Base Address on the APBH Bus												

Figure 14-2 also shows the CHAIN bit in bit 2 of the second word of the command structure. This bit is set to 1, if the NEXT_COMMAND_ADDRESS contains a pointer to another DMA command structure. If a null pointer (0) is loaded into the NEXT_COMMAND_ADDRESS, it is not detected by the DMA hardware. Only the CHAIN bit indicates whether a valid list exists beyond the current structure.

If the IRQ_COMPLETE bit is set in the command structure, then the last act of the DMA before loading the next command is to set the interrupt-status bit corresponding to the current channel. The sticky interrupt request bit in the DMA CSR remains set until cleared by the software. It can be used to interrupt the ARM platform.

The NAND_LOCK bit is monitored by the DMA channel arbiter. Once a NAND channel (from channel 0 to channel 3) succeeds in the arbiter with its NAND_LOCK bit set, then the arbiter ignores the other NAND channels until a command is completed in which the NAND_LOCK is not set. Notice that the semantic here is that the NAND_LOCK state is to limit scheduling of a non-locked DMA. A DMA channel can go from unlocked to locked in the arbiter at the beginning of a command when the NAND_LOCK bit is set. When the last DMA command of an atomic sequence is completed, the lock should be removed. To accomplish this, the last command does not have the NAND_LOCK bit. It is still locked in the atomic state within the arbiter when the command starts, so that it is the only NAND command that can be executed. At the end, it drops from the atomic state within the arbiter.

The NAND_WAIT4READY bit also has a special use for GPMI channels (from channel 0 to channel 3), i.e., the NAND device channels. The GPMI peripheral supplies a sample of the ready line from the NAND device. This ready value is used to hold off of a command with this bit set until the ready line is asserted to 1. Once the arbiter sees a command with a wait-for-ready set, it holds off that channel until ready is asserted.

Receiving an IRQ for HALTONTERMINATE (HOT) is a feature in the APBH DMA descriptor that allows GPMI to signal to the DMA engine that an error has occurred. If a commmand is stalled due to an error, a HOT signal is sent from the peripheral to the DMA engine and causes an IRQ after terminating the DMA descriptor being executed.

Therefore, it is recommended that software use this signal as follows:

- Always set HALTONTERMINATE to 1 in a DMA descriptor. That way, if a peripheral signals HOT, the transfer will end, leaving the peripheral block and the DMA engine synchronized (but at the end of a command).
- When an IRQ from an APBH channel is received, and the IRQ is determined to be due to an error (as opposed to an IRQONCOMPLETE interrupt) the software should:
  - Reset the channel.
  - Determine the error from error reporting in the peripheral block, then manage the error in the peripheral that is attached to that channel in whatever appropriate way exists for that device (software recovery, device reset, block reset, etc).

Each channel has an eight-bit counting semaphore that controls whether it is in the idle state. When the semaphore is non-zero, the channel is ready to run, process commands and perform DMA transfers. Whenever a command finishes its DMA transfer, it checks the DECREMENT_SEMAPHORE bit. If set, it decrements the counting semaphore. If the semaphore goes to 0 as a result, then the channel enters the idle state and remains there until the semaphore is incremented by the software. When the semaphore goes to non-zero and the channel is in its idle state, then it uses the value in the APBH_CHn_NXTCMDAR register (next command address register) to fetch a pointer to the next command to process.

NOTE

This is a double indirect case. This method allows the software to append to a running command list under the protection of the counting semaphore.

To start processing the first time, software creates the command list to be processed. It writes the address of the first command into the APBH_CHn_NXTCMDAR register, and then writes 1 to the counting semaphore in APBH_CHn_SEMA. The DMA channel loads APBH_CHn_CURCMDAR register and then enters the normal state machine processing for the next command. When the software writes a value to the counting semaphore, it is added to the semaphore count by hardware, protecting the case where both hardware and software are trying to change the semaphore on the same clock edge.

Software can examine the value of APBH_CHn_CURCMDAR at any time to determine the location of the command structure currently being processed.

14.4 NAND Read Status Polling Example

The following figure shows a more complicated scenario.

This subset of a NAND device workload shows that the first two command structures are used during the data-write phase of an NAND device write operation (CLE and ALE transfers omitted for clarity).

- After writing the data, one must wait until the NAND device status register indicates that the write charge has been transferred. This is built into the workload using a check status command in the NAND in a loop created from the next two DMA command structures.
- The NO_DMA_TRANSFER command is shown here performing the read check, followed by a DMA_SENSE command to branch the DMA command structure list, based on the status of a bit in the external NAND device.

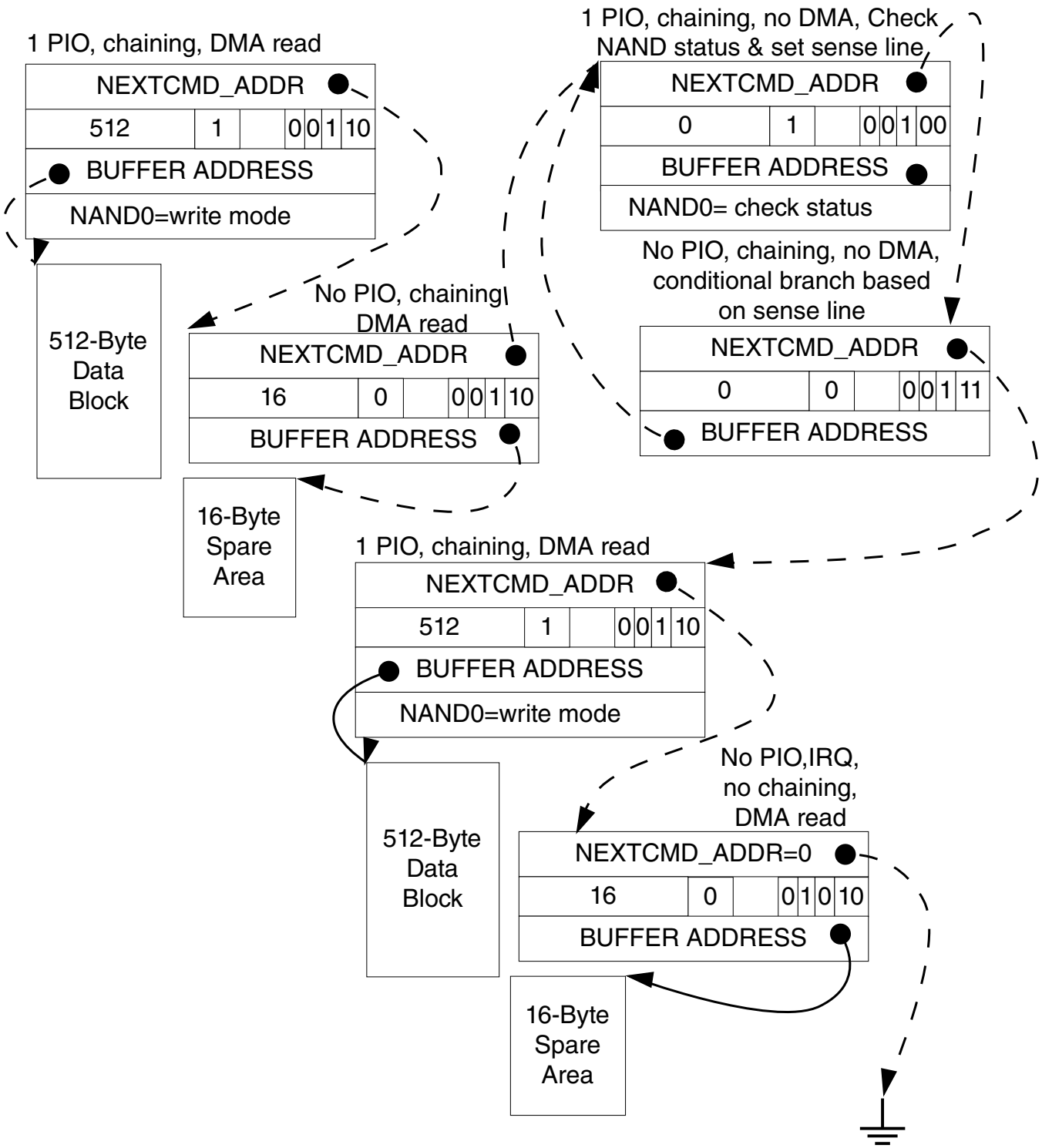


Figure 14-3. AHB-to-APBH Bridge DMA NAND Read Status Polling with DMA Sense Command

The example in the above figure shows the workload continuing immediately to the next NAND page transfer. However, one could perform a second sense operation to see if an error has occurred after the write. One could then point the sense command alternate branch at a NO_DMA_XFER command with the interrupt bit set. If the CHAIN bit is not

set on this failure branch, then the ARM platform is interrupted immediately, and the channel process is also immediately terminated in the presence of a workload-detected NAND error bit.

Note that each word of the three-word DMA command structure corresponds to a PIO register of the DMA that is accessible on the APBH bus. Normally, the DMA copies the next command structure onto these registers for processing at the start of each command by following the value of the pointer previously loaded into the NEXTCMD_ADDR register.

To start DMA processing for the first command, initialize the PIO registers of the desired channel, as follows:

- First, load the next command address register with a pointer to the first command to be loaded.
- Then, write 1 to the counting semaphore register. This causes the DMA to schedule the targeted channel for the DMA command structure load, just as if it had finished its previous command.

14.5 APBH Memory Map/Register Definition

APBH Hardware Register Format Summary

APBH memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_0000	AHB to APBH Bridge Control and Status Register 0 (APBH_CTRL0)	32	R/W	E000_0000h	14.5.1/604
11_0004	AHB to APBH Bridge Control and Status Register 0 (APBH_CTRL0_SET)	32	R/W	E000_0000h	14.5.1/604
11_0008	AHB to APBH Bridge Control and Status Register 0 (APBH_CTRL0_CLR)	32	R/W	E000_0000h	14.5.1/604
11_000C	AHB to APBH Bridge Control and Status Register 0 (APBH_CTRL0_TOG)	32	R/W	E000_0000h	14.5.1/604
11_0010	AHB to APBH Bridge Control and Status Register 1 (APBH_CTRL1)	32	R/W	0000_0000h	14.5.2/606
11_0014	AHB to APBH Bridge Control and Status Register 1 (APBH_CTRL1_SET)	32	R/W	0000_0000h	14.5.2/606
11_0018	AHB to APBH Bridge Control and Status Register 1 (APBH_CTRL1_CLR)	32	R/W	0000_0000h	14.5.2/606

Table continues on the next page...

APBH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_001C	AHB to APBH Bridge Control and Status Register 1 (APBH_CTRL1_TOG)	32	R/W	0000_0000h	14.5.2/606
11_0020	AHB to APBH Bridge Control and Status Register 2 (APBH_CTRL2)	32	R/W	0000_0000h	14.5.3/609
11_0024	AHB to APBH Bridge Control and Status Register 2 (APBH_CTRL2_SET)	32	R/W	0000_0000h	14.5.3/609
11_0028	AHB to APBH Bridge Control and Status Register 2 (APBH_CTRL2_CLR)	32	R/W	0000_0000h	14.5.3/609
11_002C	AHB to APBH Bridge Control and Status Register 2 (APBH_CTRL2_TOG)	32	R/W	0000_0000h	14.5.3/609
11_0030	AHB to APBH Bridge Channel Register (APBH_CHANNEL_CTRL)	32	R/W	0000_0000h	14.5.4/614
11_0034	AHB to APBH Bridge Channel Register (APBH_CHANNEL_CTRL_SET)	32	R/W	0000_0000h	14.5.4/614
11_0038	AHB to APBH Bridge Channel Register (APBH_CHANNEL_CTRL_CLR)	32	R/W	0000_0000h	14.5.4/614
11_003C	AHB to APBH Bridge Channel Register (APBH_CHANNEL_CTRL_TOG)	32	R/W	0000_0000h	14.5.4/614
11_0040	AHB to APBH DMA Device Assignment Register (APBH_DEVSEL)	32	R/W	0000_0000h	14.5.5/615
11_0050	AHB to APBH DMA burst size (APBH_DMA_BURST_SIZE)	32	R/W	0055_5555h	14.5.6/616
11_0060	AHB to APBH DMA Debug Register (APBH_DEBUG)	32	R/W	0000_0000h	14.5.7/617
11_0100	APBH DMA Channel n Current Command Address Register (APBH_CH0_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0110	APBH DMA Channel n Next Command Address Register (APBH_CH0_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0120	APBH DMA Channel n Command Register (APBH_CH0_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0130	APBH DMA Channel n Buffer Address Register (APBH_CH0_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0140	APBH DMA Channel n Semaphore Register (APBH_CH0_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0150	AHB to APBH DMA Channel n Debug Information (APBH_CH0_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0160	AHB to APBH DMA Channel n Debug Information (APBH_CH0_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0170	APBH DMA Channel n Current Command Address Register (APBH_CH1_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0180	APBH DMA Channel n Next Command Address Register (APBH_CH1_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0190	APBH DMA Channel n Command Register (APBH_CH1_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_01A0	APBH DMA Channel n Buffer Address Register (APBH_CH1_BAR)	32	R/W	0000_0000h	14.5.11/ 621

Table continues on the next page...

APBH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_01B0	APBH DMA Channel n Semaphore Register (APBH_CH1_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_01C0	AHB to APBH DMA Channel n Debug Information (APBH_CH1_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_01D0	AHB to APBH DMA Channel n Debug Information (APBH_CH1_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_01E0	APBH DMA Channel n Current Command Address Register (APBH_CH2_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_01F0	APBH DMA Channel n Next Command Address Register (APBH_CH2_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0200	APBH DMA Channel n Command Register (APBH_CH2_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0210	APBH DMA Channel n Buffer Address Register (APBH_CH2_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0220	APBH DMA Channel n Semaphore Register (APBH_CH2_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0230	AHB to APBH DMA Channel n Debug Information (APBH_CH2_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0240	AHB to APBH DMA Channel n Debug Information (APBH_CH2_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0250	APBH DMA Channel n Current Command Address Register (APBH_CH3_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0260	APBH DMA Channel n Next Command Address Register (APBH_CH3_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0270	APBH DMA Channel n Command Register (APBH_CH3_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0280	APBH DMA Channel n Buffer Address Register (APBH_CH3_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0290	APBH DMA Channel n Semaphore Register (APBH_CH3_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_02A0	AHB to APBH DMA Channel n Debug Information (APBH_CH3_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_02B0	AHB to APBH DMA Channel n Debug Information (APBH_CH3_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_02C0	APBH DMA Channel n Current Command Address Register (APBH_CH4_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_02D0	APBH DMA Channel n Next Command Address Register (APBH_CH4_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_02E0	APBH DMA Channel n Command Register (APBH_CH4_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_02F0	APBH DMA Channel n Buffer Address Register (APBH_CH4_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0300	APBH DMA Channel n Semaphore Register (APBH_CH4_SEMA)	32	R/W	0000_0000h	14.5.12/ 622

Table continues on the next page...

APBH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_0310	AHB to APBH DMA Channel n Debug Information (APBH_CH4_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0320	AHB to APBH DMA Channel n Debug Information (APBH_CH4_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0330	APBH DMA Channel n Current Command Address Register (APBH_CH5_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0340	APBH DMA Channel n Next Command Address Register (APBH_CH5_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0350	APBH DMA Channel n Command Register (APBH_CH5_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0360	APBH DMA Channel n Buffer Address Register (APBH_CH5_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0370	APBH DMA Channel n Semaphore Register (APBH_CH5_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0380	AHB to APBH DMA Channel n Debug Information (APBH_CH5_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0390	AHB to APBH DMA Channel n Debug Information (APBH_CH5_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_03A0	APBH DMA Channel n Current Command Address Register (APBH_CH6_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_03B0	APBH DMA Channel n Next Command Address Register (APBH_CH6_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_03C0	APBH DMA Channel n Command Register (APBH_CH6_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_03D0	APBH DMA Channel n Buffer Address Register (APBH_CH6_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_03E0	APBH DMA Channel n Semaphore Register (APBH_CH6_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_03F0	AHB to APBH DMA Channel n Debug Information (APBH_CH6_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0400	AHB to APBH DMA Channel n Debug Information (APBH_CH6_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0410	APBH DMA Channel n Current Command Address Register (APBH_CH7_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0420	APBH DMA Channel n Next Command Address Register (APBH_CH7_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0430	APBH DMA Channel n Command Register (APBH_CH7_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0440	APBH DMA Channel n Buffer Address Register (APBH_CH7_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0450	APBH DMA Channel n Semaphore Register (APBH_CH7_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0460	AHB to APBH DMA Channel n Debug Information (APBH_CH7_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623

Table continues on the next page...

APBH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_0470	AHB to APBH DMA Channel n Debug Information (APBH_CH7_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0480	APBH DMA Channel n Current Command Address Register (APBH_CH8_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0490	APBH DMA Channel n Next Command Address Register (APBH_CH8_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_04A0	APBH DMA Channel n Command Register (APBH_CH8_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_04B0	APBH DMA Channel n Buffer Address Register (APBH_CH8_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_04C0	APBH DMA Channel n Semaphore Register (APBH_CH8_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_04D0	AHB to APBH DMA Channel n Debug Information (APBH_CH8_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_04E0	AHB to APBH DMA Channel n Debug Information (APBH_CH8_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_04F0	APBH DMA Channel n Current Command Address Register (APBH_CH9_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0500	APBH DMA Channel n Next Command Address Register (APBH_CH9_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0510	APBH DMA Channel n Command Register (APBH_CH9_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0520	APBH DMA Channel n Buffer Address Register (APBH_CH9_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0530	APBH DMA Channel n Semaphore Register (APBH_CH9_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0540	AHB to APBH DMA Channel n Debug Information (APBH_CH9_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0550	AHB to APBH DMA Channel n Debug Information (APBH_CH9_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0560	APBH DMA Channel n Current Command Address Register (APBH_CH10_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0570	APBH DMA Channel n Next Command Address Register (APBH_CH10_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0580	APBH DMA Channel n Command Register (APBH_CH10_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0590	APBH DMA Channel n Buffer Address Register (APBH_CH10_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_05A0	APBH DMA Channel n Semaphore Register (APBH_CH10_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_05B0	AHB to APBH DMA Channel n Debug Information (APBH_CH10_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_05C0	AHB to APBH DMA Channel n Debug Information (APBH_CH10_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626

Table continues on the next page...

APBH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_05D0	APBH DMA Channel n Current Command Address Register (APBH_CH11_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_05E0	APBH DMA Channel n Next Command Address Register (APBH_CH11_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_05F0	APBH DMA Channel n Command Register (APBH_CH11_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0600	APBH DMA Channel n Buffer Address Register (APBH_CH11_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0610	APBH DMA Channel n Semaphore Register (APBH_CH11_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0620	AHB to APBH DMA Channel n Debug Information (APBH_CH11_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0630	AHB to APBH DMA Channel n Debug Information (APBH_CH11_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0640	APBH DMA Channel n Current Command Address Register (APBH_CH12_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_0650	APBH DMA Channel n Next Command Address Register (APBH_CH12_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0660	APBH DMA Channel n Command Register (APBH_CH12_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0670	APBH DMA Channel n Buffer Address Register (APBH_CH12_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0680	APBH DMA Channel n Semaphore Register (APBH_CH12_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0690	AHB to APBH DMA Channel n Debug Information (APBH_CH12_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_06A0	AHB to APBH DMA Channel n Debug Information (APBH_CH12_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_06B0	APBH DMA Channel n Current Command Address Register (APBH_CH13_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_06C0	APBH DMA Channel n Next Command Address Register (APBH_CH13_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_06D0	APBH DMA Channel n Command Register (APBH_CH13_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_06E0	APBH DMA Channel n Buffer Address Register (APBH_CH13_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_06F0	APBH DMA Channel n Semaphore Register (APBH_CH13_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0700	AHB to APBH DMA Channel n Debug Information (APBH_CH13_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0710	AHB to APBH DMA Channel n Debug Information (APBH_CH13_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0720	APBH DMA Channel n Current Command Address Register (APBH_CH14_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618

Table continues on the next page...

APBH memory map (continued)

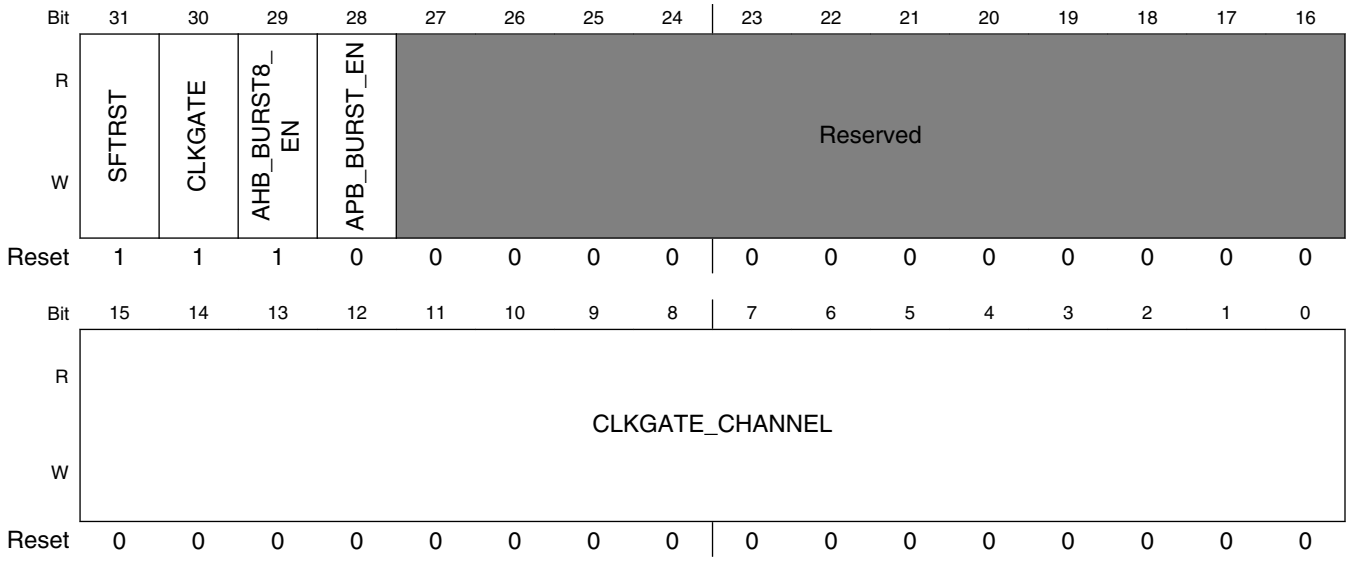
Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_0730	APBH DMA Channel n Next Command Address Register (APBH_CH14_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_0740	APBH DMA Channel n Command Register (APBH_CH14_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_0750	APBH DMA Channel n Buffer Address Register (APBH_CH14_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_0760	APBH DMA Channel n Semaphore Register (APBH_CH14_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_0770	AHB to APBH DMA Channel n Debug Information (APBH_CH14_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_0780	AHB to APBH DMA Channel n Debug Information (APBH_CH14_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0790	APBH DMA Channel n Current Command Address Register (APBH_CH15_CURCMDAR)	32	R/W	0000_0000h	14.5.8/618
11_07A0	APBH DMA Channel n Next Command Address Register (APBH_CH15_NXTCMDAR)	32	R/W	0000_0000h	14.5.9/619
11_07B0	APBH DMA Channel n Command Register (APBH_CH15_CMD)	32	R/W	0000_0000h	14.5.10/ 619
11_07C0	APBH DMA Channel n Buffer Address Register (APBH_CH15_BAR)	32	R/W	0000_0000h	14.5.11/ 621
11_07D0	APBH DMA Channel n Semaphore Register (APBH_CH15_SEMA)	32	R/W	0000_0000h	14.5.12/ 622
11_07E0	AHB to APBH DMA Channel n Debug Information (APBH_CH15_DEBUG1)	32	R/W	00A0_0000h	14.5.13/ 623
11_07F0	AHB to APBH DMA Channel n Debug Information (APBH_CH15_DEBUG2)	32	R/W	0000_0000h	14.5.14/ 626
11_0800	APBH Bridge Version Register (APBH_VERSION)	32	R/W	0301_0000h	14.5.15/ 627

14.5.1 AHB to APBH Bridge Control and Status Register 0 (APBH_CTRL0*n*)

The APBH CTRL 0 provides overall control of the AHB to APBH bridge and DMA.

This register contains module softreset, clock gating, channel clock gating/freeze bits.

Address: 11_0000h base + 0h offset + (4d × i), where i=0d to 3d



APBH_CTRL0*n* field descriptions

Field	Description
31 SFTRST	Set this bit to zero to enable normal APBH DMA operation. Set this bit to one (default) to disable clocking with the APBH DMA and hold it in its reset (lowest power) state. This bit can be turned on and then off to reset the APBH DMA block to its default state.
30 CLKGATE	This bit must be set to zero for normal operation. When set to one it gates off the clocks to the block.
29 AHB_BURST8_EN	Set this bit to one (default) to enable AHB 8-beat burst. Set to zero to disable 8-beat burst on AHB interface.
28 APB_BURST_EN	Set this bit to one to enable apb master do a continous transfers when a device request a burst dma. Set to zero will treat a burst dma request as 4/8 individual requests.
27–16 RSVD0	This field is reserved. Reserved, always set to zero.
CLKGATE_CHANNEL	These bits must be set to zero for normal operation of each channel. When set to one they gate off the individual clocks to the channels. 0x0001 **NAND0** — 0x0002 **NAND1** — 0x0004 **NAND2** — 0x0008 **NAND3** — 0x0010 **NAND4** — 0x0020 **NAND5** — 0x0040 **NAND6** — 0x0080 **NAND7** — 0x0100 **SSP** —

14.5.2 AHB to APBH Bridge Control and Status Register 1 (APBH_CTRL1*n*)

The APBH CTRL one provides overall control of the interrupts generated by the AHB to APBH DMA. This register contains the per channel interrupt status bits and the per channel interrupt enable bits. Each channel has a dedicated interrupt vector in the vectored interrupt controller.

EXAMPLE

```
                    BF_WR(APBH_CTRL1, CH5_CMDCMPLT_IRQ, 0);  // use bitfield write macro
                    BF_APBH_CTRL1.CH5_CMDCMPLT_IRQ = 0;      // or, assign to register
struct's bitfield
```

Address: 11_0000h base + 10h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CH15_CMDCMPLT_IRQ_EN	CH14_CMDCMPLT_IRQ_EN	CH13_CMDCMPLT_IRQ_EN	CH12_CMDCMPLT_IRQ_EN	CH11_CMDCMPLT_IRQ_EN	CH10_CMDCMPLT_IRQ_EN	CH9_CMDCMPLT_IRQ_EN	CH8_CMDCMPLT_IRQ_EN	CH7_CMDCMPLT_IRQ_EN	CH6_CMDCMPLT_IRQ_EN	CH5_CMDCMPLT_IRQ_EN	CH4_CMDCMPLT_IRQ_EN	CH3_CMDCMPLT_IRQ_EN	CH2_CMDCMPLT_IRQ_EN	CH1_CMDCMPLT_IRQ_EN	CH0_CMDCMPLT_IRQ_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH15_CMDCMPLT_IRQ	CH14_CMDCMPLT_IRQ	CH13_CMDCMPLT_IRQ	CH12_CMDCMPLT_IRQ	CH11_CMDCMPLT_IRQ	CH10_CMDCMPLT_IRQ	CH9_CMDCMPLT_IRQ	CH8_CMDCMPLT_IRQ	CH7_CMDCMPLT_IRQ	CH6_CMDCMPLT_IRQ	CH5_CMDCMPLT_IRQ	CH4_CMDCMPLT_IRQ	CH3_CMDCMPLT_IRQ	CH2_CMDCMPLT_IRQ	CH1_CMDCMPLT_IRQ	CH0_CMDCMPLT_IRQ
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_CTRL1*n* field descriptions

Field	Description
31 CH15_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 15.
30 CH14_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 14.

Table continues on the next page...

APBH_CTRL1*n* field descriptions (continued)

Field	Description
29 CH13_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 13.
28 CH12_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 12.
27 CH11_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 11.
26 CH10_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 10.
25 CH9_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 9.
24 CH8_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 8.
23 CH7_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 7.
22 CH6_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 6.
21 CH5_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 5.
20 CH4_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 4.
19 CH3_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 3.
18 CH2_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 2.

Table continues on the next page...

APBH_CTRL1*n* field descriptions (continued)

Field	Description
17 CH1_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 1.
16 CH0_ CMDCMPLT_ IRQ_EN	Setting this bit enables the generation of an interrupt request for APBH DMA channel 0.
15 CH15_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 15. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
14 CH14_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 14. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
13 CH13_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 13. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
12 CH12_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 12. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
11 CH11_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 11. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
10 CH10_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 10. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
9 CH9_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 9. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
8 CH8_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA Channel 8. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
7 CH7_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 7. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
6 CH6_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 6. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.

Table continues on the next page...

APBH_CTRL1*n* field descriptions (continued)

Field	Description
5 CH5_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 5. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
4 CH4_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 4. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
3 CH3_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 3. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
2 CH2_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 2. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
1 CH1_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 1. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.
0 CH0_ CMDCMPLT_ IRQ	Interrupt request status bit for APBH DMA channel 0. This sticky bit is set by DMA hardware and reset by software. It is ANDed with its corresponding enable bit to generate an interrupt.

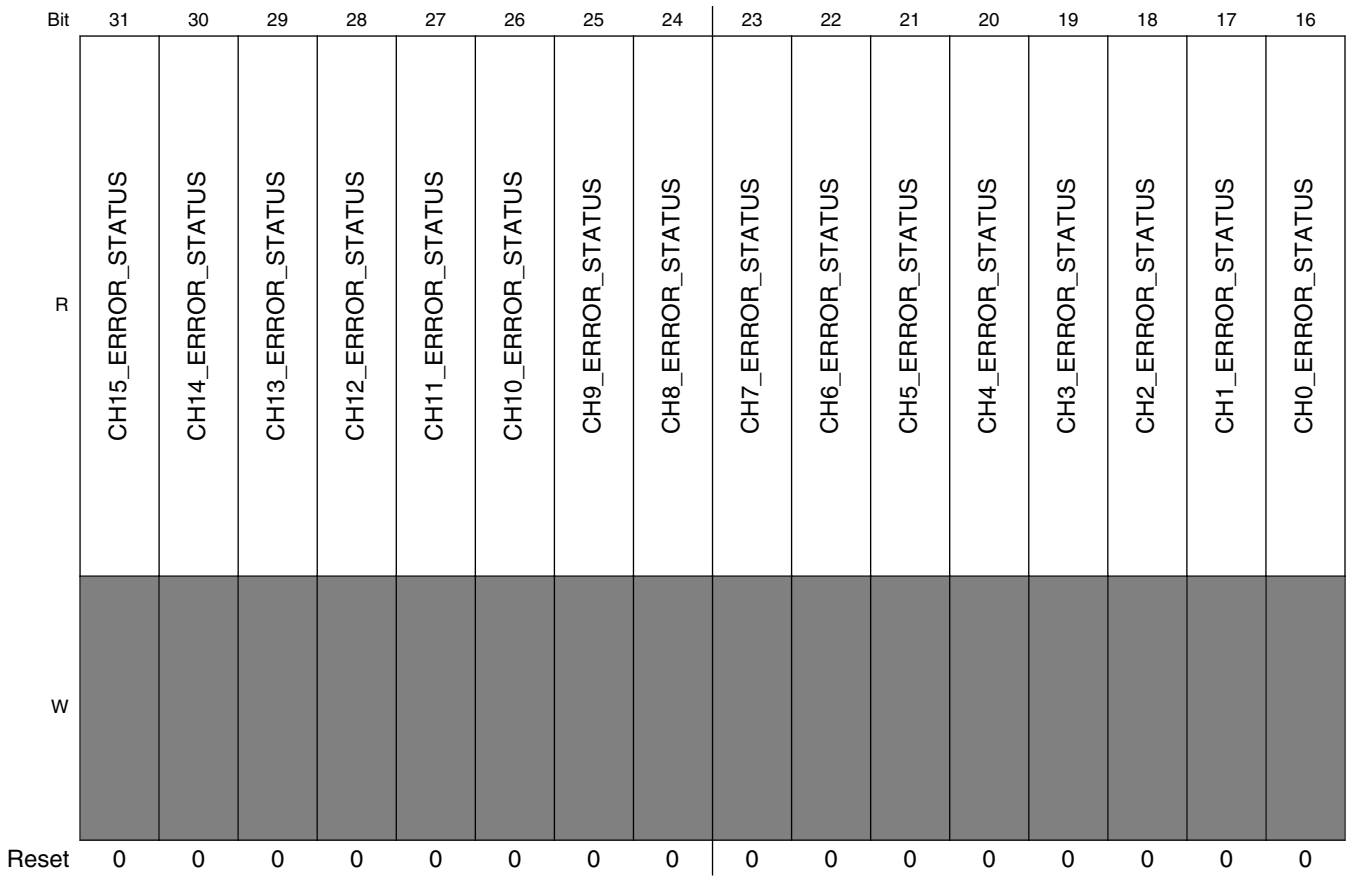
14.5.3 AHB to APBH Bridge Control and Status Register 2 (APBH_CTRL2*n*)

The APBH CTRL 2 provides channel error interrupts generated by the AHB to APBH DMA. This register contains the per channel interrupt status bits and the per channel interrupt enable bits. Each channel has a dedicated interrupt vector in the vectored interrupt controller.

EXAMPLE

```
                    BF_WR(APBH_CTRL1, CH5_CMDCMPLT_IRQ, 0);  // use bitfield write macro
                    BF_APBH_CTRL1.CH5_CMDCMPLT_IRQ = 0;      // or, assign to register
struct's bitfield
```

Address: 11_0000h base + 20h offset + (4d × i), where i=0d to 3d



APBH_CTRL2*n* field descriptions

Field	Description
31 CH15_ERROR_ STATUS	Error status bit for APBH DMA Channel 15. Valid when corresponding Error IRQ is set. 1 - AHB bus error

Table continues on the next page...

APBH_CTRL2*n* field descriptions (continued)

Field	Description
	0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
30 CH14_ERROR_ STATUS	Error status bit for APBH DMA Channel 14. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
29 CH13_ERROR_ STATUS	Error status bit for APBH DMA Channel 13. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
28 CH12_ERROR_ STATUS	Error status bit for APBH DMA Channel 12. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
27 CH11_ERROR_ STATUS	Error status bit for APBH DMA Channel 11. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
26 CH10_ERROR_ STATUS	Error status bit for APBH DMA Channel 10. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
25 CH9_ERROR_ STATUS	Error status bit for APBH DMA Channel 9. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
24 CH8_ERROR_ STATUS	Error status bit for APBH DMA Channel 8. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination.

Table continues on the next page...

APBH_CTRL2*n* field descriptions (continued)

Field	Description
	0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
23 CH7_ERROR_STATUS	Error status bit for APBX DMA Channel 7. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
22 CH6_ERROR_STATUS	Error status bit for APBX DMA Channel 6. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
21 CH5_ERROR_STATUS	Error status bit for APBX DMA Channel 5. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
20 CH4_ERROR_STATUS	Error status bit for APBX DMA Channel 4. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
19 CH3_ERROR_STATUS	Error status bit for APBX DMA Channel 3. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
18 CH2_ERROR_STATUS	Error status bit for APBX DMA Channel 2. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
17 CH1_ERROR_STATUS	Error status bit for APBX DMA Channel 1. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.

Table continues on the next page...

APBH_CTRL2*n* field descriptions (continued)

Field	Description
16 CH0_ERROR_STATUS	Error status bit for APBX DMA Channel 0. Valid when corresponding Error IRQ is set. 1 - AHB bus error 0 - channel early termination. 0x0 **TERMINATION** — An early termination from the device causes error IRQ. 0x1 **BUS_ERROR** — An AHB bus error causes error IRQ.
15 CH15_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 15. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
14 CH14_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 14. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
13 CH13_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 13. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
12 CH12_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 12. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
11 CH11_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 11. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
10 CH10_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 10. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
9 CH9_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 9. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
8 CH8_ERROR_IRQ	Error interrupt status bit for APBH DMA Channel 8. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
7 CH7_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 7. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
6 CH6_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 6. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
5 CH5_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 5. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
4 CH4_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 4. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
3 CH3_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 3. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
2 CH2_ERROR_IRQ	Error interrupt status bit for APBX DMA Channel 2. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.

Table continues on the next page...

APBH_CTRL2*n* field descriptions (continued)

Field	Description
1 CH1_ERROR_ IRQ	Error interrupt status bit for APBX DMA Channel 1. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.
0 CH0_ERROR_ IRQ	Error interrupt status bit for APBX DMA Channel 0. This sticky bit is set by DMA hardware and reset by software. It is ORed with the corresponding cmdcmplt irq to generate an irq to ARM.

14.5.4 AHB to APBH Bridge Channel Register (APBH_CHANNEL_CTRL*n*)

The APBH CHANNEL CTRL provides reset/freeze control of each DMA channel. This register contains individual channel reset/freeze bits.

Address: 11_0000h base + 30h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RESET_CHANNEL																FREEZE_CHANNEL															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_CHANNEL_CTRL*n* field descriptions

Field	Description
31–16 RESET_ CHANNEL	Setting a bit in this field causes the DMA controller to take the corresponding channel through its reset state. The bit is reset after the channel resources are cleared. 0x0001 **NAND0** — 0x0002 **NAND1** — 0x0004 **NAND2** — 0x0008 **NAND3** — 0x0010 **NAND4** — 0x0020 **NAND5** — 0x0040 **NAND6** — 0x0080 **NAND7** — 0x0100 **SSP** —
FREEZE_ CHANNEL	Setting a bit in this field will freeze the DMA channel associated with it. This field is a direct input to the DMA channel arbiter. When frozen, the channel is deined access to the central DMA resources. 0x0001 **NAND0** — 0x0002 **NAND1** — 0x0004 **NAND2** — 0x0008 **NAND3** — 0x0010 **NAND4** — 0x0020 **NAND5** — 0x0040 **NAND6** —

Table continues on the next page...

APBH_CHANNEL_CTRL*n* field descriptions (continued)

Field	Description
	0x0080 **NAND7** — 0x0100 **SSP** —

14.5.5 AHB to APBH DMA Device Assignment Register (APBH_DEVSEL)

This register allows reassignment of the APBH device connected to the DMA Channels.

In this chip, APBH DMA channel resource is enough for high speed peripherals, so this register is of no use and reserved.

Address: 11_0000h base + 40h offset = 11_0040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		Reserved	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		Reserved	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_DEVSEL field descriptions

Field	Description
31–30 CH15	This field is reserved. Reserved.
29–28 CH14	This field is reserved. Reserved.
27–26 CH13	This field is reserved. Reserved.
25–24 CH12	This field is reserved. Reserved.
23–22 CH11	This field is reserved. Reserved.
21–20 CH10	This field is reserved. Reserved.
19–18 CH9	This field is reserved. Reserved.
17–16 CH8	This field is reserved. Reserved.
15–14 CH7	This field is reserved. Reserved.
13–12 CH6	This field is reserved. Reserved.

Table continues on the next page...

APBH_DEVSEL field descriptions (continued)

Field	Description
11–10 CH5	This field is reserved. Reserved.
9–8 CH4	This field is reserved. Reserved.
7–6 CH3	This field is reserved. Reserved.
5–4 CH2	This field is reserved. Reserved.
3–2 CH1	This field is reserved. Reserved.
CH0	This field is reserved. Reserved.

14.5.6 AHB to APBH DMA burst size (APBH_DMA_BURST_SIZE)

This register programs the apbh burst size of the APBH DMA devices when a DMA burst request is issued.

This register provides a mechanism for assigning the device.

Address: 11_0000h base + 50h offset = 11_0050h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		Reserved		CH8	
Reset	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH7		CH6		CH5		CH4		CH3		CH2		CH1		CH0	
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

APBH_DMA_BURST_SIZE field descriptions

Field	Description
31–30 CH15	This field is reserved. Reserved.
29–28 CH14	This field is reserved. Reserved.
27–26 CH13	This field is reserved. Reserved.
25–24 CH12	This field is reserved. Reserved.
23–22 CH11	This field is reserved. Reserved.

Table continues on the next page...

APBH_DMA_BURST_SIZE field descriptions (continued)

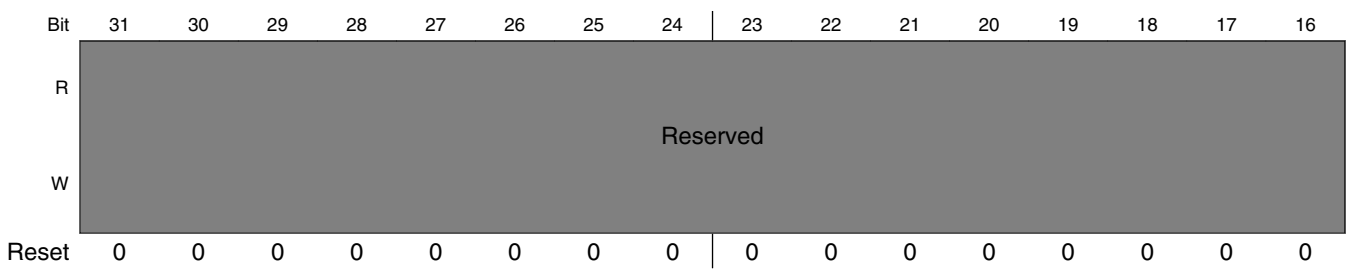
Field	Description
21–20 CH10	This field is reserved. Reserved.
19–18 CH9	This field is reserved. Reserved.
17–16 CH8	DMA burst size for SSP. 0x0 **BURST0** — 0x1 **BURST4** — 0x2 **BURST8** —
15–14 CH7	DMA burst size for GPMI channel 7. Do not change. GPMI only support burst size 4.
13–12 CH6	DMA burst size for GPMI channel 6. Do not change. GPMI only support burst size 4.
11–10 CH5	DMA burst size for GPMI channel 5. Do not change. GPMI only support burst size 4.
9–8 CH4	DMA burst size for GPMI channel 4. Do not change. GPMI only support burst size 4.
7–6 CH3	DMA burst size for GPMI channel 3. Do not change. GPMI only support burst size 4.
5–4 CH2	DMA burst size for GPMI channel 2. Do not change. GPMI only support burst size 4.
3–2 CH1	DMA burst size for GPMI channel 1. Do not change. GPMI only support burst size 4.
CH0	DMA burst size for GPMI channel 0. Do not change. GPMI only support burst size 4.

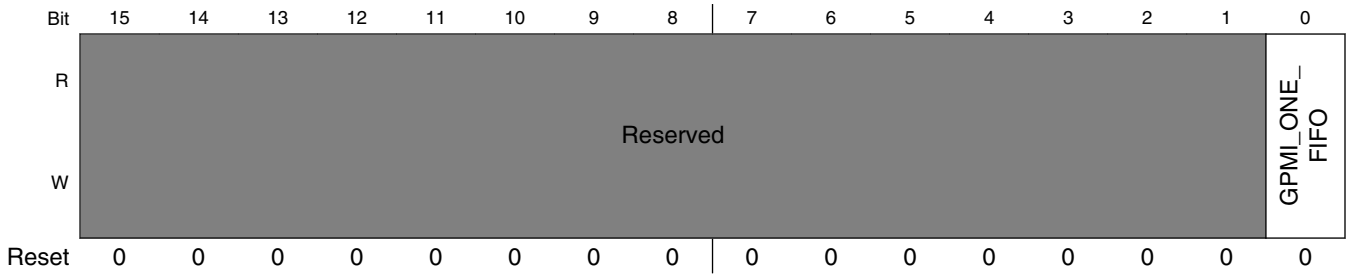
14.5.7 AHB to APBH DMA Debug Register (APBH_DEBUG)

This register is for debug purpose.

The debug register is for internal use only. Not recommend for customer useage.

Address: 11_0000h base + 60h offset = 11_0060h





APBH_DEBUG field descriptions

Field	Description
31–1 -	This field is reserved. Reserved, always set to zero.
0 GPMI_ONE_FIFO	Set to 0ne and the 8 GPMI channels will share the DMA FIFO, and when set to zero, the 8 GPMI channels will use its own DMA FIFO.

14.5.8 APBH DMA Channel n Current Command Address Register (APBH_CH*n*_CURCMDAR)

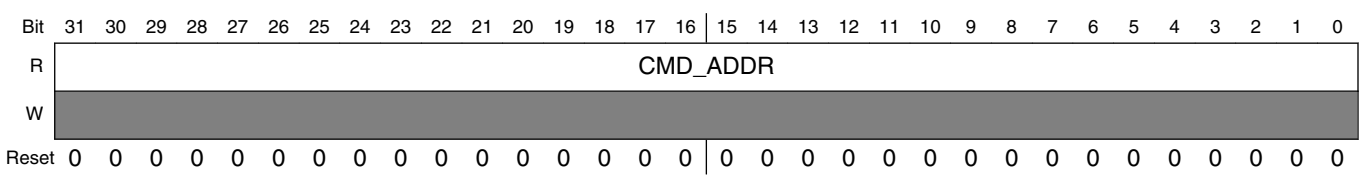
The APBH DMA channel n current command address register points to the multiword command that is currently being executed. Commands are threaded on the command address.

APBH DMA Channel n is controlled by a variable sized command structure. This register points to the command structure currently being executed.

EXAMPLE

```
pCurCmd = (apbh_chn_cmd_t *) APBH_CHn_CURCMDAR_RD(0);              // read the whole
register, since there is only one field
pCurCmd = (apbh_chn_cmd_t *) BF_RDn(APBH_CHn_CURCMDAR, 0, CMD_ADDR);  // or, use multi-
register bitfield read macro
pCurCmd = (apbh_chn_cmd_t *) APBH_CHn_CURCMDAR(0).CMD_ADDR;        // or, assign from
bitfield of indexed register's struct
```

Address: 11_0000h base + 100h offset + (112d × i), where i=0d to 15d



APBH_CH*n*_CURCMDAR field descriptions

Field	Description
CMD_ADDR	Pointer to command structure currently being processed for channel n.

14.5.9 APBH DMA Channel n Next Command Address Register (APBH_CH*n*_NXTCMDAR)

The APBH DMA Channel n Next Command Address register contains the address of the next multiword command to be executed. Commands are threaded on the command address. Set CHAIN to 1 in the DMA command word to process command lists.

APBH DMA Channel n is controlled by a variable sized command structure. Software loads this register with the address of the first command structure to process and increments the Channel n semaphore to start processing. This register points to the next command structure to be executed when the current command is completed.

EXAMPLE

```
APBH_CHn_NXTCMDAR_WR(0, (reg32_t) pCommandTwoStructure);         // write the entire
register, since there is only one field
BF_WRn(APBH_CHn_NXTCMDAR, 0, (reg32_t) pCommandTwoStructure);       // or, use multi-
register bitfield write macro
APBH_CHn_NXTCMDAR(0).CMD_ADDR = (reg32_t) pCommandTwoStructure;  // or, assign to bitfield
of indexed register's struct
```

Address: 11_0000h base + 110h offset + (112d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CMD_ADDR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_CH*n*_NXTCMDAR field descriptions

Field	Description
CMD_ADDR	Pointer to next command structure for channel n.

14.5.10 APBH DMA Channel n Command Register (APBH_CH*n*_CMD)

The APBH DMA Channel n command register specifies the DMA transaction to perform for the current command chain item.

The command register controls the overall operation of each DMA command for this channel. It includes the number of bytes to transfer to or from the device, the number of APB PIO command words included with this command structure, whether to interrupt at command completion, whether to chain an additional command to the end of this one and whether this transfer is a read or write DMA transfer.

EXAMPLE

```
apbh_chn_cmd_t  dma_cmd;
dma_cmd.XFER_COUNT = 512;                                    // transfer 512 bytes
dma_cmd.COMMAND = BV_APBH_CHn_CMD_COMMAND__DMA_WRITE;  // transfer to system memory from
peripheral device
dma_cmd.CHAIN = 1;                                           // chain an additional command
structure on to the list
dma_cmd.IRQONCMPLT = 1;                                      // generate an interrupt on
completion of this command structure
```

Address: 11_0000h base + 120h offset + (112d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	XFER_COUNT															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CMDWORDS				Reserved			HALTONTERMINATE	WAIT4ENDCMD	SEMAPHORE	NANDWAIT4READY	NANDLOCK	IRQONCMPLT	CHAIN	COMMAND	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_CH*n*_CMD field descriptions

Field	Description
31–16 XFER_COUNT	This field indicates the number of bytes to transfer to or from the appropriate PIO register in the GPMI0 device. A value of 0 indicates a 64 KBytes transfer.
15–12 CMDWORDS	This field indicates the number of command words to send to the GPMI0, starting with the base PIO address of the GPMI0 control register and incrementing from there. Zero means transfer NO command words
11–9 -	This field is reserved. Reserved, always set to zero.
8 HALTONTERMINATE	A value of one indicates that the channel will immeditately terminate the current descriptor and halt the DMA channel if a terminate signal is set. A value of 0 will still cause an immediate terminate of the channel if the terminate signal is set, but the channel will continue as if the count had been exhausted, meaning it will honor IRQONCMPLT, CHAIN, SEMAPHORE, and WAIT4ENDCMD.
7 WAIT4ENDCMD	A value of one indicates that the channel will wait for the end of command signal to be sent from the APBH device to the DMA before starting the next DMA command.
6 SEMAPHORE	A value of one indicates that the channel will decrement its semaphore at the completion of the current command structure. If the semaphore decrements to zero, then this channel stalls until software increments it again.
5 NANDWAIT4READY	A value of one indicates that the NAND DMA channel will will wait until the NAND device reports "ready" before executing the command. It is ignored for non-NAND DMA channels.
4 NANDLOCK	A value of one indicates that the NAND DMA channel will remain "locked" in the arbiter at the expense of other NAND DMA channels. It is ignored for non-NAND DMA channels.
3 IRQONCMPLT	A value of one indicates that the channel will cause the interrupt status bit to be set upon completion of the current command, i.e. after the DMA transfer is complete.
2 CHAIN	A value of one indicates that another command is chained onto the end of the current command structure. At the completion of the current command, this channel will follow the pointer in APBH_CHn_CMDAR to find the next command.
COMMAND	This bitfield indicates the type of current command: 0x0 **NO_DMA_XFER** — Perform any requested PIO word transfers but terminate command before any DMA transfer. 0x1 **DMA_WRITE** — Perform any requested PIO word transfers and then perform a DMA transfer from the peripheral for the specified number of bytes. 0x2 **DMA_READ** — Perform any requested PIO word transfers and then perform a DMA transfer to the peripheral for the specified number of bytes. 0x3 **DMA_SENSE** — Perform any requested PIO word transfers and then perform a conditional branch to the next chained device. Follow the NEXCMD_ADDR pointer if the perpheral sense is true. Follow the BUFFER_ADDRESS as a chain pointer if the peripheral sense line is false.

14.5.11 APBH DMA Channel n Buffer Address Register (APBH_CH*n*_BAR)

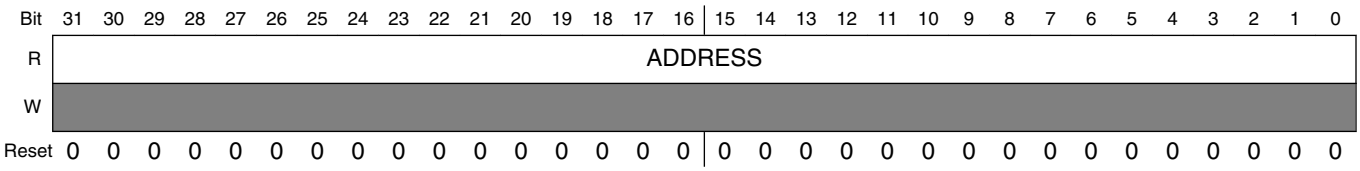
The APBH DMA Channel n buffer address register contains a pointer to the data buffer for the transfer. For immediate forms, the data is taken from this register. This is a byte address which means transfers can start on any byte boundary.

This register holds a pointer to the data buffer in system memory. After the command values have been read into the DMA controller and the device controlled by this channel, then the DMA transfer will begin, to or from the buffer pointed to by this register.

EXAMPLE

```
apbh_chn_bar_t  dma_data;
dma_data.ADDRESS = (reg32_t) pDataBuffer;
```

Address: 11_0000h base + 130h offset + (112d × i), where i=0d to 15d



APBH_CH*n*_BAR field descriptions

Field	Description
ADDRESS	Address of system memory buffer to be read or written over the AHB bus.

14.5.12 APBH DMA Channel n Semaphore Register (APBH_CH*n*_SEMA)

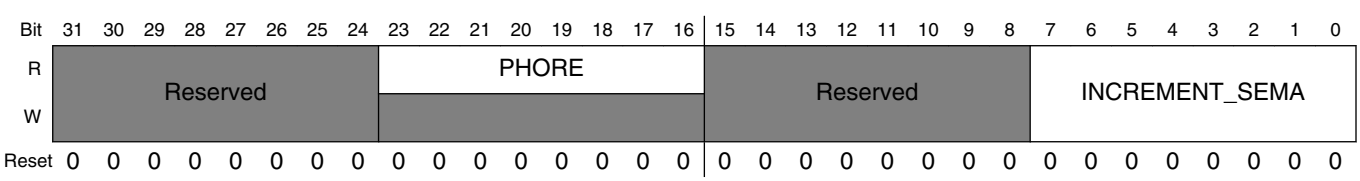
The APBH DMA Channel n semaphore register is used to synchronize the ARM platform instruction stream and the DMA chain processing state.

Each DMA channel has an 8 bit counting semaphore that is used to synchronize between the program stream and and the DMA chain processing. DMA processing continues until the DMA attempts to decrement a semaphore that has already reached a value of zero. When the attempt is made, the DMA channel is stalled until software increments the semaphore count.

EXAMPLE

```
BF_WR(APBH_CHn_SEMA, 0, INCREMENT_SEMA, 2);     // increment semaphore by two
current_sema = BF_RD(APBH_CHn_SEMA, 0, PHORE);  // get instantaneous value
```

Address: 11_0000h base + 140h offset + (112d × i), where i=0d to 15d



APBH_CH*n*_SEMA field descriptions

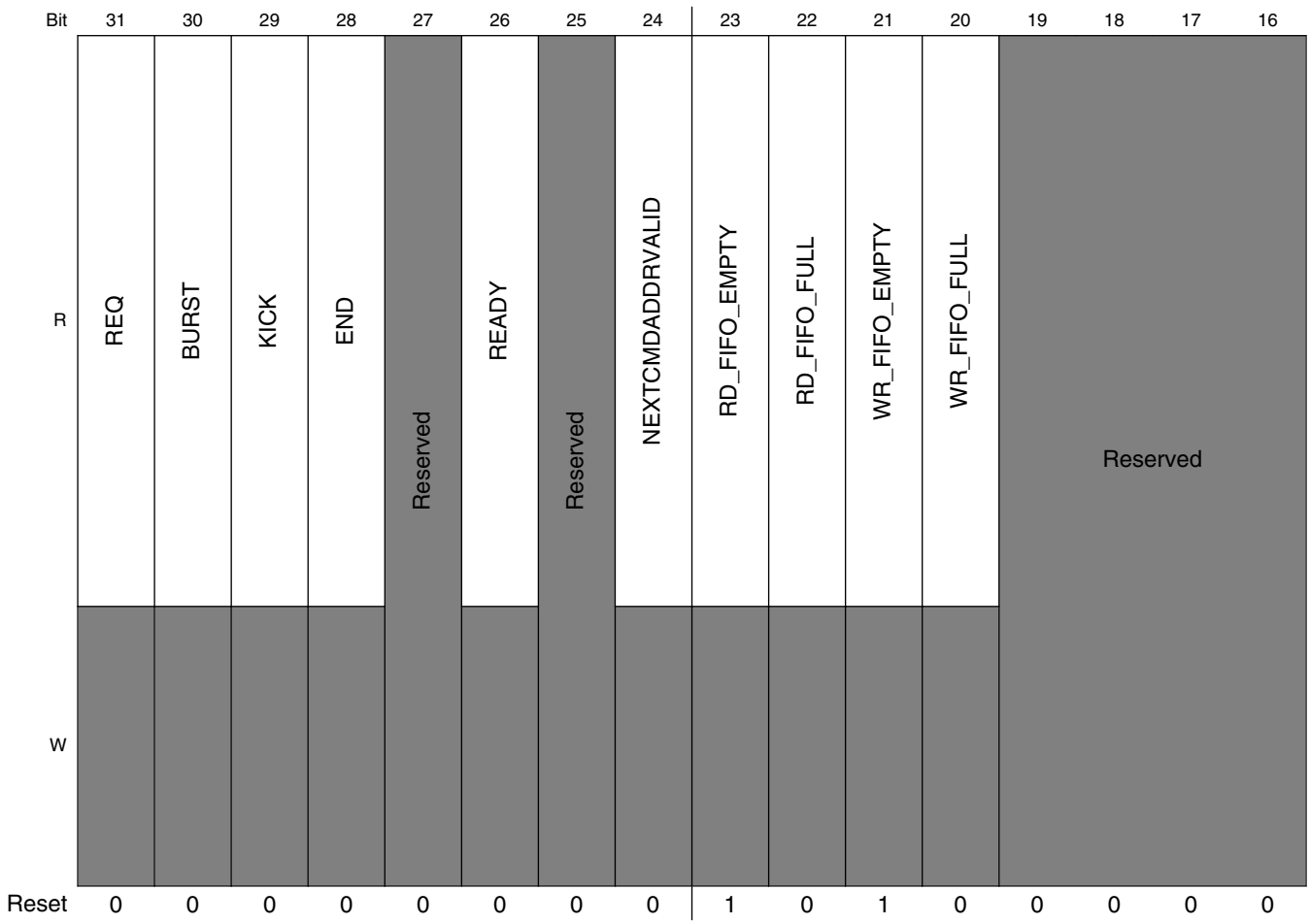
Field	Description
31–24 -	This field is reserved. Reserved, always set to zero.
23–16 PHORE	This read-only field shows the current (instantaneous) value of the semaphore counter.
15–8 -	This field is reserved. Reserved, always set to zero.
INCREMENT_ SEMA	The value written to this field is added to the semaphore count in an atomic way such that simultaneous software adds and DMA hardware substracts happening on the same clock are protected. This bit field reads back a value of 0x00. Writing a value of 0x02 increments the semaphore count by two, unless the DMA channel decrements the count on the same clock, then the count is incremented by a net one.

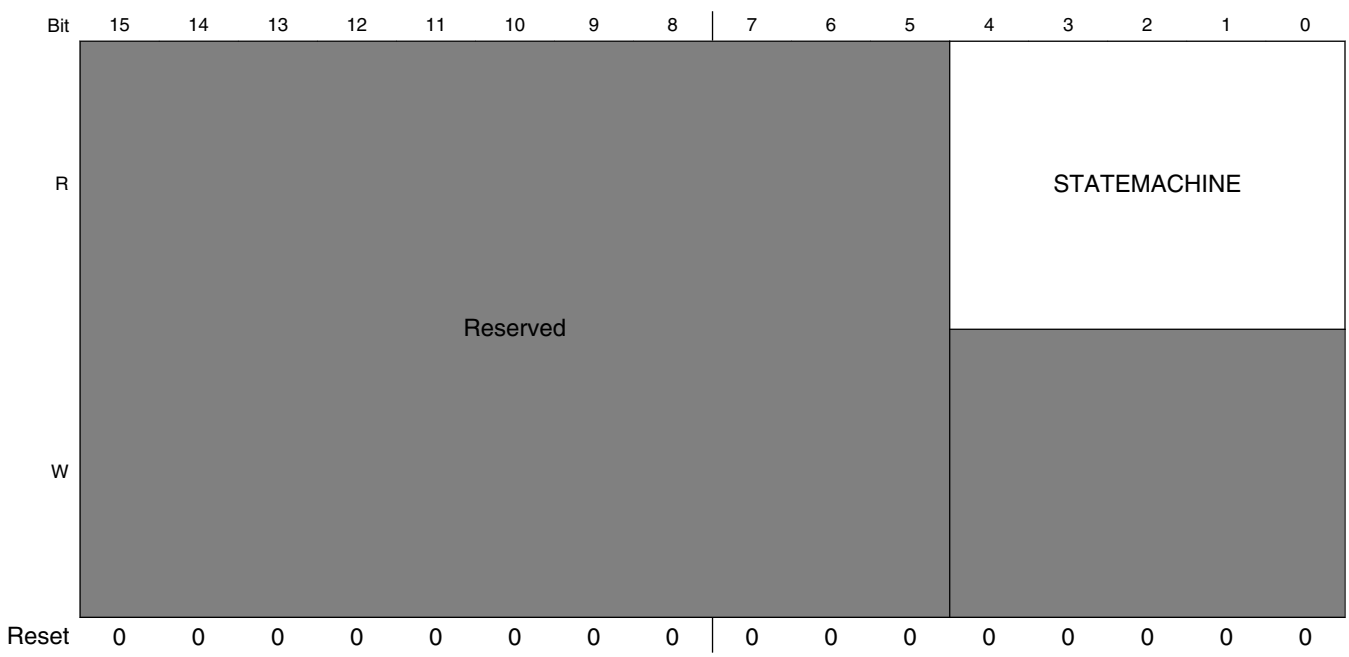
14.5.13 AHB to APBH DMA Channel n Debug Information (APBH_CH*n*_DEBUG1)

This register gives debug visibility into the APBH DMA Channel n state machine and controls.

This register allows debug visibility of the APBH DMA Channel n.

Address: 11_0000h base + 150h offset + (112d × i), where i=0d to 15d





APBH_CH*n*_DEBUG1 field descriptions

Field	Description
31 REQ	This bit reflects the current state of the DMA Request Signal from the APB device
30 BURST	This bit reflects the current state of the DMA Burst Signal from the APB device
29 KICK	This bit reflects the current state of the DMA Kick Signal sent to the APB Device
28 END	This bit reflects the current state of the DMA End Command Signal sent from the APB Device
27 SENSE	This field is reserved. This bit is reserved for this DMA Channel and always reads 0.
26 READY	This bit is reserved for this DMA Channel and always reads 0.
25 LOCK	This field is reserved. This bit is reserved for this Channel and always reads 0.
24 NEXTCMDADDRVALID	This bit reflects the internal bit which indicates whether the channel's next command address is valid.
23 RD_FIFO_EMPTY	This bit reflects the current state of the DMA Channel's Read FIFO Empty signal.
22 RD_FIFO_FULL	This bit reflects the current state of the DMA Channel's Read FIFO Full signal.
21 WR_FIFO_EMPTY	This bit reflects the current state of the DMA Channel's Write FIFO Empty signal.
20 WR_FIFO_FULL	This bit reflects the current state of the DMA Channel's Write FIFO Full signal.
19–5 RSVD1	This field is reserved. Reserved
STATEMACHINE	PIO Display of the DMA Channel n state machine state. 0x00 **IDLE** — This is the idle state of the DMA state machine. 0x01 **REQ_CMD1** — State in which the DMA is waiting to receive the first word of a command. 0x02 **REQ_CMD3** — State in which the DMA is waiting to receive the third word of a command. 0x03 **REQ_CMD2** — State in which the DMA is waiting to receive the second word of a command. 0x04 **XFER_DECODE** — The state machine processes the descriptor command field in this state and branches accordingly. 0x05 **REQ_WAIT** — The state machine waits in this state for the PIO APB cycles to complete. 0x06 **REQ_CMD4** — State in which the DMA is waiting to receive the fourth word of a command, or waiting to receive the PIO words when PIO count is greater than 1. 0x07 **PIO_REQ** — This state determines whether another PIO cycle needs to occur before starting DMA transfers. 0x08 **READ_FLUSH** — During a read transfers, the state machine enters this state waiting for the last bytes to be pushed out on the APB. 0x09 **READ_WAIT** — When an AHB read request occurs, the state machine waits in this state for the AHB transfer to complete. 0x0C **WRITE** — During DMA Write transfers, the state machine waits in this state until the AHB master arbiter accepts the request from this channel. 0x0D **READ_REQ** — During DMA Read transfers, the state machine waits in this state until the AHB master arbiter accepts the request from this channel.

Table continues on the next page...

APBH_CH*n*_DEBUG1 field descriptions (continued)

Field	Description
	0x0E **CHECK_CHAIN** — Upon completion of the DMA transfers, this state checks the value of the Chain bit and branches accordingly. 0x0F **XFER_COMPLETE** — The state machine goes to this state after the DMA transfers are complete, and determines what step to take next. 0x14 **TERMINATE** — When a terminate signal is set, the state machine enters this state until the current AHB transfer is completed. 0x15 **WAIT_END** — When the Wait for Command End bit is set, the state machine enters this state until the DMA device indicates that the command is complete. 0x1C **WRITE_WAIT** — During DMA Write transfers, the state machine waits in this state until the AHB master completes the write to the AHB memory space. 0x1D **HALT_AFTER_TERM** — If HALTONTERMINATE is set and a terminate signal is set, the state machine enters this state and effectively halts. A channel reset is required to exit this state 0x1E **CHECK_WAIT** — If the Chain bit is a 0, the state machine enters this state and effectively halts. 0x1F **WAIT_READY** — When the NAND Wait for Ready bit is set, the state machine enters this state until the GPMI device indicates that the external device is ready.

14.5.14 AHB to APBH DMA Channel n Debug Information (APBH_CH*n*_DEBUG2)

This register gives debug visibility for the APB and AHB byte counts for DMA Channel n.

This register allows debug visibility of the APBH DMA Channel n.

Address: 11_0000h base + 160h offset + (112d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	APB_BYTES																AHB_BYTES															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

APBH_CH*n*_DEBUG2 field descriptions

Field	Description
31–16 APB_BYTES	This value reflects the current number of APB bytes remaining to be transfered in the current transfer.
AHB_BYTES	This value reflects the current number of AHB bytes remaining to be transfered in the current transfer.

14.5.15 APBH Bridge Version Register (APBH_VERSION)

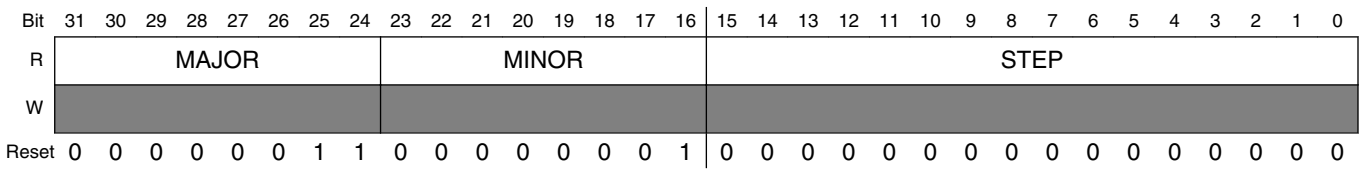
This register always returns a known read value for debug purposes it indicates the version of the block.

This register indicates the RTL version in use.

EXAMPLE

```
if (APBH_VERSION.B.MAJOR != 3)
   Error();
```

Address: 11_0000h base + 800h offset = 11_0800h



APBH_VERSION field descriptions

Field	Description
31–24 MAJOR	Fixed read-only value reflecting the MAJOR field of the RTL version.
23–16 MINOR	Fixed read-only value reflecting the MINOR field of the RTL version.
STEP	Fixed read-only value reflecting the stepping of the RTL version.

Chapter 15
Asynchronous Sample Rate Converter (ASRC)

15.1 Overview

The Asynchronous Sample Rate Converter (ASRC) converts the sampling rate of a signal associated with an input clock into a signal associated with a different output clock.

The ASRC supports concurrent sample rate conversion of up to 10 channels of about -120dB THD+N. The ASRC supports up to three sampling rate pairs.

The incoming audio data to this chip may be received from various sources at different sampling rates. The outgoing audio data of this chip may have different sampling rates and it can also be associated with output clocks that are asynchronous to the input clocks.

The ASRC is implemented as a co-processor in hardware, with minimal ARM Platform intervention required.

Figure 15-1 is a system view of the connection between the ASRC block and other blocks.

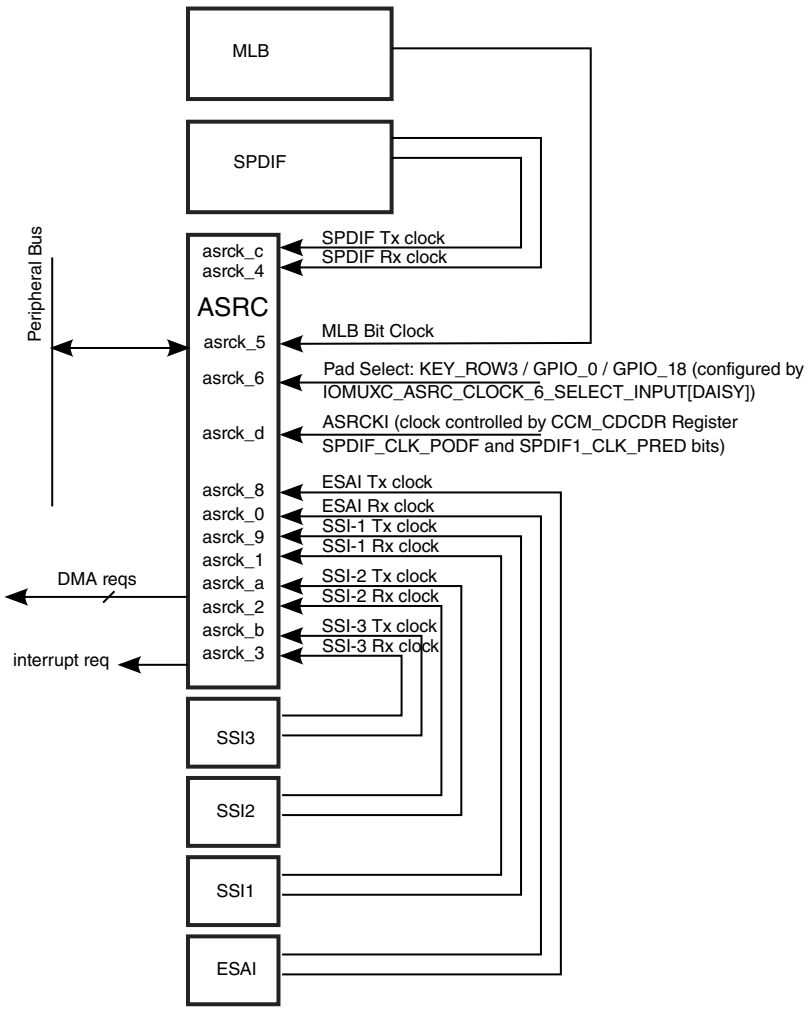


Figure 15-1. General System Overview

Figure 15-2 is the ASRC block diagram.

The red dotted line designates the ASRC block. Objects outside the dotted line represent SoC-level resources.

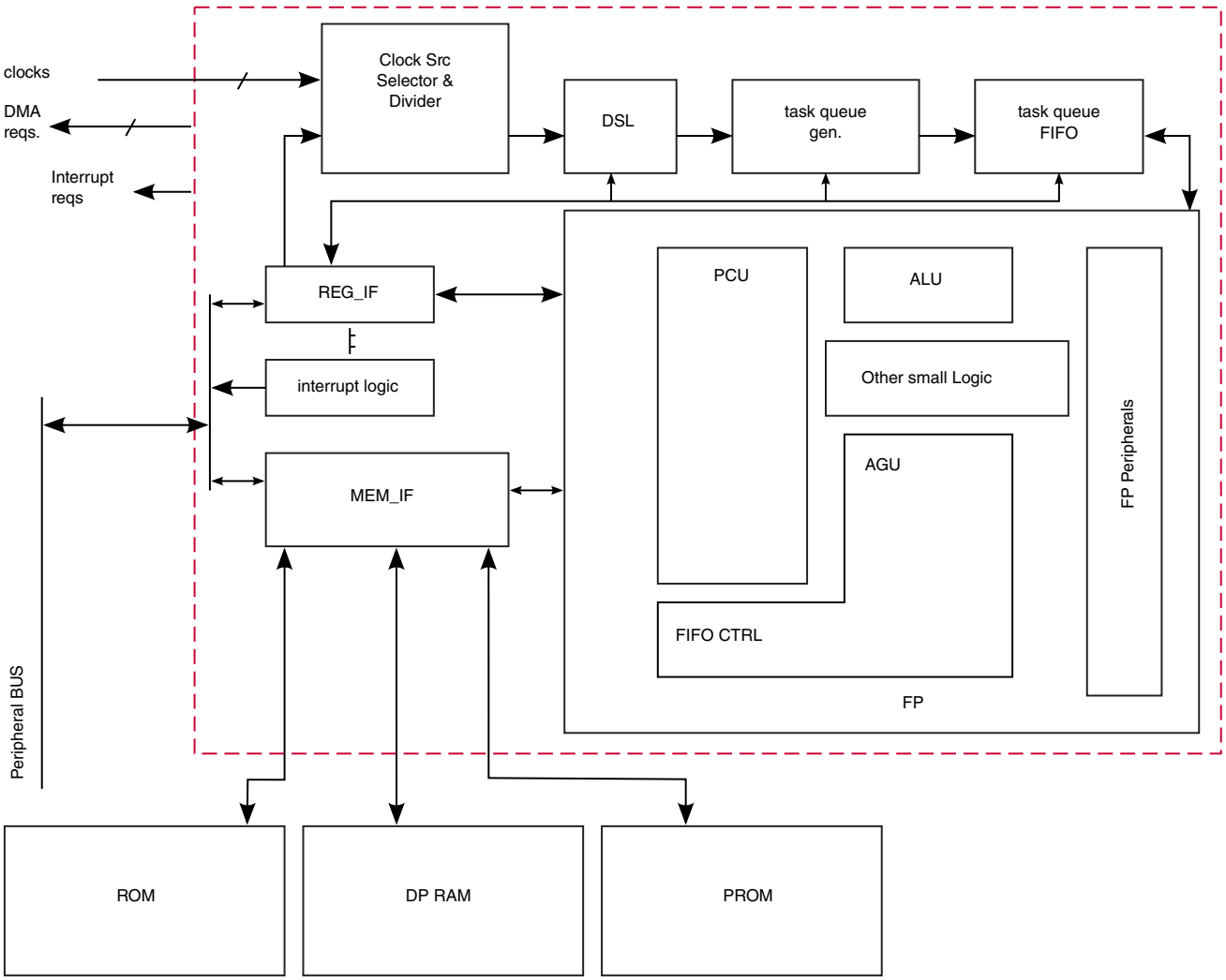


Figure 15-2. ASRC block diagram

15.1.1 Features

Table 15-1. ASRC Specifications

Parameters	Test Conditions	Minimum	Typical	Maximum	Unit
Channels Supported		0 [1]		10	
Pairs of Rate Conversion		1	-	3	
THD+N	120MHz < Fs_{ASRC}[2] <160MHz		-120		dB
Dynamic Range				144	dB
Settling Time			40		ms
Comment:					

1. When a pair has zero channels, the pair will be disabled, although the pair enable bit may be set in ASRCTR register.
2. Fs_{ASRC} is the processing clock of ASRC block.

Other Features:

- Any number (0-10) of contiguous channels can be associated to one of the sampling rate pairs.
- Support user-programmable threshold for the input/output FIFOs.
- Support flexible 8/16/24 bit width of input data, and 16/24 bit width of output data.
- Designed for rate conversion between 44.1kHz, 32kHz, 48kHz, 96kHz, and 192kHz. The useful signal bandwidth is below 24kHz.
- Other input sampling rates in the range of 8kHz to 200kHz is also supported, but possibly with less desirable bandwidth.
- Other output sampling rates in the range of 30kHz to 200kHz is also supported, but possibly with less desirable bandwidth.
- Limited support for the case when output sampling rates is between 8kHz and 30kHz. The limitation is the supported ratio (Fsin/Fsout) range as between 1/24 to 8
- Automatic accommodation to slow variations in the incoming and outgoing sampling rates.
- Linear phase
- Tolerant to sample clock jitter
- Designed for real-time streaming audio usage. The output sampling clock must be always physically available in the system.

Clock/Data Connections

- The sampling rate clocks are directly connected to the ASRC block, the ratio estimation of the input clocks with output clocks are done in ASRC hardware.
- The clock signals come from the following blocks, for example:
  - ESAI, receiving bit clock and transmitting bit clock
  - SPDIF, receiving bit clock and transmitting bit clock
  - other audio peripherals etc.
- The exchange of audio data is done by the processor accessing ASRC block through registers defined on shared peripheral bus.

15.1.2 Modes of Operation

See ASRC Memory Map/Register Definition for a definition of the registers and parameters used in ASRC.

15.1.2.1 Data Transfer Schemes

15.1.2.1.1 Data Input Modes

The input mode for each of the three channel sets may be set independently. Three modes of supplying data to the ASRC input FIFOs are available:

- Polling
- Interrupt
- DMA

In all input-data transfer schemes, the ASRC fetches data from each enabled FIFO and processes the data sample-by-sample after each rising edge of the associated input sampling clock until the FIFO level reaches a threshold.

After the threshold is reached, the ASRC requests data. The FIFO size for each channel set is 64 samples and the threshold is set at 32 samples.The threshold can be defined by interface registers ASRMCRx, x=A, B or C.

If the ASRC attempts to fetch data from an empty FIFO, an error is generated and the ASRSTR_AOLE bit is set. If the ASRC overload interrupt is enabled (ASRIER_AOLIE bit is set), an interrupt is generated.

When writing data to an input FIFO, you must ensure that it is in a predefined sequence. For example, when writing to an input FIFO, the sequence should be: channel_0, channel_1, channel_2,..., channel_n, channel_0, channel_1, channel_2, etc. Here channel_n stands for the data intended for the n-th channel. The hardware will re-allocate each data to its corresponding channel FIFO. The channel being re-allocated is shown by ASRCCR_ACIx, x=A,B or C.

Mode 1 (Polling Mode)

Polling mode is the default mode following power-on or individual reset, and is selected by clearing the associated channel set A, B, or C data-input interrupt enable bit (ASRIER_ADIEx, where x=A, B or C). In this mode, data-input interrupts are disabled. When the FIFO level is below the threshold, the associated status bit (ASRSTR_AIDEx, where x=A, B, or C) is set. To clear the status bit, the FIFO must be written with enough data to raise the level above the threshold.

Mode 2 (Interrupt Mode)

The ASRC input FIFOs can also be serviced by interrupts. To enable interrupts, the corresponding data-input interrupt enable bits (ASRIER_ADIEx, where x=A, B, or C) should be set. An interrupt is automatically generated any time the input FIFO level is below the threshold. The interrupt is cleared when enough data is written to the FIFO to raise the level above the threshold.

Mode 3 (DMA Mode)

The ASRC input FIFOs can also be filled using DMA. In this mode, the data-input interrupt-enable bits (ASRIER_ADIEx, where x=A, B, or C) should be cleared and the DMA controller should be configured to use the ASRC as a request source.

15.1.2.1.2 Data Output Modes

The output mode for each of the 3 channel sets (A, B, and C) may be set independently.

Three modes of retrieving data from the ASRC output FIFOs are available:
- Polling
- Interrupt
- DMA

In all output-data transfer schemes, the ASRC places a processed sample into the associated output FIFO. After a threshold is reached, the ASRC requests that data be transferred out of the FIFO.

The FIFO size for each channel set is 64 samples and the threshold is set at 32 samples. The threshold can be defined by interface registers ASRMCRx, x=A, B or C.

If the ASRC attempts to place data into a FIFO that is already full, an error is generated and the ASRSTR_AOLE bit is set. If the ASRC overload interrupt is enabled (ASRIER_AOLIE bit is set), an interrupt is generated.

Each output FIFO is organized in the same channel order in which the associated input FIFO was written.

Three transfer modes are supported by Interface Block.

Mode 1 (Polling Mode)

The ASRC output FIFOs can be serviced by polling. In this mode, ensure the associated output-data interrupt enable bit (ASRIER_ADOEx, where x=A, B, or C) is cleared. In this mode, all output-data interrupts are disabled. Any time the output FIFO exceeds the threshold the associated status bit (ASRSTR_AODFx, where x=A, B, or C) is set. To clear the status bit, enough data must be read from the associated output FIFO to lower the level below the threshold.

Mode 2 (Interrupt Mode)

The ASRC output FIFOs may also be serviced using interrupts. To enable this mode, the corresponding output-data interrupt-enable bits (ASRIER_ADOEx, where x=A, B, or C) should be set. Any time the output FIFO level exceeds the threshold, an interrupt is automatically generated. The interrupt is cleared when enough data is read from the FIFO to lower the level below the threshold.

Mode 3 (DMA Mode)

The ASRC output FIFOs can also be read using DMA. In this mode, the output-data interrupt-enable bits (ASRIER_ADOEx, where x=A, B, or C) should be cleared and the DMA controller should be configured to use the ASRC as a request source.

15.1.2.2 Word Alignment Supported

15.1.2.2.1 Input Data Alignment Modes

The position and length of input data word to the input data FIFOs ASRDIA, ASRDIB, ASRDIC are programmable. The control bits are defined in ASRMCR1x {x=A, B, or C}. It supports the following modes.

Table 15-2. Input Data Alignment

Format	Bit Number																															
	31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
8-bit LSB Aligned																									7	6	5	4	3	2	1	0
8-bit MSB Aligned																	7	6	5	4	3	2	1	0								
16-bit LSB Aligned																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
16-bit MSB Aligned	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
24-bit LSB Aligned									23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
24-bit MSB Aligned	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								

15.1.2.2.2 Output Data Alignment Modes

The position and length of output data word from the output data FIFOs ASRDOA, ASRDOB, ASRDOC are programmable. The control bits are defined in ASRMCR1x {x=A, B, or C}. It supports the following modes.

Table 15-3. Output Data Alignment

Format	Bit Number																															
	31	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
16-bit LSB Aligned																	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Table continues on the next page...

Table 15-3. Output Data Alignment (continued)

Format	Bit Number																															
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
16-bit LSB Aligned with Sign Extension	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
16-bit MSB Aligned	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
24-bit LSB Aligned									23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
24-bit LSB Aligned with Sign Extension	23	23	23	23	23	23	23	23	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
24-bit MSB Aligned	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0								

15.2 Clocks

The table found here describes the clock sources for ASRC.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 15-4. ASRC Clocks

Clock name	Clock Root	Description
asrck_clock_d	spdif1_clk_root	ASRC module clock (SPDIF clock)
ipg_clk	ahb_clk_root	Peripheral clock
mem_clk	ahb_clk_root	Peripheral access clock

15.3 Interrupts

ASRC has several interrupts events.

The priorities are shown as Table 15-5 .

Table 15-5. Interrupt Priorities/Vector

Priority	Description
lowest	ASRC Pair A input data needed

Table continues on the next page...

Table 15-5. Interrupt Priorities/Vector (continued)

Priority	Description
	ASRC Pair B input data needed
	ASRC Pair C input data needed
	ASRC Pair A output data ready
	ASRC Pair B output data ready
	ASRC Pair C output data ready
	ASRC Overload

15.4 DMA requests

ASRC has six DMA requests. They are directly connected to the lowest six status bits in the ASRSTR register.

Table 15-6. DMA requests

Type	Description
0	ASRC Pair A input data needed
1	ASRC Pair B input data needed
2	ASRC Pair C input data needed
3	ASRC Pair A output data ready
4	ASRC Pair B output data ready
5	ASRC Pair C output data ready

15.5 Functional Description

This section provides a complete functional description of the block.

15.5.1 Algorithm Description

15.5.1.1 Signal processing flow

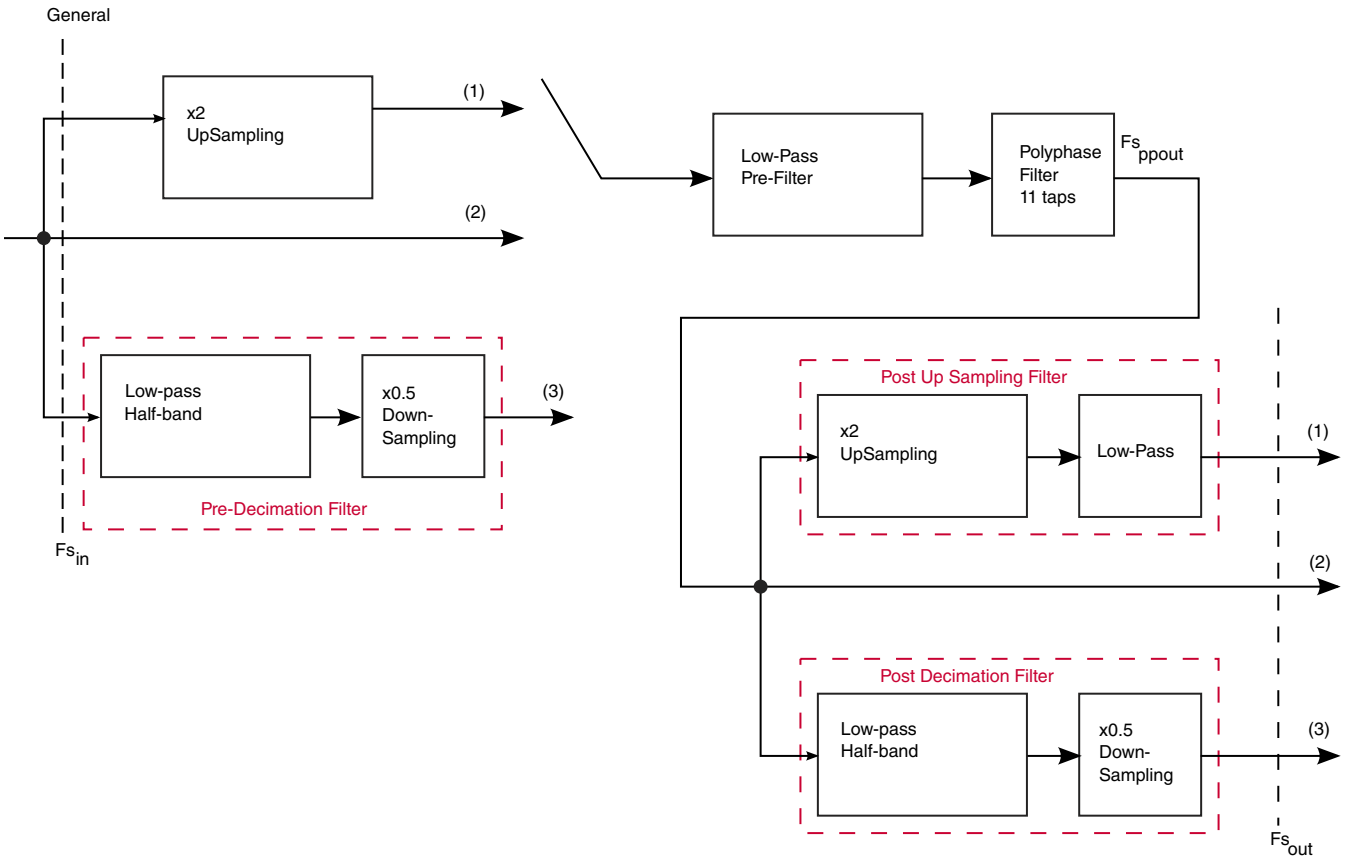


Figure 15-3. Signal processing configurations

The figure above shows the possible configurations of the ASRC. Each configuration consists of 2 to 4 stages.

- x2 up-sampling rate expander (zero insertion only) (input branch 1), direct connection (input branch 2), or low-pass pre decimation filter (consisting of a low-pass half-band FIR filter with x0.5 downsampling rate decimator) (input branch 3),
- low-pass pre-filter, the low-pass bandwidth is at most 0.25 x Fs, where Fs is the sampling rate of the input signal to this low-pass pre-filter,
- polyphase filter,
- x2 post upsampling filter (consisting of a x2 up-sampling rate expander (zero insertion only) with low-pass half-band FIR filter) (output branch 1), direct connection (output branch 2), or low-pass post decimation filter (consisting of a low-pass half-band FIR filter with x0.5 downsampling rate decimator) (output branch 3).

By flowing through different processing branches and different setups of the pre-filter, this ASRC scheme can be used to handle different rate conversion requirements.

- Configuration (a): Input Branch 1+Output Branch 1:

The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/2$.
The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}/2$.

- Configuration (b): Input Branch 1+Output Branch 2:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/2$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}$.

- Configuration (c): Input Branch 1+Output Branch 3:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/2$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = 2Fs_{out}$.

- Configuration (d): Input Branch 2+Output Branch 1:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/4$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}/2$.

- Configuration (e): Input Branch 2+Output Branch 2:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/4$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}$.

- Configuration (f): Input Branch 2+Output Branch 3:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/4$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = 2Fs_{out}$.

- Configuration (g): Input Branch 3+Output Branch 1:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/8$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}/2$.

- Configuration (h): Input Branch 3+Output Branch 2:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/8$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = Fs_{out}$.

- Configuration (i): Input Branch 3+Output Branch 3:

  The signal bandwidth observed before the polyphase filter is at most $BW_{in} = Fs_{in}/8$.
  The signal sampling rate of the polyphase filter output is $Fs_{ppout} = 2Fs_{out}$.

Table 15-7. Pre-processing, post-processing options

{Pre_Proc, Post_Proc}		Fsout (KHz)								
		8	32	44.1	48	64	88.2	96	128	192
Fsin (KHz)	8	{0,1}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}
	12	{0,2}	{0,1}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}
	16	{1,2}	{0,1}	{0,1}	{0,1}	{0,0}	{0,0}	{0,0}	{0,0}	{0,0}
	24	{1,2}	{0,1}	{0,1}	{0,1}	{0,1}	{0,0}	{0,0}	{0,0}	{0,0}
	32	{1,2}	{0,1}	{0,1}	{0,1}	{0,1}	{0,1}	{0,0}	{0,0}	{0,0}
	44.1	{2,2}	{0,2}	{0,1}	{0,1}	{0,1}	{0,1}	{0,1}	{0,0}	{0,0}
	48	{2,2}	{0,2}	{0,2}	{0,1}	{0,1}	{0,1}	{0,1}	{0,1}	{0,0}
	64	{2,2}	{0,2}	{0,2}	{0,2}	{0,1}	{0,1}	{0,1}	{0,1}	{0,0}
	88.2	NA	{1,2}	{1,2}	{1,2}	{1,1}	{1,1}	{1,1}	{1,1}	{1,1}
	96	NA	{1,2}	{1,2}	{1,2}	{1,1}	{1,1}	{1,1}	{1,1}	{1,1}
	128	NA	{1,2}	{1,2}	{1,2}	{1,1}	{1,1}	{1,1}	{1,1}	{1,1}
	192	NA	{2,2}	{2,2}	{2,2}	{2,1}	{2,1}	{2,1}	{2,1}	{2,1}

NOTE: In the {Pre_Proc, Post_Proc} pair, the meaning of the values are:

Pre_Proc:

- 0 --- Pre-processing Branch 1 as shown in Figure 15-3
- 1 --- Pre-processing Branch 2 as shown in Figure 15-3
- 2 --- Pre-processing Branch 3 as shown in Figure 15-3, decimation-by-2

Post_Proc:

- 0 --- Post-processing Branch 1 as shown in Figure 15-3
- 1 --- Post-processing Branch 2 as shown in Figure 15-3
- 2 --- Post-processing Branch 3 as shown in Figure 15-3

The latencies of the different option can be roughly calculated as follows:

- For PreProc = 0, PostProc = 1 : min latency = constant_A / input-sample-rate + constant_B / output-sample-rate
- For PreProc = 0, PostProc = 0 : min latency = constant_A / input-sample-rate + constant_C / output-sample-rate
- For PreProc = 1, PostProc = 1 : min latency = constant_D / input-sample-rate + constant_B / output-sample-rate

The constants above (e.g., constant_A means the Constant for Preproc = 0, constant_B means the Constant for Postproc = 1, ...) are only influenced by the PreProc/PostProc and (input/output) sampling rate to which they are connected. Input latencies have no relationship with the output latencies, but both elements add together to form the total latencies.

For a rough estimation, the constants can be set as:

- Constant for Preproc = 0: 39

- Constant for Preproc = 1: 78.5
- Constant for Preproc = 2: 235
- Constant for Postproc = 0: 42.5
- Constant for Postproc = 1: 8.5
- Constant for Postproc = 2: 172

The max latency can be derived from this value by using the following formula (where 32 means the input/output FIFO depth that will arouse data transfer):

- max latency = min latency + 32 / input-sample-rate + 32 / output-sample-rate

15.5.1.2 Operation of the Filter

15.5.1.2.1 Support of Physical Clocks

This design supports physical sampling clocks. The clocks can be provided by Sony/ Phillips digital interface (SPDIF), enhanced serial audio interface (ESAI), standard serial interface (SSI-1, SSI-2, SSI-3), media local bus controller (MLB) , Core master clock derivative as ASRCK1 .

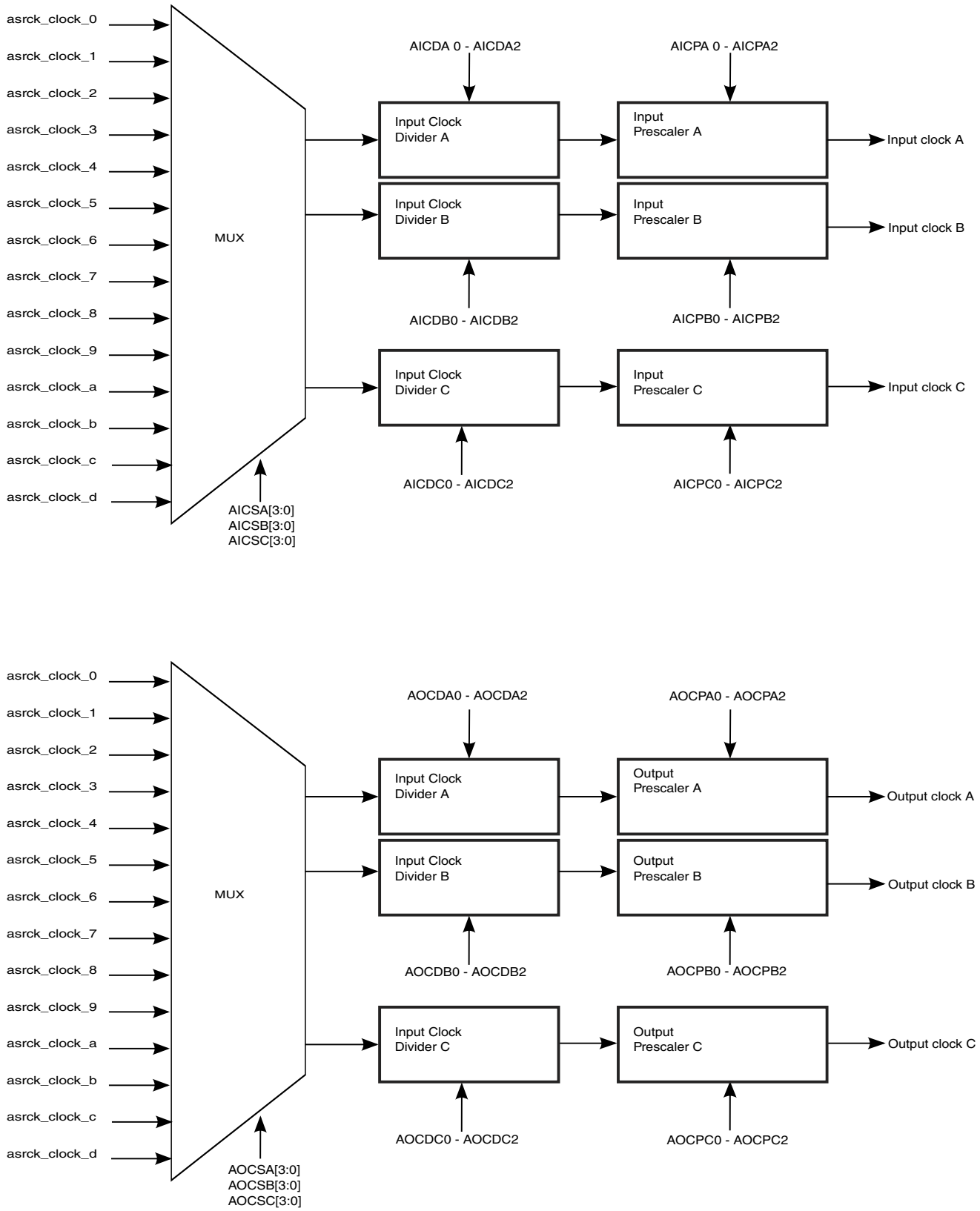


Figure 15-4. Clock Source Selector & Divider

Software can set the ASRC Clock Source Register (ASRCSR) and the Clock Divider Register to select the desired clock source and divide it to the needed sample rate clock for use by the ASRC. The clocks have the following restriction. If the prescaler is set to 1, the clock divider can only be set to 1 and the clock source must have a 50% duty cycle.

15.6 Startup Procedure

The following example shows the normal setup procedure for the ASRC block.

```
#include "asrc_common.h"

#include "stdio.h"

#include "soc_api.h"

int incnt=0;

int outcnt=0;

#include "wy_ideal_ratio_dataini_part.h"

WORD   IdealRatio_High=0x04; //

WORD   IdealRatio_Low=0x0; //

void asrc_config_alloc(WORD ASRCTR_VAL, WORD ASRIER_VAL, WORD ASRCNCR_VAL,
                           WORD ASRCFG_VAL, WORD ASRCDR1_VAL, WORD ASRCDR2_VAL,
                           WORD ASRCSR_VAL)

{  // Disable ASRC

    reg32_write(ASRC_ASRCTR,0x0);

    reg32_write(ASRC_ASRCTR,ASRCTR_VAL);

    reg32_write(ASRC_ASRIER,ASRIER_VAL);

    reg32_write(ASRC_ASRIEM,0x0);

    reg32_write(ASRC_ASRCNCR,ASRCNCR_VAL);

    reg32_write(ASRC_ASRCFG,ASRCFG_VAL);

    reg32_write(ASRC_ASRCDR1,ASRCDR1_VAL);

    reg32_write(ASRC_ASRCDR2,ASRCDR2_VAL);

    reg32_write(ASRC_ASRCSR, ASRCSR_VAL);

    reg32_write(ASRC_ASRPM1, 0x7fffff);

    reg32_write(ASRC_ASRPM2, 0x255555);

    reg32_write(ASRC_ASRPM3, 0xff7280);

    reg32_write(ASRC_ASRPM4, 0xff7280);

    reg32_write(ASRC_ASRPM5, 0xff7280);
```

```
    reg32_write(ASRC_ASRQFIFO1,0x001f00);
    reg32_write(ASRC_ASRMCRA,0x001f00);
    reg32_write(ASRC_ASRMCRB,0x001f00);
    reg32_write(ASRC_ASRMCRC,0x001f00);
}
 void sim_ideal_ratio()
  {
    WORD tmp32bit;
    #define ASRSTR_AIDEA_MASK  0x1
    #define ASRSTR_AODFA_MASK  0x1 <<3
    #define ASRSTR_AOLE_MASK   0x1<<6
    #define ASRCTR_DBG_EN             1<<23
    #define ASRCTR_IDRA               1<<13
    #define ASRCTR_USRA               1<<14
    #define ASRC_CLK_PRED_RSTRICTED 0<<28
    #define ASRC_CLK_PRED_DFLT      1<<28 // default: 596MHz div by 2
    #define ASRC_CLK_PRED_DIV3      2<<28 // 596MHz div by 3
    #define ASRC_CLK_PRED_DIV4      3<<28 // 596MHz div by 4
    #define ASRC_CLK_PRED_DIV5      4<<28 // 596MHz div by 5
    #define ASRC_CLK_PRED_DIV6      5<<28 // 596MHz div by 6
    #define ASRC_CLK_PRED_DIV7      6<<28 // 596MHz div by 7
    #define ASRC_CLK_PRED_DIV8      7<<28 // 596MHz div by 8
    #define ECSPI_CLK_PRED_DFLT     1<<25
    #define ECSPI_CLK_PODF_DFLT     1<<19
    #define ASRC_CLK_PODF_DIV1      0<<9  // pred output divide by 1 again
    #define ASRC_CLK_PODF_DIV2      1<<9  // pred output divide by 2 again
    #define ASRC_CLK_PODF_DIV3      2<<9  // pred output divide by 3 again
    #define ASRC_CLK_PODF_DIV4      3<<9  // pred output divide by 4 again
    #define ASRC_CLK_PODF_DFLT      4<<9  // default: pred output divide by 5 again
    #define ASRC_CLK_PODF_DIV6      5<<9  // pred output divide by 6 again
    #define ASRC_CLK_PODF_DIV7      6<<9  // pred output divide by 7 again
    #define ASRC_CLK_PODF_DIV25     24<<9  // pred output divide by 7 again
    #define IEEE_CLK_PRED_DFLT      1<<6  //
```

```
     #define IEEE_CLK_PODF_DFLT      4     //

     #define ASR_HFA_HFB         0

     #define ASR_PREMODA_UP2         0<<6

     #define ASR_PREMODA_DIR         1<<6

     #define ASR_PREMODA_DN2         2<<6

     #define ASR_PREMODA_PAS         3<<6

     #define ASR_POSTMODA_UP2         0<<8

     #define ASR_POSTMODA_DIR         1<<8

     #define ASR_POSTMODA_DN2         2<<8

     // program CCM for ASRC core clocks

      reg32_write(CCM_CCGR7, 0xffffffff); // enable all perihperal clocks during all modes,
except stop mode

      reg32_write(CCM_CSCDR2, ASRC_CLK_PRED_DIV8|ECSPI_CLK_PRED_DFLT|ECSPI_CLK_PODF_DFLT|
ASRC_CLK_PODF_DIV25|IEEE_CLK_PRED_DFLT|IEEE_CLK_PODF_DFLT);

      // Disable the ASRC

      reg32_write(ASRC_ASRCTR, 0x0);

      // program AHB clocks

      tmp32bit = reg32_read(CCM_CBCDR);

      tmp32bit = tmp32bit & (~0x00001C00);

      //tmp32bit = tmp32bit | (0x00000C00); // AHB 100MHz // divided-by-4

      //tmp32bit = tmp32bit | (0x00001000); // AHB 80MHz // divided-by-5

      //tmp32bit = tmp32bit | (0x00001400); // AHB 66MHz // divided-by-6

      //tmp32bit = tmp32bit | (0x00001800); // AHB 57MHz // divided-by-7

      tmp32bit = tmp32bit | (0x00001C00); // AHB 50MHz // divided-by-8

      reg32_write(CCM_CBCDR, tmp32bit); // enable all perihperal clocks during all modes,
except stop mode

      while ( (reg32_read(CCM_CDHIPR) & 0x00008) != 0);

            asrc_config_alloc ( 0x002 | ASRCTR_IDRA | ASRCTR_USRA,    // ASRCTR_VAL, Use
Ratio input, use ideal ratio, Enable Pair A,

            0x0, //0x09,                                              // ASRIER_VAL, Open
PairA input and output interrupt

            0x002,     // ASRCNCR_VAL, assign 2 channels to Pair A

            ASR_PREMODA_DIR | ASR_POSTMODA_DIR | ASR_HFA_HFB,         // ASRCFG_VAL,
POSTMODA=downsampling by 2 ; PREMODA=downsampling by 2

            0x03b03b , // ASRCDR1_VAL, AOCPA=3(FoutA/(2^3)); AICPA=3(FinA/(2^3));
AOCDA=7(div 8); AICDA=7(div 8);

            0x0  ,     // ASRCDR2_VAL,
```

```
            0x00d00d   // ASRCSR_VAL,  AOCSA=d: bit clock d: ASRCK1 clk from CCM; AICSA=d:
bit clock d: ASRCK1 clock from CCM;

            );

      reg32_write(ASRC_ASRIDRHA, 0x04); //

      reg32_write(ASRC_ASRIDRLA, 0x0); // Ideal Ratio is set to be 1.

      #define OUTFIFO_THRESH_0  8<<12

      #define INFIFO_THRESH_1   32

      reg32_write(ASRC_ASRMCRA, OUTFIFO_THRESH_0 | INFIFO_THRESH_1);

      reg32clrbit(ASRC_ASRMCRA, 23);   // zeroize Pair A buffers

      reg32setbit(ASRC_ASRMCRA, 21);   // stall conversion in case of near full/near empty
condition

      reg32clrbit(ASRC_ASRMCRA, 20); // Do not bypass polyA filter

      // Set ASRC Interrupt

      //CAPTURE_INTERRUPT(ASRC_INT_ROUTINE, asrc_handler);

      //enable_hdler(ASRC_INT_NUM);

      disable_hdler(ASRC_INT_NUM);

      incnt=0;

      outcnt=0;

      reg32setbit(ASRC_ASRCTR,0); // enable ASRC

      #define ASRCFG_INIA_FINISH  0x1<<21

      while ( (reg32_read(ASRC_ASRCFG) & ASRCFG_INIA_FINISH) == 0); // wait for ini finished.

      // Polling

          while (outcnt < 100) <

      {

              int ii;

              if ( (reg32_read(ASRC_ASRSTR) & ASRSTR_AIDEA_MASK) != 0 )

               {

                 for (ii=0;ii<2;ii++)</codeblock

                   {

                      reg32_write(ASRC_ASRDIA,asrc_input_array[incnt]);     // feed in input
data

                      reg32_write(ASRC_ASRDIA,asrc_input_array[incnt]);     // feed in input
data

                      incnt=(incnt+1)%128;

                   }
```

```
                }
              if ( (reg32_read(ASRC_ASRSTR) & ASRSTR_AODFA_MASK) != 0 )
               {
                   for (ii=0;ii<2;ii++)<
                     {
                       WORD TempRdOut;
                       TempRdOut=reg32_read(ASRC_ASRDOA);  // get output data
                       TempRdOut=reg32_read(ASRC_ASRDOA);  // get output data
                       outcnt=outcnt+1;
                     }
                 }
              if ( (reg32_read(ASRC_ASRSTR) & ASRSTR_AOLE_MASK) != 0 )
               {
                   reg32_write(ASRC_ASRSTR,ASRSTR_AOLE_MASK);      // clear overloading
errors
                }
             }
         reg32clrbit(ASRC_ASRCTR,0); // disable ASRC
   }
```

15.7 ASRC Memory Map/Register Definition

All useful registers are listed in the memory map below. The access of undefined registers will behave as normal registers.

All the interface registers are LSB aligned except the input FIFOs and the output FIFOs, and each register has only 24 effective bits.

The input FIFO and output FIFO word alignment can be defined using ASRMCR1{A,B,C} registers in 32-bit interface system.

ASRC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
203_4000	ASRC Control Register (ASRC_ASRCTR)	32	R/W	0000_0000h	15.7.1/650
203_4004	ASRC Interrupt Enable Register (ASRC_ASRIER)	32	R/W	0000_0000h	15.7.2/653
203_400C	ASRC Channel Number Configuration Register (ASRC_ASRCNCR)	32	R/W	0000_0000h	15.7.3/654
203_4010	ASRC Filter Configuration Status Register (ASRC_ASRCFG)	32	R/W	0000_0000h	15.7.4/656
203_4014	ASRC Clock Source Register (ASRC_ASRCSR)	32	R/W	0000_0000h	15.7.5/658
203_4018	ASRC Clock Divider Register 1 (ASRC_ASRCDR1)	32	R/W	0000_0000h	15.7.6/662
203_401C	ASRC Clock Divider Register 2 (ASRC_ASRCDR2)	32	R/W	0000_0000h	15.7.7/663
203_4020	ASRC Status Register (ASRC_ASRSTR)	32	R	0000_0000h	15.7.8/664
203_4040	ASRC Parameter Register n (ASRC_ASRPMn1)	32	R/W	0000_0000h	15.7.9/667
203_4044	ASRC Parameter Register n (ASRC_ASRPMn2)	32	R/W	0000_0000h	15.7.9/667
203_4048	ASRC Parameter Register n (ASRC_ASRPMn3)	32	R/W	0000_0000h	15.7.9/667
203_404C	ASRC Parameter Register n (ASRC_ASRPMn4)	32	R/W	0000_0000h	15.7.9/667
203_4050	ASRC Parameter Register n (ASRC_ASRPMn5)	32	R/W	0000_0000h	15.7.9/667
203_4054	ASRC ASRC Task Queue FIFO Register 1 (ASRC_ASRTFR1)	32	R/W	0000_0000h	15.7.10/ 668
203_405C	ASRC Channel Counter Register (ASRC_ASRCCR)	32	R/W	0000_0000h	15.7.11/ 669
203_4060	ASRC Data Input Register for Pair x (ASRC_ASRDIA)	32	W	0000_0000h	15.7.12/ 670
203_4064	ASRC Data Output Register for Pair x (ASRC_ASRDOA)	32	R	0000_0000h	15.7.13/ 670
203_4068	ASRC Data Input Register for Pair x (ASRC_ASRDIB)	32	W	0000_0000h	15.7.12/ 670
203_406C	ASRC Data Output Register for Pair x (ASRC_ASRDOB)	32	R	0000_0000h	15.7.13/ 670
203_4070	ASRC Data Input Register for Pair x (ASRC_ASRDIC)	32	W	0000_0000h	15.7.12/ 670
203_4074	ASRC Data Output Register for Pair x (ASRC_ASRDOC)	32	R	0000_0000h	15.7.13/ 670
203_4080	ASRC Ideal Ratio for Pair A-High Part (ASRC_ASRIDRHA)	32	R/W	0000_0000h	15.7.14/ 671
203_4084	ASRC Ideal Ratio for Pair A -Low Part (ASRC_ASRIDRLA)	32	R/W	0000_0000h	15.7.15/ 672
203_4088	ASRC Ideal Ratio for Pair B-High Part (ASRC_ASRIDRHB)	32	R/W	0000_0000h	15.7.16/ 672
203_408C	ASRC Ideal Ratio for Pair B-Low Part (ASRC_ASRIDRLB)	32	R/W	0000_0000h	15.7.17/ 673
203_4090	ASRC Ideal Ratio for Pair C-High Part (ASRC_ASRIDRHC)	32	R/W	0000_0000h	15.7.18/ 673
203_4094	ASRC Ideal Ratio for Pair C-Low Part (ASRC_ASRIDRLC)	32	R/W	0000_0000h	15.7.19/ 674

Table continues on the next page...

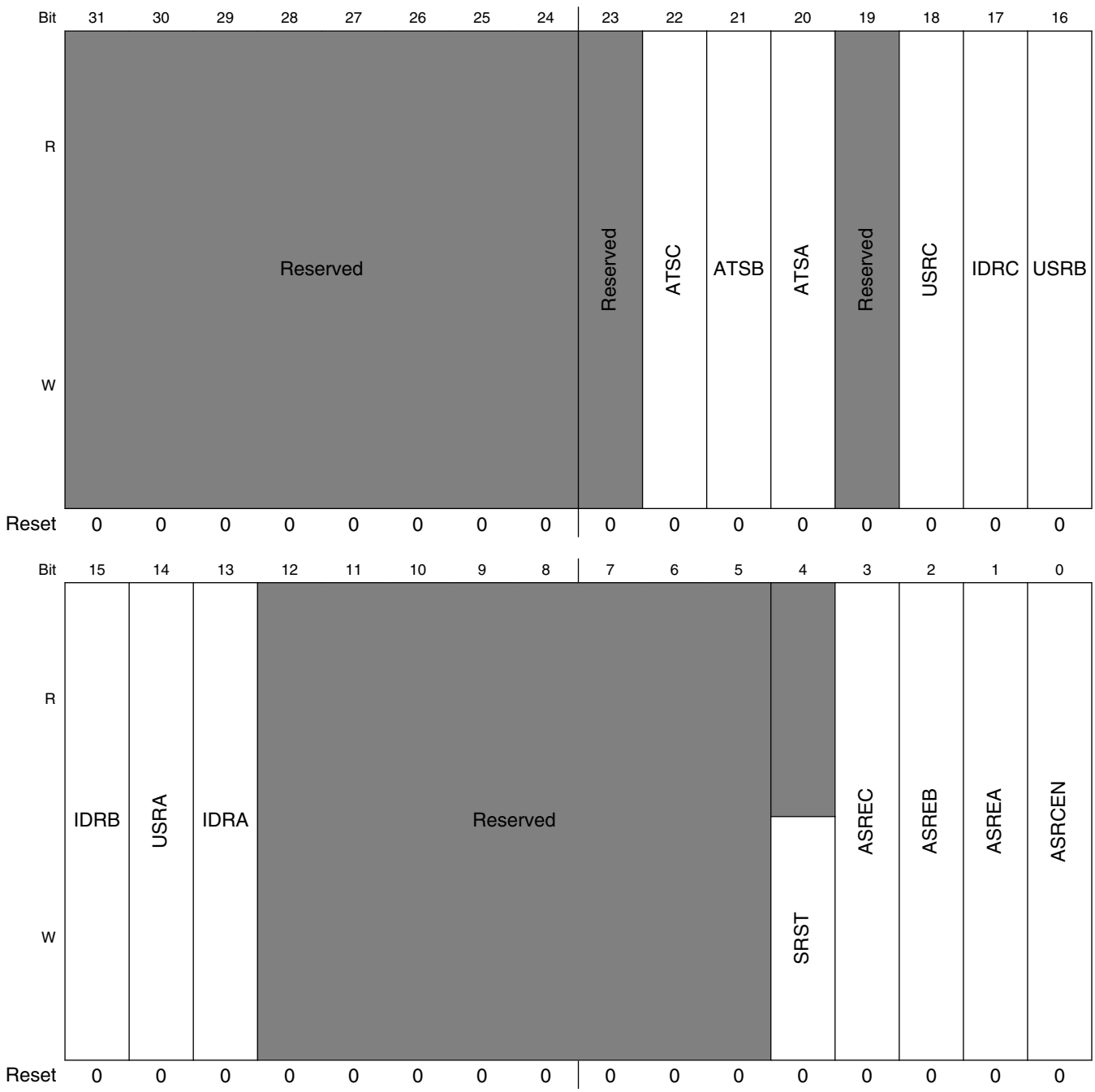
ASRC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
203_4098	ASRC 76kHz Period in terms of ASRC processing clock (ASRC_ASR76K)	32	R/W	0000_0A47h	15.7.20/ 675
203_409C	ASRC 56kHz Period in terms of ASRC processing clock (ASRC_ASR56K)	32	R/W	0000_0DF3h	15.7.21/ 676
203_40A0	ASRC Misc Control Register for Pair A (ASRC_ASRMCRA)	32	R/W	0000_0000h	15.7.22/ 677
203_40A4	ASRC FIFO Status Register for Pair A (ASRC_ASRFSTA)	32	R	0000_0000h	15.7.23/ 679
203_40A8	ASRC Misc Control Register for Pair B (ASRC_ASRMCRB)	32	R/W	0000_0000h	15.7.24/ 680
203_40AC	ASRC FIFO Status Register for Pair B (ASRC_ASRFSTB)	32	R	0000_0000h	15.7.25/ 682
203_40B0	ASRC Misc Control Register for Pair C (ASRC_ASRMCRC)	32	R/W	0000_0000h	15.7.26/ 683
203_40B4	ASRC FIFO Status Register for Pair C (ASRC_ASRFSTC)	32	R	0000_0000h	15.7.27/ 685
203_40C0	ASRC Misc Control Register 1 for Pair X (ASRC_ASRMCR1A)	32	R/W	0000_0000h	15.7.28/ 686
203_40C4	ASRC Misc Control Register 1 for Pair X (ASRC_ASRMCR1B)	32	R/W	0000_0000h	15.7.28/ 686
203_40C8	ASRC Misc Control Register 1 for Pair X (ASRC_ASRMCR1C)	32	R/W	0000_0000h	15.7.28/ 686

15.7.1 ASRC Control Register (ASRC_ASRCTR)

The ASRC control register (ASRCTR) is a 24-bit read/write register that controls the ASRC operations.

Address: 203_4000h base + 0h offset = 203_4000h



ASRC_ASRCTR field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 -	This field is reserved. Reserved. Should be written as zero for compatibility.
22 ATSC	ASRC Pair C Automatic Selection For Processing Options When this bit is 1, pair C will automatic update its pre-processing and post-processing options (ASRCFG: PREMODC, ASRCFG:POSTMODC see ASRC Misc Control Register 1 for Pair C) based on the frequencies it detected. To use this option, the two parameter registers(ASR76K and ASR56K) should be set correctly (see ASRC Misc Control Register 1 for Pair C and ASRC Misc Control Register 1 for Pair C). When this bit is 0, the user is responsible for choosing the proper processing options for pair C. This bit should be disabled when {USRC, IDRC}={1,1}.
21 ATSB	ASRC Pair B Automatic Selection For Processing Options When this bit is 1, pair B will automatic update its pre-processing and post-processing options (ASRCFG: PREMODB, ASRCFG:POSTMODB see ASRC Misc Control Register 1 for Pair C) based on the frequencies it detected. To use this option, the two parameter registers(ASR76K and ASR56K) should be set correctly (see ASRC Misc Control Register 1 for Pair C and ASRC Misc Control Register 1 for Pair C). When this bit is 0, the user is responsible for choosing the proper processing options for pair B. This bit should be disabled when {USRB, IDRB}={1,1}.
20 ATSA	ASRC Pair A Automatic Selection For Processing Options When this bit is 1, pair A will automatic update its pre-processing and post-processing options (ASRCFG: PREMODA, ASRCFG:POSTMODA see ASRC Misc Control Register 1 for Pair C) based on the frequencies it detected. To use this option, the two parameter registers(ASR76K and ASR56K) should be set correctly (see ASRC Misc Control Register 1 for Pair C and ASRC Misc Control Register 1 for Pair C). When this bit is 0, the user is responsible for choosing the proper processing options for pair A. This bit should be disabled when {USRA, IDRA}={1,1}.
19 -	This field is reserved. Reserved. Should be written as zero for compatibility.
18 USRC	Use Ratio for Pair C Use ratio as the input to ASRC. This bit is used in conjunction with IDRC control bit.
17 IDRC	Use Ideal Ratio for Pair C When USRC=0, this bit has no usage. When USRC=1 and IDRC=0, ASRC internal measured ratio will be used. When USRC=1 and IDRC=1, the idea ratio from the interface register ASRIDRHC, ASRIDRLC will be used. It is suggested to manually set ASRCFG:POSTMODC, ASRCFG:PREMODC according to Table 15-7 in this case.
16 USRB	Use Ratio for Pair B Use ratio as the input to ASRC. This bit is used in conjunction with IDRB control bit.
15 IDRB	Use Ideal Ratio for Pair B When USRB=0, this bit has no usage. When USRB=1 and IDRB=0, ASRC internal measured ratio will be used.

Table continues on the next page...

ASRC_ASRCTR field descriptions (continued)

Field	Description
	When USRB=1 and IDRB=1, the idea ratio from the interface register ASRIDRHB, ASRIDRLB will be used.It is suggested to manually set ASRCFG:POSTMODB, ASRCFG:PREMODB according to Table 15-7 in this case.
14 USRA	Use Ratio for Pair A Use ratio as the input to ASRC. This bit is used in conjunction with IDRA control bit.
13 IDRA	Use Ideal Ratio for Pair A When USRA=0, this bit has no usage. When USRA=1 and IDRA=0, ASRC internal measured ratio will be used. When USRA=1 and IDRA=1, the idea ratio from the interface register ASRIDRHA, ASRIDRLA will be used. It is suggested to manually set ASRCFG:POSTMODA, ASRCFG:PREMODA according to Table 15-7 in this case.
12–5 -	This field is reserved. Reserved. Should be written as zero for compatibility.
4 SRST	Software Reset This bit is self-clear bit. Once it is been written as 1, it will generate a software reset signal inside ASRC. After 9 cycles of the ASRC processing clock, this reset process will stop, and this bit will be cleared automatically.
3 ASREC	ASRC Enable C Enable the operation of the conversion C of ASRC. When ASREC is cleared, operation of conversion C is disabled.
2 ASREB	ASRC Enable B Enable the operation of the conversion B of ASRC. When ASREB is cleared, operation of conversion B is disabled.
1 ASREA	ASRC Enable A Enable the operation of the conversion A of ASRC. When ASREA is cleared, operation of conversion A is disabled.
0 ASRCEN	ASRC Enable Enable the operation of ASRC.

15.7.2 ASRC Interrupt Enable Register (ASRC_ASRIER)

Address: 203_4000h base + 4h offset = 203_4004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved								Reserved							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								AFPWE	AOLIE	ADOEC	ADOEB	ADOEA	ADIEC	ADIEB	ADIEA
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRIER field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–8 -	This field is reserved. Reserved. Should be written as zero for compatibility.
7 AFPWE	FP in Wait State Interrupt Enable Enables the FP in wait state interrupt. 1 interrupt enabled 0 interrupt disabled
6 AOLIE	Overload Interrupt Enable Enables the overload interrupt. 1 interrupt enabled 0 interrupt disabled
5 ADOEC	Data Output C Interrupt Enable Enables the data output C interrupt. 1 interrupt enabled 0 interrupt disabled
4 ADOEB	Data Output B Interrupt Enable Enables the data output B interrupt.

Table continues on the next page...

ASRC_ASRIER field descriptions (continued)

Field	Description
	1 interrupt enabled 0 interrupt disabled
3 ADOEA	Data Output A Interrupt Enable Enables the data output A interrupt. 1 interrupt enabled 0 interrupt disabled
2 ADIEC	Data Input C Interrupt Enable Enables the data input C interrupt. 1 interrupt enabled 0 interrupt disabled
1 ADIEB	Data Input B Interrupt Enable Enables the data input B interrupt. 1 interrupt enabled 0 interrupt disabled
0 ADIEA	Data Input A Interrupt Enable Enables the data input A Interrupt. 1 interrupt enabled 0 interrupt disabled

15.7.3 ASRC Channel Number Configuration Register (ASRC_ASRCNCR)

The ASRC channel number configuration register (ASRCNCR) is a 24-bit read/write register that sets the number of channels used by each ASRC conversion pair.

There are 10 channels available for distribution among 3 conversion pairs, they are ordered as 0,1,...,9. The bottom [0, ANCA-1] channels are used for pair A, the top [10-ANCC, 9] channels are used for pair C, and the [ANCA, ANCA+ANCB-1] channels are allocated for pair B. In case that ANCA=0, then the [0, ANCB-1] channels are assigned for pair B.

Address: 203_4000h base + Ch offset = 203_400Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved												ANCC				ANCB				ANCA			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRCNCR field descriptions

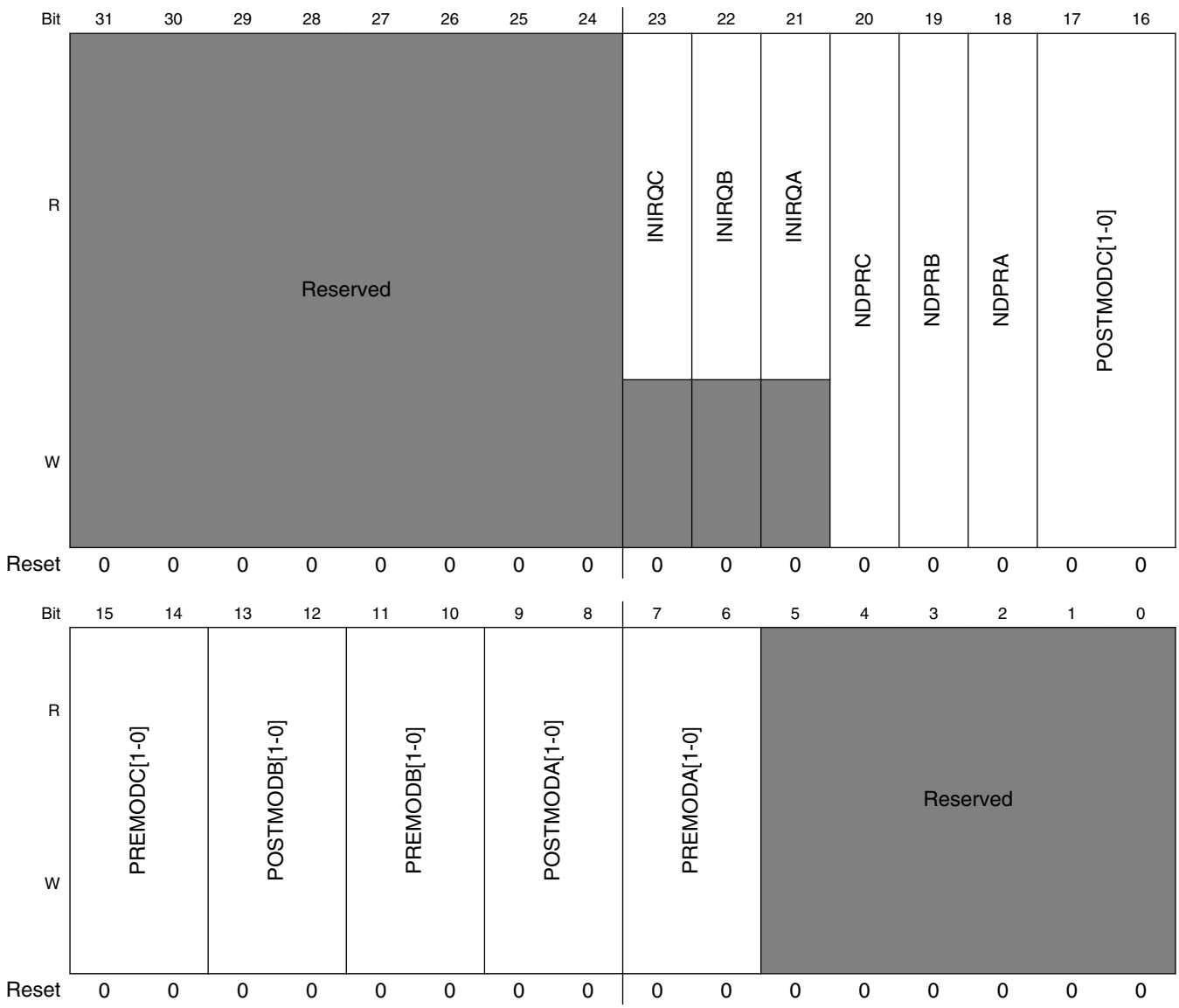
Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–12 -	This field is reserved. Reserved. Should be written as zero for compatibility.
11–8 ANCC	Number of C Channels[1] 0000 0 channels in C (Pair C is disabled) 0001 1 channel in C 0010 2 channels in C 0011 3 channels in C 0100 4 channels in C 0101 5 channels in C 0110 6 channels in C 0111 7 channels in C 1000 8 channels in C 1001 9 channels in C 1010 10 channels in C 1011-1111 Should not be used.
7–4 ANCB	Number of B Channels 0000 0 channels in B (Pair B is disabled) 0001 1 channel in B 0010 2 channels in B 0011 3 channels in B 0100 4 channels in B 0101 5 channels in B 0110 6 channels in B 0111 7 channels in B 1000 8 channels in B 1001 9 channels in B 1010 10 channels in B 1011-1111 Should not be used.
ANCA	Number of A Channels 0000 0 channels in A (Pair A is disabled) 0001 1 channel in A 0010 2 channels in A 0011 3 channels in A 0100 4 channels in A 0101 5 channels in A 0110 6 channels in A 0111 7 channels in A 1000 8 channels in A 1001 9 channels in A 1010 10 channels in A 1011-1111 Should not be used.

1. ANCC+ANCB+ANCA<=10. Hardware is not checking the constraint. Programmer should take the responsibility to ensure the constraint is satisfied.

15.7.4 ASRC Filter Configuration Status Register (ASRC_ASRCFG)

The ASRC configuration status register (ASRCFG) is a 24-bit read/write register that sets and/or automatically senses the ASRC operations.

Address: 203_4000h base + 10h offset = 203_4010h



ASRC_ASRCFG field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 INIRQC	Initialization for Conversion Pair C is served When this bit is 1, it means the initialization for conversion pair C is served. This bit is cleared by disabling the ASRC conversion pair (ASRCTR:ASREC=0 or ASRCTR:ASRCEN=0).
22 INIRQB	Initialization for Conversion Pair B is served When this bit is 1, it means the initialization for conversion pair B is served. This bit is cleared by disabling the ASRC conversion pair (ASRCTR:ASREB=0 or ASRCTR:ASRCEN=0).
21 INIRQA	Initialization for Conversion Pair A is served When this bit is 1, it means the initialization for conversion pair A is served. This bit is cleared by disabling the ASRC conversion pair (ASRCTR:ASREA=0 or ASRCTR:ASRCEN=0).
20 NDPRC	Not Use Default Parameters for RAM-stored Parameters For Conversion Pair C 0 Use default parameters for RAM-stored parameters. Override any parameters already in RAM. 1 Don't use default parameters for RAM-stored parameters. Use the parameters already stored in RAM.
19 NDPRB	Not Use Default Parameters for RAM-stored Parameters For Conversion Pair B 0 Use default parameters for RAM-stored parameters. Override any parameters already in RAM. 1 Don't use default parameters for RAM-stored parameter. Use the parameters already stored in RAM.
18 NDPRA	Not Use Default Parameters for RAM-stored Parameters For Conversion Pair A 0 Use default parameters for RAM-stored parameters. Override any parameters already in RAM. 1 Don't use default parameters for RAM-stored parameters. Use the parameters already stored in RAM.
17–16 POSTMODC[1-0]	Post-Processing Configuration for Conversion Pair C These bits will be read/write by user if ASRCTR:ATSC=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSC=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the post-processing configuration. 00 Select Upsampling-by-2 as defined in Signal Processing Flow. 01 Select Direct-Connection as defined in Signal Processing Flow. 10 Select Downsampling-by-2 as defined in Signal Processing Flow.
15–14 PREMODC[1-0]	Pre-Processing Configuration for Conversion Pair C These bits will be read/write by user if ASRCTR:ATSC=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSC=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the pre-processing configuration. 00 Select Upsampling-by-2 as defined in Signal processing flow 01 Select Direct-Connection as defined in Signal processing flow 10 Select Downsampling-by-2 as defined in Signal processing flow 11 Select passthrough mode. In this case, POSTMODC[1-0] have no use.
13–12 POSTMODB[1-0]	Post-Processing Configuration for Conversion Pair B These bits will be read/write by user if ASRCTR:ATSB=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSB=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the post-processing configuration.

Table continues on the next page...

ASRC_ASRCFG field descriptions (continued)

Field	Description
	00 Select Upsampling-by-2 as defined in Signal processing flow 01 Select Direct-Connection as defined in Signal processing flow 10 Select Downsampling-by-2 as defined in Signal processing flow
11–10 PREMODB[1-0]	Pre-Processing Configuration for Conversion Pair B These bits will be read/write by user if ASRCTR:ATSB=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSB=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the pre-processing configuration. 00 Select Upsampling-by-2 as defined in Signal processing flow 01 Select Direct-Connection as defined in Signal processing flow 10 Select Downsampling-by-2 as defined in Signal processing flow 11 Select passthrough mode. In this case, POSTMODB[1-0] have no use.
9–8 POSTMODA[1-0]	Post-Processing Configuration for Conversion Pair A These bits will be read/write by user if ASRCTR:ATSA=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSA=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the post-processing configuration. 00 Select Upsampling-by-2 as defined in Signal processing flow 01 Select Direct-Connection as defined in Signal processing flow 10 Select Downsampling-by-2 as defined in Signal processing flow
7–6 PREMODA[1-0]	Pre-Processing Configuration for Conversion Pair A These bits will be read/write by user if ASRCTR:ATSA=0, and can also be automatically updated by the ASRC internal logic if ASRCTR:ATSA=1 (see ASRC Misc Control Register 1 for Pair C). These bits set the selection of the pre-processing configuration. 00 Select Upsampling-by-2 as defined in Signal processing flow 01 Select Direct-Connection as defined in Signal processing flow 10 Select Downsampling-by-2 as defined in Signal processing flow 11 Select passthrough mode. In this case, POSTMODA[1-0] have no use.
-	This field is reserved. Reserved. Should be written as zero for compatibility.

15.7.5 ASRC Clock Source Register (ASRC_ASRCSR)

The ASRC clock source register (ASRCSR) is a 24-bit read/write register that controls the sources of the input and output clocks of the ASRC.

The clock connections are shown in ASRC Misc Control Register 1 for Pair C, also shown in Figure 1 :

Table 15-14. Bit Clock Definitions

Bit Clk Name	Definitions
0	ESAI RX clock
1	SSI-1 RX clock

Table continues on the next page...

Table 15-14. Bit Clock Definitions (continued)

Bit Clk Name	Definitions
2	SSI-2 RX clock
3	SSI-3 RX clock
4	SPDIF RX clock
5	MLB Bit clock
6	bit clock 6 should connect to one of the three pads: KEY_ROW3,GPIO_0,GPIO_18, which is configured by register IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT
7	tied to zero
8	ESAI TX clock
9	SSI-1 TX clock
a	SSI-2 TX clock
b	SSI-3 TX clock
c	SPDIF TX clock
d	bit clock d is configured by spdif1_clk_pred and spdif1_clk_podf in CCM_CDCDR, but it is better to describe it also in CCM spec.

Address: 203_4000h base + 14h offset = 203_4014h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								AOCSC				AOCSB				AOCSA				AICSC				AICSB				AICSA			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRCSR field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–20 AOCSC	**Output Clock Source C** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4 0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D 1110 bit clock E

Table continues on the next page...

ASRC_ASRCSR field descriptions (continued)

Field	Description
	1111 clock disabled, connected to zero any other value bit clock 0
19–16 AOCSB	**Output Clock Source B** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4 0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D 1110 bit clock E 1111 clock disabled, connected to zero any other value bit clock 0
15–12 AOCSA	**Output Clock Source A** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4 0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D 1110 bit clock E 1111 clock disabled, connected to zero any other value bit clock 0
11–8 AICSC	**Input Clock Source C** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4

Table continues on the next page...

ASRC_ASRCSR field descriptions (continued)

Field	Description
	0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D 1110 bit clock E 1111 clock disabled, connected to zero any other value bit clock 0
7–4 AICSB	**Input Clock Source B** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4 0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D 1110 bit clock E 1111 clock disabled, connected to zero any other value bit clock 0
AICSA	**Input Clock Source A** 0000 bit clock 0 0001 bit clock 1 0010 bit clock 2 0011 bit clock 3 0100 bit clock 4 0101 bit clock 5 0110 bit clock 6 0111 bit clock 7 1000 bit clock 8 1001 bit clock 9 1010 bit clock A 1011 bit clock B 1100 bit clock C 1101 bit clock D

Table continues on the next page...

ASRC_ASRCSR field descriptions (continued)

Field	Description
	1110 bit clock E 1111 clock disabled, connected to zero any other value bit clock 0

15.7.6 ASRC Clock Divider Register 1 (ASRC_ASRCDR1)

The ASRC clock divider register (ASRCDR1) is a 24-bit read/write register that controls the division factors of the ASRC input and output clock sources.

Address: 203_4000h base + 18h offset = 203_4018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								AOCDB			AOCPB			AOCDA		AOCPA				AICDB			AICPB			AICDA			AICPA			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRCDR1 field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–21 AOCDB	Output Clock Divider B Specify the divide ratio of the output clock divider B. The divide ratio may range from 1 to 8 (AOCDB[2:0] = 000 to 111).
20–18 AOCPB	Output Clock Prescaler B Specify the prescaling factor of the output prescaler B. The prescaling ratio may be any power of 2 from 1 to 128.
17–15 AOCDA	Output Clock Divider A Specify the divide ratio of the output clock divider A. The divide ratio may range from 1 to 8 (AOCDA[2:0] = 000 to 111).
14–12 AOCPA	Output Clock Prescaler A Specify the prescaling factor of the output prescaler A. The prescaling ratio may be any power of 2 from 1 to 128.
11–9 AICDB	Input Clock Divider B Specify the divide ratio of the input clock divider B. The divide ratio may range from 1 to 8 (AICDB[2:0] = 000 to 111).
8–6 AICPB	Input Clock Prescaler B Specify the prescaling factor of the input prescaler B. The prescaling ratio may be any power of 2 from 1 to 128.
5–3 AICDA	Input Clock Divider A

Table continues on the next page...

ASRC_ASRCDR1 field descriptions (continued)

Field	Description
	Specify the divide ratio of the input clock divider A. The divide ratio may range from 1 to 8 (AICDA[2:0] = 000 to 111).
AICPA	Input Clock Prescaler A Specify the prescaling factor of the input prescaler A. The prescaling ratio may be any power of 2 from 1 to 128.

15.7.7 ASRC Clock Divider Register 2 (ASRC_ASRCDR2)

The ASRC clock divider register (ASRCDR2) is a 24-bit read/write register that controls the division factors of the ASRC input and output clock sources.

Address: 203_4000h base + 1Ch offset = 203_401Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved												AOCDC			AOCPC			AICDC			AICPC		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

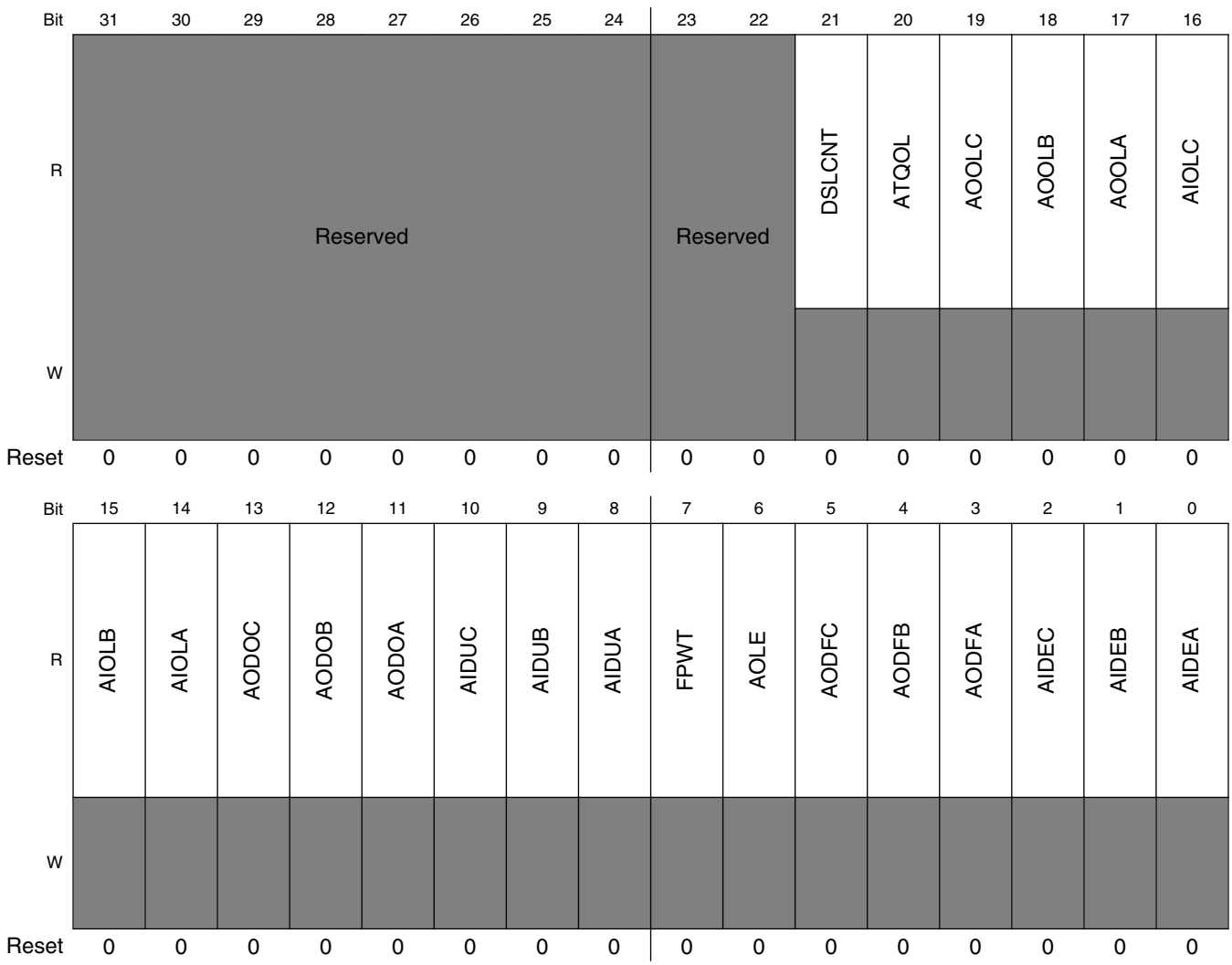
ASRC_ASRCDR2 field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–12 -	This field is reserved. Reserved. Should be written as zero for compatibility.
11–9 AOCDC	Output Clock Divider C Specify the divide ratio of the output clock divider C. The divide ratio may range from 1 to 8 (AOCDC[2:0] = 000 to 111).
8–6 AOCPC	Output Clock Prescaler C Specify the prescaling factor of the output prescaler C. The prescaling ratio may be any power of 2 from 1 to 128.
5–3 AICDC	Input Clock Divider C Specify the divide ratio of the input clock divider C. The divide ratio may range from 1 to 8 (AICDC[2:0] = 000 to 111).
AICPC	Input Clock Prescaler C Specify the prescaling factor of the input prescaler C. The prescaling ratio may be any power of 2 from 1 to 128.

15.7.8 ASRC Status Register (ASRC_ASRSTR)

The ASRC status register (ASRSTR) is a 24-bit read-write register used by the processor core to examine the status of the ASRC block and clear the overload interrupt request and AOLE flag bit. Read the status register will return the current state of ASRC.

Address: 203_4000h base + 20h offset = 203_4020h



ASRC_ASRSTR field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.

Table continues on the next page...

ASRC_ASRSTR field descriptions (continued)

Field	Description
23–22 -	This field is reserved. Reserved. Should be written as zero for compatibility.
21 DSLCNT	DSL Counter Input to FIFO ready When set, this bit indicates that new DSL counter information is stored in the internal ASRC FIFO. When clear, this bit indicates that new DSL counter information is in the process of storage into the internal ASRC FIFO. When ASRIER:AFPWE=1, the rising edge of this signal will propose an interrupt request. Writing any value with this bit set will clear the interrupt request proposed by the rising edge of this bit.
20 ATQOL	Task Queue FIFO overload When set, this bit indicates that task queue FIFO logic is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
19 AOOLC	Pair C Output Task Overload When set, this bit indicates that pair C output task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
18 AOOLB	Pair B Output Task Overload When set, this bit indicates that pair B output task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
17 AOOLA	Pair A Output Task Overload When set, this bit indicates that pair A output task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
16 AIOLC	Pair C Input Task Overload When set, this bit indicates that pair C input task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
15 AIOLB	Pair B Input Task Overload When set, this bit indicates that pair B input task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
14 AIOLA	Pair A Input Task Overload When set, this bit indicates that pair A input task is oveloaded. This may help to check the reason why overload interrupt happens. The bit is cleared when writing ASRSTR:AOLE as 1.
13 AODOC	Output Data Buffer C has overflowed When set, this bit indicates that output data buffer C has overflowed. When clear, this bit indicates that output data buffer C has not overflowed The bit is cleared when writing ASRSTR:AOLE as 1.
12 AODOB	Output Data Buffer B has overflowed

Table continues on the next page...

ASRC_ASRSTR field descriptions (continued)

Field	Description
	When set, this bit indicates that output data buffer B has overflowed. When clear, this bit indicates that output data buffer B has not overflowed The bit is cleared when writing ASRSTR:AOLE as 1.
11 AODOA	Output Data Buffer A has overflowed When set, this bit indicates that output data buffer A has overflowed. When clear, this bit indicates that output data buffer A has not overflowed The bit is cleared when writing ASRSTR:AOLE as 1.
10 AIDUC	Input Data Buffer C has underflowed When set, this bit indicates that input data buffer C has underflowed. When clear, this bit indicates that input data buffer C has not underflowed. The bit is cleared when writing ASRSTR:AOLE as 1.
9 AIDUB	Input Data Buffer B has underflowed When set, this bit indicates that input data buffer B has underflowed. When clear, this bit indicates that input data buffer B has not underflowed. The bit is cleared when writing ASRSTR:AOLE as 1.
8 AIDUA	Input Data Buffer A has underflowed When set, this bit indicates that input data buffer A has underflowed. When clear, this bit indicates that input data buffer A has not underflowed. The bit is cleared when writing ASRSTR:AOLE as 1.
7 FPWT	FP is in wait states This bit is for debug only. When set, this bit indicates that ASRC is in wait states. When clear, this bit indicates that ASRC is not in wait states.
6 AOLE	Overload Error Flag When set, this bit indicates that the task rate is too high for the ASRC to handle. The reasons for overload may be: - too high input clock frequency, - too high output clock frequency, - incorrect selection of the pre-filter, - low ASRC processing clock, - too many channels, - underrun, - or any combination of the reasons above. Since the ASRC uses the same hardware resources to perform various tasks, the real reason for the overload is not straight forward, and it should be carefully analyzed by the programmer. If ASRIER:AOLIE=1, an interrupt will be proposed when this bit is set. Write any value with this bit set as one into the status register will clear this bit and the interrupt request proposed by this bit.
5 AODFC	Number of data in Output Data Buffer C is greater than threshold

Table continues on the next page...

ASRC_ASRSTR field descriptions (continued)

Field	Description
	When set, this bit indicates that number of data already existing in ASRDORC is greater than threshold and the processor can read data from ASRDORC. When AODFC is set, the ASRC generates data output C interrupt request to the processor, if enabled (that is, ASRIER:ADOEC = 1). A DMA request is always generated when the AODFC bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.
4 AODFB	Number of data in Output Data Buffer B is greater than threshold When set, this bit indicates that number of data already existing in ASRDORB is greater than threshold and the processor can read data from ASRDORB. When AODFB is set, the ASRC generates data output B interrupt request to the processor, if enabled (that is, ASRIER:ADOEB = 1). A DMA request is always generated when the AODFB bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.
3 AODFA	Number of data in Output Data Buffer A is greater than threshold When set, this bit indicates that number of data already existing in ASRDORA is greater than threshold and the processor can read data from ASRDORA. When AODFA is set, the ASRC generates data output A interrupt request to the processor, if enabled (that is, ASRIER:ADOEA = 1). A DMA request is always generated when the AODFA bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.
2 AIDEC	Number of data in Input Data Buffer C is less than threshold When set, this bit indicates that number of data still available in ASRDIRC is less than threshold and the processor can write data to ASRDIRC. When AIDEC is set, the ASRC generates data input C interrupt request to the processor, if enabled (that is, ASRIER:ADIEC = 1). A DMA request is always generated when the AIDEC bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.
1 AIDEB	Number of data in Input Data Buffer B is less than threshold When set, this bit indicates that number of data still available in ASRDIRB is less than threshold and the processor can write data to ASRDIRB. When AIDEB is set, the ASRC generates data input B interrupt request to the processor, if enabled (that is, ASRIER:ADIEB = 1). A DMA request is always generated when the AIDEB bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.
0 AIDEA	Number of data in Input Data Buffer A is less than threshold When set, this bit indicates that number of data still available in ASRDIRA is less than threshold and the processor can write data to ASRDIRA. When AIDEA is set, the ASRC generates data input A interrupt request to the processor, if enabled (that is, ASRIER:ADIEA = 1). A DMA request is always generated when the AIDEA bit is set, but a DMA transfer takes place only if a DMA channel is active and triggered by this event.

15.7.9 ASRC Parameter Register n (ASRC_ASRPMn*n*)

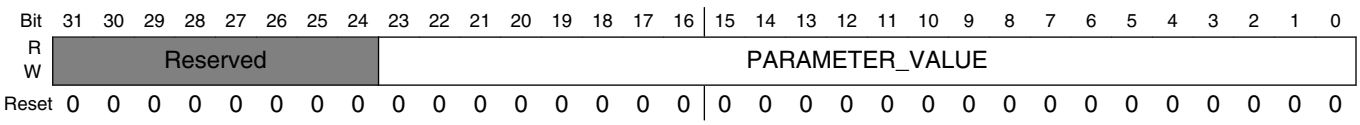
Parameter registers determine the performance of ASRC.

The parameter registers must be initialized by software before ASRC is enabled. Recommended values are given in ASRC Misc Control Register 1 for Pair C below,

Table 15-21. ASRC Parameter Registers (ASRPM1~ASRPM5)

Register	Offset	Access	Reset Value	Recommend Value
asrcpm1	0x40	R/W	0x00_0000	0x7fffff
asrcpm2	0x44	R/W	0x00_0000	0x255555
asrcpm3	0x48	R/W	0x00_0000	0xff7280
asrcpm4	0x4C	R/W	0x00_0000	0xff7280
asrcpm5	0x50	R/W	0x00_0000	0xff7280

Address: 203_4000h base + 40h offset + (4d × i), where i=0d to 4d



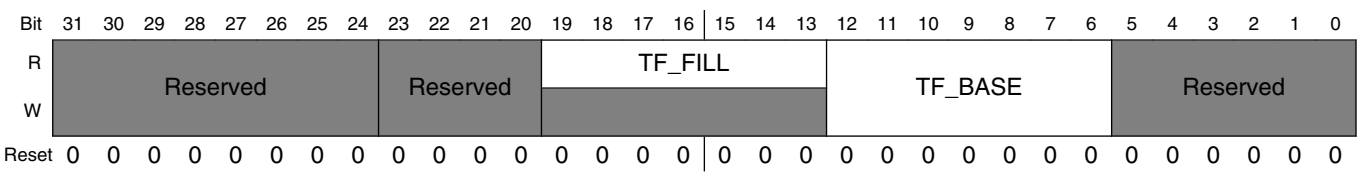
ASRC_ASRPMn*n* field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
PARAMETER_ VALUE	See recommended values table.

15.7.10 ASRC ASRC Task Queue FIFO Register 1 (ASRC_ASRTFR1)

The register defines and shows the parameters for ASRC inner task queue FIFOs.

Address: 203_4000h base + 54h offset = 203_4054h



ASRC_ASRTFR1 field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–20 -	This field is reserved. Reserved. Should be written as zero for compatibility.

Table continues on the next page...

ASRC_ASRTFR1 field descriptions (continued)

Field	Description
19–13 TF_FILL	Current number of entries in task queue FIFO.
12–6 TF_BASE	Base address for task queue FIFO. Set to 0x7C.
-	This field is reserved. Reserved. Should be written as zero for compatibility.

15.7.11 ASRC Channel Counter Register (ASRC_ASRCCR)

The ASRC channel counter register (ASRCCR) is a 24-bit read/write register that sets and reflects the current specific input/output FIFO being accessed through shared peripheral bus for each ASRC conversion pair.

Address: 203_4000h base + 5Ch offset = 203_405Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								ACOC				ACOB				ACOA				ACIC				ACIB				ACIA			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRCCR field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–20 ACOC	The channel counter for Pair C's output FIFO These bits stand for the current channel being accessed through shared peripheral bus for Pair C's output FIFO's usage. The value can be any value between [0, ANCC-1]
19–16 ACOB	The channel counter for Pair B's output FIFO These bits stand for the current channel being accessed through shared peripheral bus for Pair B's output FIFO's usage. The value can be any value between [0, ANCB-1]
15–12 ACOA	The channel counter for Pair A's output FIFO These bits stand for the current channel being accessed through shared peripheral bus for Pair A's output FIFO's usage. The value can be any value between [0, ANCA-1]
11–8 ACIC	The channel counter for Pair C's input FIFO These bits stand for the current channel being accessed through shared peripheral bus for Pair C's input FIFO's usage. The value can be any value between [0, ANCC-1]
7–4 ACIB	The channel counter for Pair B's input FIFO These bits stand for the current channel being accessed through shared peripheral bus for Pair B's input FIFO's usage. The value can be any value between [0, ANCB-1]
ACIA	The channel counter for Pair A's input FIFO

Table continues on the next page...

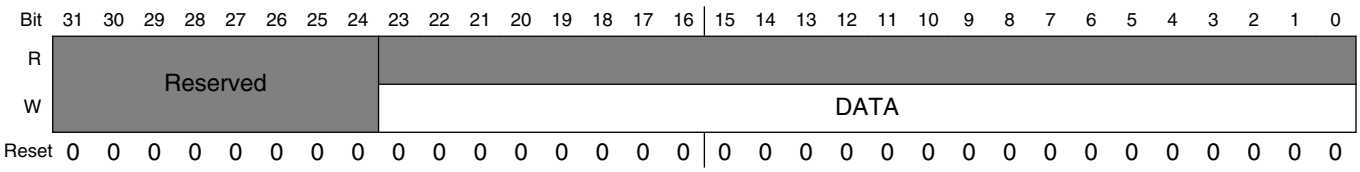
ASRC_ASRCCR field descriptions (continued)

Field	Description
	These bits stand for the current channel being accessed through shared peripheral bus for Pair A's input FIFO's usage. The value can be any value between [0, ANCA-1]

15.7.12 ASRC Data Input Register for Pair *x* (ASRC_ASRDI*n*)

These registers are the interface registers for the audio data input of pair A,B,C respectively. They are backed by FIFOs.

Address: 203_4000h base + 60h offset + (8d × i), where i=0d to 2d



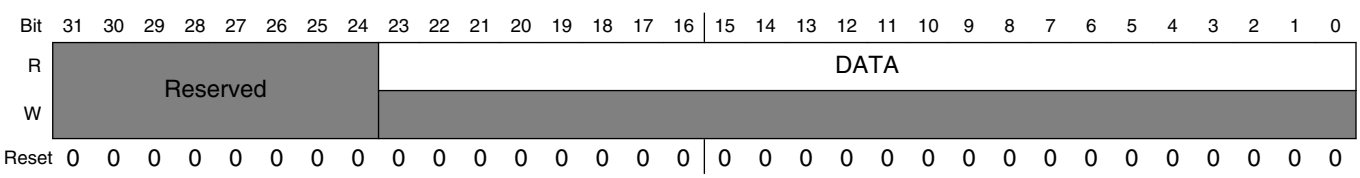
ASRC_ASRDI*n* field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
DATA	Audio data input

15.7.13 ASRC Data Output Register for Pair *x* (ASRC_ASRDO*n*)

These registers are the interface registers for the audio data output of pair A,B,C respectively. They are backed by FIFOs.

Address: 203_4000h base + 64h offset + (8d × i), where i=0d to 2d



ASRC_ASRDO*n* field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
DATA	Audio data output

15.7.14 ASRC Ideal Ratio for Pair A-High Part (ASRC_ASRIDRHA)

The ideal ratio registers (ASRIDRHA, ASRIDRLA) hold the ratio value IDRATIOA. IDRATIOA = $Fs_{inA}/Fs_{outA} = Ts_{outA}/Ts_{inA}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRA, IDRA}=2'b11.

Address: 203_4000h base + 80h offset = 203_4080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved																IDRATIOA[31:24]							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

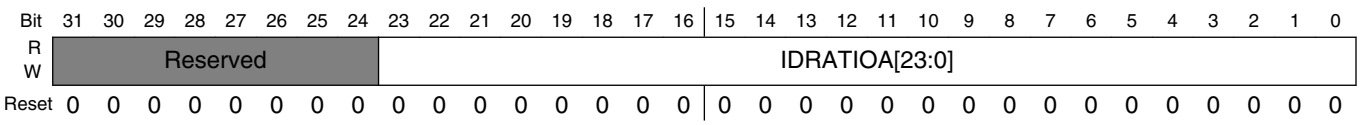
ASRC_ASRIDRHA field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–8 -	This field is reserved. Reserved
IDRATIOA[31:24]	IDRATIOA[31:24]. High part of ideal ratio value for pair A

15.7.15 ASRC Ideal Ratio for Pair A -Low Part (ASRC_ASRIDRLA)

The ideal ratio registers (ASRIDRHA, ASRIDRLA) hold the ratio value IDRATIOA. IDRATIOA = $Fs_{inA}/Fs_{outA} = Ts_{outA}/Ts_{inA}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRA, IDRA}=2'b11.

Address: 203_4000h base + 84h offset = 203_4084h



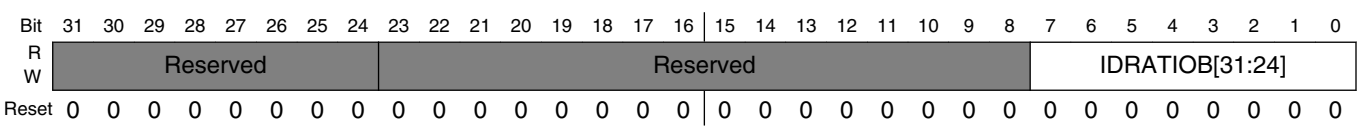
ASRC_ASRIDRLA field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
IDRATIOA[23:0]	IDRATIOA[23:0]. Low part of ideal ratio value for pair A

15.7.16 ASRC Ideal Ratio for Pair B-High Part (ASRC_ASRIDRHB)

The ideal ratio registers (ASRIDRHB, ASRIDRLB) hold the ratio value IDRATIOB. IDRATIOB = $Fs_{inB}/Fs_{outB} = Ts_{outB}/Ts_{inB}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRB, IDRB}=2'b11.

Address: 203_4000h base + 88h offset = 203_4088h



ASRC_ASRIDRHB field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.

Table continues on the next page...

ASRC_ASRIDRHB field descriptions (continued)

Field	Description
23–8 -	This field is reserved. Reserved
IDRATIOB[31:24]	IDRATIOB[31:24]. High part of ideal ratio value for pair B.

15.7.17 ASRC Ideal Ratio for Pair B-Low Part (ASRC_ASRIDRLB)

The ideal ratio registers (ASRIDRHB, ASRIDRLB) hold the ratio value IDRATIOB. IDRATIOB = $Fs_{inB}/Fs_{outB} = Ts_{outB}/Ts_{inB}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRB, IDRB}=2'b11.

Address: 203_4000h base + 8Ch offset = 203_408Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								IDRATIOB[23:0]																							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRIDRLB field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
IDRATIOB[23:0]	IDRATIOB[23:0]. Low part of ideal ratio value for pair B.

15.7.18 ASRC Ideal Ratio for Pair C-High Part (ASRC_ASRIDRHC)

The ideal ratio registers (ASRIDRHC, ASRIDRLC) hold the ratio value IDRATIOC. IDRATIOC = $Fs_{inC}/Fs_{outC} = Ts_{outC}/Ts_{inC}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRC, IDRC}=2'b11.

Address: 203_4000h base + 90h offset = 203_4090h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved																IDRATIOC[31:24]							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRIDRHC field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–8 -	This field is reserved. Reserved
IDRATIOC[31:24]	IDRATIOC[31:24]. High part of ideal ratio value for pair C.

15.7.19 ASRC Ideal Ratio for Pair C-Low Part (ASRC_ASRIDRLC)

The ideal ratio registers (ASRIDRHC, ASRIDRLC) hold the ratio value IDRATIOC. IDRATIOC = $Fs_{inC}/Fs_{outC} = Ts_{outC}/Ts_{inC}$ is a 32-bit fixed point value with 26 fractional bits. This value is only useful when ASRCTR:{USRC, IDRC}=2'b11.

Address: 203_4000h base + 94h offset = 203_4094h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								IDRATIOC[23:0]																							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRIDRLC field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
IDRATIOC[23:0]	IDRATIOC[23:0]. Low part of ideal ratio value for pair C.

15.7.20 ASRC 76kHz Period in terms of ASRC processing clock (ASRC_ASR76K)

The register (ASR76K) holds the period of the 76kHz sampling clock in terms of the ASRC processing clock with frequency Fs_{ASRC}. ASR76K = Fs_{ASRC}/Fs_{76k}. Reset value is 0x0A47 which assumes that Fs_{ASRC}=200MHz. This register is used to help the ASRC internal logic to decide the pre-processing and the post-processing options automatically (see ASRC Misc Control Register 1 for Pair C and ASRC Misc Control Register 1 for Pair C). In a system when Fs_{ASRC}=133MHz, the value should be assigned explicitly as 0x06D6 in user application code.

Address: 203_4000h base + 98h offset = 203_4098h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved								ASR76K															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	1	1	1

ASRC_ASR76K field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–17 -	This field is reserved. Reserved
ASR76K	Value for the period of the 76kHz sampling clock.

15.7.21 ASRC 56kHz Period in terms of ASRC processing clock (ASRC_ASR56K)

The register (ASR56K) holds the period of the 56kHz sampling clock in terms of the ASRC processing clock with frequency Fs_{ASRC}. ASR56K = Fs_{ASRC}/Fs_{56k}. Reset value is 0x0DF3 which assumes that Fs_{ASRC}=200MHz. This register is used to help the ASRC internal logic to decide the pre-processing and the post-processing options automatically (see ASRC Misc Control Register 1 for Pair C and ASRC Misc Control Register 1 for Pair C). In a system when Fs_{ASRC}=133MHz, the value should be assigned explicitly as 0x0947 in user application code.

Address: 203_4000h base + 9Ch offset = 203_409Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								Reserved								ASR56K															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	0	0	1	1

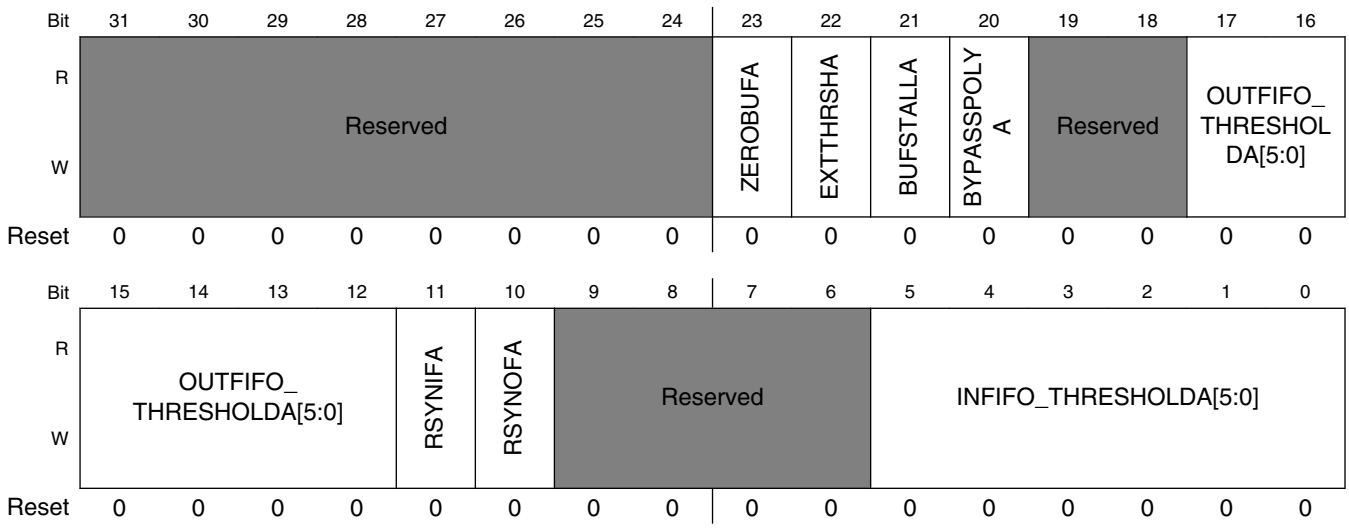
ASRC_ASR56K field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–17 -	This field is reserved. Reserved
ASR56K	Value for the period of the 56kHz sampling clock

15.7.22 ASRC Misc Control Register for Pair A (ASRC_ASRMCRA)

The register (ASRMCRA) is used to control Pair A internal logic.

Address: 203_4000h base + A0h offset = 203_40A0h



ASRC_ASRMCRA field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 ZEROBUFA	Initialize buf of Pair A when pair A is enabled. Always clear option. This bit is used to control whether the buffer is to be zeroized when pair A is enabled. 1 Don't zeroize the buffer 0 Zeroize the buffer
22 EXTTHRSHA	Use external thresholds for FIFO control of Pair A This bit will determine whether the FIFO thresholds externally defined in this register is used to control ASRC internal FIFO logic for pair A. 1 Use external defined thresholds. 0 Use default thresholds.
21 BUFSTALLA	Stall Pair A conversion in case of Buffer Near Empty/Full Condition This bit will determine whether the near empty/full FIFO condition will stall the rate conversion for pair A. This option can only work when external ratio is used. Near empty condition is the condition when input FIFO has less than 4 useful samples per channel. Near full condition is the condition when the output FIFO has less than 4 vacant sample words to fill per channel.

Table continues on the next page...

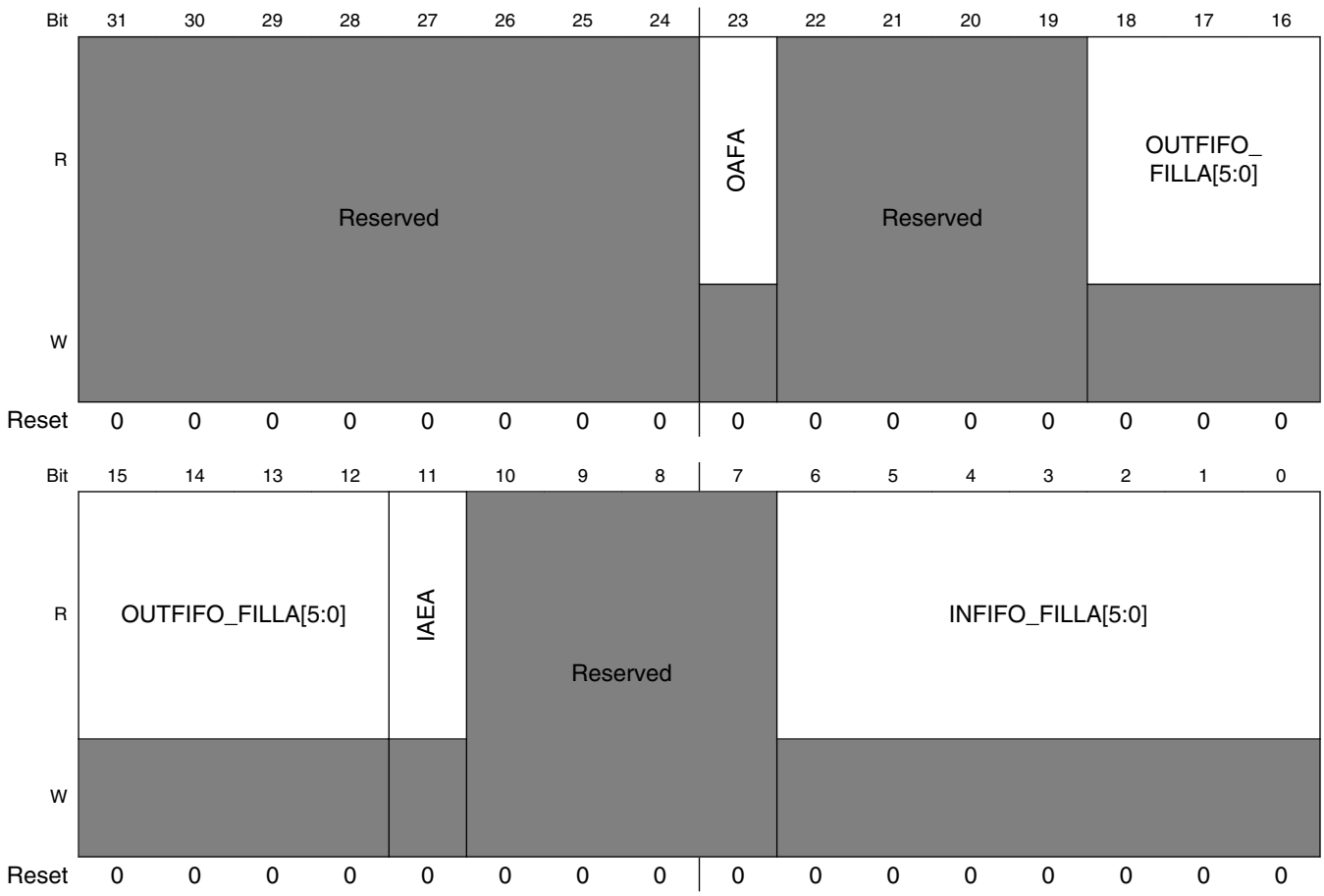
ASRC_ASRMCRA field descriptions (continued)

Field	Description
	1 Stall Pair A conversion in case of near empty/full FIFO conditions. 0 Don't stall Pair A conversion even in case of near empty/full FIFO conditions.
20 BYPASSPOLYA	Bypass Polyphase Filtering for Pair A This bit will determine whether the polyphase filtering part of Pair A conversion will be bypassed. 1 Bypass polyphase filtering. 0 Don't bypass polyphase filtering.
19–18 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
17–12 OUTFIFO_ THRESHOLDA[5:0]	The threshold for Pair A's output FIFO per channel These bits stand for the threshold for Pair A's output FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of output FIFO fillings of the pair is greater than n samples per channel, the output data ready flag is set; when the number of output FIFO fillings of the pair is less than or equal to n samples per channel, the output data ready flag is automatically cleared.
11 RSYNIFA	Re-sync Input FIFO Channel Counter If bit set, force ASRCCR:ACIA=0. If bit clear, untouch ASRCCR:ACIA.
10 RSYNOFA	Re-sync Output FIFO Channel Counter If bit set, force ASRCCR:ACOA=0. If bit clear, untouch ASRCCR:ACOA.
9–6 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_ THRESHOLDA[5:0]	The threshold for Pair A's input FIFO per channel These bits stand for the threshold for Pair A's input FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of input FIFO fillings of the pair is less than n samples per channel, the input data needed flag is set; when the number of input FIFO fillings of the pair is greater than or equal to n samples per channel, the input data needed flag is automatically cleared.

15.7.23 ASRC FIFO Status Register for Pair A (ASRC_ASRFSTA)

The register (ASRFSTA) is used to show Pair A internal FIFO conditions.

Address: 203_4000h base + A4h offset = 203_40A4h



ASRC_ASRFSTA field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 OAFA	Output FIFO is near Full for Pair A This bit is to indicate whether the output FIFO of Pair A is near full.
22–19 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
18–12 OUTFIFO_ FILLA[5:0]	The fillings for Pair A's output FIFO per channel These bits stand for the fillings for Pair A's output FIFO per channel. Possible range is [0,64].

Table continues on the next page...

ASRC_ASRFSTA field descriptions (continued)

Field	Description
11 IAEA	Input FIFO is near Empty for Pair A This bit is to indicate whether the input FIFO of Pair A is near empty.
10–7 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_FILLA[5:0]	The fillings for Pair A's input FIFO per channel These bits stand for the fillings for Pair A's input FIFO per channel. Possible range is [0,64].

15.7.24 ASRC Misc Control Register for Pair B (ASRC_ASRMCRB)

The register (ASRMCRB) is used to control Pair B internal logic.

Address: 203_4000h base + A8h offset = 203_40A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved								ZEROBUFB	EXTTHRSHB	BUFSTALLB	BYPASSPOLYB	Reserved		OUTFIFO_THRESHOLDB[5:0]	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OUTFIFO_THRESHOLDB[5:0]				RSYNIFB	RSYNOFB	Reserved				INFIFO_THRESHOLDB[5:0]					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRMCRB field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 ZEROBUFB	Initialize buf of Pair B when pair B is enabled This bit is used to control whether the buffer is to be zeroized when pair B is enabled. 1 Don't zeroize the buffer 0 Zeroize the buffer

Table continues on the next page...

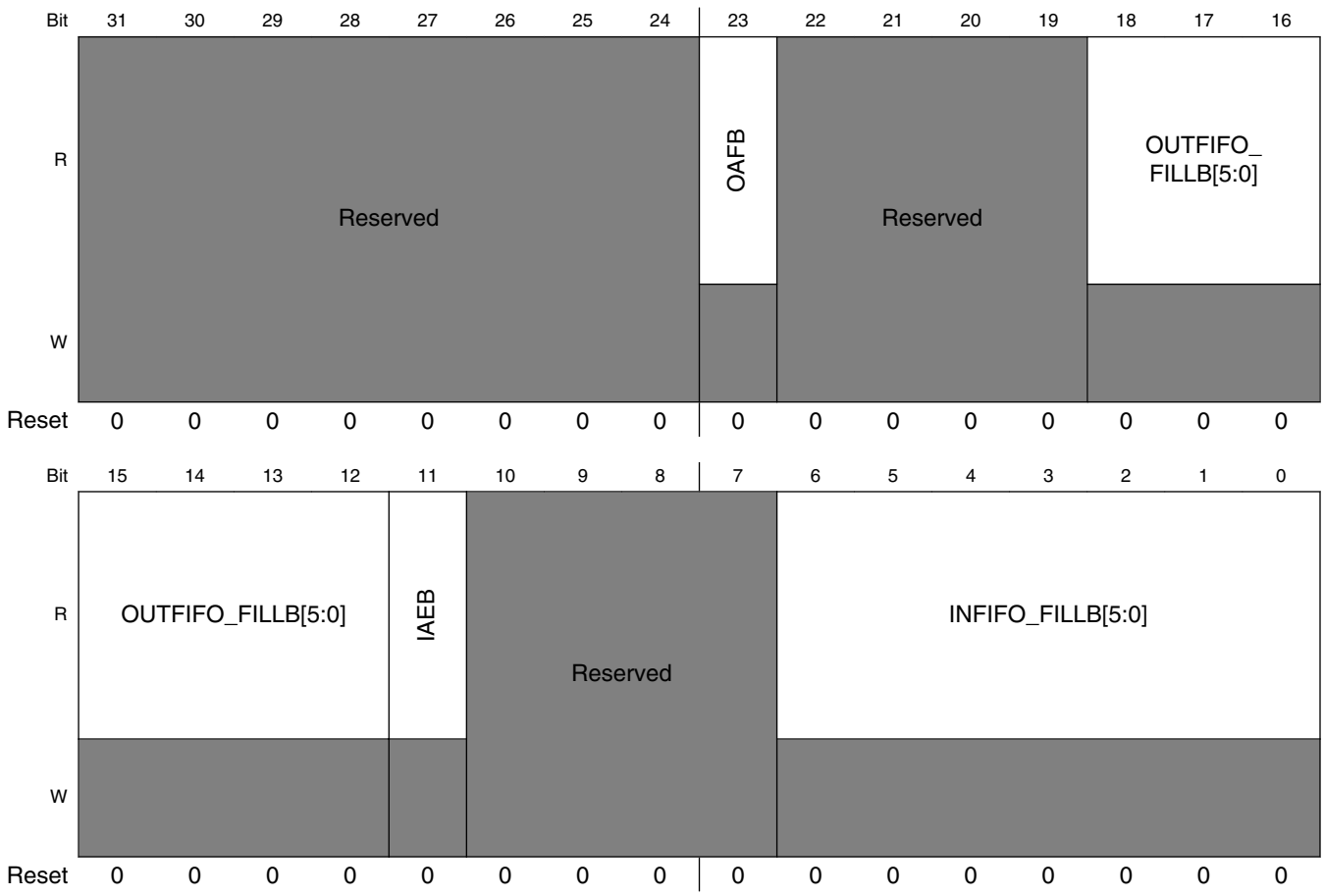
ASRC_ASRMCRB field descriptions (continued)

Field	Description
22 EXTTHRSHB	Use external thresholds for FIFO control of Pair B This bit will determine whether the FIFO thresholds externally defined in this register is used to control ASRC internal FIFO logic for pair B. 1 Use external defined thresholds. 0 Use default thresholds.
21 BUFSTALLB	Stall Pair B conversion in case of Buffer Near Empty/Full Condition This bit will determine whether the near empty/full FIFO condition will stall the rate conversion for pair B. This option can only work when external ratio is used. Near empty condition is the condition when input FIFO has less than 4 useful samples per channel. Near full condition is the condition when the output FIFO has less than 4 vacant sample words to fill per channel. 1 Stall Pair B conversion in case of near empty/full FIFO conditions. 0 Don't stall Pair B conversion even in case of near empty/full FIFO conditions.
20 BYPASSPOLYB	Bypass Polyphase Filtering for Pair B This bit will determine whether the polyphase filtering part of Pair B conversion will be bypassed. 1 Bypass polyphase filtering. 0 Don't bypass polyphase filtering.
19–18 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
17–12 OUTFIFO_ THRESHOLDB[5:0]	The threshold for Pair B's output FIFO per channel These bits stand for the threshold for Pair B's output FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of output FIFO fillings of the pair is greater than n samples per channel, the output data ready flag is set; when the number of output FIFO fillings of the pair is less than or equal to n samples per channel, the output data ready flag is automatically cleared.
11 RSYNIFB	Re-sync Input FIFO Channel Counter If bit set, force ASRCCR:ACIB=0. If bit clear, untouch ASRCCR:ACIB.
10 RSYNOFB	Re-sync Output FIFO Channel Counter If bit set, force ASRCCR:ACOB=0. If bit clear, untouch ASRCCR:ACOB.
9–6 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_ THRESHOLDB[5:0]	The threshold for Pair B's input FIFO per channel These bits stand for the threshold for Pair B's input FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of input FIFO fillings of the pair is less than n samples per channel, the input data needed flag is set; when the number of input FIFO fillings of the pair is greater than or equal to n samples per channel, the input data needed flag is automatically cleared.

15.7.25 ASRC FIFO Status Register for Pair B (ASRC_ASRFSTB)

The register (ASRFSTB) is used to show Pair B internal FIFO conditions.

Address: 203_4000h base + ACh offset = 203_40ACh



ASRC_ASRFSTB field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 OAFB	Output FIFO is near Full for Pair B This bit is to indicate whether the output FIFO of Pair B is near full.
22–19 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
18–12 OUTFIFO_ FILLB[5:0]	The fillings for Pair B's output FIFO per channel These bits stand for the fillings for Pair B's output FIFO per channel. Possible range is [0,64].

Table continues on the next page...

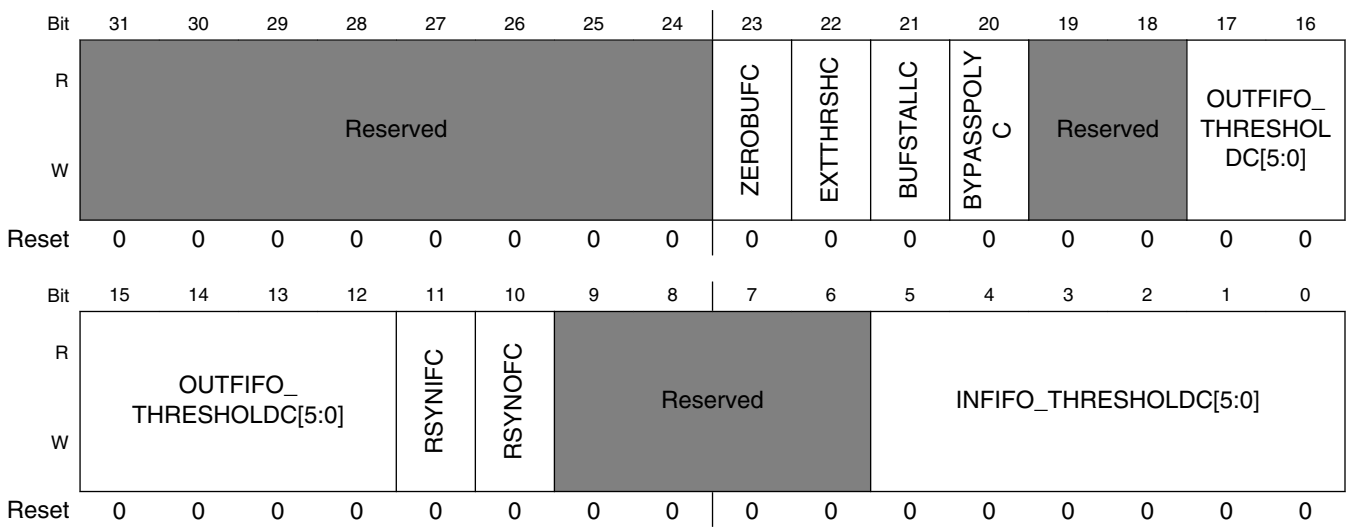
ASRC_ASRFSTB field descriptions (continued)

Field	Description
11 IAEB	Input FIFO is near Empty for Pair B This bit is to indicate whether the input FIFO of Pair B is near empty.
10–7 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_FILLB[5:0]	The fillings for Pair B's input FIFO per channel These bits stand for the fillings for Pair B's input FIFO per channel. Possible range is [0,64].

15.7.26 ASRC Misc Control Register for Pair C (ASRC_ASRMCRC)

The register (ASRMCRC) is used to control Pair C internal logic.

Address: 203_4000h base + B0h offset = 203_40B0h



ASRC_ASRMCRC field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 ZEROBUFC	Initialize buf of Pair C when pair C is enabled This bit is used to control whether the buffer is to be zeroized when pair C is enabled. 1 Don't zeroize the buffer 0 Zeroize the buffer

Table continues on the next page...

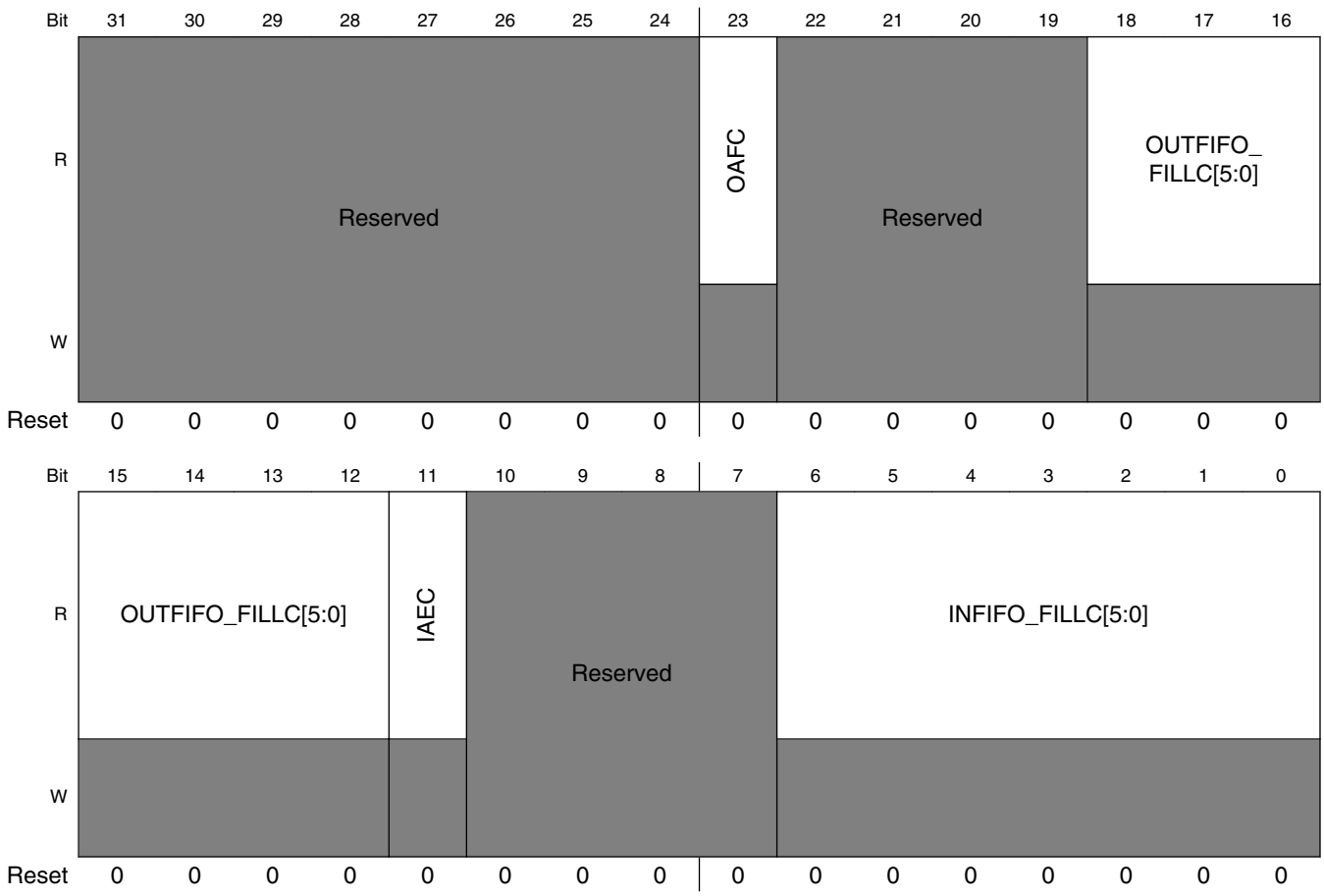
ASRC_ASRMCRC field descriptions (continued)

Field	Description
22 EXTTHRSHC	Use external thresholds for FIFO control of Pair C This bit will determine whether the FIFO thresholds externally defined in this register is used to control ASRC internal FIFO logic for pair C. 1 Use external defined thresholds. 0 Use default thresholds.
21 BUFSTALLC	Stall Pair C conversion in case of Buffer Near Empty/Full Condition This bit will determine whether the near empty/full FIFO condition will stall the rate conversion for pair C. This option can only work when external ratio is used. Near empty condition is the condition when input FIFO has less than 4 useful samples per channel. Near full condition is the condition when the output FIFO has less than 4 vacant sample words to fill per channel. 1 Stall Pair C conversion in case of near empty/full FIFO conditions. 0 Don't stall Pair C conversion even in case of near empty/full FIFO conditions.
20 BYPASSPOLYC	Bypass Polyphase Filtering for Pair C This bit will determine whether the polyphase filtering part of Pair C conversion will be bypassed. 1 Bypass polyphase filtering. 0 Don't bypass polyphase filtering.
19–18 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
17–12 OUTFIFO_ THRESHOLDC[5:0]	The threshold for Pair C's output FIFO per channel These bits stand for the threshold for Pair C's output FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of output FIFO fillings of the pair is greater than n samples per channel, the output data ready flag is set; when the number of output FIFO fillings of the pair is less than or equal to n samples per channel, the output data ready flag is automatically cleared.
11 RSYNIFC	Re-sync Input FIFO Channel Counter If bit set, force ASRCCR:ACIC=0. If bit clear, untouch ASRCCR:ACIC.
10 RSYNOFC	Re-sync Output FIFO Channel Counter If bit set, force ASRCCR:ACOC=0. If bit clear, untouch ASRCCR:ACOC.
9–6 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_ THRESHOLDC[5:0]	The threshold for Pair C's input FIFO per channel These bits stand for the threshold for Pair C's input FIFO per channel. Possible range is [0,63]. When the value is n, it means that: when the number of input FIFO fillings of the pair is less than n samples per channel, the input data needed flag is set; when the number of input FIFO fillings of the pair is greater than or equal to n samples per channel, the input data needed flag is automatically cleared.

15.7.27 ASRC FIFO Status Register for Pair C (ASRC_ASRFSTC)

The register (ASRFSTC) is used to show Pair C internal FIFO conditions.

Address: 203_4000h base + B4h offset = 203_40B4h



ASRC_ASRFSTC field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23 OAFC	Output FIFO is near Full for Pair C This bit is to indicate whether the output FIFO of Pair C is near full.
22–19 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
18–12 OUTFIFO_ FILLC[5:0]	The fillings for Pair C's output FIFO per channel These bits stand for the fillings for Pair C's output FIFO per channel. Possible range is [0,64].

Table continues on the next page...

ASRC_ASRFSTC field descriptions (continued)

Field	Description
11 IAEC	Input FIFO is near Empty for Pair C This bit is to indicate whether the input FIFO of Pair C is near empty.
10–7 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
INFIFO_FILLC[5:0]	The fillings for Pair C's input FIFO per channel These bits stand for the fillings for Pair C's input FIFO per channel. Possible range is [0,64].

15.7.28 ASRC Misc Control Register 1 for Pair X (ASRC_ASRMCR1*n*)

The register (ASRMCR1x) is used to control Pair *x* internal logic (for data alignment etc.).

The bit assignment for all the input data formats is the same as that supported by the SSI.

Address: 203_4000h base + C0h offset + (4d × i), where i=0d to 2d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved								Reserved							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				IWD[2:0]			IMSB	Reserved					OMSB	OSGN	OW16
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASRC_ASRMCR1*n* field descriptions

Field	Description
31–24 [unimplemented]	This is a 24-bit register the upper byte is unimplemented. This field is reserved.
23–12 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
11–9 IWD[2:0]	Data Width of the input FIFO These three bits will determine the bitwidth for the audio data into ASRC All other settings not shown are reserved. 3'b000 24-bit audio data.

Table continues on the next page...

ASRC_ASRMCR1*n* field descriptions (continued)

Field	Description
	3'b001 16-bit audio data. 3'b010 8-bit audio data.
8 IMSB	Data Alignment of the input FIFO This bit will determine the data alignment of the input FIFO. 1 MSB aligned. 0 LSB aligned.
7–3 -	This field is reserved. Reserved. Should be written as zero for future compatibility.
2 OMSB	Data Alignment of the output FIFO This bit will determine the data alignment of the output FIFO. 1 MSB aligned. 0 LSB aligned.
1 OSGN	Sign Extension Option of the output FIFO This bit will determine the sign extension option of the output FIFO. 1 Sign extension. 0 No sign extension.
0 OW16	Bit Width Option of the output FIFO This bit will determine the bit width option of the output FIFO. 1 16-bit output data 0 24-bit output data.

Chapter 16
Digital Audio Multiplexer (AUDMUX)

16.1 Overview

The Digital Audio Multiplexer (AUDMUX) provides a programmable interconnect device for voice, audio, and synchronous data routing between Synchronous Serial Interface Controller (SSI) and audio/voice codec's (also known as coder-decoders) peripheral serial interfaces.

This section includes a top level diagram that shows the functional organization of the block, including all off-chip signals.

AUDMUX allows the users to reconfigure the audio system signal routing through programming, as opposed to altering the PCB design. The full description of the block is in Functional Description.

Figure 16-1 shows the block diagram.

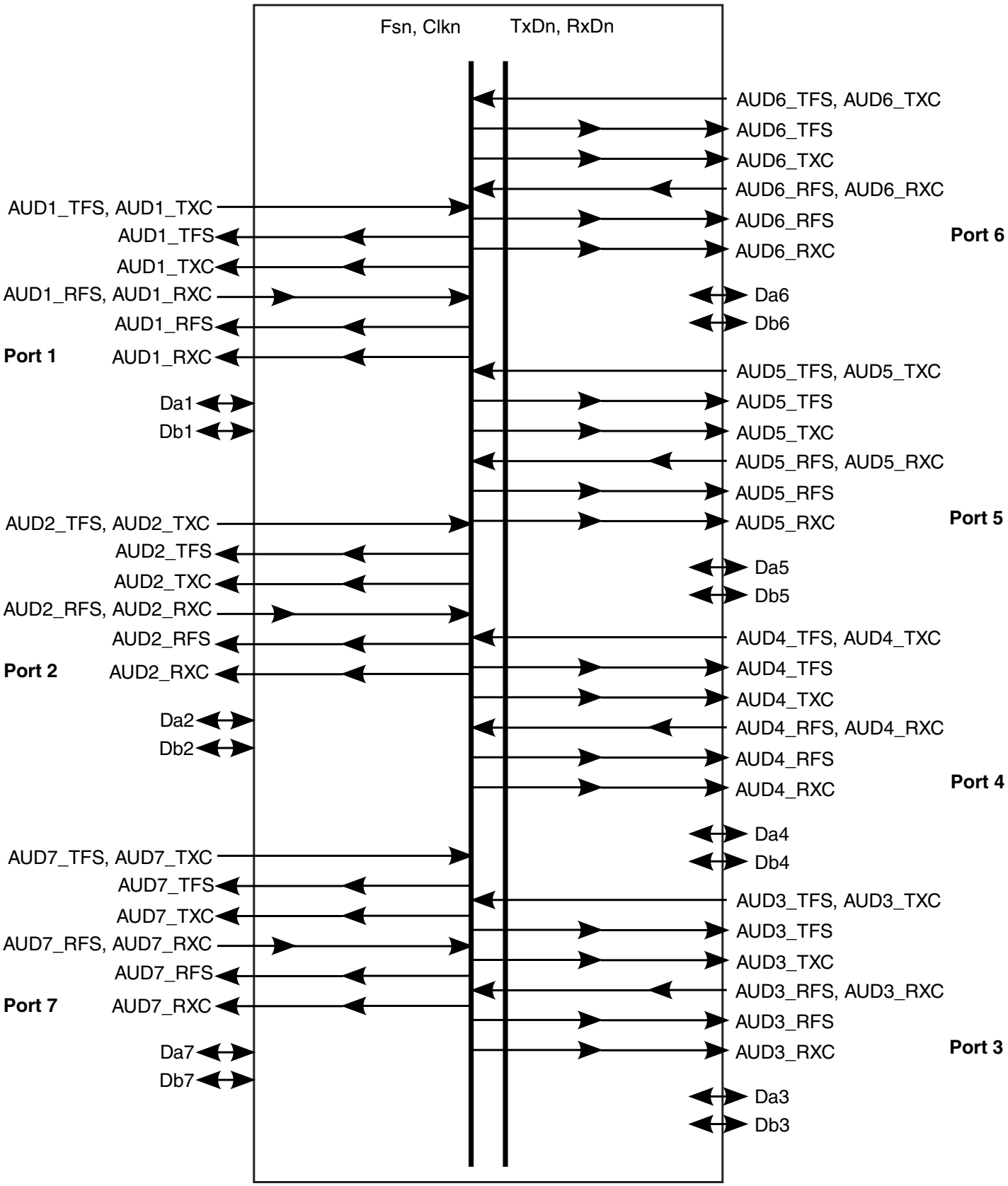


Figure 16-1. AUDMUX Block Diagram

AUDMUX supports single source to single destination connectivity or single source to multiple destination connectivity.

AUDMUX includes two types of interfaces: Internal ports and External ports. Internal ports are hard wired to Synchronous Serial Interface Controller (SSI). The connection between each SSI and AUDMUX's Internal ports cannot be modified, however, routing of the signals connected to each of the internal port can be routed within AUDMUX. External ports are connected to IOMUX module where the ports connect to off-chip audio devices and serial interfaces of other processors. The connectivity of the External port and IOMUX cannot be configured, but the output or input of the signal can be routed easily by setting the appropriate AUDMUX registers.

16.1.1 Features

Key features of the block include:

- Three internal ports
- Four external ports
- Full 6-wire SSI interfaces for asynchronous receive and transmit
- Configurable 4-wire (synchronous) or 6-wire (asynchronous) peripheral interfaces
- Independent Tx/Rx Frame sync and clock direction selection for host or peripheral
- Each host interface's capability to connect to any other host or peripheral interface in a point-to-point or point-to-multipoint (network mode)

16.1.2 Modes and Operations

The AUDMUX supports the modes described in Operating Modes.

16.2 External Signals

The following table describes the external signals of AUDMUX:

Table 16-1. AUDMUX External Signals

Signal	Description	Pad	Mode	Direction
AUD3_RXC (RXC)	Receive clock signal	CSI0_DAT10	ALT1	IO
AUD3_RXD (RXD)	Data receive signal	CSI0_DAT7	ALT4	IO
AUD3_RXFS (RXFS)	Receive Frame sync signal	CSI0_DAT11	ALT1	IO
AUD3_TXC (TXC)	Transmit clock signal	CSI0_DAT4	ALT4	IO

Table continues on the next page...

Table 16-1. AUDMUX External Signals (continued)

Signal	Description	Pad	Mode	Direction
AUD3_TXD (TXD)	Data transmit signal	CSI0_DAT5	ALT4	IO
AUD3_TXFS (TXFS)	Transmit Frame sync signal	CSI0_DAT6	ALT4	IO
AUD4_RXC (RXC)	Receive clock signal	DISP0_DAT19	ALT4	IO
		SD2_CMD	ALT3	
AUD4_RXD (RXD)	Data receive signal	DISP0_DAT23	ALT3	IO
		SD2_DAT0	ALT3	
AUD4_RXFS (RXFS)	Receive Frame sync signal	DISP0_DAT18	ALT4	IO
		SD2_CLK	ALT3	
AUD4_TXC (TXC)	Transmit clock signal	DISP0_DAT20	ALT3	IO
		SD2_DAT3	ALT3	
AUD4_TXD (TXD)	Data transmit signal	DISP0_DAT21	ALT3	IO
		SD2_DAT2	ALT3	
AUD4_TXFS (TXFS)	Transmit Frame sync signal	DISP0_DAT22	ALT3	IO
		SD2_DAT1	ALT3	
AUD5_RXC (RXC)	Receive clock signal	DISP0_DAT14	ALT3	IO
		EIM_D25	ALT6	
AUD5_RXD (RXD)	Data receive signal	DISP0_DAT19	ALT3	IO
		KEY_ROW1	ALT2	
AUD5_RXFS (RXFS)	Receive Frame sync signal	DISP0_DAT13	ALT3	IO
		EIM_D24	ALT6	
AUD5_TXC (TXC)	Transmit clock signal	DISP0_DAT16	ALT3	IO
		KEY_COL0	ALT2	
AUD5_TXD (TXD)	Data transmit signal	DISP0_DAT17	ALT3	IO
		KEY_ROW0	ALT2	
AUD5_TXFS (TXFS)	Transmit Frame sync signal	DISP0_DAT18	ALT3	IO
		KEY_COL1	ALT2	
AUD6_RXC (RXC)	Receive clock signal	DISP0_DAT6	ALT3	IO
AUD6_RXD (RXD)	Data receive signal	DI0_PIN4	ALT2	IO
AUD6_RXFS (RXFS)	Receive Frame sync signal	DISP0_DAT5	ALT3	IO
AUD6_TXC (TXC)	Transmit clock signal	DI0_PIN15	ALT2	IO
AUD6_TXD (TXD)	Data transmit signal	DI0_PIN2	ALT2	IO
AUD6_TXFS (TXFS)	Transmit Frame sync signal	DI0_PIN3	ALT2	IO

16.3 Clocks

This section provides information about AUDMUX clocking including clock inputs and the clock diagram.

The following table describes the clock source for AUDMUX. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 16-2. AUDMUX Clocks

Clock name	Clock Root	Description
ipg_clk_s	ipg_clk_root	Peripheral access clock

16.3.1 Clock Inputs

The IP Bus read/write clock-peripheral clock (ipg_clk_s) is an input to the AUDMUX. It is used for all AUDMUX register accesses. It is driven only when there is an AUDMUX access on the IP Bus.

16.3.2 Clock Diagram

The figure below shows the clocking used in the AUDMUX.

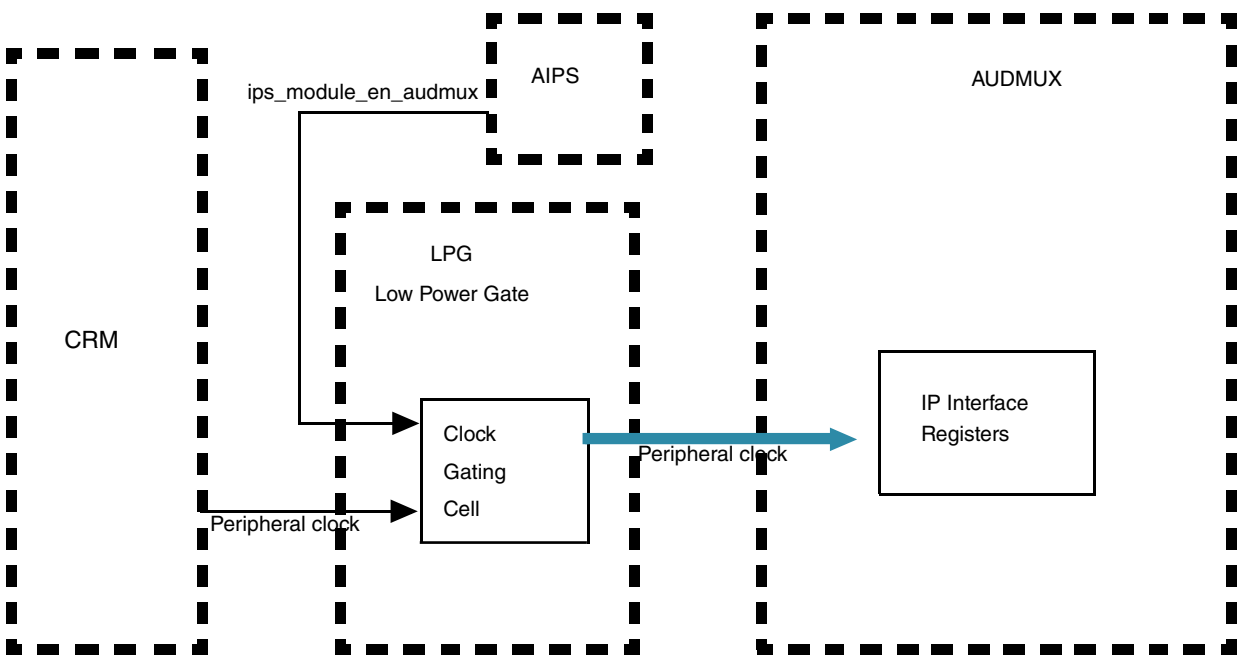


Figure 16-2. AUDMUX Clocking Scheme

16.3.3 Clocking Restrictions

- Since the AUDMUX requires only peripheral clock, it places no restrictions on the bus frequency.
- All registers in the AUDMUX are control registers so their values will not change frequently. These values will be programmed when changing between use cases (not during operation in a particular mode).

16.4 Default Register Configuration

There are two configuration registers for each port. Each pair of configuration registers is identical for each port; however, the default values following a reset differ as shown in the Memory Map.

Default Port Configuration, describes the default configuration of the ports.

16.4.1 Default Port Configuration

After a reset, each port defaults to normal mode (PDCR*n*[MODE] = 0) with synchronous timing mode (PTCR*n*[SYN] = 1) enabled.

The default port-to-port connections are as follows:

- Port 1 to Port 6
  - Port 6 provides the clock and frame sync.
- Port 2 to Port 5
  - Port 5 provides the clock and frame sync.
- Port 3 to Port 4
  - Port 4 provides the clock and frame sync.
- Port 7 to Port 7 (in data loopback mode)
  - Clock and frame syncs are inputs.

16.5 Functional Description

This section provides a complete functional description of the AUDMUX.

16.5.1 Operating Modes

This section describes all functional operation modes of the AUDMUX.

Figure 16-1 shows the AUDMUX block diagram.

All of the ports are essentially identical; there is no functional difference among Ports 1 through 7. The main difference is whether a port is hard wired to synchronous serial interface (SSI) or hard wired to the chip's pads. Each of the connection is hard wired to specific AUDMUX port. AUDMUX provides flexibility in routing the signal within the module, but all Internal and External port connections are fixed to specific configuration.

All ports can be configured as four- or six-wire interfaces. When configured as a six-wire interface, Receive Frame Sync (RXFS) and Receive Clock (RXC) signals of SSI interface enable the serial interface to be used in asynchronous mode with separate receive and transmit clocks.

AUDMUX supports both Normal mode (not to be confused with SSI’s Normal Mode), External Network mode and Internal Network mode. The definition of each mode will be given in the next section.

All ports have a TXRXEN bit to provide flexibility in supporting network mode configurations. The TXRXEN bit reverses the functions of transmit and receive data lines where the transmit line is configured as receive and transmit line to be configured as transmit line. This function is provided so that mastership of the serial bus can be passed among multiple external devices connected to a single port.

In addition to supporting the External Network mode (default), all ports support an internal network mode:

- With internal network mode, single point-to-multipoint network configuration with an arbitrary number of slaves can be supported if the external slaves are put into the high-impedance state (as defined in the SSI network mode protocol) and have pull-up resistors on their TxD pins. (Alternatively, this can be viewed as requiring a pull-up resistor on the corresponding AUDMUX RxD pin.)

Bit clock direction selection enables each port to be configured as a master or slave in the flow.

Possible scenarios include:

- SSI (hard wired to internal port) transmits data to a voice codec and a BT (Blue tooth) codec (both on external Port 4Port 5) and the Bottom Connector (on external Port 5Port 6) simultaneously using network mode. SSI is configured as the master.
- An external processor (external port - Port 4Port 3) drives a voice codec and a BT codec (both on external Port 5Port 4) and the Bottom Connector (on Port 6Port 5) simultaneously using network mode. The external processor is the master.

16.5.1.1 Port Receive Data Modes

Each port has logic to select which data lines are used to create the RXD line for the corresponding host interface.

Figure 16-3 shows the logic used to create the RXD line for Port 1. This logic has the following modes of operation (as determined by MODE:

- Normal (not to be confused with SSI's normal mode)
- Internal network mode

The subsequent sections describe the various modes of the port receive data logic. The following terms are used to define the operation of the AUDMUX:

- Network mode- Time-Division Multiplexed protocol for sending unique data to multiple devices on a serial bus or single devices with multi-channel capabilities.
- Internal network mode-Physical bus configuration where multiple serial buses are effectively connected within the AUDMUX via digital logic to create point-to multipoint connectivity. An arbitrary number of devices are supported. Devices must be put into the high-impedance state as specified by the network mode protocol.
- External network mode-Physical bus configuration where multiple serial buses are electrically connected together on a printed circuit board (that is, external to the AUDMUX). Devices must put their TXD lines into the high-impedance state as specified by the network mode protocol. TXD lines of devices must be pulled high.

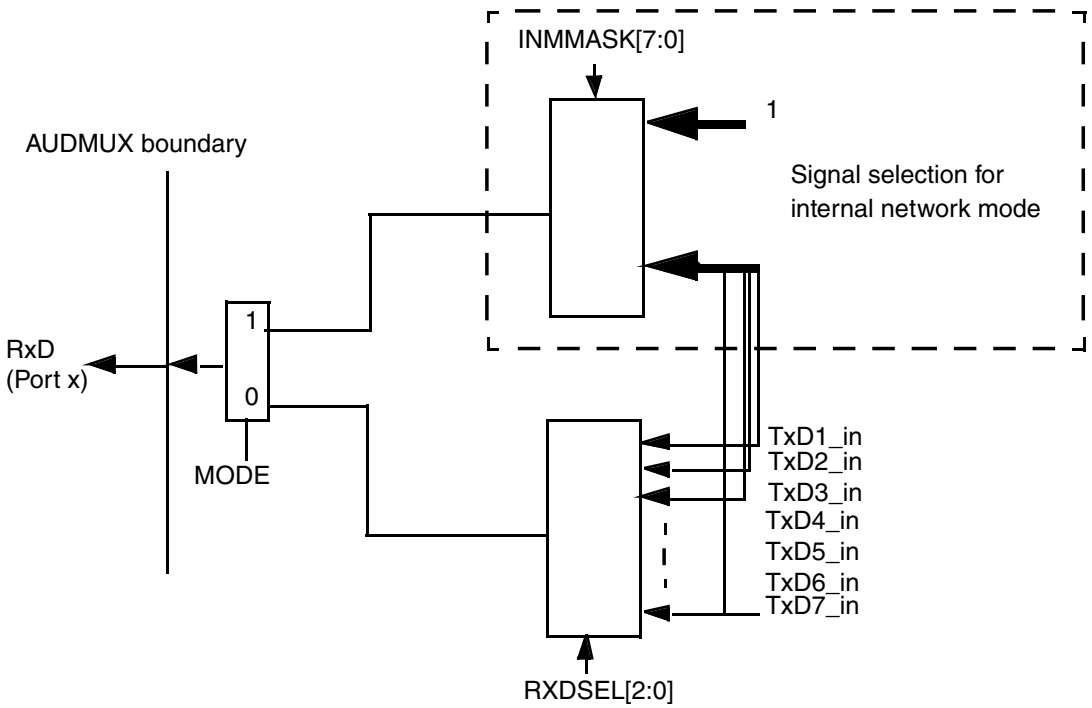


Figure 16-3. Receive Data Logic for Port x

16.5.1.1.1 Normal Mode

In normal mode (MODE = 0) , not to be confused with SSI's Normal Mode, the port is connected in a single device point-to-single device port configuration (as a master or a slave) and the RXDSEL[2:0] setting selects the transmit signal from any port. In normal mode, any data format can be used (that is, SSI normal mode, SSI network mode, AC-97, and others.

If a user wishes to transmit SSI1's data (TXD) to port 5, PDCR5's RXDSEL[2:0] must be set to 000b. If the clock (TCK) and frame clock (TFS) are sent out to the external device from SSI1, PTCR5's TFSEL[3:0] and TCSEL[3:0] must be set to 0000b (port1) and 0000b (port1), while setting PTCR5's TFSDIR and TCLKDIR to 1.

If either or both of the clocks are to be received from external device to SSI1, PTCR5's TCLKDIR and/or TFSDIR must be set to 0 (input), and PTCR1's TFSEL[3:0] and/or TCLKDIR[3:0] to 0100b (port5).

Likewise, if a user wishes to receive serial data from Port 4 and send the receiving data to SSI2's RXD, one must set PDCR2's RXDSEL[2:0] to 11b. If the frame (RXFS) and bit clocks (RXC) are to be received from the external device, PTCR2's RFSEL[3:0] and RCSEL[3:0] must be set to 011b while PTCR4's RFSDIR and RCLKDIR set to 0.

16.5.1.1.2 Internal Network Mode

In internal network mode (MODE = 1) , the output of the AND gate is routed (via the output of the port) to the RXD signal of the corresponding host interface.

The INMMASK bit vector selects the transmit signals of the ports that are to be connected in network mode. The transmit signals received at the AUDMUX ports (TxDn_in) are ANDed together to form the output. In internal network mode, only one device can be transmitting in its predesignated timeslot and all other transmit signals must remain high (be in high-impedance state and pulled-up). Therefore, non-active signals in the selection will be high and do not influence the output of the AND gate.

Network mode is a protocol where a master SSI is connected to more than one slave SSI device and communication occurs on a time-slotted frame. Though network mode can allow master-slave and slave-slave communication, internal network mode supports only master-slave communication.

There are two scenarios where internal network mode can be used with external network mode:

1. Slave-only devices are attached to an external port.
2. A master device is attached to an external port and all slave devices connected to the same external port are disabled.

NOTE

When internal network mode is enabled at an external port, RXDSEL[3:0] for RxDn_obe selection is ignored and RxD_obe is always driven high (that is, asserted for all timeslots). All slave devices connected to the same port must be disabled.

Internal Network Mode Example 1

SSI_m and SSI_n are used with Port 3 in internal network mode as shown in Figure 16-4. No pull-up resistors are required because the interfaces combined in internal network mode are on-chip interfaces.

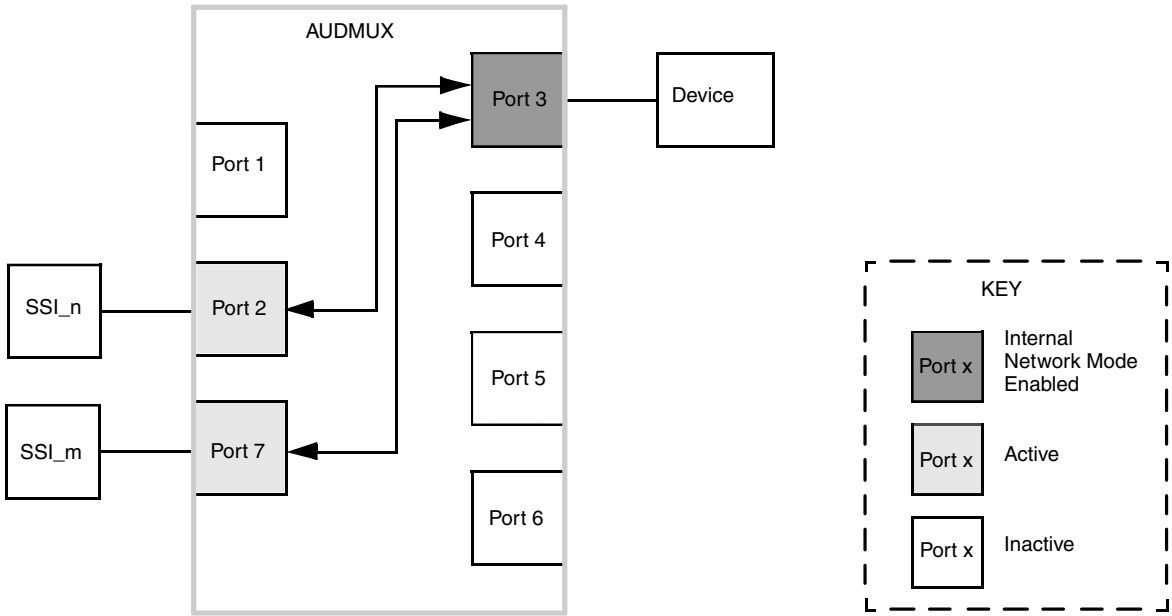


Figure 16-4. Block Diagram For Example 1

See Figure 16-5 for the timing diagram of Example 1. The clock and frame sync signals show the bit and frame timing for the serial bus. The vertical dashed lines divide the frame into four timeslots.

The data lines for SSI_m and SSI_n (as well as their output enables) are shown. Note that the on-chip interfaces drive a logic '1' when their output enables are logic '0'. The combined TXD line, which is the logical AND of the individual TXD lines, is used for Port 3's TXD line.

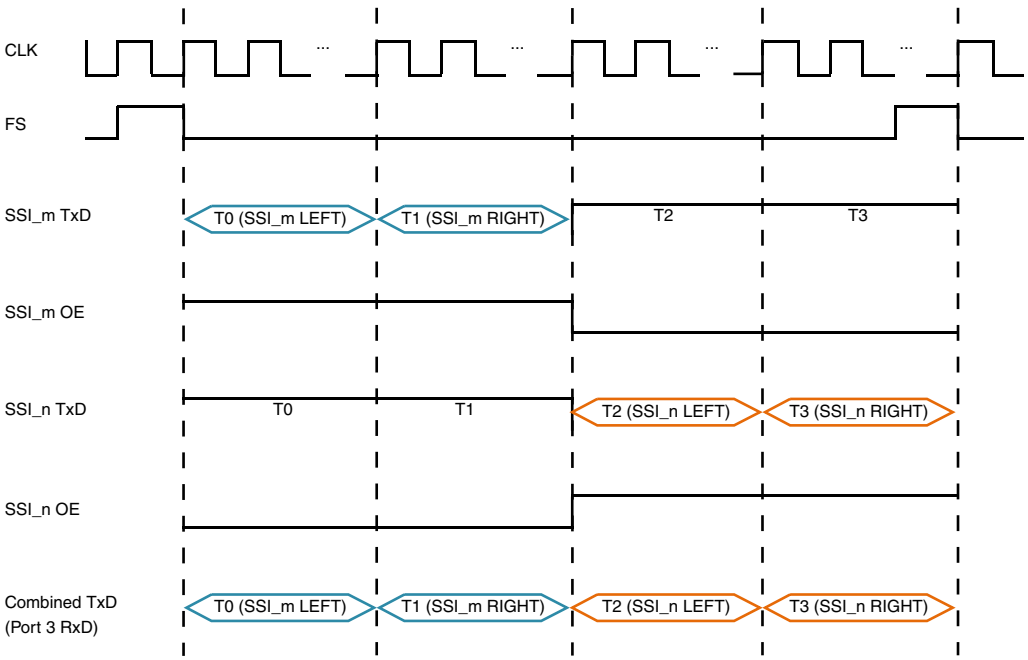


Figure 16-5. Example Using Internal Ports For Transmit Data

Internal Network Mode Example 2

The SSI, Port 3, and Port 4 are used with Port 5 in internal network mode, as shown in the following figure. Note that Port 3 and Port 4 are external ports. Therefore, pull-up resistors are required on the Port 3 RXD and Port 4 RXD pins. This example shows the timing associated with using adjacent timeslots for the SSI, Port 3 and Port 4 .

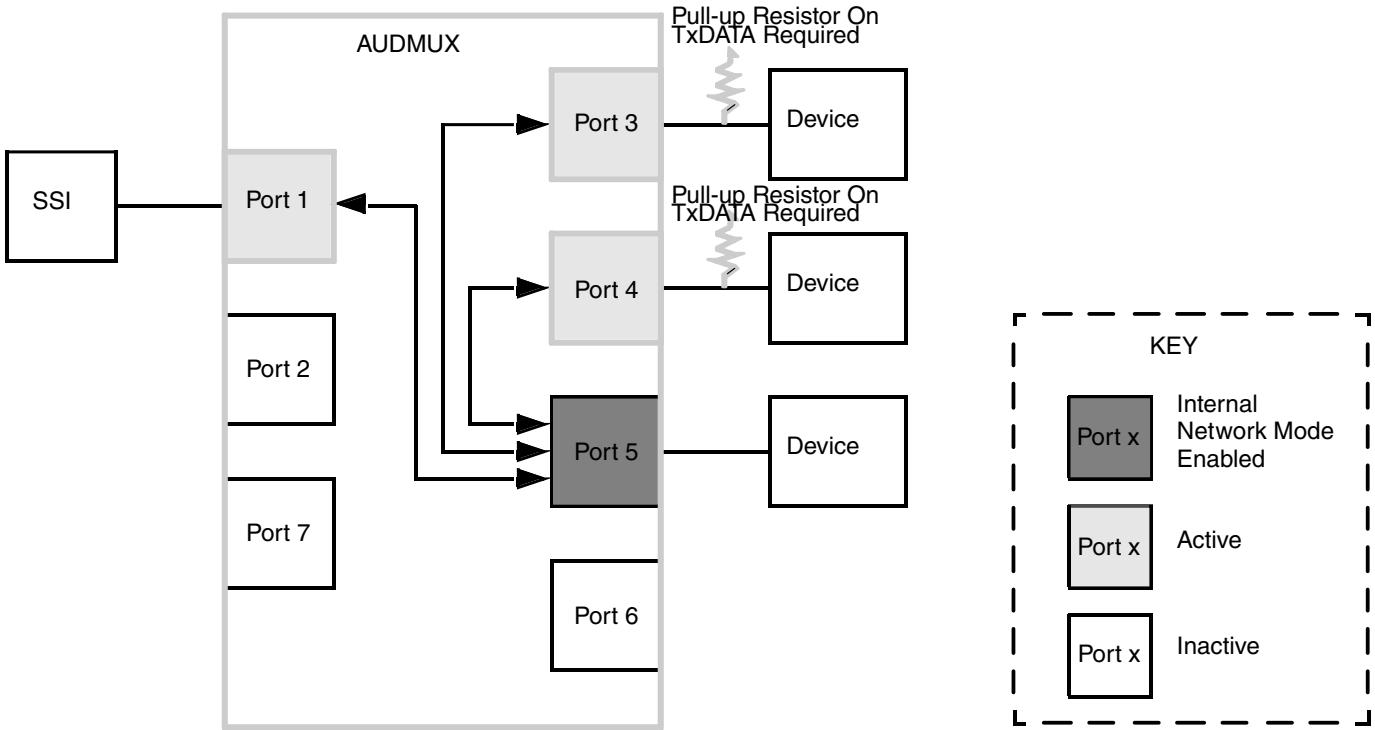


Figure 16-6. Block Diagram For Example 2

The resistance value of the pull-up resistors must be sufficiently high such that a value of '0' can be pulled up to logic '1' within half of a period of the bitclock. The required resistance must be no larger than:

Rmax = 1 / (2 * f_{bc} * C) where:

- f_{bc} is the frequency of the bitclock
- C is the total system capacitance (ICs, board traces, and so on)

The following figure shows the timing diagram for this example. The clock and frame sync signals show the bit and frame timing for the serial bus. The vertical dashed lines divide the frame into four timeslots.

The data lines for the SSI, Port 3 and Port 4 are shown. Note that the SSI transmits a logic '1' when its corresponding output enable is a logic '0'. The data lines from Port 3 and Port 4 at the pad are pulled high by pull-up resistors when they are in the high-impedance state. The data lines from Port 3 and Port 4 at the AUDMUX are pure digital signals and are constantly driven. The combined TXD line, which is the logical AND of the SSI, Port 3 and Port 4's TXD lines, is used for Port 5's TXD line.

Note the highlighted areas in the Figure 16-7. This shows the transition time that occurs while a TXD line is being pulled high. In this example, this transition time is a maximum of 1/2 the period of the serial bitclock. This prevents corruption of the first data bit of the next timeslot. It is critical that the pull-up resistance is sufficient for the given bitclock frequency and system capacitance.

Note that hysteresis should be enabled at Port 3's RXD pad and Port 4's RXD pad to prevent the digital signals created by the pad from toggling rapidly during the pull-up period. The pads typically require a transition within 25ns unless hysteresis is enabled. Instead of using hysteresis, one could select a pull-up resistor sufficiently high to pull-up the signal at the pad within 25 ns; however, that would result in a higher resistance value and higher current drain.

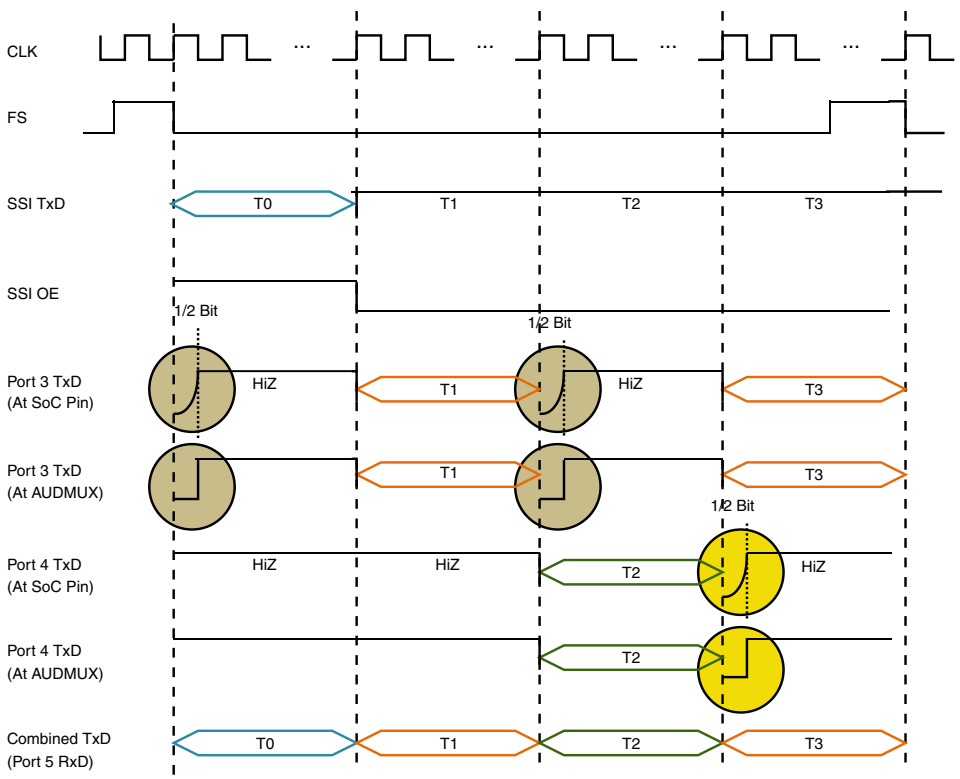


Figure 16-7. Example Using External Ports for Transmit Data in Consecutive Timeslots

Internal Network Mode Example 3

The SSI and Port 3 are used with Port 5 in internal network mode as shown in the following figure. Note that Port 3 is an external port. Therefore, a pull-up resistor is required on the Port 3TXD pin. This example shows the timing associated with inserting empty timeslots after the timeslots have been used by external ports.

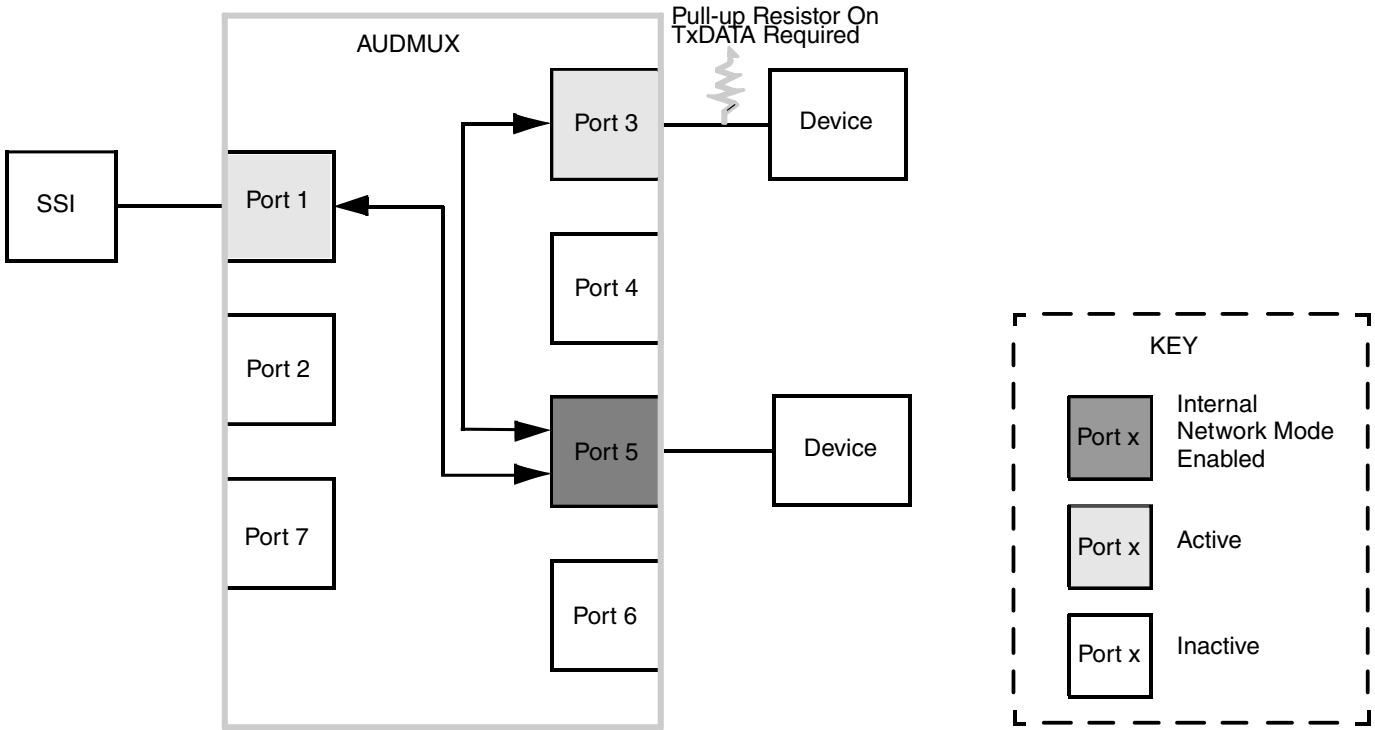


Figure 16-8. Block Diagram For Example 3

The resistance value of the pull-up resistors must be sufficiently high such that a value of '0' can be pulled up to logic '1' by the time that the next occupied timeslot occurs. This allows a much weaker pull-up to be used as compared to Example 2. The required resistance must be no larger than:

Rmax = (4 * n + 1) / (2* f_{bc} * C) where:

- n is the number of bits per timeslot
- f_{bc} is the frequency of the bitclock
- C is the total system capacitance (ICs, board traces, and so on)

The figure below shows the timing diagram for this example. The clock and frame sync signals show the bit and frame timing for the serial bus. The vertical dashed lines divide the frame into four timeslots.

The data lines for the SSI and Port 3 are shown. Note that the SSI transmits a logic '1' when its corresponding output enable is a logic '0'. The data line from Port 3 at the pad is pulled high by a pull-up resistor when they are in the high-impedance state. The data line from Port 3 at the AUDMUX is a pure digital signal and is constantly driven. The combined TXD line, which is the logical AND of the SSI and Port 3's TXD lines, is used for Port 5's RXD line.

Note the highlighted area in the Figure 16-9. This shows the transition time that occurs while Port 3's TXD line is being pulled high. In this example, this transition time is a maximum of two timeslots plus 1/2 the period of the serial bitclock. This prevents corruption of the first data bit of the next timeslot. It is critical that the pull-up resistance is sufficient for the given bitclock frequency and system capacitance.

Note that hysteresis must be enabled at Port 3's RXD pad to prevent the digital signal created by the pad from toggling rapidly during the extended pull-up period. The pads typically require a transition within 25 ns unless hysteresis is enabled.

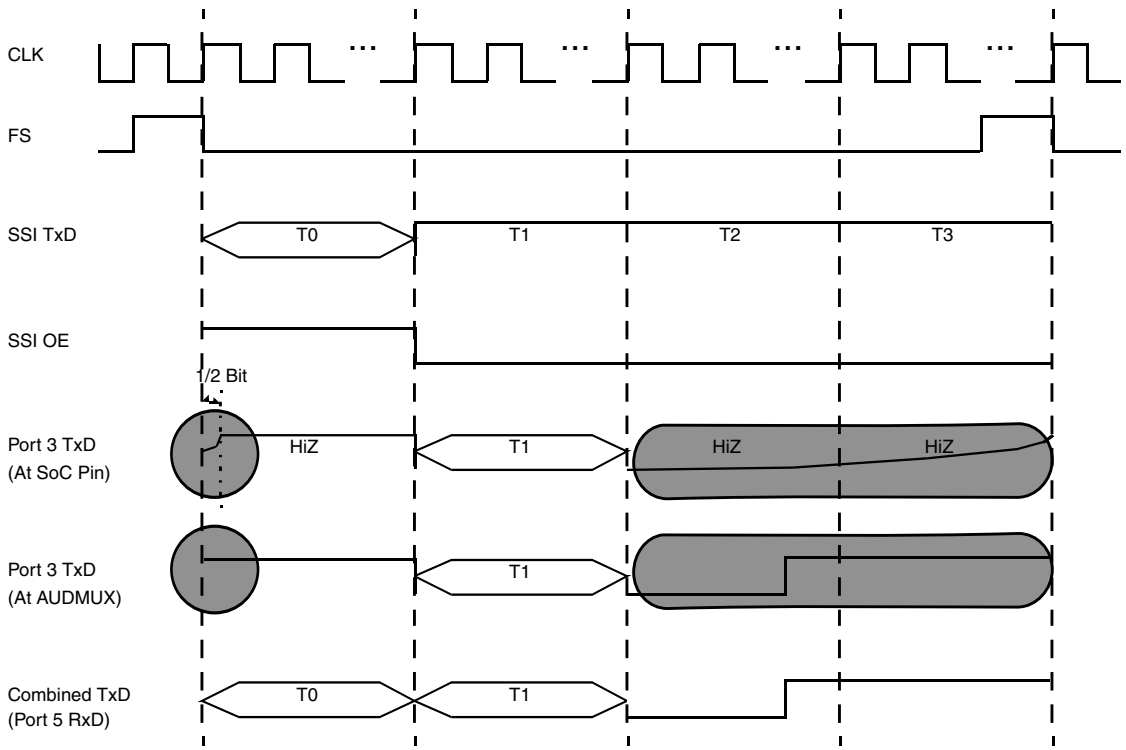


Figure 16-9. Example Using External Ports For Transmit Data In Nonconsecutive Timeslots

16.5.1.1.3 Transmit Data Output Enable Assertion

The TXD line from the internal network mode master (connected at any internal port) is put into the high-impedance state at the pad depending upon the assertion or deassertion of TxD_obe, its corresponding output enable generated by the network mode master.

In the case of an external network mode master (connected at an external port), the corresponding TxD_obe is always asserted after the port data register configuration.

16.5.1.2 Tx/Rx Switch and External Network Mode

External network mode is the traditional network mode connection. It is called external network mode to differentiate from the internal network mode. In external network mode, devices are connected to a single external port in a star or multi-drop configuration.

In network mode, there can be only one master (driving the frame sync and clock source) with the other devices configured in normal slave mode or network slave mode. Unlike internal network mode, both master-slave and slave-slave communication can take place in external network mode. Codec devices transmit on a single timeslot while processor serial interfaces (that is, SSI) can process more than one timeslot of data while in network master or slave mode.

The following figure shows the Tx/Rx data switch. RxD_obe is the output buffer enable signal and RxD_out is the data transmit signal from the serial interface. The TxD_in signal is the receive data signal going towards the RXDSEL muxes of all ports.

D_TxRx is the data pin which serves as the chip-level transmit data pin when the TxRx switch is not enabled. D_RxTx is the data pin which serves as the chip-level receive data pin when the TxRx switch is not enabled. The roles of these pins are reversed when the TxRx switch is enabled.

When TXRXEN is disabled (TXRXEN=0), RxD_out is routed to D_TxRx and D_RxTx is routed to TxD_in. The output buffer enable, selected by RXDSEL[2:0], is routed to Db_obe.

When the Tx/Rx switch is enabled (TXRXEN=1), RxD_out is routed to D_RxTx and D_TxRx is routed to TxD_in. The output buffer enable, selected by RXDSEL[2:0], is routed to Da_obe.

If the RXDSELn[2:0] field for any Port *n* is configured to select data from an internal port, the output buffer enable is selected by RXDSELn[2:0] and is routed to Da*n*_obe/ Db*n*_obe. In the case when the RXDSELn[2:0] field for Port *n* is configured to select data from an external port, the output buffer enable is always high and routed to Dan_obe/Dbn_obe, depending on the TXRXENn switch configuration.

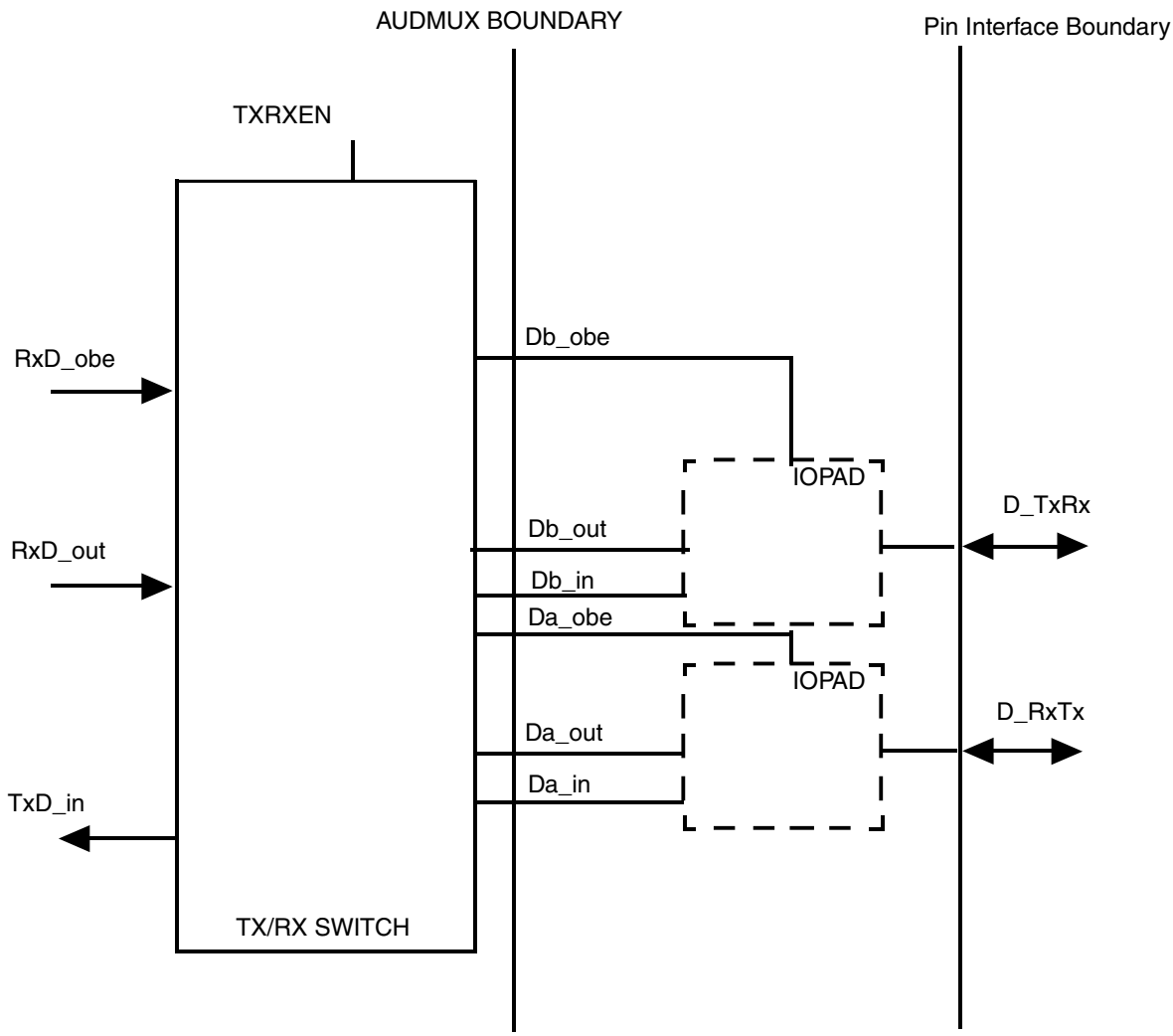


Figure 16-10. Tx/Rx Switch

16.5.1.3 Timing Modes

The AUDMUX ports are constructed as 6-wire interfaces. However, they can be used either in synchronous or asynchronous modes as determined by the SYN bit.

16.5.1.3.1 Synchronous Mode (4-Wire Interface)

In Synchronous mode, each port set in SYN=1 will be a 4-wire interface (that is, RXD, TXD, TXC, TXFS). In this setup, receive data timing is determined by TXC and TXFS..

As shown in the following figure, SSI signals interfaced to Port x signals are routed to Port y. When SYN = 1 the 6-wire signal from SSI is reduced to 4-wire within the internal logic.

TFS_in, RFS_in, TCLK_in, and RCLK_in are the input frame sync and bit clocks from the serial interface (Port x) with their corresponding output buffer enable signals (_obe). TFS_out, RFS_out, TCLK_out, and RCLK_out are the frame sync and bit clocks that are transmitted to the serial interface from the other ports.

The TFS_out and TCLK_out are selected at Port x by the TFSEL and TCSEL mux settings, respectively. RFS_out and RCLK_out are selected at Port x by the RFSEL and RCSEL mux settings, respectively. Similarly, in the external direction, Port y is configured as a 4-wire port; TFSEL selects the FS_obe and FS_out signals. In this mode, the configuration of RFSEL and RCSEL is not used, since the RFS_out and RCLK_out pins at Port y are not available.

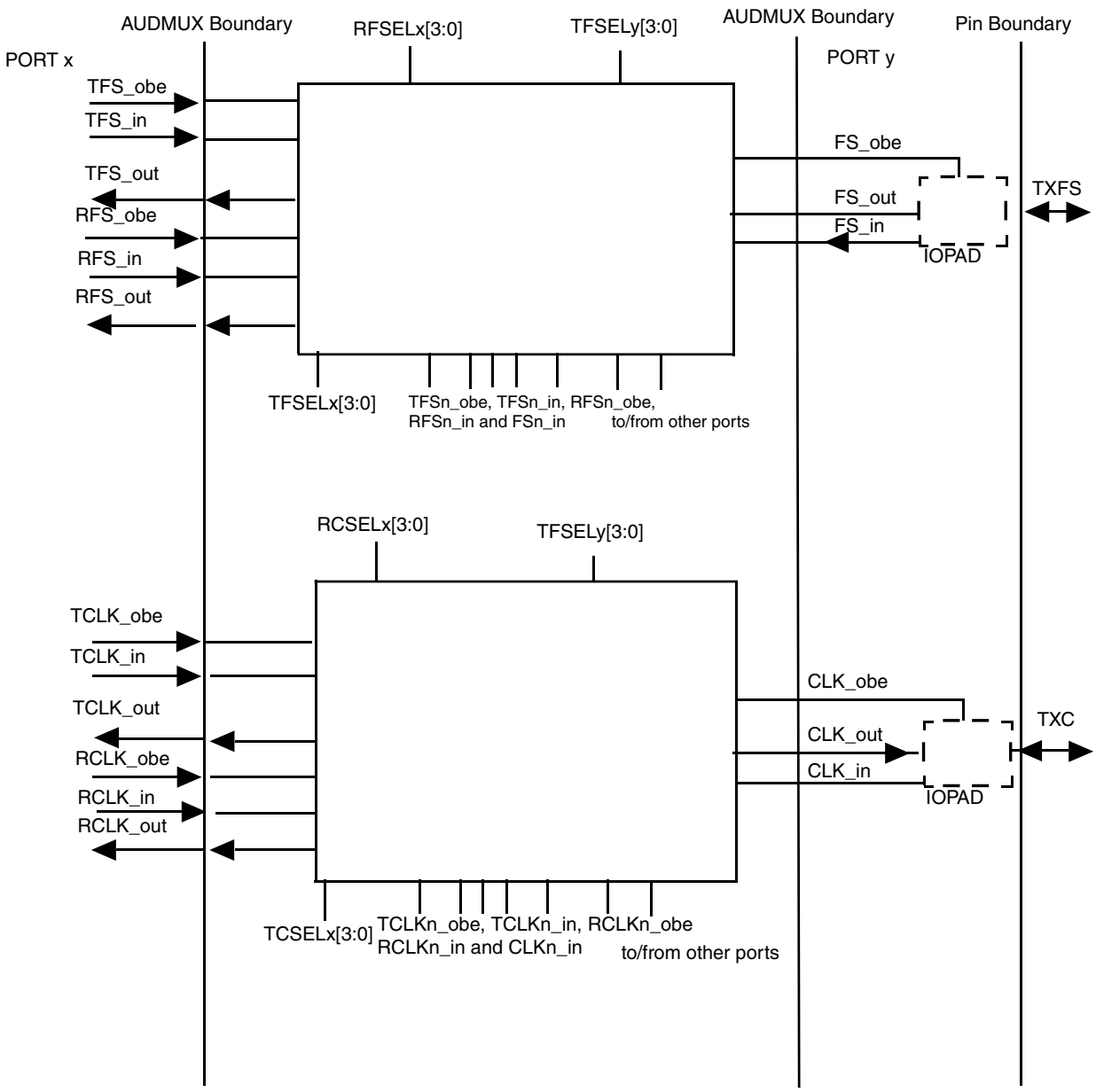


Figure 16-11. Frame Sync and Clock Routing When External Port Is 4-Wire

16.5.1.3.2 Asynchronous Mode (6-Wire Interface)

In Asynchronous mode, the port has a 6-wire interface (meaning RXD, TXD, TXC, TXFS, RXC, RXFS). This mode has additional receive clock (RXC) and frame sync (RXFS) signals as compared to the synchronous or 4-wire interface.

As shown in the figures below, Port x signals can be routed to Port y, producing 6-wire to 6-wire port connectivity.

TFS_in, RFS_in, TCLK_in, and RCLK_in are input frame sync and bit clocks from the serial interface (Port x) with their corresponding output buffer enable signals (_obe). TFS_out, RFS_out, TCLK_out, and RCLK_out are the frame sync and bit clocks that are transmitted to the serial interface from the other ports.

TFS_out and TCLK_out are selected by the TFSEL and TCSEL mux settings, respectively. RFS_out and RCLK_out are selected by the RFSEL and RCSEL mux settings, respectively. Similarly, in the external direction, the TFSEL selects the TxFS_obe and TxFS_out signals and TCSEL selects the TxCLK_obe and TxClk_out signals. The RFSEL selects the RxFS_obe and RxFS_out signals and RCSEL selects the RxCLK_obe and RxCLK_out signals.

NOTE

Because FS_in and CLK_in from external interfaces are also routed to the TFSEL and TCSEL muxes of the external ports, these signals do not have corresponding buffer enable signals. Consequently, their corresponding inputs to the TFSEL and TCSEL mux of the external ports have to be tied high.

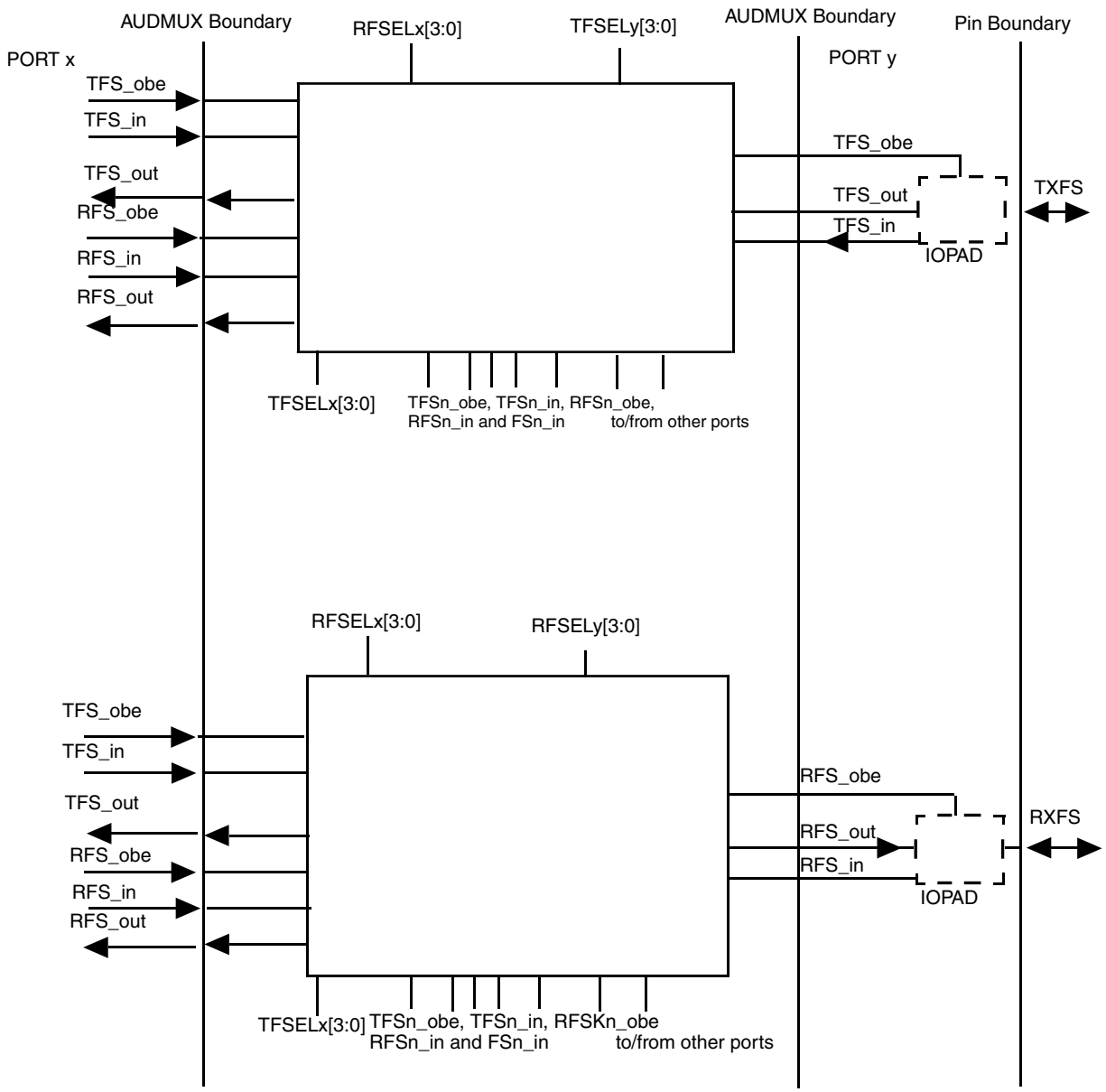


Figure 16-12. Frame Sync Routing When External Port Is 6-Wire

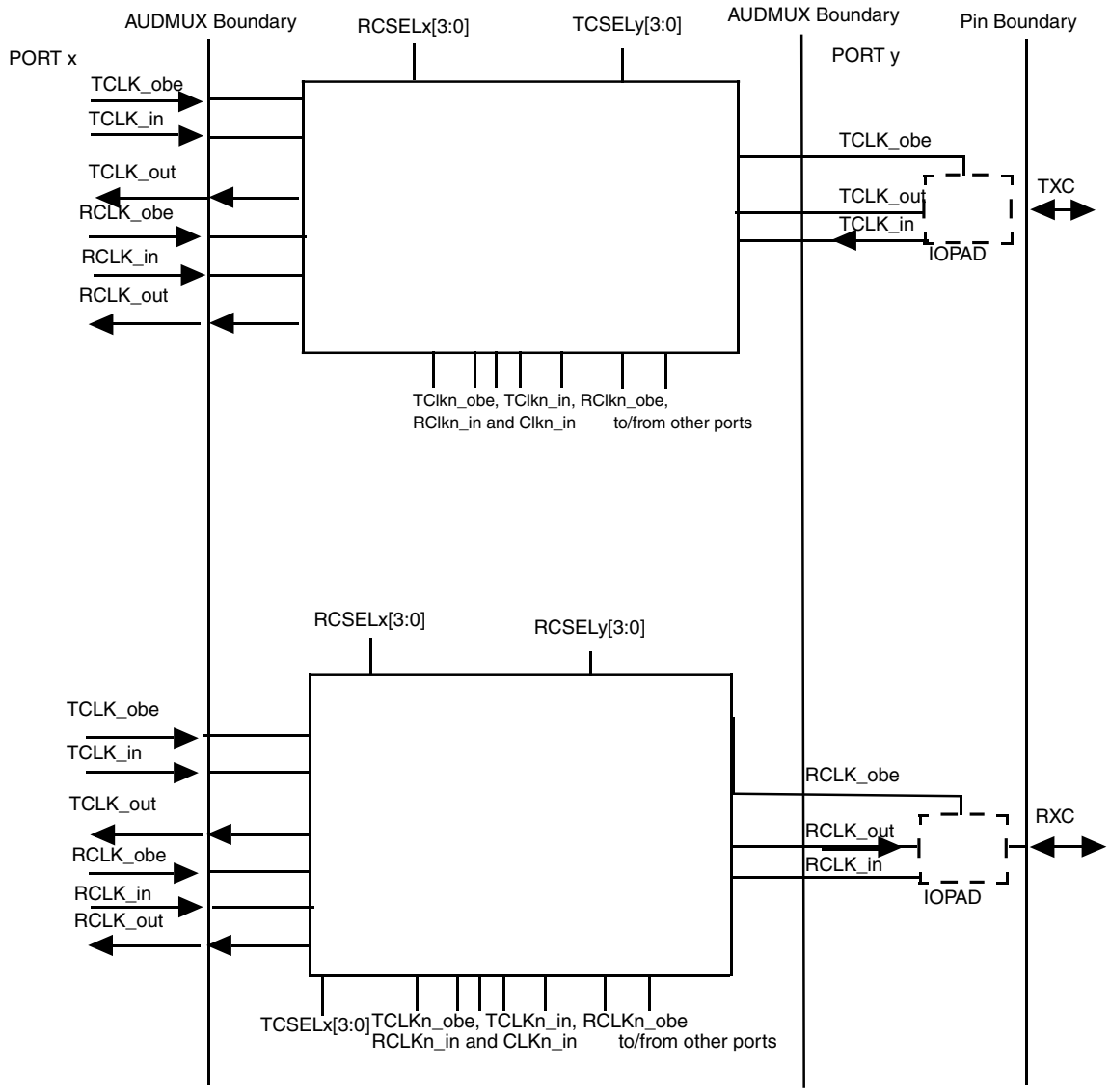


Figure 16-13. Clock Routing When External Port Is 6 Wire

16.5.2 Connectivity Between Ports

Four basic types of connections are provided by the AUDMUX:

- Internal port to external port
- External port to external port
- Internal port to internal port
- Loopback

The corresponding data connections are described in the following sections.

16.5.2.1 Internal Port to External Port Connectivity

The internal port is connected to a processor's serial interface. TxD_obe is the buffer enable signal from the serial interface, TxD_in is the input transmit data from the serial interface to the AUDMUX, and RxD_out is the receive data output from the AUDMUX to the serial interface.

RXDSEL[2:0] of the external port selects the buffer enable signal (TxD_obe) and transmit data output (TxD_out) signal from the TxD_obe and RxD_in signals. RXDSEL[2:0] is a common signal to both selection muxes.

NOTE

Because buffer TxD_in signals from external interfaces do not have corresponding buffer enable signals, their buffer enable signals into the selection mux are tied high. This will ensure that selection of TxD_in, as RxD_out will also drive the RxD_obe output high.

Transmit Data from the serial interface goes into the RXDSEL data mux and comes out as RxD_out. RxD_out is routed to Da_TxRx when TXRXEN is disabled and to D_RxTx when TXRXEN is enabled. Similarly, D_RxTx is routed to TxD_in when TXRXEN is disabled and D_TxRx is routed to TxD_in when TXRXEN is enabled. The routing of frame syncs is shown in Figure 16-12and the routing of interface clocks is shown in Figure 16-10.

If internal network mode is disabled, then RXDSEL selects the TxD_in, which is sent from the AUDMUX to the serial interface connected at Port x. When the internal network mode is selected, RxD_out is constructed by ANDing selected TxD_in signals from the ports (as determined by INMMASK).

If there is more than one device attached to the external port at D_TxRx and D_RxTx and one of the devices is a network master, then two conditions must be noted:

1. When the external master is enabled in network mode, then the serial interface at Port x must be configured as a slave (normal or network mode). No Tx/Rx switching is required.
2. When the external master is disabled and the serial interface at Port x and other slave devices must communicate, then the serial interface at Port x must be configured as a network mode master and the Tx/Rx switch at Port y must be enabled (TXRXEN=1). This will ensure that the transmit and receive paths are connected appropriately.

To communicate with more than one port, internal network mode can be enabled at Port x. In internal network mode, it is possible to communicate with any device attached to the other ports. Internal network mode shall be enabled at the port that is the SSI network mode master.

16.5.2.2 External Port to External Port Connectivity

External ports can communicate with external ports directly.

External ports can communicate together in three ways:

1. Each port's receive logic is configured in normal mode (MODE = 0) . Each port's RXDSEL[2:0] field is configured to select the other port's transmit data. Bit fields associated with clock/frame sync selection and direction are configured for each port. Either port can be the master.
2. One port is configured in internal network mode (MODE = 1) . All desired data lines are combined by the AND gate as determined by INMMASK[7:0]. Since an external port is being used as the internal network mode master, all other devices on the same AUDMUX port as the internal network mode master must be disabled. This configuration can be used with a combination of internal and external ports. All external ports must have a pull-up resistor on its RXD pin. Bit fields associated with clock/frame sync selection and direction are configured for each port. Any port can be the master.

16.5.2.3 Internal Port to Internal Port Connectivity

Internal ports can communicate with other internal ports directly, thereby providing a means for synchronous interprocessor communication.

Internal ports can communicate together in two ways:

1. Each port's receive logic is configured in normal mode (MODE = 0) . Each port's RXDSEL[2:0] field is configured to select the other port's transmit data. Bit fields associated with clock/frame sync selection and direction are configured for each port. Either port can be the master.
2. One port is configured in internal network mode (MODE = 1) . All desired data lines are combined by the AND gate as determined by INMMASK[7:0]. This configuration can be used with a combination of internal and external ports. All external ports must have a pull-up resistor on its RXD pin. Bit fields associated with clock/frame sync selection and direction are configured for each port. Any port can be the master.

16.5.2.4 Loopback Connectivity

AUDMUX ports can communicate with themselves in order to provide loopback functionality. Port x can route its TXD signal to its own RxD_out signal by setting RXDSELx[2:0] to its own port number. This is supported by all ports in the AUDMUX.

In addition, ports can provide loopback support in internal network mode. With internal network mode, the internal network mode master can loop its TXD signal (combined with those of other ports, if desired) back into its RxD_out signal. Port x's INMMASK should be set such that bit (x - 1) is clear in order to enable the loopback.

16.6 AUDMUX Memory Map/Register Definition

This section includes the block memory map and detailed descriptions of all registers. For the base address of a specific sub-block instantiation, see the system memory map in this manual.

The AUDMUX memory map is shown in the following table.

AUDMUX memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21D_8000	Port Timing Control Register 1 (AUDMUX_PTCR1)	32	R/W	AD40_0800h	16.6.1/715
21D_8004	Port Data Control Register 1 (AUDMUX_PDCR1)	32	R/W	0000_A000h	16.6.2/717
21D_8008	Port Timing Control Register 2 (AUDMUX_PTCR2)	32	R/W	A500_0800h	16.6.3/718
21D_800C	Port Data Control Register 2 (AUDMUX_PDCR2)	32	R/W	0000_8000h	16.6.4/720
21D_8010	Port Timing Control Register 3 (AUDMUX_PTCR3)	32	R/W	9CC0_0800h	16.6.5/721
21D_8014	Port Data Control Register 3 (AUDMUX_PDCR3)	32	R/W	0000_6000h	16.6.6/723
21D_8018	Port Timing Control Register 4 (AUDMUX_PTCR4)	32	R/W	0000_0800h	16.6.7/724
21D_801C	Port Data Control Register 4 (AUDMUX_PDCR4)	32	R/W	0000_4000h	16.6.8/726
21D_8020	Port Timing Control Register 5 (AUDMUX_PTCR5)	32	R/W	0000_0800h	16.6.9/727
21D_8024	Port Data Control Register 5 (AUDMUX_PDCR5)	32	R/W	0000_2000h	16.6.10/ 729
21D_8028	Port Timing Control Register 6 (AUDMUX_PTCR6)	32	R/W	0000_0800h	16.6.11/ 730
21D_802C	Port Data Control Register 6 (AUDMUX_PDCR6)	32	R/W	0000_0000h	16.6.12/ 732
21D_8030	Port Timing Control Register 7 (AUDMUX_PTCR7)	32	R/W	0000_0800h	16.6.13/ 733

Table continues on the next page...

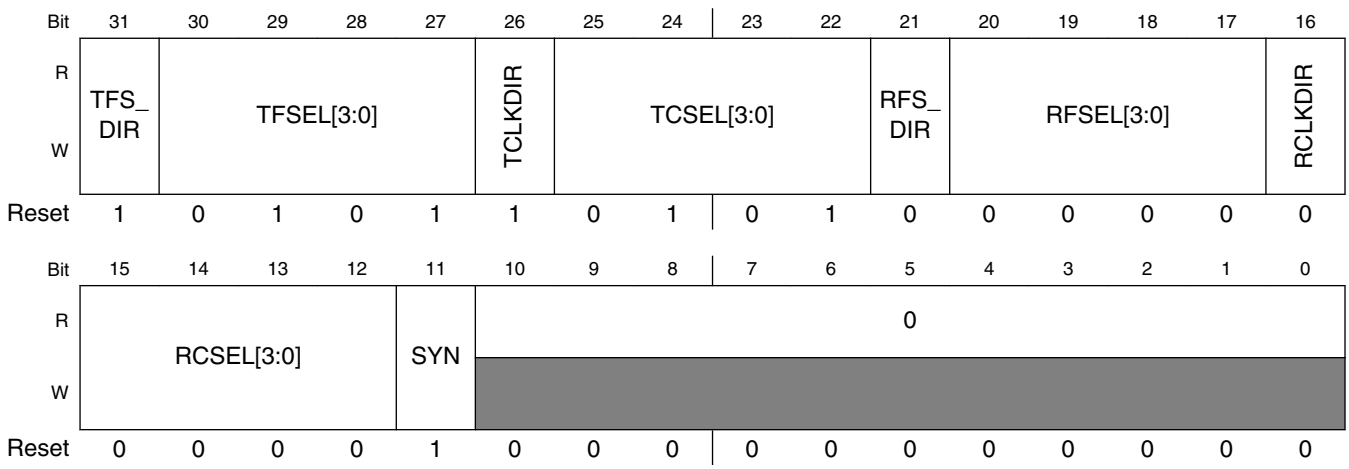
AUDMUX memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21D_8034	Port Data Control Register 7 (AUDMUX_PDCR7)	32	R/W	0000_C000h	16.6.14/ 735

16.6.1 Port Timing Control Register 1 (AUDMUX_PTCR1)

PTCR1 is the Port Timing Control Register for Port 1.

Address: 21D_8000h base + 0h offset = 21D_8000h



AUDMUX_PTCR1 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock.

Table continues on the next page...

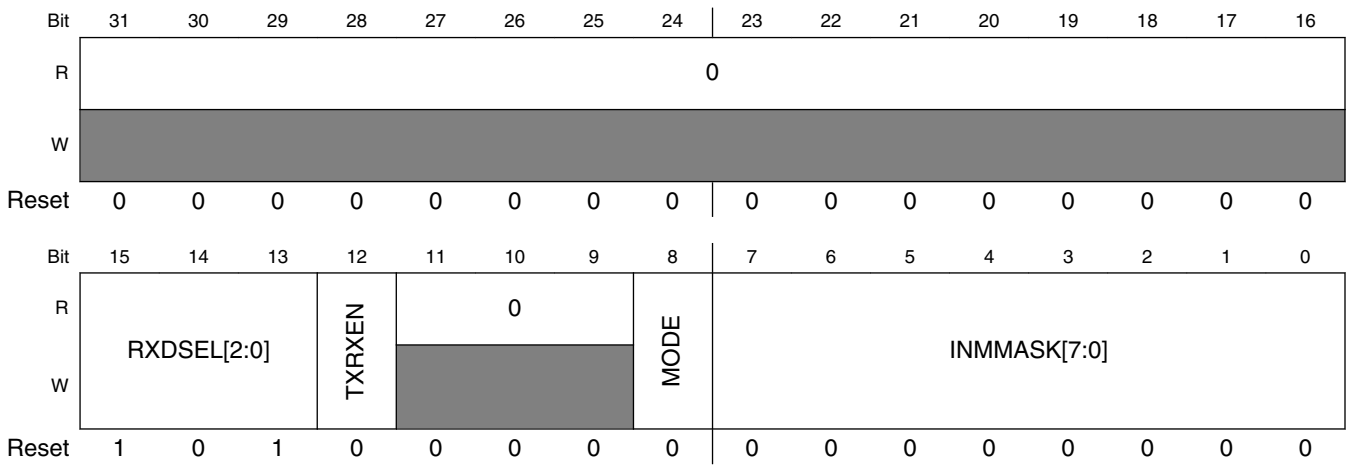
AUDMUX_PTCR1 field descriptions (continued)

Field	Description
	0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.2 Port Data Control Register 1 (AUDMUX_PDCR1)

PDCR1 is the Port Data Control Register for Port 1.

Address: 21D_8000h base + 4h offset = 21D_8004h



AUDMUX_PDCR1 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled). xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode

Table continues on the next page...

AUDMUX_PDCR1 field descriptions (continued)

Field	Description
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.3 Port Timing Control Register 2 (AUDMUX_PTCR2)

PTCR2 is the Port Timing Control Register for Port 2.

Address: 21D_8000h base + 8h offset = 21D_8008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RCSEL[3:0]				SYN	0										
W																
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR2 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved

Table continues on the next page...

AUDMUX_PTCR2 field descriptions (continued)

Field	Description
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time.

Table continues on the next page...

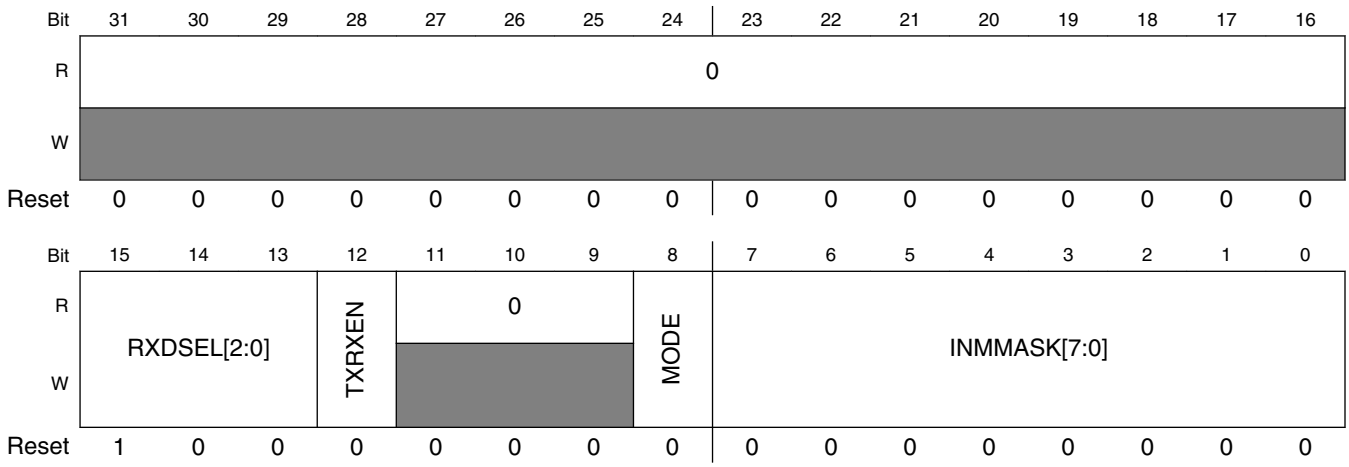
AUDMUX_PTCR2 field descriptions (continued)

Field	Description
	0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.4 Port Data Control Register 2 (AUDMUX_PDCR2)

PDCR2 is the Port Data Control Register for Port 2.

Address: 21D_8000h base + Ch offset = 21D_800Ch



AUDMUX_PDCR2 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled). xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

AUDMUX_PDCR2 field descriptions (continued)

Field	Description
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.5 Port Timing Control Register 3 (AUDMUX_PTCR3)

PTCR3 is the Port Timing Control Register for Port 3.

Address: 21D_8000h base + 10h offset = 21D_8010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	1	0	0	1	1	1	0	0	1	1	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RCSEL[3:0]				SYN	0										
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR3 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.

Table continues on the next page...

AUDMUX_PTCR3 field descriptions (continued)

Field	Description
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved

Table continues on the next page...

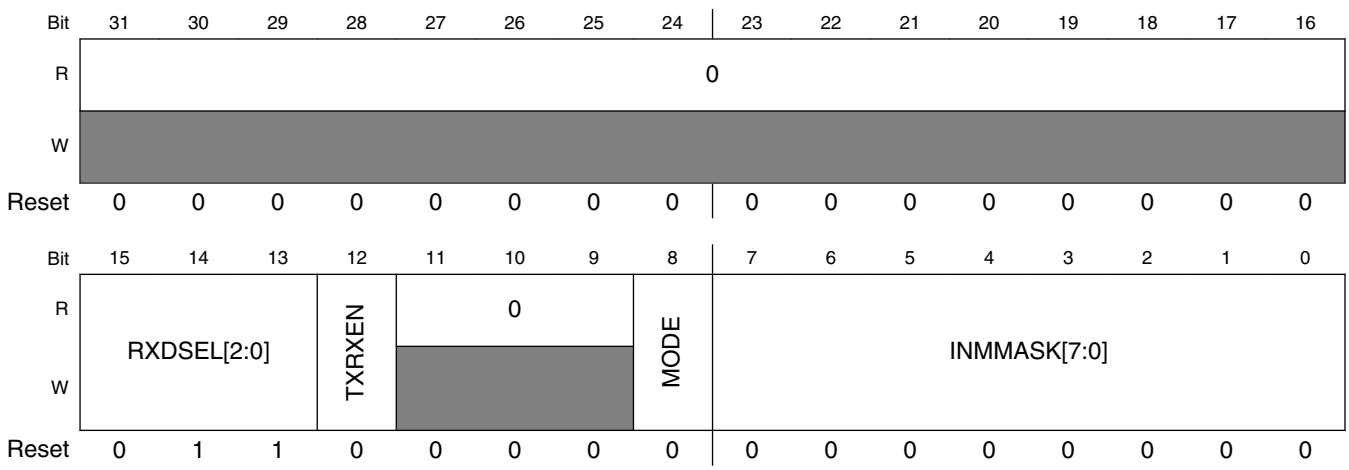
AUDMUX_PTCR3 field descriptions (continued)

Field	Description
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.6 Port Data Control Register 3 (AUDMUX_PDCR3)

PDCR3 is the Port Data Control Register for Port 3.

Address: 21D_8000h base + 14h offset = 21D_8014h



AUDMUX_PDCR3 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled). xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved

Table continues on the next page...

AUDMUX_PDCR3 field descriptions (continued)

Field	Description
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.7 Port Timing Control Register 4 (AUDMUX_PTCR4)

Port Timing Control Register for Port 4

Address: 21D_8000h base + 18h offset = 21D_8018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RCSEL[3:0]				SYN	0										
W																
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR4 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports.

Table continues on the next page...

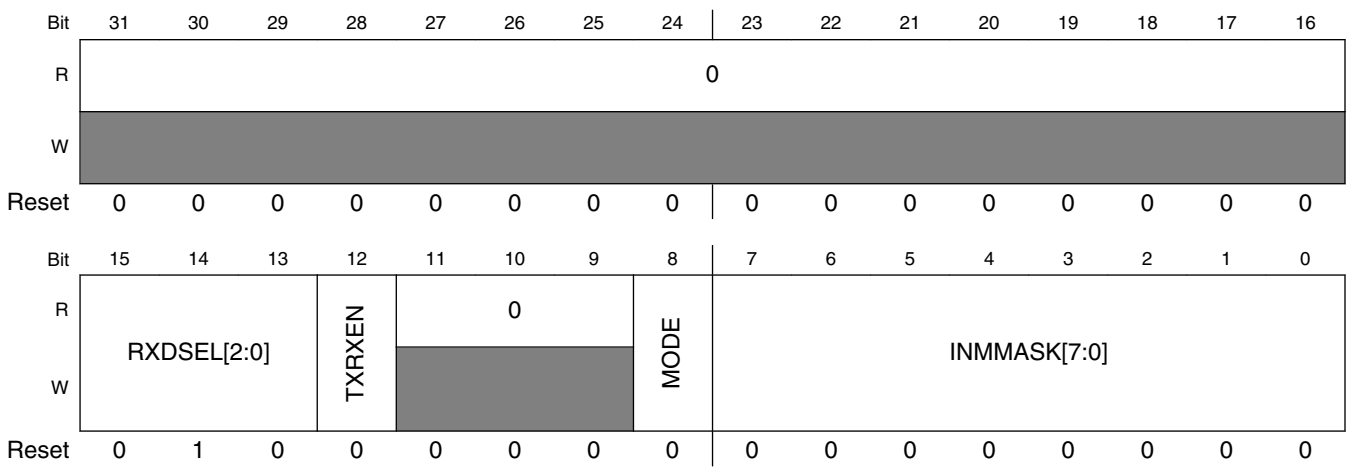
AUDMUX_PTCR4 field descriptions (continued)

Field	Description
	0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.8 Port Data Control Register 4 (AUDMUX_PDCR4)

PDCR4 is the Port Data Control Register for Port 4.

Address: 21D_8000h base + 1Ch offset = 21D_801Ch



AUDMUX_PDCR4 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled).

Table continues on the next page...

AUDMUX_PDCR4 field descriptions (continued)

Field	Description
	xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.9 Port Timing Control Register 5 (AUDMUX_PTCR5)

Port Timing Control Register for Port 5

Address: 21D_8000h base + 20h offset = 21D_8020h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RCSEL[3:0]				SYN	0										
W																
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR5 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports.

Table continues on the next page...

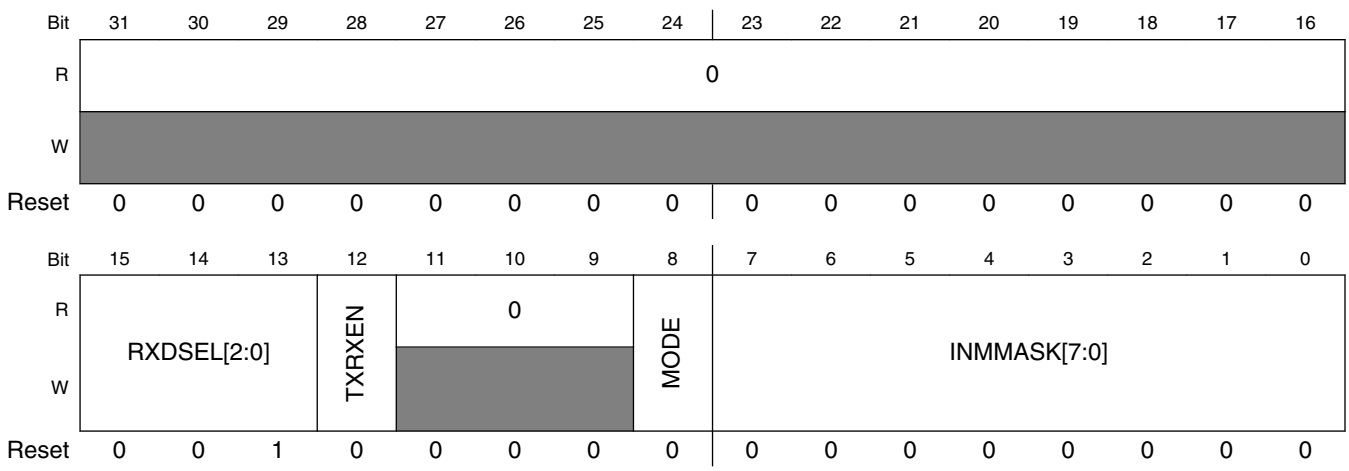
AUDMUX_PTCR5 field descriptions (continued)

Field	Description
	0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.10 Port Data Control Register 5 (AUDMUX_PDCR5)

PDCR5 is the Port Data Control Register for Port 5.

Address: 21D_8000h base + 24h offset = 21D_8024h



AUDMUX_PDCR5 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled).

Table continues on the next page...

AUDMUX_PDCR5 field descriptions (continued)

Field	Description
	xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.11 Port Timing Control Register 6 (AUDMUX_PTCR6)

Port Timing Control Register for Port 6

Address: 21D_8000h base + 28h offset = 21D_8028h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RCSEL[3:0]				SYN	0										
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR6 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports.

Table continues on the next page...

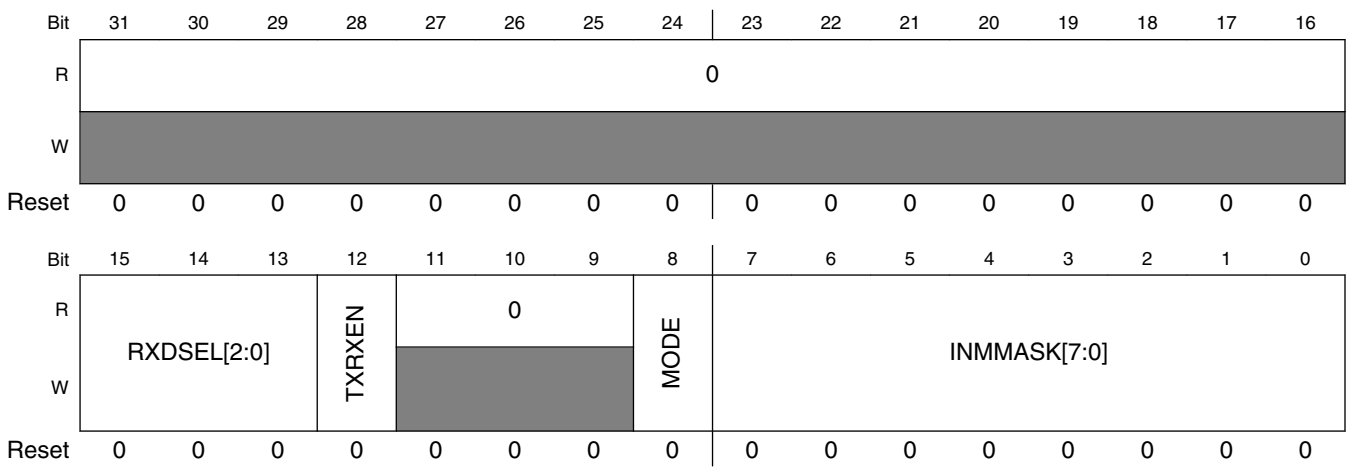
AUDMUX_PTCR6 field descriptions (continued)

Field	Description
	0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.12 Port Data Control Register 6 (AUDMUX_PDCR6)

PDCR6 is the Port Data Control Register for Port 6.

Address: 21D_8000h base + 2Ch offset = 21D_802Ch



AUDMUX_PDCR6 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled).

Table continues on the next page...

AUDMUX_PDCR6 field descriptions (continued)

Field	Description
	xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

16.6.13 Port Timing Control Register 7 (AUDMUX_PTCR7)

Port Timing Control Register for Port 7

Address: 21D_8000h base + 30h offset = 21D_8030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TFS_DIR	TFSEL[3:0]				TCLKDIR	TCSEL[3:0]				RFS_DIR	RFSEL[3:0]				RCLKDIR
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RCSEL[3:0]				SYN	0										
W																
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

AUDMUX_PTCR7 field descriptions

Field	Description
31 TFS_DIR	Transmit Frame Sync Direction Control. This bit sets the direction of the TXFS pin of the interface as an output or input. When set as an input, the TFSEL settings are ignored. When set as an output, the TFSEL settings determine the source port of the frame sync. 0 TXFS is an input. 1 TXFS is an output.
30–27 TFSEL[3:0]	Transmit Frame Sync Select. Selects the source port from which TXFS is sourced. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
26 TCLKDIR	Transmit Clock Direction Control. This bit sets the direction of the TXC pin of the interface as an output or input. When set as an input, the TCSEL settings are ignored. When set as an output, the TCSEL settings determine the source port of the clock. 0 TXC is an input. 1 TXC is an output.
25–22 TCSEL[3:0]	Transmit Clock Select. Selects the source port from which TXC is sourced. 0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
21 RFS_DIR	Receive Frame Sync Direction Control. This bit sets the direction of the RXFS pin of the interface as an output or input. When set as an input, the RFSEL settings are ignored. When set as an output, the RFSEL settings determine the source port of the frame sync. 0 RXFS is an input. 1 RXFS is an output.
20–17 RFSEL[3:0]	Receive Frame Sync Select. Selects the source port from which RXFS is sourced. RXFS can be sourced from TXFS and RXFS from other ports. 0xxx Selects TXFS from port. 1xxx Selects RXFS from port. x000 Port 1 x110 Port 7 x111 Reserved
16 RCLKDIR	Receive Clock Direction Control. This bit sets the direction of the RXC pin of the interface as an output or input. When set as an input, the RCSEL settings are ignored. When set as an output, the RCSEL settings determine the source port of the clock. **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 RXC is an input. 1 RXC is an output.
15–12 RCSEL[3:0]	Receive Clock Select. Selects the source port from which RXC is sourced. RXC can be sourced from TXC and RXC from other ports.

Table continues on the next page...

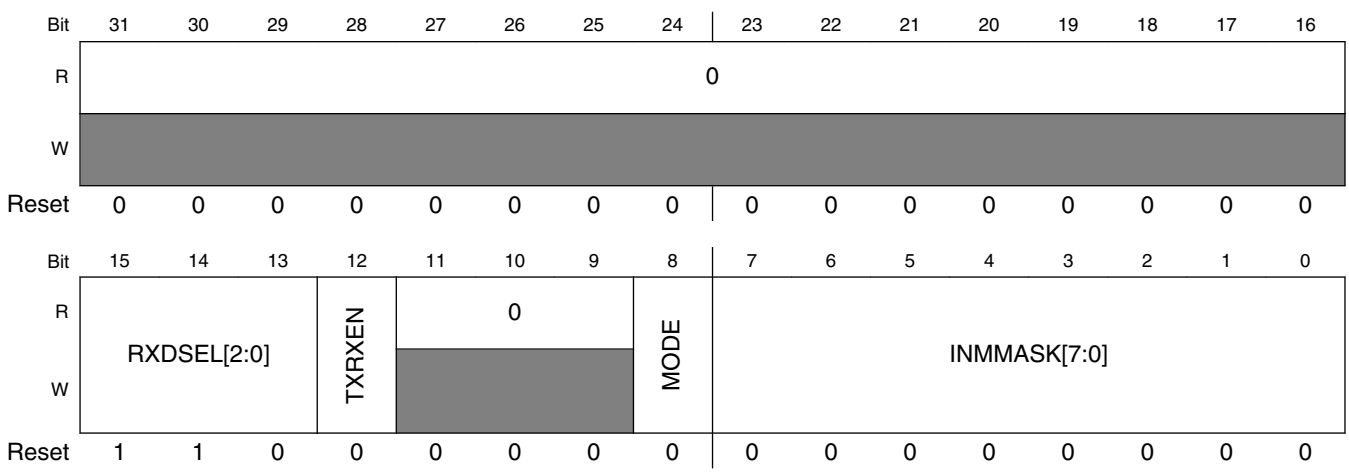
AUDMUX_PTCR7 field descriptions (continued)

Field	Description
	0xxx Selects TXC from port. 1xxx Selects RXC from port. x000 Port 1 x110 Port 7 x111 Reserved
11 SYN	Synchronous/Asynchronous Select. When SYN is set, synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals (that is, the port is a 4-wire interface). When SYN is cleared, asynchronous mode is chosen and separate clock and frame sync signals are used for the transmit and receive sections (that is, the port is a 6-wire interface). **NOTE:** RCLKDIR and SYN should not be changed at the same time. 0 Asynchronous mode 1 Synchronous mode (default)
Reserved	This read-only field is reserved and always has the value 0.

16.6.14 Port Data Control Register 7 (AUDMUX_PDCR7)

PDCR7 is the Port Data Control Register for Port 7.

Address: 21D_8000h base + 34h offset = 21D_8034h



AUDMUX_PDCR7 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 RXDSEL[2:0]	Receive Data Select. Selects the source port for the RXD data. RXDSEL is ignored if MODE = 1 (that is, Internal Network Mode is enabled).

Table continues on the next page...

AUDMUX_PDCR7 field descriptions (continued)

Field	Description
	xxx Port number for RXD 000 Port 1 110 Port 7 111 Reserved
12 TXRXEN	Transmit/Receive Switch Enable. Swaps the transmit and receive signals. 0 No switch (Transmit Pin = Transmit, Receive Pin = Receive) 1 Switch (Transmit Pin = Receive, Receive Pin = Transmit)
11–9 Reserved	This read-only field is reserved and always has the value 0.
8 MODE	Mode Select. This field selects the mode in which the port is to operate. The modes of operation include the following: • Normal mode, in which the RXD from the port selected by RXDSEL is routed to the port. • Internal Network mode in which RXD from other ports are ANDed together. RXDSEL is ignored. INMMASK determines which RXD signals are ANDed together. 0 Normal mode 1 Internal Network mode
INMMASK[7:0]	Internal Network Mode Mask. Bit mask that selects the ports from which the RXD signals are to be ANDed together for internal network mode. Bit 6 represents RXD from Port 7 and bit0 represents RXD from Port 1. 0 Includes RXDn for ANDing 1 Excludes RXDn from ANDing

Chapter 17
40-BIT Correcting ECC Accelerator (BCH)

17.1 Overview

The hardware ECC accelerator provides a forward error-correction function for improving the reliability of various storage media that may be attached to the device.

For example, NAND flash devices use a spare area to store ecc codes to correct some hard bit errors in data stored within the device, allowing higher device yields and, therefore, lower NAND device costs.

The Bose, Ray-Chaudhuri, Hocquenghem (BCH) Encoder and Decoder module is capable of correcting from 2 to 40 single bit errors within a block of data no larger than about 1900 bytes (512 bytes or 1024 bytes are typical) in applications such as protecting data and resources stored on modern NAND flash devices. The correction level in the BCH block is programmable to provide flexibility for varying applications and configurations of flash page size. The design can be programmed to encode protection of 2 to 40 bit errors when writing flash and to correct the corresponding number of errors on decode. The correction level when decoding MUST be programmed to the same correction level as was used during the encode phase.

BCH-codes are a type of block-code, which implies that all error-correction is performed over a block of N-symbols. The BCH operation will be performed over GF(2^{13} = 8192) or GF(2^{14} = 16384), which is the Galois Field consisting of 8191 or 16383 one-bit symbols. BCH-encoding (or encode for any block-code) can be performed by two algorithms: systematic encoding or multiplicative encoding. Systematic encoding is the process of reading all the symbols which constitute a block, dividing continuously these symbols by the generator polynomial for the GF(8192) or GF(16384) and appending the resulting ***t*** parity symbols to the block to create a BCH codeword (where ***t*** is the number of correctable bits).

The BCH encode process creates *t**13 (or *t**14)-bit parity symbols for each data block when the data is written to the flash device. The parity symbols are written to the flash device after the corresponding data block, and together these are collectively called the codeword. The codeword can be used during the decode process to correct errors that occur in either the data or parity blocks.

The BCH decoder processes code words in a 4-step fashion:

1. Syndrome Calculation (SC): This is the process of reading in all of the symbols of the codeword and continuously dividing by the generator polynomial for the field. 2**t* syndromes must be calculated for each codeword and inspection of the syndromes determines if there are errors: a non-zero set of syndromes indicates one or more errors. This process is implemented parallel hardware to minimize processing time since it must be done every time the decode is performed.
2. Key Equation Solver (KES): The syndromes represent 2t-linear equations with 2t-unknown variables. The process of solving these equations and selecting from the numerous solutions constitutes the KES module. When the KES block completes its operations, it generates an error locator polynomial (sigma) that is used in the proceeding block to determine the locations and values of the errors.
3. Chien Search (CS): This block takes input from the KES block and uses the Chien Algorithm for finding the locations of the errors based on the error locator polynomial. The method basically involves substituting all 8191 symbols from the GF(8192) or 16383 symbols from the GF(16383) into the locator polynomial. All evaluations that produce a zero solution indicate locations of the various errors. Since each located error corresponds to a single bit, the bit in the original data may be corrected by simply flipping the polarity of the incorrect location.
4. Correction: this block has to convert the symbol index and mask information to memory byte indexes and masks.

The BCH block, shown in the figure, was designed to operate in a pipelined fashion to maximize throughput. Aside from the initial latency to fill the pipeline stages, the BCH throughput is about 7/4 cycles/byte. Thus, the bottleneck in performing NAND reads and error corrections is the BCH rate. Current GPMI read rates are approximately 1/2 cycles/byte maximally for the current generation of NAND flash. Fortunately, BCH has a different master clock from GPMI, this gives some flexibility to match the throughput rate. The CPU is not directly involved in generating parity symbols, checking for errors, or correcting them.

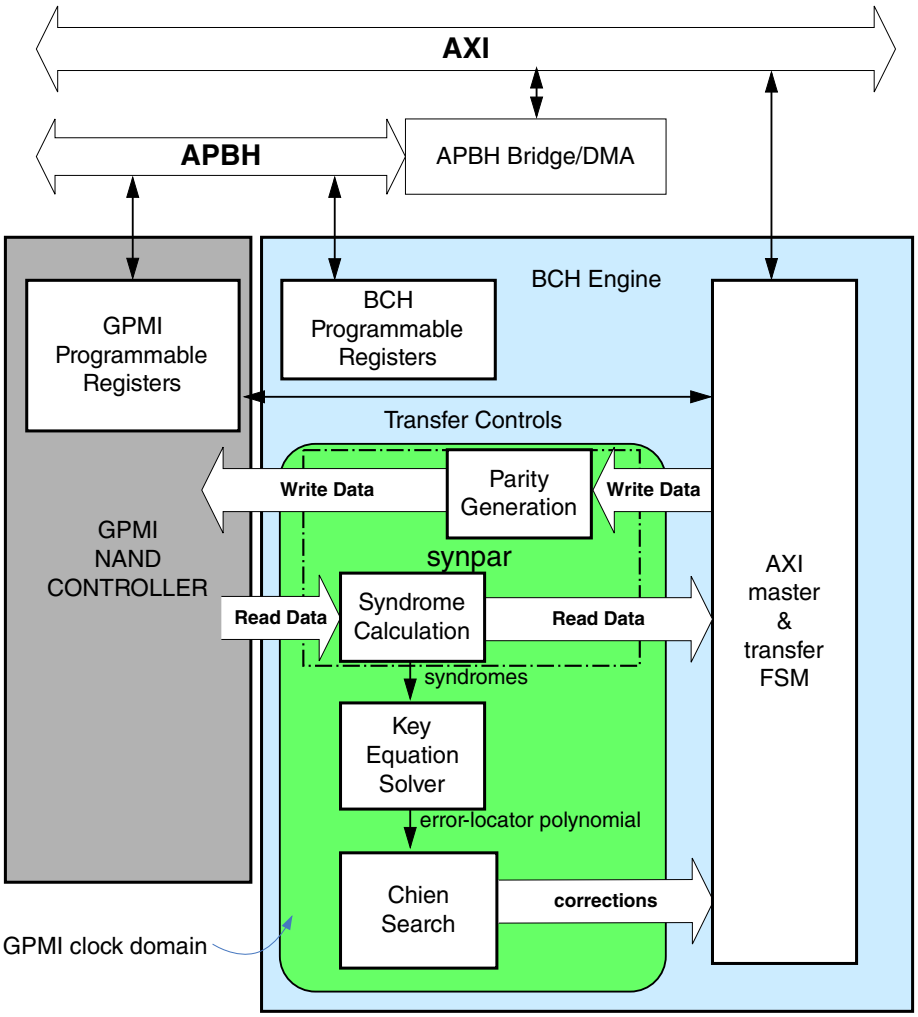


Figure 17-1. Hardware BCH Accelerator

17.2 Operation

Before performing any NAND flash read or write operations, software should first program the BCH's flash layout registers (see Flash Page Layout) to specify how data is to be formatted on the flash device.

The BCH hardware allows full programmability over the flash page layout to enable users flexibility in balancing ECC correction levels and ever-changing flash page sizes.

To initiate a NAND Flash write, software will program a GPMI DMA operation. The DMA need only program the GPMI control registers (and handle the requisite flash addressing handshakes) since the BCH will handle all data operations using its AXI bus interface. The BCH will then send the data to the GPMI controller to be written to flash

as it computes the parity symbols. At the end of each data block the BCH will insert the parity symbols into the data stream so that the GPMI sees only a continuous stream of data to be written.

NAND Flash read operations operate in a similar manner. As the GPMI controller reads the device, all data is sent to the BCH hardware for error detection/correction. The BCH controller writes all incoming read data to system memory and in parallel computes the syndromes used to detect bit errors. If errors are detected within a block, the BCH hardware activates the error correction logic to determine where bit errors have occurred and ultimately correct them in the data buffer in system memory. After an entire flash page has been read and corrected, the BCH will signal an interrupt to the CPU.

The figure below indicates how data read from the GPMI is operated on within the BCH hardware. As the BCH receives data from the GPMI (top row), it is written to memory by the BCH's Bus Interface Unit (BIU) (second row). For blocks requiring correction, the KES logic will be activated after the entire block has been received. Once the error locator polynomial has been computed, the corrections are determined by the Chien Search and fed back to the BIU, which performs a read, modify, write operation on the buffer in memory to correct the data.

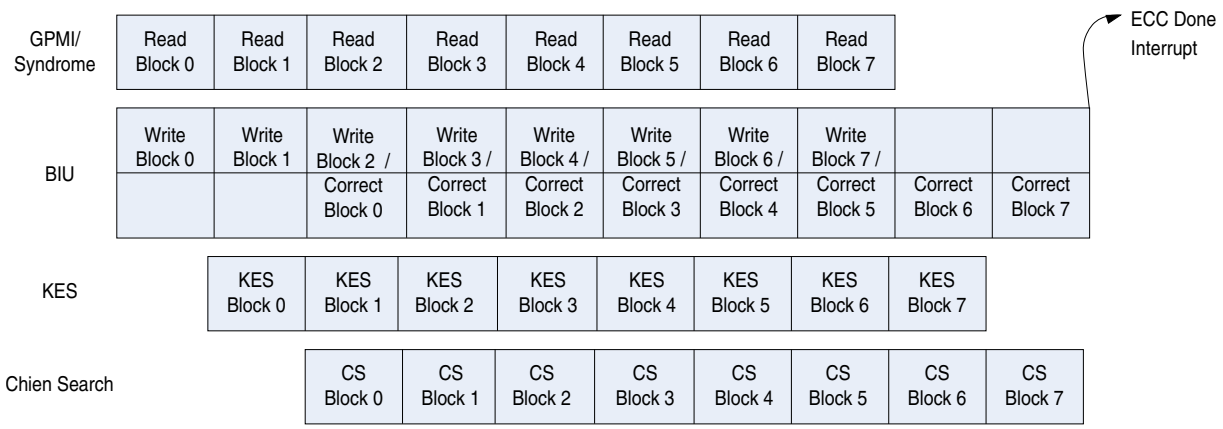


Figure 17-2. Block Pipeline while Reading Flash

17.2.1 BCH Limitations and Assumptions

- The BCH is programmable to support 2 to 40 bit error correction. ECC0 is supported as a pass-through, non-correcting mode.
- Data block sizes must be a multiple of 4 bytes and be aligned in system memory.
- The BCH supports a programmable number of metadata/auxiliary data bytes, from 0 to 255.

- Metadata will be written at the beginning of the flash page to facilitate fast access for filesystem operations.
- Metadata may be treated as an independent block for ECC purposes or combined with the first data block to conserve bits in the flash.
- The BCH does not support a partial page write (this can be accomplished by programming the BCH layout registers such that the BCH only sees a portion of the page).
- Flash read operations can read the entire page or the first block on the page.
- The BCH also supports a memory-to-memory mode of operation that does not require the use of DMA or the GPMI.

17.2.2 Flash Page Layout

The BCH supports a fully programmable flash page layout. The BCH maintains four independent layout registers that can describe four completely different NAND devices or layouts.

When the BCH initiates an operation, it selects one of the layouts by using the chip select as an index into the BCH_LAYOUTSELECT register that determines which layout should be used for the operation.

Three possible (generic) flash layout schemes are supported, as indicated in the figure below. (In each case, the metadata size may also be programmed to 0 bytes). Metadata may either be combined with the first block of data or the size of the first data block can be programmed to 0 to allow the metadata to be protected by its own ECC parity bits.

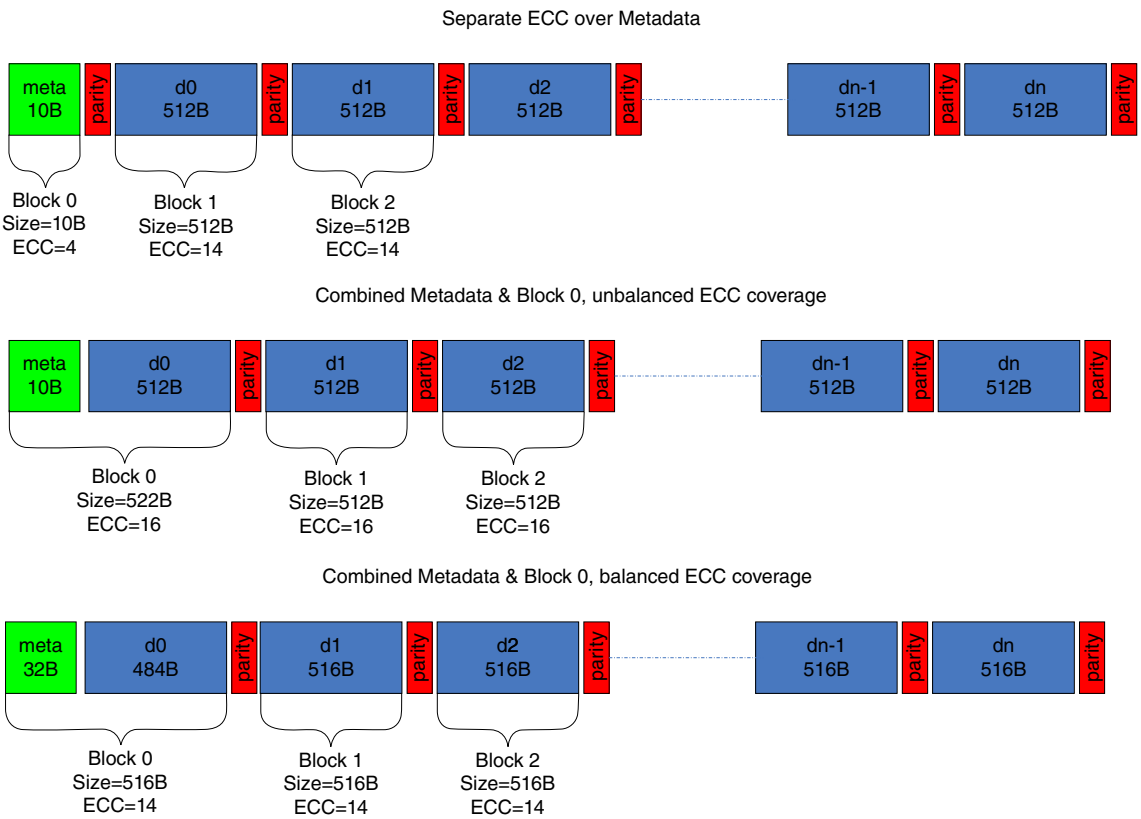


Figure 17-3. FLASH Page Layout Options

Each layout is determined by a pair of registers that define the following parameters:

- DATA0_SIZE: Indicates the number of data bytes in the first block on the page (this should not include parity or metadata bytes). This should be set to 0 when the metadata is to be covered separately with its own ECC. This must be a multiple of 4 bytes.
- ECC0: Indicates the ECC level to be used for the first block on the flash (data0+metadata).
- META_SIZE: Indicates the number of bytes (from 0-255) that are stored as metadata.
- NBLOCKS: Indicates the number of subsequent DATAN blocks on the flash, or the number of blocks following the DATA0 block.
- DATAN_SIZE: Indicates the number of data bytes in all subsequent data blocks. This MUST be a multiple of 4 bytes.
- ECCN: Indicates the ECC level to be used for the subsequent data blocks.

- GF0 or GFN: Indicates the Galois field the meta / data blocks are using
- PAGE_SIZE: Indicates the total number of bytes available per page on the physical flash device. This includes the spare area and is typically 4096+128, 4096+218, or 2048+64 bytes.

17.2.3 Determining the ECC layout for a device

Since the BCH is programmable, a system can trade off ECC levels for flash size and layout configurations.

The following examples indicate how to determine a valid layout based on the required storage space and flash size. For all cases, the size of the parity will be 13 (or 14 for GF(2^{14}))*ECC level ***bits***-- so for ECC8, 13 (or 14) bytes are required (per block).

17.2.3.1 4K+218 flash, 10 bytes metadata, 512 byte data blocks, separate metadata, Assuming GF(2^{13})

In this case, we have 8 data blocks each consisting of 512 bytes. Since the flash has 218 spare bytes (1744 bits), first estimate an ECC level for the data blocks by first subtracting the number of metadata bytes from the spare bytes (218 – 10 = 208 bytes = 1664 bits) then dividing the number of bits by 8 (number of blocks) and then by 13 (bits per ECC level).

(218 - 10) x 8 = 1664/13(8) = 16

Therefore all the data blocks could be covered by ECC16 if the metadata had no parity. This isn't acceptable, so assume ECC14 for all the data blocks. Now calculate the number of free bits for the metadata parity as

1664 - (14) x 13 x 8 = 208

Therefore, 208 bits remain for metadata parity. Dividing by 13 (bits/ECC) gives 16, so the metadata can be covered with ECC16. The settings for this device would then be

Table 17-1. Settings for 4K+218 FLASH

Setting	Value
PAGE_SIZE	4096+218=4314=0x10DA
META_SIZE	10=0x0A
DATA0_SIZE	0
ECC0	16=0x10

Table continues on the next page...

Table 17-1. Settings for 4K+218 FLASH (continued)

Setting	Value
GF0	GF(2^{13})
DATAN_SIZE	512=0x200 (in register interface, assigned as 0x80)
ECCN	14=0x0E
GFN	GF(2^{13})
NBLOCKS	8

17.2.3.2 4K+128 flash, 10 bytes metadata, 1024 byte data blocks, separate metadata, assuming GF(2^{13}) for data and GF(2^{14}) for metadata

This flash will have 118 bytes available for ECC (after subtracting the metadata size), therefore, 994 bits.

Dividing by 4*14 (number of blocks * ECC level) we get 17.75, therefore we can support ECC16 on the data blocks. The number of free spare bits becomes 944 - 16 * 4 * 14 = 944 - 896 = 48, divided by 13 = 3.69, therefore the metadata can be also covered by ECC2.

Table 17-2. Settings for 4K+128 FLASH

Setting	Value
PAGE_SIZE	4096+128=4224=0x1080
META_SIZE	10=0x0A
DATA0_SIZE	0
ECC0	2
GF0	GF(2^{13})
DATAN_SIZE	1024=0x400 (in register interface, assigned as 0x100)
ECCN	16
GFN	GF(2^{14})
NBLOCKS	4

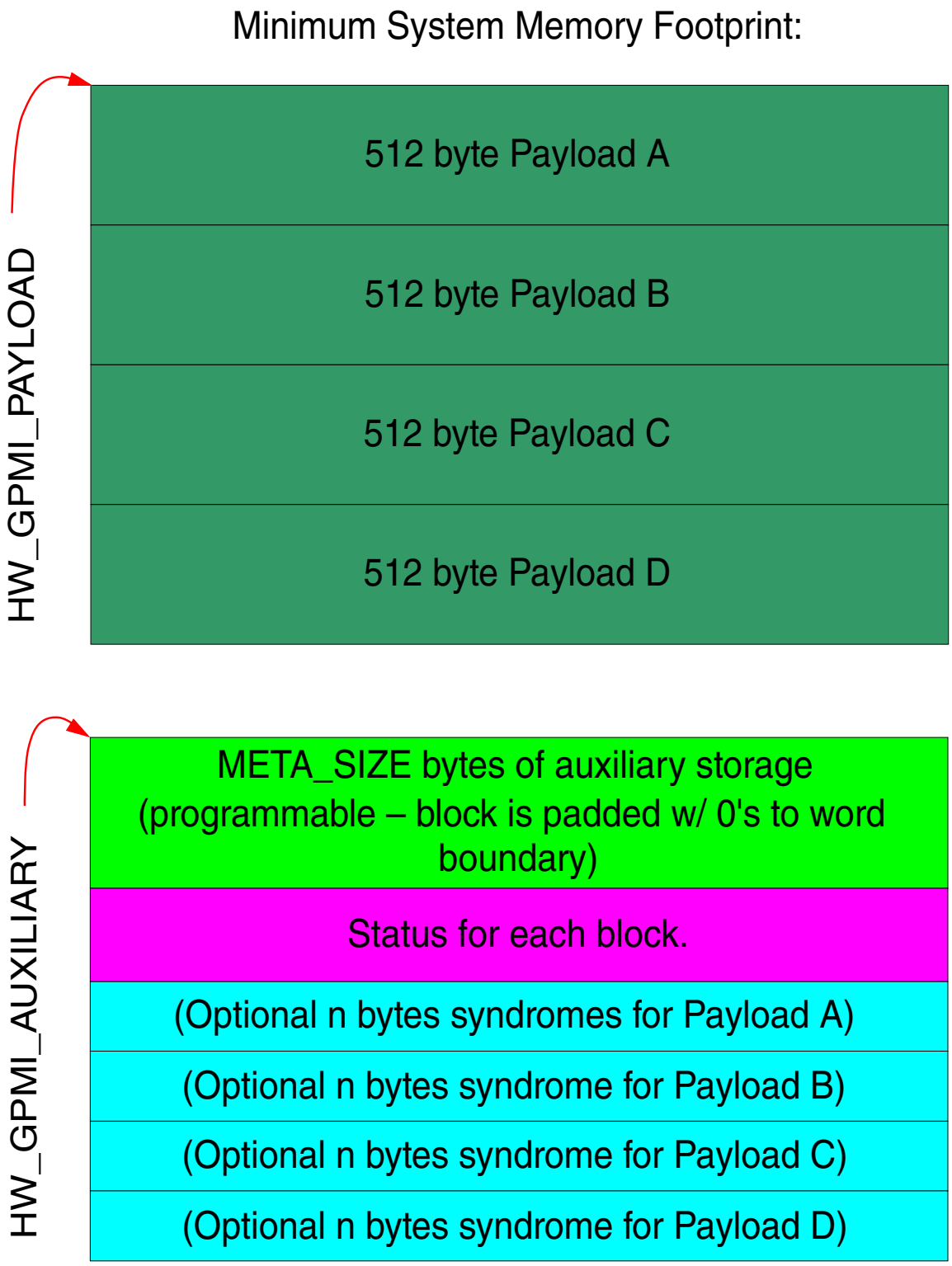
In this case, there will be additional unused spare bits, with the BCH will pad out with zeros.

17.2.4 Data Buffers in System Memory

While the data on the flash is interleaved with parity symbols, the BCH assumes that the data buffers in memory are contiguous.

Metadata read from the flash will be stored to the location pointed to by the GPMI_AUXILIARY register and data will be written to the address specified in the GPMI_PAYLOAD register as is shown in the following figure where the block length is 512 bytes for example. Since the number of blocks on a flash page is programmable, the BCH also writes individual block correction status to the auxiliary pointer at the word-aligned address following the end of the metadata. Optionally, the computed syndromes may also be written to the auxiliary area if the DEBUGSYNDROME bit is set in the control register.

As blocks complete processing, the bus master will accumulate the status for each block and write it to the auxiliary data buffer following the metadata. The metadata area will be padded with 0's until the next word boundary and the status for blocks 0-3 will be written to the next word. The status for subsequent blocks will then be written to the buffer. The status for the first block (metadata block) is also stored in the STATUS_BLK0 register in the BCH_STATUS register. The completion codes for the blocks are indicated in the Table 17-3. Note that the definition of the bytes and their ordering in the auxiliary and payload storage areas are user defined. When this data is read back from the flash and put into memory, it will resemble the original buffer that was written out to the flash.



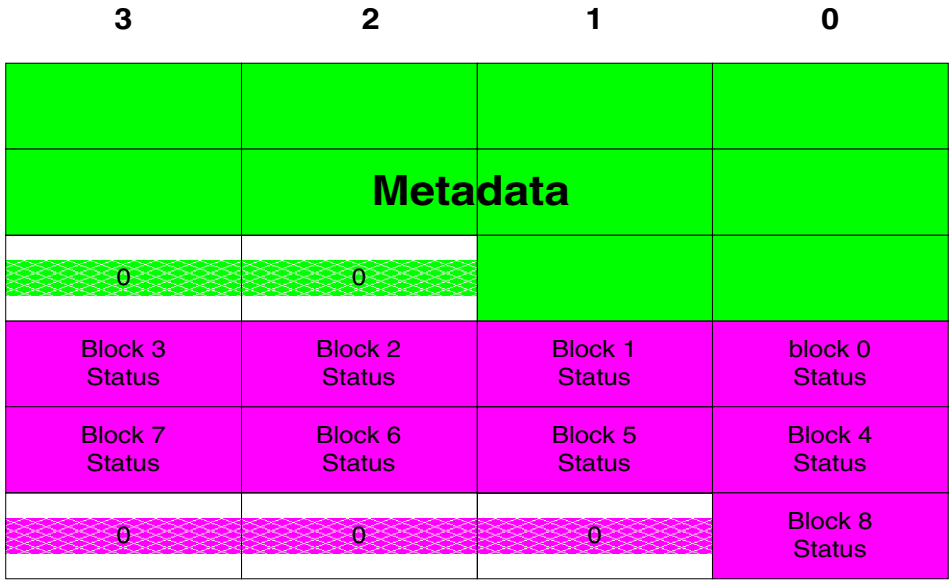
*Computed syndrome area consists of 2*t 13 (or 14)-bit symbols written as 16-bit half word.*

Figure 17-4. BCH Data Buffers in Memory

Table 17-3. Status Block Completion Codes

Code	Description
0xFF	Block is erased
0xFE	Block is uncorrectable
0x00	No errors found
0x01-0x28	Number of errors corrected

The following figure shows the layout of the bytes within the status field.



Status bytes are allocated based on the NBLOCKS programmed into the flash format register. The number of status bytes will be computed by the NBLOCKS+1. The status area will be padded with zeros to the next word boundary.

Syndrome data written for debug purposes will follow the end of the status block.

Figure 17-5. Memory-to-Memory Operations

17.3 Memory to Memory (Loopback) Operation

The BCH supports a memory-to-memory mode of operation where both the encoded and decoded buffers reside in system memory.

This can be useful for applications where data must be protected by ECC, but the storage device does not reside on the GPMI bus.

The BCH operation in memory to memory mode is much simpler than in GPMI mode since DMAs are not required to manage the operation. Instead, software simply writes the BCH_DATAPTR and BCH_METAPTR with the addresses of the data and metadata (auxiliary) buffers and the BCH_ENCODEPTR with the address of the buffer for encoded data. To initiate the operation, software simply sets the M2M_ENCODE and M2M_ENABLE bits in the control register. The BCH can be programmed to either issue an interrupt at the end of the operation or software may poll the status bits for completion.

Memory to memory decode operations work in a similar manner. The encoded data address is written to the BCH_ENCODEPTR and the data and meta pointers are written to buffers that correspond to the desired decoded data addresses. To initiate a decode, software must set the M2M_ENCODE bit to 0 while writing the M2M_ENABLE bit. Note that the addresses written to the BCH_DATAPTR, BCH_METAPTR and BCH_ENCODEPTR registers should always be aligned on a 4 byte boundary. In other words, the 2 lower bits of the address should always be written with zeros.

17.4 Programming the BCH/GPMI Interfaces

Programming the BCH for NAND operations consists largely of disabling the soft reset and clock bits (SFTRST and CLKGATE) from the BCH_CTRL register and then programming the flash layout registers to correspond to the format of the attached NAND device(s).

The BCH_LAYOUTSELECT register should also be programmed to map the chip select of each attached device into one of the four layout registers.

The bulk of the programming is actually applied to the GPMI through PIO operations embedded in DMA command structures. The DMA will perform all the requisite handshaking with the GPMI interface to negotiate the address portion of the transfer, then the BCH will handle all the movement of data from memory to the GPMI (writes) or the GPMI to memory (reads). The BCH will direct all data blocks to the buffer pointed to by the PAYLOAD_BUFFER and the metadata will be written to the AUXILIARY_BUFFER. Both of these registers are located in the GPMI PIO data space and are communicated to the BCH hardware at the beginning of the transfer. Thus, the normal multi-NAND DMA based device interleaving is preserved, that is, four NANDs on four separate chip selects can be scheduled for read or write operations using the BCH. Whichever channel finishes its ready wait first and enters the DMA arbiter with its lock bit set owns the GPMI command interface and through it owns the BCH resources for the duration of its processing.

17.4.1 BCH Encoding for NAND Writes

The BCH encoder flowchart in Figure 17-6 shows the detailed steps involved in programming and using the BCH encoder. This flowchart shows how to use the BCH block with the GPMI.

To use the BCH encoder with the GPMI's DMA, create a DMA command chain containing ten descriptor structures, as shown in Figure 17-8 and detailed in the DMA structure code example that follows it in DMA Structure Code Example. The ten descriptors perform the following tasks:

1. Disable the BCH block (in case it was enabled) and issue NAND write setup command byte (under CLE) and address bytes (under ALE).
2. Configure and enable the BCH and GPMI blocks to perform the NAND write.
3. Disable the BCH block and issue NAND write execute command byte (under CLE).
4. Wait for the NAND device to finish writing the data by watching the ready signal.
5. Check for NAND timeout through PSENSE.
6. Issue NAND status command byte (under CLE).
7. Read the status and compare against expected.
8. If status is incorrect or incomplete, branch to error handling descriptor chain.
9. Otherwise, write is complete and emit GPMI interrupt.

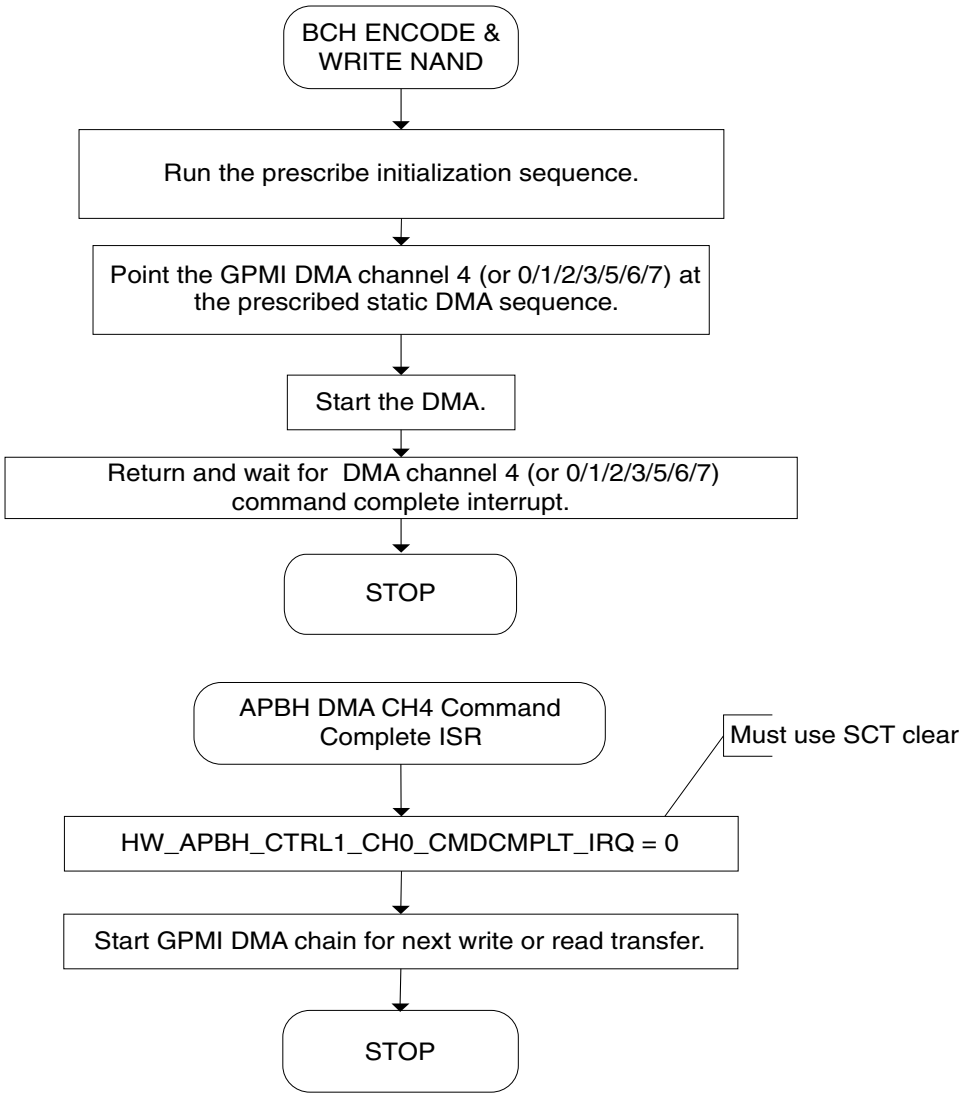


Figure 17-6. BCH Encode Flowchart

Descriptor Legend

NEXT CMD ADDR										
CMD	<=	xfer_count	cmdwords	wait4endcmd	semaphore	nandwait4ready	nandlock	irqoncmplt	chain	command
BUFFER ADDR										
HW_GPMI_CTRL0	<=	command_mode	word_length	lock_cs	CS	address	address_increment	xfer_count		
HW_GPMI_COMPARE	<=	mask				reference				
HW_GPMI_ECCCTRL	<=	ecc_cmd			enable_ecc				buffer_mask	
HW_GPMI_ECCCOUNT										
HW_GPMI_PAYLOAD										
HW_GPMI_AUXILIARY										

Figure 17-7. BCH DMA Descriptor Legend

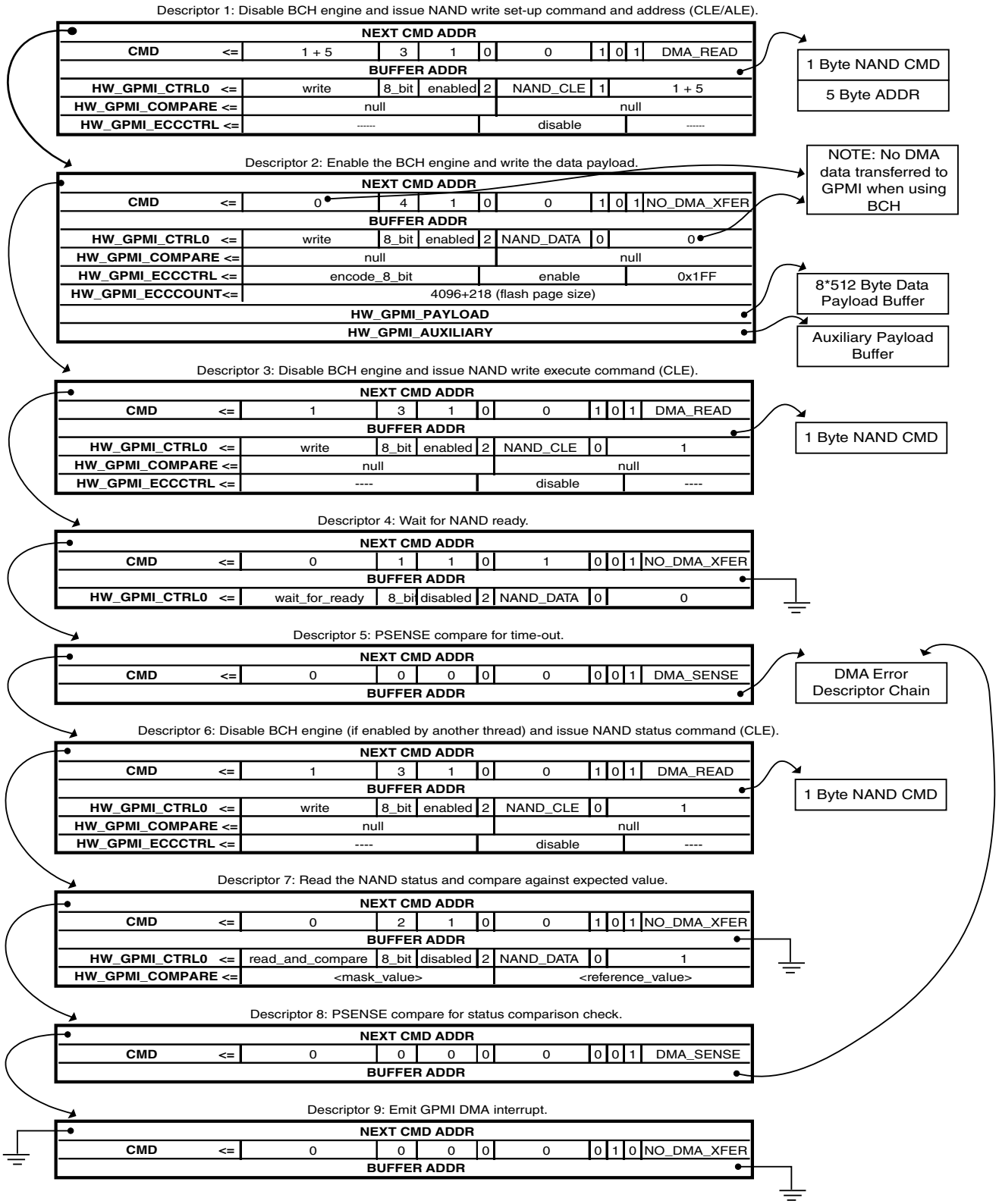


Figure 17-8. BCH Encode DMA Descriptor Chain

17.4.1.1 DMA Structure Code Example

The following code sample illustrates the coding for one write transaction involving 4096 bytes of data payload (eight 512-byte blocks) and 10 bytes of auxiliary payload (also referred to as metadata) to a 4K NAND page sitting on GPMI CS2.

```
//-----------------------------------------------------------------------------
// generic DMA/GPMI/ECC descriptor struct, order sensitive!
//-----------------------------------------------------------------------------
typedef struct {
  // DMA related fields
  unsigned int dma_nxtcmdar;
  unsigned int dma_cmd;
  unsigned int dma_bar;
  // GPMI related fields
  unsigned int gpmi_ctrl0;
  unsigned int gpmi_compare;
  unsigned int gpmi_eccctrl;
  unsigned int gpmi_ecccount;
  unsigned int gpmi_data_ptr;
  unsigned int gpmi_aux_ptr;
} GENERIC_DESCRIPTOR;
//-----------------------------------------------------------------------------
// allocate 10 descriptors for doing a NAND ECC Write
//-----------------------------------------------------------------------------
GENERIC_DESCRIPTOR write[10];
//-----------------------------------------------------------------------------
// DMA descriptor pointer to handle error conditions from psense checks
//-----------------------------------------------------------------------------
unsigned int * dma_error_handler;
//-----------------------------------------------------------------------------
// 8 byte NAND command and address buffer
// any alignment is ok, it is read by the GPMI DMA
//   byte 0 is write setup command
//   bytes 1-5 is the NAND address
//   byte 6 is write execute command
//   byte 7 is status command
//-----------------------------------------------------------------------------
unsigned char nand_cmd_addr_buffer[8];
//-----------------------------------------------------------------------------
// 4096 byte payload buffer used for reads or writes
// needs to be word aligned
//-----------------------------------------------------------------------------
unsigned int write_payload_buffer[(4096/4)];
//-----------------------------------------------------------------------------
// 65 byte meta-data to be written to NAND
// needs to be word aligned
//-----------------------------------------------------------------------------
unsigned int write_aux_buffer[65];
//-----------------------------------------------------------------------------
// Descriptor 1: issue NAND write setup command (CLE/ALE)
//-----------------------------------------------------------------------------
write[0].dma_nxtcmdar = &write[1];                          // point to the next descriptor
write[0].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (1 + 5)|   // 1 byte command, 5 byte address
                   BF_APBH_CHn_CMD_CMDWORDS      (3)    |   // send 3 words to the GPMI
                 BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)    |  // wait for command to finish
before
                                                            //    continuing
                   BF_APBH_CHn_CMD_SEMAPHORE     (0)    |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)    |
                   BF_APBH_CHn_CMD_NANDLOCK      (1)    |   // prevent other DMA channels
from
                                                            //    taking over
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0)    |
                   BF_APBH_CHn_CMD_CHAIN         (1)    |   // follow chain to next command
```

```
                BV_FLD(APBH_CHn_CMD, COMMAND, DMA_READ); // read data from DMA, write to
NAND
write[0].dma_bar = &nand_cmd_addr_buffer;           // byte 0 write setup, bytes 1 - 5 NAND
address
// 3 words sent to the GPMI
write[0].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)     | // write to the NAND
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)     |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      ENABLED)   |
                    BF_GPMI_CTRL0_CS              (2)      | // must correspond to NAND CS
used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_CLE) |
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (1)       | // send command and
address
                    BF_GPMI_CTRL0_XFER_COUNT       (1 + 5);    // 1 byte command, 5 byte
address
write[0].gpmi_compare = NULL;                            // field not used but necessary to
set eccctrl
write[0].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, DISABLE); // disable the ECC block
//-------------------------------------------------------------------------
// Descriptor 2: write the data payload (DATA)
//-------------------------------------------------------------------------
write[1].dma_nxtcmdar = &write[2];                        // point to the next descriptor
write[1].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT      (0)|  // NOTE: No DMA data transfer
                   BF_APBH_CHn_CMD_CMDWORDS        (4)|  // send 4 words to the GPMI
                   BF_APBH_CHn_CMD_WAIT4ENDCMD     (1)|  // Wait to end
                   BF_APBH_CHn_CMD_SEMAPHORE       (0)|
                   BF_APBH_CHn_CMD_NANDWAIT4READY  (0)|
                   BF_APBH_CHn_CMD_NANDLOCK        (1)|  // maintain resource lock
                   BF_APBH_CHn_CMD_IRQONCMPLT      (0)|
                   BF_APBH_CHn_CMD_CHAIN           (1)|  // follow chain to next command
                   BV_FLD(APBH_CHn_CMD, COMMAND, DMA_NO_XFER);  // No data transferred
write[1].dma_bar = &write_payload_buffer;                       // pointer for the 4K byte
data area
// 4 words sent to the GPMI
write[1].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)     | // write to the NAND
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)     |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      ENABLED)   |
                    BF_GPMI_CTRL0_CS                (2)       | // must correspond to NAND
CS used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA)|
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)       |
                      BF_GPMI_CTRL0_XFER_COUNT         (0);        // NOTE: this field
contains
                                                                  //  the total amount
                              // DMA transferred to GPMI via DMA (0)!
write[1].gpmi_compare = NULL;                                  // field not used but necessary
to
set eccctrl
write[1].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ECC_CMD,    ENCODE_8_BIT) | // specify t = 8
mode
                        BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, ENABLE)       | // enable ECC module
                    BF_GPMI_ECCCTRL_BUFFER_MASK   (0x1FF);          // write all 8 data
blocks
                                                                    //   and 1 aux block
write[1].gpmi_ecccount = BF_GPMI_ECCCOUNT_COUNT(4096+218);      // specify number of bytes
                                                                //   written to NAND
write[1].gpmi_data_pointer = &write_payload_pointer;            // data buffer address
write[1].gpmi_aux_pointer  = &write_aux_pointer;                // metadata pointer
//-------------------------------------------------------------------------
// Descriptor 3: issue NAND write execute command (CLE)
//-------------------------------------------------------------------------
write[2].dma_nxtcmdar = &write[3];                          // point to the next descriptor
write[2].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (1) |     // 1 byte command
                   BF_APBH_CHn_CMD_CMDWORDS      (3) |     // send 3 words to the GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1) |    // wait for command to finish
before
                                                         //    continuing
                   BF_APBH_CHn_CMD_SEMAPHORE     (0) |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0) |
                   BF_APBH_CHn_CMD_NANDLOCK      (1) |     // maintain resource lock
```

```
                    BF_APBH_CHn_CMD_IRQONCMPLT    (0) |
                    BF_APBH_CHn_CMD_CHAIN         (1) |     // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, DMA_READ);   // read data from DMA, write to
NAND
write[2].dma_bar = &nand_cmd_addr_buffer[6];             // point to byte 6, write execute
command
// 3 words sent to the GPMI
write[2].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)    | // write to the NAND
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)    |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      ENABLED)  |
                    BF_GPMI_CTRL0_CS               (2)      | // must correspond to NAND CS
used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_CLE) |
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)       |
                      BF_GPMI_CTRL0_XFER_COUNT         (1);       // 1 byte command
write[2].gpmi_compare = NULL;                           // field not used but necessary to set
eccctrl
write[2].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, DISABLE); // disable the ECC block
//-------------------------------------------------------------------------
// Descriptor 4: wait for ready (CLE)
//-------------------------------------------------------------------------
write[3].dma_nxtcmdar = &write[4];                       // point to the next descriptor

write[3].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0) |   // no dma transfer
                   BF_APBH_CHn_CMD_CMDWORDS      (1) |   // send 1 word to the GPMI
                   BF_APBH_CHn_CMD_WAIT4ENDCMD   (1) |   // wait for command to finish before
                                                         //      continuing
                   BF_APBH_CHn_CMD_SEMAPHORE     (0) |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(1) |   // wait for nand to be ready
                   BF_APBH_CHn_CMD_NANDLOCK      (0) |   // relinquish nand lock
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0) |
                   BF_APBH_CHn_CMD_CHAIN         (1) |   // follow chain to next command
                   BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);          // no dma transfer
write[3].dma_bar = NULL;                                                // field not used
// 1 word sent to the GPMI
write[3].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WAIT_FOR_READY) | // wait for NAND
ready
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)         |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)      |
                  BF_GPMI_CTRL0_CS              (2)         | // must correspond to NAND CS
used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA)     |
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)            |
                      BF_GPMI_CTRL0_XFER_COUNT         (0);
//-------------------------------------------------------------------------
// Descriptor 5: psense compare (time out check)
//-------------------------------------------------------------------------
write[4].dma_nxtcmdar = &write[5];                         // point to the next descriptor
write[4].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)       | // no dma transfer
                   BF_APBH_CHn_CMD_CMDWORDS      (0)       | // no words sent to GPMI
                   BF_APBH_CHn_CMD_WAIT4ENDCMD   (0)       | // do not wait to continue
                   BF_APBH_CHn_CMD_SEMAPHORE     (0)       |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)       |
                   BF_APBH_CHn_CMD_NANDLOCK      (0)       |
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0)       |
                   BF_APBH_CHn_CMD_CHAIN         (1)       | // follow chain to next command
                   BV_FLD(APBH_CHn_CMD, COMMAND, DMA_SENSE);  // perform a sense check
write[4].dma_bar = dma_error_handler;              // if sense check fails, branch to error
handler
//-------------------------------------------------------------------------
// Descriptor 6: issue NAND status command (CLE)
//-------------------------------------------------------------------------
write[5].dma_nxtcmdar = &write[6];                       // point to the next descriptor
write[5].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (1)    |  // 1 byte command
                   BF_APBH_CHn_CMD_CMDWORDS      (3)    |  // send 3 words to the GPMI
                 BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)    |  // wait for command to finish
before
continuing
                   BF_APBH_CHn_CMD_SEMAPHORE     (0)    |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)    |
```

```
                  BF_APBH_CHn_CMD_NANDLOCK      (1)    |  // prevent other DMA channels from
taking over
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0)    |
                   BF_APBH_CHn_CMD_CHAIN         (1)    |  // follow chain to next command
                BV_FLD(APBH_CHn_CMD, COMMAND, DMA_READ);  // read data from DMA, write to
NAND
write[5].dma_bar = &nand_cmd_addr_buffer[7];                // point to byte 7, status
command
write[5].gpmi_compare = NULL;                            // field not used but necessary to set
eccctrl
write[5].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, DISABLE); // disable the ECC block
// 3 words sent to the GPMI
write[5].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)    | // write to the NAND
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)    |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      ENABLED)  |
                   BF_GPMI_CTRL0_CS               (2)     | // must correspond to NAND CS
used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_CLE) |
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)       |
                      BF_GPMI_CTRL0_XFER_COUNT         (1);       // 1 byte command
//-----------------------------------------------------------------------------
// Descriptor 7: read status and compare (DATA)
//-----------------------------------------------------------------------------
write[6].dma_nxtcmdar = &write[7];                          // point to the next descriptor
write[6].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)  |    // no dma transfer
                   BF_APBH_CHn_CMD_CMDWORDS      (2)  |    // send 2 words to the GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)  |    // wait for command to finish
before
                                                          //      continuing
                   BF_APBH_CHn_CMD_SEMAPHORE     (0)  |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)  |
                   BF_APBH_CHn_CMD_NANDLOCK      (1)  |    // maintain resource lock
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0)  |
                   BF_APBH_CHn_CMD_CHAIN         (1)  |    // follow chain to next command
                   BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);  // no dma transfer
write[6].dma_bar = NULL;                                    // field not used
// 2 word sent to the GPMI
write[6].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, READ_AND_COMPARE) | // read from the
                                                                  //       NAND and
//                                                               // compare to expect
                      BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)     |
                      BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)  |
                   BF_GPMI_CTRL0_CS             (2)       |   // must correspond to NAND CS
used
                      BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA) |
                      BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)        |
                      BF_GPMI_CTRL0_XFER_COUNT         (1);
write[6].gpmi_compare = <MASK_AND_REFERENCE_VALUE>;  // NOTE: mask and reference values are
NAND
                                                          // SPECIFIC to evaluate the NAND
status
//-----------------------------------------------------------------------------
// Descriptor 8: psense compare (time out check)
//-----------------------------------------------------------------------------
write[7].dma_nxtcmdar = &write[8];                            // point to the next descriptor
write[7].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)        | // no dma transfer
                   BF_APBH_CHn_CMD_CMDWORDS      (0)        | // no words sent to GPMI
                   BF_APBH_CHn_CMD_WAIT4ENDCMD   (0)        | // do not wait to continue
                   BF_APBH_CHn_CMD_SEMAPHORE     (0)        |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)        |
                   BF_APBH_CHn_CMD_NANDLOCK      (0)        | // relinquish nand lock
                   BF_APBH_CHn_CMD_IRQONCMPLT    (0)        |
                   BF_APBH_CHn_CMD_CHAIN         (1)        | // follow chain to next command
                   BV_FLD(APBH_CHn_CMD, COMMAND, DMA_SENSE);  // perform a sense check
write[7].dma_bar = dma_error_handler;             // if sense check fails, branch to error
handler
//-----------------------------------------------------------------------------
// Descriptor 9: emit GPMI interrupt
//-----------------------------------------------------------------------------
write[8].dma_nxtcmdar = NULL;                                 // not used since this is
```

```
last
descriptor
write[8].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT     (0)          | // no dma transfer
                   BF_APBH_CHn_CMD_CMDWORDS       (0)          | // no words sent to GPMI
                   BF_APBH_CHn_CMD_WAIT4ENDCMD    (0)          | // do not wait to continue
                   BF_APBH_CHn_CMD_SEMAPHORE      (0)          |
                   BF_APBH_CHn_CMD_NANDWAIT4READY(0)           |
                   BF_APBH_CHn_CMD_NANDLOCK       (0)          |
                   BF_APBH_CHn_CMD_IRQONCMPLT     (1)          | // emit GPMI interrupt
                   BF_APBH_CHn_CMD_CHAIN          (0)          | // terminate DMA chain
processing
                   BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);  // no dma transfer
```

17.4.1.2 Using the BCH Encoder

To use the BCH encoder, first turn off the module-wide soft reset bit in both the GPMI and BCH blocks before starting any DMA activity.

Turning off the soft reset must take place by itself, prior to programming the rest of the control registers. Turn off the BCH bus master soft reset bit. Turn off the clock gate bits.

Program the remainder of the GPMI, BCH and APBH DMA as follows:

```
// bring APBH out of reset
APBH_CTRL0_CLR(BM_APBH_CTRL0_SFRST);
APBH_CTRL0_CLR(BM_APBH_CTRL0_CLKGATE);

// bring BCH out of reset
BCH_CTRL_CLR(BM_BCH_CTRL_SFTRST);
BCH_CTRL_CLR(BM_BCH_CTRL_CLKGATE);

// bring gpmi out of reset
GPMI_CTRL0_CLR(BM_GPMI_CTRL0_SFTRST);
GPMI_CTRL0_CLR(BM_GPMI_CTRL0_CLKGATE);
GPMI_CTRL1_SET(BM_GPMI_CTRL1_DEV_RESET | // deassert reset
                   BM_GPMI_CTRL1_BCH_MODE  );    // enable BCH mode

// enable pinctrl
PINCTRL_CTRL_WR(0x00000000);

// enable gpmi pins
PINCTRL_MUXSEL0_CLR(0x0000ffff);    // data bits
PINCTRL_MUXSEL1_CLR(0x03ffffff);    // control bits
PINCTRL_MUXSEL8_CLR(0x0003f3ff);    // control bits
PINCTRL_MUXSEL8_SET(0x00015155);    // control bits
```

Note that for writing NANDs (ECC encoding), only GPMI DMA command complete interrupts are used. The BCH engine is used for writing to the NAND but may optionally produces an interrupt. From the sample code in DMA Structure Code Example:

- DMA descriptor 1 prepares the NAND for data write by using the GPMI to issue a write setup command byte under CLE, then sends a 5-byte address under ALE. The BCH engine is disabled and not used for these commands.

- DMA descriptor 2 enables the BCH engine for encoding to begin the initial writing of the NAND data by specifying where the data and auxiliary payload are coming from in system memory.
- DMA descriptor 3 issues the write commit command byte under CLE to the NAND.
- DMA descriptor 4 waits for the NAND to complete the write commit/transfer by watching the NAND's ready line status. This descriptor relinquishes the NANDLOCK on the GPMI to enable the other DMA channels to initiate NAND transactions on different NAND CS lines.
- DMA descriptor 6 issues a NAND status command byte under "CLE" to check the status of the NAND device following the page write.
- DMA descriptor 7 reads back the NAND status and compares the status with an expected value. If there are differences, then the DMA processing engine follows an error-handling DMA descriptor path.
- DMA descriptor 8 disables the BCH engine and emits a GPMI interrupt to indicate that the NAND write has been completed.

17.4.2 BCH Decoding for NAND Reads

When a page is read from NAND flash, BCH syndromes will be computed and, if correctable errors are found, they will be corrected on a per block basis within the NAND page.

This decoding process is fully overlapped with other NAND data reads and with CPU execution. The BCH decoder flowchart in the figure below shows the steps involved in programming the decoder. The hardware flow of reading and decoding a 4096-byte page is shown in Figure 17-10.

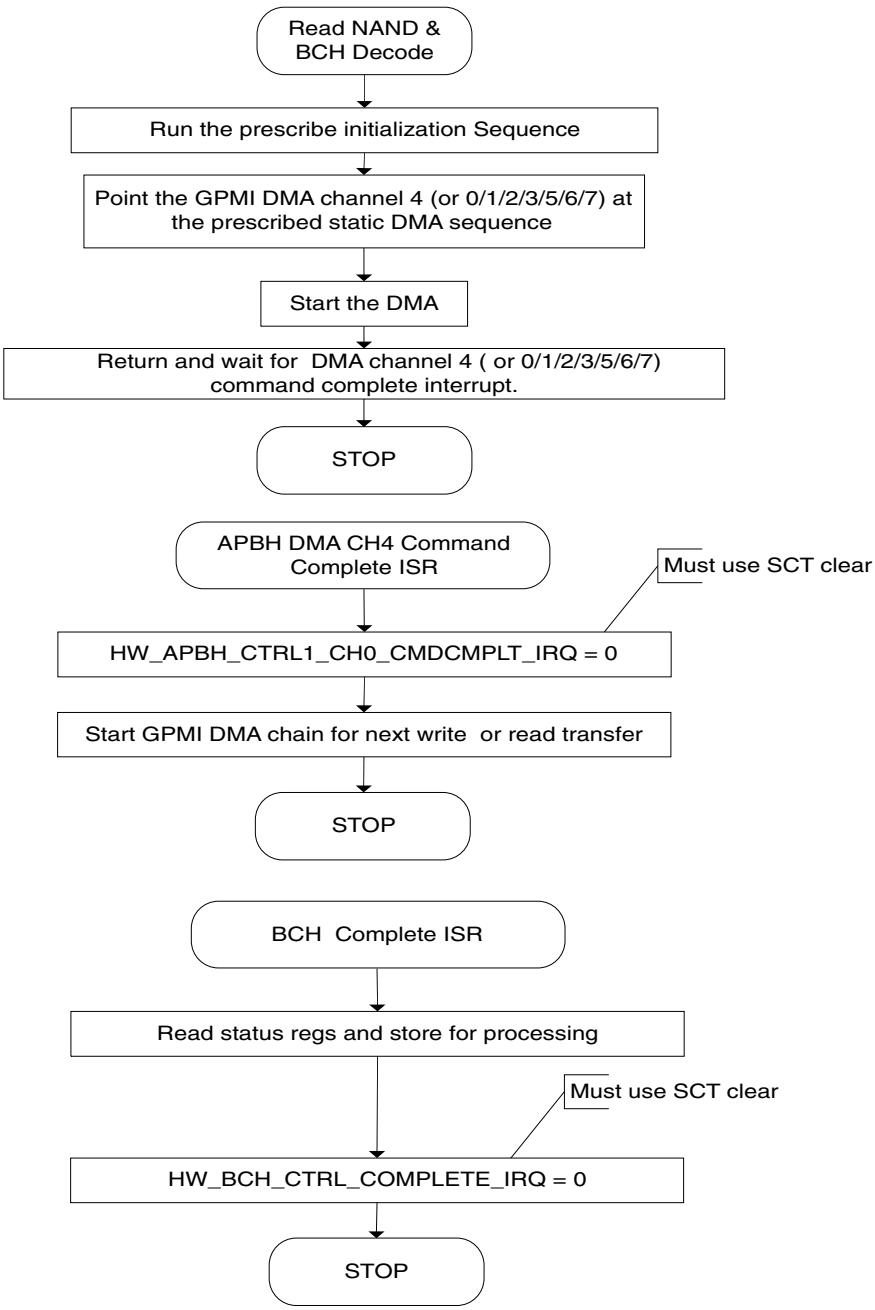


Figure 17-9. BCH Decode Flowchart

Conceptually, an APHB DMA Channel (0,1,2 or 3) command chain with seven command structures linked together is used to perform the BCH decode operation (as shown in Figure 17-10).

Note

The GPMI's DMA command structures controls the BCH decode operation.

To use the BCH decoder with the GPMI's DMA, create a DMA command chain containing seven descriptor structures, as shown in the figure below and detailed in the DMA structure code example that follows it in DMA Structure Code Example. The seven DMA descriptors perform the following tasks:

1. Issue NAND read setup command byte (under "CLE") and address bytes (under "ALE").
2. Issue NAND read execute command byte (under "CLE").
3. Wait for the NAND device to complete accessing the block data by watching the ready signal.
4. Check for NAND timeout through "PSENSE".
5. Configure and enable the BCH block and read the NAND block data.
6. Disable the BCH block.
7. Descriptor NOP to allow NANDLOCK in the previous descriptor to the thread-safe.

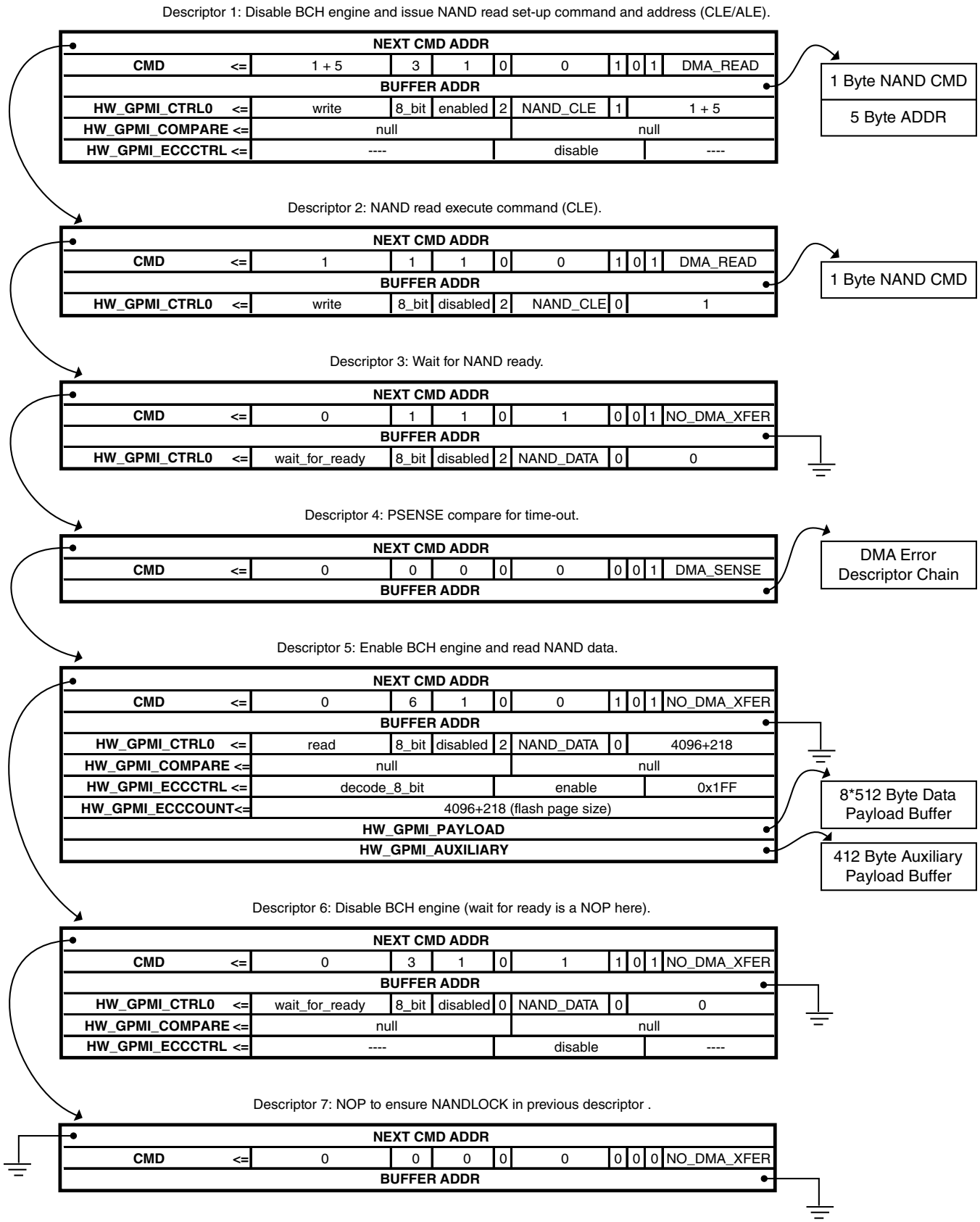


Figure 17-10. BCH Decode DMA Descriptor Chain

17.4.2.1 DMA Structure Code Example

The following sample code illustrates the coding for one read transaction, consisting of a seven DMA command structure chain for reading all 4096 bytes of payload data (eight 512-byte blocks) and 65 bytes of metadata with the associative parity bytes (8 * (18) + 9) from a 4K NAND page sitting on GPMI CS2.

```
//-----------------------------------------------------------------------------
// generic DMA/GPMI/ECC descriptor struct, order sensitive!
//-----------------------------------------------------------------------------
typedef struct {
  // DMA related fields
  unsigned int dma_nxtcmdar;
  unsigned int dma_cmd;
  unsigned int dma_bar;
  // GPMI related fields
  unsigned int gpmi_ctrl0;
  unsigned int gpmi_compare;
  unsigned int gpmi_eccctrl;
  unsigned int gpmi_ecccount;
  unsigned int gpmi_data_ptr;
  unsigned int gpmi_aux_ptr;
} GENERIC_DESCRIPTOR;
//-----------------------------------------------------------------------------
// allocate 7 descriptors for doing a NAND ECC Read
//-----------------------------------------------------------------------------
GENERIC_DESCRIPTOR read[7];
//-----------------------------------------------------------------------------
// DMA descriptor pointer to handle error conditions from psense checks
//-----------------------------------------------------------------------------
unsigned int * dma_error_handler;
//-----------------------------------------------------------------------------
// 7 byte NAND command and address buffer
// any alignment is ok, it is read by the GPMI DMA
//   byte 0 is read setup command
//   bytes 1-5 is the NAND address
//   byte 6 is read execute command
//-----------------------------------------------------------------------------
unsigned char nand_cmd_addr_buffer[7];
//-----------------------------------------------------------------------------
// 4096 byte payload buffer used for reads or writes
// needs to be word aligned
//-----------------------------------------------------------------------------
unsigned int read_payload_buffer[(4096/4)];
//-----------------------------------------------------------------------------
// 412 byte auxiliary buffer used for reads
// needs to be word aligned
//-----------------------------------------------------------------------------
unsigned int read_aux_buffer[(412/4)];
//-----------------------------------------------------------------------------
// Descriptor 1: issue NAND read setup command (CLE/ALE)
//-----------------------------------------------------------------------------
read[0].dma_nxtcmdar = &read[1];                          // point to the next descriptor
read[0].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (1 + 5) |  // 1 byte command, 5 byte address
                  BF_APBH_CHn_CMD_CMDWORDS      (3)     |  // send 3 words to the GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)     |  // wait for command to finish
                                                           //     before continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)     |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(0)     |
                  BF_APBH_CHn_CMD_NANDLOCK      (1)     |  // prevent other DMA channels from
                                                           //    taking over
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)     |
```

```
                  BF_APBH_CHn_CMD_CHAIN         (1)      |   // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, DMA_READ);  // read data from DMA, write to
NAND
read[0].dma_bar = &nand_cmd_addr_buffer;            // byte 0 read setup, bytes 1 - 5 NAND
address
// 3 words sent to the GPMI
read[0].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)     | // write to the NAND
                     BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)     |
                     BV_FLD(GPMI_CTRL0, LOCK_CS,      ENABLED)   |
                     BF_GPMI_CTRL0_CS                 (2)        | // must correspond to NAND
CS used
                     BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_CLE)  |
                     BF_GPMI_CTRL0_ADDRESS_INCREMENT  (1)        | // send command and address
                     BF_GPMI_CTRL0_XFER_COUNT         (1 + 5);    // 1 byte command, 5 byte
address
read[0].gpmi_compare = NULL;                        // field not used but necessary to set
eccctrl
read[0].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, DISABLE); // disable the ECC block
//-------------------------------------------------------------------------
// Descriptor 2: issue NAND read execute command (CLE)
//-------------------------------------------------------------------------
read[1].dma_nxtcmdar = &read[2];                          // point to the next descriptor
read[1].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (1)    |   // 1 byte read command
                  BF_APBH_CHn_CMD_CMDWORDS      (1)    |   // send 1 word to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)    |   // wait for command to finish
before
                                                         //    continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)    |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(0)    |
                  BF_APBH_CHn_CMD_NANDLOCK      (1)    |   // prevent other DMA channels from
                                                         //     taking over
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)    |
                  BF_APBH_CHn_CMD_CHAIN         (1)    |   // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, DMA_READ);     // read data from DMA, write
to NAND
read[1].dma_bar = &nand_cmd_addr_buffer[6];              // point to byte 6, read execute
command
// 1 word sent to the GPMI
read[1].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WRITE)     | // write to the NAND
                     BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)     |
                     BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)  |
                     BF_GPMI_CTRL0_CS                 (2)        | // must correspond to NAND
CS used
                     BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_CLE)  |
                     BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)        |
                     BF_GPMI_CTRL0_XFER_COUNT         (1);       // 1 byte command
//-------------------------------------------------------------------------
// Descriptor 3: wait for ready (DATA)
//-------------------------------------------------------------------------
read[2].dma_nxtcmdar = &read[3];                          // point to the next descriptor
read[2].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)      | // no dma transfer
                  BF_APBH_CHn_CMD_CMDWORDS      (1)      | // send 1 word to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)      | // wait for command to finish
before
                                                         //     continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)      |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(1)      | // wait for nand to be ready
                  BF_APBH_CHn_CMD_NANDLOCK      (0)      | // relinquish nand lock
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)      |
                  BF_APBH_CHn_CMD_CHAIN         (1)      | // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);   // no dma transfer
read[2].dma_bar = NULL;                                    // field not used
// 1 word sent to the GPMI
read[2].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, WAIT_FOR_READY) | // wait for NAND
ready
                     BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)          |
                     BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)       |
                     BF_GPMI_CTRL0_CS                 (2)             | // must correspond
to NAND CS used
                     BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA)      |
```

```
                    BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)             |
                    BF_GPMI_CTRL0_XFER_COUNT         (0);
//-----------------------------------------------------------------------------
// Descriptor 4: psense compare (time out check)
//-----------------------------------------------------------------------------
read[3].dma_nxtcmdar = &read[4];                                     // point to the next
descriptor
read[3].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)        |      // no dma transfer
                  BF_APBH_CHn_CMD_CMDWORDS      (0)        |      // no words sent to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (0)        |      // do not wait to continue
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)        |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(0)        |
                  BF_APBH_CHn_CMD_NANDLOCK      (0)        |
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)        |
                  BF_APBH_CHn_CMD_CHAIN         (1)        |      // follow chain to next
command
                  BV_FLD(APBH_CHn_CMD, COMMAND, DMA_SENSE);       // perform a sense check
read[3].dma_bar = dma_error_handler;                       // if sense check fails, branch to
error handler
//-----------------------------------------------------------------------------
// Descriptor 5: read 4K page plus 65 byte meta-data Nand data
//               and send it to ECC block (DATA)
//-----------------------------------------------------------------------------
read[4].dma_nxtcmdar = &read[5];                             // point to the next descriptor
read[4].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)    | // no dma transfer
                  BF_APBH_CHn_CMD_CMDWORDS      (6)    | // send 6 words to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)    | // wait for command to finish before
                                                         //       continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)    |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(0)    |
                  BF_APBH_CHn_CMD_NANDLOCK      (1)    | // prevent other DMA channels from
taking over
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)    | // ECC block generates BCH interrupt
                                                         //      on completion
                  BF_APBH_CHn_CMD_CHAIN         (1)    | // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);     // no DMA transfer,
                                                                  // ECC block handles
transfer
read[4].dma_bar = NULL;                                          // field not used
// 6 words sent to the GPMI
read[4].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, READ)       |  // read from the NAND
                     BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)      |
                     BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)   |
                     BF_GPMI_CTRL0_CS                 (2)         |  // must correspond to
NAND CS used
                     BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA)  |
                     BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)         |
                     BF_GPMI_CTRL0_XFER_COUNT         (4096+218);    // eight 512 byte data
blocks
                                                                     // metadata, and parity

read[4].gpmi_compare = NULL;                                 // field not used but necessary to set
eccctrl
// GPMI ECCCTRL PIO This launches the 4K byte transfer through BCH's
// bus master. Setting the ECC_ENABLE bit redirects the data flow
// within the GPMI so that read data flows to the BCH engine instead
// of flowing to the GPMI's DMA channel.
read[4].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ECC_CMD,    DECODE_8_BIT) |    // specify t = 8
mode
                       BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, ENABLE)       |    // enable ECC
module
                       BF_GPMI_ECCCTRL_BUFFER_MASK     (0X1FF);  // read all 8 data blocks
and 1 aux block
read[4].gpmi_ecccount = BF_GPMI_ECCCOUNT_COUNT(4096+218);        // specify number of bytes
                                                                 //        read from NAND
read[4].gpmi_data_ptr = &read_payload_buffer;                    // pointer for the 4K byte
                                                                 //    data area
read[4].gpmi_aux_ptr = &read_aux_buffer;                         // pointer for the 65 byte
aux area +
                                                                 // parity and syndrome
```

```
bytes for both
                                                                    // data and aux blocks.
//------------------------------------------------------------------------------
// Descriptor 6: disable ECC block
//------------------------------------------------------------------------------
read[5].dma_nxtcmdar = &read[6];                                    // point to the next descriptor
read[5].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)       | // no dma transfer
                  BF_APBH_CHn_CMD_CMDWORDS      (3)       | // send 3 words to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (1)       | // wait for command to finish
before
                                                            //     continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)       |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(1)       | // wait for nand to be ready
                  BF_APBH_CHn_CMD_NANDLOCK      (1)       | // need nand lock to be
                                                            // thread safe while turn-off BCH
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)       |
                  BF_APBH_CHn_CMD_CHAIN         (1)       | // follow chain to next command
                  BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);      // no dma transfer
read[5].dma_bar = NULL;                                             // field not used
// 3 words sent to the GPMI
read[5].gpmi_ctrl0 = BV_FLD(GPMI_CTRL0, COMMAND_MODE, READ)       |
                     BV_FLD(GPMI_CTRL0, WORD_LENGTH,  8_BIT)      |
                     BV_FLD(GPMI_CTRL0, LOCK_CS,      DISABLED)   |
                     BF_GPMI_CTRL0_CS                 (2)         | // must correspond to
NAND CS used
                     BV_FLD(GPMI_CTRL0, ADDRESS,      NAND_DATA)  |
                     BF_GPMI_CTRL0_ADDRESS_INCREMENT  (0)         |
                     BF_GPMI_CTRL0_XFER_COUNT         (0);
read[5].gpmi_compare = NULL;                                // field not used but necessary to set
eccctrl
read[5].gpmi_eccctrl = BV_FLD(GPMI_ECCCTRL, ENABLE_ECC, DISABLE);  // disable the ECC block
//------------------------------------------------------------------------------
// Descriptor 7: deassert nand lock
//------------------------------------------------------------------------------
read[6].dma_nxtcmdar = NULL;                                        // not used since this is last
descriptor
read[6].dma_cmd = BF_APBH_CHn_CMD_XFER_COUNT    (0)    |    // no dma transfer
                  BF_APBH_CHn_CMD_CMDWORDS      (0)    |    // no words sent to GPMI
                  BF_APBH_CHn_CMD_WAIT4ENDCMD   (0)    |    // wait for command to finish
before
                                                            //        continuing
                  BF_APBH_CHn_CMD_SEMAPHORE     (0)    |
                  BF_APBH_CHn_CMD_NANDWAIT4READY(0)    |
                  BF_APBH_CHn_CMD_NANDLOCK      (0)    |    // relinquish nand lock
                  BF_APBH_CHn_CMD_IRQONCMPLT    (0)    |    // BCH engine generates interrupt
                  BF_APBH_CHn_CMD_CHAIN         (0)    |    // terminate DMA chain processing
                  BV_FLD(APBH_CHn_CMD, COMMAND, NO_DMA_XFER);      // no dma transfer
read[6].dma_bar = NULL;                                             // field not used
```

17.4.2.2 Using the Decoder

As illustrated in Figure 17-10 and the sample code in DMA Structure Code Example :

- DMA descriptor 1 prepares the NAND for data read by using the GPMI to issue a NAND read setup command byte under CLE, then sends a 5-byte address under ALE. The BCH engine is not used for these commands.
- DMA descriptor 2 issues a one-byte read execute command to the NAND device that triggers its read access. The NAND then goes not ready.

- DMA descriptor 3 performs a wait for ready operation allowing the DMA chain to remain dormant until the NAND device completes its read access time.
- DMA descriptor 5 handles the reading and error correction of the NAND data. This command's PIOs activate the BCH engine to write the read NAND data to system memory and to process it for any errors that need to be corrected. This DMA descriptor contains two PIO values that are system memory addresses pointing to the PAYLOAD data area and to the AUXILIARY data area. These addresses are used by the BCH engine's AHB master to move data into system memory and to correct it. While this example is reading an entire 4K page—payload plus metadata—it is equally possible to read just one 512-byte payload block or just the uniquely protected metadata block in a single 7 DMA structure transfer.
- DMA descriptor 6 disables the BCH engine with the NANDLOCK asserted. This is necessary to ensure that the GPMI resource is not arbitrated to another DMA channel when multiple DMA channels are active concurrently.
- DMA descriptor 7 deasserts the NANDLOCK to free up the GPMI resource to another channel.

As the BCH block receives data from the GPMI:

- The decoder transforms the read NAND data block into a BCH code word and computes the codeword syndrome.
- If no errors are present, then the BCH block can immediately report back to firmware. This report is passed as the BCH_CTRL_COMPLETE_IRQ interrupt status bit and the associated status registers in BCH_STATUS0/1 registers.
- If an error is present, then the BCH block corrects the necessary data block or parity block bytes, if possible (not all errors are correctable).

As the BCH decoder reads the data and parity blocks, it records a special condition, i.e., that all of the bits of a payload data block or metadata block are one, including any associated parity bytes. The all-ones case for both parity and data indicates an erased block in the NAND device.

The BCH_STATUS0 register contains a 4-bit field that indicates the final status of the auxiliary block. A value of 0x0 indicates no errors found for a block.

- A value of 1 to 20 inclusive indicates that many correctable errors were found and fixed.
- A value of 0xFE indicates uncorrectable errors detected on the block.

- A value of 0xFF indicates that the block was in the special ALL ONES state and is therefore considered to be an ERASED block.
- All other values are disallowed by the hardware design.

Recall that up to eight NAND devices can have DMA chains in-flight at once, i.e. they can all be contending for access to the GPMI data bus. It is impossible to predict which NAND device will enter the BCH engine with a transfer first, because each chain includes a wait4ready command structure. As a result, firmware should look at the BCH_STATUS0_COMPLETED_CE bit field to determine which block is being reported in the status register. There is also a 16-bit HANDLE field in the GPMI_ECCCTRL register that is passed down the pipeline with each transaction. This handle field can be used to speed firmware's detection of which transaction is being reported.

These examples of reading and writing have focused on full page transfers of 4K page NAND devices. Other device configurations can be specified by changing the ECCOUNT field in the GPMI registers and reprogramming the BCH's FLASHnLAYOUTm registers.

The BCH and GPMI blocks are designed to be very efficient at reading single 512 (or 1024)-byte pages in one transaction. With no errors, the transaction takes less than 20 HCLKs longer than the time to read the raw data from the NAND.

To summarize, the APBH DMA command chain for a BCH decode operation is shown in Figure 17-10. Seven DMA command structures must be present for each NAND read transaction decoded by the BCH. The seven DMA command structures for multiple NAND read transaction blocks can be chained together to make larger units of work for the BCH, and each will produce an appropriate error report in the BCH PIO space. Multiple NAND devices can have such multiple chains scheduled. The results can come back out of order with respect to the multiple chains.

17.4.3 Interrupts

There are two interrupt sources used in processing BCH protected NAND read and write transfers.

Since all BCH operations are initiated by GPMI DMA command structures, the DMA completion interrupt for the GPMI is an important ISR. Both of the flow charts of Figure 17-6 and Figure 17-9 show the GPMI DMA complete ISR skeleton. In both reads and writes, the GPMI DMA completion interrupt is used to schedule work ***INTO*** the error correction pipeline. As the front end processing completes, the DMA interrupt is

generated and additional work, such as DMA chains, are passed to the GPMI DMA to keep it *fed*. For write operations, this is the only interrupt that is generated for processing the NAND write transfer.

For reads, however, two interrupts are needed. Every read is started by a GPMI DMA command chain and the front end queue is fed as described above. The back end of the read pipeline is drained by monitoring the BCH completion interrupt found int HW_BCH_CTRL_COMPLETE_IRQ.

An BCH transaction consists of reading or writing all of the blocks requested in the HW_GPMI_ECCCTRL_BUFFER_MASK bit field. As every read transaction completes, it posts the status of all of the blocks to the HW_BCH_STATUS0 and HW_BCH_STATUS1 registers and sets the completion interrupt. The five stages of the BCH read pipeline completes, one in the GPMI and four in the BCH, are independently stalled as they complete and try to deliver to the next stage in the data flow. Several of these stages can be skipped if no-errors are found or once an uncorrectable error is found in a block.

In any case, the final stage will stall if the status register is busy waiting for the CPU to take status register results. The hardware monitors the state of the HW_BCH_CTRL_COMPLETE_IRQ bit. If it is still set when the last pipeline stage is ready to post data, then the stage will stall. It follows that the next previous stage will stall when it is ready to hand off work to the final stage, and so on up the pipeline.

CAUTION

It is important that firmware read the STATUS0/1 results and save them before clearing the interrupt request bit. Otherwise, a transaction and its results could be completely lost.

17.5 Behavior During Reset

A soft reset (SFTRST) can take multiple clock periods to complete, so do NOT set CLKGATE when setting SFTRST.

The reset process gates the clocks automatically. The exemplary code is shown below.

```
// A soft reset can take multiple clocks to complete, so do NOT gate the
// clock when setting soft reset. The reset process will gate the clock
// automatically. Poll until this has happened before subsequently
// preparing soft-reset and clock gate
  BCH_CTRL_CLR(BM_BCH_CTRL_SFTRST);
  BCH_CTRL_CLR(BM_BCH_CTRL_CLKGATE);
// asserting soft-reset
  BCH_CTRL_SET(BM_BCH_CTRL_SFTRST);
// waiting for confirmation of soft-reset
  while (!BCH_CTRL.B.CLKGATE)
```

```
  {
// busy wait
  }
// Done.
  BCH_CTRL_CLR(BM_BCH_CTRL_SFTRST);
  BCH_CTRL_CLR(BM_BCH_CTRL_CLKGATE);
```

17.6 BCH Memory Map/Register Definition

BCH Hardware Register Format Summary

BCH memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_4000	Hardware BCH ECC Accelerator Control Register (BCH_CTRL)	32	R/W	C000_0000h	17.6.1/772
11_4004	Hardware BCH ECC Accelerator Control Register (BCH_CTRL_SET)	32	R/W	C000_0000h	17.6.1/772
11_4008	Hardware BCH ECC Accelerator Control Register (BCH_CTRL_CLR)	32	R/W	C000_0000h	17.6.1/772
11_400C	Hardware BCH ECC Accelerator Control Register (BCH_CTRL_TOG)	32	R/W	C000_0000h	17.6.1/772
11_4010	Hardware ECC Accelerator Status Register 0 (BCH_STATUS0)	32	R	0000_0010h	17.6.2/774
11_4014	Hardware ECC Accelerator Status Register 0 (BCH_STATUS0_SET)	32	R	0000_0010h	17.6.2/774
11_4018	Hardware ECC Accelerator Status Register 0 (BCH_STATUS0_CLR)	32	R	0000_0010h	17.6.2/774
11_401C	Hardware ECC Accelerator Status Register 0 (BCH_STATUS0_TOG)	32	R	0000_0010h	17.6.2/774
11_4020	Hardware ECC Accelerator Mode Register (BCH_MODE)	32	R/W	0000_0000h	17.6.3/776
11_4024	Hardware ECC Accelerator Mode Register (BCH_MODE_SET)	32	R/W	0000_0000h	17.6.3/776
11_4028	Hardware ECC Accelerator Mode Register (BCH_MODE_CLR)	32	R/W	0000_0000h	17.6.3/776
11_402C	Hardware ECC Accelerator Mode Register (BCH_MODE_TOG)	32	R/W	0000_0000h	17.6.3/776
11_4030	Hardware BCH ECC Loopback Encode Buffer Register (BCH_ENCODEPTR)	32	R/W	0000_0000h	17.6.4/776
11_4034	Hardware BCH ECC Loopback Encode Buffer Register (BCH_ENCODEPTR_SET)	32	R/W	0000_0000h	17.6.4/776
11_4038	Hardware BCH ECC Loopback Encode Buffer Register (BCH_ENCODEPTR_CLR)	32	R/W	0000_0000h	17.6.4/776
11_403C	Hardware BCH ECC Loopback Encode Buffer Register (BCH_ENCODEPTR_TOG)	32	R/W	0000_0000h	17.6.4/776

Table continues on the next page...

BCH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_4040	Hardware BCH ECC Loopback Data Buffer Register (BCH_DATAPTR)	32	R/W	0000_0000h	17.6.5/777
11_4044	Hardware BCH ECC Loopback Data Buffer Register (BCH_DATAPTR_SET)	32	R/W	0000_0000h	17.6.5/777
11_4048	Hardware BCH ECC Loopback Data Buffer Register (BCH_DATAPTR_CLR)	32	R/W	0000_0000h	17.6.5/777
11_404C	Hardware BCH ECC Loopback Data Buffer Register (BCH_DATAPTR_TOG)	32	R/W	0000_0000h	17.6.5/777
11_4050	Hardware BCH ECC Loopback Metadata Buffer Register (BCH_METAPTR)	32	R/W	0000_0000h	17.6.6/777
11_4054	Hardware BCH ECC Loopback Metadata Buffer Register (BCH_METAPTR_SET)	32	R/W	0000_0000h	17.6.6/777
11_4058	Hardware BCH ECC Loopback Metadata Buffer Register (BCH_METAPTR_CLR)	32	R/W	0000_0000h	17.6.6/777
11_405C	Hardware BCH ECC Loopback Metadata Buffer Register (BCH_METAPTR_TOG)	32	R/W	0000_0000h	17.6.6/777
11_4070	Hardware ECC Accelerator Layout Select Register (BCH_LAYOUTSELECT)	32	R/W	E4E4_E4E4h	17.6.7/778
11_4074	Hardware ECC Accelerator Layout Select Register (BCH_LAYOUTSELECT_SET)	32	R/W	E4E4_E4E4h	17.6.7/778
11_4078	Hardware ECC Accelerator Layout Select Register (BCH_LAYOUTSELECT_CLR)	32	R/W	E4E4_E4E4h	17.6.7/778
11_407C	Hardware ECC Accelerator Layout Select Register (BCH_LAYOUTSELECT_TOG)	32	R/W	E4E4_E4E4h	17.6.7/778
11_4080	Hardware BCH ECC Flash 0 Layout 0 Register (BCH_FLASH0LAYOUT0)	32	R/W	070A_4080h	17.6.8/779
11_4084	Hardware BCH ECC Flash 0 Layout 0 Register (BCH_FLASH0LAYOUT0_SET)	32	R/W	070A_4080h	17.6.8/779
11_4088	Hardware BCH ECC Flash 0 Layout 0 Register (BCH_FLASH0LAYOUT0_CLR)	32	R/W	070A_4080h	17.6.8/779
11_408C	Hardware BCH ECC Flash 0 Layout 0 Register (BCH_FLASH0LAYOUT0_TOG)	32	R/W	070A_4080h	17.6.8/779
11_4090	Hardware BCH ECC Flash 0 Layout 1 Register (BCH_FLASH0LAYOUT1)	32	R/W	10DA_4080h	17.6.9/781
11_4094	Hardware BCH ECC Flash 0 Layout 1 Register (BCH_FLASH0LAYOUT1_SET)	32	R/W	10DA_4080h	17.6.9/781
11_4098	Hardware BCH ECC Flash 0 Layout 1 Register (BCH_FLASH0LAYOUT1_CLR)	32	R/W	10DA_4080h	17.6.9/781
11_409C	Hardware BCH ECC Flash 0 Layout 1 Register (BCH_FLASH0LAYOUT1_TOG)	32	R/W	10DA_4080h	17.6.9/781
11_40A0	Hardware BCH ECC Flash 1 Layout 0 Register (BCH_FLASH1LAYOUT0)	32	R/W	070A_4080h	17.6.10/ 782
11_40A4	Hardware BCH ECC Flash 1 Layout 0 Register (BCH_FLASH1LAYOUT0_SET)	32	R/W	070A_4080h	17.6.10/ 782

Table continues on the next page...

BCH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_40A8	Hardware BCH ECC Flash 1 Layout 0 Register (BCH_FLASH1LAYOUT0_CLR)	32	R/W	070A_4080h	17.6.10/ 782
11_40AC	Hardware BCH ECC Flash 1 Layout 0 Register (BCH_FLASH1LAYOUT0_TOG)	32	R/W	070A_4080h	17.6.10/ 782
11_40B0	Hardware BCH ECC Flash 1 Layout 1 Register (BCH_FLASH1LAYOUT1)	32	R/W	10DA_4080h	17.6.11/ 784
11_40B4	Hardware BCH ECC Flash 1 Layout 1 Register (BCH_FLASH1LAYOUT1_SET)	32	R/W	10DA_4080h	17.6.11/ 784
11_40B8	Hardware BCH ECC Flash 1 Layout 1 Register (BCH_FLASH1LAYOUT1_CLR)	32	R/W	10DA_4080h	17.6.11/ 784
11_40BC	Hardware BCH ECC Flash 1 Layout 1 Register (BCH_FLASH1LAYOUT1_TOG)	32	R/W	10DA_4080h	17.6.11/ 784
11_40C0	Hardware BCH ECC Flash 2 Layout 0 Register (BCH_FLASH2LAYOUT0)	32	R/W	070A_4080h	17.6.12/ 785
11_40C4	Hardware BCH ECC Flash 2 Layout 0 Register (BCH_FLASH2LAYOUT0_SET)	32	R/W	070A_4080h	17.6.12/ 785
11_40C8	Hardware BCH ECC Flash 2 Layout 0 Register (BCH_FLASH2LAYOUT0_CLR)	32	R/W	070A_4080h	17.6.12/ 785
11_40CC	Hardware BCH ECC Flash 2 Layout 0 Register (BCH_FLASH2LAYOUT0_TOG)	32	R/W	070A_4080h	17.6.12/ 785
11_40D0	Hardware BCH ECC Flash 2 Layout 1 Register (BCH_FLASH2LAYOUT1)	32	R/W	10DA_4080h	17.6.13/ 787
11_40D4	Hardware BCH ECC Flash 2 Layout 1 Register (BCH_FLASH2LAYOUT1_SET)	32	R/W	10DA_4080h	17.6.13/ 787
11_40D8	Hardware BCH ECC Flash 2 Layout 1 Register (BCH_FLASH2LAYOUT1_CLR)	32	R/W	10DA_4080h	17.6.13/ 787
11_40DC	Hardware BCH ECC Flash 2 Layout 1 Register (BCH_FLASH2LAYOUT1_TOG)	32	R/W	10DA_4080h	17.6.13/ 787
11_40E0	Hardware BCH ECC Flash 3 Layout 0 Register (BCH_FLASH3LAYOUT0)	32	R/W	070A_4080h	17.6.14/ 788
11_40E4	Hardware BCH ECC Flash 3 Layout 0 Register (BCH_FLASH3LAYOUT0_SET)	32	R/W	070A_4080h	17.6.14/ 788
11_40E8	Hardware BCH ECC Flash 3 Layout 0 Register (BCH_FLASH3LAYOUT0_CLR)	32	R/W	070A_4080h	17.6.14/ 788
11_40EC	Hardware BCH ECC Flash 3 Layout 0 Register (BCH_FLASH3LAYOUT0_TOG)	32	R/W	070A_4080h	17.6.14/ 788
11_40F0	Hardware BCH ECC Flash 3 Layout 1 Register (BCH_FLASH3LAYOUT1)	32	R/W	10DA_4080h	17.6.15/ 790
11_40F4	Hardware BCH ECC Flash 3 Layout 1 Register (BCH_FLASH3LAYOUT1_SET)	32	R/W	10DA_4080h	17.6.15/ 790
11_40F8	Hardware BCH ECC Flash 3 Layout 1 Register (BCH_FLASH3LAYOUT1_CLR)	32	R/W	10DA_4080h	17.6.15/ 790
11_40FC	Hardware BCH ECC Flash 3 Layout 1 Register (BCH_FLASH3LAYOUT1_TOG)	32	R/W	10DA_4080h	17.6.15/ 790

Table continues on the next page...

BCH memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_4100	Hardware BCH ECC Debug Register0 (BCH_DEBUG0)	32	R/W	0000_0000h	17.6.16/ 791
11_4104	Hardware BCH ECC Debug Register0 (BCH_DEBUG0_SET)	32	R/W	0000_0000h	17.6.16/ 791
11_4108	Hardware BCH ECC Debug Register0 (BCH_DEBUG0_CLR)	32	R/W	0000_0000h	17.6.16/ 791
11_410C	Hardware BCH ECC Debug Register0 (BCH_DEBUG0_TOG)	32	R/W	0000_0000h	17.6.16/ 791
11_4110	KES Debug Read Register (BCH_DBGKESREAD)	32	R	0000_0000h	17.6.17/ 793
11_4114	KES Debug Read Register (BCH_DBGKESREAD_SET)	32	R	0000_0000h	17.6.17/ 793
11_4118	KES Debug Read Register (BCH_DBGKESREAD_CLR)	32	R	0000_0000h	17.6.17/ 793
11_411C	KES Debug Read Register (BCH_DBGKESREAD_TOG)	32	R	0000_0000h	17.6.17/ 793
11_4120	Chien Search Debug Read Register (BCH_DBGCSFEREAD)	32	R	0000_0000h	17.6.18/ 793
11_4124	Chien Search Debug Read Register (BCH_DBGCSFEREAD_SET)	32	R	0000_0000h	17.6.18/ 793
11_4128	Chien Search Debug Read Register (BCH_DBGCSFEREAD_CLR)	32	R	0000_0000h	17.6.18/ 793
11_412C	Chien Search Debug Read Register (BCH_DBGCSFEREAD_TOG)	32	R	0000_0000h	17.6.18/ 793
11_4130	Syndrome Generator Debug Read Register (BCH_DBGSYNDGENREAD)	32	R	0000_0000h	17.6.19/ 794
11_4134	Syndrome Generator Debug Read Register (BCH_DBGSYNDGENREAD_SET)	32	R	0000_0000h	17.6.19/ 794
11_4138	Syndrome Generator Debug Read Register (BCH_DBGSYNDGENREAD_CLR)	32	R	0000_0000h	17.6.19/ 794
11_413C	Syndrome Generator Debug Read Register (BCH_DBGSYNDGENREAD_TOG)	32	R	0000_0000h	17.6.19/ 794
11_4140	Bus Master and ECC Controller Debug Read Register (BCH_DBGAHBMREAD)	32	R	0000_0000h	17.6.20/ 794
11_4144	Bus Master and ECC Controller Debug Read Register (BCH_DBGAHBMREAD_SET)	32	R	0000_0000h	17.6.20/ 794
11_4148	Bus Master and ECC Controller Debug Read Register (BCH_DBGAHBMREAD_CLR)	32	R	0000_0000h	17.6.20/ 794
11_414C	Bus Master and ECC Controller Debug Read Register (BCH_DBGAHBMREAD_TOG)	32	R	0000_0000h	17.6.20/ 794
11_4150	Block Name Register (BCH_BLOCKNAME)	32	R	2048_4342h	17.6.21/ 795
11_4154	Block Name Register (BCH_BLOCKNAME_SET)	32	R	2048_4342h	17.6.21/ 795

Table continues on the next page...

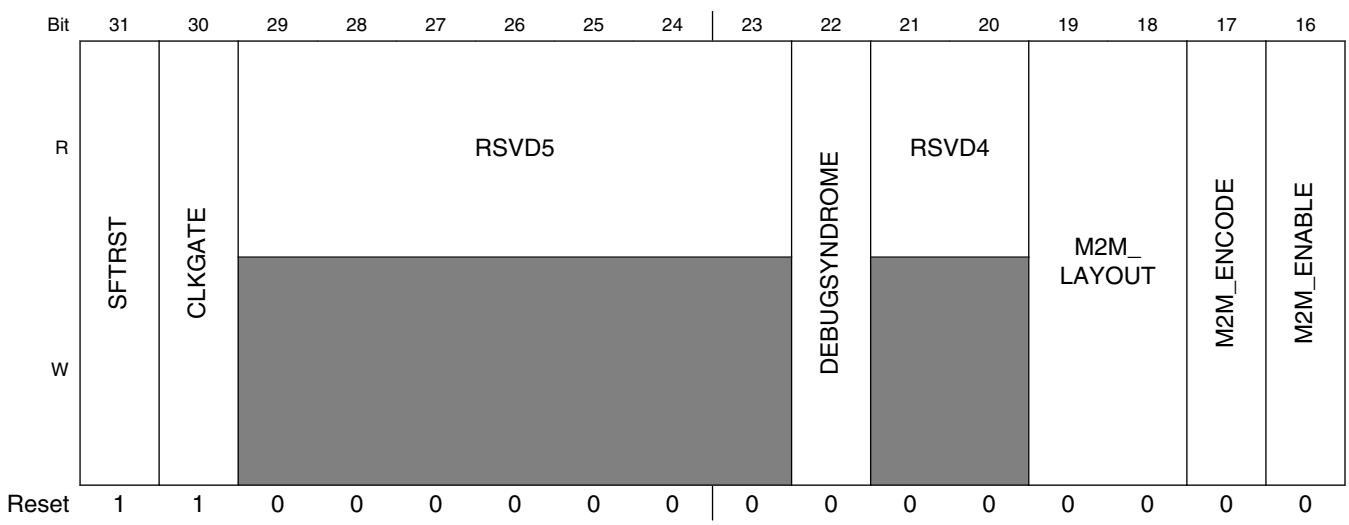
BCH memory map (continued)

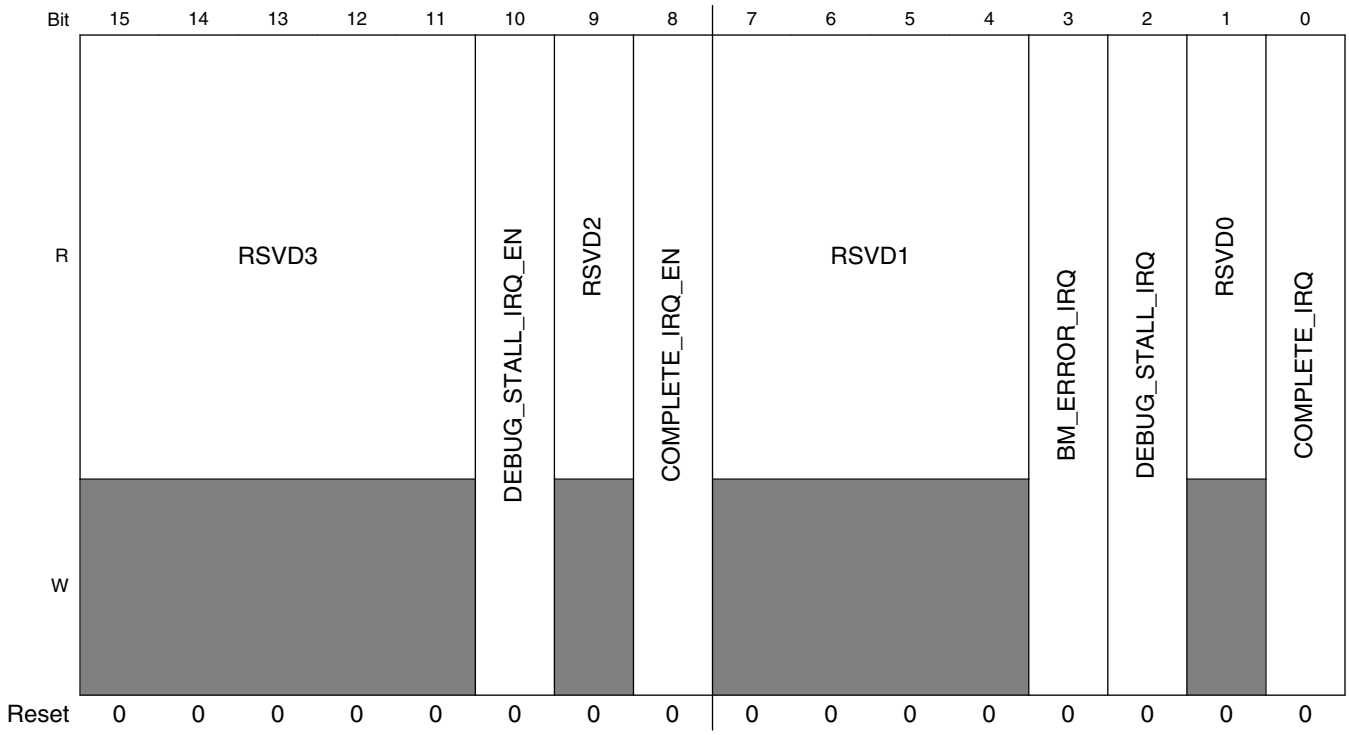
Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_4158	Block Name Register (BCH_BLOCKNAME_CLR)	32	R	2048_4342h	17.6.21/ 795
11_415C	Block Name Register (BCH_BLOCKNAME_TOG)	32	R	2048_4342h	17.6.21/ 795
11_4160	BCH Version Register (BCH_VERSION)	32	R	0100_0000h	17.6.22/ 795
11_4164	BCH Version Register (BCH_VERSION_SET)	32	R	0100_0000h	17.6.22/ 795
11_4168	BCH Version Register (BCH_VERSION_CLR)	32	R	0100_0000h	17.6.22/ 795
11_416C	BCH Version Register (BCH_VERSION_TOG)	32	R	0100_0000h	17.6.22/ 795

17.6.1 Hardware BCH ECC Accelerator Control Register (BCH_CTRL*n*)

The BCH CTRL provides overall control of the hardware ECC accelerator

Address: 11_4000h base + 0h offset + (4d × i), where i=0d to 3d





BCH_CTRL*n* field descriptions

Field	Description
31 SFTRST	Set this bit to zero to enable normal BCH operation. Set this bit to one (default) to disable clocking with the BCH and hold it in its reset (lowest power) state. This bit can be turned on and then off to reset the BCH block to its default state. This bit resets all state machines except for the AHB master state machine 0x0 **RUN** — Allow BCH to operate normally. 0x1 **RESET** — Hold BCH in reset.
30 CLKGATE	This bit must be set to zero for normal operation. When set to one it gates off the clocks to the block. 0x0 **RUN** — Allow BCH to operate normally. 0x1 **NO_CLKS** — Do not clock BCH gates in order to minimize power consumption.
29–23 RSVD5	Reserved, always set this bit to zero.
22 DEBUGSYNDROME	(For debug purposes only). Enable write of computed syndromes to memory on BCH decode operations. Computed syndromes will be written to the auxiliary buffer after the status block. Syndromes will be written as padded 16-bit values.
21–20 RSVD4	Reserved, always set these bits to zero.
19–18 M2M_LAYOUT	Selects the flash page format for memory-to-memory operations.
17 M2M_ENCODE	Selects encode (parity generation) or decode (correction) mode for memory-to-memory operations.
16 M2M_ENABLE	NOTE! WRITING THIS BIT INITIATES A MEMORY-TO-MEMORY OPERATION. The BCH module must be inactive (not processing data from the GPMI) when this bit is set. The M2M_ENCODE and M2M_LAYOUT bits as well as the ENCODEPTR, DATAPTR, and METAPTR registers are used for memory-to-memory operations and must be correctly programmed before writing this bit.

Table continues on the next page...

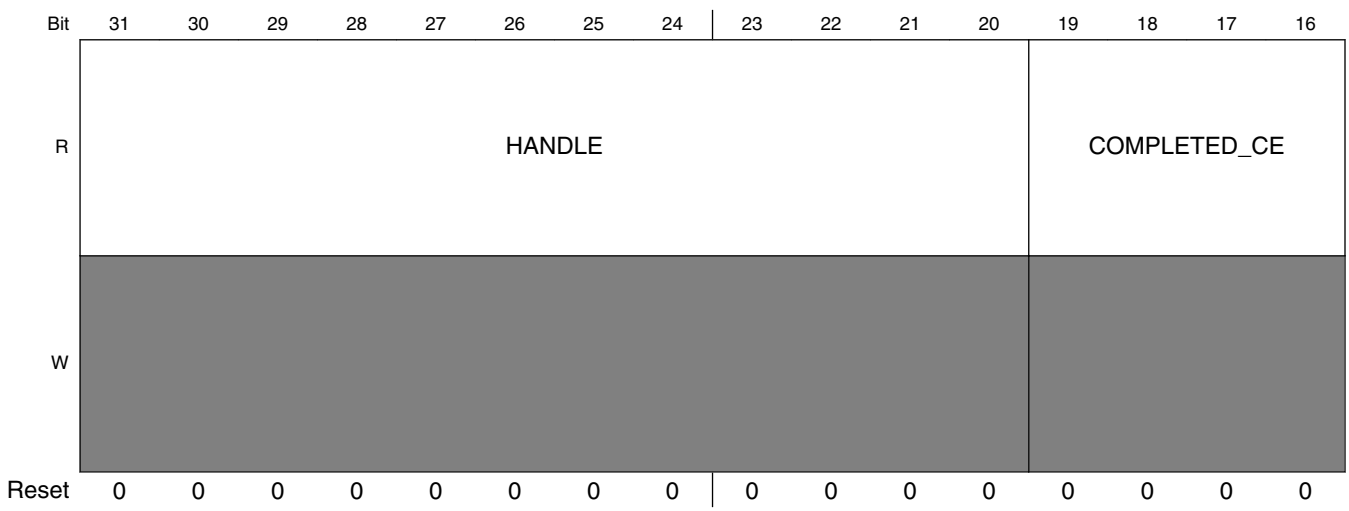
BCH_CTRL*n* field descriptions (continued)

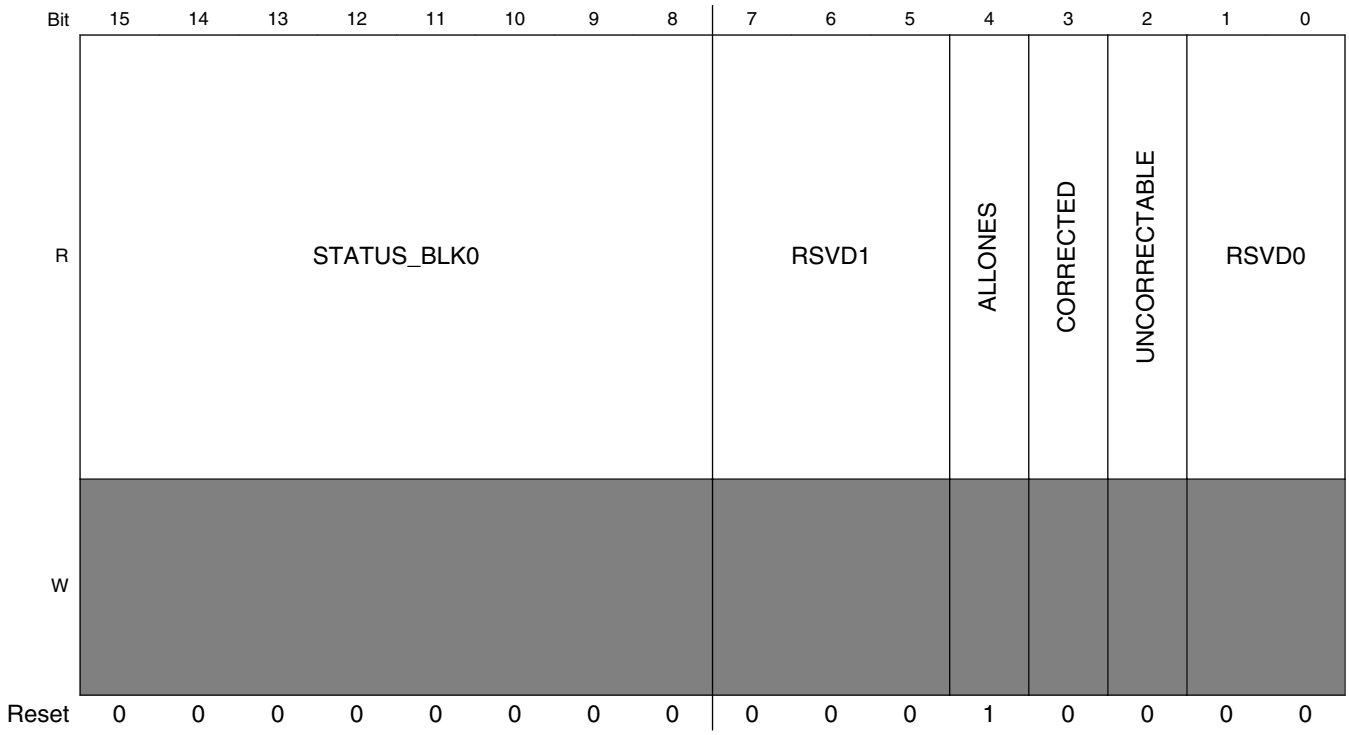
Field	Description
15–11 RSVD3	Reserved, always set these bits to zero.
10 DEBUG_STALL_ IRQ_EN	1 = interrupt on debug stall mode is enabled. The irq is raised on every block
9 RSVD2	Reserved, always set these bits to zero.
8 COMPLETE_IRQ_ EN	1 = interrupt on completion of correction is enabled.
7–4 RSVD1	Reserved, always set these bits to zero.
3 BM_ERROR_IRQ	AHB Bus interface Error Interrupt Status. Write a one to the SCT clear address to clear the interrupt status bit.
2 DEBUG_STALL_IRQ	DEBUG STALL Interrupt Status. Write a one to the SCT clear address to clear the interrupt status bit.
1 RSVD0	Reserved, always set these bits to zero.
0 COMPLETE_IRQ	This bit indicates the state of the external interrupt line. Write a one to the SCT clear address to clear the interrupt status bit. NOTE: subsequent ECC completions will be held off as long as this bit is set. Be sure to read the data from BCH_STATUS0,1 before clearing this interrupt bit.

17.6.2 Hardware ECC Accelerator Status Register 0 (BCH_STATUS0*n*)

The BCH STAT register provides visibility into the run-time status of the BCH and status information when processing is complete. It provides overall status of the hardware ECC accelerator.

Address: 11_4000h base + 10h offset + (4d × i), where i=0d to 3d





BCH_STATUS0*n* field descriptions

Field	Description
31–20 HANDLE	Software supplies a 12 bit handle for this transfer as part of the GPMI DMA PIO operation that started the transaction. That handle passes down the pipeline and ends up here at the time the BCH interrupt is signaled.
19–16 COMPLETED_CE	This is the chip enable number corresponding to the NAND device from which this data came.
15–8 STATUS_BLK0	Count of symbols in error during processing of first block of flash (metadata block). The number of errors reported will be in the range of 0 to the ECC correction level for block 0. 0x00 **ZERO** — No errors found on block. 0x01 **ERROR1** — One error found on block. 0x02 **ERROR2** — One errors found on block. 0x03 **ERROR3** — One errors found on block. 0x04 **ERROR4** — One errors found on block. 0xFE **UNCORRECTABLE** — Block exhibited uncorrectable errors. 0xFF **ERASED** — Page is erased.
7–5 RSVD1	Reserved, always set these bits to zero.
4 ALLONES	1 = All data bits of this transaction are ONE.
3 CORRECTED	1 = At least one correctable error encountered during last processing cycle.
2 UNCORRECTABLE	1 = Uncorrectable error encountered during last processing cycle.
RSVD0	Reserved, always set these bits to zero.

17.6.3 Hardware ECC Accelerator Mode Register (BCH_MODE*n*)

The BCH MODE register provides additional mode controls.

Contains additional global mode controls for the BCH engine.

Address: 11_4000h base + 20h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RSVD																								ERASE_THRESHOLD							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_MODE*n* field descriptions

Field	Description
31–8 RSVD	Reserved, always set these bits to zero.
ERASE_ THRESHOLD	This value indicates the maximum number of zero bits on a flash page for it to be considered erased. For SLC NAND devices, this value should be programmed to 0 (meaning that the entire page should consist of bytes of 0xFF. For MLC NAND devices, bit errors may occur on reads (even on blank pages), so this threshold can be used to tune the erased page checking algorithm.

17.6.4 Hardware BCH ECC Loopback Encode Buffer Register (BCH_ENCODEPTR*n*)

When performing memory to memory operations, indicates the address of the encode buffer. This register should be programmed before writing a 1 to the M2M_ENABLE bit in the CTRL register.

For memory to memory operations, this register is used as the pointer to the encoded data, which is an output when encoding and an input while decoding.

Address: 11_4000h base + 30h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ADDR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_ENCODEPTR*n* field descriptions

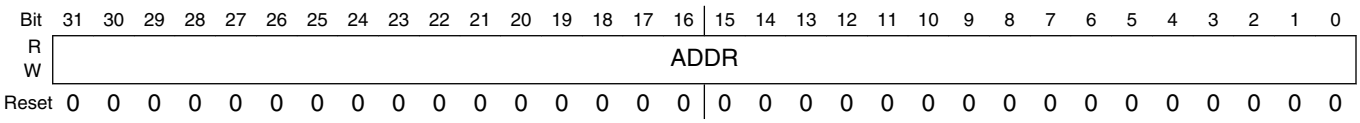
Field	Description
ADDR	Address pointer to encode buffer. This is the source for decode operations and the destination for encode operations. This value must be aligned on a 4 byte boundary.

17.6.5 Hardware BCH ECC Loopback Data Buffer Register (BCH_DATAPTR*n*)

When performing memory to memory operations, indicates the address of the data buffer.

For memory to memory operations, this register is used as the pointer to the data to encode or the destination buffer for decode operations.

Address: 11_4000h base + 40h offset + (4d × i), where i=0d to 3d



BCH_DATAPTR*n* field descriptions

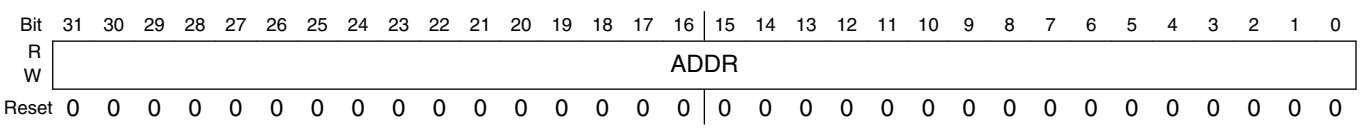
Field	Description
ADDR	Address pointer to data buffer. This is the source for encode operations and the destination for decode operations. This register should be programmed before writing a 1 to the M2M_ENABLE bit in the CTRL register. This value must be aligned on a 4 byte boundary.

17.6.6 Hardware BCH ECC Loopback Metadata Buffer Register (BCH_METAPTR*n*)

When performing memory to memory operations, indicates the address of the metadata buffer.

For memory to memory operations, this register is used as the pointer to the metadata to encode or the extracted metadata for decode operations.

Address: 11_4000h base + 50h offset + (4d × i), where i=0d to 3d



BCH_METAPTR*n* field descriptions

Field	Description
ADDR	Address pointer to metadata buffer. This is the source for encode metadata read operations and the destination for metadata decode operations. This register should be programmed before writing a 1 to the M2M_ENABLE bit in the CTRL register. This value must be aligned on a 4 byte boundary.

17.6.7 Hardware ECC Accelerator Layout Select Register (BCH_LAYOUTSELECT*n*)

The BCH LAYOUTSELECT register provides a mapping of chip selects to layout registers.

When the BCH engine receives a request to process a data block from the GPMI interface, it will use this register to map the incoming chip select to one of the four possible flash layout registers

Address: 11_4000h base + 70h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CS15_ SELECT		CS14_ SELECT		CS13_ SELECT		CS12_ SELECT		CS11_ SELECT		CS10_ SELECT		CS9_ SELECT		CS8_ SELECT	
Reset	1	1	1	0	0	1	0	0	1	1	1	0	0	1	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CS7_ SELECT		CS6_ SELECT		CS5_ SELECT		CS4_ SELECT		CS3_ SELECT		CS2_ SELECT		CS1_ SELECT		CS0_ SELECT	
Reset	1	1	1	0	0	1	0	0	1	1	1	0	0	1	0	0

BCH_LAYOUTSELECT*n* field descriptions

Field	Description
31–30 CS15_SELECT	Selects which layout is used for chip select 15.
29–28 CS14_SELECT	Selects which layout is used for chip select 14.
27–26 CS13_SELECT	Selects which layout is used for chip select 13.
25–24 CS12_SELECT	Selects which layout is used for chip select 12.
23–22 CS11_SELECT	Selects which layout is used for chip select 11.
21–20 CS10_SELECT	Selects which layout is used for chip select 10.
19–18 CS9_SELECT	Selects which layout is used for chip select 9.
17–16 CS8_SELECT	Selects which layout is used for chip select 8.
15–14 CS7_SELECT	Selects which layout is used for chip select 7.
13–12 CS6_SELECT	Selects which layout is used for chip select 6.
11–10 CS5_SELECT	Selects which layout is used for chip select 5.

Table continues on the next page...

BCH_LAYOUTSELECT*n* field descriptions (continued)

Field	Description
9–8 CS4_SELECT	Selects which layout is used for chip select 4.
7–6 CS3_SELECT	Selects which layout is used for chip select 3.
5–4 CS2_SELECT	Selects which layout is used for chip select 2.
3–2 CS1_SELECT	Selects which layout is used for chip select 1.
CS0_SELECT	Selects which layout is used for chip select 0.

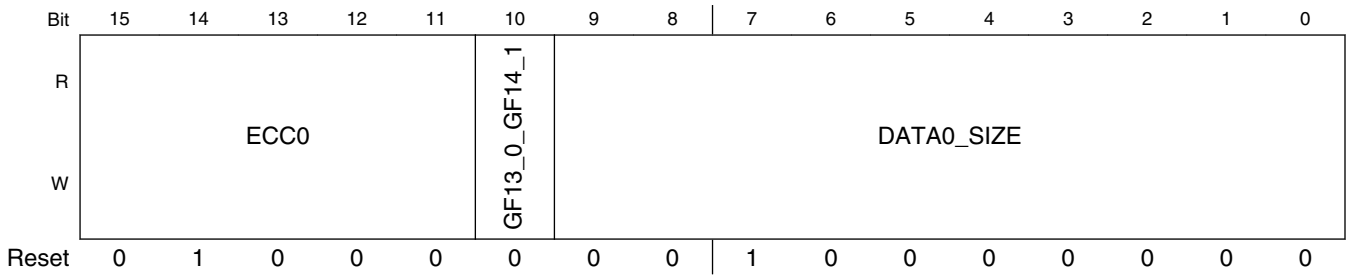
17.6.8 Hardware BCH ECC Flash 0 Layout 0 Register (BCH_FLASH0LAYOUT0*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH0LAYOUT1 register to control the format for the devices selecting layout 0 in the LAYOUTSELECT register.

Each pair of layout registers describes one of four supported flash configurations. Software should program the LAYOUTSELECT register for each supported GPMI chip select to select from one of the four layout values. Each pair of registers contains settings that are used by the BCH block while reading/writing the flash page to control data, metadata, and flash page sizes as well as the ECC correction level. The first block written to flash can be programmed to have different ECC, metadata, and data sizes from subsequent data blocks on the device. In addition, the number of blocks stored on a page of flash is not fixed, but instead is determined by the number of bytes consumed by the initial (block 0) and subsequent data blocks. See the BCH programming reference manual for more information on setting up the flash layout registers.

Address: 11_4000h base + 80h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	NBLOCKS								META_SIZE							
Reset	0	0	0	0	0	1	1	1	0	0	0	0	1	0	1	0



BCH_FLASH0LAYOUT0*n* field descriptions

Field	Description
31–24 NBLOCKS	Number of subsequent blocks on the flash page (excluding the data0 block). A value of 0 indicates that only the DATA0 block is present and a value of 8 indicates that 8 subsequent blocks are present for a total of 9 blocks on the flash (including the DATA0 block). Any values from 0 to 255 are supported by the hardware.
23–16 META_SIZE	Indicates the size of the metadata (in bytes) to be stored on a flash page. The BCH design support from 0 to 255 bytes for metadata -- if set to zero, no metadata will be stored. Metadata is stored before the associated data in block 0. If the DATA0_SIZE field is programmed to a zero, then metadata effectively be stored with its own parity. When both the metadata and data0 fields are programmed with non-zero values, the first block will contain both portions of data and will be covered by a single parity block.
15–11 ECC0	Indicates the ECC level for the first block on the flash page. The first block covers metadata plus the associated data from the DATA0_SIZE field. 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed 0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATA0_SIZE	Indicates the size of the data 0 block (in DWORDS / four bytes) to be stored on the flash page. If set to zero, the first block will only contain metadata.

17.6.9 Hardware BCH ECC Flash 0 Layout 1 Register (BCH_FLASH0LAYOUT1*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH0LAYOUT0 register to control the format for the device selecting layout 0 in the LAYOUTSELECT register.

Address: 11_4000h base + 90h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	PAGE_SIZE															
Reset	0	0	0	1	0	0	0	0	1	1	0	1	1	0	1	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ECCN					GF13_0_GF14_1	DATAN_SIZE									
Reset	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0

BCH_FLASH0LAYOUT1*n* field descriptions

Field	Description
31–16 PAGE_SIZE	Indicates the total size of the flash page (in bytes). This should be set to the page size including spare area. The page size is programmable to accomodate different flash configurations that may be available in the future.
15–11 ECCN	Indicates the ECC level for the subsequent blocks on the flash page (blocks 1-n). Subsequent blocks only contain data (no metadata). 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed

Table continues on the next page...

BCH_FLASH0LAYOUT1*n* field descriptions (continued)

Field	Description
	0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATAN_SIZE	Indicates the size of the subsequent data blocks (in DWORDS / four bytes) to be stored on the flash page. The size of subsequent data blocks does not have to match the data size for block 0, which is important when metadata is stored separately or for balancing the amount of data stored in each block.

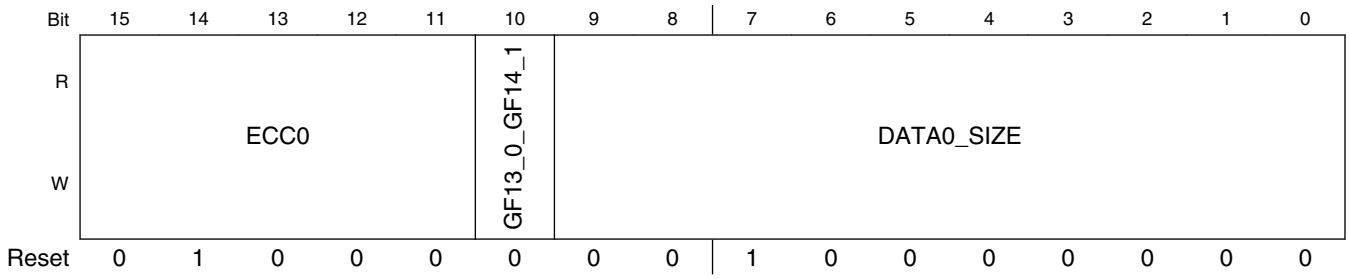
17.6.10 Hardware BCH ECC Flash 1 Layout 0 Register (BCH_FLASH1LAYOUT0*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH1LAYOUT1 register to control the format for the devices selecting layout 1 in the LAYOUTSELECT register.

Each pair of layout registers describes one of four supported flash configurations. Software should program the LAYOUTSELECT register for each supported GPMI chip select to select from one of the four layout values. Each pair of registers contains settings that are used by the BCH block while reading/writing the flash page to control data, metadata, and flash page sizes as well as the ECC correction level. The first block written to flash can be programmed to have different ECC, metadata, and data sizes from subsequent data blocks on the device. In addition, the number of blocks stored on a page of flash is not fixed, but instead is determined by the number of bytes consumed by the initial (block 0) and subsequent data blocks. See the BCH programming reference manual for more information on setting up the flash layout registers.

Address: 11_4000h base + A0h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	NBLOCKS								META_SIZE							
W																
Reset	0	0	0	0	0	1	1	1	0	0	0	0	1	0	1	0



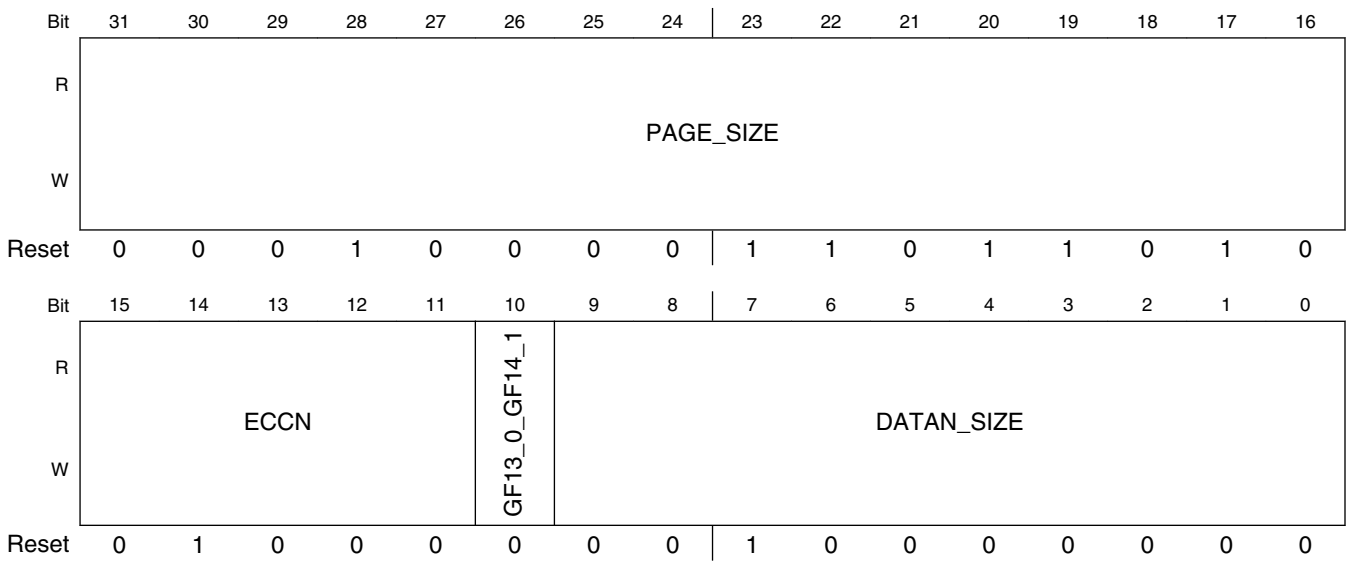
BCH_FLASH1LAYOUT0*n* field descriptions

Field	Description
31–24 NBLOCKS	Number of subsequent blocks on the flash page (excluding the data0 block). A value of 0 indicates that only the DATA0 block is present and a value of 8 indicates that 8 subsequent blocks are present for a total of 9 blocks on the flash (including the DATA0 block). Any values from 0 to 255 are supported by the hardware.
23–16 META_SIZE	Indicates the size of the metadata (in bytes) to be stored on a flash page. The BCH design support from 0 to 255 bytes for metadata -- if set to zero, no metadata will be stored. Metadata is stored before the associated data in block 0. If the DATA0_SIZE field is programmed to a zero, then metadata effectively be stored with its own parity. When both the metadata and data0 fields are programmed with non-zero values, the first block will contain both portions of data and will be covered by a single parity block.
15–11 ECC0	Indicates the ECC level for the first block on the flash page. The first block covers metadata plus the associated data from the DATA0_SIZE field. 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed 0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATA0_SIZE	Indicates the size of the data 0 block (in DWORDS / four bytes) to be stored on the flash page. If set to zero, the first block will only contain metadata.

17.6.11 Hardware BCH ECC Flash 1 Layout 1 Register (BCH_FLASH1LAYOUT1*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH1LAYOUT0 register to control the format for the device selecting layout 1 in the LAYOUTSELECT register.

Address: 11_4000h base + B0h offset + (4d × i), where i=0d to 3d



BCH_FLASH1LAYOUT1*n* field descriptions

Field	Description
31–16 PAGE_SIZE	Indicates the total size of the flash page (in bytes). This should be set to the page size including spare area. The page size is programmable to accomodate different flash configurations that may be available in the future.
15–11 ECCN	Indicates the ECC level for the subsequent blocks on the flash page (blocks 1-n). Subsequent blocks only contain data (no metadata). 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed

Table continues on the next page...

BCH_FLASH1LAYOUT1*n* field descriptions (continued)

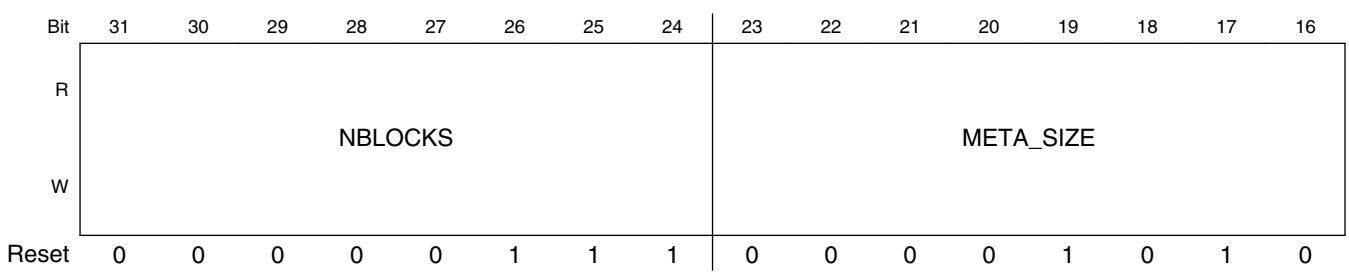
Field	Description
	0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATAN_SIZE	Indicates the size of the subsequent data blocks (in DWORDS / four bytes) to be stored on the flash page. The size of subsequent data blocks does not have to match the data size for block 0, which is important when metadata is stored separately or for balancing the amount of data stored in each block.

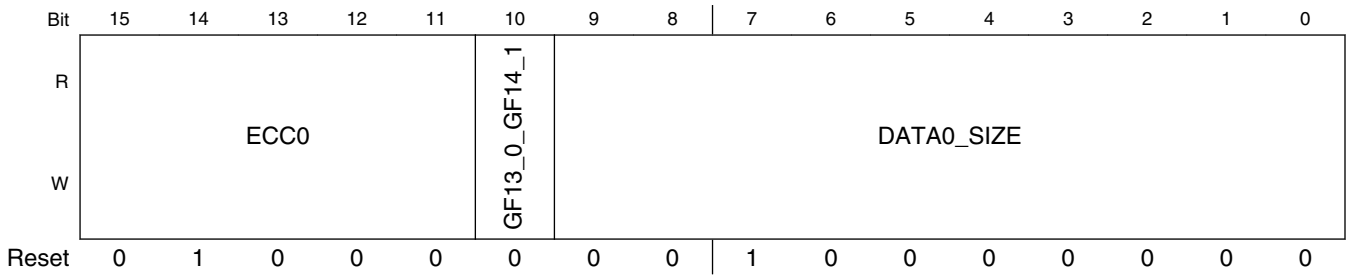
17.6.12 Hardware BCH ECC Flash 2 Layout 0 Register (BCH_FLASH2LAYOUT0*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH2LAYOUT1 register to control the format for the devices selecting layout 2 in the LAYOUTSELECT register.

Each pair of layout registers describes one of four supported flash configurations. Software should program the LAYOUTSELECT register for each supported GPMI chip select to select from one of the four layout values. Each pair of registers contains settings that are used by the BCH block while reading/writing the flash page to control data, metadata, and flash page sizes as well as the ECC correction level. The first block written to flash can be programmed to have different ECC, metadata, and data sizes from subsequent data blocks on the device. In addition, the number of blocks stored on a page of flash is not fixed, but instead is determined by the number of bytes consumed by the initial (block 0) and subsequent data blocks. See the BCH programming reference manual for more information on setting up the flash layout registers.

Address: 11_4000h base + C0h offset + (4d × i), where i=0d to 3d





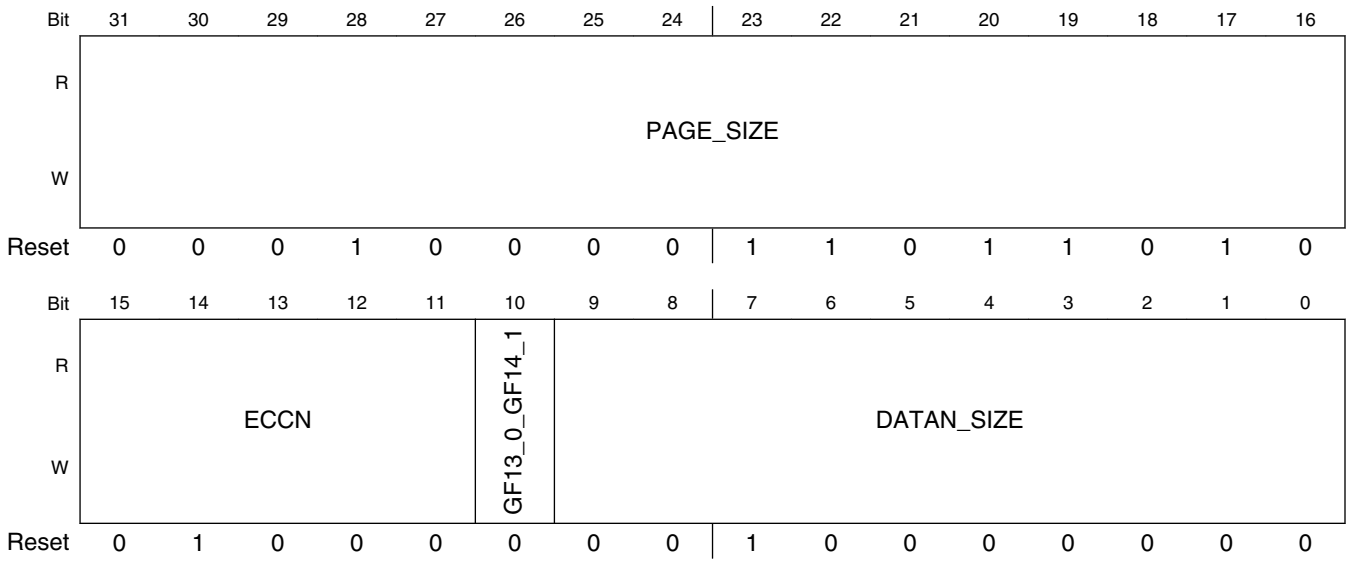
BCH_FLASH2LAYOUT0*n* field descriptions

Field	Description
31–24 NBLOCKS	Number of subsequent blocks on the flash page (excluding the data0 block). A value of 0 indicates that only the DATA0 block is present and a value of 8 indicates that 8 subsequent blocks are present for a total of 9 blocks on the flash (including the DATA0 block). Any values from 0 to 255 are supported by the hardware.
23–16 META_SIZE	Indicates the size of the metadata (in bytes) to be stored on a flash page. The BCH design support from 0 to 255 bytes for metadata -- if set to zero, no metadata will be stored. Metadata is stored before the associated data in block 0. If the DATA0_SIZE field is programmed to a zero, then metadata effectively be stored with its own parity. When both the metadata and data0 fields are programmed with non-zero values, the first block will contain both portions of data and will be covered by a single parity block.
15–11 ECC0	Indicates the ECC level for the first block on the flash page. The first block covers metadata plus the associated data from the DATA0_SIZE field. 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed 0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATA0_SIZE	Indicates the size of the data 0 block (in DWORDS / four bytes) to be stored on the flash page. If set to zero, the first block will only contain metadata.

17.6.13 Hardware BCH ECC Flash 2 Layout 1 Register (BCH_FLASH2LAYOUT1*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH2LAYOUT0 register to control the format for the device selecting layout 2 in the LAYOUTSELECT register.

Address: 11_4000h base + D0h offset + (4d × i), where i=0d to 3d



BCH_FLASH2LAYOUT1*n* field descriptions

Field	Description
31–16 PAGE_SIZE	Indicates the total size of the flash page (in bytes). This should be set to the page size including spare area. The page size is programmable to accomodate different flash configurations that may be available in the future.
15–11 ECCN	Indicates the ECC level for the subsequent blocks on the flash page (blocks 1-n). Subsequent blocks only contain data (no metadata). 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed

Table continues on the next page...

BCH_FLASH2LAYOUT1*n* field descriptions (continued)

Field	Description
	0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATAN_SIZE	Indicates the size of the subsequent data blocks (in DWORDS / four bytes) to be stored on the flash page. The size of subsequent data blocks does not have to match the data size for block 0, which is important when metadata is stored separately or for balancing the amount of data stored in each block.

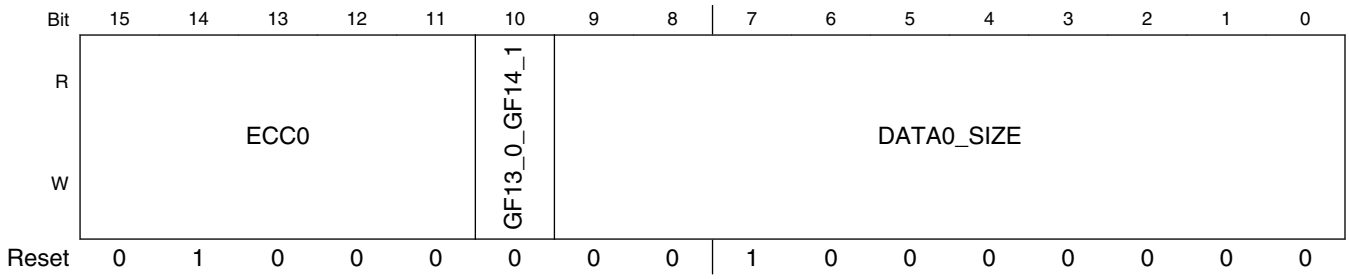
17.6.14 Hardware BCH ECC Flash 3 Layout 0 Register (BCH_FLASH3LAYOUT0*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjunction with the FLASH3LAYOUT1 register to control the format for the devices selecting layout 3 in the LAYOUTSELECT register.

Each pair of layout registers describes one of four supported flash configurations. Software should program the LAYOUTSELECT register for each supported GPMI chip select to select from one of the four layout values. Each pair of registers contains settings that are used by the BCH block while reading/writing the flash page to control data, metadata, and flash page sizes as well as the ECC correction level. The first block written to flash can be programmed to have different ECC, metadata, and data sizes from subsequent data blocks on the device. In addition, the number of blocks stored on a page of flash is not fixed, but instead is determined by the number of bytes consumed by the initial (block 0) and subsequent data blocks. See the BCH programming reference manual for more information on setting up the flash layout registers.

Address: 11_4000h base + E0h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	NBLOCKS								META_SIZE							
W																
Reset	0	0	0	0	0	1	1	1	0	0	0	0	1	0	1	0



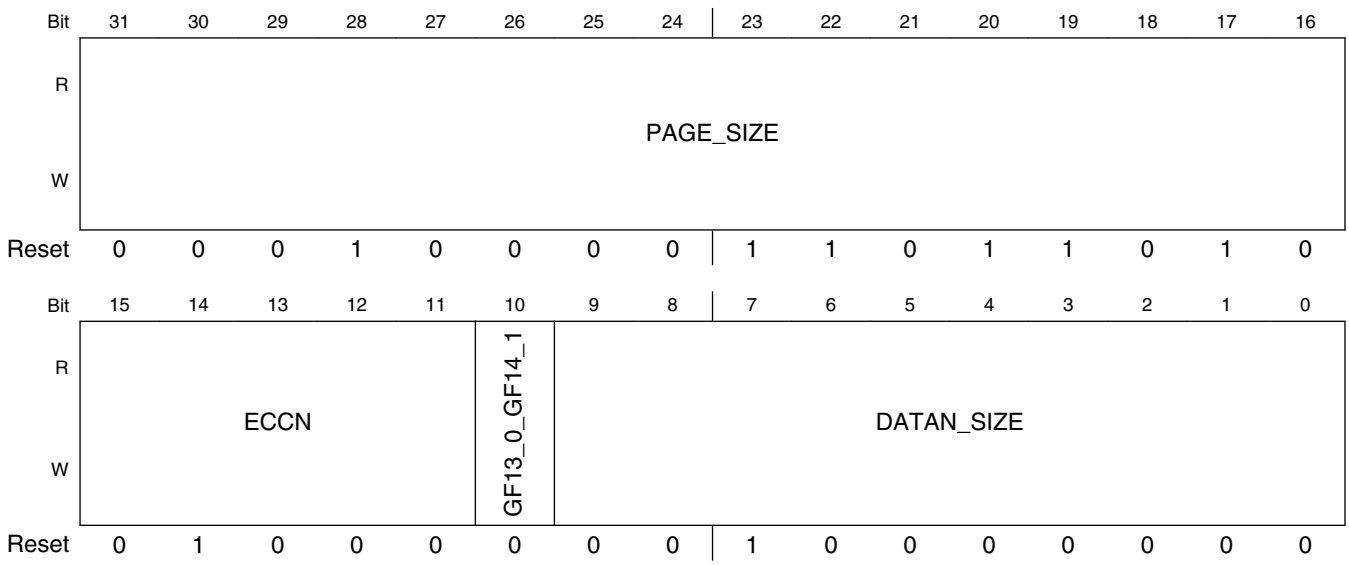
BCH_FLASH3LAYOUT0*n* field descriptions

Field	Description
31–24 NBLOCKS	Number of subsequent blocks on the flash page (excluding the data0 block). A value of 0 indicates that only the DATA0 block is present and a value of 8 indicates that 8 subsequent blocks are present for a total of 9 blocks on the flash (including the DATA0 block). Any values from 0 to 255 are supported by the hardware.
23–16 META_SIZE	Indicates the size of the metadata (in bytes) to be stored on a flash page. The BCH design support from 0 to 255 bytes for metadata -- if set to zero, no metadata will be stored. Metadata is stored before the associated data in block 0. If the DATA0_SIZE field is programmed to a zero, then metadata effectively be stored with its own parity. When both the metadata and data0 fields are programmed with non-zero values, the first block will contain both portions of data and will be covered by a single parity block.
15–11 ECC0	Indicates the ECC level for the first block on the flash page. The first block covers metadata plus the associated data from the DATA0_SIZE field. 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed 0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATA0_SIZE	Indicates the size of the data 0 block (in DWORDS / four bytes) to be stored on the flash page. If set to zero, the first block will only contain metadata.

17.6.15 Hardware BCH ECC Flash 3 Layout 1 Register (BCH_FLASH3LAYOUT1*n*)

The flash format register contains a description of the logical layout of data on the flash device. This register is used in conjucction with the FLASH3LAYOUT0 register to control the format for the device selecting layout 3 in the LAYOUTSELECT register.

Address: 11_4000h base + F0h offset + (4d × i), where i=0d to 3d



BCH_FLASH3LAYOUT1*n* field descriptions

Field	Description
31–16 PAGE_SIZE	Indicates the total size of the flash page (in bytes). This should be set to the page size including spare area. The page size is programmable to accomodate different flash configurations that may be available in the future.
15–11 ECCN	Indicates the ECC level for the subsequent blocks on the flash page (blocks 1-n). Subsequent blocks only contain data (no metadata). 0x0 **NONE** — No ECC to be performed 0x1 **ECC2** — ECC 2 to be performed 0x2 **ECC4** — ECC 4 to be performed 0x3 **ECC6** — ECC 6 to be performed 0x4 **ECC8** — ECC 8 to be performed 0x5 **ECC10** — ECC 10 to be performed 0x6 **ECC12** — ECC 12 to be performed 0x7 **ECC14** — ECC 14 to be performed 0x8 **ECC16** — ECC 16 to be performed 0x9 **ECC18** — ECC 18 to be performed 0xA **ECC20** — ECC 20 to be performed 0xB **ECC22** — ECC 22 to be performed 0xC **ECC24** — ECC 24 to be performed

Table continues on the next page...

BCH_FLASH3LAYOUT1*n* field descriptions (continued)

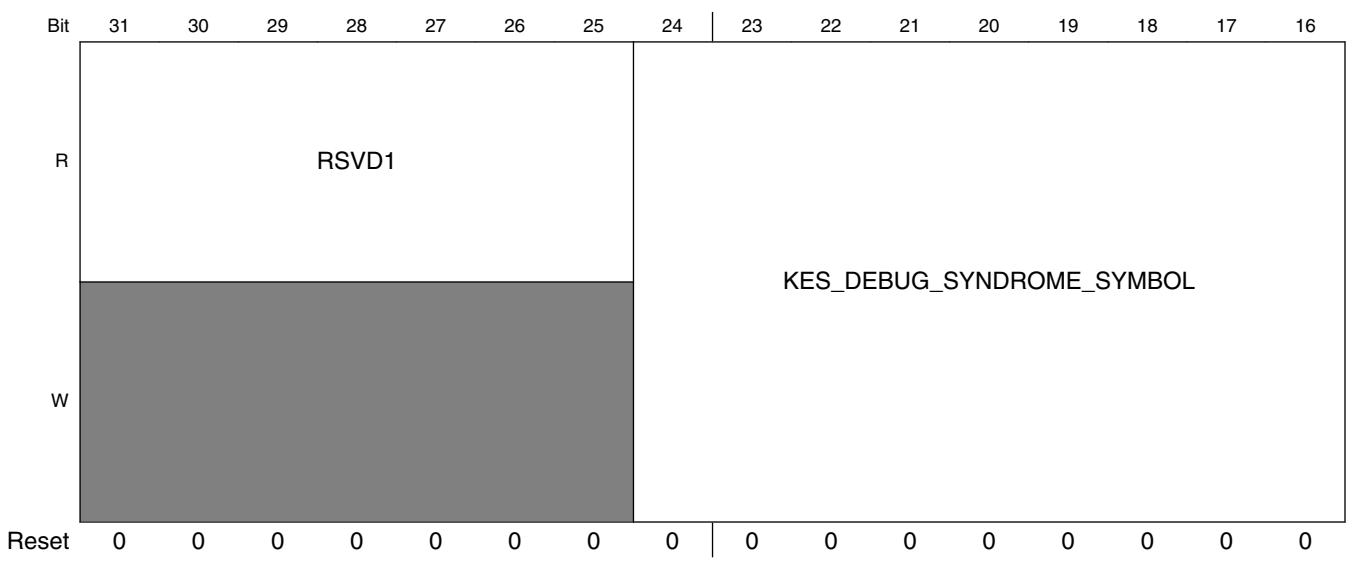
Field	Description
	0xD **ECC26** — ECC 26 to be performed 0xE **ECC28** — ECC 28 to be performed 0xF **ECC30** — ECC 30 to be performed 0x10 **ECC32** — ECC 32 to be performed 0x11 **ECC34** — ECC 34 to be performed 0x12 **ECC36** — ECC 36 to be performed 0x13 **ECC38** — ECC 38 to be performed 0x14 **ECC40** — ECC 40 to be performed
10 GF13_0_GF14_1	Select GF13 or GF14: 0-GF13; 1-GF14
DATAN_SIZE	Indicates the size of the subsequent data blocks (in DWORDS / four bytes) to be stored on the flash page. The size of subsequent data blocks does not have to match the data size for block 0, which is important when metadata is stored separately or for balancing the amount of data stored in each block.

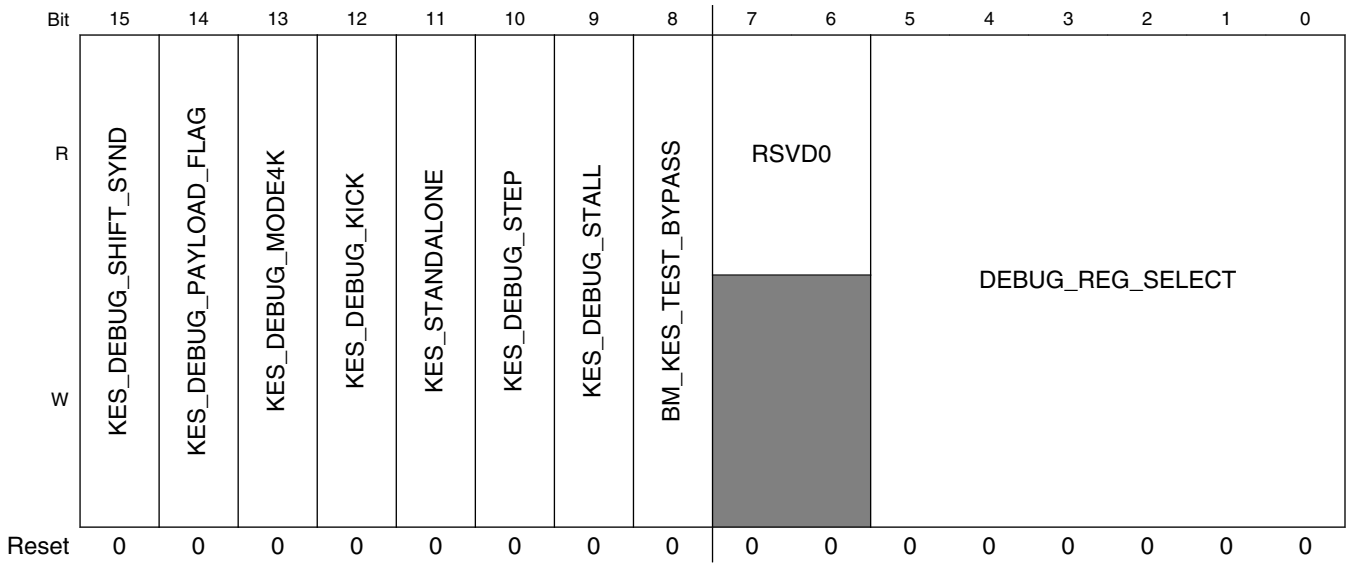
17.6.16 Hardware BCH ECC Debug Register0 (BCH_DEBUG0*n*)

The hardware BCH accelerator internal state machines and signals can be seen in the ECC debug register.

The BCH_DEBUG0 register provides access to various internal state information which might prove useful during hardware debug and validation.

Address: 11_4000h base + 100h offset + (4d × i), where i=0d to 3d





BCH_DEBUG0*n* field descriptions

Field	Description
31–25 RSVD1	Reserved, always set these bits to zero.
24–16 KES_DEBUG_ SYNDROME_ SYMBOL	The 9 bit value in this bit field will be shifted into the syndrome register array at the input of the KES engine whenever BCH_DEBUG0_KES_DEBUG_SHIFT_SYND is toggled. 0x0 **NORMAL** — Bus master address generator for synd_gen writes operates normally. 0x1 **TEST_MODE** — Bus master address generator always addresses last four bytes in Auxilliary block.
15 KES_DEBUG_ SHIFT_SYND	Toggling this bit causes the value in BCH_DEBUG0_KES_SYNDROME_SYMBOL to be shift into the syndrome register array at the input to the KES engine. After shifting in 16 symbols, one can kick off both KES and CF cycles by toggling BCH_DEBUG0_KES_DEBUG_KICK. Be sure to set KES_BCH_DEBUG0_KES_STANDALONE mode to 1 before kicking.
14 KES_DEBUG_ PAYLOAD_FLAG	When running the stand alone debug mode on the error calculator, the state of this bit is presented to the KES engine as the input payload flag. 0x1 **DATA** — Payload is set for 512 byte data block. 0x1 **AUX** — Payload is set for 65 or 19 byte auxilliary block.
13 KES_DEBUG_ MODE4K	When running the stand alone debug mode on the error calculator, the state of this bit is presented to the KES engine as the input mode (4K or 2K pages). 0x1 **4k** — Mode is set for 4K NAND pages. 0x1 **2k** — Mode is set for 2K NAND pages.
12 KES_DEBUG_ KICK	Toggling causes KES engine FSM to start as if kick by the Bus Master. This allows stand alone testing of the KES and Chien Search engines. Be sure to set KES_BCH_DEBUG0_KES_STANDALONE mode to 1 before kicking.
11 KES_ STANDALONE	Set to one to cause the KES engine to suppress toggling the KES_BM_DONE signal to the bus master and to suppress toggling the CF_BM_DONE signal by the CF engine. 0x0 **NORMAL** — Bus master address generator for synd_gen writes operates normally. 0x1 **TEST_MODE** — Bus master address generator always addresses last four bytes in Auxilliary block.

Table continues on the next page...

BCH_DEBUG0*n* field descriptions (continued)

Field	Description
10 KES_DEBUG_STEP	Toggling this bit causes the KES FSM to skip passed the stall state if it is in DEBUG_STALL mode and it has completed processing a block.
9 KES_DEBUG_STALL	Set to one to cause KES FSM to stall after notifying Chien search engine to start processing its block but before notifying the bus master that the KES computation is complete. This allows a diagnostic to stall the FSM after each blocks key equations are solved. This also has the effect of stalling the CSFE search engine so it's state can be examined after it finishes processing the KES stalled block. 0x0 **NORMAL** — KES FSM proceeds to next block supplied by bus master. 0x1 **WAIT** — KES FSM waits after current equations are solved and the search engine is started.
8 BM_KES_TEST_BYPASS	1 = Point all synd_gen writes to dummy area at the end of the AUXILLIARY block so that diagnostics can preload all payload, parity bytes and computed syndrome bytes for test the KES engine. 0x0 **NORMAL** — Bus master address generator for synd_gen writes operates normally. 0x1 **TEST_MODE** — Bus master address generator always addresses last four bytes in Auxilliary block.
7–6 RSVD0	Reserved, always set these bits to zero.
DEBUG_REG_SELECT	The value loaded in this bit field is used to select the internal register state view of KES engine or the Chien search engine.

17.6.17 KES Debug Read Register (BCH_DBGKESREAD*n*)

The hardware BCH ECC accelerator key equation solver internal state machines and signals can be seen in the ECC debug registers.

Address: 11_4000h base + 110h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VALUES																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_DBGKESREAD*n* field descriptions

Field	Description
VALUES	This register will return the ROM BIST CRC value after a BIST test.

17.6.18 Chien Search Debug Read Register (BCH_DBGCSFEREAD*n*)

The hardware BCH ECC accelerator Chien Search internal state machines and signals can be seen in the ECC debug registers.

Address: 11_4000h base + 120h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VALUES																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_DBGCSFEREAD*n* field descriptions

Field	Description
VALUES	Reserved

17.6.19 Syndrome Generator Debug Read Register (BCH_DBGSYNDGENREAD*n*)

The hardware BCH ECC accelerator syndrome generator internal state machines and signals can be seen in the ECC debug registers.

Address: 11_4000h base + 130h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VALUES																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_DBGSYNDGENREAD*n* field descriptions

Field	Description
VALUES	Reserved

17.6.20 Bus Master and ECC Controller Debug Read Register (BCH_DBGAHBMREAD*n*)

The hardware BCH ECC accelerator bus master and ecc controller internal state machines and signals can be seen in the ECC debug registers.

Address: 11_4000h base + 140h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VALUES																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_DBGAHBMREAD*n* field descriptions

Field	Description
VALUES	Reserved

17.6.21 Block Name Register (BCH_BLOCKNAME*n*)

Read only view of the block name string BCH.

Fixed pattern read only value for test purposes. Can be read as an ASCII string with the zero termination coming from the first byte of the BLOCKVERSION register.

Address: 11_4000h base + 150h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	NAME																															
W																																
Reset	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	1	0	1	0	0	0	0	1	0

BCH_BLOCKNAME*n* field descriptions

Field	Description
NAME	Should be the ASCII characters BCH (0x20, H, C, B).

17.6.22 BCH Version Register (BCH_VERSION*n*)

This register always returns a known read value for debug purposes and indicates the version of the block and RTL version in use.

Address: 11_4000h base + 160h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	MAJOR								MINOR								STEP															
W																																
Reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BCH_VERSION*n* field descriptions

Field	Description
31–24 MAJOR	Fixed read-only value reflecting the MAJOR field of the RTL version.
23–16 MINOR	Fixed read-only value reflecting the MINOR field of the RTL version.
STEP	Fixed read-only value reflecting the stepping of the RTL version.

Chapter 18
Clock Controller Module (CCM)

18.1 Overview

The Clock Control Module (CCM) generates and controls clocks to the various modules in the design and manages low power modes. This module uses the available clock sources to generate the clock roots.

The Clock Controller Module controls the following functions:

- Uses the available clock sources to generate clock roots to various parts of the chip:
  - PLL1 also referenced as ARM PLL
  - PLL2 also referenced as System PLL
  - PLL3 also referenced as USB1 PLL
  - PLL4 also referenced as Audio PLL
  - PLL5 also referenced as Video PLL
  - PLL6 also referenced as ENET PLL
  - PLL7 also referenced as USB2 PLL (This PLL is only used by the USB UTM interface through a direct connection.)
  - PLL8 also referenced as MLB PLL
- Uses programmable bits to control frequencies of the clock roots.
- Controls the low power mechanism.
- Provides control signals to LPCG for gating clocks.
- Provides handshake with SRC for reset performance.
- Provides handshake with GPC for support of DVFS and power gating operations.

18.1.1 Features

The CCM includes these distinctive features:

- Provides root clock to SoC modules based on several source clocks.
- ARM core root clock is generated from a dedicated source clock.

- Includes separate dividers to control generation of core and bus root clocks (AXI, AHB, IPG).
- Includes separate dividers and clock source selectors for each serial root clock.
- Optional external clocks to bypass PLL clocks.
- Selects clock signals to output on CCM_CLKO1 and CCM_CLKO2 onto the pads for observability.
- Controllable registers are accessible via IP bus.
- Manages the Low Power Modes, namely RUN, WAIT and STOP. The gating of the peripheral clocks is programmable in RUN and WAIT modes.
- Manages frequency scaling procedure for ARM core clock by shifting between PLL sources, without loss of clocks.
- Manages frequency scaling procedure for peripheral root clock by programmable divider. The division is done on the fly without loss of clocks.
- Interface for the following IPs:
    - PLL - Interfaces for each PLL
    - LPCG - Low Power Clock Gating unit
    - SRC - System Reset Controller
    - GPC - General Power Controller

18.1.2 CCM Block Diagram

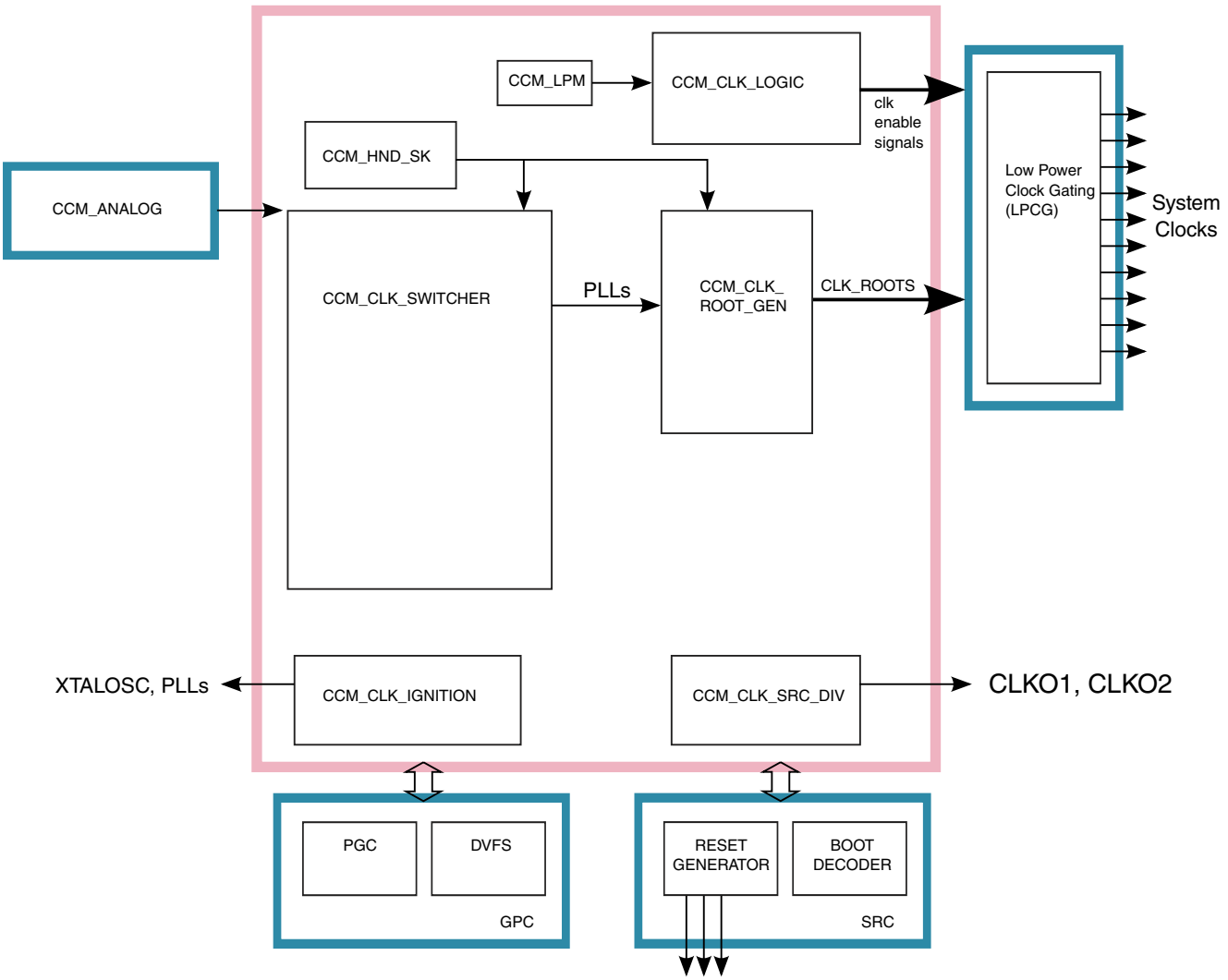


Figure 18-1. Block Diagram

CCM Contains the following sub-blocks:

Table 18-1. CCM Sub-blocks

Sub-block	Description
CCM_CLK_IGNITION	Manages the ignition process. This module starts its functionality once CCM comes out of reset. It manages the process that begins with starting the OSC, PLLs and finishes with creation of stable output root clocks after reset.
CCM_CLK_SWITCHER	Receives the clock outputs of the PLLs, together with the bypass clocks for the PLLs, and generates three switcher clock outputs (pll1_sw_clk, pll3_sw_clk) for the CCM_CLK_ROOT_GEN sub-module.
CCM_CLK_ROOT_GEN	Receives the main clocks (PLLs / PFDs) and generates the output root clocks.
CCM_CLK_LOGIC	Generates the clock enables. It generates the clock enable signals based on info from CCM_LPM and CCM_IP. The clock enables are used in LPCG to turn off and on the split clocks.

Table continues on the next page...

Table 18-1. CCM Sub-blocks (continued)

Sub-block	Description
CCM_LPM	Manages the low power modes of the IC.
CCM_CLK_SRC_DIV	Muxes different clocks to two output clocks, CCM_CLKO1 and CCM_CLKO2, for observability. These output clocks are connected to the pads and can provide support for testing.
CCM_HND_SK	Manages the handshake when changing root clock dividers that require handshake, and manages the frequency change in case of dvfs scenario.

18.2 External Signals

The following table describes the external signals of CCM:

Table 18-2. CCM External Signals

Signal	Description	Pad	Mode	Direction
CCM_CLKO1	Observability clock 1 output	CSI0_MCLK	ALT3	O
		GPIO_0	ALT0	
		GPIO_19	ALT3	
		GPIO_5	ALT3	
CCM_CLKO2	Observability clock 2 output	GPIO_3	ALT4	O
		NANDF_CS2	ALT4	
CCM_PMIC_STBY_REQ	Goes to PMIC_STBY_REQ pin, which notifies external power management IC to move from functional voltage to standby voltage.	PMIC_STBY_REQ	No Muxing (ALT0)	O
CCM_REF_EN_B	Enable external reference clock (CKIH)	GPIO_9	ALT3	O
CCM_PLL3_BYP	PLL3 bypass clock	RGMII_TD1	ALT7	I

18.3 CCM Clock Tree

The figure found here shows the clock tree configuration and clock roots for CCM.

For detailed sub-block information, please see:

- Clock Switcher
- Clock Root Generator
- Low Power Clock Gating module (LPCG)
- System Clocks

NOTE

The default frequency values (in MHz) for the PLLs and PFDs is the maximum allowed frequency. The PLL and PFD control registers should not be programmed to exceed these values.

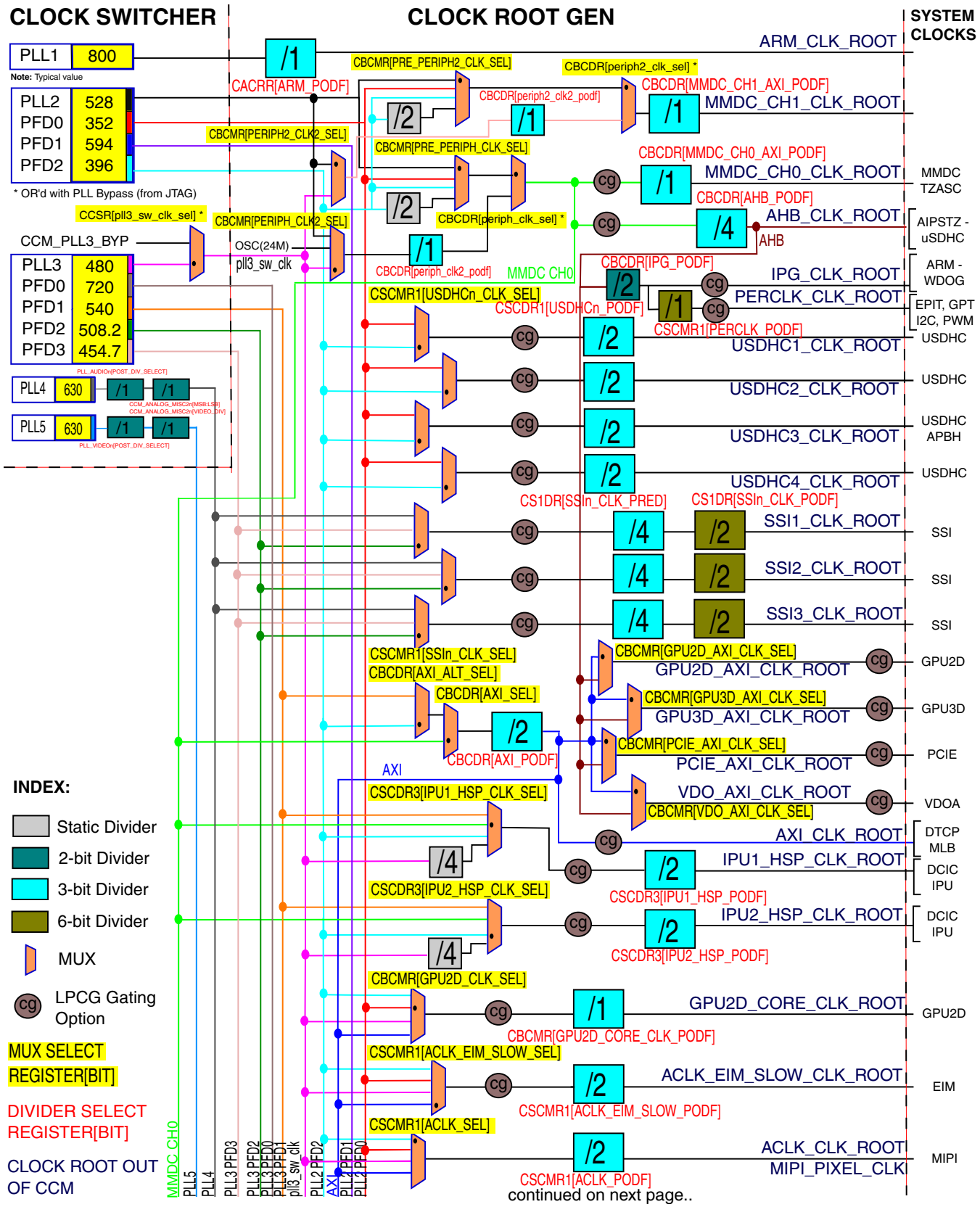


Figure 18-2. Clock Tree - Part 1

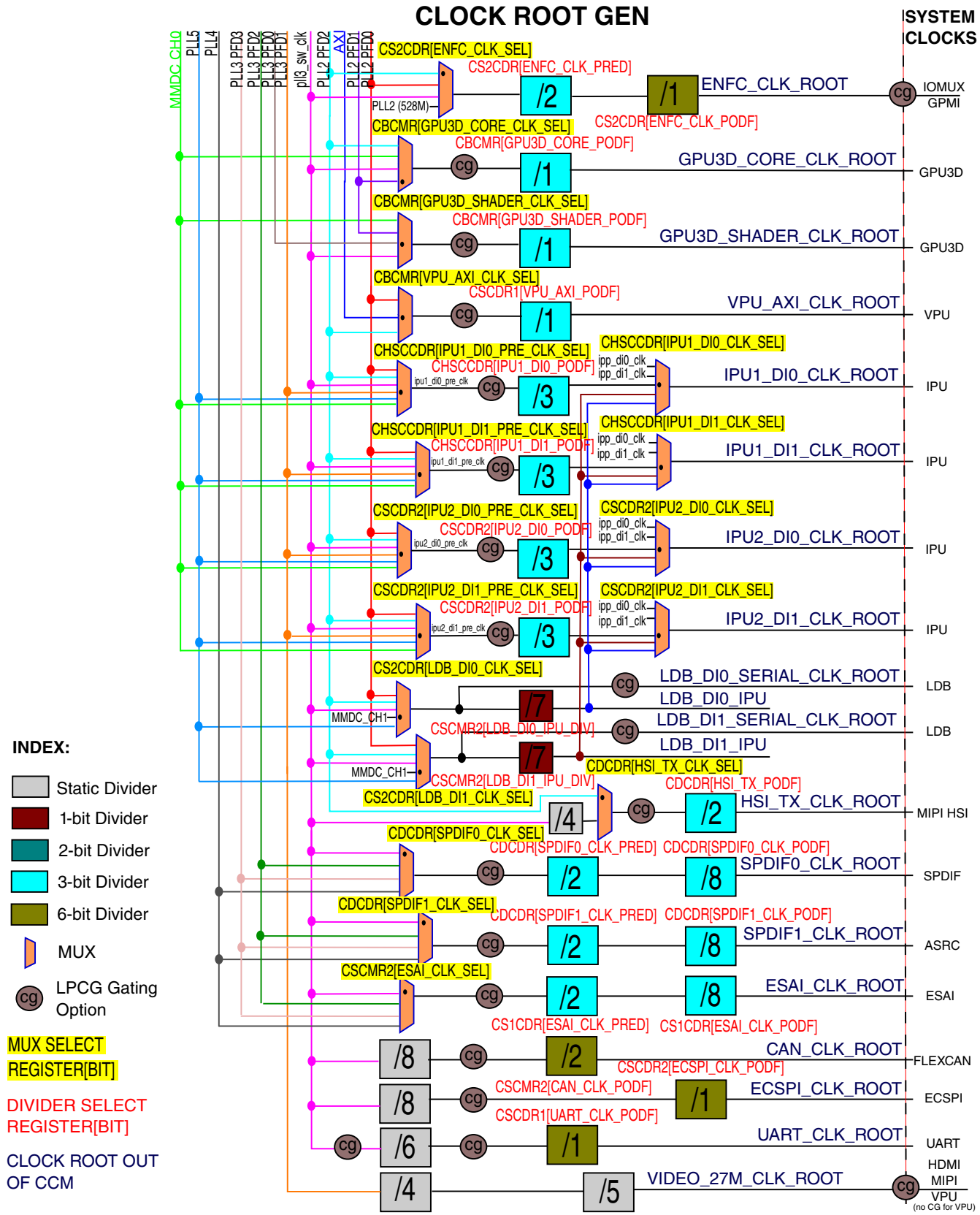


Figure 18-3. Clock Tree - Part 2

18.4 System Clocks

The table found here shows the CCM output clocks' system-level connectivity.

The gating option in the table can either be CGR bit or clock enable from the block itself. Applicable override bits are also shown.

Table 18-3. System Clocks, Gating, and Override

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
AIPSTZ*n*	hclk	ahb_clk_root	CCGR0[CG0] (aips_tz1_clk_enable) CCGR0[CG1] (aips_tz2_clk_enable)	
APBH	hclk	usdhc3_clk_root	CCGR0[CG2] (apbhdma_hclk_enable)	
	sec_mst_hclk	usdhc3_clk_root	CCGR0[CG2] (apbhdma_hclk_enable)	
ARM	clk_ahb	ahb_clk_root	CCGR0[CG11] (arm_dbg_clk_enable)	
	clk_apb_dbg	ahb_clk_root	CCGR0[CG11] (arm_dbg_clk_enable)	
	clk_atb	ahb_clk_root	CCGR0[CG11] (arm_dbg_clk_enable)	
	ipg_clk	ipg_clk_root	CCGR0[CG11] (arm_dbg_clk_enable)	
	trace_clk_in	ahb_clk_root	CCGR0[CG11] (arm_dbg_clk_enable)	
ASRC	asrck_clock_d	spdif1_clk_root		
	ipg_clk	ahb_clk_root	CCGR0[CG3] (asrc_clk_enable)	
	mem_clk	ahb_clk_root	CCGR0[CG3] (asrc_clk_enable)	
AUDMUX	ipg_clk_s	ipg_clk_root		
CAAM	sec_mem_clk	ahb_clk_root	CCGR0[CG4] (caam_secure_mem_clk_enable)	
	aclk	ahb_clk_root	CCGR0[CG5] (caam_wrapper_aclk_enable)	
	ckil	ckil_sync_clk_root		
	ipg_clk	ipg_clk_root	CCGR0[CG6] (caam_wrapper_ipg_enable)	
	ipg_clk_s	ipg_clk_root		
	exsc_aclk_exsc	ahb_clk_root	CCGR0[CG5] (caam_wrapper_aclk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
CCM	analog_apb_clk	ipg_clk_root		
	ipg_clk	ipg_clk_root		
	ipg_clk_s	ipg_clk_root		
CSU	ap_ckil_clk	ckil_sync_clk_root		
	ipg_clk_s	ipg_clk_root		
DBGMON	clk	ipg_clk_root		
DCIC*n*	hsp_clk	ipu1_hsp_clk_root	CCGR0[CG12] (dcic1_clk_enable) CCGR0[CG13] (dcic2_clk_enable)	
	ipg_clk_s	ipu1_hsp_clk_root	CCGR0[CG12] (dcic1_clk_enable) CCGR0[CG13] (dcic2_clk_enable)	
DTCP	clk	axi_clk_root	CCGR0[CG14] (dtcp_clk_enable)	
	ram_CLK	axi_clk_root	CCGR0[CG14] (dtcp_clk_enable)	
	rom_CLK	axi_clk_root	CCGR0[CG14] (dtcp_clk_enable)	
	sec_mst_hclk	axi_clk_root	CCGR0[CG14] (dtcp_clk_enable)	
ECSPI*n*	ipg_clk	ipg_clk_root	CCGR1[CG0] (ecspi1_clk_enable) CCGR1[CG1] (ecspi2_clk_enable) CCGR1[CG2] (ecspi3_clk_enable) CCGR1[CG3] (ecspi4_clk_enable) CCGR1[CG4] (ecspi5_clk_enable)	
	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_per	ecspi_clk_root	CCGR1[CG0] (ecspi1_clk_enable) CCGR1[CG1] (ecspi2_clk_enable) CCGR1[CG2] (ecspi3_clk_enable) CCGR1[CG3] (ecspi4_clk_enable) CCGR1[CG4] (ecspi5_clk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
	ipg_clk_s	ipg_clk_root		
EIM	aclk	aclk_eim_slow_clk_root	CCGR6[CG5] (eim_slow_clk_enable)	
	aclk_slow	aclk_eim_slow_clk_root	CCGR6[CG5] (eim_slow_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	aclk_exsc	aclk_eim_slow_clk_root	CCGR6[CG5] (eim_slow_clk_enable)	
ENET	ipg_clk	ahb_clk_root	CCGR1[CG5] (enet_clk_enable)	
	ipg_clk_mac0	ahb_clk_root	CCGR1[CG5] (enet_clk_enable)	
	ipg_clk_mac0_s	ipg_clk_root	CCGR1[CG5] (enet_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	ipg_clk_time	ipg_clk_root	CCGR1[CG5] (enet_clk_enable)	
	mac0_rxmem_clk	ahb_clk_root	CCGR1[CG5] (enet_clk_enable)	
	mac0_txmem_clk	ahb_clk_root	CCGR1[CG5] (enet_clk_enable)	
EPIT*n*	ipg_clk	ipg_clk_root	CCGR1[CG6] (epit1_clk_enable) CCGR1[CG7] (epit2_clk_enable)	CMEOR[mod_en_ov_epit]
	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_highfreq	perclk_clk_root	CCGR1[CG6] (epit1_clk_enable) CCGR1[CG7] (epit2_clk_enable)	
	ipg_clk_s	ipg_clk_root		
ESAI	extal_clk	esai_clk_root	CCGR1[CG8] (esai_clk_enable)	
	ipg_clk_esai	ahb_clk_root	CCGR1[CG8] (esai_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	mem_clk	ahb_clk_root	CCGR1[CG8] (esai_clk_enable)	
FLEXCAN*n*	ipg_clk	ipg_clk_root	CCGR0[CG7] (can1_clk_enable) CCGR0[CG9] (can2_clk_enable)	
	ipg_clk_chi	ipg_clk_root	CCGR0[CG7] (can1_clk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
			CCGR0[CG9] (can2_clk_enable)	
	ipg_clk_pe	can_clk_root	CCGR0[CG8] (can1_serial_clk_enable) CCGR0[CG10] (can2_serial_clk_enable)	CMEOR[mod_en_ov_can1_cpi] CMEOR[mod_en_ov_can2_cpi]
	ipg_clk_pe_nogate	can_clk_root	CCGR0[CG8] (can1_serial_clk_enable) CCGR0[CG10] (can2_serial_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	ram_CLK	ipg_clk_root	CCGR0[CG7] (can1_clk_enable) CCGR0[CG9] (can2_clk_enable)	
GPC	ipg_clk	ipg_clk_root		
	ipg_clk_s	ipg_clk_root		
	pgc_clk	ipg_clk_root		
	sys_clk	ipg_clk_root		
GPIO*n*	ipg_clk_s	ipg_clk_root		
GPMI	u_bch_input_apb_clk	usdhc3_clk_root	CCGR4[CG12] (rawnand_u_bch_input_apb_clk_enable)	
	u_gpmi_bch_input_bch_clk	usdhc4_clk_root	CCGR4[CG13] (rawnand_u_gpmi_bch_input_bch_clk_enable)	
	u_gpmi_bch_input_gpmi_io_clk	enfc_clk_root	CCGR4[CG14] (rawnand_u_gpmi_bch_input_gpmi_io_clk_enable)	
	u_gpmi_input_apb_clk	usdhc3_clk_root	CCGR4[CG15] (rawnand_u_gpmi_input_apb_clk_enable)	
GPT	ipg_clk	ipg_clk_root	CCGR1[CG10] (gpt_clk_enable)	CMEOR[mod_en_ov_gpt]
	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_highfreq	perclk_clk_root	CCGR1[CG11] (gpt_serial_clk_enable)	
	ipg_clk_s	ipg_clk_root		
GPU2D	ACLK0	gpu2d_axi_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	CMEOR[mod_en_ov_gpu2d]
	ACLK1	gpu2d_axi_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	
	clk2x	gpu2d_core_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	CMEOR[mod_en_ov_gpu2d]

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
	HCLK	ahb_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	CMEOR[mod_en_ov_gpu2d]
	V2D_ACLK	gpu2d_axi_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	
	V2D_clk2x	gpu2d_core_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	
	V2D_HCLK	ahb_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	
	sec_mst_hclk	ahb_clk_root	CCGR1[CG12] (gpu2d_clk_enable)	CMEOR[mod_en_ov_gpu2d]
GPU3D	ACLK0	gpu3d_axi_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	CMEOR[mod_en_ov_gpu3d]
	ACLK1	gpu3d_axi_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	
	clk1x	gpu3d_core_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	CMEOR[mod_en_ov_gpu3d]
	clkShader	gpu3d_shader_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	
	HCLK	ahb_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	CMEOR[mod_en_ov_gpu3d]
	sec_mst_hclk	ahb_clk_root	CCGR1[CG13] (gpu3d_clk_enable)	CMEOR[mod_en_ov_gpu3d]
HDMI	sec_mst_hclk	ahb_clk_root	CCGR2[CG0] (hdmi_tx_enable)	
	iahbclk	ahb_clk_root	CCGR2[CG0] (hdmi_tx_enable)	
	icecclk	ckil_sync_clk_root		
	ihclk	ahb_clk_root	CCGR2[CG0] (hdmi_tx_enable)	
	isfrclk	video_27m_clk_root	CCGR2[CG2] (hdmi_tx_isfrclk_enable)	
	vl_sms_proc_sms_hdmi_sram_27_clk_sms	ipg_clk_root		
I2C*n*	ipg_clk_patref	perclk_clk_root	CCGR2[CG3] (i2c1_serial_clk_enable) CCGR2[CG4] (i2c2_serial_clk_enable) CCGR2[CG5] (i2c3_serial_clk_enable)	
	ipg_clk_s	ipg_clk_root		
IOMUXC	ipt_clk_io	enfc_clk_root	CCGR2[CG7] (iomux_ipt_clk_io_enable)	
	ipg_clk_s	ipg_clk_root		
IPU*n*	hsp_clk	ipu1_hsp_clk_root	CCGR3[CG0] (ipu1_clk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
			CCGR3[CG3] (ipu2_clk_enable)	
	ipp_di_0_ext_clk ipu1_di0_pre_clk ipu2_di0_pre_clk	ipu1_di0_clk_root ipu2_di0_clk_root	CCGR3[CG1] (ipu1_di0_clk_enable) CCGR3[CG4] (ipu2_di0_clk_enable)	
	ipp_di_1_ext_clk ipu1_di1_pre_clk ipu2_di1_pre_clk	ipu1_di1_clk_root ipu2_di1_clk_root	CCGR3[CG2] (ipu1_di1_clk_enable) CCGR3[CG5] (ipu2_di1_clk_enable)	
	ipu_master_hclk	ahb_clk_root	CCGR3[CG0] (ipu1_clk_enable) CCGR3[CG3] (ipu2_clk_enable)	
	vl_sms_proc_sms_ipu_sram_266_clk_sms	ipu1_hsp_clk_root	CCGR3[CG0] (ipu1_clk_enable) CCGR3[CG3] (ipu2_clk_enable)	
	sec_mst_hclk	ipu1_hsp_clk_root	CCGR3[CG0] (ipu1_clk_enable) CCGR3[CG3] (ipu2_clk_enable)	
KPP	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_s	ipg_clk_root		
LDB	ch_0_serial_clk	ldb_di0_serial_clk_root	CCGR3[CG6] (ldb_di0_clk_enable)	
	ch_1_serial_clk	ldb_di1_serial_clk_root	CCGR3[CG7] (ldb_di1_clk_enable)	
	di_0_clk_nc	ldb_di0_serial_clk_root	CCGR3[CG6] (ldb_di0_clk_enable)	
	di_1_clk_nc	ldb_di1_serial_clk_root	CCGR3[CG7] (ldb_di1_clk_enable)	
MIPI	ac_clk_125m	ahb_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	mipi_pixel_clk	aclk_clk_root		
	cfg_clk	video_27m_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	ips_clk	ipg_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	ips_clk_s	ipg_clk_root		
	pll_refclk	video_27m_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	vl_sms_proc_sms_mipi_rf_125_clk_sms	ipg_clk_root		

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
MIPI_HSI	h_clk	ahb_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	tx_ref_clk	hsi_tx_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
	sec_mst_hclk	ahb_clk_root	CCGR3[CG8] (mipi_core_cfg_clk_enable)	
MLB	hclk	ahb_clk_root	CCGR3[CG9] (mlb_clk_enable)	
	ipg_clk_s	ipg_clk_root	CCGR3[CG9] (mlb_clk_enable)	
	sys_clk	axi_clk_root	CCGR3[CG9] (mlb_clk_enable)	
	mem_ct_CLK	axi_clk_root	CCGR3[CG9] (mlb_clk_enable)	
	mem_db_CLK	axi_clk_root	CCGR3[CG9] (mlb_clk_enable)	
MMDC	aclk_fast_core_p0	mmdc_ch0_clk_root	CCGR3[CG10] (mmdc_core_aclk_fast_core_p0_enable)	
	aclk_fast_core_p1	GND		
	ipg_clk_p0	ipg_clk_root	CCGR3[CG12] (mmdc_core_ipg_clk_p0_enable	
	ipg_clk_p1	ipg_clk_root		
	aclk_fast_phy_p0	mmdc_ch0_clk_root	CCGR3[CG10] (mmdc_core_aclk_fast_core_p0_enable)	
	aclk_fast_phy_p1	GND		
OCOTP	ipg_clk	ipg_clk_root	CCGR2[CG6] (iim_clk_enable)	
	ipg_clk_s	ipg_clk_root		
OCRAM	clk	ahb_clk_root	CCGR3[CG14] (ocram_clk_enable)	
	aclk_exsc	ahb_clk_root	CCGR3[CG14] (ocram_clk_enable)	
	mem_clk	ahb_clk_root	CCGR3[CG14] (ocram_clk_enable)	
PCIE	rst_aux_clk	ipg_clk_root	CCGR4[CG0] (pcie_root_enable)	
	rst_dbi_axi_clk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	rst_mstr_axi_clk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	rst_slv_axi_clk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
	mstr_aclk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	slv_aclk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	aclk_exsc	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	ram_mstr_axi_clk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
	ram_slv_axi_clk	pcie_axi_clk_root	CCGR4[CG0] (pcie_root_enable)	
PWM*n*	ipg_clk	ipg_clk_root	CCGR4[CG8] (pwm1_clk_enable) CCGR4[CG9] (pwm2_clk_enable) CCGR4[CG10] (pwm3_clk_enable) CCGR4[CG11] (pwm4_clk_enable)	
	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_highfreq	perclk_clk_root	CCGR4[CG8] (pwm1_clk_enable) CCGR4[CG9] (pwm2_clk_enable) CCGR4[CG10] (pwm3_clk_enable) CCGR4[CG11] (pwm4_clk_enable)	
	ipg_clk_s	ipg_clk_root		
ROMCP	hclk	ahb_clk_root	CCGR5[CG0] (rom_clk_enable)	
	hclk_reg	ipg_clk_root	CCGR5[CG0] (rom_clk_enable)	
	sec_mst_hclk	ahb_clk_root	CCGR5[CG0] (rom_clk_enable)	
SATA	hclk	ahb_clk_root	CCGR5[CG2] (sata_clk_enable)	
	clk_app	ahb_clk_root	CCGR5[CG2] (sata_clk_enable)	
	hclk_sata_mem	ahb_clk_root	CCGR5[CG2] (sata_clk_enable)	
	sec_mst_hclk	ahb_clk_root	CCGR5[CG2] (sata_clk_enable)	
SDMA	clk_sms	ipg_clk_root		

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
	ips_hostctrl_clk	ipg_clk_root	CCGR5[CG3] (sdma_clk_enable)	
	sdma_ap_ahb_clk	ahb_clk_root	CCGR5[CG3] (sdma_clk_enable)	
	sdma_core_clk	ipg_clk_root	CCGR5[CG3] (sdma_clk_enable)	
	tck	ioring_ipp_ind_jtag_tck		
	clk	ahb_clk_root	CCGR5[CG3] (sdma_clk_enable)	
SNVS	hp_ipg_clk	ipg_clk_root		
	hp_ipg_clk_s	ipg_clk_root		
	ipg_hp_rtc_clk	ckil_sync_clk_root		
	lp_ipg_clk	ipg_clk_root		
	lp_ipg_clk_s	ipg_clk_root		
SPBA	ipg_clk	ipg_clk_root	CCGR5[CG6] (spba_clk_enable)	
	ipg_clk_s	ipg_clk_root	CCGR5[CG6] (spba_clk_enable)	
SPDIF	gclkw_t0	ipg_clk_root	CCGR5[CG7] (spdif_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	tx_clk	spdif0_clk_root	CCGR5[CG7] (spdif_clk_enable)	
SRC	ipg_clk	ipg_clk_root		
	ipg_clk_s	ipg_clk_root		
SSI*n*	ccm_ssi_clk	ssi1_clk_root	CCGR5[CG9] (ssi1_clk_enable) CCGR5[CG10] (ssi2_clk_enable) CCGR5[CG11] (ssi3_clk_enable)	
	ipg_clk	ipg_clk_root	CCGR5[CG9] (ssi1_clk_enable) CCGR5[CG10] (ssi2_clk_enable) CCGR5[CG11] (ssi3_clk_enable)	
	ipg_clk_s	ipg_clk_root		
TZASC*n*	aclk	mmdc_ch0_clk_root	CCGR2[CG11] (ipsync_ip2apb_tzasc1_ipg_master_clk_enable)	
	aclk	mmdc_ch0_clk_root	CCGR2[CG12] (ipsync_ip2apb_tzasc2_ipg_master_clk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
UART*n*	ipg_clk	ipg_clk_root	CCGR5[CG12] (uart_clk_enable)	
	ipg_clk_s	ipg_clk_root		
	ipg_perclk	uart_clk_root	CCGR5[CG13] (uart_serial_clk_enable)	
USB	ipg_ahb_clk	ahb_clk_root	CCGR6[CG0] (usboh3_clk_enable)	
	ipg_clk_32khz	ckil_sync_clk_root		
	ipg_clk_s	ipg_clk_root		
	ipg_clk_s_pl301	ipg_clk_root	CCGR6[CG0] (usboh3_clk_enable)	
	test_clk_240m	ipg_clk_root	CCGR6[CG0] (usboh3_clk_enable)	
	test_clk_480m	ipg_clk_root	CCGR6[CG0] (usboh3_clk_enable)	
	test_clk_60m	ipg_clk_root	CCGR6[CG0] (usboh3_clk_enable)	
USDHC*n*	hclk	ahb_clk_root	CCGR6[CG0] (usdhc1_clk_enable) CCGR6[CG1] (usdhc2_clk_enable) CCGR6[CG2] (usdhc3_clk_enable) CCGR6[CG3] (usdhc4_clk_enable)	CMEOR[mod_en_ov_usdhc]
	ipg_clk	ipg_clk_root	CCGR6[CG0] (usdhc1_clk_enable) CCGR6[CG1] (usdhc2_clk_enable) CCGR6[CG2] (usdhc3_clk_enable) CCGR6[CG3] (usdhc4_clk_enable)	CMEOR[mod_en_ov_usdhc]
	ipg_clk_perclk	usdhc1_clk_root	CCGR6[CG0] (usdhc1_clk_enable) CCGR6[CG1] (usdhc2_clk_enable) CCGR6[CG2] (usdhc3_clk_enable) CCGR6[CG3] (usdhc4_clk_enable)	CMEOR[mod_en_ov_usdhc]
	ipg_clk_s	ipg_clk_root		
VDOA	ipg_clk_s	vdo_axi_clk_root	CCGR6[CG6] (vdoaxi_clk_enable)	

Table continues on the next page...

Table 18-3. System Clocks, Gating, and Override (continued)

Block Instance	Clock	Clock Root	Gating CCM_CCGR bits	Override CCM_CMEOR bits
	vdoa_clk	vdo_axi_clk_root	CCGR6[CG6] (vdoaxi_clk_enable)	CMEOR[mod_en_ov_vdoa]
	vdoa_clk	vdo_axi_clk_root	CCGR6[CG6] (vdoaxi_clk_enable)	CMEOR[mod_en_ov_vdoa]
VPU	aclk	vpu_axi_clk_root	CCGR6[CG7] (vpu_clk_enable)	CMEOR[mod_en_ov_vpu]
	cclk	vpu_axi_clk_root	CCGR6[CG7] (vpu_clk_enable)	CMEOR[mod_en_ov_vpu]
	ipg_clk_s	ipg_clk_root		
	rclk	video_27m_clk_root		CMEOR[mod_en_ov_vpu]
WDOG*n*	ipg_clk_32k	ckil_sync_clk_root		
	ipg_clk_s	ipg_clk_root		

Table 18-4. System Clock Frequency Values

Clock Root	Default Frequency (MHz)	Maximum Frequency (MHz)
ARM_CLK_ROOT	792	
MMDC_CH0_CLK_ROOT	528	528
MMDC_CH1_CLK_ROOT	528	528
AHB_CLK_ROOT	132	133
IPG_CLK_ROOT	66	66.5
PERCLK_CLK_ROOT	66	80
USDHCn_CLK_ROOT	198	208
SSIn_CLK_ROOT	63.5	66.5
GPU2D_AXI_CLK_ROOT	270	540
GPU3D_AXI_CLK_ROOT	270	540
PCIE_AXI_CLK_ROOT	270	540
VDO_AXI_CLK_ROOT	270	540
AXI_CLK_ROOT	270	540
IPU1_HSP_CLK_ROOT	264	264
IPU2_HSP_CLK_ROOT	264	264
GPU2D_CORE_CLK_ROOT	352	532
ACLK_EIM_SLOW_CLK_ROOT	396	540
ACLK_CLK_ROOT	198	540
ENFC_CLK_ROOT	198	200
GPU3D_CORE_CLK_ROOT	270	540
GPU3D_SHADER_CLK_ROOT	270	660
VPU_AXI_CLK_ROOT	352	540
IPU1_DI0_CLK_ROOT	180	266
IPU1_DI1_CLK_ROOT	180	266

Table continues on the next page...

Table 18-4. System Clock Frequency Values (continued)

Clock Root	Default Frequency (MHz)	Maximum Frequency (MHz)
IPU2_DI0_CLK_ROOT	132	266
IPU2_DI1_CLK_ROOT	180	266
LDB_DI0_SERIAL_CLK_ROOT	540	595
LDB_DI0_IPU	77.1	170
LDB_DI1_SERIAL_CLK_ROOT	540	595
LDB_DI1_IPU	77.1	170
SPDIF0_CLK_ROOT	31.8	66.5
SPDIF1_CLK_ROOT	31.8	66.5
ESAI_CLK_ROOT	31.8	66.5
HSI_TX_CLK_ROOT	198	198
CAN_CLK_ROOT	30	66.5
ECSPI_CLK_ROOT	60	66.5
UART_CLK_ROOT	80	80
VIDEO_27M_CLK_ROOT	27	27

18.5 Functional Description

This section provides a complete functional description of the block.

18.5.1 Clock Generation

18.5.1.1 External Low Frequency Clock - CKIL

The chip can use a 32 kHz or 32.768 kHz crystal as the external low-frequency source (XTALOSC). Throughout this chapter, the low-frequency crystal is referred to as the 32 kHz crystal.

This clock source should always be active when the chip is powered on. The 32 kHz entering the CCM are referred to as CKIL. CKIL is synchronized to IPG_CLK and supplied to modules that need it.

18.5.1.1.1 CKIL synchronizing to IPG_CLK

CKIL is synchronized to ipg_clk when the system is in functional mode. When the system is in STOP mode (when there is no IPG_CLK) the CKIL synchronizer is bypassed, and raw CKIL is supplied to the system.

18.5.1.2 External High Frequency Clock - CKIH and internal oscillator

The chip uses an internal oscillator to generate the reference clock (OSC). The internal oscillator is connected to the external crystal (XTALOSC) which generates the 24 MHz reference clock.

18.5.1.3 PLL reference clock

There are several PLLs in this chip.

PLL1 - ARM PLL (typical functional frequency 1 GHz)

PLL2 - System PLL (functional frequency 528 MHz)

PLL3 - USB1 PLL (functional frequency 480 MHz)

PLL4 - Audio PLL

PLL5 - Video PLL

PLL6 - ENET PLL

PLL7 - USB2 PLL (functional frequency 480 MHz)

PLL8 - MLB PLL

Some of the PLLs are described in the sections below. See CCM Analog Memory Map/ Register Definition for register information.

18.5.1.3.1 ARM PLL

This PLL synthesizes a low jitter clock from a 24 MHz reference clock. The clock output frequency for this PLL ranges from 650 MHz to 1.3 GHz. The output frequency is selected by a 7-bit register field CCM_ANALOG_PLL_ARM[DIV_SELECT].

PLL output frequency = Fref * DIV_SEL/2

NOTE

The upper frequency range may exceed the maximum frequency supported. Please see the datasheet for more information.

18.5.1.3.2 USB PLLs

These PLLs synthesize a low jitter clock from the 24 MHz reference clock. USB1 PLL has 4 frequency-programmable PFD (phase fractional divider) outputs.

The output frequency of USB1 PLL is 480 MHz. Even though USB1 PLL has a DIV_SELECT register field, this PLL should always be set to 480 MHz in normal operation. USB2 PLL is only used by the USB UTM interface through a direct connection.

18.5.1.3.3 System PLL

This PLL synthesizes a low jitter clock from the 24 MHz reference clock. The PLL has one output clock, plus 3 PFD outputs. The System PLL supports spread spectrum modulation for use in applications to minimize radiated emissions. The spread spectrum PLL output clock is frequency modulated so that the energy is spread over a wider bandwidth, thereby reducing peak radiated emissions. Due to this feature support, the associated lock time of this PLL is longer than other PLLs in the SoC that do not support spread spectrum modulation.

Spread spectrum operation is controlled by configuring the CCM_ANALOG_PLL_SYS_SS register. When enabled, the PLL output frequency will decrease by the amount defined in the STEP field, until it reaches the limiting frequency in the STOP field. The frequency will then similarly return to the original nominal frequency. The following equations control the spread-spectrum operation:

$$\text{Spread spectrum range} = \text{Fref} \times \frac{\text{CCM\_ANALOG\_PLL\_SYS\_SS[STOP]}}{\text{CCM\_ANALOG\_PLL\_SYS\_DENOM[B]}}$$

$$\text{Modulation frequency} = \text{Fref} \times \frac{\text{CCM\_ANALOG\_PLL\_SYS\_SS[STEP]}}{2 \times \text{CCM\_ANALOG\_PLL\_SYS\_SS[STOP]}}$$

Although this PLL does have a DIV_SELECT register field, it is intended that this PLL will only be run at the default frequency of 528 MHz.

This PLL also supports a Fractional-N synthesizer.

18.5.1.3.4 Audio / Video PLL

The audio PLL and video PLL each synthesize a low jitter clock from a 24 MHz reference clock. The clock output frequency range for this PLL is from 650 MHz to 1.3 GHz. It has a Fractional-N synthesizer.

There are /1, /2, /4, /8, /16 post dividers for the Video PLL and /1, /2, /4 post dividers for the Audio PLL. The output frequency can be set by programming the fields in the CCM_ANALOG_PLL_AUDIO, CCM_ANALOG_PLL_VIDEO, and CCM_ANALOG_MISC2 register sets according to the following equation.

PLL output frequency = Fref * (DIV_SELECT + NUM/DENOM)

18.5.1.3.5 MLB PLL

The MediaLB PLL is necessary in the MediaLB 6-Pin implementation to phase align the internal and external clock edges, effectively tuning out the delay of the differential clock receiver. The PLL is also responsible for generating the higher speed internal clock when the internal-to-external clock ratio is not 1:1.

18.5.1.3.6 Ethernet PLL

This PLL synthesizes a low jitter clock from the 24 MHz reference clock.

The PLL outputs a 500 MHz clock. The reference clocks generated by this PLL are:

- Ref_PCIe =125 MHz
- Ref_SATA=100 MHz
- Ref_ethernet, which is configurable based on the PLL_ENET[1:0] register field.

18.5.1.4 Phase Fractional Dividers (PFD)

There are several PFD outputs from the System PLL and USB1 PLL.

Each PFD output generates a fractional multiplication of the associated PLL's VCO frequency. Where the output frequency is equal to Fvco*18/N, N can range from 12-35. The PFDs allow for clock frequency changes without forcing the relock of the root PLL. This feature is useful in support of dynamic voltage and frequency scaling (DVFS). See CCM Analog Memory Map/Register Definition.

When the related PLL is powered up from the power down state or made to go through a relock cycle due to PLL regrogramming, it is required that the related PFDx_CLKGATE bit in CCM_ANALOG_PFD_480n or CCM_ANALOG_PFD_528n, be cycled on and off (1 to 0) after PLL lock. The PFDs can be in the clock gated state during PLL relock but

must be un-clock gated only after lock is achieved. See the engineering bulletin, *Configuration of Phase Fractional Dividers (EB790)* at www.freescale.com for procedure details.

18.5.1.5 CCM internal clock generation

The clock generation is comprised of two sub-modules:

CCM_CLK_SWITCHER

CCM_CLK_ROOT_GEN

18.5.1.5.1 Clock Switcher

The Clock Switcher (CCM_CLK_SWITCHER) sub-module receives the PLL output clocks and the PLL bypass clocks.

Figure 18-4 describes the generation of the three switcher clocks.

The figure also includes the Frequency Switch Control sub-module responsible for frequency change.

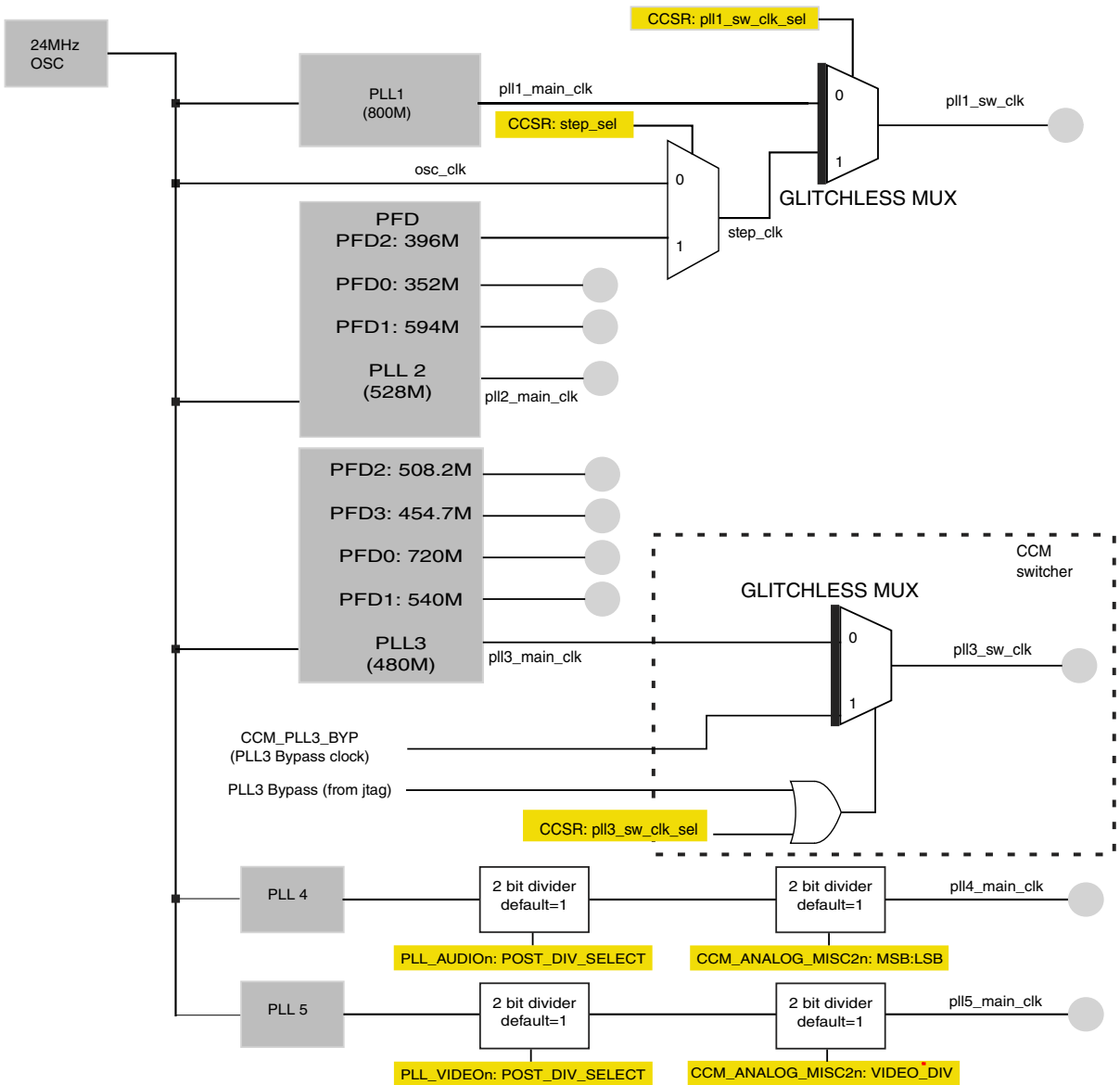


Figure 18-4. Switcher clock generation

18.5.1.5.2 PLL bypass procedure

In addition to PLL bypass options in CCM_ANALOG module, switcher and clk_root_gen sub-modules includes capability for each of the PLL clocks to be bypassed with an external bypass clock.

18.5.1.5.3 PLL clock change

In order to modify or stop the clock output of a specific PLL, all the clocks generated from the current PLL must be transitioned to the new PLL whose frequency is not being modified.

For clocks which can't be stopped (core and bus clocks), this should be done via the glitchless mux. Before changing the PLL setting, power it down. Power up the PLL after the change. See Disabling / Enabling PLLs for more information.

18.5.1.5.4 Clock Root Generator

The Clock Root Generator (CCM_CLK_ROOT_GEN) sub-module generates the root clocks to be delivered to LPCG.

The following figures describe clock generation. The frequencies in parantheses are the default typical frequencies.

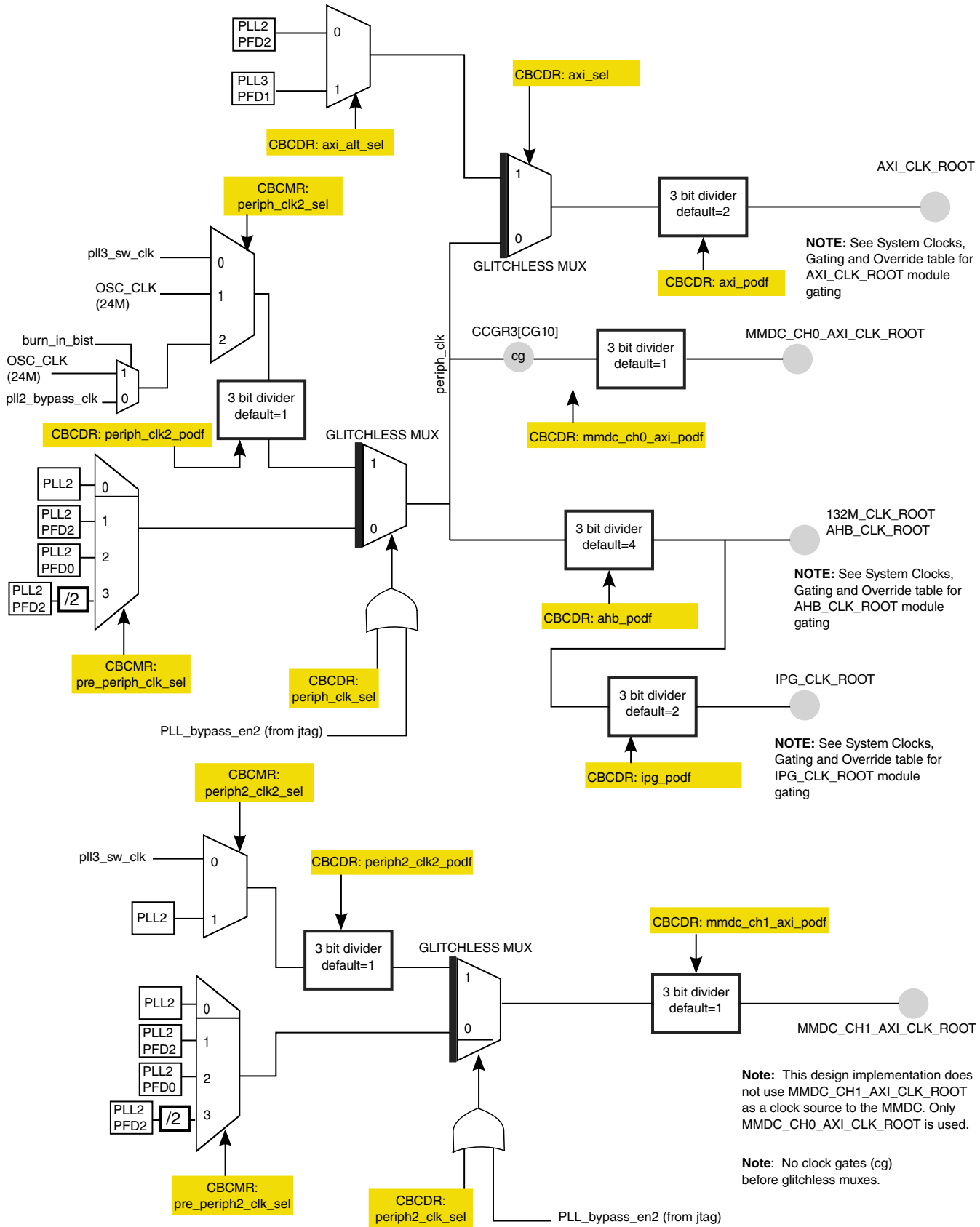
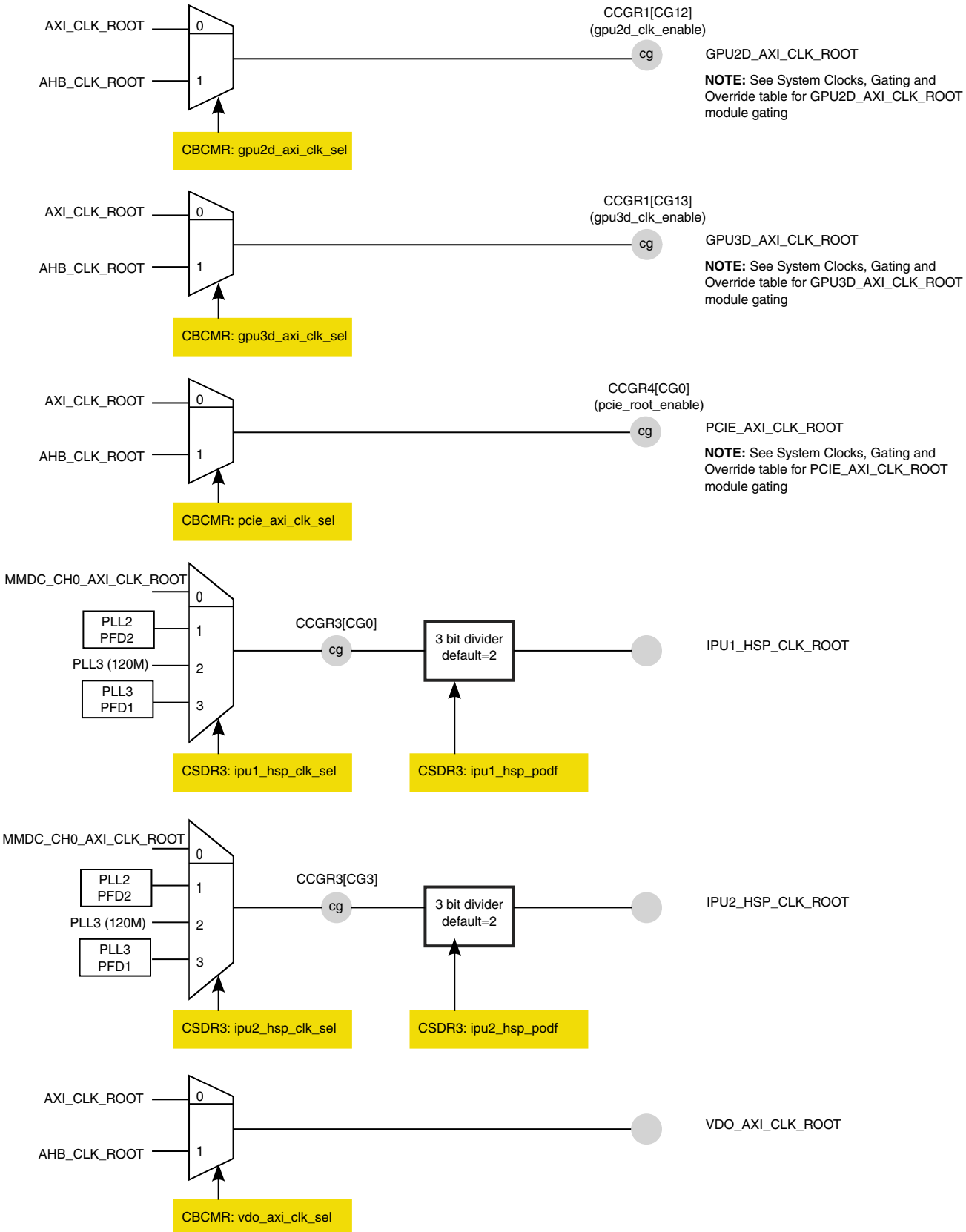


Figure 18-5. BUS clock generation


AXI_CLK_ROOT
AHB_CLK_ROOT
0
1
CBCMR: gpu2d_axi_clk_sel
CCGR1[CG12]
(gpu2d_clk_enable)
cg
GPU2D_AXI_CLK_ROOT
NOTE: See System Clocks, Gating and Override table for GPU2D_AXI_CLK_ROOT module gating
CBCMR: gpu3d_axi_clk_sel
CCGR1[CG13]
(gpu3d_clk_enable)
GPU3D_AXI_CLK_ROOT
NOTE: See System Clocks, Gating and Override table for GPU3D_AXI_CLK_ROOT module gating
CBCMR: pcie_axi_clk_sel
CCGR4[CG0]
(pcie_root_enable)
PCIE_AXI_CLK_ROOT
NOTE: See System Clocks, Gating and Override table for PCIE_AXI_CLK_ROOT module gating
MMDC_CH0_AXI_CLK_ROOT
PLL2 PFD2
PLL3 (120M)
PLL3 PFD1
2
3
CCGR3[CG0]
3 bit divider default=2
IPU1_HSP_CLK_ROOT
CSDR3: ipu1_hsp_clk_sel
CSDR3: ipu1_hsp_podf
CCGR3[CG3]
IPU2_HSP_CLK_ROOT
CSDR3: ipu2_hsp_clk_sel
CSDR3: ipu2_hsp_podf
VDO_AXI_CLK_ROOT
CBCMR: vdo_axi_clk_sel

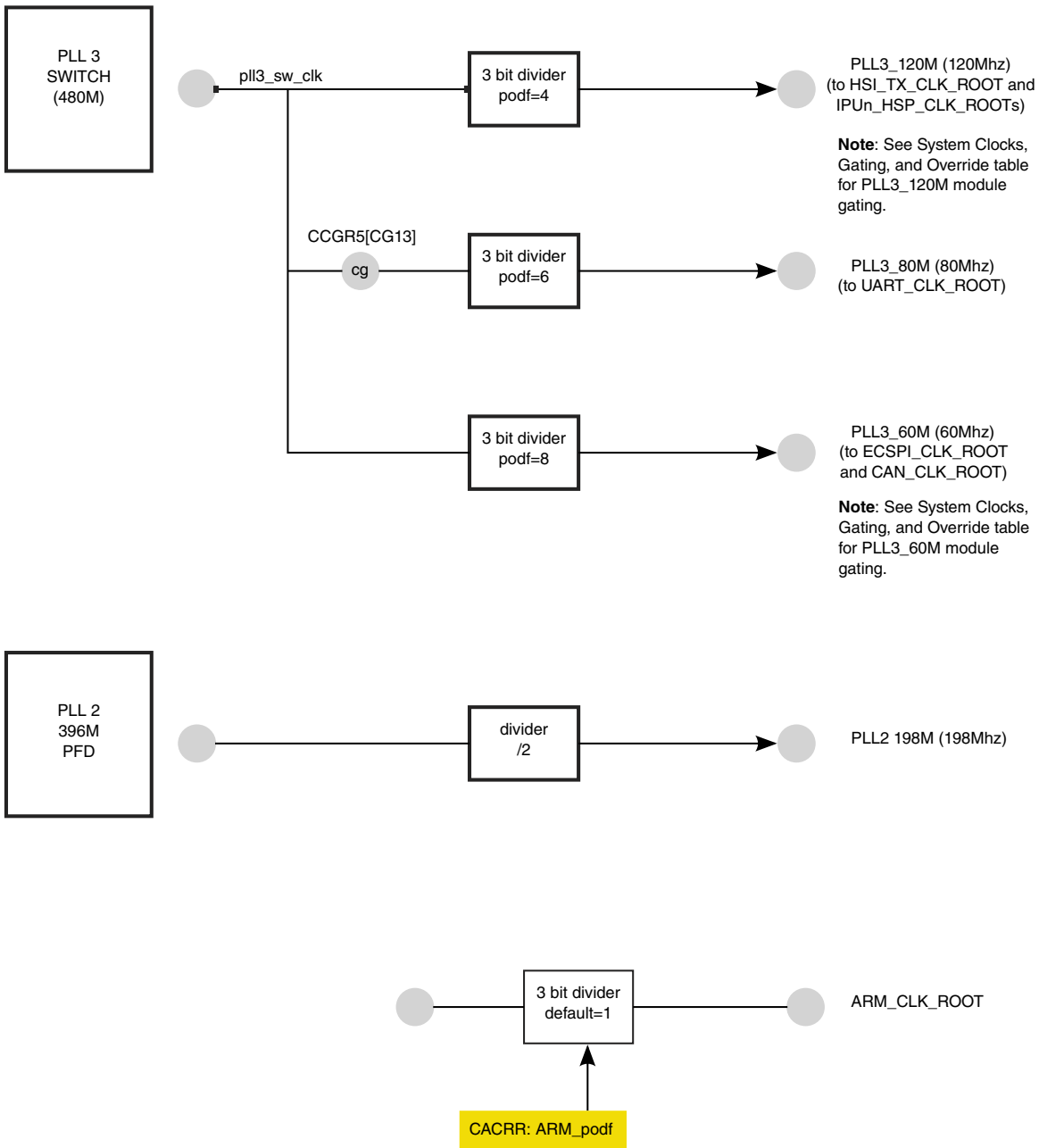
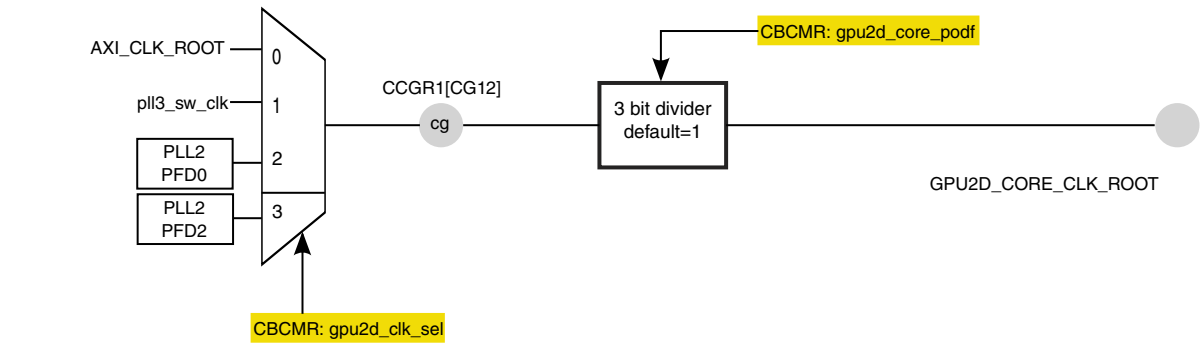
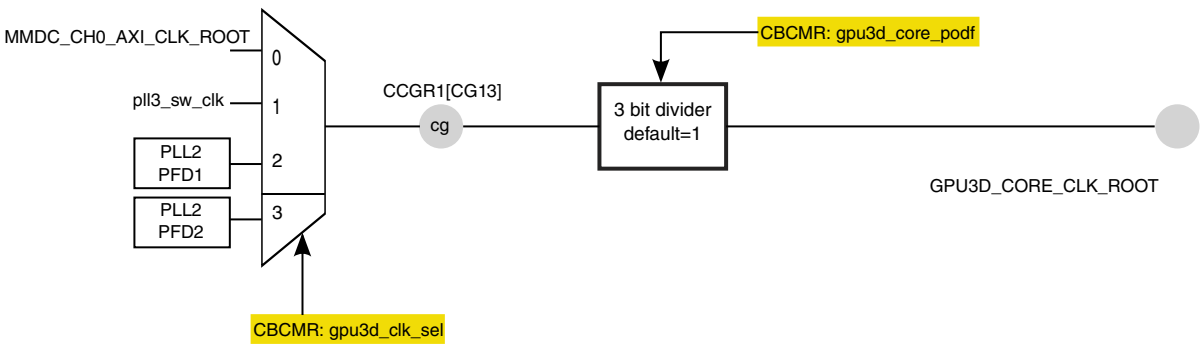
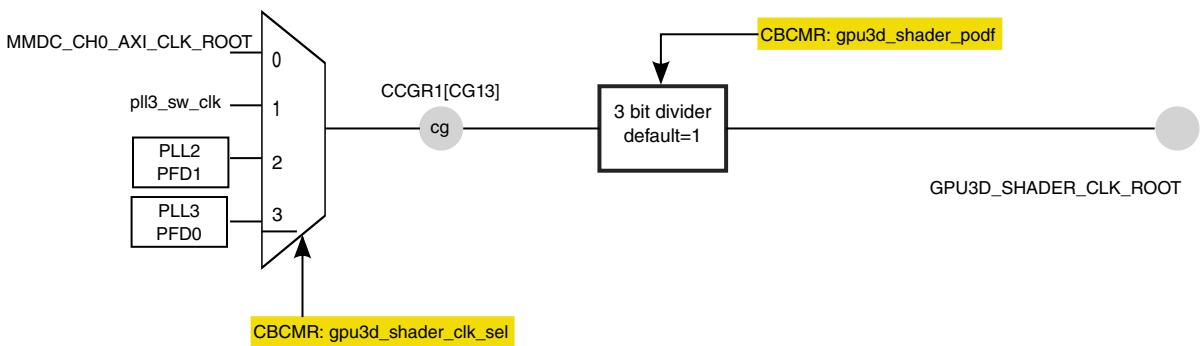
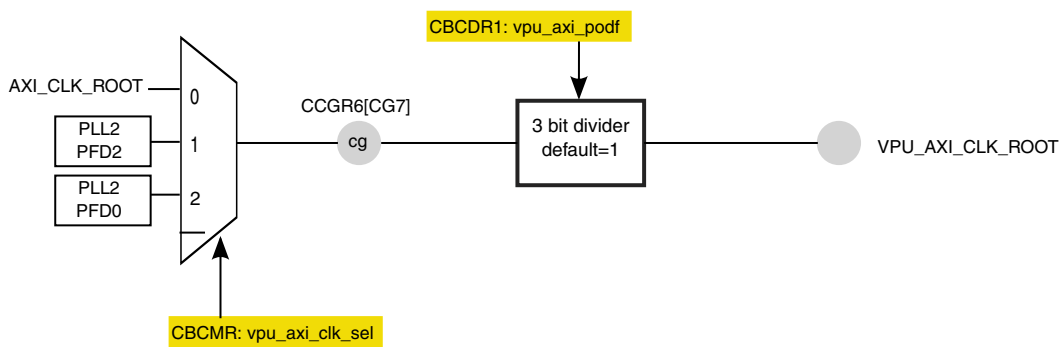


Figure 18-6. AXI clocks generation


AXI_CLK_ROOT
0
pll3_sw_clk
1
PLL2
PFD0
2
PLL2
PFD2
3
CBCMR: gpu2d_clk_sel
CCGR1[CG12]
cg
CBCMR: gpu2d_core_podf
3 bit divider
default=1
GPU2D_CORE_CLK_ROOT


MMDC_CH0_AXI_CLK_ROOT
0
pll3_sw_clk
1
PLL2
PFD1
2
PLL2
PFD2
3
CBCMR: gpu3d_clk_sel
CCGR1[CG13]
cg
CBCMR: gpu3d_core_podf
3 bit divider
default=1
GPU3D_CORE_CLK_ROOT


MMDC_CH0_AXI_CLK_ROOT
0
pll3_sw_clk
1
PLL2
PFD1
2
PLL3
PFD0
3
CBCMR: gpu3d_shader_clk_sel
CCGR1[CG13]
cg
CBCMR: gpu3d_shader_podf
3 bit divider
default=1
GPU3D_SHADER_CLK_ROOT


AXI_CLK_ROOT
0
PLL2
PFD2
1
PLL2
PFD0
2
CBCMR: vpu_axi_clk_sel
CCGR6[CG7]
cg
CBCDR1: vpu_axi_podf
3 bit divider
default=1
VPU_AXI_CLK_ROOT

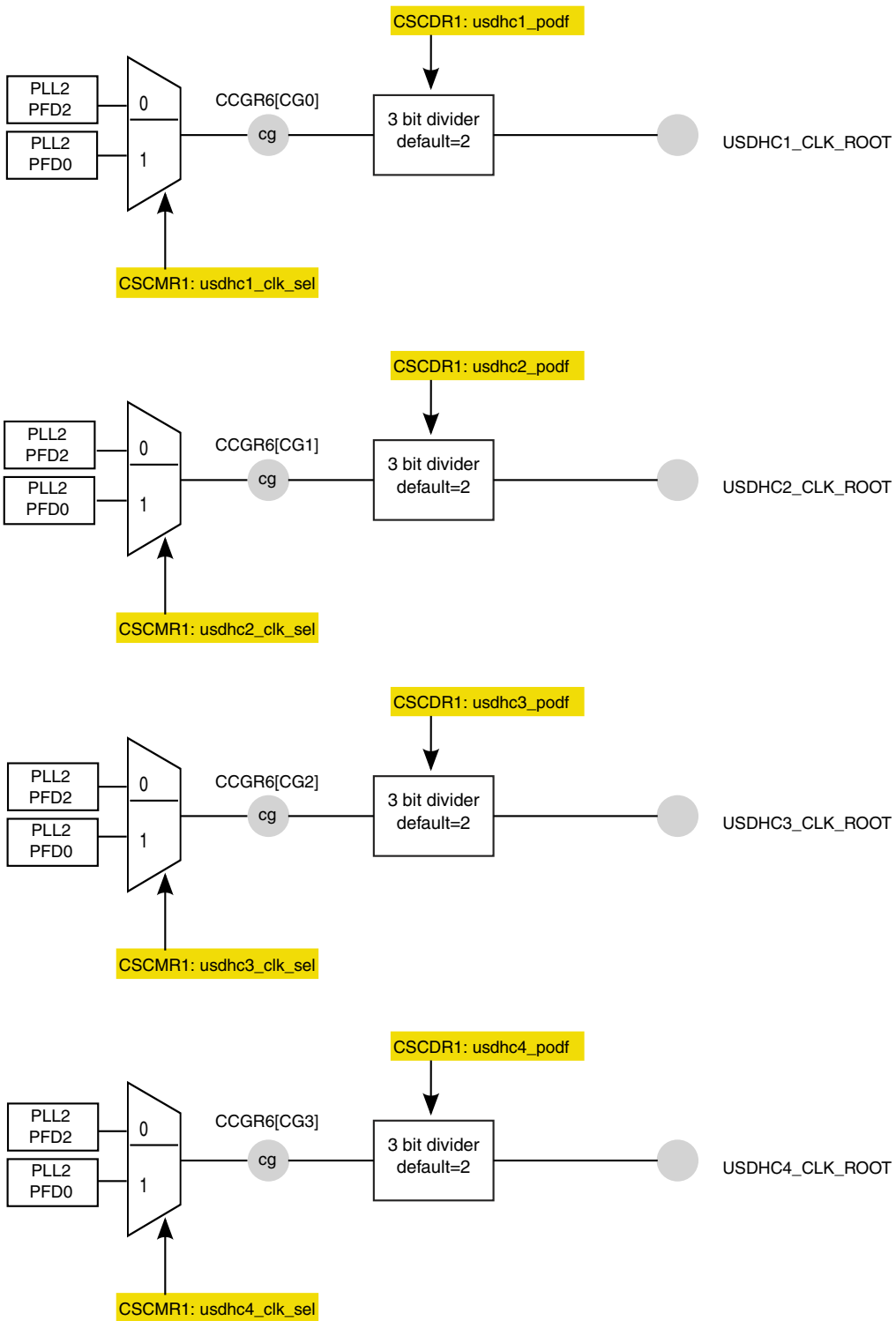
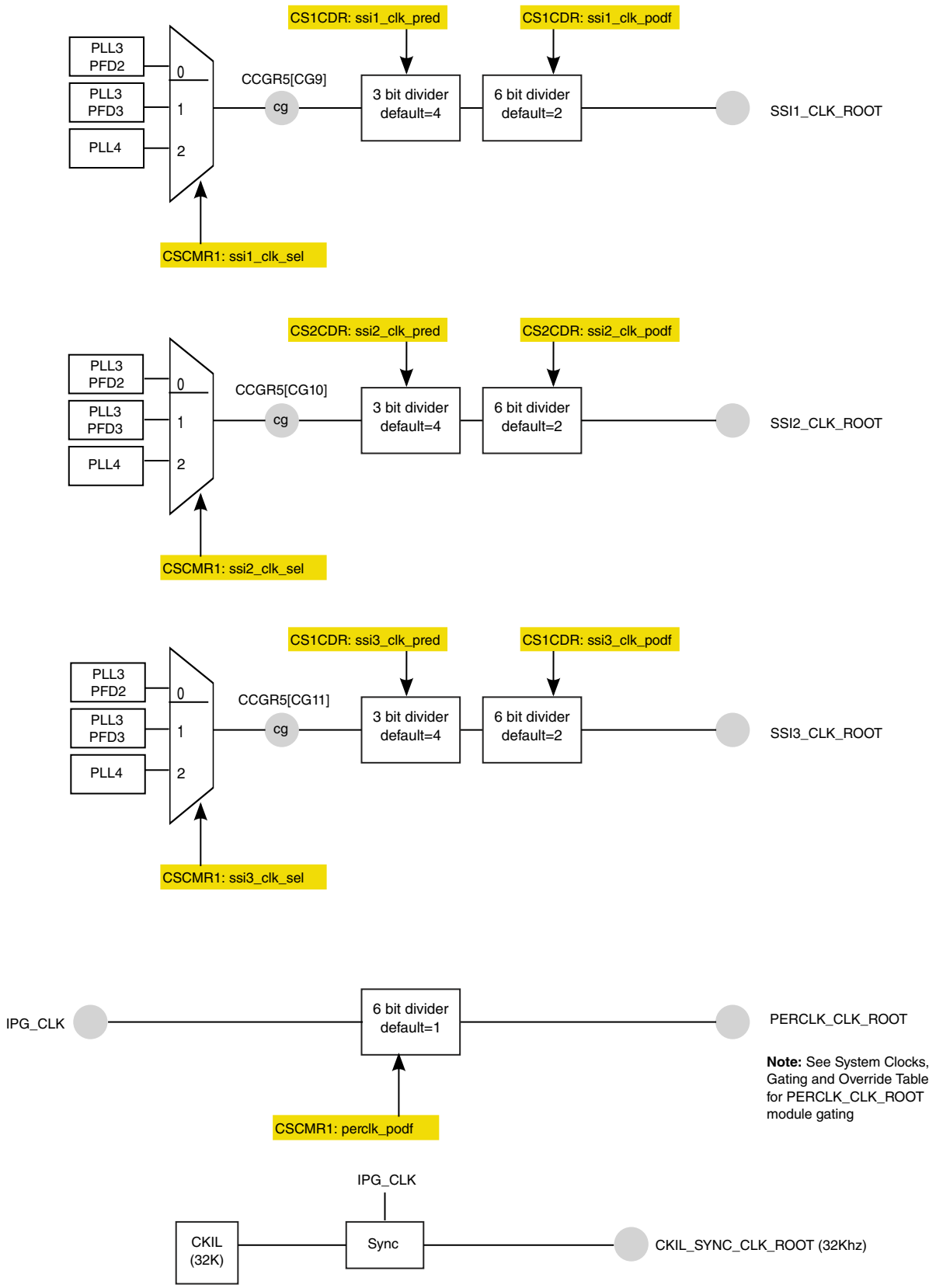
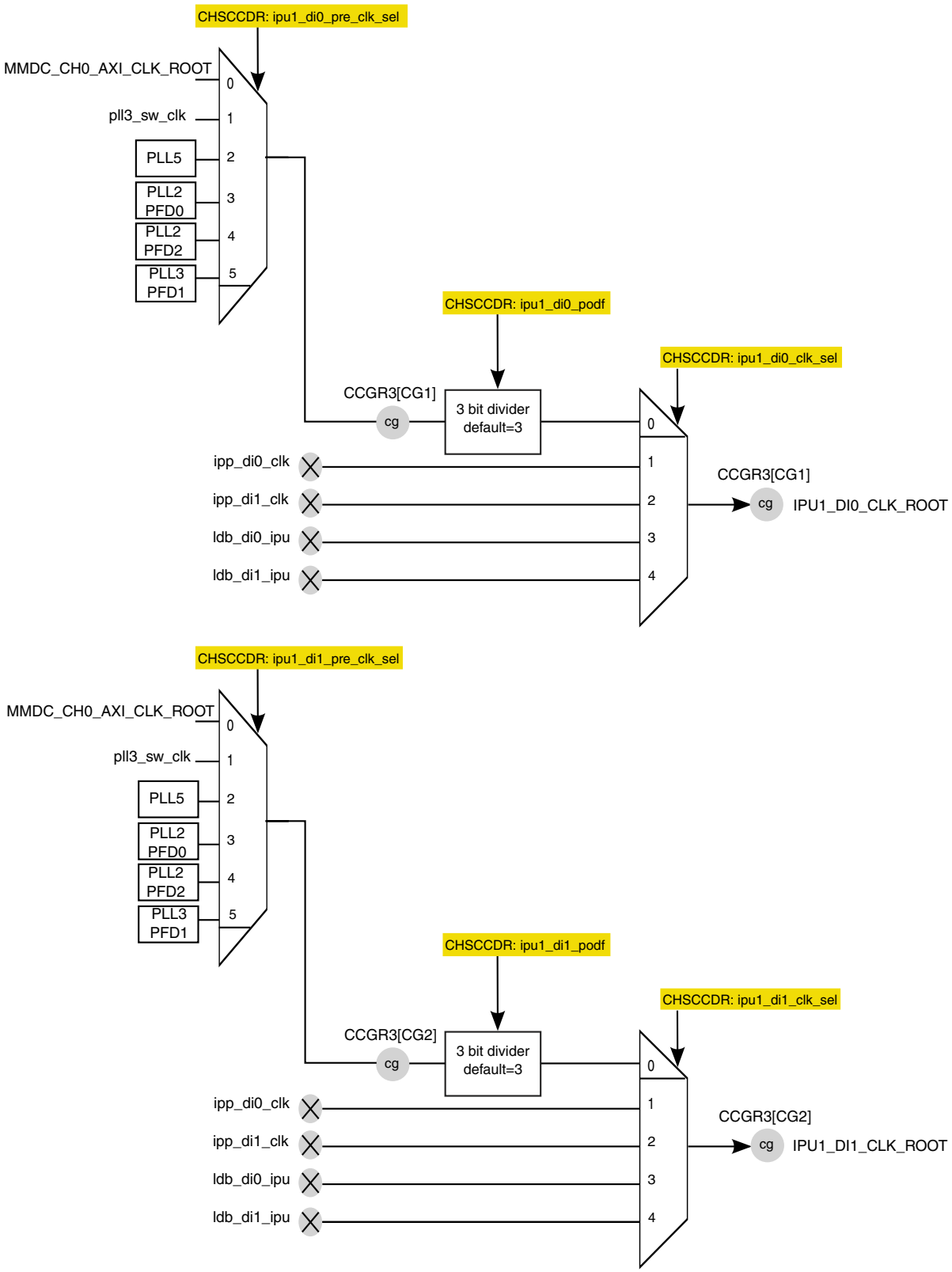
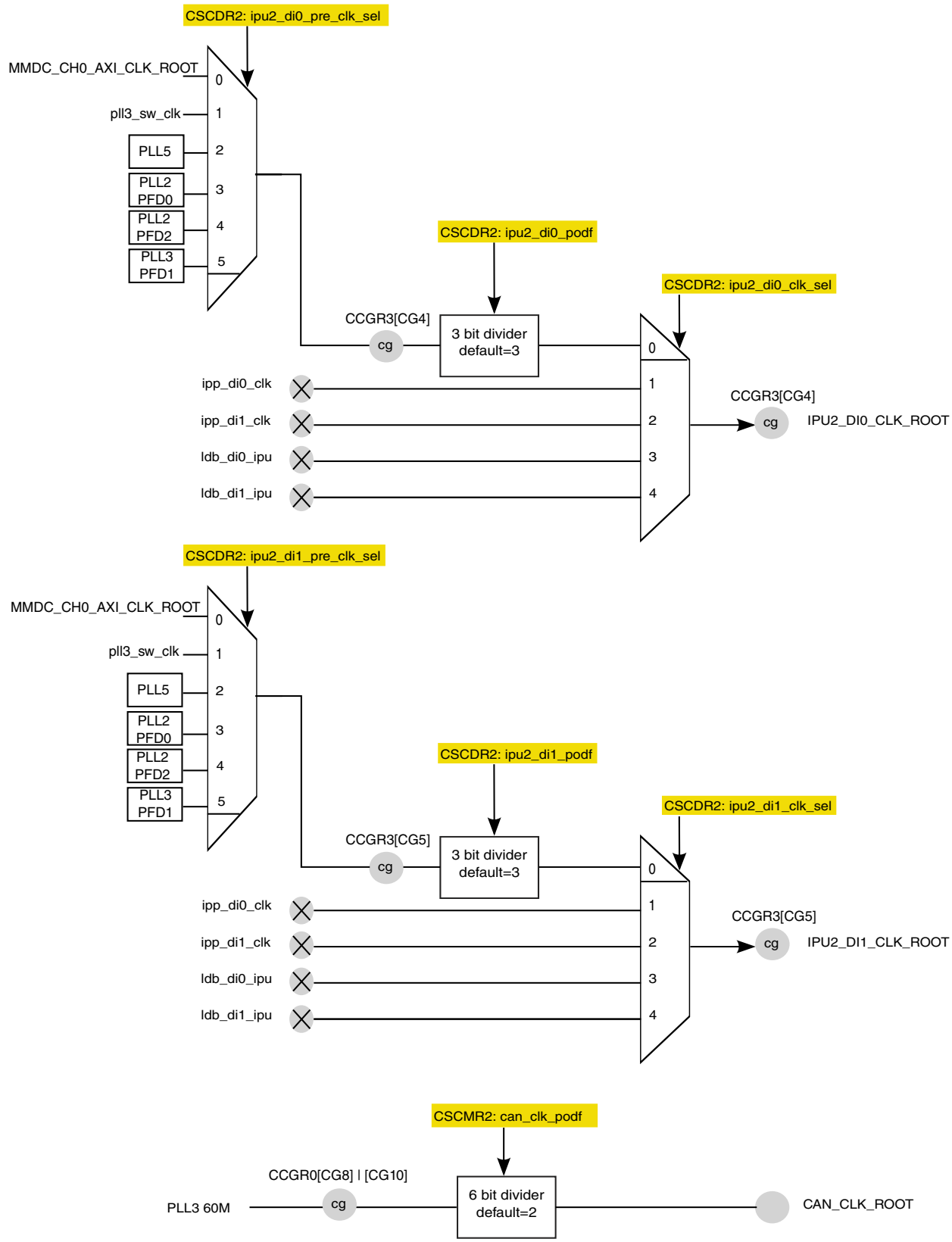


Figure 18-7. Serial clock generation



Note: See System Clocks, Gating and Override Table for PERCLK_CLK_ROOT module gating


CHSCCDR: ipu1_di0_pre_clk_sel
MMDC_CH0_AXI_CLK_ROOT
pll3_sw_clk
PLL5
PLL2 PFD0
PLL2 PFD2
PLL3 PFD1
0
1
2
3
4
5
CHSCCDR: ipu1_di0_podf
CHSCCDR: ipu1_di0_clk_sel
CCGR3[CG1]
cg
3 bit divider default=3
ipp_di0_clk
ipp_di1_clk
ldb_di0_ipu
ldb_di1_ipu
CCGR3[CG1]
IPU1_DI0_CLK_ROOT
CHSCCDR: ipu1_di1_pre_clk_sel
CHSCCDR: ipu1_di1_podf
CHSCCDR: ipu1_di1_clk_sel
CCGR3[CG2]
IPU1_DI1_CLK_ROOT


CSCDR2: ipu2_di0_pre_clk_sel
MMDC_CH0_AXI_CLK_ROOT
pll3_sw_clk
PLL5
PLL2 PFD0
PLL2 PFD2
PLL3 PFD1
0
1
2
3
4
5
CSCDR2: ipu2_di0_podf
CCGR3[CG4]
cg
3 bit divider default=3
CSCDR2: ipu2_di0_clk_sel
ipp_di0_clk
ipp_di1_clk
ldb_di0_ipu
ldb_di1_ipu
CCGR3[CG4]
IPU2_DI0_CLK_ROOT
CSCDR2: ipu2_di1_pre_clk_sel
CSCDR2: ipu2_di1_podf
CCGR3[CG5]
CSCDR2: ipu2_di1_clk_sel
IPU2_DI1_CLK_ROOT
CSCMR2: can_clk_podf
CCGR0[CG8] | [CG10]
PLL3 60M
6 bit divider default=2
CAN_CLK_ROOT

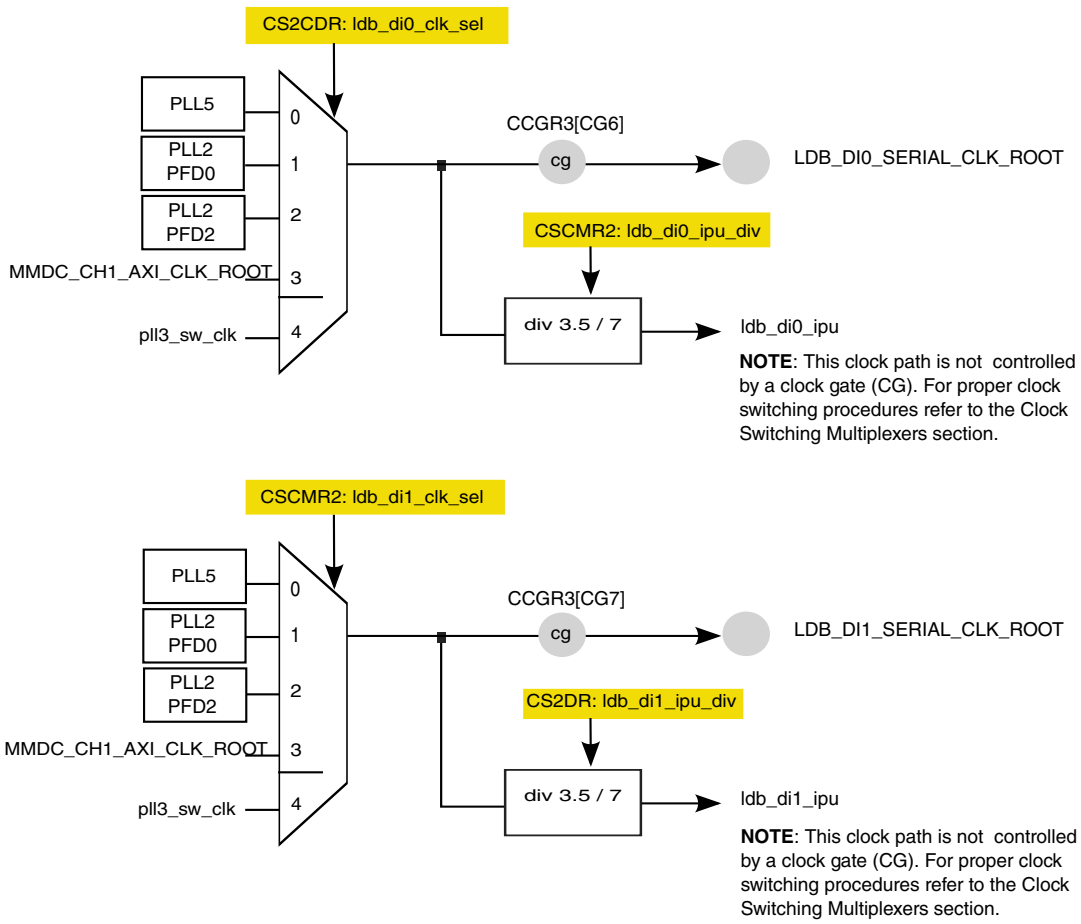
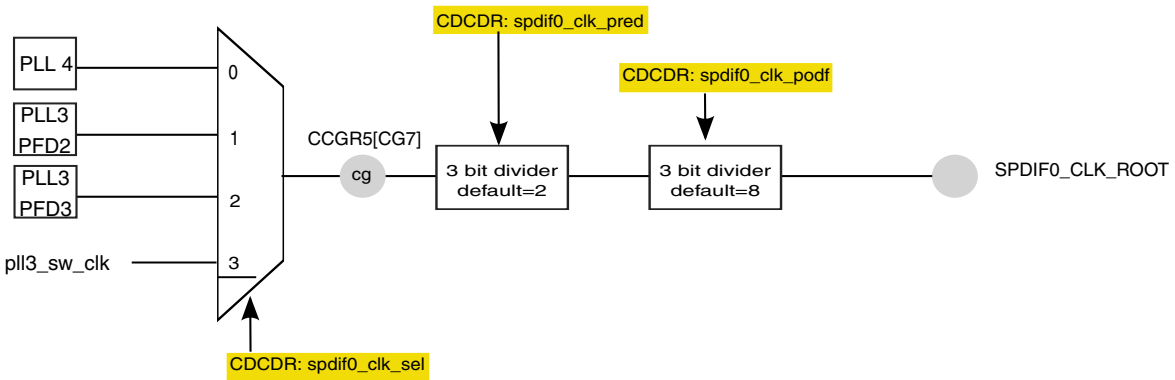
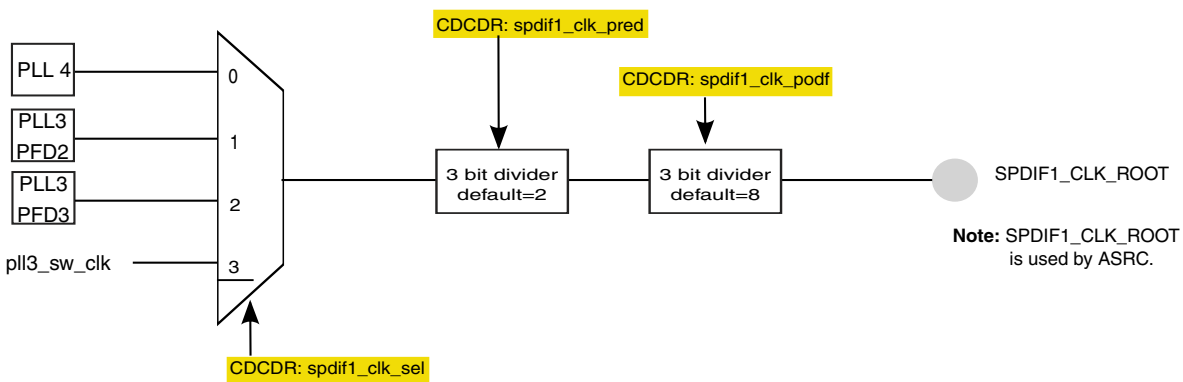
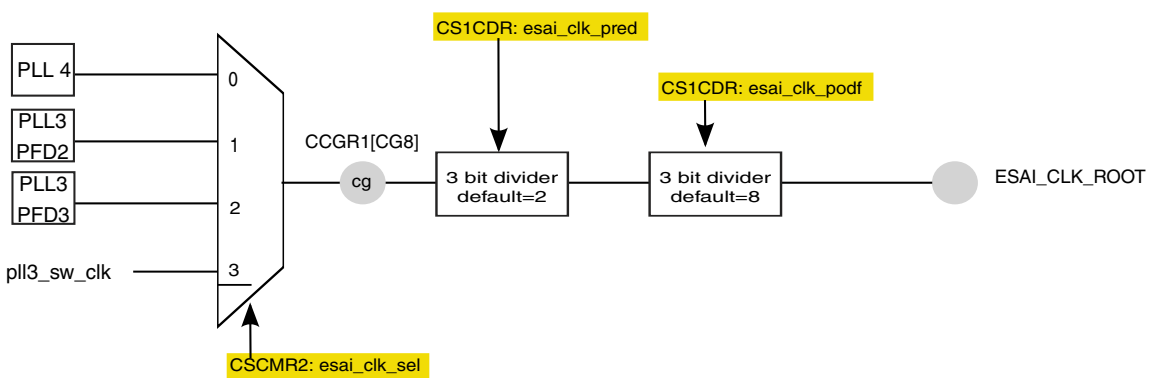
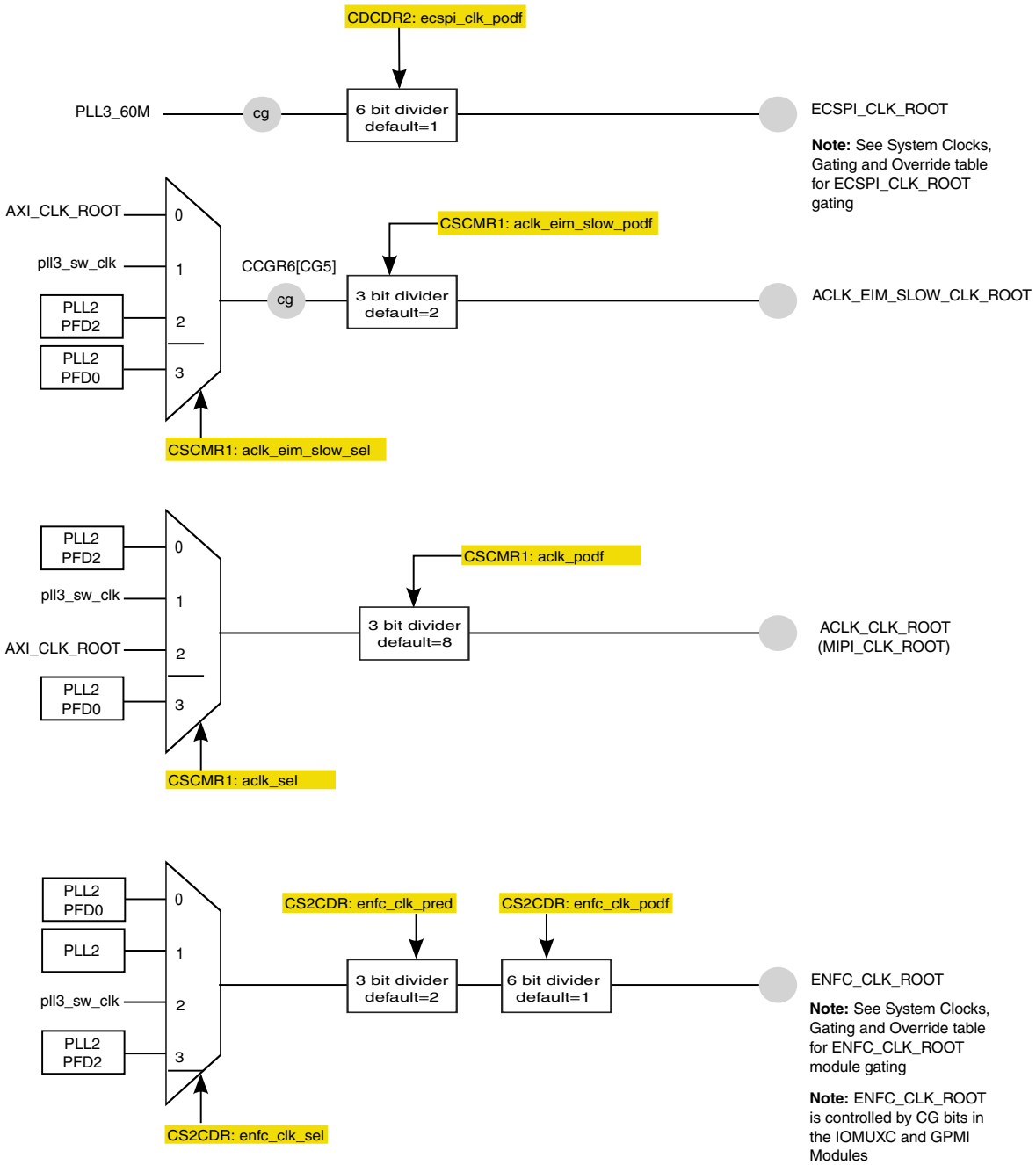


Figure 18-8. Serial clock generation (cont)


PLL 4
PLL3 PFD2
PLL3 PFD3
pll3_sw_clk
0
1
2
3
CCGR5[CG7]
cg
CDCDR: spdif0_clk_pred
3 bit divider default=2
CDCDR: spdif0_clk_podf
3 bit divider default=8
SPDIF0_CLK_ROOT
CDCDR: spdif0_clk_sel


PLL 4
PLL3 PFD2
PLL3 PFD3
pll3_sw_clk
0
1
2
3
CDCDR: spdif1_clk_pred
3 bit divider default=2
CDCDR: spdif1_clk_podf
3 bit divider default=8
SPDIF1_CLK_ROOT
Note: SPDIF1_CLK_ROOT is used by ASRC.
CDCDR: spdif1_clk_sel


PLL 4
PLL3 PFD2
PLL3 PFD3
pll3_sw_clk
0
1
2
3
CCGR1[CG8]
cg
CS1CDR: esai_clk_pred
3 bit divider default=2
CS1CDR: esai_clk_podf
3 bit divider default=8
ESAI_CLK_ROOT
CSCMR2: esai_clk_sel


CDCDR2: ecspi_clk_podf
PLL3_60M
cg
6 bit divider
default=1
ECSPI_CLK_ROOT
Note: See System Clocks, Gating and Override table for ECSPI_CLK_ROOT gating
AXI_CLK_ROOT
pll3_sw_clk
PLL2 PFD2
PLL2 PFD0
0
1
2
3
CCGR6[CG5]
cg
CSCMR1: aclk_eim_slow_podf
3 bit divider
default=2
ACLK_EIM_SLOW_CLK_ROOT
CSCMR1: aclk_eim_slow_sel
PLL2 PFD2
pll3_sw_clk
AXI_CLK_ROOT
PLL2 PFD0
CSCMR1: aclk_podf
3 bit divider
default=8
ACLK_CLK_ROOT
(MIPI_CLK_ROOT)
CSCMR1: aclk_sel
PLL2 PFD0
PLL2
pll3_sw_clk
PLL2 PFD2
CS2CDR: enfc_clk_pred
CS2CDR: enfc_clk_podf
3 bit divider
default=2
6 bit divider
default=1
ENFC_CLK_ROOT
Note: See System Clocks, Gating and Override table for ENFC_CLK_ROOT module gating
Note: ENFC_CLK_ROOT is controlled by CG bits in the IOMUXC and GPMI Modules
CS2CDR: enfc_clk_sel

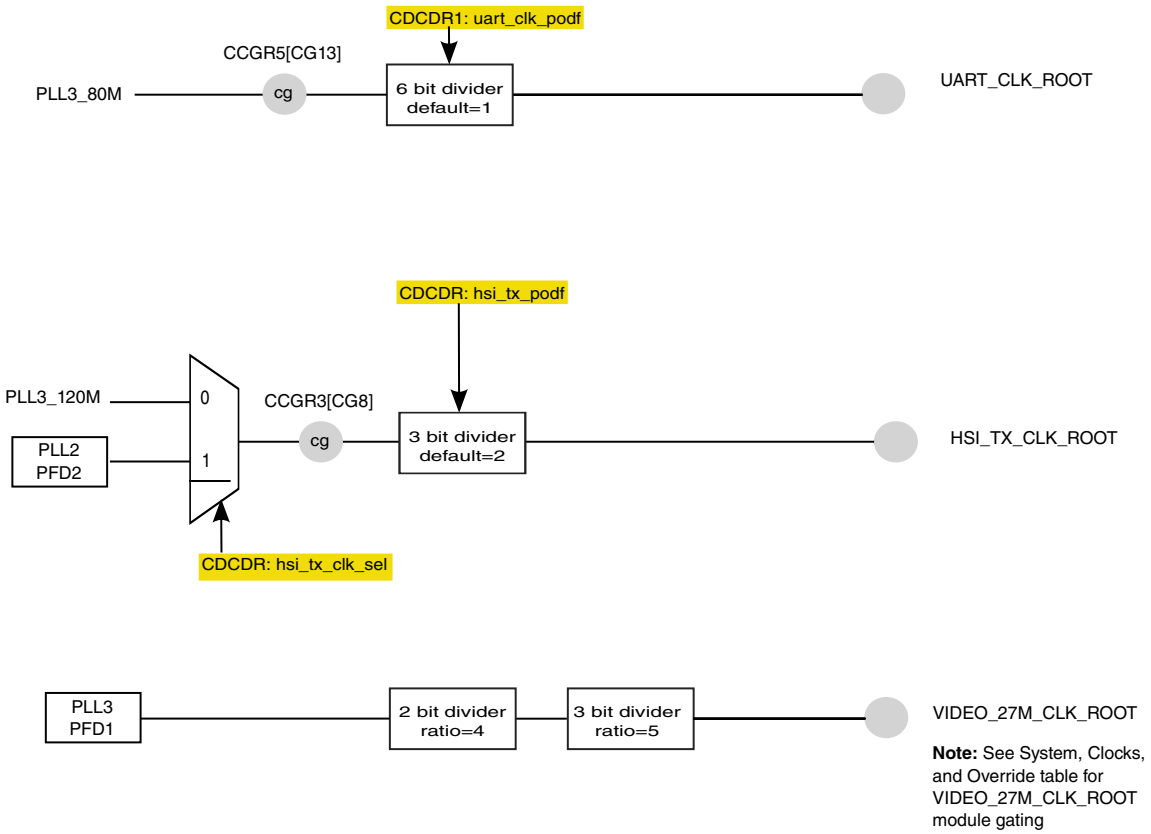


Figure 18-9. UART, HSI, Video clock generation (cont)

NOTE

All 6-bit PODF dividers found in the diagrams above can operate on low frequency.

18.5.1.5.5 Initial values controlled by the System JTAG Controller (SJC).

The initial values of the following dividers and muxes can be controlled by SJC.

In regular functional mode, the SJC will drive the reset values stated in the CCM register memory map. If SJC is programmed to change those values, then the reset value for those dividers/muxes will be taken from the SJC programability.

Software can update the changed reset value after reset sequence. The control signals and the dividers/muxes are listed below:

- [2:0] init_mmdc_ch1_axi_podf
- [2:0] init_periph2_clk2_podf
- [1:0] init_ipg_podf
- [2:0] init_ahb_podf

- [2:0] init_axi_podf
- [2:0] init_mmdc_ch0_axi_podf
- [2:0] init_periph_clk2_podf
- init_periph_clk_sel
- init_periph2_clk_sel

18.5.1.5.6 Divider change handshake

Modifying the following dividers will start the handshake with MMDC CH1 and/or CH0.

- mmdc_ch0_axi_podf
- mmdc_ch1_axi_podf
- periph_clk_sel
- periph2_clk_sel
- arm_podf
- axi_podf
- ahb_podf

18.5.1.6 Disabling / Enabling PLLs

PLL disabling and enabling is done via analog module.

Before disabling a PLL using the analog registers, software should first move all the clocks generated from that specific PLL to another source. This alternate source could be another PLL, or a PFD driven by another PLL. Alternatively, software can bypass the PLL and use the PLL reference clock (usually 24MHz) as the output clock. Bypassing the PLL is done by setting the analog BYPASS bit in the control register for that PLL.

18.5.1.7 Clock Switching Multiplexers

There are a multitude of multiplexers available throughout the clock generation logic that provide alternate clock sources for the system clocks controlled by the CCM. The CCM utilizes several synchronous glitchless clock multiplexers as well as asynchronous glitchy clock multiplexers.

Synchronous muxes ensure there are no glitches between the transition of two asynchronous clocks and that there will be no pulses that are of a frequency higher than either input clock. In order for the synchronous multiplexer to work properly, both the current clock and the clock to be selected must remain active during the entire selection process.

There are five glitchless (synchronous) muxes used in the CCM. The table below lists the muxes and the respective control bits.

Table 18-5. Glitchless Multiplexers

Glitchless Mux	Mux Select Bit	Handshake Bit
periph_clk_mux	CBCDR[periph_clk_sel]	CDHIPR[periph_clk_sel_busy]
periph2_clk_mux	CBCDR[periph2_clk_sel]	CDHIPR[periph2_clk_sel_busy]
axi_alt_clk_mux	CBCDR[axi_sel]	
pll3_sw_clk_mux	CCSR[pll3_sw_clk_sel]	
pll1_sw_clk_mux	CCSR[pll1_sw_clk_sel]	

NOTE

Any change of the periph_clk_sel and periph2_clk_sel sync mux select will involve handshake with the MMDC. Refer to the CCDR and CDHIPR registers for the handshake bypass and busy bits.

Table 18-6. Multiplexers

Signal	Mux Select Bit	Glitchless
pll1_sw_clk	CCSR[pll1_sw_clk_sel]	Yes
step_clk	CCSR[step_sel]	No
pll3_sw_clk	CCSR[pll3_sw_clk_sel]	Yes
axi_alt	CBCDR[axi_alt_sel]	No
AXI_CLK_ROOT	CBCDR[axi_sel]	Yes
periph_clk2	CBCMR[periph_clk2_sel]	No
periph_clk	CBCDR[periph_clk_sel]	Yes
pre_periph_clk	CBCMR[pre_periph_clk_sel]	No
periph2_clk2	CBCMR[periph2_clk2_sel]	No
pre_periph2_clk	CBCMR[pre_periph2_clk_sel]	No
periph2_clk	CBCDR[periph2_clk_sel]	Yes
GPU2D_AXI_CLK_ROOT	CBCMR[gpu2d_axi_clk_sel]	No
GPU3D_AXI_CLK_ROOT	CBCMR[gpu3d_axi_clk_sel]	No
PCIE_AXI_CLK_ROOT	CBCMR[pcie_axi_clk_sel]	No
IPU1_HSP_CLK_ROOT	CSCDR3[ipu1_hsp_clk_sel]	No
IPU2_HSP_CLK_ROOT	CSCDR3[ipu2_hsp_clk_sel]	No
VDO_AXI_CLK_ROOT	CBCMR[vdo_axi_clk_sel]	No
GPU2D_CORE_CLK_ROOT	CBCMR[gpu2d_core_clk_sel]	No
GPU3D_CORE_CLK_ROOT	CBCMR[gpu3d_core_clk_sel]	No
GPU3D_SHADER_CLK_ROOT	CBCMR[gpu3d_shader_clk_sel]	No
VPU_AXI_CLK_ROOT	CBCMR[vpu_axi_clk_sel]	No
USDHC1_CLK_ROOT	CSCMR1[usdhc1_clk_sel]	No

Table continues on the next page...

Table 18-6. Multiplexers (continued)

Signal	Mux Select Bit	Glitchless
USDHC2_CLK_ROOT	CSCMR1[usdhc2_clk_sel]	No
USDHC3_CLK_ROOT	CSCMR1[usdhc3_clk_sel]	No
USDHC4_CLK_ROOT	CSCMR1[usdhc4_clk_sel]	No
SSI1_CLK_ROOT	CSCMR1[ssi1_clk_sel]	No
SSI2_CLK_ROOT	CSCMR1[ssi2_clk_sel]	No
SSI3_CLK_ROOT	CSCMR1[ssi3_clk_sel]	No
IPU1_DI0_CLK_ROOT	CHSCCDR[ipu1_di0_clk_sel]	No
IPU1_DI1_CLK_ROOT	CHSCCDR[ipu1_di1_clk_sel]	No
IPU2_DI0_CLK_ROOT	CSCDR2[ipu2_di0_clk_sel]	No
IPU2_DI1_CLK_ROOT	CSCDR2[ipu2_di1_clk_sel]	No
LDB_DI0_SERIAL_CLK_ROOT	CS2CDR[ldb_di0_clk_sel]	No
LDB_DI1_SERIAL_CLK_ROOT	CS2CDR[ldb_di1_clk_sel]	No
SPDIF0_CLK_ROOT	CDCDR[spdif0_clk_sel]	No
SPDIF1_CLK_ROOT	CDCDR[spdif1_clk_sel]	No
ESAI_CLK_ROOT	CSCMR2[esai_clk_sel]	No
ACLK_EIM_SLOW_CLK_ROOT	CSCMR1[aclk_eim_slow_sel]	No
ACLK_CLK_ROOT	CSCMR1[aclk_sel]	No
ENFC_CLK_ROOT	CS2CDR[enfc_clk_sel]	No
HSI_TX_CLK_ROOT	CDCDR[hsi_tx_clk_sel]	No

For critical system bus clocks, changing the clock source can be done in the CCM using the glitchless clock muxes in Figure 18-5 . In the figure, the thick bar on the input side indicates the glitchless muxes. Those without the thick bar are regular muxes (not glitchless).

For example, before disabling PLL2, software can switch the FABRIC_CLK_ROOT away from the PLL2 or one of its PFDs by programming CBCMR[PERIPH2_CLK2_SEL] and CBCDR[PERIPH2_CLK2_PODF] to provide an appropriate frequency clock, then glitchlessly switch to it by programming CBCDR[PERIPH2_CLK_SEL].

Asynchronous multiplexers or glitchy multiplexers, allow the clock to switch immediately after the multiplexer select changed. This immediate switch of two asynchronous clock domains can cause the output clock to glitch. Since both clock sources to the mux are asynchronous, switching the clocks from one source to the other can cause a glitch to be generated, regardless of the input clock source.

The input clocks to the mux are required to be gated before switching the source clock in the CCM clock mux and the output should also be gated. If the input and output clocks are not gated, clock glitches can propagate to the logic that follows the clock mux, causing the logic to behave unpredictably.

For serial clocks, software should first disable the module, then gate its clock in the LPCG. Then it should move the mux controlling the source of the clocks to another PLL, and reset the module and its clocks. Only then is it safe to disable the PLL. The mux for the serial clocks is not glitchless.

18.5.1.8 Low Power Clock Gating module (LPCG)

The LPCG module receives the root clocks and splits them to clock branches for each module. The clock branches are gated clocks.

The enables for those gates can come from four sources:

1. Clock enable signal from CCM - this signal is generated by configuring of the CGR bits in the CCM. It is based on the low power mode.
2. Clock enable signal from the module - this signal is generated by the module based on internal logic of the module. Not every enable signal from the module is used. For used clock enable signals from the module, CCM will generate an override signal based on a programable bit in CCM (CMEOR).
3. Clock enable signal from Reset controller (SRC) - this signal will enable the clock during the reset procedure. Please see the SRC chapter for details on the clock enable signal during reset procedure.
4. Hard-coded enable from fuse box.

These enable signals are ANDed to generate the enable signal for the gating cell.

The enable signal for the gating cell is synchronized with the clock it needs to gate in order to prevent glitches on the gated clock.

Notifications are generated for CCM to indicate when clock roots should be opened and closed. All notifications that correspond to the same clock root will be ORed to generate one notification signal to CCM for clock root gating.

The following figure describes the clock split inside the LPCG module. It describes the case of two modules; one module is without an enable signal and one is shown with an enable signal. SRC enable signals and sync flip flops are omitted from this figure.

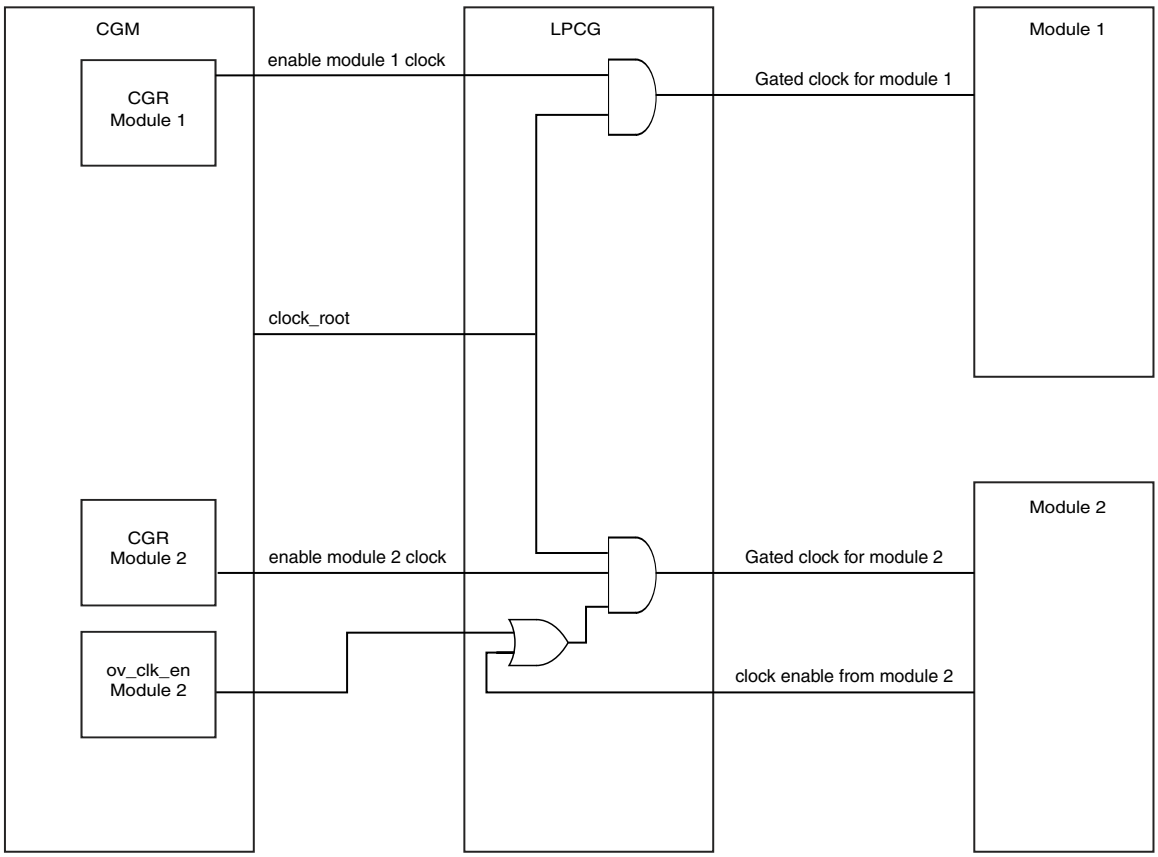


Figure 18-10. Clock split in LPCG

18.5.1.8.1 MMDC handshake

CCM will assert the mmdc_freq_change_req signal.

MMDC will assert the mmdc0_freq_change_ack and mmdc1_freq_change_ack signals to acknowledge that the frequency change request has been received and that the frequency can now be changed safely.

CCM will commence the actual change of division ratio of mmdc0 and/or mmdc1 dividers or apply mux change on root clocks once both of the non-masked acknowledges are asserted.

NOTE

MMDC handshakes can be masked.

18.5.2 DVFS support

When performing DVFS, the frequency shift procedure for the ARM core clock domain can be performed by software.

CPU PLL frequency and CCM ARM clock divider is controlled by CCM and CPU power domain supply voltage value is controlled by CCM_ANALOG module.

NOTE

The frequency should be shifted down first and then voltage value reduced, and vice-versa, when shifting the frequency up.

NOTE

CCM_ANALOG will not control the voltage value in Bypass mode

18.5.3 Power modes

The chip supports 3 low power modes: RUN mode, WAIT mode, STOP mode.

18.5.3.1 RUN mode

This is the normal/functional operating mode. In this mode, the CPU runs in its normal operational mode. Clocks to the modules can be gated by configuring the corresponding CCGRx bits.

18.5.3.2 WAIT mode

In this mode the CPU clock is gated. All other clocks are functional and can be gated by programming their CGR bits when all ARM cores are in WFI, and L2 cache and SCU are idle.

18.5.3.2.1 Entering WAIT mode

If the CLPCR[LPM] bit is set by software to WAIT mode, when CPU executes the next wait for interrupt (WFI) instruction, WAIT mode sequence will start.

As part of the WFI routine, alternative interrupt controller in GPC should be updated; the CPU platform interrupt controller will be disabled first by software and will be not functional, due to clock gating. Interrupts during WAIT mode are monitored by alternative interrupt controller.

After execution of the WFI routine, the CPU platform will assert idle signals for each component of the platform and CCM will gate clock to the platform.

The next actions can be programmed during WAIT mode:

1. CCM requests an acknowledge to close clocks to MMDC if its CGR bits indicate to close its clocks on WAIT mode, and if those clocks are not already closed in run mode. The request will be issued if the handshake is not bypassed by programing the CLPCR register. If the corresponding bits are set, the request signal will not be issued to the corresponding module and CCM will not wait for its acknowledge in the process of entering low power mode. Once CCM receives all the acknowledge signals needed, then it will enter WAIT mode.
2. Close the clocks to the modules which were defined to be shut at WAIT mode in the CCGR bits.
3. Observability to indicate WAIT mode.

NOTE

Setting MMDC CGR bits to 01 can hang the entire system since the MMDC clock and fabric clock share the same clock root.

Any enabled interrupt assertion will start the exit from WAIT mode.

18.5.3.2.2 Exiting WAIT mode

As soon as enabled interrupt is asserted, CPU supply will be restored if CPU SRPG was applied and clocks are enabled to CPU and other modules.

18.5.3.3 STOP mode

In this mode all system clocks are stopped, along with the CPU, system buses and all PLLs. Power gating can be applied for ARM platform, GPU3D, GPU2D and VPU. External supply voltage can be reduced to decrease leakage.

18.5.3.3.1 Entering STOP mode

Procedure entering STOP mode is the same, as entering WAIT mode until the moment of disabling clocks to modules. (LPM bit should be configured to STOP mode.)

After clocks to modules are gated, the following actions will be taken:

- PLLs are disabled
- CCM_PMIC_STBY_REQ asserted, if vstby bit is set

- osc_en signal is negated
- osc_pwrdn is asserted, if sbyos bit is set

Counter will be triggered after CCM_PMIC_STBY_REQ assertion to allow to external regulator or PMIC to decrease voltage until valid voltage range. On counter completion, stop_mode signal will be asserted, that will trigger disabling analog elements in anatop.

CCM's low power state machine will remain in state STOP_GPC until STOP mode is exited.

18.5.3.3.2 Exiting STOP mode

As soon as an enabled interrupt is asserted, the CCM will begin the process of exiting STOP mode.

The following will take place:

1. If vstby bit was set, deassert PMIC_STBY_REQ to notify power management IC to change voltage from standby voltage to functional voltage.
2. If well bias was enabled, deasserted well bias controls.
3. If sbyos was set, and CCM closed either external oscillator or on board oscillator, then CCM will start oscillator by asserting ref_en_b signal and deasserting cosc_pwrdown sigal respectively.
4. After the amount of CKILs defined in stby_count bits, wait until PMIC functional voltage is ready. This is the notification from power management IC that the voltage is ready at its functional value. Only then will CCM continue the steps.
5. Start osc. If oscillator was started, wait until oscnt has finished its counting to make sure that oscillator is ready.
6. Start PLLs. Only the PLLs that were configured to be on prior to the entrance to STOP mode will be started.
7. CCM will request GPC to restore ARM power by GPC_PUP_REQ. If power was removed from the ARM platform, GPC will notify CCM by asserting signal GPC_PUP_ACK that power to ARM is back on, and its safe to exit from STOP mode. Only then will the CCM progress to the next step.
8. Once assertion of notification from src that the resets for the power gated modules has been finished, (src_power_gating_reset_done is set) negate the low power request signals to all modules and enable all module clocks including ARM clocks and CKIL sync, and return to run mode. (Clocks whose CCGR bits are not to be opened in RUN mode will not be opened; they will continued to be gated.)

Once the system is in run mode, negate signals ccm_ipg_stop and system_in_stop_mode.

18.6 CCM Memory Map/Register Definition

NOTE

CCM Register reset values may not be the same in ROM. See for more information.

CCM memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20C_4000	CCM Control Register (CCM_CCR)	32	R/W	0401_16FFh	18.6.1/843
20C_4004	CCM Control Divider Register (CCM_CCDR)	32	R/W	0000_0000h	18.6.2/845
20C_4008	CCM Status Register (CCM_CSR)	32	R	0000_0010h	18.6.3/846
20C_400C	CCM Clock Switcher Register (CCM_CCSR)	32	R/W	0000_0100h	18.6.4/847
20C_4010	CCM Arm Clock Root Register (CCM_CACRR)	32	R/W	0000_0000h	18.6.5/849
20C_4014	CCM Bus Clock Divider Register (CCM_CBCDR)	32	R/W	0001_8D00h	18.6.6/850
20C_4018	CCM Bus Clock Multiplexer Register (CCM_CBCMR)	32	R/W	0002_0324h	18.6.7/852
20C_401C	CCM Serial Clock Multiplexer Register 1 (CCM_CSCMR1)	32	R/W	00F0_0000h	18.6.8/855
20C_4020	CCM Serial Clock Multiplexer Register 2 (CCM_CSCMR2)	32	R/W	02B9_2F06h	18.6.9/858
20C_4024	CCM Serial Clock Divider Register 1 (CCM_CSCDR1)	32	R/W	0049_0B00h	18.6.10/ 859
20C_4028	CCM SSI1 Clock Divider Register (CCM_CS1CDR)	32	R/W	0EC1_02C1h	18.6.11/ 862
20C_402C	CCM SSI2 Clock Divider Register (CCM_CS2CDR)	32	R/W	0007_36C1h	18.6.12/ 864
20C_4030	CCM D1 Clock Divider Register (CCM_CDCDR)	32	R/W	33F7_1F92h	18.6.13/ 866
20C_4034	CCM HSC Clock Divider Register (CCM_CHSCCDR)	32	R/W	0002_A150h	18.6.14/ 868
20C_4038	CCM Serial Clock Divider Register 2 (CCM_CSCDR2)	32	R/W	0002_A150h	18.6.15/ 870
20C_403C	CCM Serial Clock Divider Register 3 (CCM_CSCDR3)	32	R/W	0001_0841h	18.6.16/ 872
20C_4048	CCM Divider Handshake In-Process Register (CCM_CDHIPR)	32	R	0000_0000h	18.6.17/ 874
20C_4054	CCM Low Power Control Register (CCM_CLPCR)	32	R/W	0000_0079h	18.6.18/ 877
20C_4058	CCM Interrupt Status Register (CCM_CISR)	32	w1c	0000_0000h	18.6.19/ 880
20C_405C	CCM Interrupt Mask Register (CCM_CIMR)	32	R/W	FFFF_FFFFh	18.6.20/ 883
20C_4060	CCM Clock Output Source Register (CCM_CCOSR)	32	R/W	000A_0001h	18.6.21/ 886
20C_4064	CCM General Purpose Register (CCM_CGPR)	32	R/W	0000_FE62h	18.6.22/ 889

Table continues on the next page...

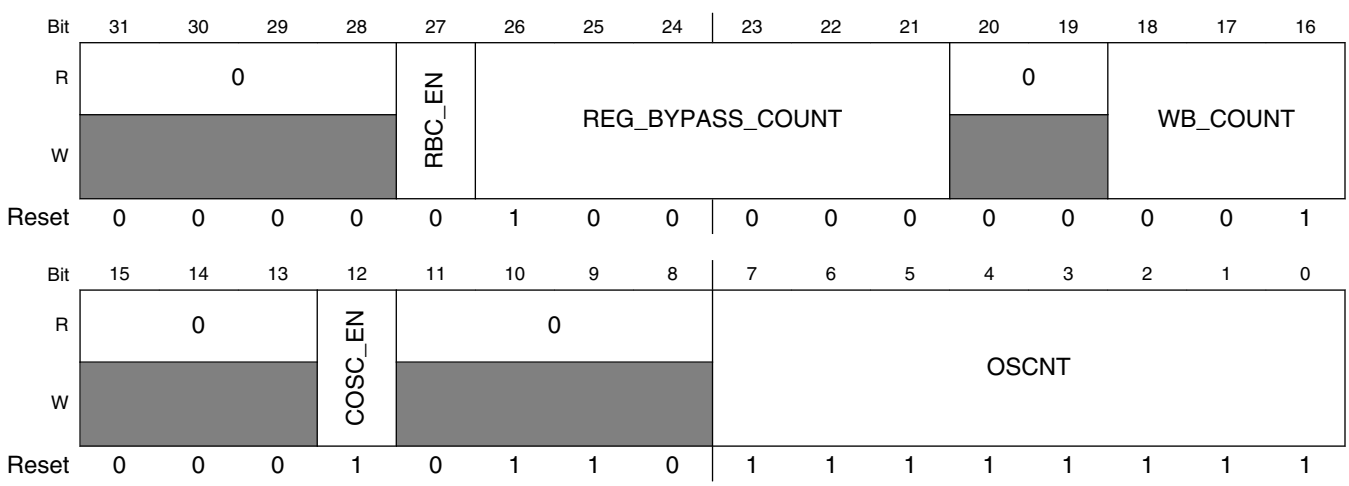
CCM memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20C_4068	CCM Clock Gating Register 0 (CCM_CCGR0)	32	R/W	FFFF_FFFFh	18.6.23/ 890
20C_406C	CCM Clock Gating Register 1 (CCM_CCGR1)	32	R/W	FFFF_FFFFh	18.6.24/ 892
20C_4070	CCM Clock Gating Register 2 (CCM_CCGR2)	32	R/W	FC3F_FFFFh	18.6.25/ 893
20C_4074	CCM Clock Gating Register 3 (CCM_CCGR3)	32	R/W	FFFF_FFFFh	18.6.26/ 895
20C_4078	CCM Clock Gating Register 4 (CCM_CCGR4)	32	R/W	FFFF_FFFFh	18.6.27/ 896
20C_407C	CCM Clock Gating Register 5 (CCM_CCGR5)	32	R/W	FFFF_FFFFh	18.6.28/ 897
20C_4080	CCM Clock Gating Register 6 (CCM_CCGR6)	32	R/W	FFFF_FFFFh	18.6.29/ 899
20C_4088	CCM Module Enable Overide Register (CCM_CMEOR)	32	R/W	FFFF_FFFFh	18.6.30/ 900

18.6.1 CCM Control Register (CCM_CCR)

The figure below represents the CCM Control Register (CCR), which contains bits to control general operation of CCM. The table below provides its field descriptions.

Address: 20C_4000h base + 0h offset = 20C_4000h



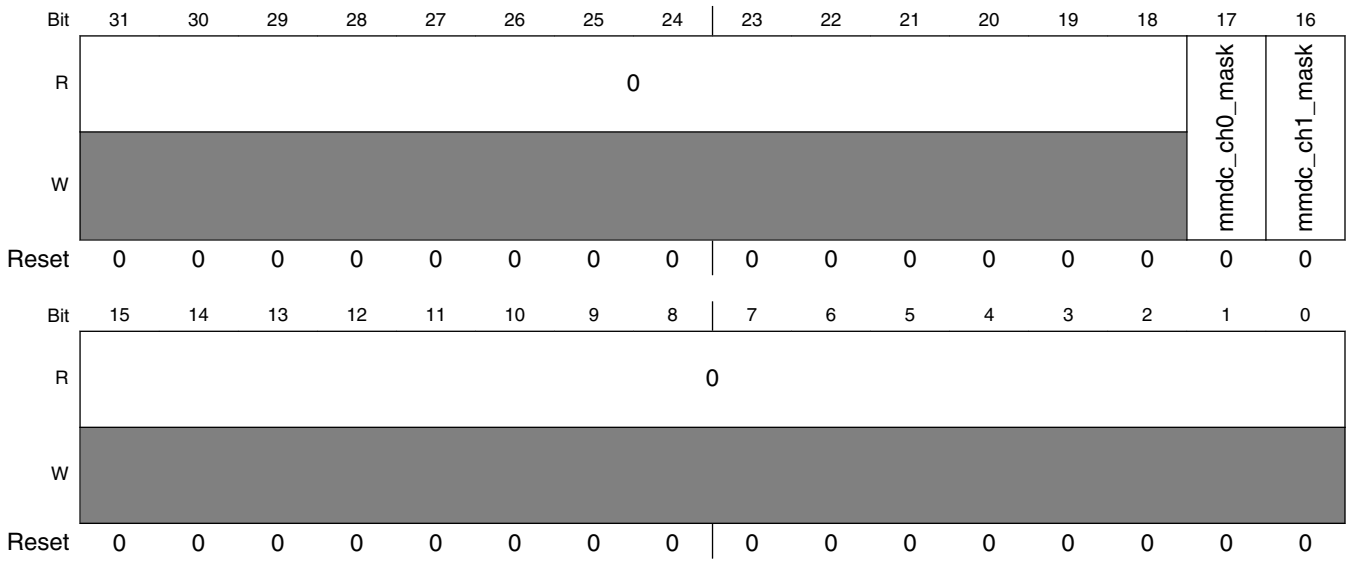
CCM_CCR field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27 RBC_EN	Enable for REG_BYPASS_COUNTER. If enabled, analog_reg_bypass signal will be asserted after REG_BYPASS_COUNT clocks of CKIL, after standby voltage is requested. If standby voltage is not requested analog_reg_bypass won't be asserted, event if counter is enabled. 1 REG_BYPASS_COUNTER enabled. 0 REG_BYPASS_COUNTER disabled
26–21 REG_BYPASS_ COUNT	Counter for analog_reg_bypass signal assertion after standby voltage request by PMIC_STBY_REQ. Should be zeroed and reconfigured after exit from low power mode. REG_BYPASS_COUNT can also be used for holding off interrupts when the PGC unit is sending signals to power gate the core. 000000 no delay 000001 1 CKIL clock period delay 111111 63 CKIL clock periods delay
20–19 Reserved	This read-only field is reserved and always has the value 0.
18–16 WB_COUNT	Well Bias counter. Delay, defined by this value, counted by CKIL clock will be applied till well bias is enabled at exit from wait or stop low power mode. Counter will be used if wb_core_at_lpm or wb_per_at_lpm bits are set. Should be zeroed and reconfigured after exit from low power mode. 000 no delay 001 1 CKIL clock delay 111 7 CKIL clocks delay
15–13 Reserved	This read-only field is reserved and always has the value 0.
12 COSC_EN	On chip oscillator enable bit - this bit value is reflected on the output cosc_en. The system will start with on chip oscillator enabled to supply source for the PLLs. Software can change this bit if a transition to the bypass PLL clocks was performed for all the PLLs. In cases that this bit is changed from '0' to '1' then CCM will enable the on chip oscillator and after counting oscnt ckil clock cycles it will notify that on chip oscillator is ready by a interrupt cosc_ready and by status bit cosc_ready. The cosc_en bit should be changed only when on chip oscillator is not chosen as the clock source. 0 disable on chip oscillator 1 enable on chip oscillator
11–8 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
OSCNT	Oscillator ready counter value. These bits define value of 32KHz counter, that serve as counter for oscillator lock time. This is used for oscillator lock time. Current estimation is ~5ms. This counter will be used in ignition sequence and in wake from stop sequence if sbyos bit was defined, to notify that on chip oscillator output is ready for the dpll_ip to use and only then the gate in dpll_ip can be opened. 00000000 count 1 ckil 11111111 count 256 ckil's

18.6.2 CCM Control Divider Register (CCM_CCDR)

The figure below represents the CCM Control Divider Register (CCDR), which contains bits that control the loading of the dividers that need handshake with the modules they affect. The table below provides its field descriptions.

Address: 20C_4000h base + 4h offset = 20C_4004h



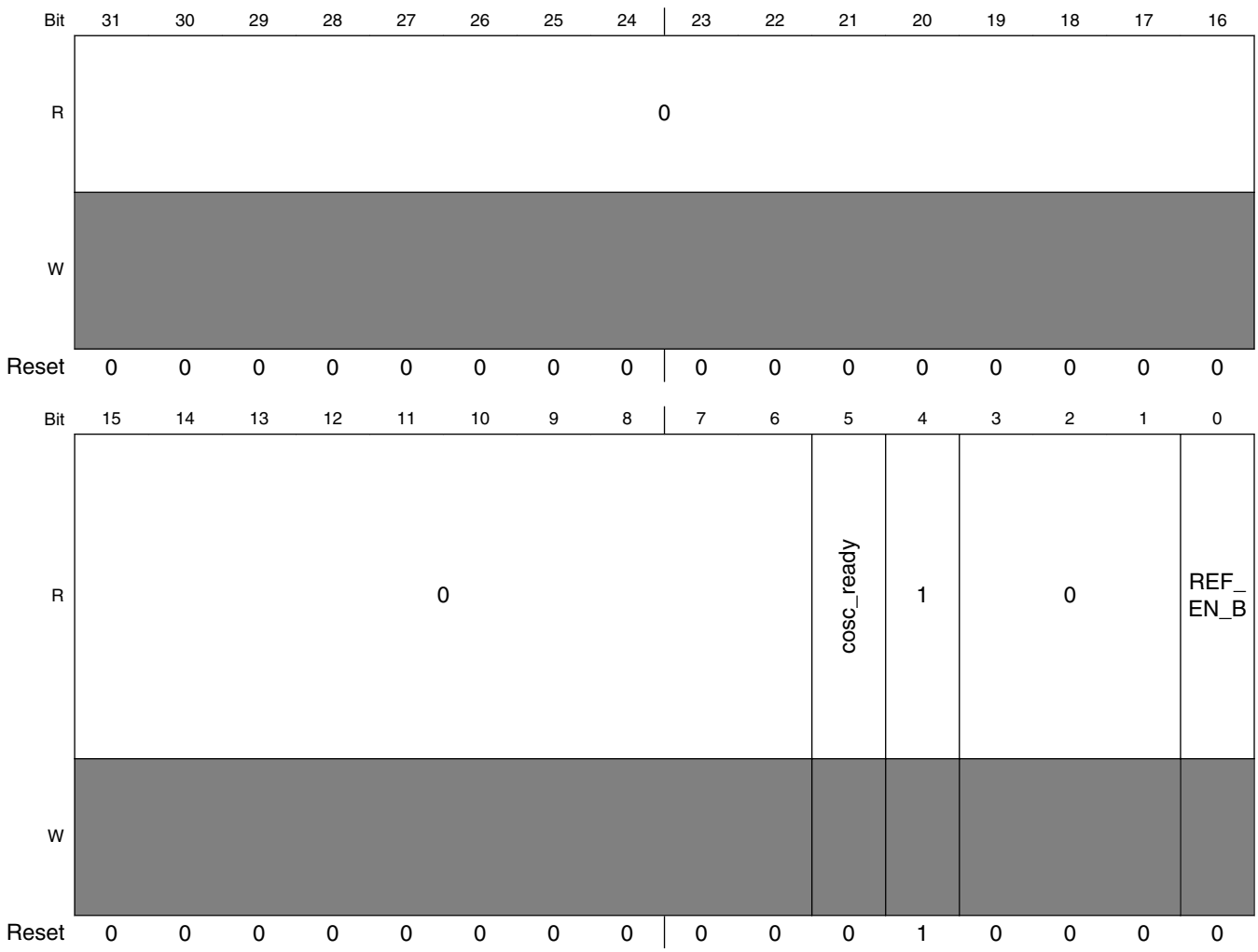
CCM_CCDR field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17 mmdc_ch0_mask	During divider ratio mmdc_ch0_axi_podf change or sync mux periph_clk_sel change (but not jtag) or SRC request during warm reset, mask handshake with mmdc_ch0 module. 0 allow handshake with mmdc_ch0 module 1 mask handshake with mmdc_ch0. Request signal will not be generated.
16 mmdc_ch1_mask	During divider ratio mmdc_ch1_axi_podf change or sync mux periph2_clk_sel change (but not jtag) or SRC request during warm reset, mask handshake with mmdc_ch1 module. 0 allow handshake with mmdc_ch1 module 1 mask handshake with mmdc_ch1. Request signal will not be generated.
Reserved	This read-only field is reserved and always has the value 0.

18.6.3 CCM Status Register (CCM_CSR)

The figure below represents the CCM status Register (CSR). The status bits are read-only bits. The table below provides its field descriptions.

Address: 20C_4000h base + 8h offset = 20C_4008h



CCM_CSR field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5 cosc_ready	Status indication of on board oscillator. This bit will be asserted if on chip oscillator is enabled and on chip oscillator is not powered down, and if oscnt counter has finished counting.

Table continues on the next page...

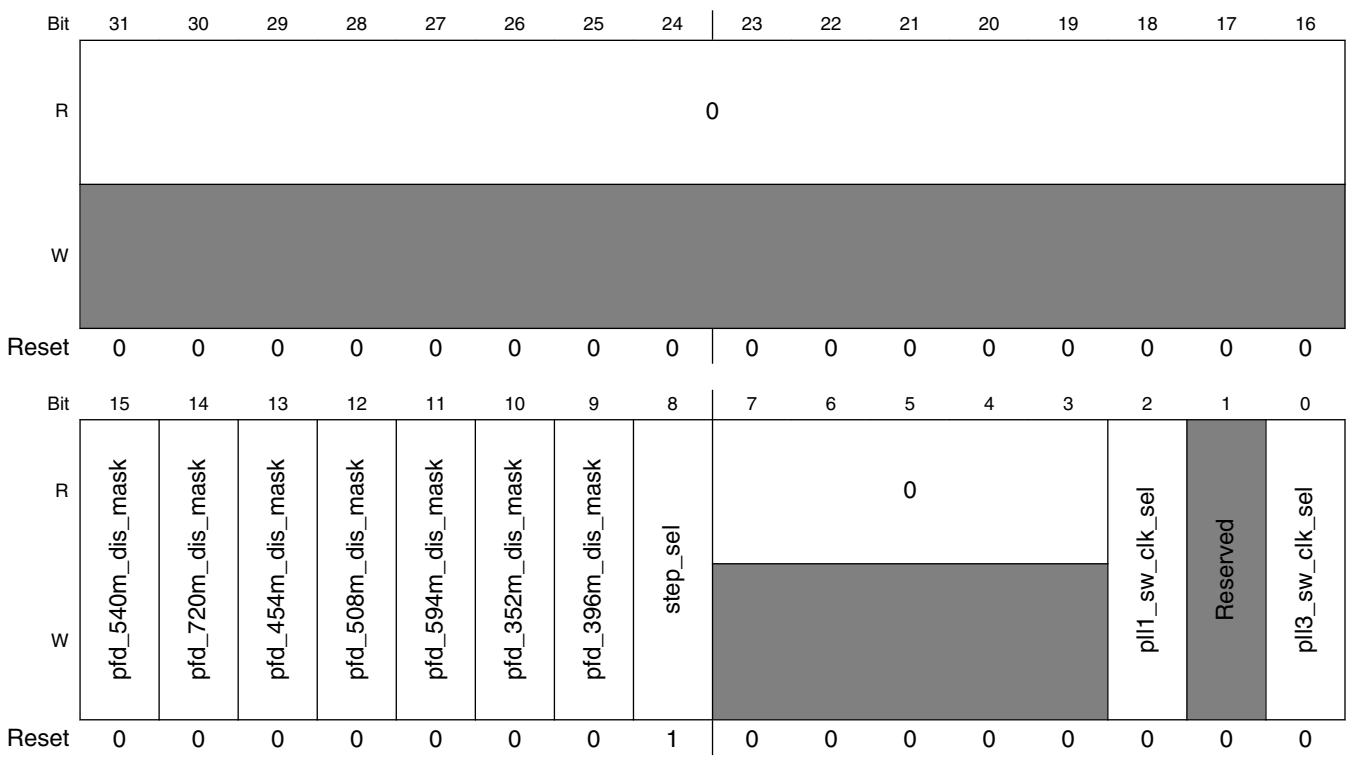
CCM_CSR field descriptions (continued)

Field	Description
	0 on board oscillator is not ready. 1 on board oscillator is ready.
4 Reserved	This read-only field is reserved and always has the value 1.
3–1 Reserved	This read-only field is reserved and always has the value 0.
0 REF_EN_B	Status of the value of CCM_REF_EN_B output of ccm 0 value of CCM_REF_EN_B is '0' 1 value of CCM_REF_EN_B is '1'

18.6.4 CCM Clock Switcher Register (CCM_CCSR)

The figure below represents the CCM Clock Switcher register (CCSR). The CCSR register contains bits to control the switcher sub-module dividers and multiplexers. The table below provides its field descriptions.

Address: 20C_4000h base + Ch offset = 20C_400Ch



CCM_CCSR field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15 pfd_540m_dis_mask	Mask of 540M PFD auto-disable. 0 - 540M PFD disable=0 (PFD always on) 1 540M PFD disable is managed by associated dividers disable. If all 540M-driven dividers are closed, PFD is disabled.
14 pfd_720m_dis_mask	Mask of 720M PFD auto-disable. 0 720M PFD disable=0 (PFD always on) 1 720M PFD disable is managed by associated dividers disable. If all 720M-driven dividers are closed, PFD is disabled.
13 pfd_454m_dis_mask	Mask of 454M PFD auto-disable. 0 454M PFD disable=0 (PFD always on) 1 454M PFD disable is managed by associated dividers disable. If all 454M-driven dividers are closed, PFD is disabled.
12 pfd_508m_dis_mask	Mask of 508M PFD auto-disable. 0 508M PFD disable=0 (PFD always on) 1 508M PFD disable is managed by associated dividers disable. If all 508M-driven dividers are closed, PFD is disabled.
11 pfd_594m_dis_mask	Mask of 594M PFD auto-disable. 0 594M PFD disable=0 (PFD always on) 1 594M PFD disable is managed by associated dividers disable. If all 594M-driven dividers are closed, PFD is disabled.
10 pfd_352m_dis_mask	Mask of 352M PFD auto-disable. 0 352M PFD disable=0 (PFD always on) 1 352M PFD disable is managed by associated dividers disable. If all 352M-driven dividers are closed, PFD is disabled.
9 pfd_396m_dis_mask	Mask of 396M PFD auto-disable. 0 396M PFD disable=0 (PFD always on) 1 396M PFD disable is managed by associated dividers disable. If all 396M-driven dividers are closed, PFD is disabled.
8 step_sel	Selects the option to be chosen for the step frequency when shifting ARM frequency. This will control the step_clk. **NOTE:** This mux is allowed to be changed only if its output is not used, i.e. ARM uses the output of pll1, and step_clk is not used. 0 osc_clk (24M) - source for lp_apm. 1 PLL2 PFD2 clock
7–3 Reserved	This read-only field is reserved and always has the value 0.
2 pll1_sw_clk_sel	Selects source to generate pll1_sw_clk. 0 pll1_main_clk 1 step_clk

Table continues on the next page...

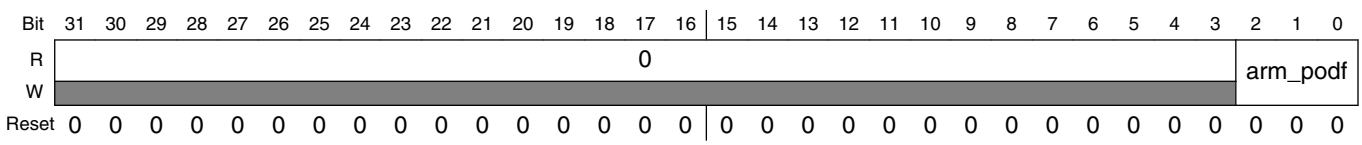
CCM_CCSR field descriptions (continued)

Field	Description
1 -	This field is reserved. Reserved
0 pll3_sw_clk_sel	Selects source to generate pll3_sw_clk. This bit should only be used for testing purposes. 0 pll3_main_clk 1 pll3 bypass clock

18.6.5 CCM Arm Clock Root Register (CCM_CACRR)

The figure below represents the CCM Arm Clock Root register (CACRR). The CACRR register contains bits to control the ARM clock root generation. The table below provides its field descriptions.

Address: 20C_4000h base + 10h offset = 20C_4010h



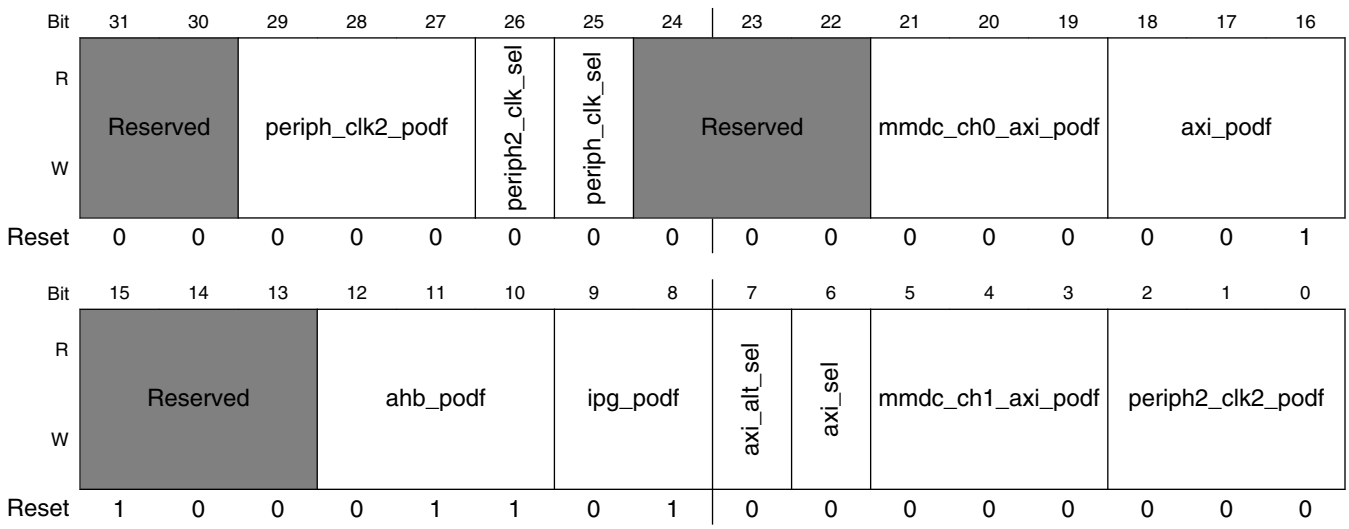
CCM_CACRR field descriptions

Field	Description
31–3 Reserved	This read-only field is reserved and always has the value 0.
arm_podf	Divider for ARM clock root. **NOTE:** If arm_freq_shift_divider is set to '1' then any new write to arm_podf will be held until arm_clk_switch_req signal is asserted. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8

18.6.6 CCM Bus Clock Divider Register (CCM_CBCDR)

The figure below represents the CCM Bus Clock Divider Register (CBCDR). The CBCDR register contains bits to control the clock generation sub module dividers. The table below provides its field descriptions.

Address: 20C_4000h base + 14h offset = 20C_4014h



CCM_CBCDR field descriptions

Field	Description
31–30 -	This field is reserved. Reserved
29–27 periph_clk2_podf	Divider for periph2 clock podf. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
26 periph2_clk_sel	Selector for peripheral2 main clock (source of mmdc_ch1_clk_root). **NOTE:** Any change of this mux select will involve handshake with the MMDC. Refer to the CCDR and CDHIPR registers for the handshake bypass and busy bits. 0 PLL2 (pll2_main_clk) 1 derive clock from periph_clk2_clk clock source.

Table continues on the next page...

CCM_CBCDR field descriptions (continued)

Field	Description
25 periph_clk_sel	Selector for peripheral main clock (source of MMDC_CH0_CLK_ROOT). **NOTE:** Alternative clock source should be used when PLL is relocked. For PLL relock procedure pls refer to the PLL chapter. **NOTE:** Any change of this sync mux select will involve handshake with the MMDC. Refer to the CCDR and CDHIPR registers for the handshake bypass and busy bits. 0 PLL2 (pll2_main_clk) 1 derive clock from periph_clk2_clk clock source.
24–22 -	This field is reserved. Reserved
21–19 mmdc_ch0_axi_podf	Divider for mmdc_ch0_axi podf. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
18–16 axi_podf	Divider for axi podf. **NOTE:** Any change of this divider might involve handshake with EMI and IPU. See CDHIPR register for the handshake busy bits. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
15–13 -	This field is reserved. Reserved
12–10 ahb_podf	Divider for AHB PODF. **NOTE:** Any change of this divider might involve handshake with EMI and IPU. See CDHIPR register for the handshake busy bits. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8

Table continues on the next page...

CCM_CBCDR field descriptions (continued)

Field	Description
9–8 ipg_podf	Divider for ipg podf. **NOTE:** SDMA module will not support ratio of 1:3 and 1:4 for ahb_clk:ipg_clk. In case SDMA is used, then those ratios should not be used. 00 divide by 1 01 divide by 2 10 divide by 3 11 divide by 4
7 axi_alt_sel	AXI alternative clock select 0 pll2 396MHz PFD will be selected as alternative clock for AXI root clock 1 pll3 540MHz PFD will be selected as alternative clock for AXI root clock
6 axi_sel	AXI clock source select 0 Periph_clk output will be used as AXI clock root 1 AXI alternative clock will be used as AXI clock root
5–3 mmdc_ch1_axi_podf	Divider for mmdc_ch1_axi podf. **NOTE:** This design implementation does not use MMDC_CH1_CLK_ROOT as a clock source to the MMDC. Only MMDC_CH0_CLK_ROOT is used. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
periph2_clk2_podf	Divider for periph2_clk2 podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8

18.6.7 CCM Bus Clock Multiplexer Register (CCM_CBCMR)

The figure below represents the CCM Bus Clock Multiplexer Register (CBCMR). The CBCMR register contains bits to control the multiplexers that generate the bus clocks. The table below provides its field descriptions.

NOTE

Any change on the above multiplexer will have to be done while the module that its clock is affected is not functional and the respective clock is gated in LPCG. If the change will be done during operation of the module, then it is not guaranteed that the modules operation will not be harmed.

The change for arm_clk_sel should be done through sdma so that ARM will not use this clock during the change and the clock will be gated in LPCG.

Address: 20C_4000h base + 18h offset = 20C_4018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	gpu3d_shader_podf			gpu3d_core_podf			gpu2d_core_clk_podf		pre_periph2_clk_sel			periph2_clk2_sel	pre_periph_clk_sel		gpu2d_core_clk_sel	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	vpu_axi_clk_sel		periph_clk2_sel		vdoaxi_clk_sel	pcie_axi_clk_sel	gpu3d_shader_clk_sel		Reserved		gpu3d_core_clk_sel		Reserved		gpu3d_axi_clk_sel	gpu2d_axi_clk_sel
Reset	0	0	0	0	0	0	1	1	0	0	1	0	0	1	0	0

CCM_CBCMR field descriptions

Field	Description
31–29 gpu3d_shader_podf	Divider for gpu3d_shader clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
28–26 gpu3d_core_podf	Divider for gpu3d_core clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2

Table continues on the next page...

CCM_CBCMR field descriptions (continued)

Field	Description
	010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
25–23 gpu2d_core_clk_podf	Divider for gpu2d_core clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
22–21 pre_periph2_clk_sel	Selector for pre_periph2 clock multiplexer 00 derive clock from PLL2 main 528MHz clock 01 derive clock from 396MHz PLL2 PFD 10 derive clock from 352M PFD 11 derive clock from 198MHz clock (divided 396MHz PLL2 PFD)
20 periph2_clk2_sel	Selector for periph2_clk2 clock multiplexer 0 derive clock from pll3_sw_clk 1 derive clock from PLL2 Main
19–18 pre_periph_clk_sel	Selector for pre_periph clock multiplexer 00 derive clock from PLL2 main 528MHz clock 01 derive clock from 396MHz PLL2 PFD 10 derive clock from 352M PFD 11 derive clock from 198MHz clock (divided 396MHz PLL2 PFD)
17–16 gpu2d_core_clk_sel	Selector for open vg (GPU2D Core) clock multiplexer 00 derive clock from axi 01 derive clock from pll3_sw_clk 10 352M PFD 11 derive clock from 396M PFD
15–14 vpu_axi_clk_sel	Selector for VPU axi clock multiplexer 00 derive clock from AXI 01 derive clock from 396M PFD 10 derive clock from 352M PFD 11 Reserved
13–12 periph_clk2_sel	Selector for peripheral clk2 clock multiplexer 00 derive clock from pll3_sw_clk

Table continues on the next page...

CCM_CBCMR field descriptions (continued)

Field	Description
	01 derive clock from OSC_CLK (pll1_ref_clk) 10 derive clock from PLL2 (pll2_main_clk) 11 reserved
11 vdoaxi_clk_sel	Selector for vdoaxi clock multiplexer 0 derive clock from axi clk 1 derive clock from 132M clock
10 pcie_axi_clk_sel	Selector for pcie_axi clock multiplexer 0 derive clock from axi clk 1 derive clock from system_133M clk
9–8 gpu3d_shader_clk_sel	Selector for gpu3d_shader clock multiplexer 00 derive clock from mmdc_ch0 clk 01 derive clock from pll3_sw_clk 10 derive clock from PFD 594M 11 derive clock from 720M PFD
7–6 -	This field is reserved. Reserved
5–4 gpu3d_core_clk_sel	Selector for gpu3d_core clock multiplexer 00 derive clock from mmdc_ch0 01 derive clock from pll3_sw_clk 10 derive clock from 594M PFD 11 derive clock from 396M PFD
3–2 -	This field is reserved. Reserved
1 gpu3d_axi_clk_sel	Selector for gpu3d_axi clock multiplexer 0 derive clock from axi 1 derive clock from system_133M_clk
0 gpu2d_axi_clk_sel	Selector for gpu2d_axi clock multiplexer 0 derive clock from axi 1 derive clock from system_133M_clk

18.6.8 CCM Serial Clock Multiplexer Register 1 (CCM_CSCMR1)

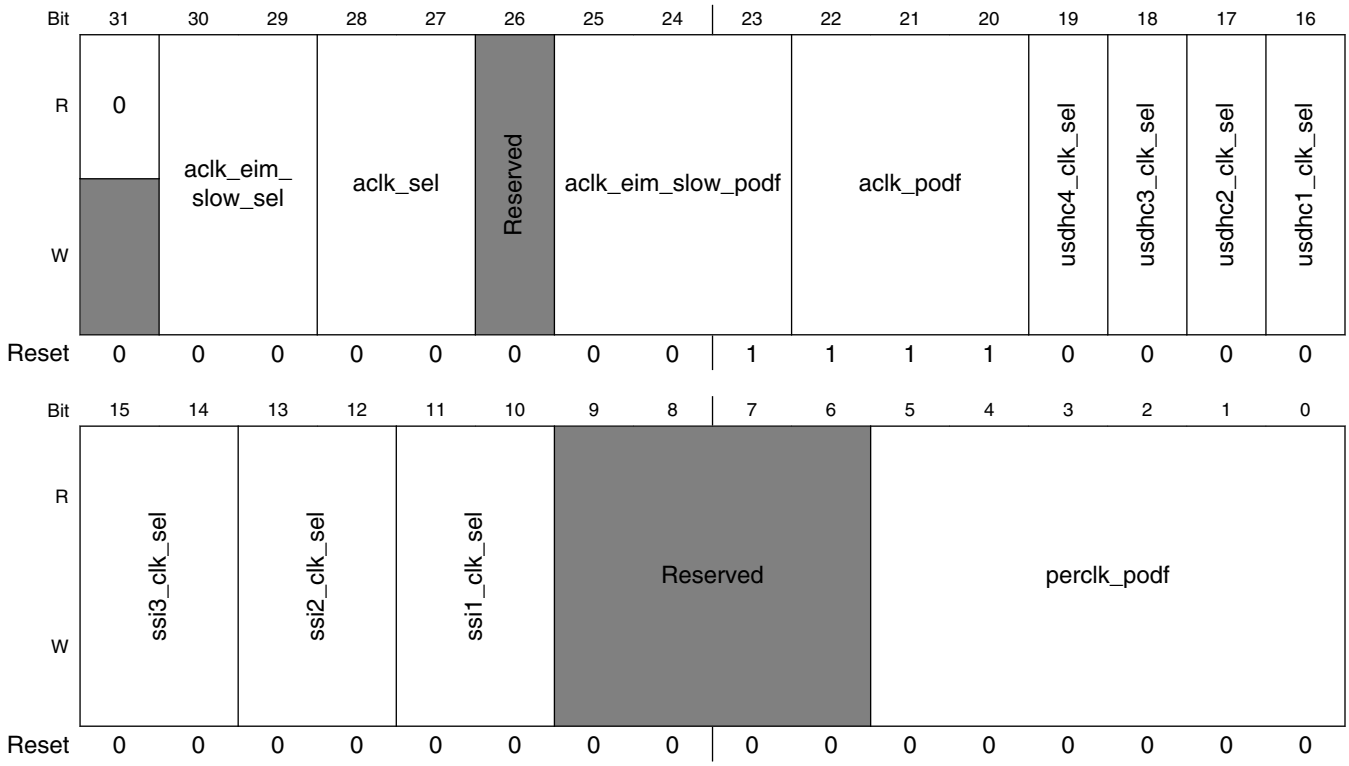
The figure below represents the CCM Serial Clock Multiplexer Register 1 (CSCMR1). The CSCMR1 register contains bits to control the multiplexers that generate the serial clocks. The table below provides its field descriptions.

NOTE

Any change on the above multiplexer will have to be done while the module that its clock is affected is not functional and

the clock is gated. If the change will be done during operation of the module, then it is not guaranteed that the modules operation will not be harmed.

Address: 20C_4000h base + 1Ch offset = 20C_401Ch



CCM_CSCMR1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 aclk_eim_slow_sel	Selector for aclk_eim_slow root clock multiplexer 00 derive clock from AXI clk root 01 derive clock from pll3_sw_clk 10 derive clock from 396M PFD 11 derive clock from 352M PFD
28–27 aclk_sel	Selector for aclk root clock multiplexer 00 derive clock from 396M PFD 01 derive clock from pll3_sw_clk 10 derive clock from AXI clk root 11 derive clock from 352M PFD
26 -	This field is reserved. Reserved
25–23 aclk_eim_slow_podf	Divider for aclk_eim_slow clock root. 000 divide by 1

Table continues on the next page...

CCM_CSCMR1 field descriptions (continued)

Field	Description
	001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
22–20 aclk_podf	Divider for aclk clock root. **NOTE:** These bits are inverted between R/W and are not sequential. 000 divide by 7 (Read value 110) 001 divide by 8 (Read value 111) 010 divide by 5 (Read value 100) 011 divide by 6 (Read value 101) 100 divide by 3 (Read value 010) 101 divide by 4 (Read value 011) 110 divide by 1 (Read value 000) 111 divide by 2 (Read value 001)
19 usdhc4_clk_sel	Selector for usdhc4 clock multiplexer 0 derive clock from 396M PFD 1 derive clock from 352M PFD
18 usdhc3_clk_sel	Selector for usdhc3 clock multiplexer 0 derive clock from 396M PFD 1 derive clock from 352M PFD
17 usdhc2_clk_sel	Selector for usdhc2 clock multiplexer 0 derive clock from 396M PFD 1 derive clock from 352M PFD
16 usdhc1_clk_sel	Selector for usdhc1 clock multiplexer 0 derive clock from 396M PFD 1 derive clock from 352M PFD
15–14 ssi3_clk_sel	Selector for ssi3 clock multiplexer 00 derive clock from 508.2M PFD 01 derive clock from 454.7M PFD 10 derive clock from pll4 11 Reserved
13–12 ssi2_clk_sel	Selector for ssi2 clock multiplexer 00 derive clock from 508.2M PFD 01 derive clock from 454.7M PFD 10 derive clock from pll4 11 Reserved
11–10 ssi1_clk_sel	Selector for ssi1 clock multiplexer

Table continues on the next page...

CCM_CSCMR1 field descriptions (continued)

Field	Description
	00 derive clock from 508.2M PFD 01 derive clock from 454.7M PFD 10 derive clock from pll4 11 Reserved
9–6 -	This field is reserved. Reserved
perclk_podf	Divider for perclk podf. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8

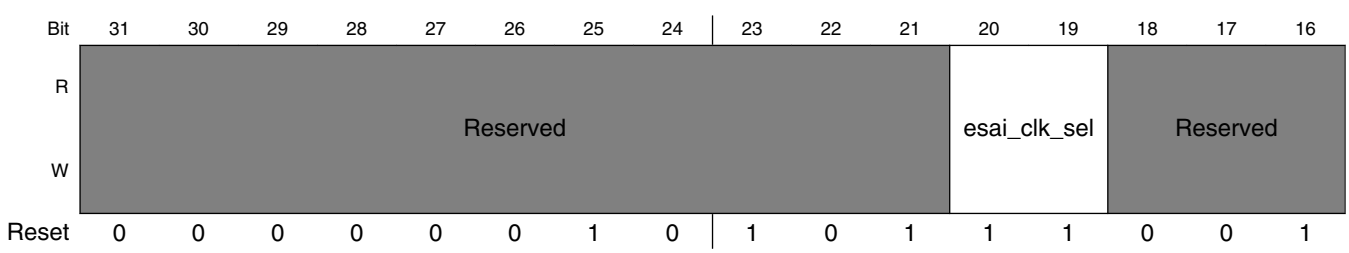
18.6.9 CCM Serial Clock Multiplexer Register 2 (CCM_CSCMR2)

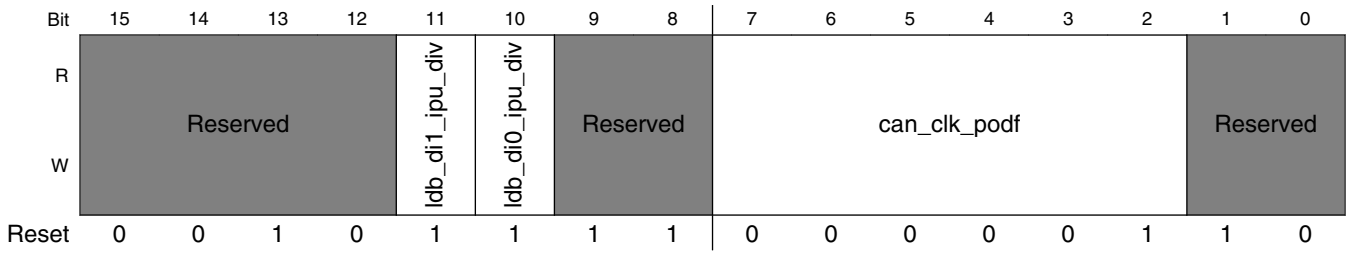
The figure below represents the CCM Serial Clock Multiplexer Register 2 (CSCMR2). The CSCMR2 register contains bits to control the multiplexers that generate the serial clocks. The table below provides its field descriptions.

NOTE

Any change on the above multiplexer will have to be done while the module that its clock is affected is not functional and the clock is gated. If the change will be done during operation of the module, then it is not guaranteed that the modules operation will not be harmed.

Address: 20C_4000h base + 20h offset = 20C_4020h





CCM_CSCMR2 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved
20–19 esai_clk_sel	Selector for esai clock multiplexer 00 derive clock from pll4 divided clock 01 derive clock from 508M PFD clock 10 derive clock from 454M PFD clock 11 derive clock from pll3_sw_clk
18–12 -	This field is reserved. Reserved
11 ldb_di1_ipu_div	Control for divider of ldb clock for IPU di1 0 divide by 3.5 1 divide by 7
10 ldb_di0_ipu_div	Control for divider of ldb clock for IPU di0 0 divide by 3.5 1 divide by 7
9–8 -	This field is reserved. Reserved
7–2 can_clk_podf	Divider for can clock podf. 000000 divide by 1 000111 divide by 8 111111 divide by 2^6
-	This field is reserved. Reserved

18.6.10 CCM Serial Clock Divider Register 1 (CCM_CSCDR1)

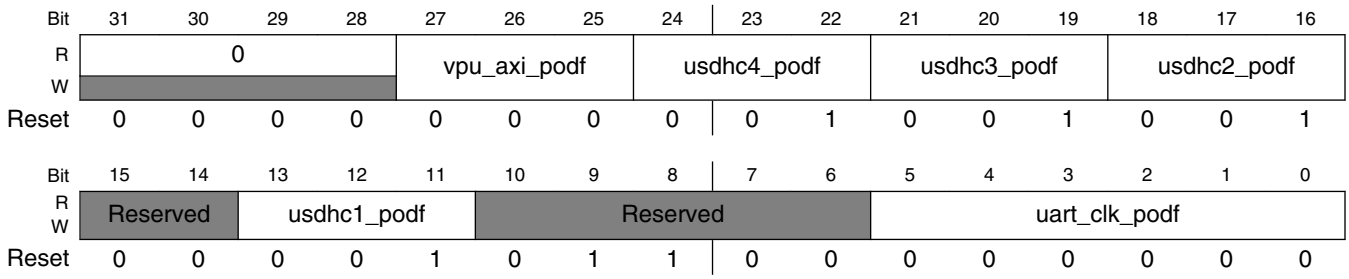
The figure below represents the CCM Serial Clock Divider Register 1 (CSCDR1). The CSCDR1 register contains bits to control the clock generation sub-module dividers. The table below provides its field descriptions.

NOTE

Any change on the above dividers will have to be done while the module that its clock is affected is not functional and the

affected clock is gated. If the change will be done during operation of the module, then it is not guaranteed that the modules operation will not be harmed.

Address: 20C_4000h base + 24h offset = 20C_4024h



CCM_CSCDR1 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–25 vpu_axi_podf	Divider for vpu axi clock podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
24–22 usdhc4_podf	Divider for esdhc4 clock pred. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
21–19 usdhc3_podf	Divider for usdhc3 clock podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4

Table continues on the next page...

CCM_CSCDR1 field descriptions (continued)

Field	Description
	100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
18–16 usdhc2_podf	Divider for usdhc2 clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
15–14 -	This field is reserved. Reserved
13–11 usdhc1_podf	Divider for usdhc1 clock podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
10–6 -	This field is reserved. Reserved.
uart_clk_podf	Divider for uart clock podf. 000000 divide by 1 111111 divide by 2^6

18.6.11 CCM SSI1 Clock Divider Register (CCM_CS1CDR)

The figure below represents the CCM SSI1, SSI3, ESAI Clock Divider Register (CS1CDR). The CS1CDR register contains bits to control the ssi1 clock generation dividers. The table below provides its field descriptions.

Address: 20C_4000h base + 28h offset = 20C_4028h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0				esai_clk_podf			ssi3_clk_pred			ssi3_clk_podf					
W																
Reset	0	0	0	0	1	1	1	0	1	1	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				esai_clk_pred			ssi1_clk_pred			ssi1_clk_podf					
W																
Reset	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	1

CCM_CS1CDR field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–25 esai_clk_podf	Divider for esai clock podf. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
24–22 ssi3_clk_pred	Divider for ssi3 clock pred. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
21–16 ssi3_clk_podf	Divider for ssi3 clock podf. The input clock to this divider should be lower than 300Mhz, the predivider can be used to achieve this.

Table continues on the next page...

CCM_CS1CDR field descriptions (continued)

Field	Description
	000000 divide by 1 111111 divide by 2^6
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 esai_clk_pred	Divider for esai clock pred. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
8–6 ssi1_clk_pred	Divider for ssi1 clock pred. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
ssi1_clk_podf	Divider for ssi1 clock podf. The input clock to this divider should be lower than 300Mhz, the predivider can be used to achieve this. 000000 divide by 1 111111 divide by 2^6

18.6.12 CCM SSI2 Clock Divider Register (CCM_CS2CDR)

The figure below represents the CCM SSI2, LDB Clock Divider Register (CS2CDR). The CS2CDR register contains bits to control the ssi2 clock generation dividers, and ldb serial clocks select. The table below provides its field descriptions.

Address: 20C_4000h base + 2Ch offset = 20C_402Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0					enfc_clk_podf						enfc_clk_pred			enfc_clk_sel	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	ldb_di1_clk_sel			ldb_di0_clk_sel			ssi2_clk_pred			ssi2_clk_podf					
W																
Reset	0	0	0	1	1	0	1	1	0	1	1	0	0	0	0	0	1

CCM_CS2CDR field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–21 enfc_clk_podf	Divider for enfc clock divider. 000000 divide by 1 000001 divide by 2 111111 divide by 2^6
20–18 enfc_clk_pred	Divider for enfc clock pred divider. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
17–16 enfc_clk_sel	Selector for enfc clock multiplexer **NOTE:** Multiplexor should be updated when output clock is gated. 00 pll2 352M PFD 01 pll2 clock

Table continues on the next page...

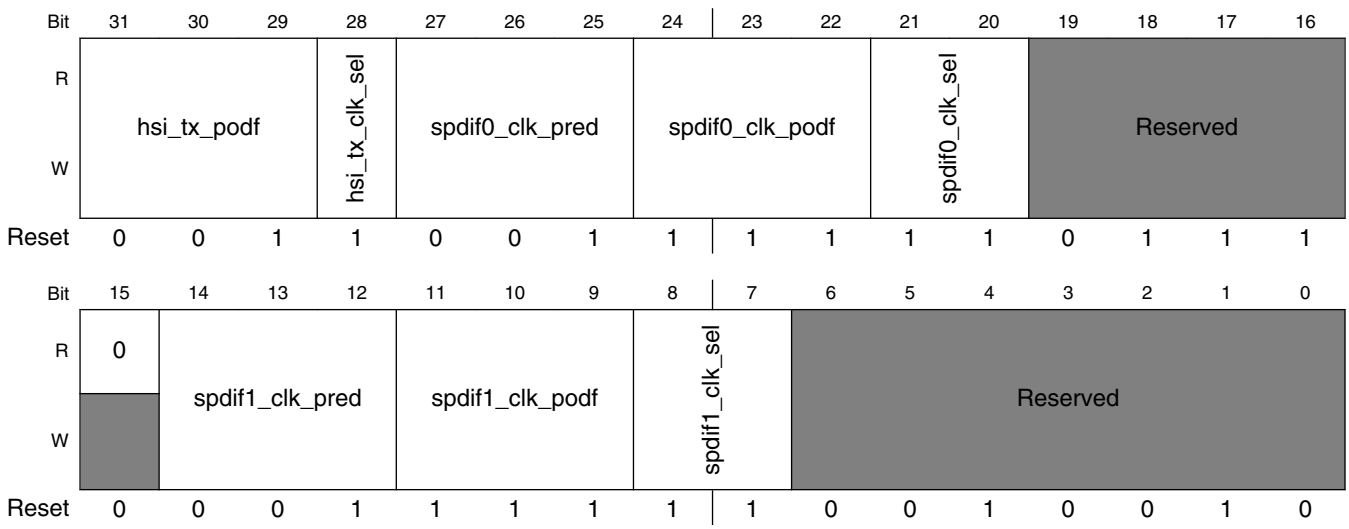
CCM_CS2CDR field descriptions (continued)

Field	Description
	10 pll3_sw_clk 11 pll2 396M PFD
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 ldb_di1_clk_sel	Selector for ldb_di1 clock multiplexer **NOTE:** Multiplexor should be updated when both input and output clocks are gated. 000 pll5 clock 001 pll2 352M PFD 010 pll2 396M PFD 011 MMDC_CH1 clock 100 pll3_sw_clk 101-111 Reserved
11–9 ldb_di0_clk_sel	Selector for ldb_di1 clock multiplexer **NOTE:** Multiplexor should be updated when both input and output clocks are gated. 000 pll5 clock 001 pll2 352M PFD 010 pll2 396M PFD 011 MMDC_CH1 clock 100 pll3_sw_clk 101-111 Reserved
8–6 ssi2_clk_pred	Divider for ssi2 clock pred. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
ssi2_clk_podf	Divider for ssi2 clock podf. The input clock to this divider should be lower than 300Mhz, the predivider can be used to achieve this. **NOTE:** Divider should be updated when output clock is gated. 000000 divide by 1 111111 divide by 2^6

18.6.13 CCM D1 Clock Divider Register (CCM_CDCDR)

The figure below represents the CCM DI Clock Divider Register (CDCDR). The table below provides its field descriptions.

Address: 20C_4000h base + 30h offset = 20C_4030h



CCM_CDCDR field descriptions

Field	Description
31–29 hsi_tx_podf	Divider for hsi_tx clock podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
28 hsi_tx_clk_sel	Selector for hsi_tx clock multiplexer 0 derive from pll3 120M clock 1 derive from pll2 396M PFD
27–25 spdif0_clk_pred	Divider for spdif0 clock pred. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 (do not use with high input frequencies) 001 divide by 2

Table continues on the next page...

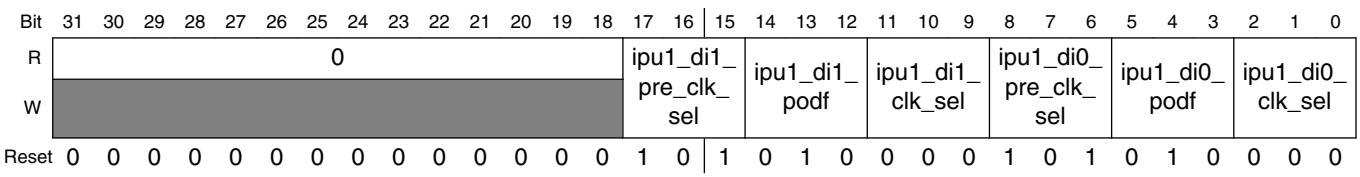
CCM_CDCDR field descriptions (continued)

Field	Description
	010 divide by 3 111 divide by 8
24–22 spdif0_clk_podf	Divider for spdif0 clock podf. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 111 divide by 8
21–20 spdif0_clk_sel	Selector for spdif0 clock multiplexer 00 derive clock from pll4 divided clock 01 derive clock from 508M PFD clock 10 derive clock from 454M PFD clock 11 derive clock from pll3_sw_clk
19–16 -	This field is reserved. Reserved
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 spdif1_clk_pred	Divider for spdif1 clock pred. **NOTE:** Divider should be updated when output clock is gated. **NOTE:** Used for ASRC clock, not related to SPDIF module 000 divide by 1 (do not use with high input frequencies) 001 divide by 2 010 divide by 3 111 divide by 8
11–9 spdif1_clk_podf	Divider for spdif1 clock podf. **NOTE:** Divider should be updated when output clock is gated. **NOTE:** Used for ASRC clock, not related to SPDIF module 000 divide by 1 111 divide by 8
8–7 spdif1_clk_sel	Selector for spdif1 clock multiplexer **NOTE:** Used for ASRC clock, not related to SPDIF module 00 derive clock from pll4 divided clock 01 derive clock from 508M PFD clock 10 derive clock from 454M PFD clock 11 derive clock from pll3_sw_clk
-	This field is reserved. Reserved

18.6.14 CCM HSC Clock Divider Register (CCM_CHSCCDR)

The figure below represents the CCM HSC Clock Divider Register (CHSCCDR). The CHSCCDR register contains bits to control the ipu di clock generation dividers. The table below provides its field descriptions.

Address: 20C_4000h base + 34h offset = 20C_4034h



CCM_CHSCCDR field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17–15 ipu1_di1_pre_ clk_sel	Selector for ipu1 di1 root clock pre-multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from mmdc_ch0 clock 001 derive clock from pll3_sw_clk 010 derive clock from pll5 011 derive clock from 352M PFD 100 derive clock from 396M PFD 101 derive clock from 540M PFD 110-111 Reserved
14–12 ipu1_di1_podf	Divider for ipu1_di clock divider. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
11–9 ipu1_di1_clk_sel	Selector for ipu1 di1 root clock multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits.

Table continues on the next page...

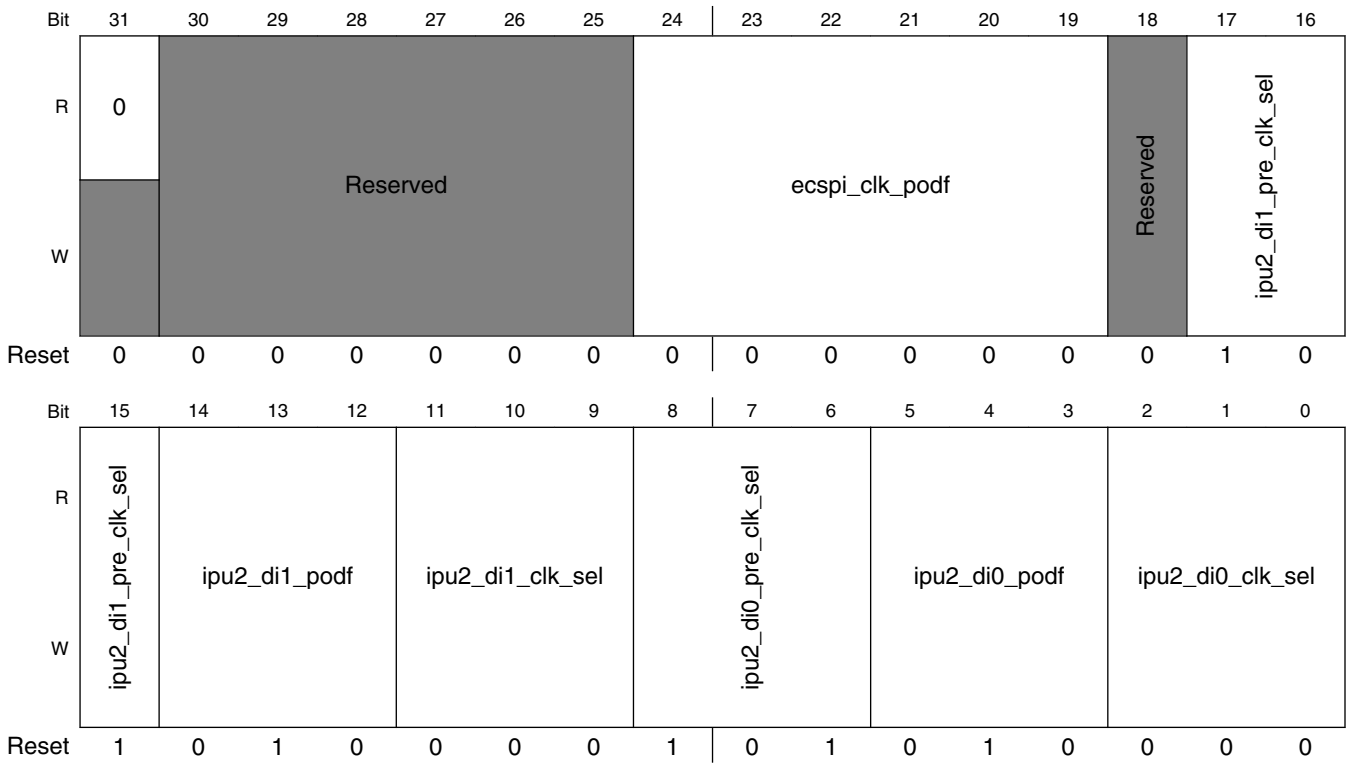
CCM_CHSCCDR field descriptions (continued)

Field	Description
	000 derive clock from divided pre-muxed ipu1 di1 clock 001 derive clock from ipp_di0_clk 010 derive clock from ipp_di1_clk 011 derive clock from ldb_di0_clk 100 derive clock from ldb_di1_clk 101-111 Reserved
8–6 ipu1_di0_pre_clk_sel	Selector for ipu1 di0 root clock pre-multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from mmdc_ch0 clock 001 derive clock from pll3_sw_clk 010 derive clock from pll5 011 derive clock from 352M PFD 100 derive clock from 396M PFD 101 derive clock from 540M PFD 110-111 Reserved
5–3 ipu1_di0_podf	Divider for ipu1_di0 clock divider. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
ipu1_di0_clk_sel	Selector for ipu1 di0 root clock multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from divided pre-muxed ipu1 di0 clock 001 derive clock from ipp_di0_clk 010 derive clock from ipp_di1_clk 011 derive clock from ldb_di0_clk 100 derive clock from ldb_di1_clk 101-111 Reserved

18.6.15 CCM Serial Clock Divider Register 2 (CCM_CSCDR2)

The figure below represents the CCM Serial Clock Divider Register 2(CSCDR2). The CSCDR2 register contains bits to control the clock generation sub-module dividers. The table below provides its field descriptions.

Address: 20C_4000h base + 38h offset = 20C_4038h



CCM_CSCDR2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–25 -	This field is reserved. Reserved
24–19 ecspi_clk_podf	Divider for ecspi clock podf. **NOTE:** Divider should be updated when output clock is gated. **NOTE:** The input clock to this divider should be lower than 300Mhz, the predivider can be used to achieve this. 000000 divide by 1 111111 divide by 2^6

Table continues on the next page...

CCM_CSCDR2 field descriptions (continued)

Field	Description
18 - Reserved	This field is reserved.
17–15 ipu2_di1_pre_clk_sel	Selector for ipu2 di1 root clock pre-multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from mmdc_ch0 clock 001 derive clock from pll3_sw_clk 010 derive clock from pll5 011 derive clock from 352M PFD 100 derive clock from 396M PFD 101 derive clock from 540M PFD 110-111 Reserved
14–12 ipu2_di1_podf	Divider for ipu2_di1 clock divider. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
11–9 ipu2_di1_clk_sel	Selector for ipu1 di2 root clock multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from divided pre-muxed ipu1 di1 clock 001 derive clock from ipp_di0_clk 010 derive clock from ipp_di1_clk 011 derive clock from ldb_di0_clk 100 derive clock from ldb_di1_clk 101-111 Reserved
8–6 ipu2_di0_pre_clk_sel	Selector for ipu2 di0 root clock pre-multiplexer **NOTE:** Multiplexor should only be updated when the output clock is gated. Please refer to System Clocks for the respective output clock gating bits. 000 derive clock from mmdc_ch0 clock 001 derive clock from pll3_sw_clk 010 derive clock from pll5 011 derive clock from 352M PFD 100 derive clock from 396M PFD

Table continues on the next page...

CCM_CSCDR2 field descriptions (continued)

Field	Description
	101 derive clock from 540M PFD 110-111 Reserved
5–3 ipu2_di0_podf	Divider for ipu2_di0 clock divider. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
ipu2_di0_clk_sel	Selector for ipu2 di0 root clock multiplexer **NOTE:** Multiplexor should be updated only when the output clock is gated. 000 derive clock from divided pre-muxed ipu1 di0 clock 001 derive clock from ipp_di0_clk 010 derive clock from ipp_di1_clk 011 derive clock from ldb_di0_clk 100 derive clock from ldb_di1_clk 101-111 Reserved

18.6.16 CCM Serial Clock Divider Register 3 (CCM_CSCDR3)

The figure below represents the CCM Serial Clock Divider Register 3(CSCDR3). The CSCDR3 register contains bits to control the clock generation sub-module dividers. The table below provides its field descriptions.

Address: 20C_4000h base + 3Ch offset = 20C_403Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0													ipu2_hsp_podf		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	ipu2_hsp_ clk_sel		ipu1_hsp_podf			ipu1_hsp_ clk_sel		Reserved								
W																
Reset	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1

CCM_CSCDR3 field descriptions

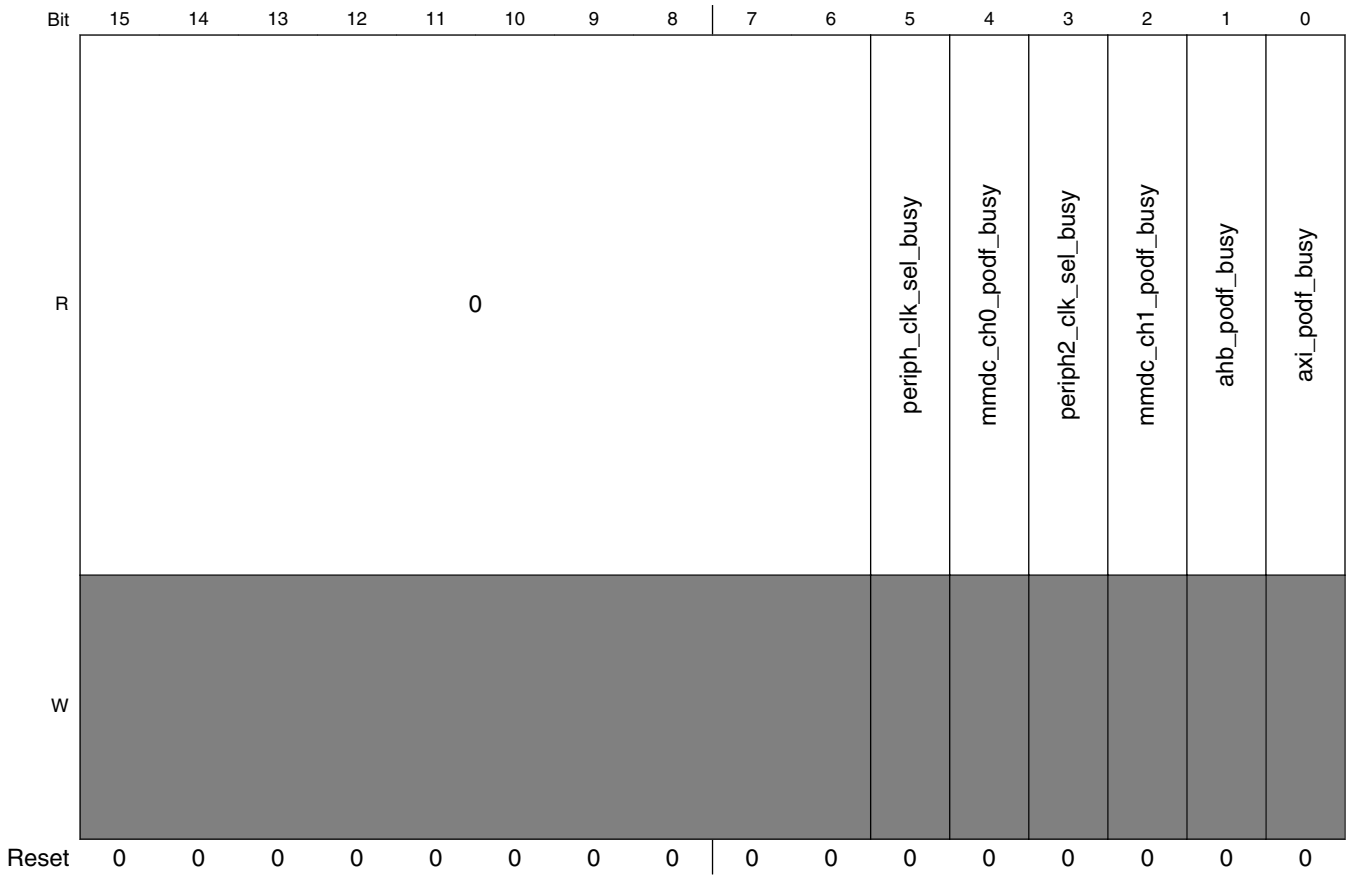
Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0.
18–16 ipu2_hsp_podf	Divider for ipu2_hsp clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
15–14 ipu2_hsp_clk_sel	Selector for ipu2_hsp clock multiplexer 00 derive clock from mmdc_ch0 clock 01 derive clock from 396M PFD 10 derive clock from 120M 11 derive clock from 540M PFD
13–11 ipu1_hsp_podf	Divider for ipu1_hsp clock. **NOTE:** Divider should be updated when output clock is gated. 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
10–9 ipu1_hsp_clk_sel	Selector for ipu1_hsp clock multiplexer 00 derive clock from mmdc_ch0 clock 01 derive clock from 396M PFD 10 derive clock from 120M 11 derive clock from 540M PFD
-	This field is reserved. Reserved

18.6.17 CCM Divider Handshake In-Process Register (CCM_CDHIPR)

The figure below represents the CCM Divider Handshake In-Process Register (CDHIPR). The CDHIPR register contains read-only bits that indicate that CCM is in the process of updating dividers or muxes that might need handshake with modules.

Address: 20C_4000h base + 48h offset = 20C_4048h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															arm_podf_busy
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



CCM_CDHIPR field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 arm_podf_busy	Busy indicator for arm_podf. 0 divider is not busy and its value represents the actual division. 1 divider is busy with handshake process with module. The value read in the divider represents the previous value of the division factor, and after the handshake the written value of the arm_podf will be applied.
15–6 Reserved	This read-only field is reserved and always has the value 0.
5 periph_clk_sel_busy	Busy indicator for periph_clk_sel mux control. 0 mux is not busy and its value represents the actual division. 1 mux is busy with handshake process with module. The value read in the periph_clk_sel represents the previous value of select, and after the handshake periph_clk_sel value will be applied.
4 mmdc_ch0_podf_busy	Busy indicator for mmdc_ch0_axi_podf. 0 divider is not busy and its value represents the actual division. 1 divider is busy with handshake process with module. The value read in the divider represents the previous value of the division factor, and after the handshake the written value of the mmdc_ch0_axi_podf will be applied.

Table continues on the next page...

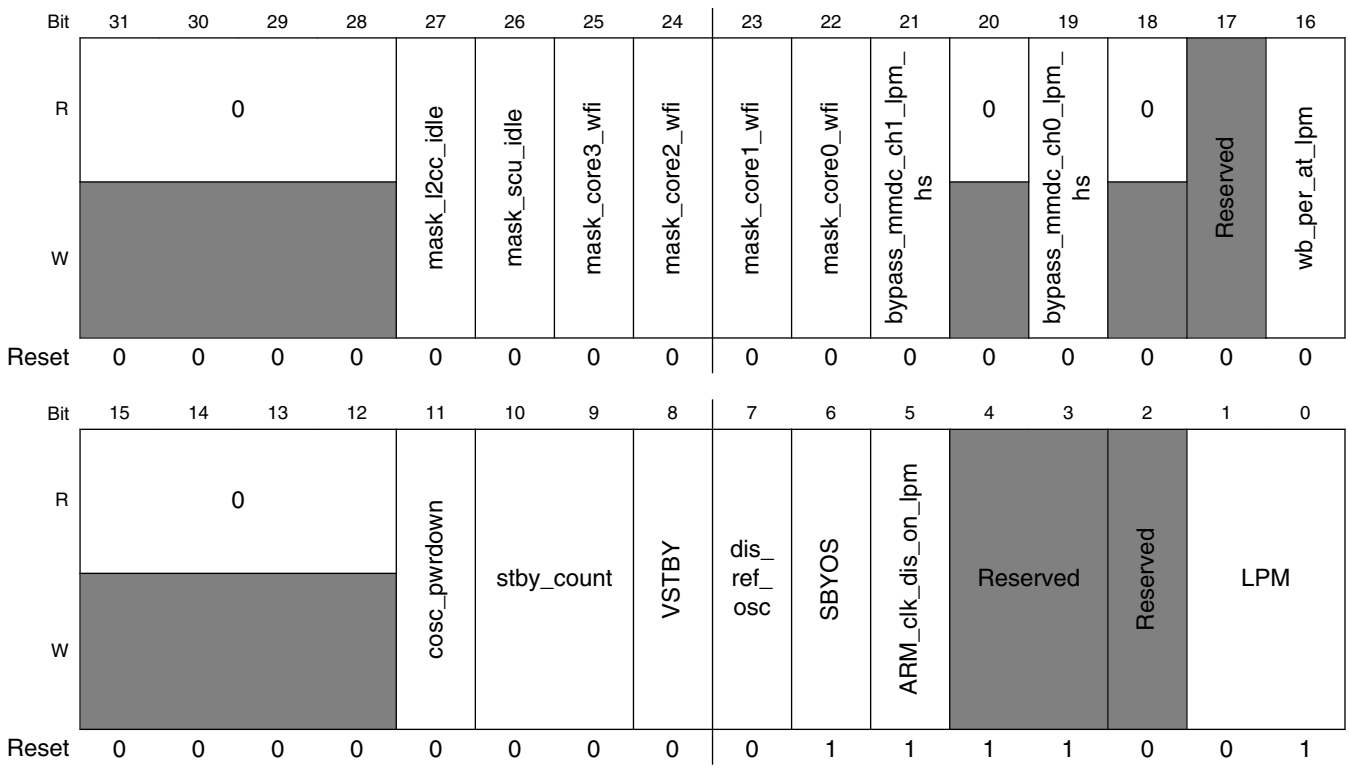
CCM_CDHIPR field descriptions (continued)

Field	Description
3 periph2_clk_sel_busy	Busy indicator for periph2_clk_sel mux control. 0 mux is not busy and its value represents the actual division. 1 mux is busy with handshake process with module. The value read in the periph2_clk_sel represents the previous value of select, and after the handshake periph2_clk_sel value will be applied.
2 mmdc_ch1_podf_busy	Busy indicator for mmdc_ch1_axi_podf. 0 divider is not busy and its value represents the actual division. 1 divider is busy with handshake process with module. The value read in the divider represents the previous value of the division factor, and after the handshake the written value of the mmdc_ch1_axi_podf will be applied.
1 ahb_podf_busy	Busy indicator for ahb_podf. 0 divider is not busy and its value represents the actual division. 1 divider is busy with handshake process with module. The value read in the divider represents the previous value of the division factor, and after the handshake the written value of the ahb_podf will be applied.
0 axi_podf_busy	Busy indicator for axi_podf. 0 divider is not busy and its value represents the actual division. 1 divider is busy with handshake process with module. The value read in the divider represents the previous value of the division factor, and after the handshake the written value of the axi_podf will be applied.

18.6.18 CCM Low Power Control Register (CCM_CLPCR)

The figure below represents the CCM Low Power Control Register (CLPCR). The CLPCR register contains bits to control the low power modes operation.The table below provides its field descriptions.

Address: 20C_4000h base + 54h offset = 20C_4054h



CCM_CLPCR field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27 mask_l2cc_idle	Mask L2CC IDLE for entering low power mode. **NOTE:** Assertion of all bits[27:22] will generate low power mode request 1 L2CC IDLE is masked 0 L2CC IDLE is not masked
26 mask_scu_idle	Mask SCU IDLE for entering low power mode **NOTE:** Assertion of all bits[27:22] will generate low power mode request 1 SCU IDLE is masked 0 SCU IDLE is not masked

Table continues on the next page...

CCM_CLPCR field descriptions (continued)

Field	Description
25 mask_core3_wfi	Mask WFI of core3 for entering low power mode **NOTE:** Assertion of all bits[27:22] will generate low power mode request 1 WFI of core3 is masked 0 WFI of core3 is not masked
24 mask_core2_wfi	Mask WFI of core2 for entering low power mode **NOTE:** Assertion of all bits[27:22] will generate low power mode request 1 WFI of core2 is masked 0 WFI of core2 is not masked
23 mask_core1_wfi	Mask WFI of core1 for entering low power mode **NOTE:** Assertion of all bits[27:22] will generate low power mode request 1 WFI of core1 is masked 0 WFI of core1 is not masked
22 mask_core0_wfi	Mask WFI of core0 for entering low power mode **NOTE:** Assertion of all bits[27:22] will generate low power mode request 0 WFI of core0 is not masked 1 WFI of core0 is masked
21 bypass_mmdc_ch1_lpm_hs	Bypass handshake with mmdc_ch1 on next entrance to low power mode (STOP or WAIT). CCM doesn't wait for the module's acknowledge. 0 Handshake with mmdc_ch1 on next entrance to low power mode will be performed. 1 Handshake with mmdc_ch1 on next entrance to low power mode will be bypassed.
20 Reserved	This read-only field is reserved and always has the value 0.
19 bypass_mmdc_ch0_lpm_hs	Bypass handshake with mmdc_ch0 on next entrance to low power mode (STOP or WAIT). CCM doesn't wait for the module's acknowledge. Handshake will also be bypassed, if CGR3 CG10 is set to gate fast mmdc_ch0 clock. 0 Handshake with mmdc_ch0 on next entrance to low power mode will be performed. . 1 Handshake with mmdc_ch0 on next entrance to low power mode will be bypassed.
18 Reserved	This read-only field is reserved and always has the value 0.
17 -	This field is reserved. Reserved
16 wb_per_at_lpm	Enable periphery charge pump for well biasing at low power mode (stop or wait) 0 Periphery charge pump won't be enabled at STOP or WAIT low power modes 1 Periphery charge pump will be enabled at STOP or WAIT low power modes
15–12 Reserved	This read-only field is reserved and always has the value 0.
11 cosc_pwrdown	In run mode, software can manually control powering down of on chip oscillator, i.e. generating '1' on cosc_pwrdown signal. If software manually powered down the on chip oscillator, then sbyos functionality for on chip oscillator will be bypassed.

Table continues on the next page...

CCM_CLPCR field descriptions (continued)

Field	Description
	The manual closing of onchip oscillator should be performed only in case the reference oscilator is not the source of all the clocks generation. 0 On chip oscillator will not be powered down, i.e. cosc_pwrdown = '0'. 1 On chip oscillator will be powered down, i.e. cosc_pwrdown = '1'.
10–9 stby_count	Standby counter definition. These two bits define, in the case of stop exit (if VSTBY bit was set). **NOTE:** Clock cycles ratio depends on pmic_delay_scaler, defined by CGPR[0] bit. 00 CCM will wait (1*pmic_delay_scaler)+1 ckil clock cycles 01 CCM will wait (3*pmic_delay_scaler)+1 ckil clock cycles 10 CCM will wait (7*pmic_delay_scaler)+1 ckil clock cycles 11 CCM will wait (15*pmic_delay_scaler)+1 ckil clock cycles
8 VSTBY	Voltage standby request bit. This bit defines if PMIC_STBY_REQ pin, which notifies external power management IC to move from functional voltage to standby voltage, will be asserted in STOP mode. 0 Voltage will not be changed to standby voltage after next entrance to STOP mode. (PMIC_STBY_REQ will remain negated - '0') 1 Voltage will be requested to change to standby voltage after next entrance to stop mode. (PMIC_STBY_REQ will be asserted - '1').
7 dis_ref_osc	dis_ref_osc - in run mode, software can manually control closing of external reference oscillator clock, i.e. generating '1' on CCM_REF_EN_B signal. If software closed manually the external reference clock, then sbyos functionality will be bypassed. The manual closing of external reference oscilator should be performed only in case the reference oscilator is not the source of any clock generation. **NOTE:** When returning from stop mode, the PMIC_STBY_REQ will be deasserted (if it was asserted when entering stop mode). See stby_count bits. 0 external high frequency oscillator will be enabled, i.e. CCM_REF_EN_B = '0'. 1 external high frequency oscillator will be disabled, i.e. CCM_REF_EN_B = '1'
6 SBYOS	Standby clock oscillator bit. This bit defines if cosc_pwrdown, which power down the on chip oscillator, will be asserted in STOP mode. This bit is discarded if cosc_pwrdown='1' for the on chip oscillator. 0 On-chip oscillator will not be powered down, after next entrance to STOP mode. (CCM_REF_EN_B will remain asserted - '0' and cosc_pwrdown will remain de asserted - '0') 1 On-chip oscillator will be powered down, after next entrance to STOP mode. (CCM_REF_EN_B will be deasserted - '1' and cosc_pwrdown will be asserted - '1'). When returning from STOP mode, external oscillator will be enabled again, on-chip oscillator will return to oscillator mode, and after oscnt count, CCM will continue with the exit from the STOP mode process.
5 ARM_clk_dis_on_lpm	Define if ARM clocks (arm_clk, soc_mxclk, soc_pclk, soc_dbg_pclk, vl_wrck) will be disabled on wait mode. This is useful for debug mode, when the user still wants to simulate entering wait mode and still keep ARM clock functioning. **NOTE:** Software should not enable ARM power gating in wait mode if this bit is cleared. 0 ARM clock enabled on wait mode. 1 ARM clock disabled on wait mode. .
4–3 -	This field is reserved. Reserved
2 -	This field is reserved. Reserved

Table continues on the next page...

CCM_CLPCR field descriptions (continued)

Field	Description
LPM	Setting the low power mode that system will enter on next assertion of dsm_request signal. **NOTE:** Set CCM_CGPR[INT_MEM_CLK_LPM] and CCM_CGPR[1] bits to 1 when setting CCM_CLPCR[LPM] bits to 01 (WAIT Mode) or 10 (STOP mode) without power gating. CCM_CGPR[INT_MEM_CLK_LPM] and CCM_CGPR[1] bits do not have to be set for STOP mode entry. 00 Remain in run mode 01 Transfer to wait mode 10 Transfer to stop mode 11 Reserved

18.6.19 CCM Interrupt Status Register (CCM_CISR)

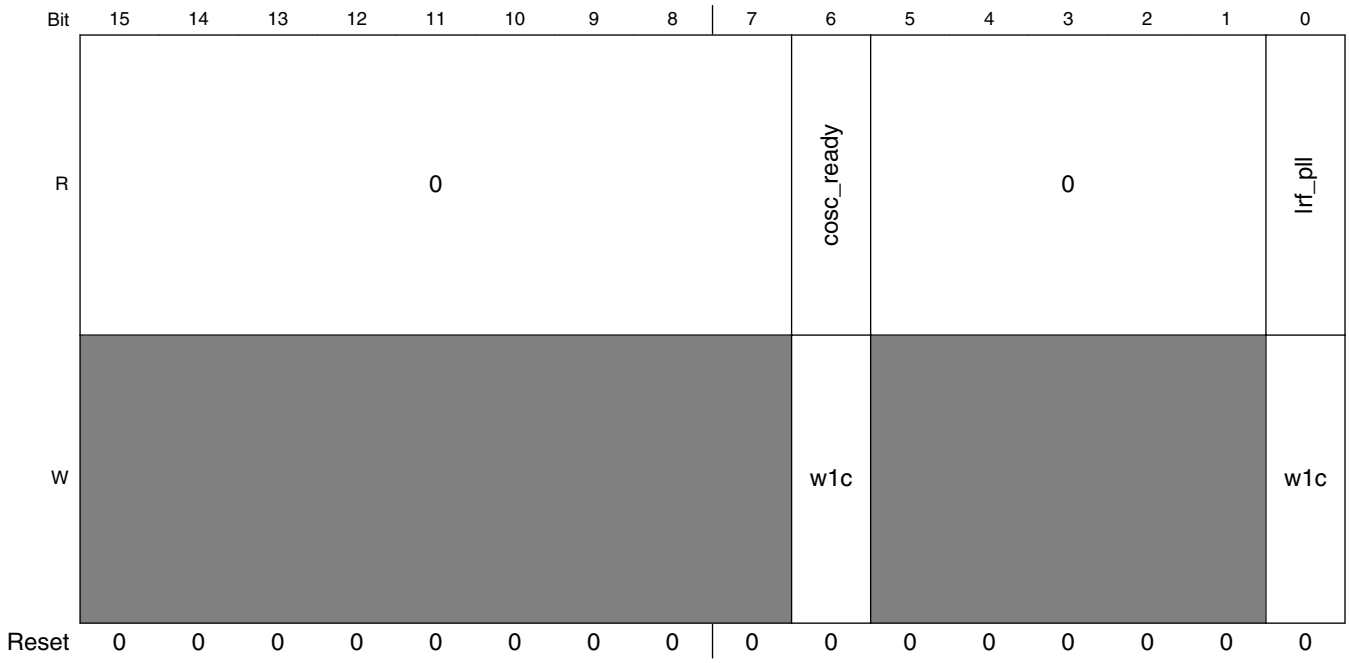
The figure below represents the CCM Interrupt Status Register (CISR). This is a write one to clear register. Once a interrupt is generated, software should write one to clear it. The table below provides its field descriptions.

NOTE

CCM interrupt request 1 can be masked by CCM interrupt request 1 mask bit. CCM interrupt request 2 can be masked by CCM interrupt request 2 mask bit.

Address: 20C_4000h base + 58h offset = 20C_4058h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0					arm_podf_loaded	0		mmdc_ch0_podf_loaded	periph_clk_sel_loaded	mmdc_ch1_podf_loaded	ahb_podf_loaded	periph2_clk_sel_loaded	0	axi_podf_loaded	0
W						w1c			w1c	w1c	w1c	w1c	w1c		w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



CCM_CISR field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26 arm_podf_loaded	CCM interrupt request 1 generated due to frequency change of arm_podf. The interrupt will commence only if arm_podf is loaded during a arm dvfs operation. 0 interrupt is not generated due to frequency change of arm_podf 1 interrupt generated due to frequency change of arm_podf
25–24 Reserved	This read-only field is reserved and always has the value 0.
23 mmdc_ch0_ podf_loaded	CCM interrupt request 1 generated due to update of mmdc_ch0_axi_podf. 0 interrupt is not generated due to update of mmdc_ch0_axi_podf. 1 interrupt generated due to update of mmdc_ch0_axi_podf*
22 periph_clk_sel_ loaded	CCM interrupt request 1 generated due to update of periph_clk_sel. 0 interrupt is not generated due to update of periph_clk_sel. 1 interrupt generated due to update of periph_clk_sel.
21 mmdc_ch1_ podf_loaded	CCM interrupt request 1 generated due to frequency change of mmdc_ch0_podf_ **loaded** 0 interrupt is not generated due to frequency change of mmdc_ch0_podf_ **loaded** 1 interrupt generated due to frequency change of mmdc_ch0_podf_ **loaded**
20 ahb_podf_loaded	CCM interrupt request 1 generated due to frequency change of ahb_podf 0 interrupt is not generated due to frequency change of ahb_podf 1 interrupt generated due to frequency change of ahb_podf
19 periph2_clk_sel_ loaded	CCM interrupt request 1 generated due to frequency change of periph2_clk_sel

Table continues on the next page...

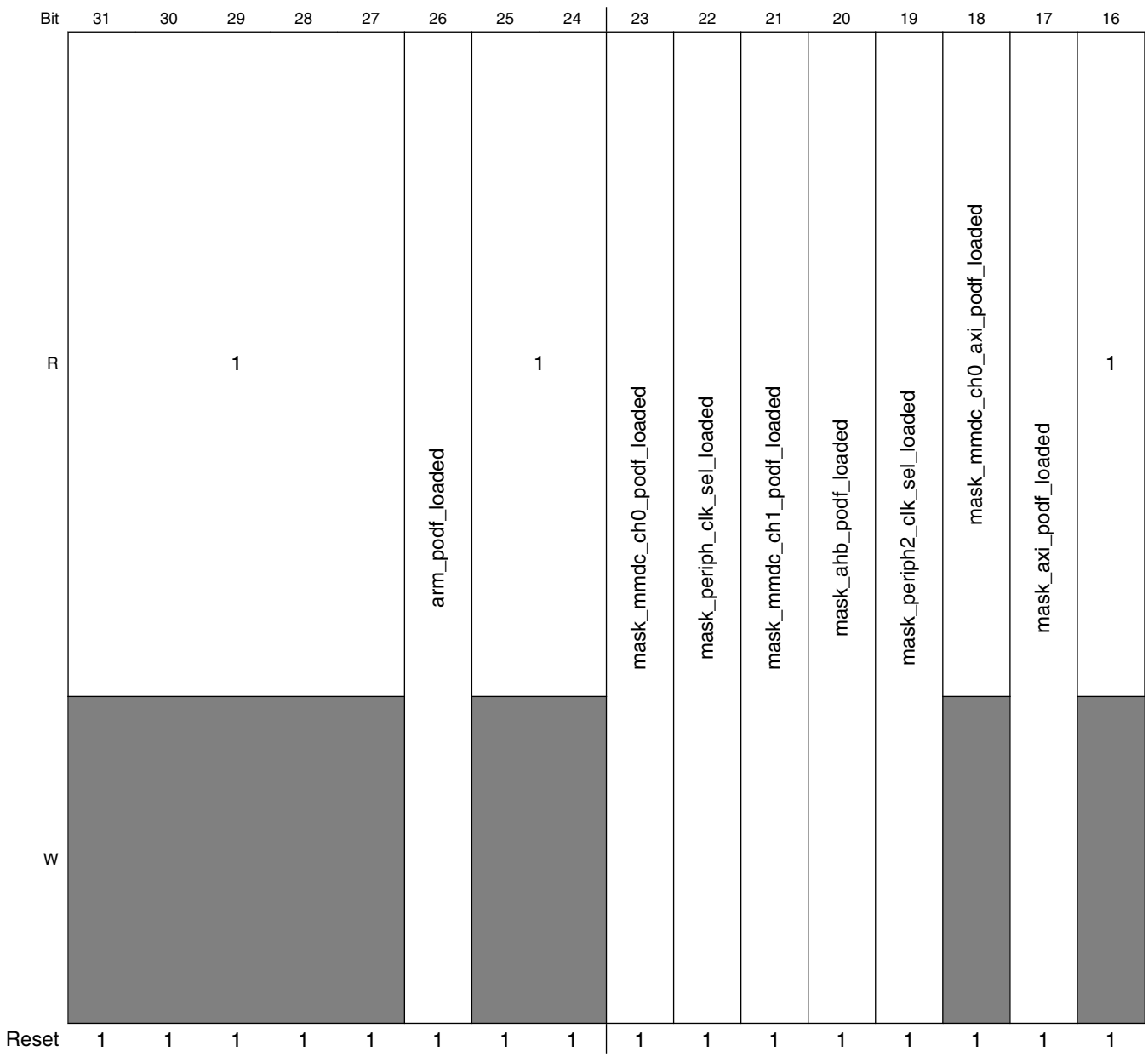
CCM_CISR field descriptions (continued)

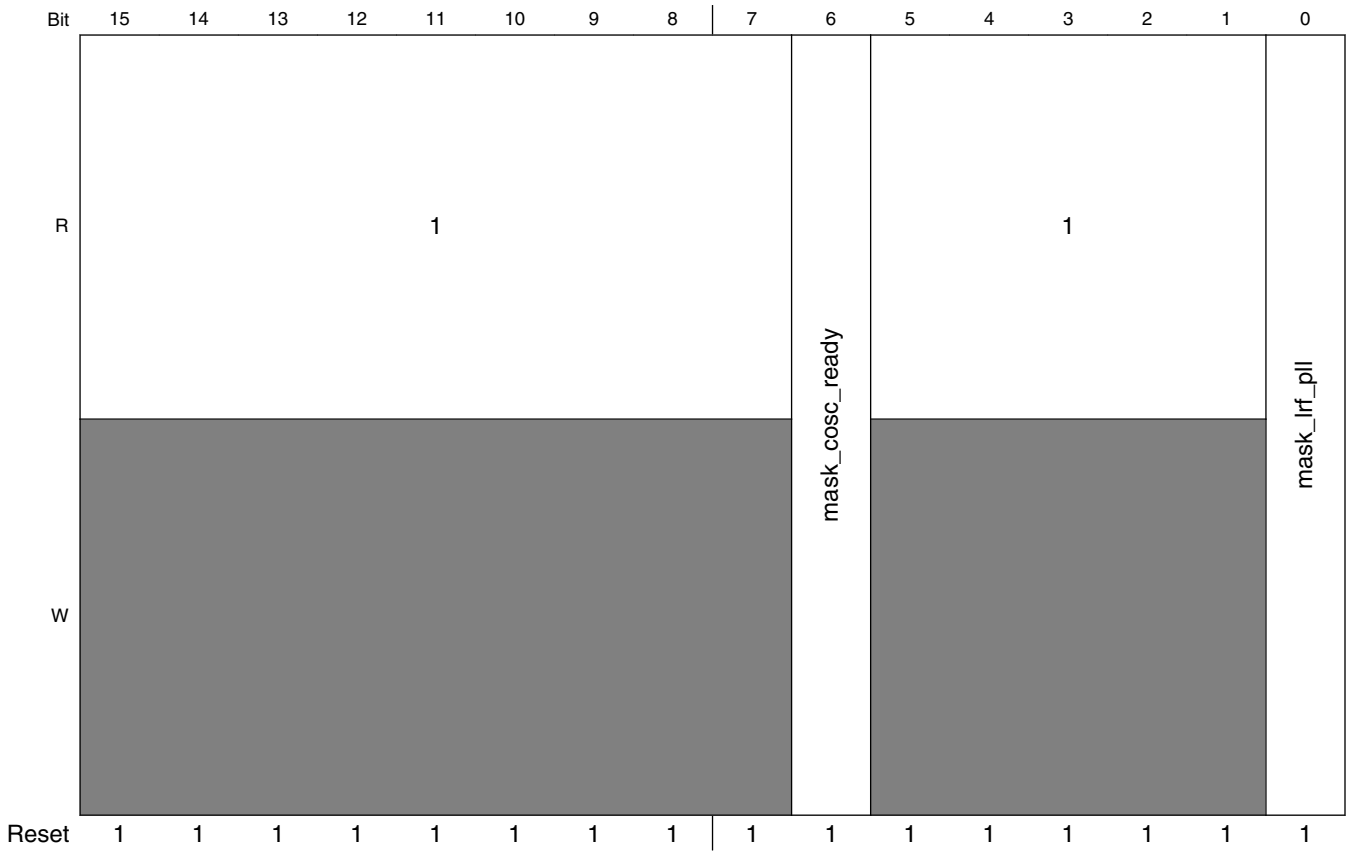
Field	Description
	0 interrupt is not generated due to frequency change of periph2_clk_sel 1 interrupt generated due to frequency change of periph2_clk_sel
18 Reserved	This read-only field is reserved and always has the value 0. 0 interrupt is not generated due to frequency change of mmdc_ch0_axi_podf 1 interrupt generated due to frequency change of mmdc_ch0_axi_podf
17 axi_podf_loaded	CCM interrupt request 1 generated due to frequency change of axi_podf 0 interrupt is not generated due to frequency change of axi_podf 1 interrupt generated due to frequency change of axi_podf
16–7 Reserved	This read-only field is reserved and always has the value 0.
6 cosc_ready	CCM interrupt request 2 generated due to on board oscillator ready, i.e. oscnt has finished counting. 0 interrupt is not generated due to on board oscillator ready 1 interrupt generated due to on board oscillator ready
5–1 Reserved	This read-only field is reserved and always has the value 0.
0 lrf_pll	CCM interrupt request 2 generated due to lock of all enabled and not bypaseed PLLs 0 interrupt is not generated due to lock ready of all enabled and not bypaseed PLLs 1 interrupt generated due to lock ready of all enabled and not bypaseed PLLs

18.6.20 CCM Interrupt Mask Register (CCM_CIMR)

The figure below represents the CCM Interrupt Mask Register (CIMR). The table below provides its field descriptions.

Address: 20C_4000h base + 5Ch offset = 20C_405Ch





CCM_CIMR field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 1.
26 arm_podf_loaded	mask interrupt generation due to frequency change of arm_podf 0 don't mask interrupt due to frequency change of arm_podf - interrupt will be created 1 mask interrupt due to frequency change of arm_podf
25–24 Reserved	This read-only field is reserved and always has the value 1.
23 mask_mmdc_ ch0_podf_loaded	mask interrupt generation due to update of mask_mmdc_ch0_podf 0 don't mask interrupt due to update of mask_mmdc_ch0_podf - interrupt will be created 1 mask interrupt due to update of mask_mmdc_ch0_podf
22 mask_periph_ clk_sel_loaded	mask interrupt generation due to update of periph_clk_sel. 0 don't mask interrupt due to update of periph_clk_sel - interrupt will be created 1 mask interrupt due to update of periph_clk_sel
21 mask_mmdc_ ch1_podf_loaded	mask interrupt generation due to update of mask_mmdc_ch1_podf 0 don't mask interrupt due to update of mask_mmdc_ch1_podf - interrupt will be created 1 mask interrupt due to update of mask_mmdc_ch1_podf

Table continues on the next page...

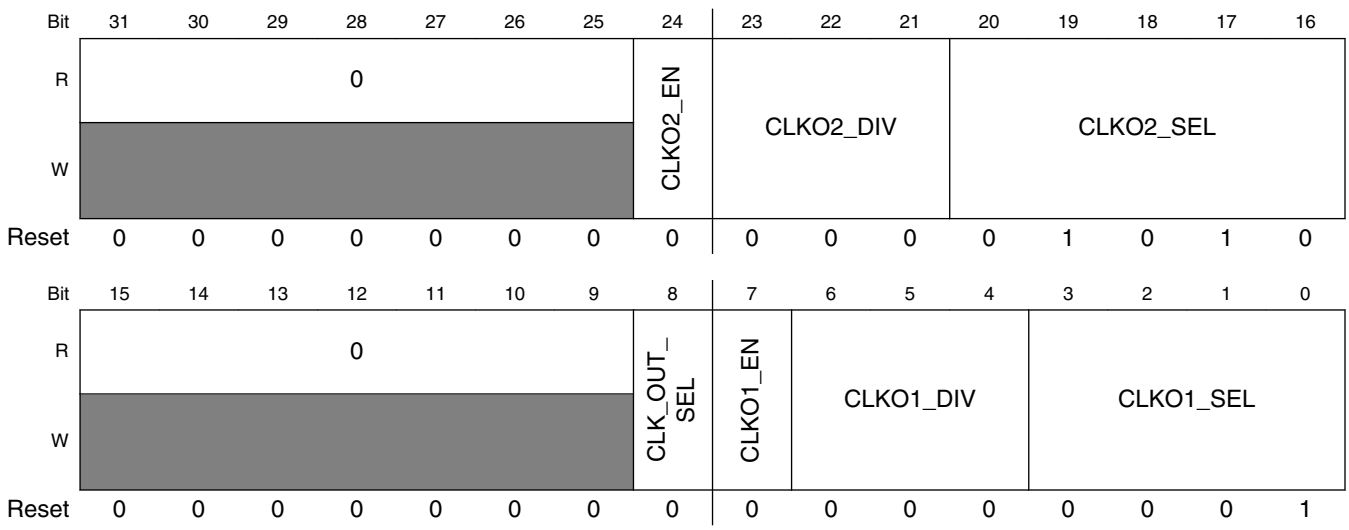
CCM_CIMR field descriptions (continued)

Field	Description
20 mask_ahb_podf_loaded	mask interrupt generation due to frequency change of ahb_podf 0 don't mask interrupt due to frequency change of ahb_podf - interrupt will be created 1 mask interrupt due to frequency change of ahb_podf
19 mask_periph2_clk_sel_loaded	mask interrupt generation due to update of periph2_clk_sel. 0 don't mask interrupt due to update of periph2_clk_sel - interrupt will be created 1 mask interrupt due to update of periph2_clk_sel
18 mask_mmdc_ch0_axi_podf_loaded	mask interrupt generation due to frequency change of mmdc_ch0_axi_podf 0 don't mask interrupt due to frequency change of mmdc_ch0_axi_podf - interrupt will be created 1 mask interrupt due to frequency change of mmdc_ch0_axi_podf
17 mask_axi_podf_loaded	mask interrupt generation due to frequency change of axi_podf 0 don't mask interrupt due to frequency change of axi_podf - interrupt will be created 1 mask interrupt due to frequency change of axi_podf
16–7 Reserved	This read-only field is reserved and always has the value 1.
6 mask_cosc_ready	mask interrupt generation due to on board oscillator ready 0 don't mask interrupt due to on board oscillator ready - interrupt will be created 1 mask interrupt due to on board oscillator ready
5–1 Reserved	This read-only field is reserved and always has the value 1.
0 mask_lrf_pll	mask interrupt generation due to lrf of PLLs 0 don't mask interrupt due to lrf of PLLs - interrupt will be created 1 mask interrupt due to lrf of PLLs

18.6.21 CCM Clock Output Source Register (CCM_CCOSR)

The figure below represents the CCM Clock Output Source Register (CCOSR). The CCOSR register contains bits to control the clocks that will be generated on the output ipp_do_clko1 (CCM_CLKO1) and ipp_do_clko2 (CCM_CLKO2).The table below provides its field descriptions.

Address: 20C_4000h base + 60h offset = 20C_4060h



CCM_CCOSR field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 CLKO2_EN	Enable of CCM_CLKO2 clock 0 CCM_CLKO2 disabled. 1 CCM_CLKO2 enabled.
23–21 CLKO2_DIV	Setting the divider of CCM_CLKO2 000 divide by 1 001 divide by 2 010 divide by 3 011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
20–16 CLKO2_SEL	Selection of the clock to be generated on CCM_CLKO2

Table continues on the next page...

CCM_CCOSR field descriptions (continued)

Field	Description
	00000 mmdc_ch0_clk_root 00001 mmdc_ch1_clk_root 00010 usdhc4_clk_root 00011 usdhc1_clk_root 00100 gpu2d_axi_clk_root 00101 wrck_clk_root 00110 ecspi_clk_root 00111 gpu3d_axi_clk_root 01000 usdhc3_clk_root 01001 125M_clk_root 01010 arm_clk_root 01011 ipu1_hsp_clk_root 01100 ipu2_hsp_clk_root 01101 vdo_axi_clk_root 01110 osc_clk 01111 gpu2d_core_clk_root 10000 gpu3d_core_clk_root 10001 usdhc2_clk_root 10010 ssi1_clk_root 10011 ssi2_clk_root 10100 ssi3_clk_root 10101 gpu3d_shader_clk_root 10110 vpu_axi_clk_root 10111 can_clk_root 11000 ldb_di0_serial_clk_root 11001 ldb_di1_serial_clk_root 11010 esai_clk_root 11011 aclk_eim_slow_clk_root 11100 uart_clk_root 11101 spdif0_clk_root 11110 spdif1_clk_root 11111 hsi_tx_clk_root
15–9 Reserved	This read-only field is reserved and always has the value 0.
8 CLK_OUT_SEL	CCM_CLKO1 output to reflect CCM_CLKO1 or CCM_CLKO2 clocks 0 CCM_CLKO1 output drives CCM_CLKO1 clock 1 CCM_CLKO1 output drives CCM_CLKO2 clock
7 CLKO1_EN	Enable of CCM_CLKO1 clock 0 CCM_CLKO1 disabled. 1 CCM_CLKO1 enabled.
6–4 CLKO1_DIV	Setting the divider of CCM_CLKO1 000 divide by 1 001 divide by 2 010 divide by 3

Table continues on the next page...

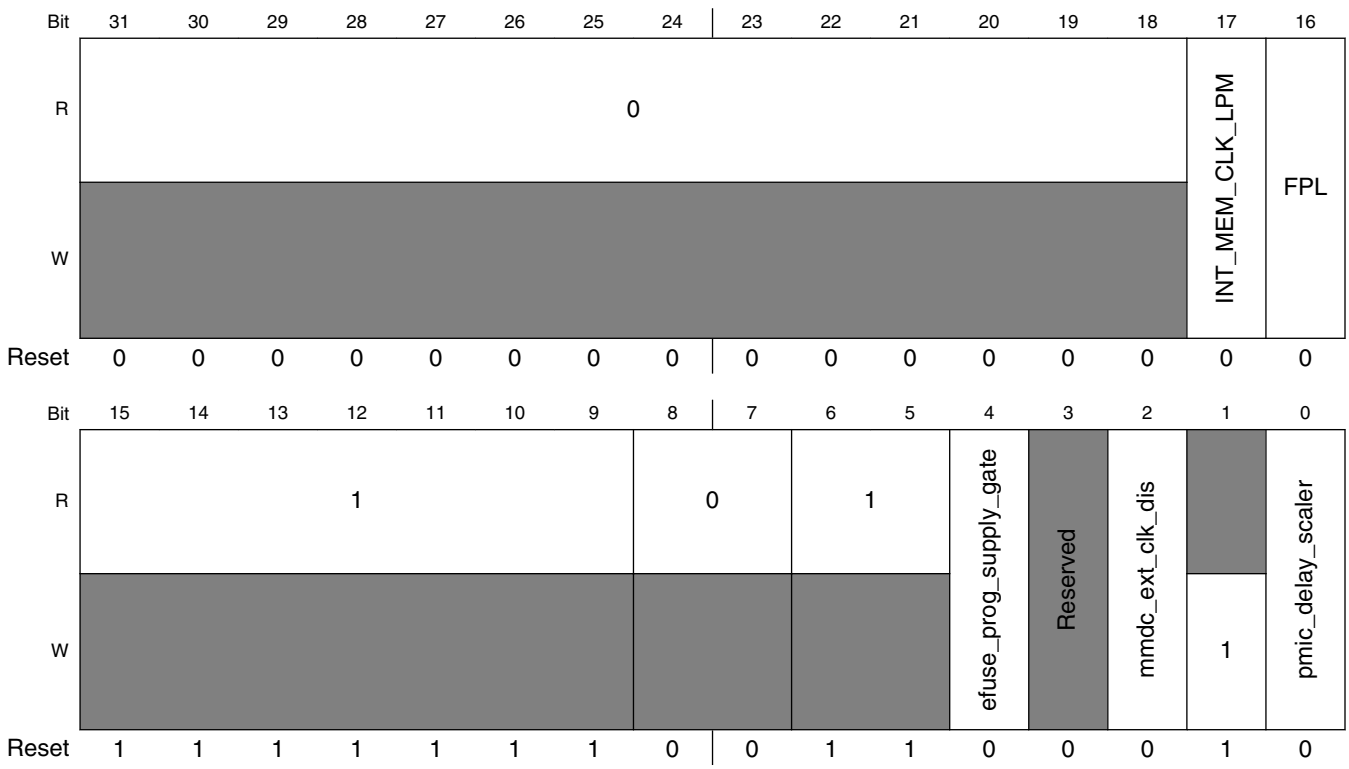
CCM_CCOSR field descriptions (continued)

Field	Description
	011 divide by 4 100 divide by 5 101 divide by 6 110 divide by 7 111 divide by 8
CLKO1_SEL	Selection of the clock to be generated on CCM_CLKO1 0000 pll3_sw_clk (this inputs has additional constant division /2) 0001 pll2_main_clk (this inputs has additional constant division /2) 0010 pll1_main_clk (this inputs has additional constant division /2) 0011 pll5_main_clk (this inputs has additional constant division /2) 0100 video_27M_clk_root 0101 axi_clk_root 0110 enfc_clk_root 0111 ipu1_di0_clk_root 1000 ipu1_di1_clk_root 1001 ipu2_di0_clk_root 1010 ipu2_di1_clk_root 1011 ahb_clk_root 1100 ipg_clk_root 1101 perclk_root 1110 ckil_sync_clk_root 1111 pll4_main_clk

18.6.22 CCM General Purpose Register (CCM_CGPR)

Fast PLL enable. Can be used to engage PLL faster after STOP mode, if 24MHz OSC was active

Address: 20C_4000h base + 64h offset = 20C_4064h



CCM_CGPR field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17 INT_MEM_CLK_LPM	Control for the Deep Sleep signal to the ARM Platform memories with additional control logic based on the ARM WFI signal. Used to keep the ARM Platform memory clocks enabled if an interrupt is pending when entering low power mode. **NOTE:** This bit should always bet set when the CCM_CLPCR_LPM bits are set to 01(WAIT Mode) or 10 (STOP mode) without power gating. This bit does not have to be set for STOP mode entry. 0 Disable the clock to the ARM platform memories when entering Low Power Mode 1 Keep the clocks to the ARM platform memories enabled only if an interrupt is pending when entering Low Power Modes (WAIT and STOP without power gating)
16 FPL	Fast PLL enable.

Table continues on the next page...

CCM_CGPR field descriptions (continued)

Field	Description
	0 Engage PLL enable default way. 1 Engage PLL enable 3 CKIL clocks earlier at exiting low power mode (STOP). Should be used only if 24MHz OSC was active in low power mode.
15–9 Reserved	This read-only field is reserved and always has the value 1.
8–7 Reserved	This read-only field is reserved and always has the value 0.
6–5 Reserved	This read-only field is reserved and always has the value 1.
4 efuse_prog_supply_gate	Defines the value of the output signal cgpr_dout[4]. Gate of program supply for efuse programing 0 fuse programing supply voltage is gated off to the efuse module 1 allow fuse programing.
3 -	This field is reserved. Reserved
2 mmdc_ext_clk_dis	Disable external clock driver of MMDC during STOP mode 1 disable during stop mode 0 don't disable during stop mode.
1 -	Reserved. Keep default value set to '1' for proper operation.
0 pmic_delay_scaler	Defines clock dividion of clock for stby_count (pmic delay counter) 0 clock is not divided 1 clock is divided /8

18.6.23 CCM Clock Gating Register 0 (CCM_CCGR0)

CG(i) bits CCGR 0-6

These bits are used to turn on/off the clock to each module independently. The following table details the possible clock activity conditions for each module.

CGR value	Clock Activity Description
00	Clock is off during all modes. Stop enter hardware handshake is disabled.
01	Clock is on in run mode, but off in WAIT and STOP modes
10	Not applicable (Reserved).
11	Clock is on during all modes, except STOP mode.

Module should be stopped, before set its bits to "0"; clocks to the module will be stopped immediately.

The tables above show the register mapings for the different CGRs. The clock connectivity table should be used to match the "CCM output affected" to the actual clocks going into the modules.

The figure below represents the CCM Clock Gating Register 0 (CCM_CCGR0). The clock gating Registers define the clock gating for power reduction of each clock (CG(i) bits). There are 7 CGR registers. The number of registers required is according to the number of peripherals in the system.

Address: 20C_4000h base + 68h offset = 20C_4068h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR0 field descriptions

Field	Description
31–30 CG15	Reserved
29–28 CG14	dtcp clocks (dtcp_clk_enable)
27–26 CG13	dcic2 clocks (dcic2_clk_enable)
25–24 CG12	dcic 1 clocks (dcic1_clk_enable)
23–22 CG11	CPU debug clocks (arm_dbg_clk_enable)
21–20 CG10	can2_serial clock (can2_serial_clk_enable)
19–18 CG9	can2 clock (can2_clk_enable)
17–16 CG8	can1_serial clock (can1_serial_clk_enable)
15–14 CG7	can1 clock (can1_clk_enable)
13–12 CG6	caam_wrapper_ipg clock (caam_wrapper_ipg_enable)
11–10 CG5	caam_wrapper_aclk clock (caam_wrapper_aclk_enable)
9–8 CG4	caam_secure_mem clock (caam_secure_mem_clk_enable)
7–6 CG3	asrc clock (asrc_clk_enable)

Table continues on the next page...

CCM_CCGR0 field descriptions (continued)

Field	Description
5–4 CG2	apbhdma hclk clock (apbhdma_hclk_enable)
3–2 CG1	aips_tz2 clocks (aips_tz2_clk_enable)
CG0	aips_tz1 clocks (aips_tz1_clk_enable)

18.6.24 CCM Clock Gating Register 1 (CCM_CCGR1)

The figure below represents the CCM Clock Gating Register 1(CCM_CCGR1). The clock gating registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 6Ch offset = 20C_406Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR1 field descriptions

Field	Description
31–30 CG15	Reserved
29–28 CG14	Reserved
27–26 CG13	gpu3d clock (gpu3d_clk_enable)
25–24 CG12	gpu2d clock (gpu2d_clk_enable) **NOTE:** GPU2D clock cannot be gated without gating OPENVG clock as well. Configure both CG bits (CCM_ANALOG_CCGR1[CG12] and CCM_ANALOG_CCGR3[CG15]), to gate GPU2D.
23–22 CG11	gpt serial clock (gpt_serial_clk_enable)
21–20 CG10	gpt bus clock (gpt_clk_enable)
19–18 CG9	Reserved

Table continues on the next page...

CCM_CCGR1 field descriptions (continued)

Field	Description
17–16 CG8	esai clocks (esai_clk_enable)
15–14 CG7	epit2 clocks (epit2_clk_enable)
13–12 CG6	epit1 clocks (epit1_clk_enable)
11–10 CG5	enet clock (enet_clk_enable)
9–8 CG4	ecspi5 clocks (ecspi5_clk_enable)
7–6 CG3	ecspi4 clocks (ecspi4_clk_enable)
5–4 CG2	ecspi3 clocks (ecspi3_clk_enable)
3–2 CG1	ecspi2 clocks (ecspi2_clk_enable)
CG0	ecspi1 clocks (ecspi1_clk_enable)

18.6.25 CCM Clock Gating Register 2 (CCM_CCGR2)

The figure below represents the CCM Clock Gating Register 2 (CCM_CCGR2). The clock gating registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 70h offset = 20C_4070h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	0	0	0	0	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR2 field descriptions

Field	Description
31–30 CG15	Reserved
29–28 CG14	Reserved

Table continues on the next page...

CCM_CCGR2 field descriptions (continued)

Field	Description
27–26 CG13	ipsync_vdoa_ipg clocks (ipsync_vdoa_ipg_master_clk_enable)
25–24 CG12	ipsync_ip2apb_tzasc2_ipg clocks (ipsync_ip2apb_tzasc2_ipg_master_clk_enable) >
23–22 CG11	ipsync_ip2apb_tzasc1_ipg clocks (ipsync_ip2apb_tzasc1_ipg_master_clk_enable)
21–20 CG10	ipmux3 clock (ipmux3_clk_enable)
19–18 CG9	ipmux2 clock (ipmux2_clk_enable)
17–16 CG8	ipmux1 clock (ipmux1_clk_enable)
15–14 CG7	iomux_ipt_clk_io clock (iomux_ipt_clk_io_enable)
13–12 CG6	OCOTP_CTRL clock (iim_clk_enable)
11–10 CG5	i2c3_serial clock (i2c3_serial_clk_enable)
9–8 CG4	i2c2_serial clock (i2c2_serial_clk_enable)
7–6 CG3	i2c1_serial clock (i2c1_serial_clk_enable)
5–4 CG2	hdmi_tx_isfrclk clock (hdmi_tx_isfrclk_enable)
3–2 CG1	Reserved
CG0	hdmi_tx_iahbclk, hdmi_tx_ihclk clock (hdmi_tx_enable)

18.6.26 CCM Clock Gating Register 3 (CCM_CCGR3)

The figure below represents the CCM Clock Gating Register 3 (CCM_CCGR3). The clock gating Registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 74h offset = 20C_4074h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR3 field descriptions

Field	Description
31–30 CG15	openvgaxiclk clock (openvgaxiclk_clk_root_enable) **NOTE:** OPENVG clock cannot be gated without gating GPU2D clock as well. Configure both CG bits (CCM_ANALOG_CCGR1[CG12] and CCM_ANALOG_CCGR3[CG15]) to gate OPENVG.
29–28 CG14	ocram clock (ocram_clk_enable)
27–26 CG13	Reserved
25–24 CG12	mmdc_core_ipg_clk_p0 clock (mmdc_core_ipg_clk_p0_enable)
23–22 CG11	Reserved
21–20 CG10	mmdc_core_aclk_fast_core_p0 clock (mmdc_core_aclk_fast_core_p0_enable)
19–18 CG9	mlb clock (mlb_clk_enable)
17–16 CG8	mipi_core_cfg clock (mipi_core_cfg_clk_enable)
15–14 CG7	ldb_di1 clock (ldb_di1_clk_enable)
13–12 CG6	ldb_di0 clock (ldb_di0_clk_enable)
11–10 CG5	ipu2_di1 clock and pre-clock (ipu2_ipu_di1_clk_enable)
9–8 CG4	ipu2_di0 clock and pre-clock (ipu2_ipu_di0_clk_enable)

Table continues on the next page...

CCM_CCGR3 field descriptions (continued)

Field	Description
7–6 CG3	ipu2_ipu clock (ipu2_ipu_clk_enable)
5–4 CG2	ipu1_di1 clock and pre-clock (ipu1_ipu_di1_clk_enable)
3–2 CG1	ipu1_di0 clock and pre-clock (ipu1_ipu_di0_clk_enable)
CG0	ipu1_ipu clock (ipu1_ipu_clk_enable)

18.6.27 CCM Clock Gating Register 4 (CCM_CCGR4)

The figure below represents the CCM Clock Gating Register 4 (CCM_CCGR4). The clock gating Registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 78h offset = 20C_4078h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR4 field descriptions

Field	Description
31–30 CG15	rawnand_u_gpmi_input_apb clock (rawnand_u_gpmi_input_apb_clk_enable)
29–28 CG14	rawnand_u_gpmi_bch_input_gpmi_io clock (rawnand_u_gpmi_bch_input_gpmi_io_clk_enable)
27–26 CG13	rawnand_u_gpmi_bch_input_bch clock (rawnand_u_gpmi_bch_input_bch_clk_enable)
25–24 CG12	rawnand_u_bch_input_apb clock (rawnand_u_bch_input_apb_clk_enable)
23–22 CG11	pwm4 clocks (pwm4_clk_enable)
21–20 CG10	pwm3 clocks (pwm3_clk_enable)
19–18 CG9	pwm2 clocks (pwm2_clk_enable)

Table continues on the next page...

CCM_CCGR4 field descriptions (continued)

Field	Description
17–16 CG8	pwm1 clocks (pwm1_clk_enable)
15–14 CG7	pl301_mx6qper2_mainclk_enable (pl301_mx6qper2_mainclk_enable)
13–12 CG6	pl301_mx6qper1_bch clocks (pl301_mx6qper1_bchclk_enable)
11–10 CG5	Reserved
9–8 CG4	pl301_mx6qfast1_s133 clock (pl301_mx6qfast1_s133clk_enable)
7–6 CG3	Reserved.
5–4 CG2	Reserved.
3–2 CG1	Reserved.
CG0	pcie clock (pcie_root_enable)

18.6.28 CCM Clock Gating Register 5 (CCM_CCGR5)

The figure below represents the CCM Clock Gating Register 5 (CCM_CCGR5). The clock gating Registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 7Ch offset = 20C_407Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR5 field descriptions

Field	Description
31–30 CG15	Reserved
29–28 CG14	Reserved

Table continues on the next page...

CCM_CCGR5 field descriptions (continued)

Field	Description
27–26 CG13	uart_serial clock (uart_serial_clk_enable)
25–24 CG12	uart clock (uart_clk_enable)
23–22 CG11	ssi3 clocks (ssi3_clk_enable)
21–20 CG10	ssi2 clocks (ssi2_clk_enable)
19–18 CG9	ssi1 clocks (ssi1_clk_enable)
17–16 CG8	Reserved
15–14 CG7	spdif clock (spdif_clk_enable)
13–12 CG6	spba clock (spba_clk_enable)
11–10 CG5	Reserved
9–8 CG4	Reserved
7–6 CG3	sdma clock (sdma_clk_enable)
5–4 CG2	sata clock (sata_clk_enable)
3–2 CG1	Reserved
CG0	rom clock (rom_clk_enable)

18.6.29 CCM Clock Gating Register 6 (CCM_CCGR6)

The figure below represents the CCM Clock Gating Register 6 (CCM_CCGR6). The clock gating Registers define the clock gating for power reduction of each clock (CG(i) bits). There are 8 CGR registers. The number of registers required is determined by the number of peripherals in the system.

Address: 20C_4000h base + 80h offset = 20C_4080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CG15		CG14		CG13		CG12		CG11		CG10		CG9		CG8	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CG7		CG6		CG5		CG4		CG3		CG2		CG1		CG0	
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CCM_CCGR6 field descriptions

Field	Description
31–30 CG15	Reserved
29–28 CG14	Reserved
27–26 CG13	Reserved
25–24 CG12	Reserved
23–22 CG11	Reserved
21–20 CG10	Reserved
19–18 CG9	Reserved
17–16 CG8	Reserved
15–14 CG7	vpu clocks (vpu_clk_enable)
13–12 CG6	vdoaxiclk root clock (vdoaxiclk_clk_enable)
11–10 CG5	eim_slow clocks (eim_slow_clk_enable)
9–8 CG4	usdhc4 clocks (usdhc4_clk_enable)

Table continues on the next page...

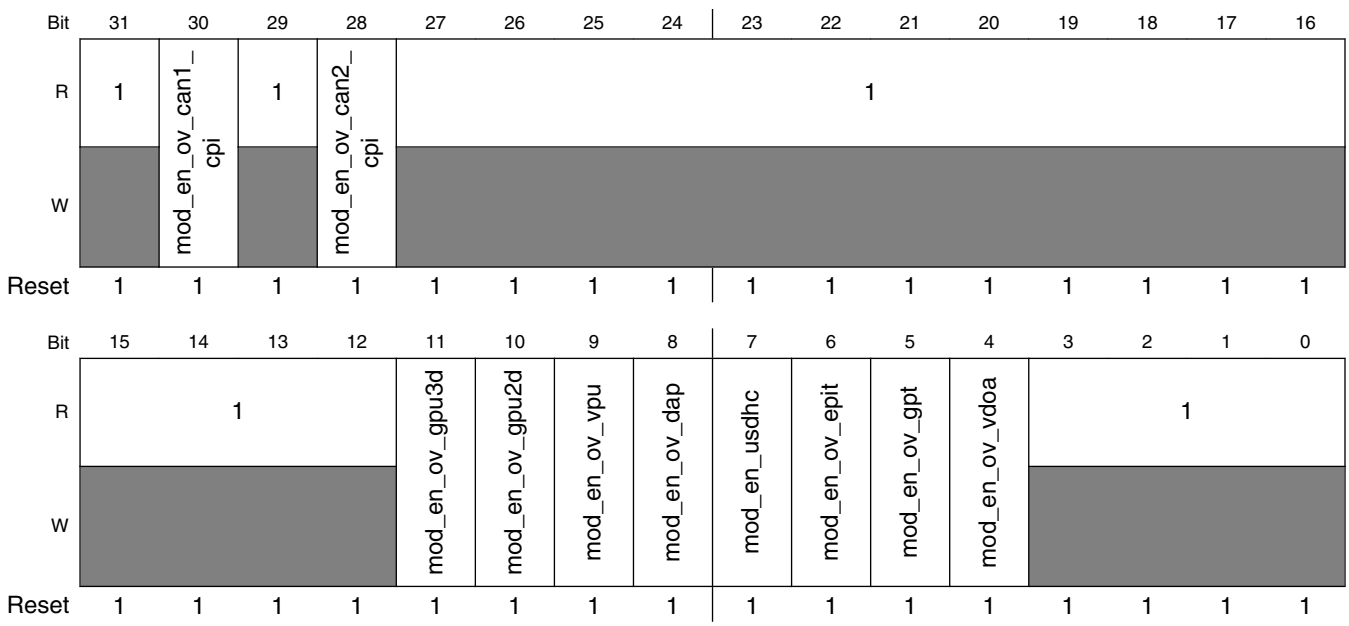
CCM_CCGR6 field descriptions (continued)

Field	Description
7–6 CG3	usdhc3 clocks (usdhc3_clk_enable)
5–4 CG2	usdhc2 clocks (usdhc2_clk_enable)
3–2 CG1	usdhc1 clocks (usdhc1_clk_enable)
CG0	usboh3 clock (usboh3_clk_enable)

18.6.30 CCM Module Enable Overide Register (CCM_CMEOR)

The following figure represents the CCM Module Enable Override Register (CMEOR). The CMEOR register contains bits to override the clock enable signal from the module. This bit is applicable only for modules whose clock enable signals are used. The following table provides its field descriptions.

Address: 20C_4000h base + 88h offset = 20C_4088h



CCM_CMEOR field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 1.

Table continues on the next page...

CCM_CMEOR field descriptions (continued)

Field	Description
30 mod_en_ov_ can1_cpi	Overide clock enable signal from CAN1 - clock will not be gated based on CAN's signal 'enable_clk_cpi'. 0 don't overide module enable signal 1 overide module enable signal
29 Reserved	This read-only field is reserved and always has the value 1.
28 mod_en_ov_ can2_cpi	Overide clock enable signal from CAN2 - clock will not be gated based on CAN's signal 'enable_clk_cpi'. 0 don't override module enable signal 1 override module enable signal
27–12 Reserved	This read-only field is reserved and always has the value 1.
11 mod_en_ov_ gpu3d	Overide clock enable signal from GPU3D - clock will not be gated based on GPU3D's signal. 0 don't override module enable signal 1 override module enable signal
10 mod_en_ov_ gpu2d	Overide clock enable signal from GPU2D - clock will not be gated based on GPU's signal 'gpu2d_busy' . 0 don't override module enable signal 1 override module enable signal
9 mod_en_ov_vpu	Overide clock enable signal from VPU- clock will not be gated based on VPU's signal 'vpu_idle' . 0 don't override module enable signal 1 override module enable signal
8 mod_en_ov_dap	Overide clock enable signal from DAP- clock will not be gated based on DAP's signal 'dap_dbgen' . 0 don't override module enable signal 1 override module enable signal
7 mod_en_usdhc	Overide clock enable signal from USDHC. 0 don't override module enable signal 1 override module enable signal
6 mod_en_ov_epit	Overide clock enable signal from EPIT - clock will not be gated based on EPIT's signal 'ipg_enable_clk' . 0 don't override module enable signal 1 override module enable signal
5 mod_en_ov_gpt	Overide clock enable signal from GPT - clock will not be gated based on GPT's signal 'ipg_enable_clk' . 0 don't override module enable signal 1 override module enable signal
4 mod_en_ov_ vdoa	Overide clock enable signal from vdoa - clock will not be gated based on vdoa signal. 0 don't override module enable signal 1 override module enable signal
Reserved	This read-only field is reserved and always has the value 1.

18.7 CCM Analog Memory Map/Register Definition

This section describes the registers for the analog PLLs. The registers which have the same description are grouped within {}. The register offsets for the various PLLs are:

- ARM PLL: {0h000, 0h004, 0h008, 0h00C}.
- USB1 PLL: {0h010, 0h014, 0h018, 0h01C}, {0h0F0, 0h0F4, 0h0F8, 0h0FC}.
- System PLL: {0h030, 0h034, 0h038, 0h03C}, 0h040, 0h050, 0h060, {0h100, 0h104, 0h108, 0h10C}.
- Audio / Video PLL: {0h070, 0h074, 0h078, 0h07C}, 0h080, 0h090, {0h0A0, 0h0A4, 0h0A8, 0h0AC}, 0h0B0, 0h0C0

CCM_ANALOG memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20C_8000	Analog ARM PLL control Register (CCM_ANALOG_PLL_ARM)	32	R/W	0001_3042h	18.7.1/905
20C_8004	Analog ARM PLL control Register (CCM_ANALOG_PLL_ARM_SET)	32	R/W	0001_3042h	18.7.1/905
20C_8008	Analog ARM PLL control Register (CCM_ANALOG_PLL_ARM_CLR)	32	R/W	0001_3042h	18.7.1/905
20C_800C	Analog ARM PLL control Register (CCM_ANALOG_PLL_ARM_TOG)	32	R/W	0001_3042h	18.7.1/905
20C_8010	Analog USB1 480MHz PLL Control Register (CCM_ANALOG_PLL_USB1)	32	R/W	0001_2000h	18.7.2/907
20C_8014	Analog USB1 480MHz PLL Control Register (CCM_ANALOG_PLL_USB1_SET)	32	R/W	0001_2000h	18.7.2/907
20C_8018	Analog USB1 480MHz PLL Control Register (CCM_ANALOG_PLL_USB1_CLR)	32	R/W	0001_2000h	18.7.2/907
20C_801C	Analog USB1 480MHz PLL Control Register (CCM_ANALOG_PLL_USB1_TOG)	32	R/W	0001_2000h	18.7.2/907
20C_8020	Analog USB2 480MHz PLL Control Register (CCM_ANALOG_PLL_USB2)	32	R/W	0001_2000h	18.7.3/909
20C_8024	Analog USB2 480MHz PLL Control Register (CCM_ANALOG_PLL_USB2_SET)	32	R/W	0001_2000h	18.7.3/909
20C_8028	Analog USB2 480MHz PLL Control Register (CCM_ANALOG_PLL_USB2_CLR)	32	R/W	0001_2000h	18.7.3/909
20C_802C	Analog USB2 480MHz PLL Control Register (CCM_ANALOG_PLL_USB2_TOG)	32	R/W	0001_2000h	18.7.3/909
20C_8030	Analog System PLL Control Register (CCM_ANALOG_PLL_SYS)	32	R/W	0001_3001h	18.7.4/911
20C_8034	Analog System PLL Control Register (CCM_ANALOG_PLL_SYS_SET)	32	R/W	0001_3001h	18.7.4/911
20C_8038	Analog System PLL Control Register (CCM_ANALOG_PLL_SYS_CLR)	32	R/W	0001_3001h	18.7.4/911

Table continues on the next page...

CCM_ANALOG memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20C_803C	Analog System PLL Control Register (CCM_ANALOG_PLL_SYS_TOG)	32	R/W	0001_3001h	18.7.4/911
20C_8040	528MHz System PLL Spread Spectrum Register (CCM_ANALOG_PLL_SYS_SS)	32	R/W	0000_0000h	18.7.5/913
20C_8050	Numerator of 528MHz System PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_SYS_NUM)	32	R/W	0000_0000h	18.7.6/913
20C_8060	Denominator of 528MHz System PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_SYS_DENOM)	32	R/W	0000_0012h	18.7.7/914
20C_8070	Analog Audio PLL control Register (CCM_ANALOG_PLL_AUDIO)	32	R/W	0001_1006h	18.7.8/915
20C_8074	Analog Audio PLL control Register (CCM_ANALOG_PLL_AUDIO_SET)	32	R/W	0001_1006h	18.7.8/915
20C_8078	Analog Audio PLL control Register (CCM_ANALOG_PLL_AUDIO_CLR)	32	R/W	0001_1006h	18.7.8/915
20C_807C	Analog Audio PLL control Register (CCM_ANALOG_PLL_AUDIO_TOG)	32	R/W	0001_1006h	18.7.8/915
20C_8080	Numerator of Audio PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_AUDIO_NUM)	32	R/W	05F5_E100h	18.7.9/917
20C_8090	Denominator of Audio PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_AUDIO_DENOM)	32	R/W	2964_619Ch	18.7.10/ 918
20C_80A0	Analog Video PLL control Register (CCM_ANALOG_PLL_VIDEO)	32	R/W	0001_100Ch	18.7.11/ 919
20C_80A4	Analog Video PLL control Register (CCM_ANALOG_PLL_VIDEO_SET)	32	R/W	0001_100Ch	18.7.11/ 919
20C_80A8	Analog Video PLL control Register (CCM_ANALOG_PLL_VIDEO_CLR)	32	R/W	0001_100Ch	18.7.11/ 919
20C_80AC	Analog Video PLL control Register (CCM_ANALOG_PLL_VIDEO_TOG)	32	R/W	0001_100Ch	18.7.11/ 919
20C_80B0	Numerator of Video PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_VIDEO_NUM)	32	R/W	05F5_E100h	18.7.12/ 921
20C_80C0	Denominator of Video PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_VIDEO_DENOM)	32	R/W	10A2_4447h	18.7.13/ 922
20C_80D0	MLB PLL Control Register (CCM_ANALOG_PLL_MLB)	32	R/W	0001_0000h	18.7.14/ 923
20C_80D4	MLB PLL Control Register (CCM_ANALOG_PLL_MLB_SET)	32	R/W	0001_0000h	18.7.14/ 923
20C_80D8	MLB PLL Control Register (CCM_ANALOG_PLL_MLB_CLR)	32	R/W	0001_0000h	18.7.14/ 923
20C_80DC	MLB PLL Control Register (CCM_ANALOG_PLL_MLB_TOG)	32	R/W	0001_0000h	18.7.14/ 923
20C_80E0	Analog ENET PLL Control Register (CCM_ANALOG_PLL_ENET)	32	R/W	0001_1001h	18.7.15/ 925
20C_80E4	Analog ENET PLL Control Register (CCM_ANALOG_PLL_ENET_SET)	32	R/W	0001_1001h	18.7.15/ 925

Table continues on the next page...

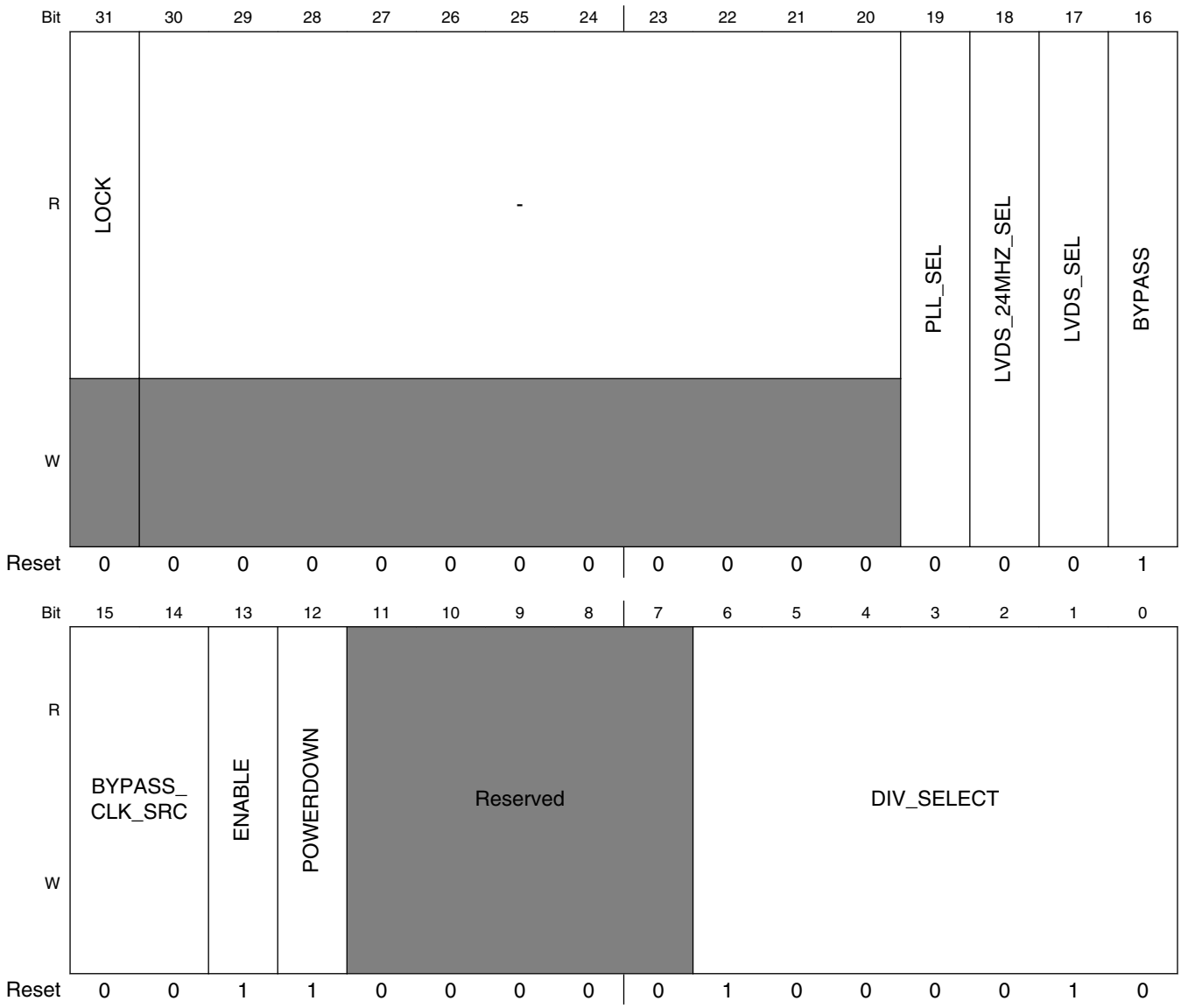
CCM_ANALOG memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20C_80E8	Analog ENET PLL Control Register (CCM_ANALOG_PLL_ENET_CLR)	32	R/W	0001_1001h	18.7.15/ 925
20C_80EC	Analog ENET PLL Control Register (CCM_ANALOG_PLL_ENET_TOG)	32	R/W	0001_1001h	18.7.15/ 925
20C_80F0	480MHz Clock (PLL3) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_480)	32	R/W	1311_100Ch	18.7.16/ 927
20C_80F4	480MHz Clock (PLL3) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_480_SET)	32	R/W	1311_100Ch	18.7.16/ 927
20C_80F8	480MHz Clock (PLL3) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_480_CLR)	32	R/W	1311_100Ch	18.7.16/ 927
20C_80FC	480MHz Clock (PLL3) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_480_TOG)	32	R/W	1311_100Ch	18.7.16/ 927
20C_8100	528MHz Clock (PLL2) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_528)	32	R/W	1018_101Bh	18.7.17/ 929
20C_8104	528MHz Clock (PLL2) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_528_SET)	32	R/W	1018_101Bh	18.7.17/ 929
20C_8108	528MHz Clock (PLL2) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_528_CLR)	32	R/W	1018_101Bh	18.7.17/ 929
20C_810C	528MHz Clock (PLL2) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_528_TOG)	32	R/W	1018_101Bh	18.7.17/ 929
20C_8150	Miscellaneous Register 0 (CCM_ANALOG_MISC0)	32	R/W	0400_0000h	18.7.18/ 932
20C_8154	Miscellaneous Register 0 (CCM_ANALOG_MISC0_SET)	32	R/W	0400_0000h	18.7.18/ 932
20C_8158	Miscellaneous Register 0 (CCM_ANALOG_MISC0_CLR)	32	R/W	0400_0000h	18.7.18/ 932
20C_815C	Miscellaneous Register 0 (CCM_ANALOG_MISC0_TOG)	32	R/W	0400_0000h	18.7.18/ 932
20C_8160	Miscellaneous Register 1 (CCM_ANALOG_MISC1)	32	R/W	0000_0000h	18.7.19/ 935
20C_8164	Miscellaneous Register 1 (CCM_ANALOG_MISC1_SET)	32	R/W	0000_0000h	18.7.19/ 935
20C_8168	Miscellaneous Register 1 (CCM_ANALOG_MISC1_CLR)	32	R/W	0000_0000h	18.7.19/ 935
20C_816C	Miscellaneous Register 1 (CCM_ANALOG_MISC1_TOG)	32	R/W	0000_0000h	18.7.19/ 935
20C_8170	Miscellaneous Register 2 (CCM_ANALOG_MISC2)	32	R/W	0027_2727h	18.7.20/ 938
20C_8174	Miscellaneous Register 2 (CCM_ANALOG_MISC2_SET)	32	R/W	0027_2727h	18.7.20/ 938
20C_8178	Miscellaneous Register 2 (CCM_ANALOG_MISC2_CLR)	32	R/W	0027_2727h	18.7.20/ 938
20C_817C	Miscellaneous Register 2 (CCM_ANALOG_MISC2_TOG)	32	R/W	0027_2727h	18.7.20/ 938

18.7.1 Analog ARM PLL control Register (CCM_ANALOG_PLL_ARM*n*)

The control register provides control for the system PLL.

Address: 20C_8000h base + 0h offset + (4d × i), where i=0d to 3d



CCM_ANALOG_PLL_ARM*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked. 0 - PLL is not currently locked.

Table continues on the next page...

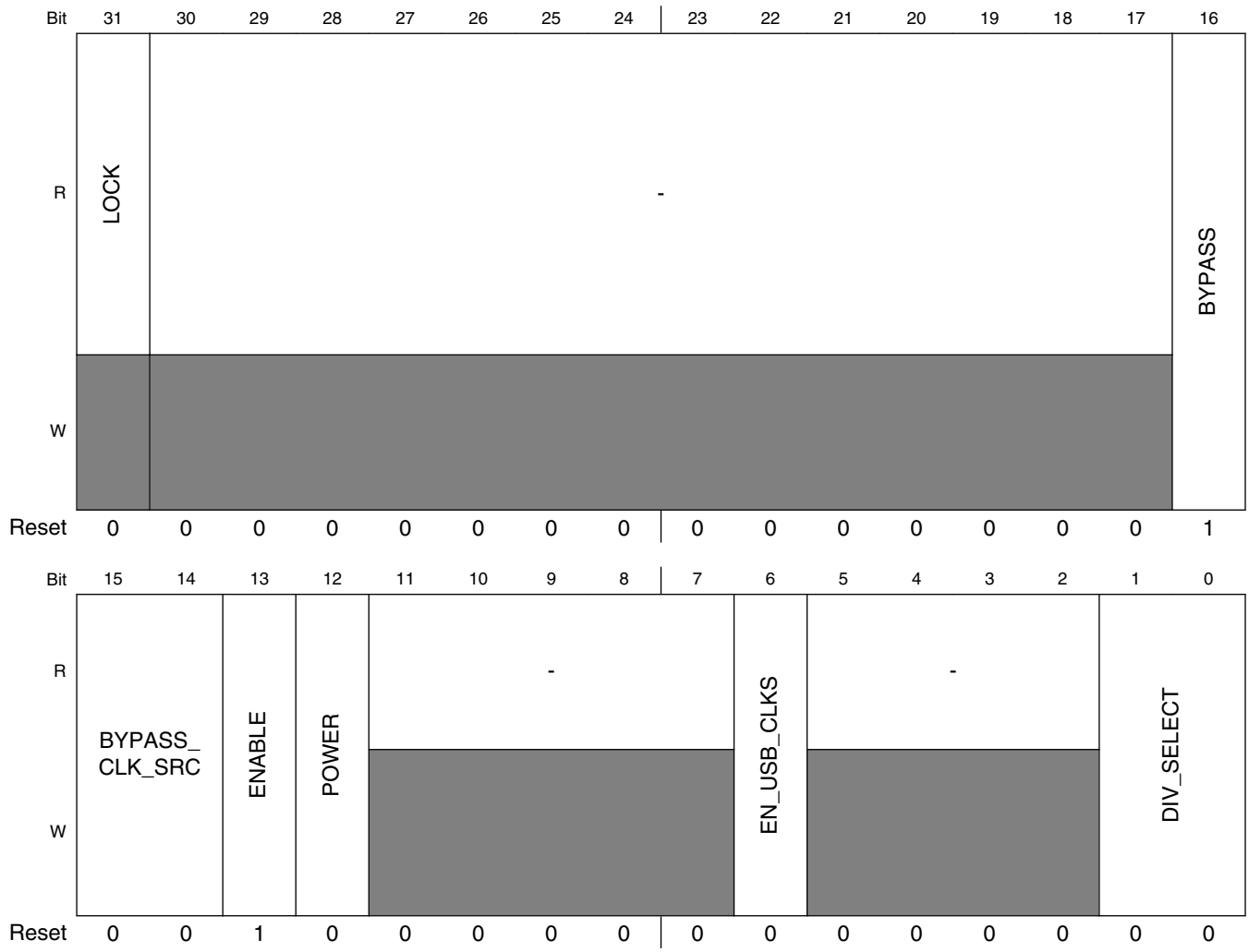
CCM_ANALOG_PLL_ARM*n* field descriptions (continued)

Field	Description
30–20 -	Always set to zero (0).
19 PLL_SEL	Reserved
18 LVDS_24MHZ_SEL	Analog Debug Bit
17 LVDS_SEL	Analog Debug Bit
16 BYPASS	Bypass the PLL.
15–14 BYPASS_CLK_SRC	Determines the bypass source. **NOTE:** Changing the Bypass clock source also changes the PLL reference clock source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.
13 ENABLE	Enable the clock output.
12 POWERDOWN	Powers down the PLL.
11–7 -	This field is reserved. Reserved.
DIV_SELECT	This field controls the PLL loop divider. Valid range for divider value: 54-108. Fout = Fin * div_select/2.0.

18.7.2 Analog USB1 480MHz PLL Control Register (CCM_ANALOG_PLL_USB1*n*)

The control register provides control for USBPHY0 480MHz PLL.

Address: 20C_8000h base + 10h offset + (4d × i), where i=0d to 3d



CCM_ANALOG_PLL_USB1*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked. 0 - PLL is not currently locked.
30–17 -	Always set to zero (0).
16 BYPASS	Bypass the PLL.

Table continues on the next page...

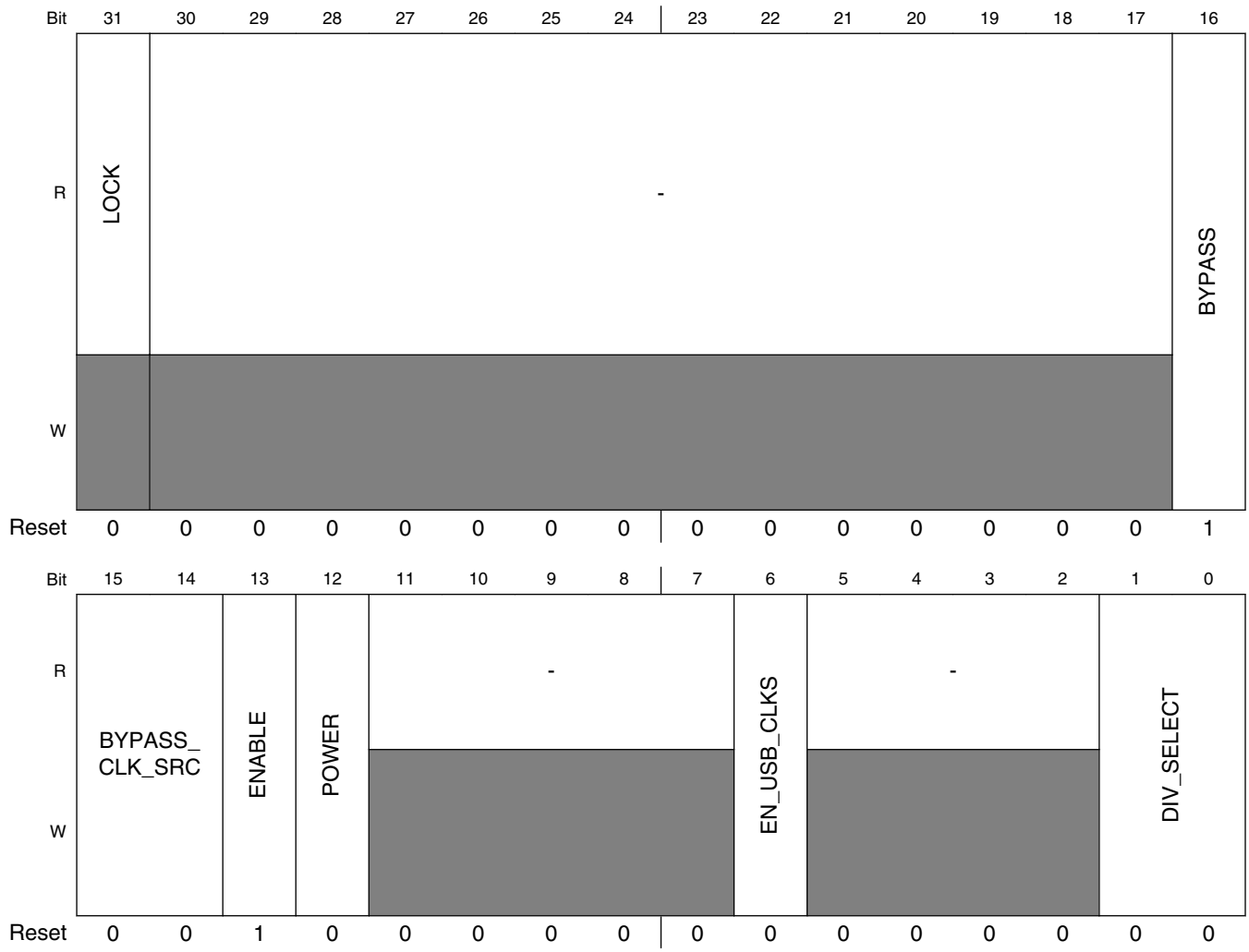
CCM_ANALOG_PLL_USB1*n* field descriptions (continued)

Field	Description
15–14 BYPASS_CLK_SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.
13 ENABLE	Enable the PLL clock output.
12 POWER	Powers up the PLL. This bit will be set automatically when USBPHY0 remote wakeup event happens.
11–7 -	Always set to zero (0).
6 EN_USB_CLKS	Powers the 9-phase PLL outputs for USBPHYn. Additionally, the UTMI clock gate must be deasserted in the USBPHYn to enable USBn operation (clear CLKGATE bit in USBPHYn_CTRL). This bit will be set automatically when USBPHYn remote wakeup event occurs. 0 PLL outputs for USBPHYn off. 1 PLL outputs for USBPHYn on.
5–2 -	Always set to zero (0).
DIV_SELECT	This field controls the PLL loop divider. 0 - Fout=Fref*20; 1 - Fout=Fref*22.

18.7.3 Analog USB2 480MHz PLL Control Register (CCM_ANALOG_PLL_USB2*n*)

The control register provides control for USBPHY1 480MHz PLL.

Address: 20C_8000h base + 20h offset + (4d × i), where i=0d to 3d



CCM_ANALOG_PLL_USB2*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked. 0 - PLL is not currently locked.
30–17 -	Always set to zero (0).
16 BYPASS	Bypass the PLL.

Table continues on the next page...

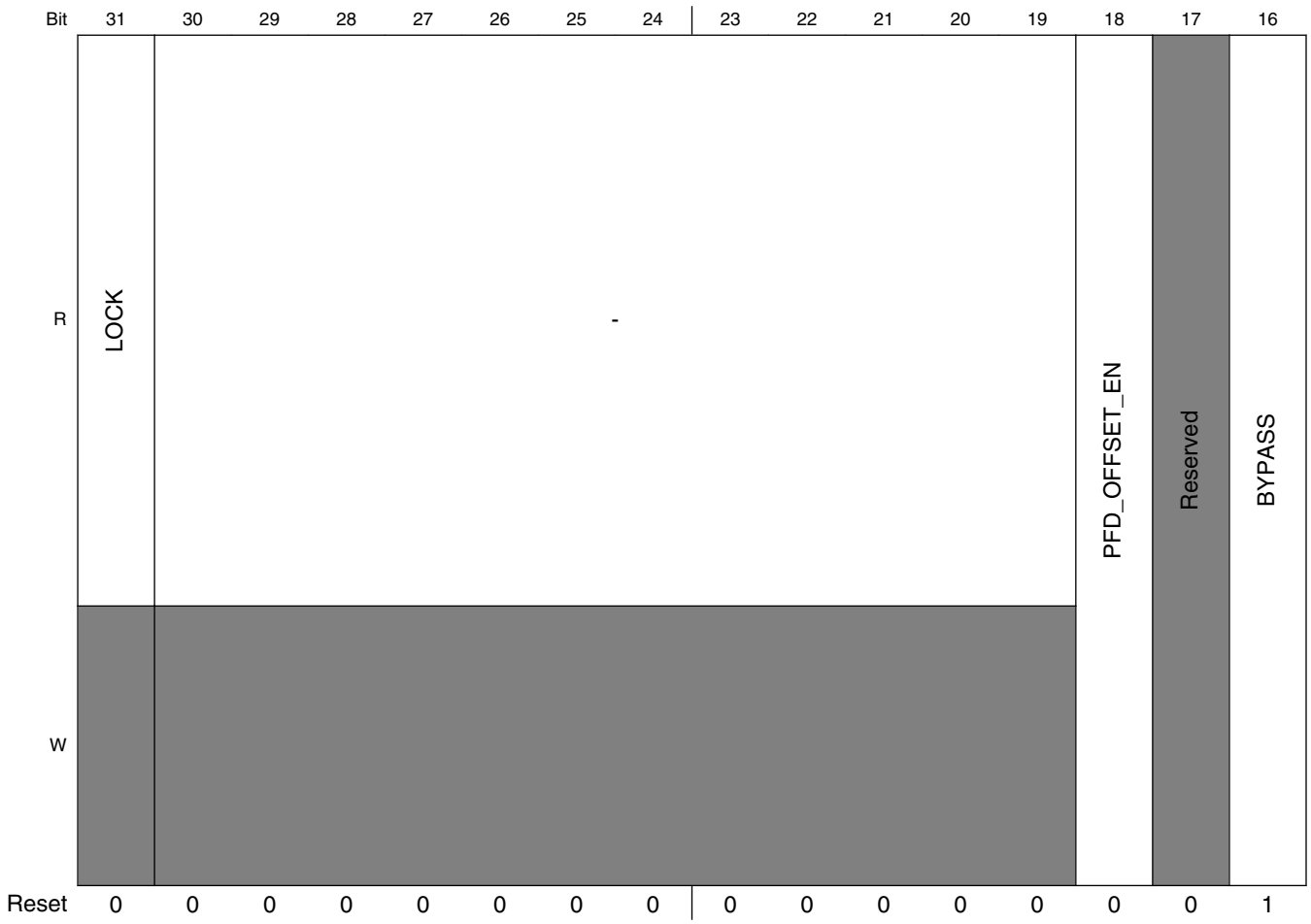
CCM_ANALOG_PLL_USB2*n* field descriptions (continued)

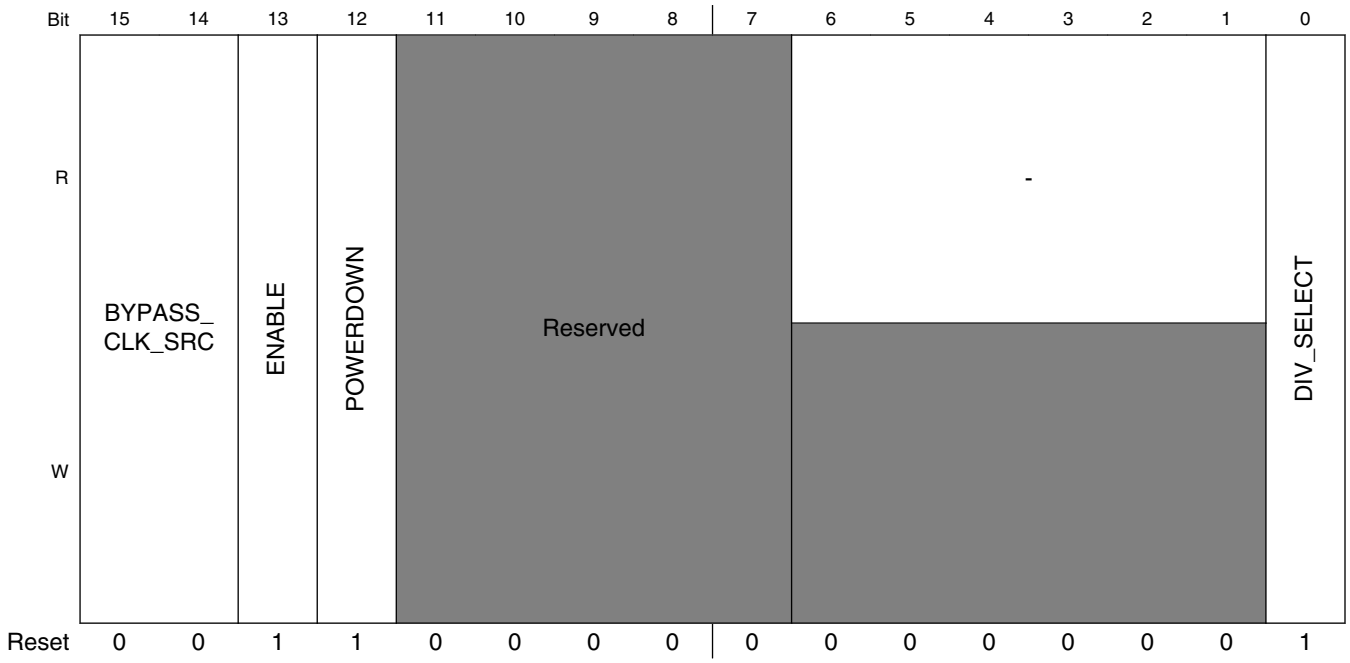
Field	Description
15–14 BYPASS_CLK_SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.
13 ENABLE	Enable the PLL clock output.
12 POWER	Powers up the PLL. This bit will be set automatically when USBPHY1 remote wakeup event happens.
11–7 -	Always set to zero (0).
6 EN_USB_CLKS	0: 8-phase PLL outputs for USBPHY1 are powered down. If set to 1, 8-phase PLL outputs for USBPHY1 are powered up. Additionally, the utmi clock gate must be deasserted in the USBPHY1 to enable USB0 operation (clear CLKGATE bit in USBPHY1_CTRL).This bit will be set automatically when USBPHY1 remote wakeup event happens.
5–2 -	Always set to zero (0).
DIV_SELECT	This field controls the PLL loop divider. 0 - Fout=Fref*20; 1 - Fout=Fref*22.

18.7.4 Analog System PLL Control Register (CCM_ANALOG_PLL_SYS*n*)

The control register provides control for the 528MHz PLL.

Address: 20C_8000h base + 30h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PLL_SYS*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked; 0 - PLL is not currently locked.
30–19 -	Always set to zero (0).
18 PFD_OFFSET_ EN	Enables an offset in the phase frequency detector.
17 -	This field is reserved. Reserved
16 BYPASS	Bypass the PLL.
15–14 BYPASS_CLK_ SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.
13 ENABLE	Enable PLL output
12 POWERDOWN	Powers down the PLL.
11–7 -	This field is reserved. Reserved.
6–1 -	Always set to zero (0).
0 DIV_SELECT	This field controls the PLL loop divider. 0 - Fout=Fref*20; 1 - Fout=Fref*22.

18.7.5 528MHz System PLL Spread Spectrum Register (CCM_ANALOG_PLL_SYS_SS)

This register contains the 528 PLL spread spectrum controls.

Address: 20C_8000h base + 40h offset = 20C_8040h

CCM_ANALOG_PLL_SYS_SS field descriptions

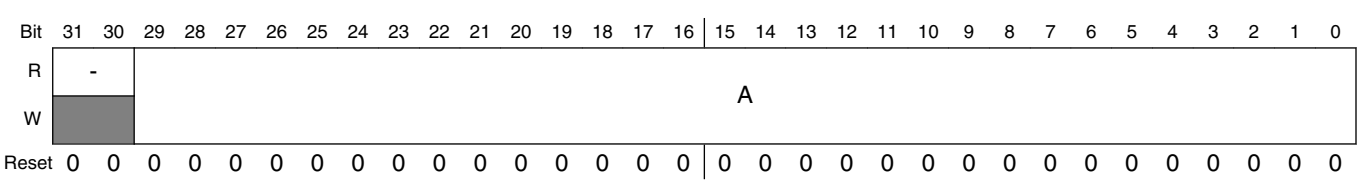
Field	Description
31–16 STOP	Frequency change = stop/CCM_ANALOG_PLL_SYS_DENOM[B]*24MHz.
15 ENABLE	0 — Spread spectrum modulation disabled 1 — Soread spectrum modulation enabled
STEP	Frequency change step = step/CCM_ANALOG_PLL_SYS_DENOM[B]*24MHz.

18.7.6 Numerator of 528MHz System PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_SYS_NUM)

This register contains the numerator of 528MHz PLL fractional loop divider (signed number).

Absoulte value should be less than denominator

Address: 20C_8000h base + 50h offset = 20C_8050h



CCM_ANALOG_PLL_SYS_NUM field descriptions

Field	Description
31–30 -	Always set to zero (0).
A	30 bit numerator (A) of fractional loop divider (signed integer).

18.7.7 Denominator of 528MHz System PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_SYS_DENOM)

This register contains the Denominator of 528MHz PLL fractional loop divider.

Address: 20C_8000h base + 60h offset = 20C_8060h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	-																B															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0

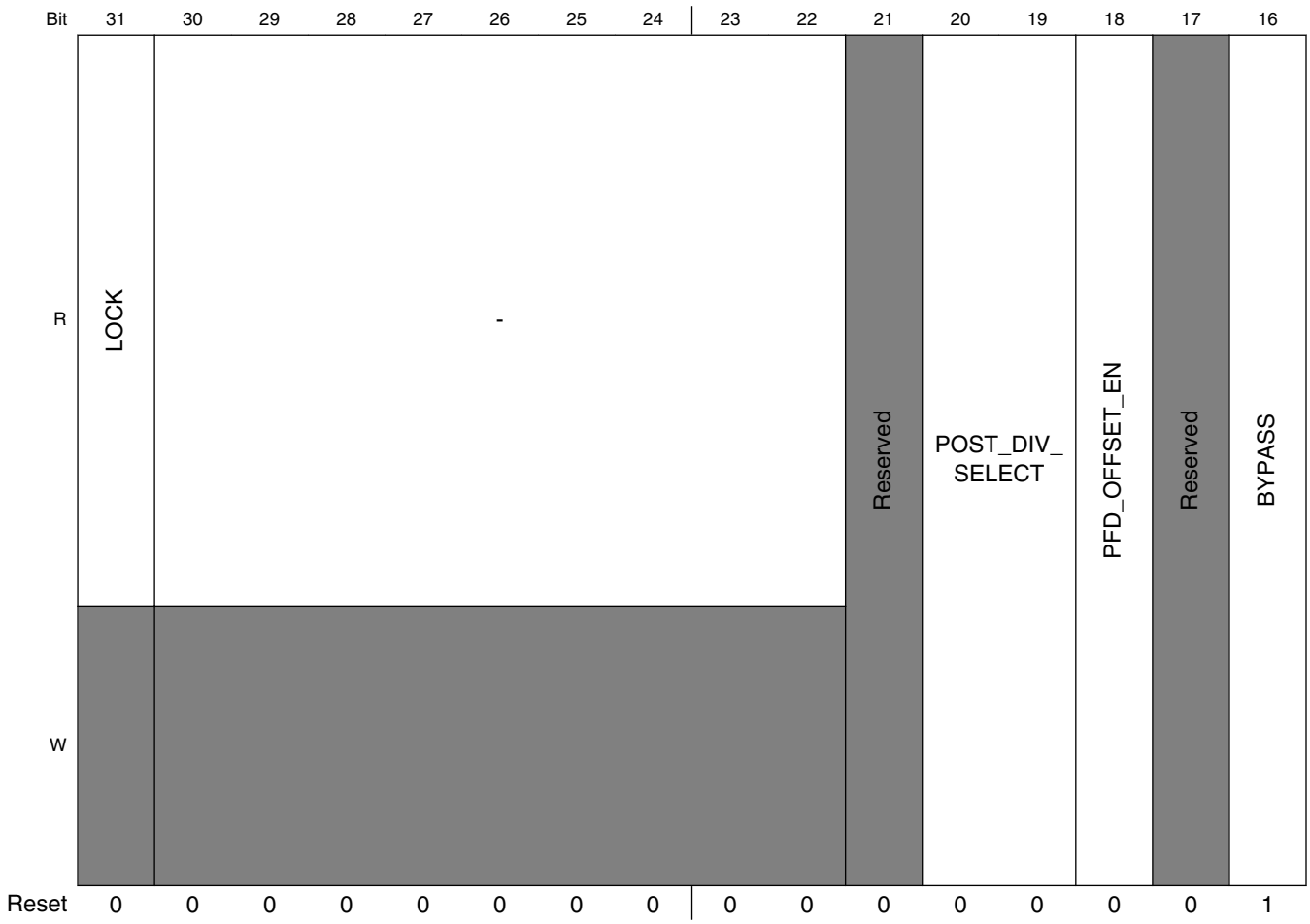
CCM_ANALOG_PLL_SYS_DENOM field descriptions

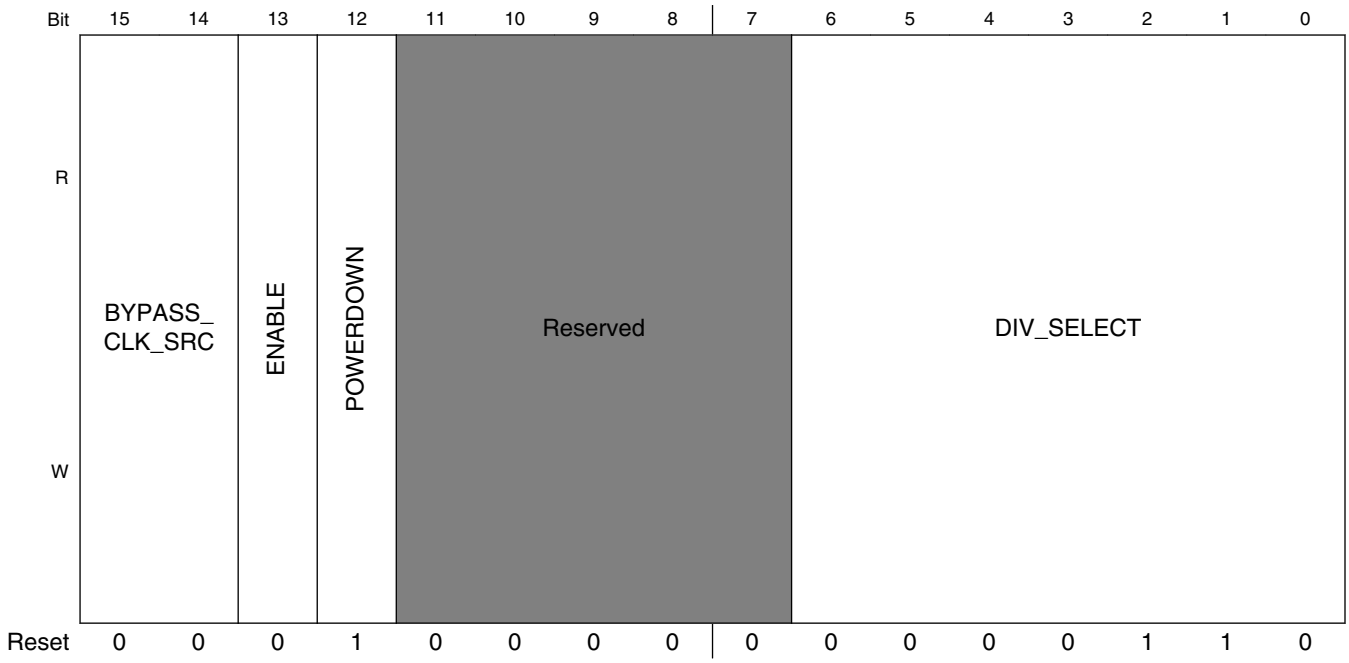
Field	Description
31–30 -	Always set to zero (0).
B	30 bit Denominator (B) of fractional loop divider (unsigned integer).

18.7.8 Analog Audio PLL control Register (CCM_ANALOG_PLL_AUDIO*n*)

The control register provides control for the audio PLL.

Address: 20C_8000h base + 70h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PLL_AUDIO*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked. 0 - PLL is not currently locked.
30–22 -	Always set to zero (0).
21 -	This field is reserved. Reserved
20–19 POST_DIV_ SELECT	These bits implement a divider after the PLL, but before the enable and bypass mux. 00 — Divide by 4. 01 — Divide by 2. 10 — Divide by 1. 11 — Reserved
18 PFD_OFFSET_ EN	Enables an offset in the phase frequency detector.
17 -	This field is reserved. Revsered
16 BYPASS	Bypass the PLL.
15–14 BYPASS_CLK_ SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.

Table continues on the next page...

CCM_ANALOG_PLL_AUDIO*n* field descriptions (continued)

Field	Description
13 ENABLE	Enable PLL output
12 POWERDOWN	Powers down the PLL.
11–7 -	This field is reserved. Reserved.
DIV_SELECT	This field controls the PLL loop divider. Valid range for DIV_SELECT divider value: 27~54.

18.7.9 Numerator of Audio PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_AUDIO_NUM)

This register contains the numerator (A) of Audio PLL fractional loop divider.(Signed number), absolute value should be less than denominator

Absolute value should be less than denominator

Address: 20C_8000h base + 80h offset = 20C_8080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	-																A															
W																																
Reset	0	0	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0

CCM_ANALOG_PLL_AUDIO_NUM field descriptions

Field	Description
31–30 -	Always set to zero (0).
A	30 bit numerator of fractional loop divider.

18.7.10 Denominator of Audio PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_AUDIO_DENOM)

This register contains the Denominator (B) of Audio PLL fractional loop divider. (unsigned number)

Address: 20C_8000h base + 90h offset = 20C_8090h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	-																B															
W																																
Reset	0	0	1	0	1	0	0	1	0	1	1	0	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	1	1	1	0	0

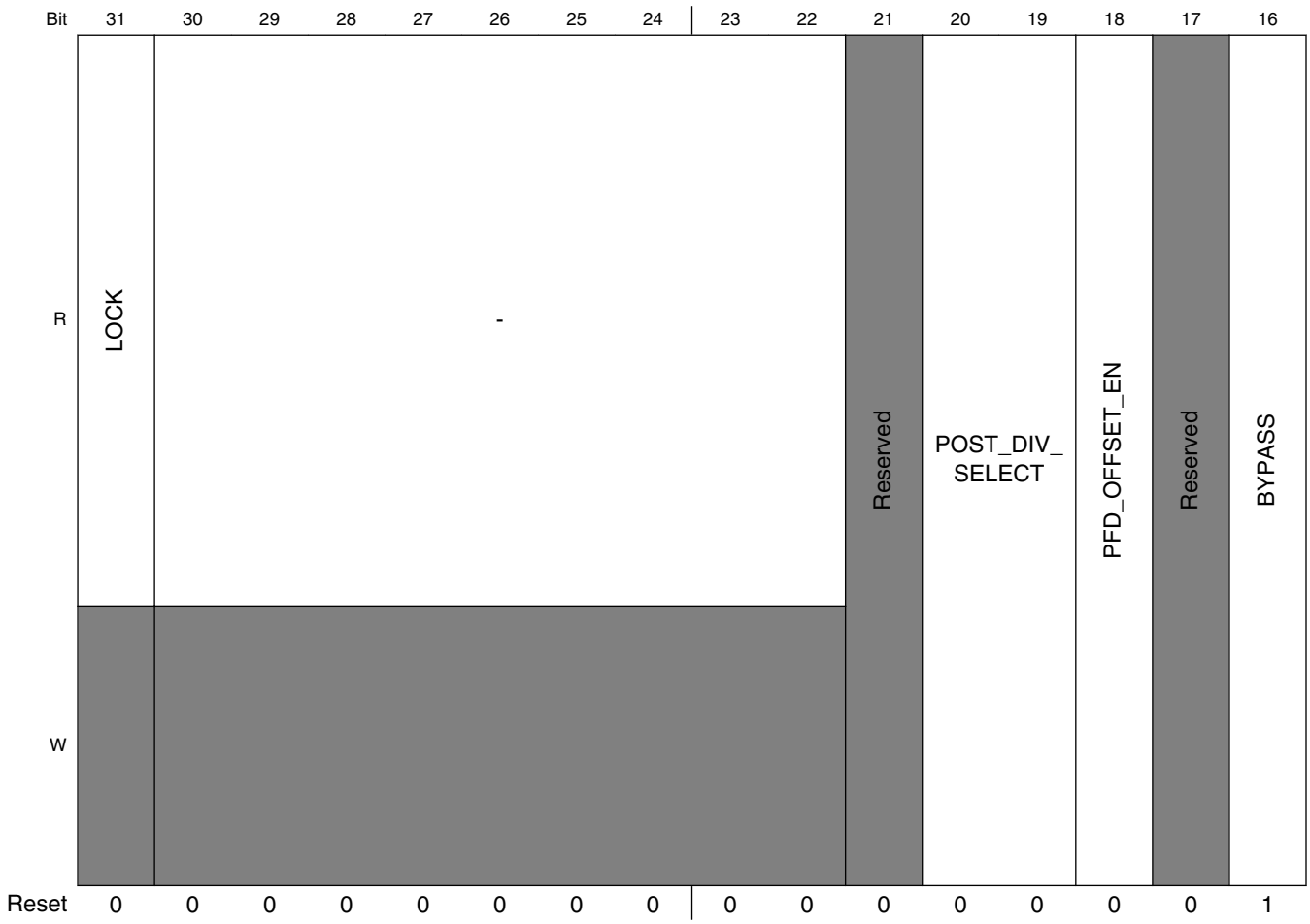
CCM_ANALOG_PLL_AUDIO_DENOM field descriptions

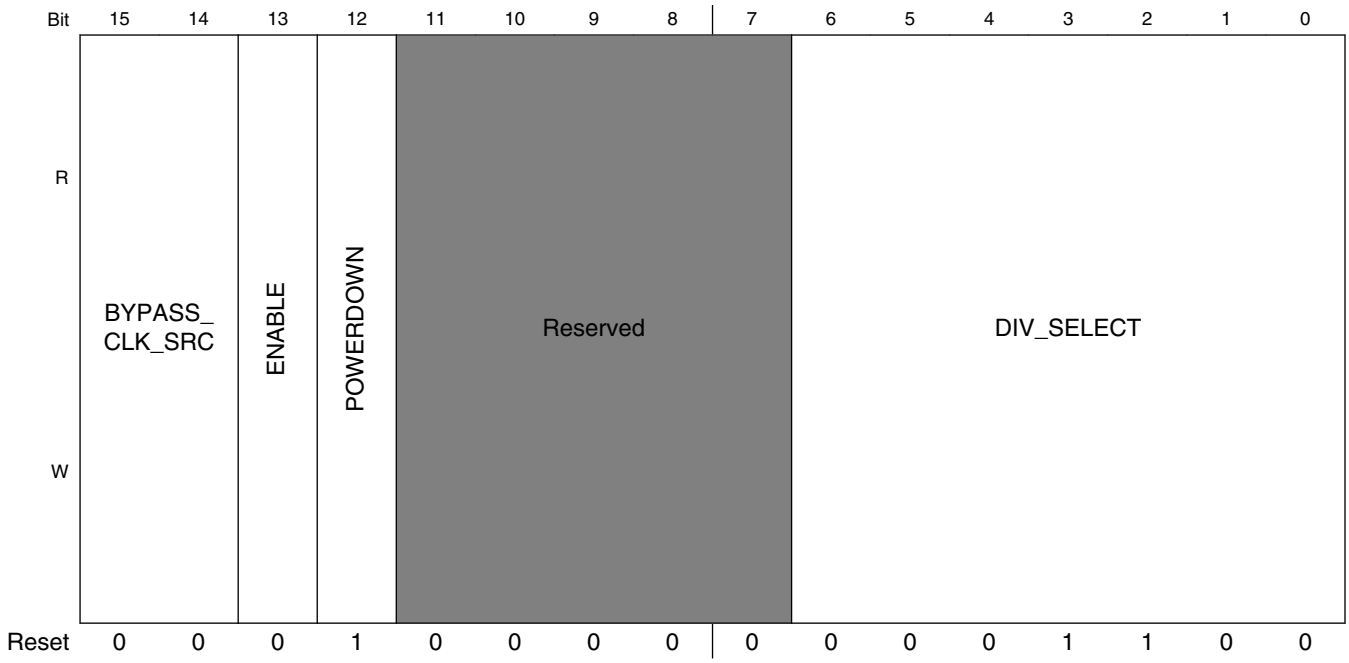
Field	Description
31–30 -	Always set to zero (0).
B	30 bit Denominator of fractional loop divider.

18.7.11 Analog Video PLL control Register (CCM_ANALOG_PLL_VIDEO*n*)

The control register provides control for the Video PLL.

Address: 20C_8000h base + A0h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PLL_VIDEO*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked; 0 - PLL is not currently locked.
30–22 -	Always set to zero (0).
21 -	This field is reserved. Revserved
20–19 POST_DIV_ SELECT	These bits implement a divider after the PLL, but before the enable and bypass mux. 00 — Divide by 4. 01 — Divide by 2. 10 — Divide by 1. 11 — Reserved
18 PFD_OFFSET_ EN	Enables an offset in the phase frequency detector.
17 -	This field is reserved. Reserved
16 BYPASS	Bypass the PLL.
15–14 BYPASS_CLK_ SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.

Table continues on the next page...

CCM_ANALOG_PLL_VIDEO*n* field descriptions (continued)

Field	Description
13 ENABLE	Enalbe PLL output
12 POWERDOWN	Powers down the PLL.
11–7 -	This field is reserved. Reserved.
DIV_SELECT	This field controls the PLL loop divider. Valid range for DIV_SELECT divider value: 27~54.

18.7.12 Numerator of Video PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_VIDEO_NUM)

This register contains the numerator (A) of Video PLL fractional loop divider.(Signed number)

Absolute value should be less than denominator

Address: 20C_8000h base + B0h offset = 20C_80B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	-																A															
W																																
Reset	0	0	0	0	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0

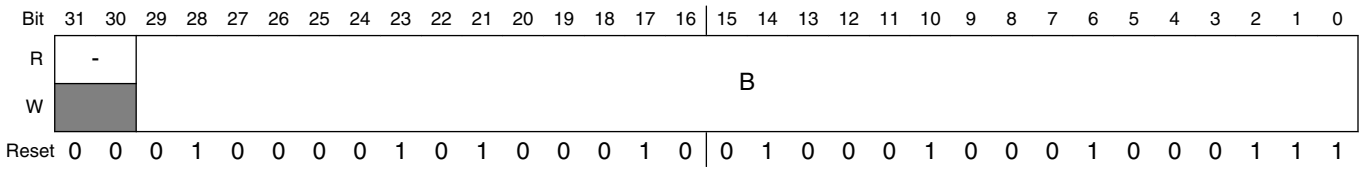
CCM_ANALOG_PLL_VIDEO_NUM field descriptions

Field	Description
31–30 -	Always set to zero (0).
A	30 bit numerator of fractional loop divider(Signed number), absolute value should be less than denominator

18.7.13 Denominator of Video PLL Fractional Loop Divider Register (CCM_ANALOG_PLL_VIDEO_DENOM)

This register contains the Denominator (B) of Video PLL fractional loop divider. (Unsigned number)

Address: 20C_8000h base + C0h offset = 20C_80C0h



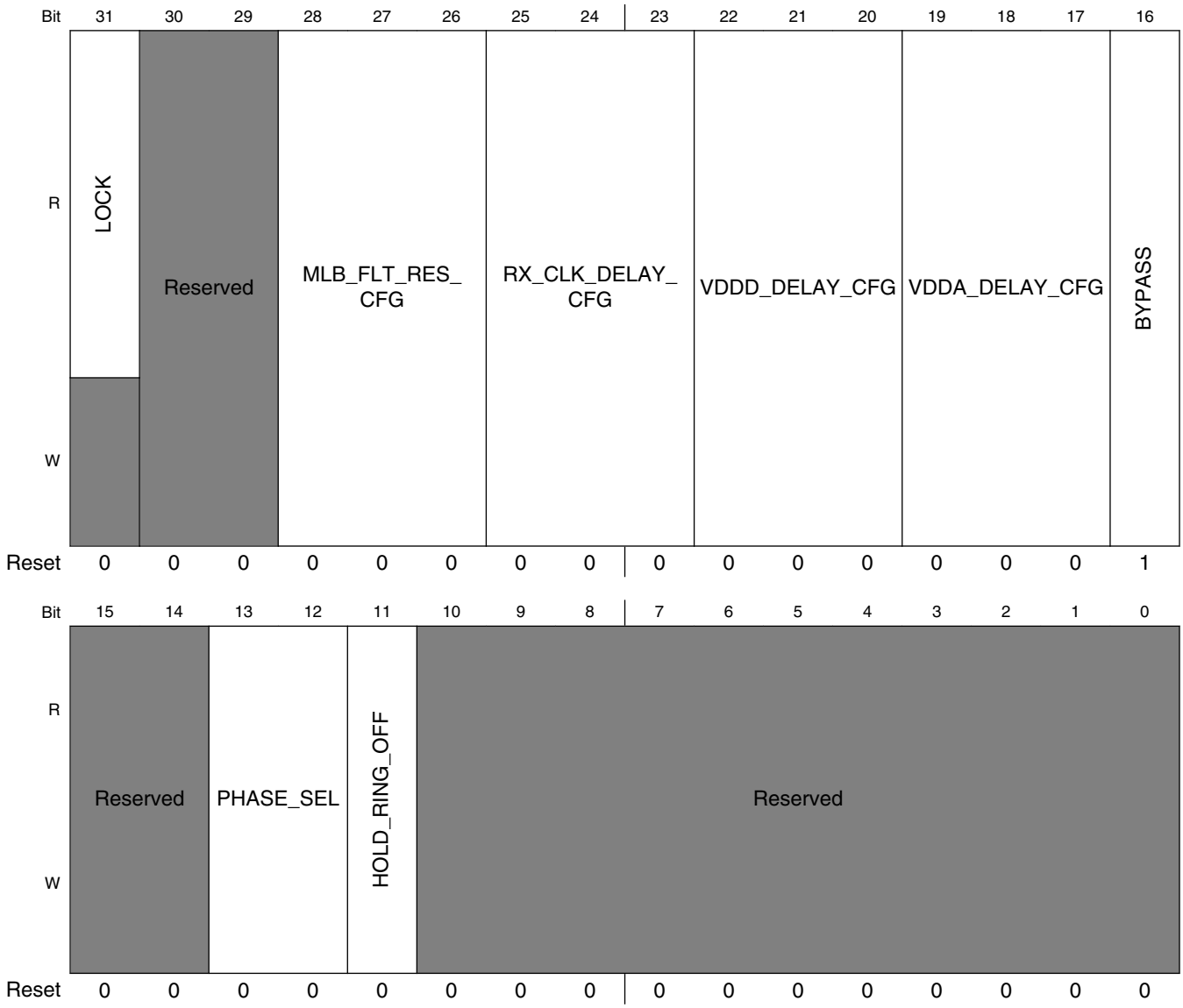
CCM_ANALOG_PLL_VIDEO_DENOM field descriptions

Field	Description
31–30 -	Always set to zero (0).
B	30 bit Denominator of fractional loop divider.

18.7.14 MLB PLL Control Register (CCM_ANALOG_PLL_MLB*n*)

This register defines the control bits for the MLB PLL.

Address: 20C_8000h base + D0h offset + (4d × i), where i=0d to 3d



CCM_ANALOG_PLL_MLB*n* field descriptions

Field	Description
31 LOCK	Lock bit 0 PLL is not currently locked 1 PLL is currently locked.

Table continues on the next page...

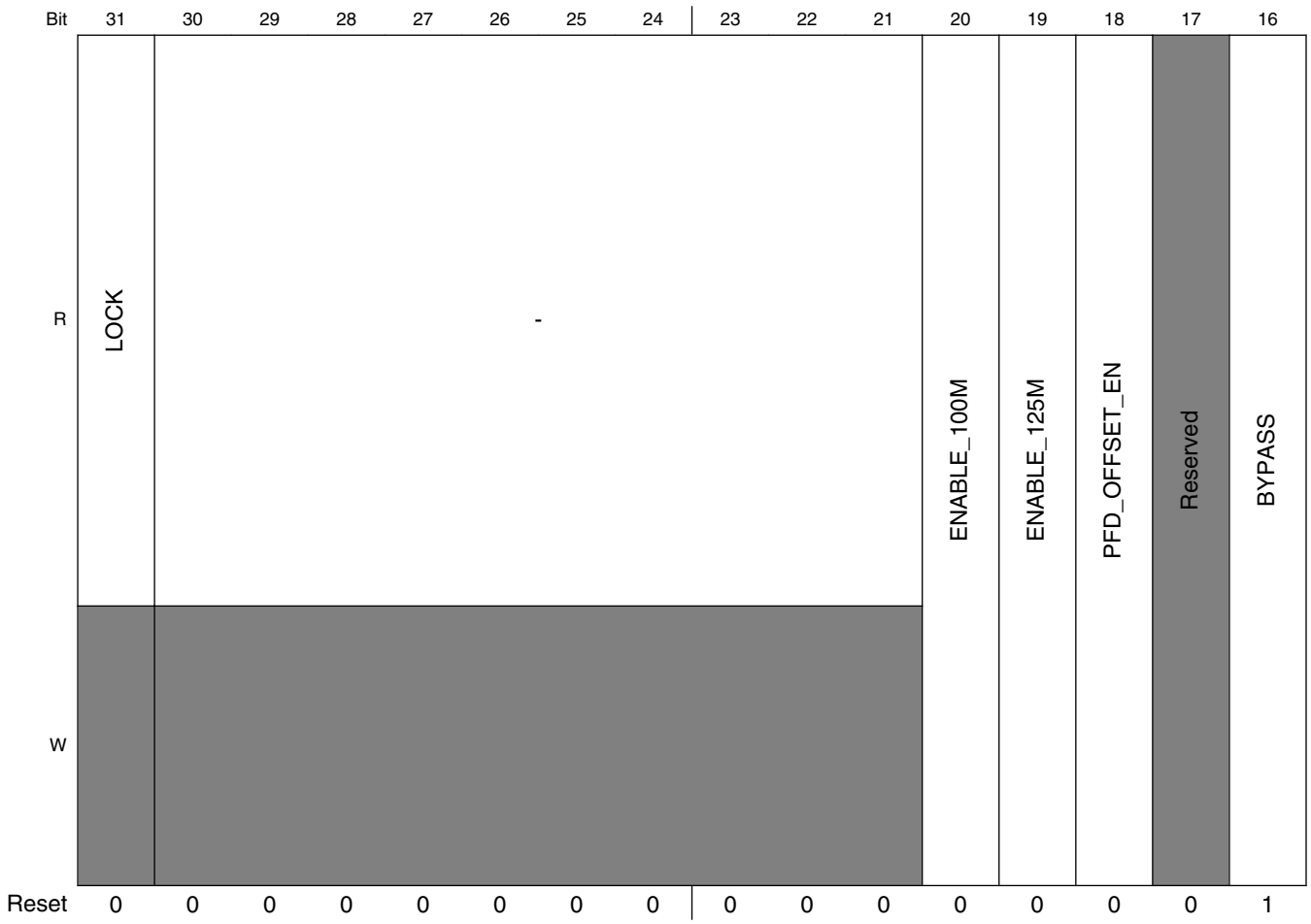
CCM_ANALOG_PLL_MLB*n* field descriptions (continued)

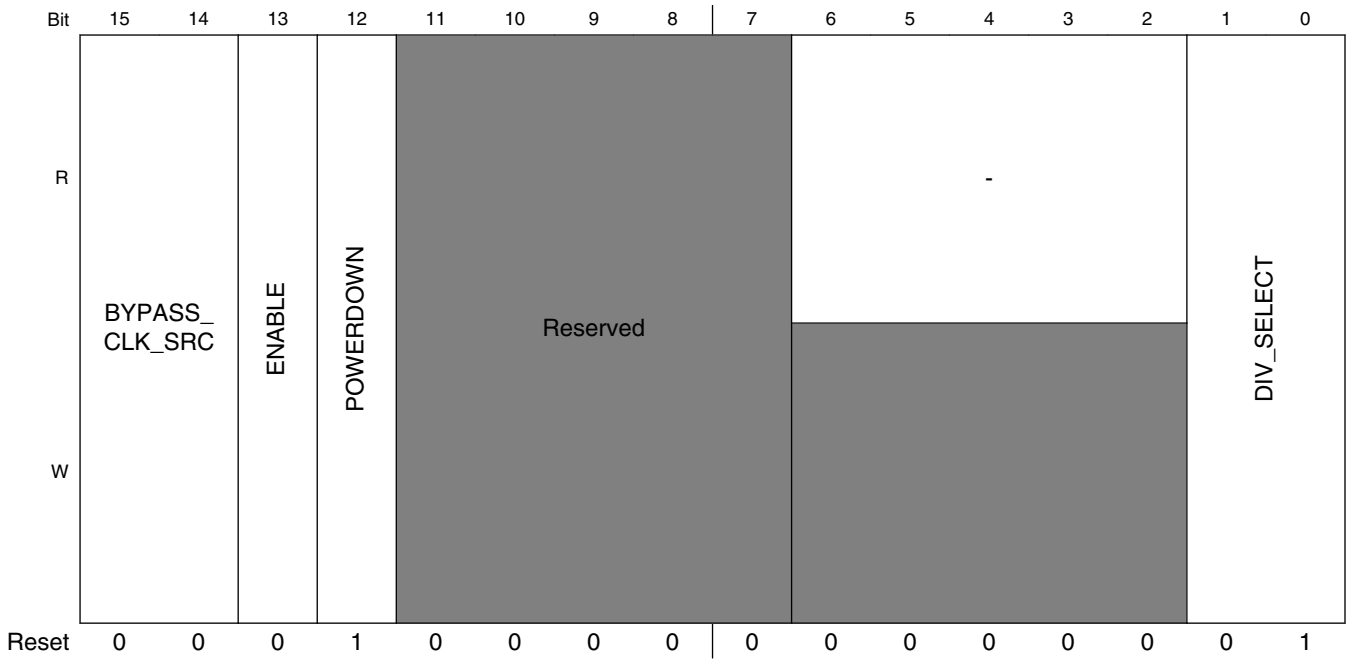
Field	Description
30–29 -	This field is reserved. Reserved.
28–26 MLB_FLT_RES_CFG	Configure the filter resistor for different divider ratio of MLB PLL.
25–23 RX_CLK_DELAY_CFG	Configure the phase delay of the MLB PLL RX Clock.
22–20 VDDD_DELAY_CFG	Configure the phase delay of the MLB PLL by adjusting the delay line in core Vdd poser domain.
19–17 VDDA_DELAY_CFG	Configure the phase delay of the MLB PLL by adjusting the delay line in Vddio power domain.
16 BYPASS	Bypass the PLL.
15–14 -	This field is reserved. Reserved.
13–12 PHASE_SEL	Analog debut bit.
11 HOLD_RING_OFF	Analog debug bit.
-	This field is reserved. Reserved.

18.7.15 Analog ENET PLL Control Register (CCM_ANALOG_PLL_ENET*n*)

The control register provides control for the ENET PLL.

Address: 20C_8000h base + E0h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PLL_ENET*n* field descriptions

Field	Description
31 LOCK	1 - PLL is currently locked; 0 - PLL is not currently locked.
30–21 -	Always set to zero (0).
20 ENABLE_100M	Enables an offset in the phase frequency detector.
19 ENABLE_125M	Enables an offset in the phase frequency detector.
18 PFD_OFFSET_EN	Enables an offset in the phase frequency detector.
17 -	This field is reserved. Reserved
16 BYPASS	Bypass the PLL.
15–14 BYPASS_CLK_SRC	Determines the bypass source. 0x0 **REF_CLK_24M** — Select the 24MHz oscillator as source. 0x1 **CLK1** — Select the CLK1_N / CLK1_P as source. 0x2 **CLK2** — Select the CLK2_N / CLK2_P as source. 0x3 **XOR** — Select the XOR of CLK1_N / CLK1_P and CLK2_N / CLK2_P as source.
13 ENABLE	Enable the ethernet clock output.
12 POWERDOWN	Powers down the PLL.
11–7 -	This field is reserved. Reserved.

Table continues on the next page...

CCM_ANALOG_PLL_ENET*n* field descriptions (continued)

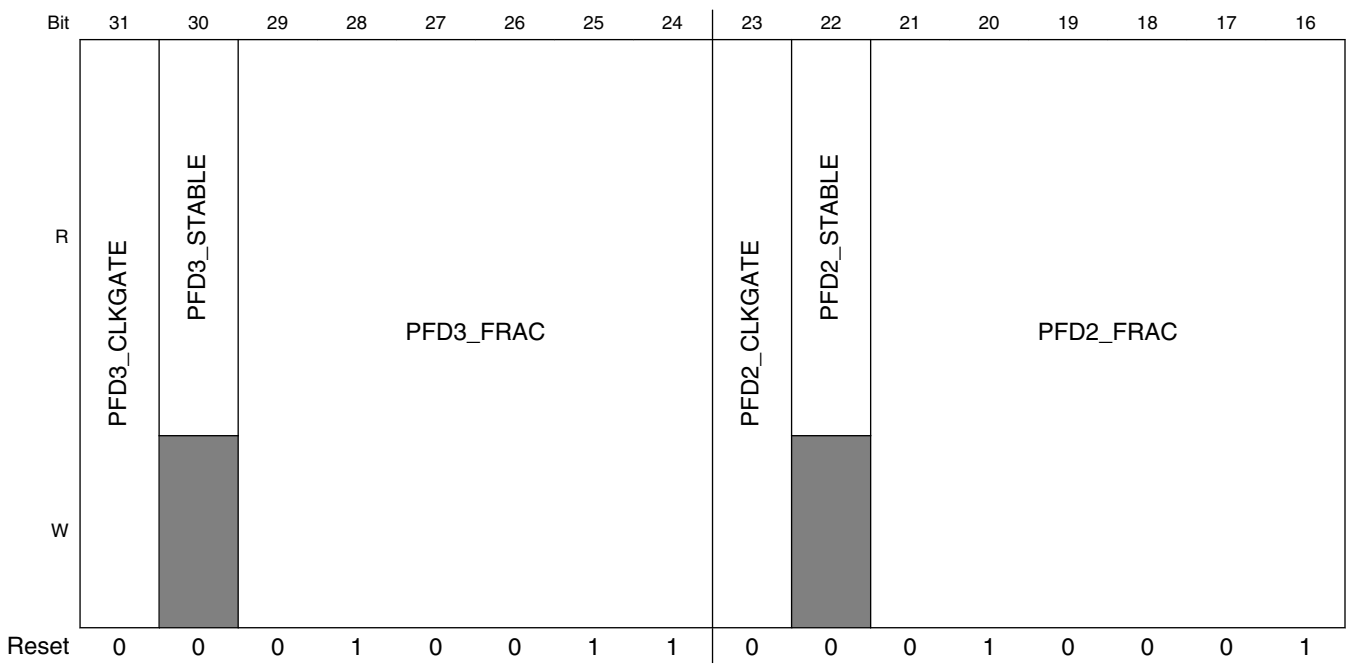
Field	Description
6–2 -	Always set to zero (0).
DIV_SELECT	Controls the frequency of the ethernet reference clock.00 - 25MHz; 01 - 50MHz; 10 - 100MHz (not 50% duty cycle); 11 - 125MHz;

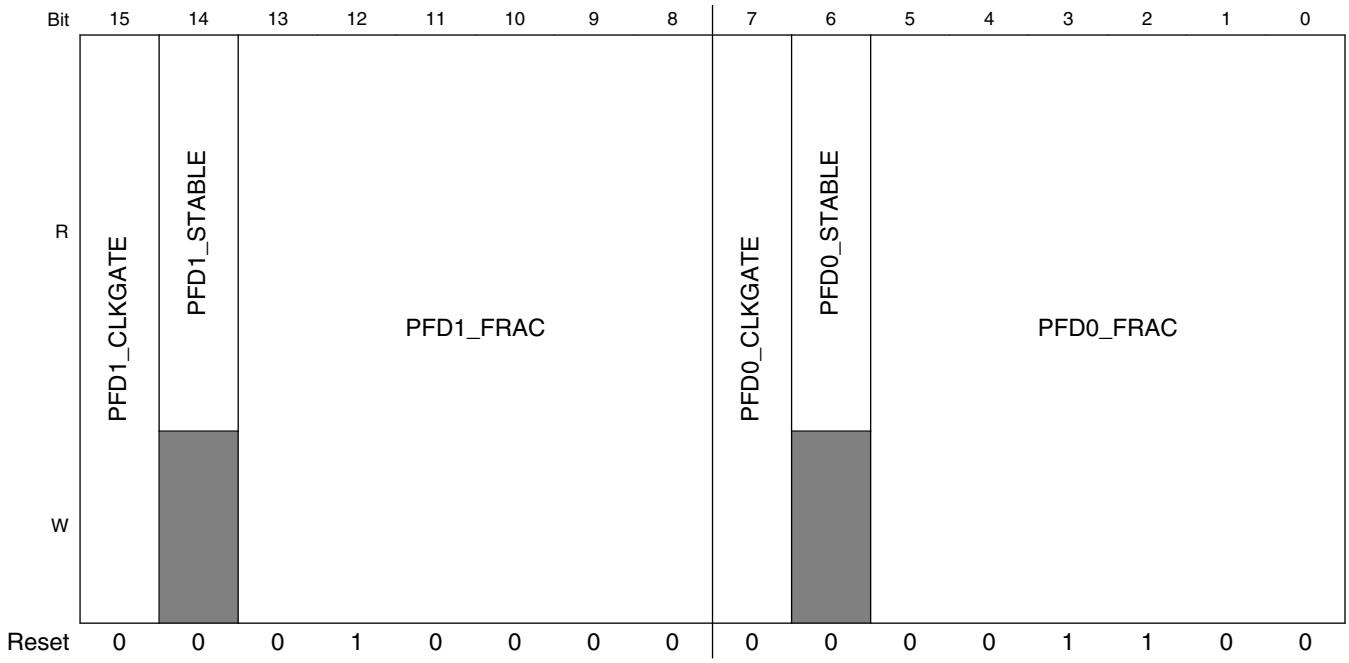
18.7.16 480MHz Clock (PLL3) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_480*n*)

The PFD_480 control register provides control for PFD clock generation.

This register controls the 4-phase fractional clock dividers. The fractional clock frequencies are a product of the values in these registers.

Address: 20C_8000h base + F0h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PFD_480*n* field descriptions

Field	Description
31 PFD3_CLKGATE	IO Clock Gate. If set to 1, the 3rd fractional divider clock (reference ref_pfd3) is off (power savings). 0: ref_pfd3 fractional divider clock is enabled. Need to assert this bit before PLL is powered down
30 PFD3_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
29–24 PFD3_FRAC	This field controls the fractional divide value. The resulting frequency shall be 480*18/PFD3_FRAC where PFD3_FRAC is in the range 12-35.
23 PFD2_CLKGATE	IO Clock Gate. If set to 1, the IO fractional divider clock (reference ref_pfd2) is off (power savings). 0: ref_pfd2 fractional divider clock is enabled. Need to assert this bit before PLL is powered down
22 PFD2_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
21–16 PFD2_FRAC	This field controls the fractional divide value. The resulting frequency shall be 480*18/PFD2_FRAC where PFD2_FRAC is in the range 12-35.
15 PFD1_CLKGATE	IO Clock Gate. If set to 1, the IO fractional divider clock (reference ref_pfd1) is off (power savings). 0: ref_pfd1 fractional divider clock is enabled. Need to assert this bit before PLL is powered down
14 PFD1_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit,

Table continues on the next page...

CCM_ANALOG_PFD_480*n* field descriptions (continued)

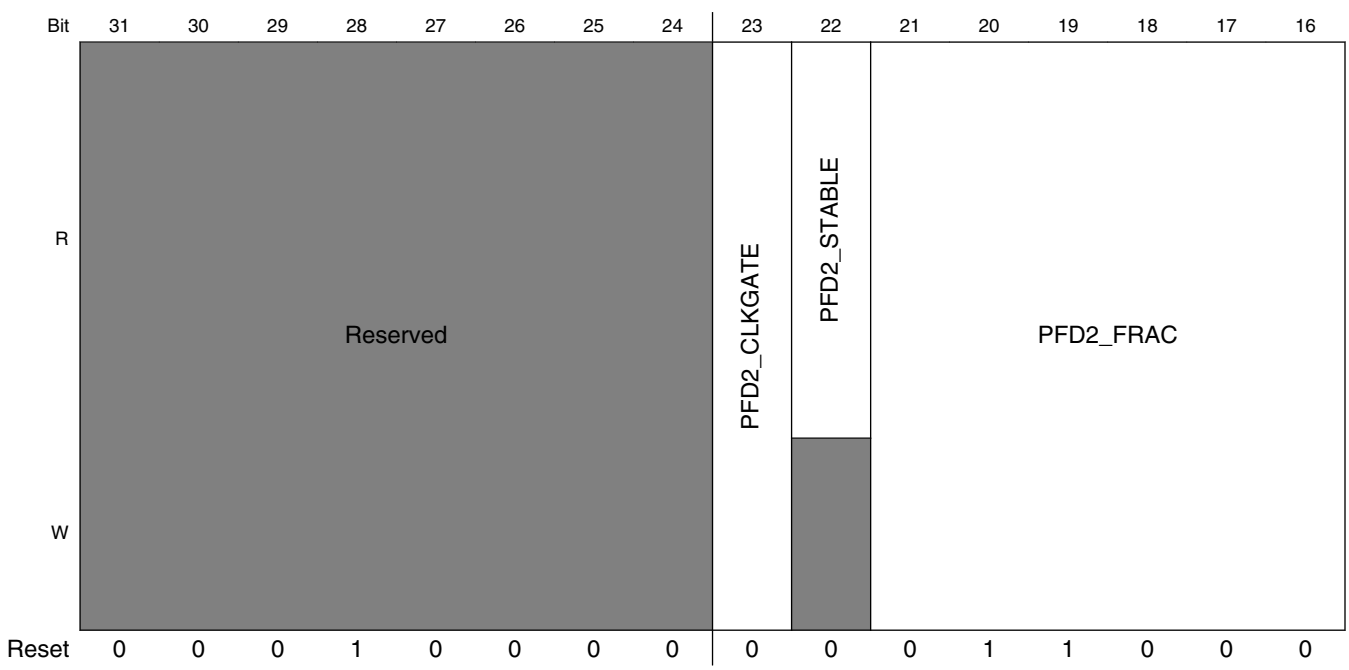
Field	Description
	program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
13–8 PFD1_FRAC	This field controls the fractional divide value. The resulting frequency shall be 480*18/PFD1_FRAC where PFD1_FRAC is in the range 12-35.
7 PFD0_CLKGATE	If set to 1, the IO fractional divider clock (reference ref_pfd0) is off (power savings). 0: ref_pfd0 fractional divider clock is enabled. Need to assert this bit before PLL is powered down
6 PFD0_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
PFD0_FRAC	This field controls the fractional divide value. The resulting frequency shall be 480*18/PFD0_FRAC where PFD0_FRAC is in the range 12-35.

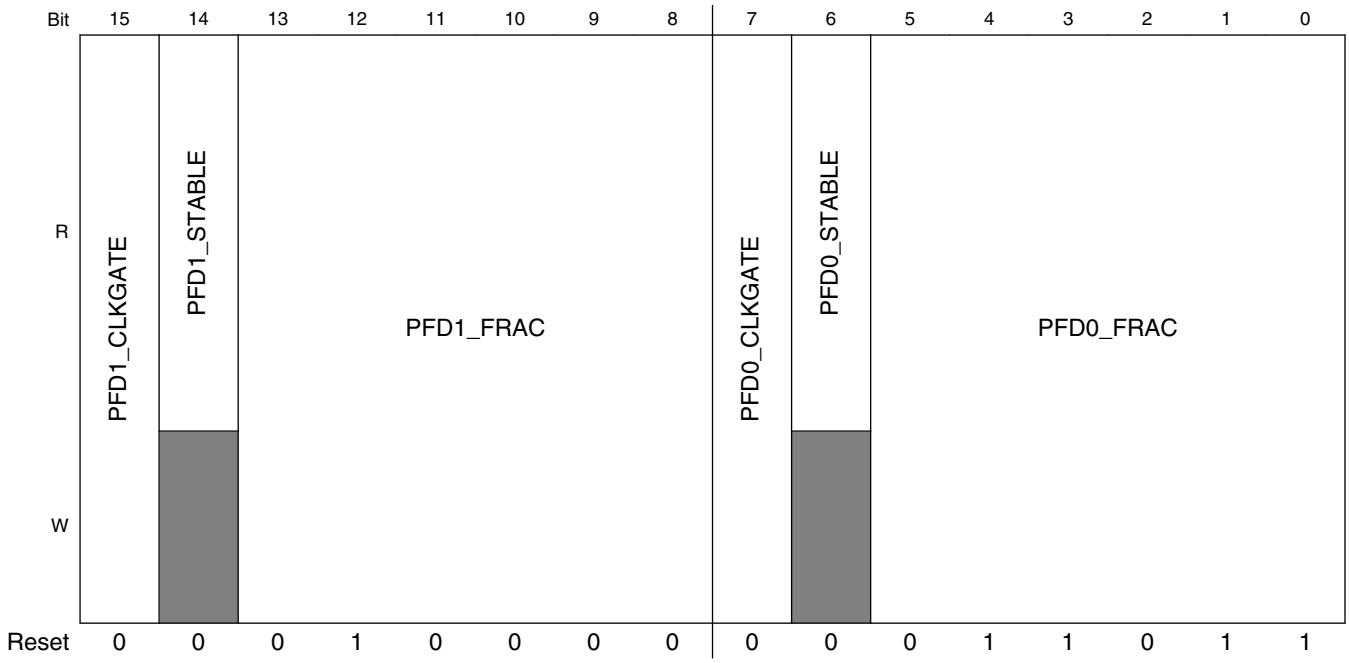
18.7.17 528MHz Clock (PLL2) Phase Fractional Divider Control Register (CCM_ANALOG_PFD_528*n*)

The PFD_528 control register provides control for PFD clock generation.

This register controls the 3-phase fractional clock dividers. The fractional clock frequencies are a product of the values in these registers.

Address: 20C_8000h base + 100h offset + (4d × i), where i=0d to 3d





CCM_ANALOG_PFD_528*n* field descriptions

Field	Description
31–24 -	This field is reserved. Reserved
23 PFD2_CLKGATE	IO Clock Gate. If set to 1, the IO fractional divider clock (reference ref_pfd2) is off (power savings). 0: ref_pfd2 fractional divider clock is enabled. Need to assert this bit before PLL powered down
22 PFD2_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
21–16 PFD2_FRAC	This field controls the fractional divide value. The resulting frequency shall be 528*18/PFD2_FRAC where PFD2_FRAC is in the range 12-35.
15 PFD1_CLKGATE	IO Clock Gate. If set to 1, the IO fractional divider clock (reference ref_pfd1) is off (power savings). 0: ref_pfd1 fractional divider clock is enabled. Need to assert this bit before PLL powered down
14 PFD1_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
13–8 PFD1_FRAC	This field controls the fractional divide value. The resulting frequency shall be 528*18/PFD1_FRAC where PFD1_FRAC is in the range 12-35.
7 PFD0_CLKGATE	If set to 1, the IO fractional divider clock (reference ref_pfd0) is off (power savings). 0: ref_pfd0 fractional divider clock is enabled. Need to assert this bit before PLL powered down

Table continues on the next page...

CCM_ANALOG_PFD_528*n* field descriptions (continued)

Field	Description
6 PFD0_STABLE	This read-only bitfield is for DIAGNOSTIC PURPOSES ONLY since the fractional divider should become stable quickly enough that this field will never need to be used by either device driver or application code. The value inverts when the new programmed fractional divide value has taken effect. Read this bit, program the new value, and when this bit inverts, the phase divider clock output is stable. Note that the value will not invert when the fractional divider is taken out of or placed into clock-gated state.
PFD0_FRAC	This field controls the fractional divide value. The resulting frequency shall be 528*18/PFD0_FRAC where PFD0_FRAC is in the range 12-35.

18.7.18 Miscellaneous Register 0 (CCM_ANALOG_MISC0*n*)

This register defines the control and status bits for miscellaneous analog blocks.

Address: 20C_8000h base + 150h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved			CLKGATE_DELAY			CLKGATE_CTRL	Reserved					WBCP_VPW_THRESH		OSC_XTALOK_EN	OSC_XTALOK
W																
Reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	OSC_I		Reserved	STOP_MODE_CONFIG	Reserved				REFTOP_VBGUP	REFTOP_VBGADJ			REFTOP_SELFBIASOFF	Reserved		REFTOP_PWD
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CCM_ANALOG_MISC0*n* field descriptions

Field	Description
31–29 -	This field is reserved.
28–26 CLKGATE_ DELAY	This field specifies the delay between powering up the XTAL 24MHz clock and releasing the clock to the digital logic inside the analog block. **NOTE:** Do not change the field during a low power event. This is not a field that the user would normally need to modify. **NOTE:** Not related to CCM. See Crystal Oscillator (XTALOSC) 000 0.5ms 001 1.0ms 010 2.0ms 011 3.0ms 100 4.0ms 101 5.0ms 110 6.0ms 111 7.0ms
25 CLKGATE_CTRL	This bit allows disabling the clock gate (always ungated) for the xtal 24MHz clock that clocks the digital logic in the analog block. **NOTE:** Do not change the field during a low power event. This is not a field that the user would normally need to modify. **NOTE:** Not related to CCM. See Crystal Oscillator (XTALOSC) 0 **ALLOW_AUTO_GATE** — Allow the logic to automatically gate the clock when the XTAL is powered down. 1 **NO_AUTO_GATE** — Prevent the logic from ever gating off the clock.
24–20 -	This field is reserved. Always set to zero.
19–18 WBCP_VPW_ THRESH	This signal alters the voltage that the pwell is charged pumped to. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 00 **NOMINAL_BIAS** — Nominal output pwell bias voltage. 01 **PLUS_25MV** — Increase pwell output voltage by 25mV. 10 **MINUS_25MV** — Decrease pwell output pwell voltage by 25mV. 11 **MINUS_50MV** — Decrease pwell output pwell voltage by 50mV.
17 OSC_XTALOK_ EN	This bit enables the detector that signals when the 24MHz crystal oscillator is stable. **NOTE:** Not related to CCM. See Crystal Oscillator (XTALOSC)
16 OSC_XTALOK	Status bit that signals that the output of the 24-MHz crystal oscillator is stable. Generated from a timer and active detection of the actual frequency. **NOTE:** Not related to CCM. See Crystal Oscillator (XTALOSC)

Table continues on the next page...

CCM_ANALOG_MISC0*n* field descriptions (continued)

Field	Description
15–14 OSC_I	This field determines the bias current in the 24MHz oscillator. The aim is to start up with the highest bias current, which can be decreased after startup if it is determined to be acceptable. **NOTE:** Not related to CCM. See Crystal Oscillator (XTALOSC) 00 **NOMINAL** — Nominal 01 **MINUS_12_5_PERCENT** — Decrease current by 12.5% 10 **MINUS_25_PERCENT** — Decrease current by 25.0% 11 **MINUS_37_5_PERCENT** — Decrease current by 37.5%
13 Reserved	This field is reserved. Reserved
12 STOP_MODE_CONFIG	Configure the analog behavior in stop mode. 0x0 **DEEP** — Deep Stop Mode - 0x0 All analog except RTC powered down on Stop mode assertion 0x1 **LIGHT** — Light Stop Mode - 0x1 All the analog domain except the LDO_1P1, LDO_2P5, and PLL3 is powered down on STOP mode assertion. If required the CCM can be configured not to power down the oscillator (XTALOSC). PLL3 can be disabled with register settings if desired.
11–8 -	This field is reserved. Reserved
7 REFTOP_VBGUP	Status bit that signals the analog bandgap voltage is up and stable. 1 - Stable. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
6–4 REFTOP_VBGADJ	**NOTE:** Not related to CCM. See Power Management Unit (PMU) 000 Nominal VBG 001 VBG+0.78% 010 VBG+1.56% 011 VBG+2.34% 100 VBG-0.78% 101 VBG-1.56% 110 VBG-2.34% 111 VBG-3.12%
3 REFTOP_SELFBIASOFF	Control bit to disable the self-bias circuit in the analog bandgap. The self-bias circuit is used by the bandgap during startup. This bit should be set after the bandgap has stabilized and is necessary for best noise performance of analog blocks using the outputs of the bandgap. **NOTE:** Value should be returned to zero before removing vddhigh_in or asserting bit 0 of this register (REFTOP_PWD) to assure proper restart of the circuit. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 0 Uses coarse bias currents for startup 1 Uses bandgap-based bias currents for best performance.
2–1 -	This field is reserved.
0 REFTOP_PWD	Control bit to power-down the analog bandgap reference circuitry.

Table continues on the next page...

CCM_ANALOG_MISC0*n* field descriptions (continued)

Field	Description
	NOTE: A note of caution, the bandgap is necessary for correct operation of most of the LDO, PLL, and other analog functions on the die. **NOTE:** Not related to CCM. See Power Management Unit (PMU)

18.7.19 Miscellaneous Register 1 (CCM_ANALOG_MISC1*n*)

This register defines the control and status bits for miscellaneous analog blocks. The LVDS1 and LVDS2 controls below control the behavior of the anaclk1/1b and anaclk2/2b LVDS IO's.

Address: 20C_8000h base + 160h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IRQ_DIG_BO	IRQ_ANA_BO	IRQ_TEMPSENSE	Reserved												
W	w1c	w1c	w1c													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved		LVDSCLK2_IBEN	LVDSCLK1_IBEN	LVDSCLK2_OBEN	LVDSCLK1_OBEN	LVDS2_CLK_SEL					LVDS1_CLK_SEL				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CCM_ANALOG_MISC1*n* field descriptions

Field	Description
31 IRQ_DIG_BO	This status bit is set to one when when any of the digital regulator brownout interrupts assert. Check the regulator status bits to discover which regulator interrupt asserted. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
30 IRQ_ANA_BO	This status bit is set to one when when any of the analog regulator brownout interrupts assert. Check the regulator status bits to discover which regulator interrupt asserted. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
29 IRQ_ TEMPSENSE	This status bit is set to one when when the temperature sensor interrupt asserts. **NOTE:** Not related to CCM. See Temperature Monitor (TEMPMON)
28–14 -	This field is reserved.
13 LVDSCLK2_ IBEN	This enables the LVDS input buffer for anaclk2/2b. Do not enable input and output buffers simultaneously.
12 LVDSCLK1_ IBEN	This enables the LVDS input buffer for anaclk1/1b. Do not enable input and output buffers simultaneously.
11 LVDSCLK2_ OBEN	This enables the LVDS output buffer for anaclk2/2b. Do not enable input and output buffers simultaneously.
10 LVDSCLK1_ OBEN	This enables the LVDS output buffer for anaclk1/1b. Do not enable input and output buffers simultaneously.
9–5 LVDS2_CLK_ SEL	This field selects the clk to be routed to anaclk2/2b. 00000 **ARM_PLL** — Arm PLL 00001 **SYS_PLL** — System PLL 00010 **PFD4** — pfd4 00011 **PFD5** — pfd5 00100 **PFD6** — pfd6 00101 **PFD7** — pfd7 00110 **AUDIO_PLL** — Audio PLL 00111 **VIDEO_PLL** — Video PLL 01000 **MLB_PLL** — MLB PLL 01001 **ETHERNET_REF** — ethernet ref clock 01010 **PCIE_REF** — PCIe ref clock 01011 **SATA_REF** — SATA ref clock 01100 **USB1_PLL** — USB1 PLL clock 01101 **USB2_PLL** — USB2 PLL clock 01110 **PFD0** — pfd0 01111 **PFD1** — pfd1 10000 **PFD2** — pfd2 10001 **PFD3** — pfd3 10010 **XTAL** — xtal

Table continues on the next page...

CCM_ANALOG_MISC1*n* field descriptions (continued)

Field	Description
	10011 **LVDS1** — LVDS1 (loopback) 10100 **LVDS2** — LVDS2 (not useful) 10101 to 11111 pfd7
LVDS1_CLK_SEL	This field selects the clk to be routed to anaclk2/2b. 00000 **ARM_PLL** — Arm PLL 00001 **SYS_PLL** — System PLL 00010 **PFD4** — pfd4 00011 **PFD5** — pfd5 00100 **PFD6** — pfd6 00101 **PFD7** — pfd7 00110 **AUDIO_PLL** — Audio PLL 00111 **VIDEO_PLL** — Video PLL 01000 **MLB_PLL** — MLB PLL 01001 **ETHERNET_REF** — ethernet ref clock 01010 **PCIE_REF** — PCIe ref clock 01011 **SATA_REF** — SATA ref clock 01100 **USB1_PLL** — USB1 PLL clock 01101 **USB2_PLL** — USB2 PLL clock 01110 **PFD0** — pfd0 01111 **PFD1** — pfd1 10000 **PFD2** — pfd2 10001 **PFD3** — pfd3 10010 **XTAL** — xtal 10011 **LVDS1** — LVDS1 (loopback) 10100 **LVDS2** — LVDS2 (not useful) 10101 to 11111 pfd7

18.7.20 Miscellaneous Register 2 (CCM_ANALOG_MISC2*n*)

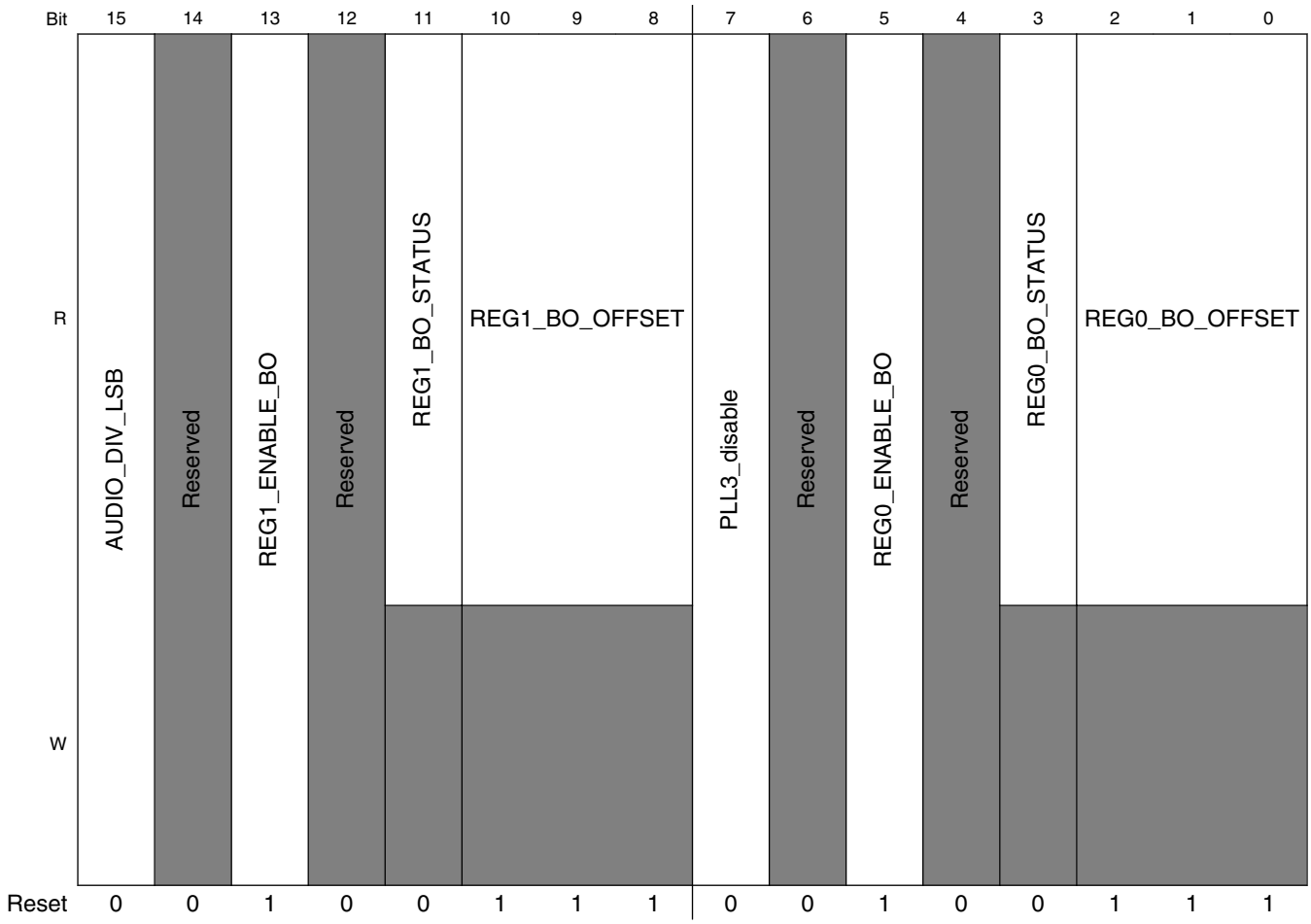
This register defines the control for miscellaneous analog blocks.

NOTE

This register is shared with PMU.

Address: 20C_8000h base + 170h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	VIDEO_DIV		REG2_STEP_TIME		REG1_STEP_TIME		REG0_STEP_TIME		AUDIO_DIV_MSB	REG2_OK	REG2_ENABLE_BO	Reserved	REG2_BO_STATUS	REG2_BO_OFFSET		
W																
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1



CCM_ANALOG_MISC2*n* field descriptions

Field	Description
31–30 VIDEO_DIV	Post-divider for video. The output clock of the video PLL should be gated prior to changing this divider to prevent glitches. This divider is feed by PLL_VIDEOn[POST_DIV_SELECT] to achieve division ratios of /1, /2, /4, /8, and /16. 00 divide by 1 (Default) 01 divide by 2 10 divide by 1 11 divide by 4
29–28 REG2_STEP_TIME	Number of clock periods (24MHz clock). **NOTE:** Not related to CCM. See Power Management Unit (PMU) 00 **64_CLOCKS** — 64 01 **128_CLOCKS** — 128 10 **256_CLOCKS** — 256 11 **512_CLOCKS** — 512
27–26 REG1_STEP_TIME	Number of clock periods (24MHz clock).

Table continues on the next page...

CCM_ANALOG_MISC2*n* field descriptions (continued)

Field	Description
	NOTE: Not related to CCM. See Power Management Unit (PMU) 00 **64_CLOCKS** — 64 01 **128_CLOCKS** — 128 10 **256_CLOCKS** — 256 11 **512_CLOCKS** — 512
25–24 REG0_STEP_TIME	Number of clock periods (24MHz clock). **NOTE:** Not related to CCM. See Power Management Unit (PMU) 00 **64_CLOCKS** — 64 01 **128_CLOCKS** — 128 10 **256_CLOCKS** — 256 11 **512_CLOCKS** — 512
23 AUDIO_DIV_MSB	MSB of Post-divider for Audio PLL. The output clock of the video PLL should be gated prior to changing this divider to prevent glitches. This divider is feed by PLL_AUDIOn[POST_DIV_SELECT] to achieve division ratios of /1, /2, /4, /8, and /16. **NOTE:** MSB bit value pertains to the first bit, please program the LSB bit (bit 15) as well to change divider value for more information. 00 divide by 1 (Default) 01 divide by 2 10 divide by 1 11 divide by 4
22 REG2_OK	Signals that the voltage is above the brownout level for the SOC supply. 1 = regulator output > brownout_target **NOTE:** Not related to CCM. See Power Management Unit (PMU)
21 REG2_ENABLE_BO	Enables the brownout detection. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
20 -	This field is reserved.
19 REG2_BO_STATUS	Reg2 brownout status bit. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
18–16 REG2_BO_OFFSET	This field defines the brown out voltage offset for the xPU power domain. IRQ_DIG_BO is also asserted. Single-bit increments reflect 25mV brownout voltage steps. The reset brown-offset is 175mV below the programmed target code. Brownout target = OUTPUT_TRG - BO_OFFSET. Some steps may be irrelevant because of input supply limitations or load operation. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 100 Brownout offset = 0.100V 111 Brownout offset = 0.175V

Table continues on the next page...

CCM_ANALOG_MISC2*n* field descriptions (continued)

Field	Description
15 AUDIO_DIV_LSB	LSB of Post-divider for Audio PLL. The output clock of the video PLL should be gated prior to changing this divider to prevent glitches. This divider is feed by PLL_AUDIOn[POST_DIV_SELECT] to achieve division ratios of /1, /2, /4, /8, and /16. **NOTE:** LSB bit value pertains to the last bit, please program the MSB bit (bit 23) as well, to change divider value for more information. 00 divide by 1 (Default) 01 divide by 2 10 divide by 1 11 divide by 4
14 - Reserved	This field is reserved.
13 REG1_ENABLE_BO	Enables the brownout detection. **NOTE:** Not related to CCM. See Power Management Unit (PMU)
12 -	This field is reserved.
11 REG1_BO_STATUS	Reg1 brownout status bit. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 1 Brownout, supply is below target minus brownout offset.
10–8 REG1_BO_OFFSET	This field defines the brown out voltage offset for the xPU power domain. IRQ_DIG_BO is also asserted. Single-bit increments reflect 25mV brownout voltage steps. The reset brown-offset is 175mV below the programmed target code. Brownout target = OUTPUT_TRG - BO_OFFSET. Some steps may be irrelevant because of input supply limitations or load operation. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 100 Brownout offset = 0.100V 111 Brownout offset = 0.175V
7 PLL3_disable	When USB is in low power suspend mode this Control bit is used to indicate if other system peripherals require the USB PLL3 clock when the SoC is not in low power mode. A user needs to set this bit if they want to optionally disable PLL3 while the SoC is not in any low power mode to save power. When the system does go into low power mode this bit setting would not have any affect. **NOTE:** When USB is in low power suspend mode users would need to ensure PLL3 is not being used before setting this bit in RUN mode. Please refer to the correct PLL disabling procedure in Disabling / Enabling PLLs 0 PLL3 is being used by peripherals and is enabled when SoC is not in any low power mode 1 PLL3 can be disabled when the SoC is not in any low power mode
6 -	This field is reserved.
5 REG0_ENABLE_BO	Enables the brownout detection. **NOTE:** Not related to CCM. See Power Management Unit (PMU)

Table continues on the next page...

CCM_ANALOG_MISC2*n* field descriptions (continued)

Field	Description
4 -	This field is reserved.
3 REG0_BO_ STATUS	Reg0 brownout status bit. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 1 Brownout, supply is below target minus brownout offset.
REG0_BO_ OFFSET	This field defines the brown out voltage offset for the CORE power domain. IRQ_DIG_BO is also asserted. Single-bit increments reflect 25mV brownout voltage steps. Some steps may be irrelevant because of input supply limitations or load operation. **NOTE:** Not related to CCM. See Power Management Unit (PMU) 100 Brownout offset = 0.100V 111 Brownout offset = 0.175V

Chapter 19
MIPI CSI to IPU Gasket (CSI2IPU)

19.1 Overview

The CSI2IPU gasket is a digital core that functions as a gasket interface between the MIPI CSI-2 host controller and the IPU system. This facilitates communication between a MIPI CSI-2 compliant camera sensor and IPU (the image processing unit). The gasket's main functions are to synchronize the CSI-2 input 32-bit data bus with the 16-bit data bus and to separate the four virtual channels.

The following block diagram shows the CSI2IPU gasket's position in the system as part of the CSI to IPU connectivity.

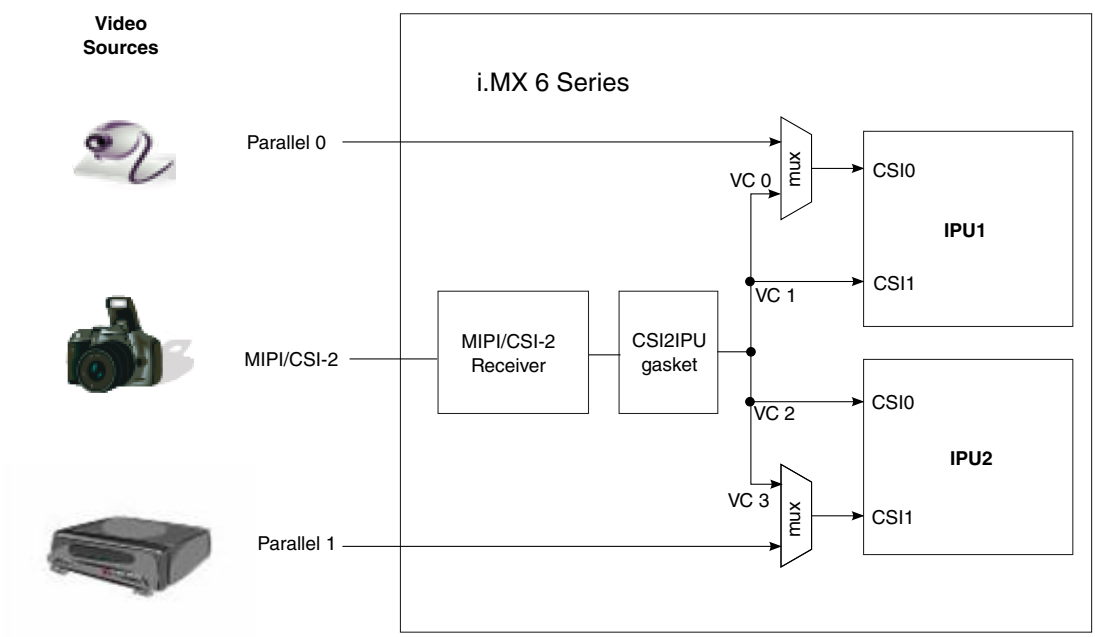


Figure 19-1. CSI2IPU gasket connectivity

19.1.1 CSI2IPU feature summary

The CSI2IPU gasket supports:

- Up to 4 virtual channels of MIPI CSI-2 host controller
- All data types of the MIPI Alliance Standard for Camera Serial Interface (CSI)
- Dynamically configurable pixel clock gating or non-gating for the IPU module
- Dynamically configurable RGB444 and YUV422 data format for the IPU module
- A software reset to reset the program during operation

19.1.2 CSI2IPU architectural description

The following figure shows the overall architecture of the CSI2IPU gasket.

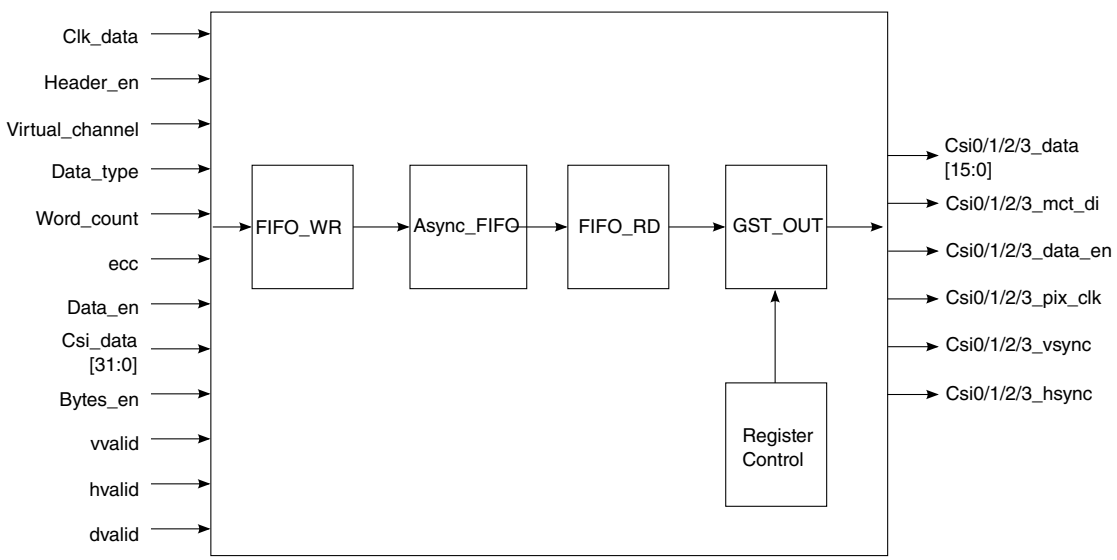


Figure 19-2. CSI2IPU gasket architecture

The main blocks are:

- FIFO WRITE—Responsible for receiving the CSI-2 host input signals and integrating them into a wr_data[46:0] bus
- ASYNC FIFO—Used to synchronize the wr_data to the read clock domain. It also generates FIFO full and empty signals.
- FIFO READ—Responsible for reading FIFO output and changing the 32-bit data bus to the 16-bit data bus.

- GST OUT—Generates output data according to the format controlled by the register bank.
- REGISTER CONTROL—register bank used to control the GST OUT block.

19.2 CSI2IPU signals

The following tables describe the CSI2IPU gasket's input and output signals.

Table 19-1. Input signals to CSI2IPU

Signal	Description
clk_data	Clock input for the CSI host controller
header_en	Shows that the header data at the input (signals virtual_channel, data_type, word_count, and ecc) is valid for the packet being transferred. This signal stays HIGH during the complete packet transfer.
virtual_channel[1:0]	Virtual channel identifier value • 00— VC0 • 01—VC1 • 10—VC2 • 11—VC3
data_type[5:0]	• 00h–07h—Synchronization short packet data types • 08h–0Fh—Generic short packet data types • 10h–17h—Generic long packet data types • 18h–1Fh—YUV data • 20h–27h—RGB data • 28h–2Fh—RAW data • 30h– 37h—User defined byte-based data • 38h–3Fh—Reserved
word_count[15:0]	16-bit word count information from the packet header
ecc[7:0]	8-bit error correction code for the packet header
data_en	Shows that new payload data is present at the input (signals csi_data and bytes_en). This signal is only asserted when receiving long packets. Note that the signal header_en is also asserted when data_en is asserted.
bytes_en[1:0]	Shows how may bytes are in the csi_data output signal • 00 = 1 valid byte in csi_data[7:0] • 01 = 2 valid bytes in csi_data[15:0] • 10 = 3 valid bytes in csi_data[23:0] • 11 = 4 valid bytes in csi_data[31:0]
vvalid[3:0]	This signal is asserted when a Frame Start is detected and de-asserted when a Frame End is detected. Each virtual channel has one valid signal. • vvalid[0]—vvalid for Virtual Channel 0 • vvalid[1]—vvalid for Virtual Channel 1 • vvalid[2]—vvalid for Virtual Channel 2 • vvalid[3]—vvalid for Virtual Channel 3
hvalid[3:0]	This signal is asserted when a Line Start is detected and de-asserted when a Line End is detected. Line Start and Line End are optional; when they are not available, hvalid has the same behavior as dvalid. Each virtual channel has one valid signal. • hvalid[0]—hvalid for Virtual Channel 0

Table continues on the next page...

Table 19-1. Input signals to CSI2IPU (continued)

Signal	Description
	• hvalid[1]—hvalid for Virtual Channel 1 • hvalid[2]—hvalid for Virtual Channel 2 • hvalid[3]—hvalid for Virtual Channel 3
dvalid[3:0]	Used to signal when valid data is available
csi_data_out[31:0]	Payload data
hw_resetn	System hardware reset
sft_resetn	System software reset from top module
gst_clk_sel	• 0—Gating clock mode • 1—Non-gating clock mode
gst_rgb444_fm	Rgb444 output format select • 0—{4'h0,r4g4b4} • 1—{r4,1'b0,g4,1'b0,b4,1'b0}
gst_yuv422_8bit_fm	Yuv422 output format select • 0—output YUYV • 1—output UYVY
ccm_pixel_clk	Reading clock from CCM

Table 19-2. CSI2IPU output signals to IPU

Signal	Description
csi*n*[1] _data[15:0]	Virtual channel *n*[1] CSI data
csi*n*[1] _mct_di[7:0]	Virtual channel *n*[1] {Channel_ID[1:0],Data type[5:0]}
csi*n*[1] _data_en	virtual channel *n*[1] data enable signal
csi*n*[1] _pix_clk	virtual channel *n*[1] pixel clock signal clock
csi*n*[1] _vsync	virtual channel *n*[1] vertical synchronism signal(Frame start)
csi*n*[1] _hsync	virtual channel *n*[1] horizontal synchronism signal
csi*n*[1] _Byte_en	virtual channel *n*[1] byte enable • 0–[7:0] valid • 1:[15:0] valid

1. n = 0 - 3

19.3 Timing interface

The following figure shows the timing for the image data interface.

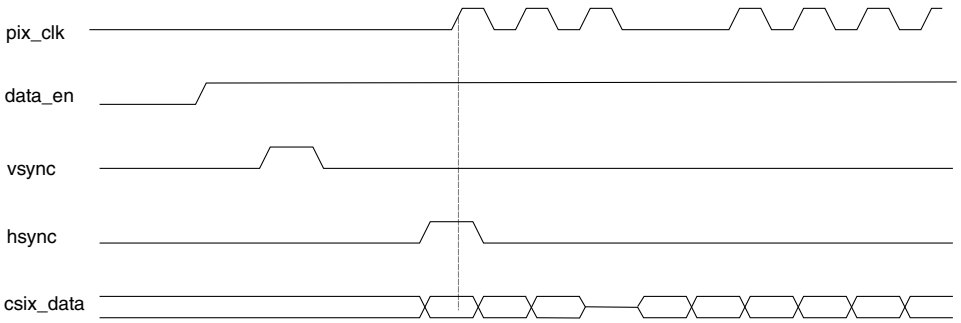


Figure 19-3. Image data interface

19.4 Payload data output formats

The following figures illustrate the formats for the payload data output.

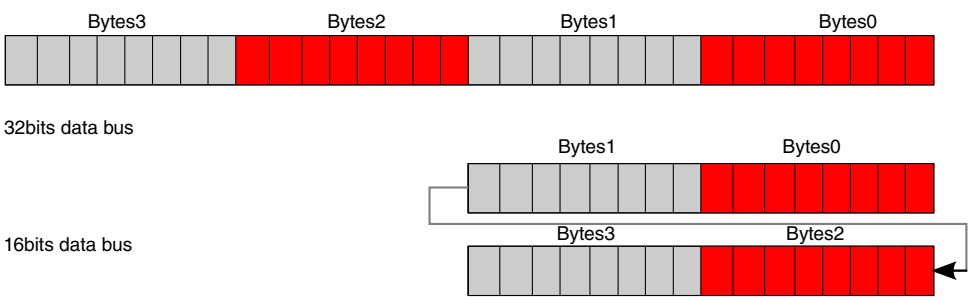


Figure 19-4. General/arbitrary data reception

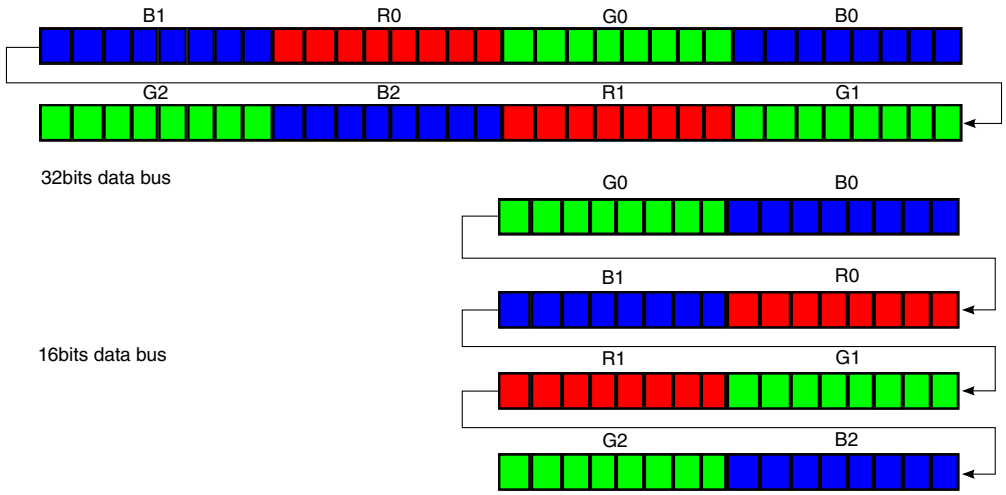


Figure 19-5. RGB888 data reception

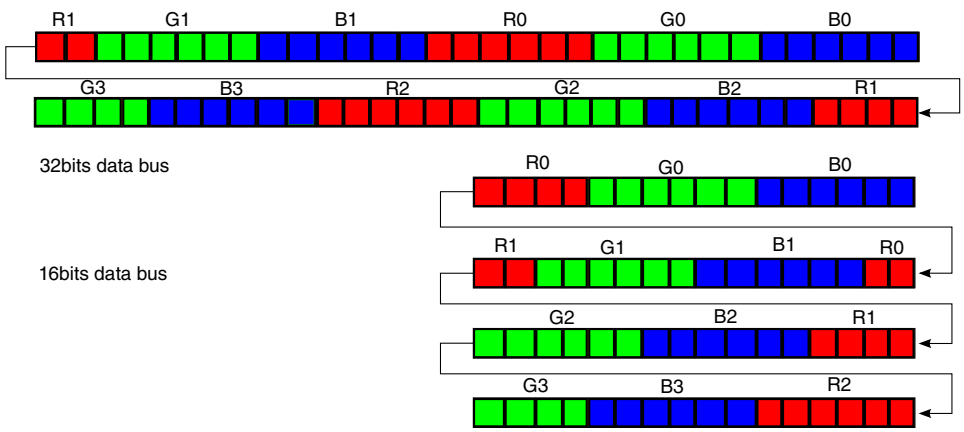


Figure 19-6. RGB666 data reception

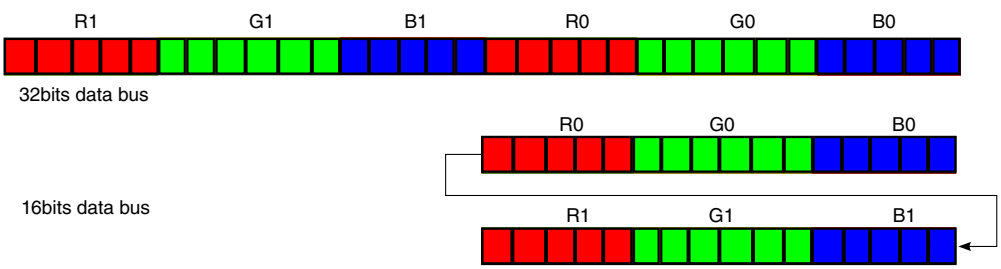


Figure 19-7. RGB565 data reception

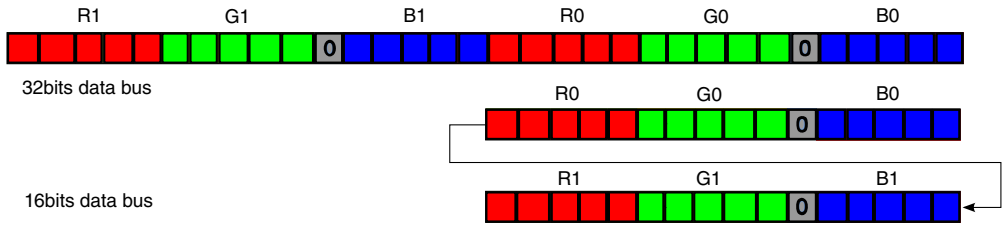


Figure 19-8. RGB555 data reception

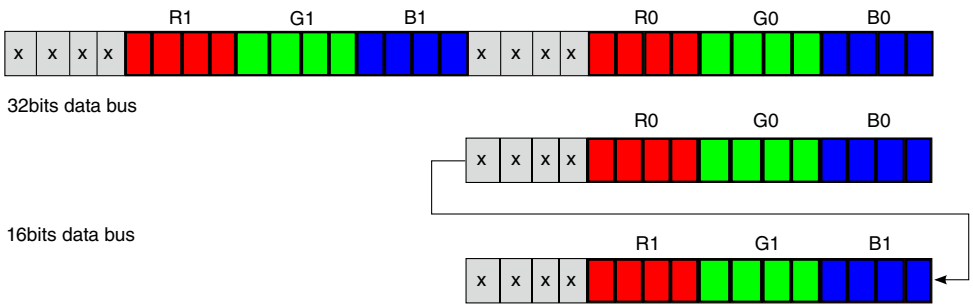


Figure 19-9. RGB444 data reception

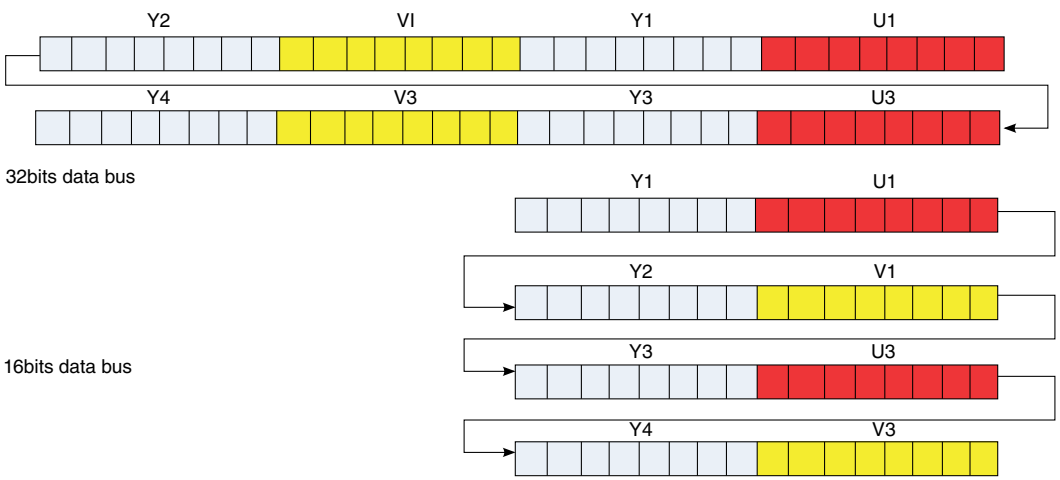


Figure 19-10. YUV422-8 data reception

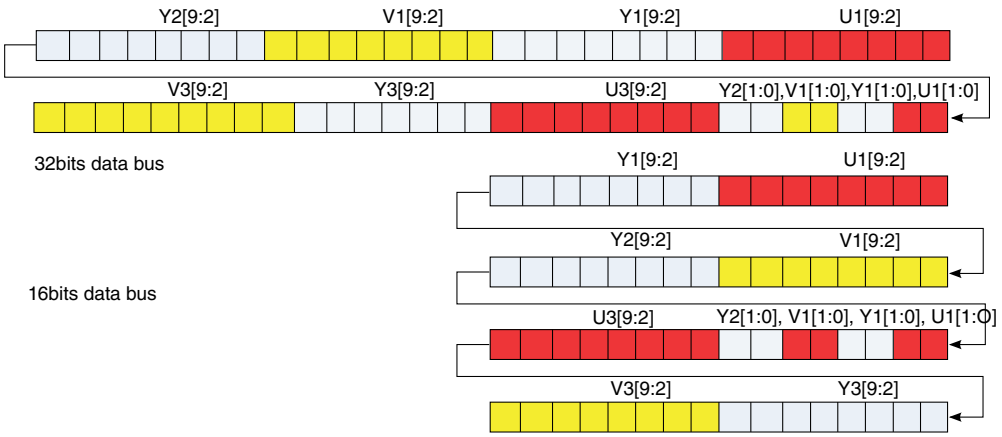


Figure 19-11. YUV422-10 data reception

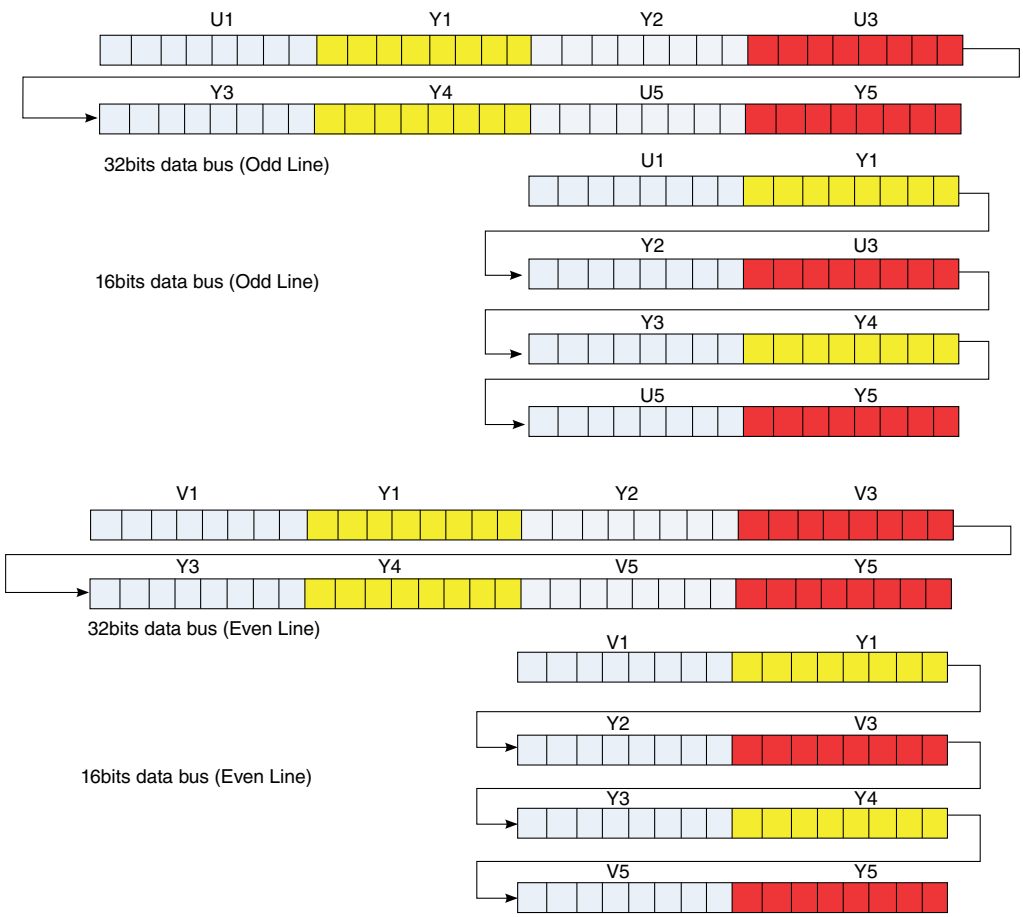


Figure 19-12. YUV420-8 (legacy) data reception

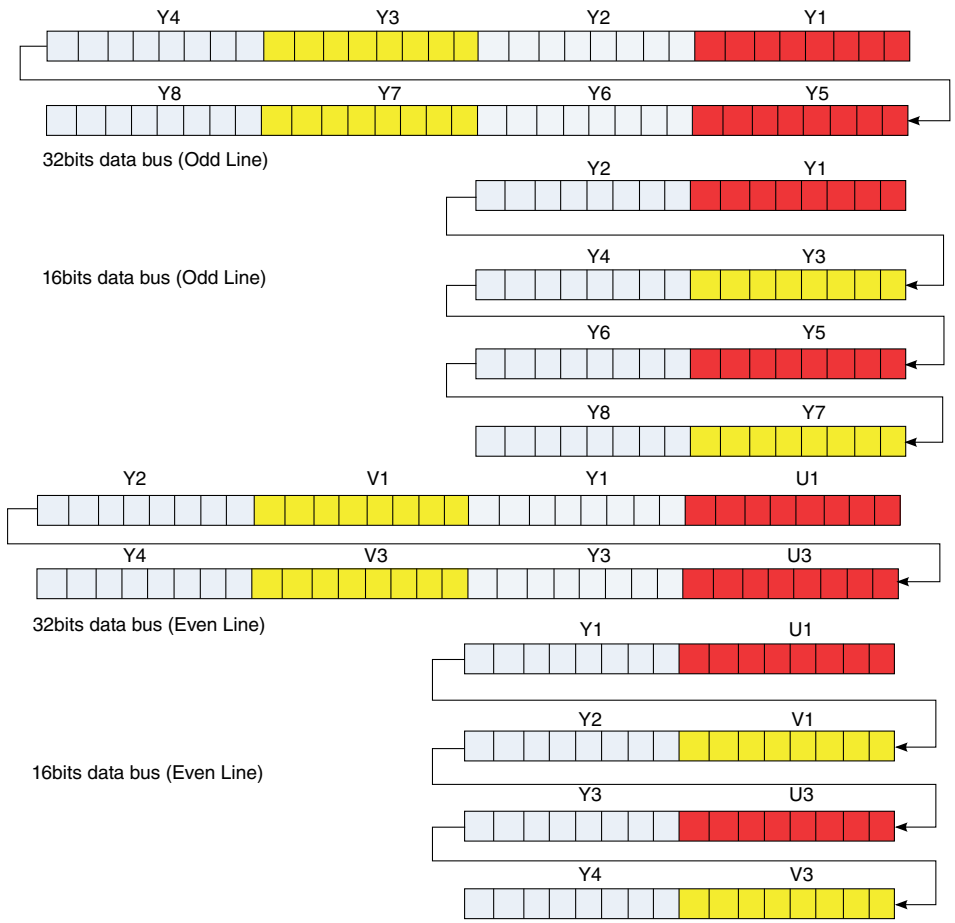


Figure 19-13. YUV420-8 data reception

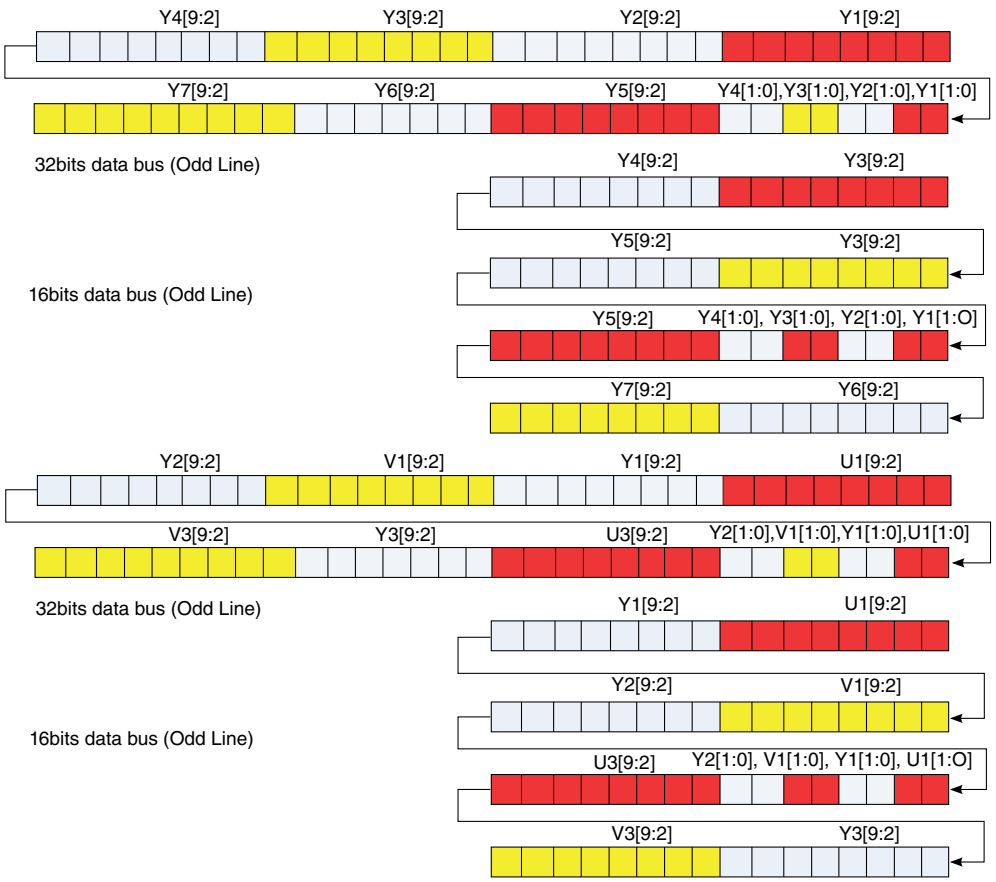


Figure 19-14. YUV420-10 data reception

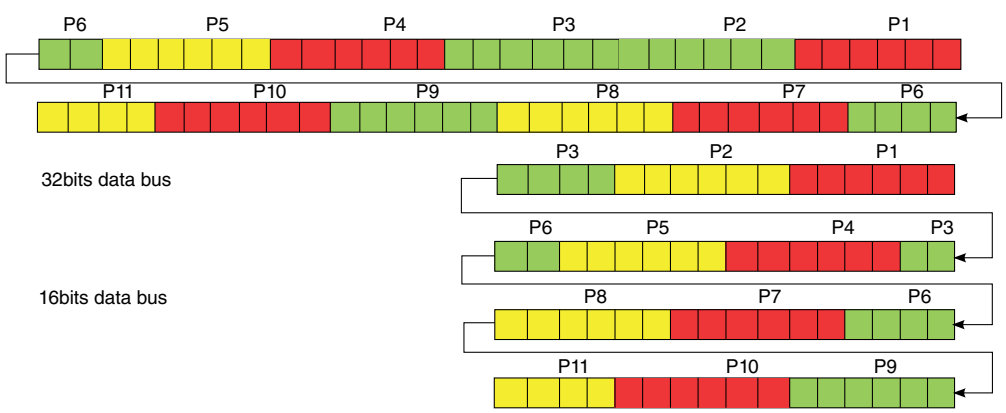


Figure 19-15. RAW-6 data reception

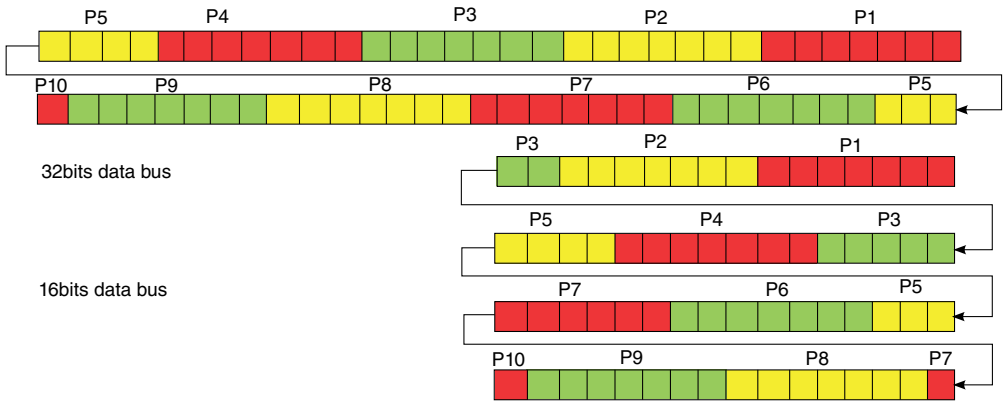


Figure 19-16. RAW-7 data reception

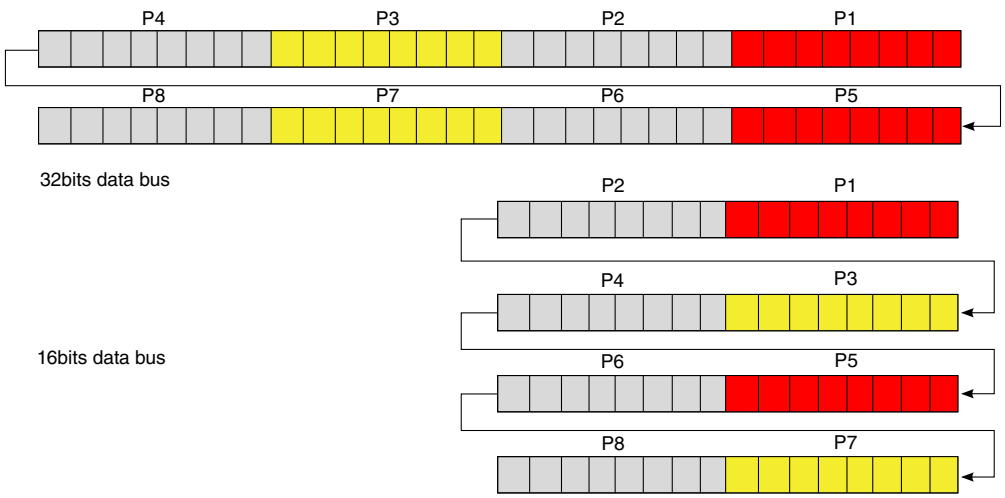


Figure 19-17. RAW-8 data reception

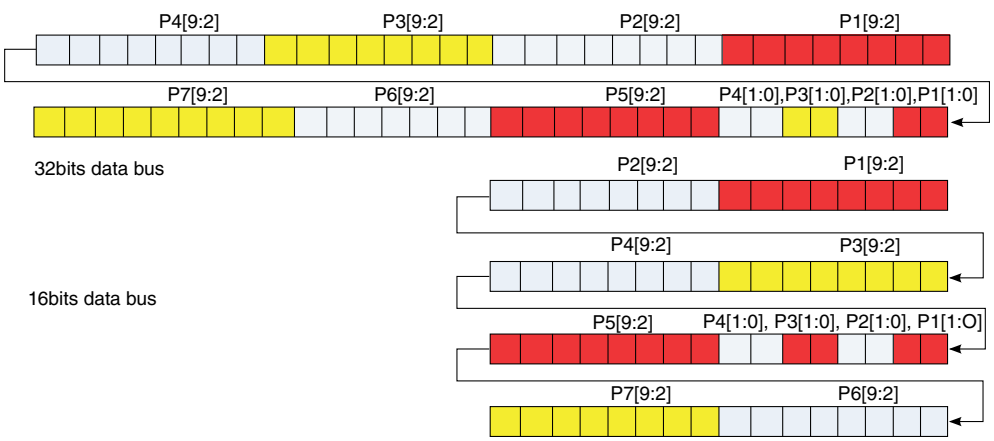


Figure 19-18. RAW-10 data reception

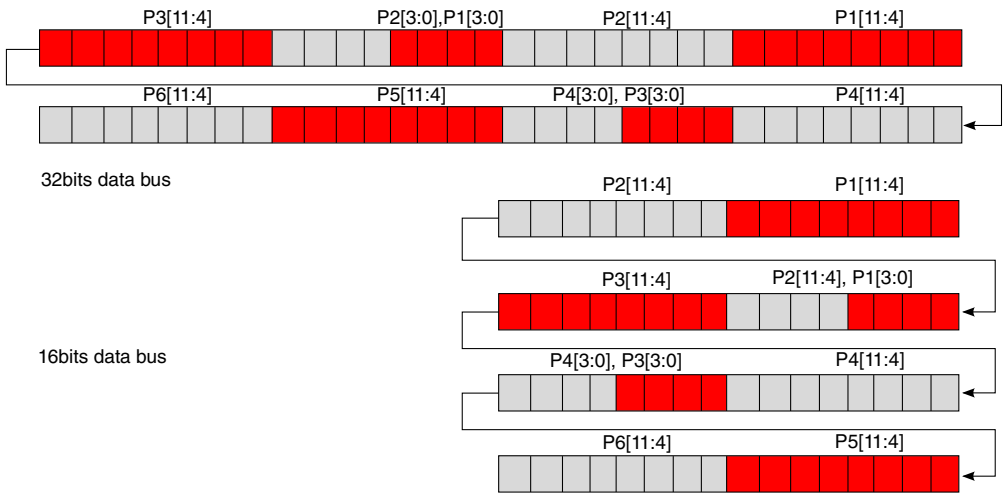


Figure 19-19. RAW-12 data reception

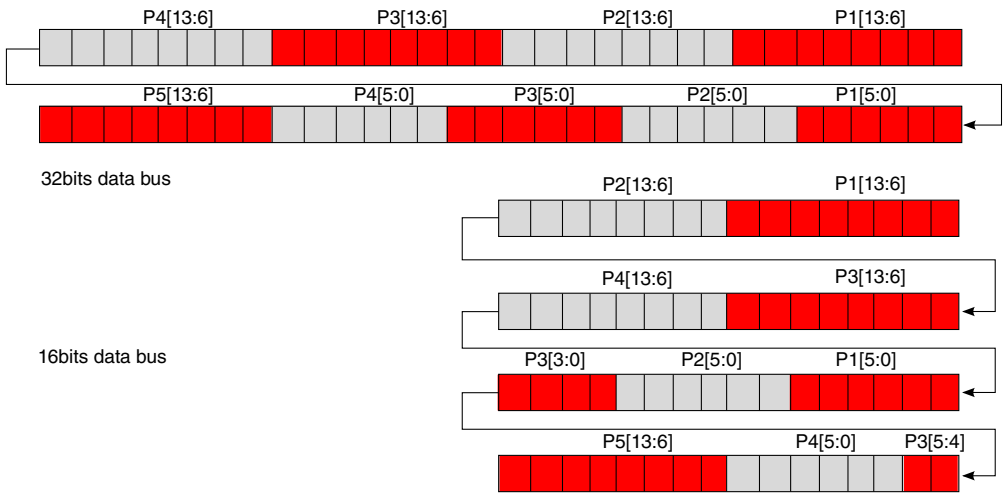


Figure 19-20. RAW-14 data reception

19.5 CSI2IPU Memory Map/Register Definition

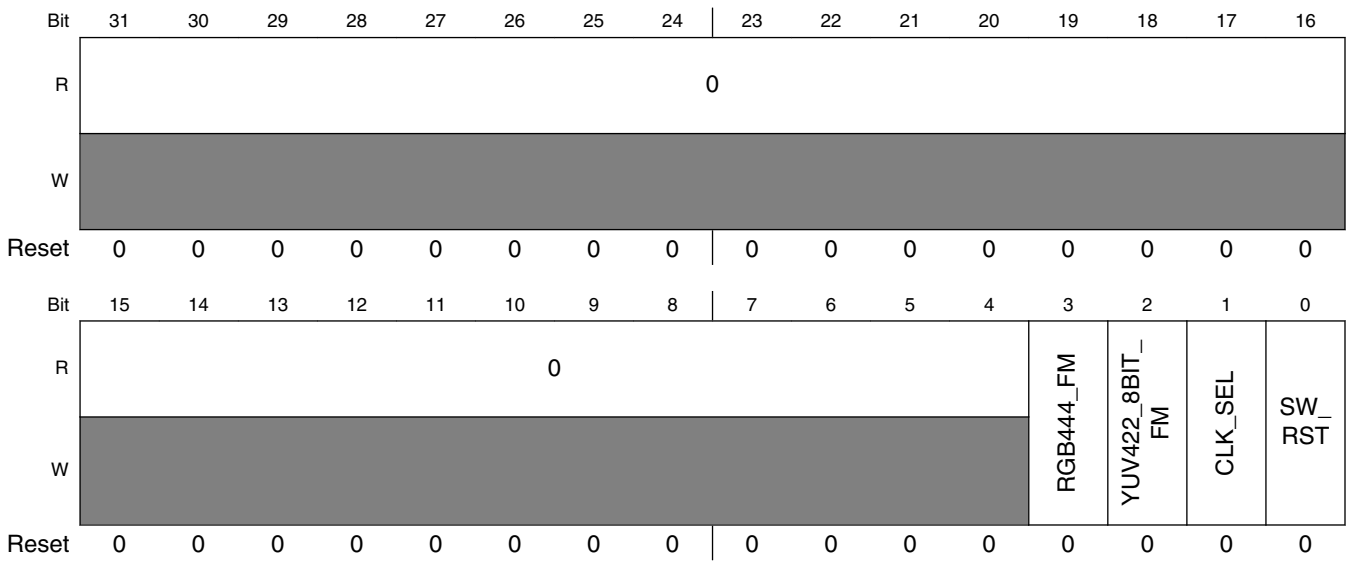
CSI2IPU memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21D_CF00	CSI 2 IPU Gasket Software Reset (CSI2IPU_SW_RST)	32	R/W	0000_0000h	19.5.1/955

19.5.1 CSI 2 IPU Gasket Software Reset (CSI2IPU_SW_RST)

This register describes the IPU interface signals.

Address: 21D_C000h base + F00h offset = 21D_CF00h



CSI2IPU_SW_RST field descriptions

Field	Description
31–4 Reserved	This read-only field is reserved and always has the value 0.
3 RGB444_FM	rgb444 mode selection 0 {4'h0,r4b4g4} 1 {r4,1'b0,g4,2'b00,b4,1'b0}
2 YUV422_8BIT_FM	YUV422 8-bit mode selection 0 YUYV 1 UYVY
1 CLK_SEL	Clock mode selection 0 Gated Mode 1 Non-Gated Mode
0 SW_RST	Software Reset 0 Software Reset Disable 1 Software Reset Enable

Chapter 20
Display Content Integrity Checker (DCIC)

20.1 Overview

The goal of the DCIC is to verify that a safety-critical information sent to a display is not corrupted.

Such a verification is mandatory for warning icons in the instrument cluster of a car, to comply with the ASIL B (Automotive Safety Integrity Level B) specification. It is also required in other safety-sensitive systems.

Using external muxing DCIC can monitor either one of the IPU display port outputs or feedback signals going from IO pads of Parallel display interface. The figure below shows DCIC integration in system with two IPU blocks.

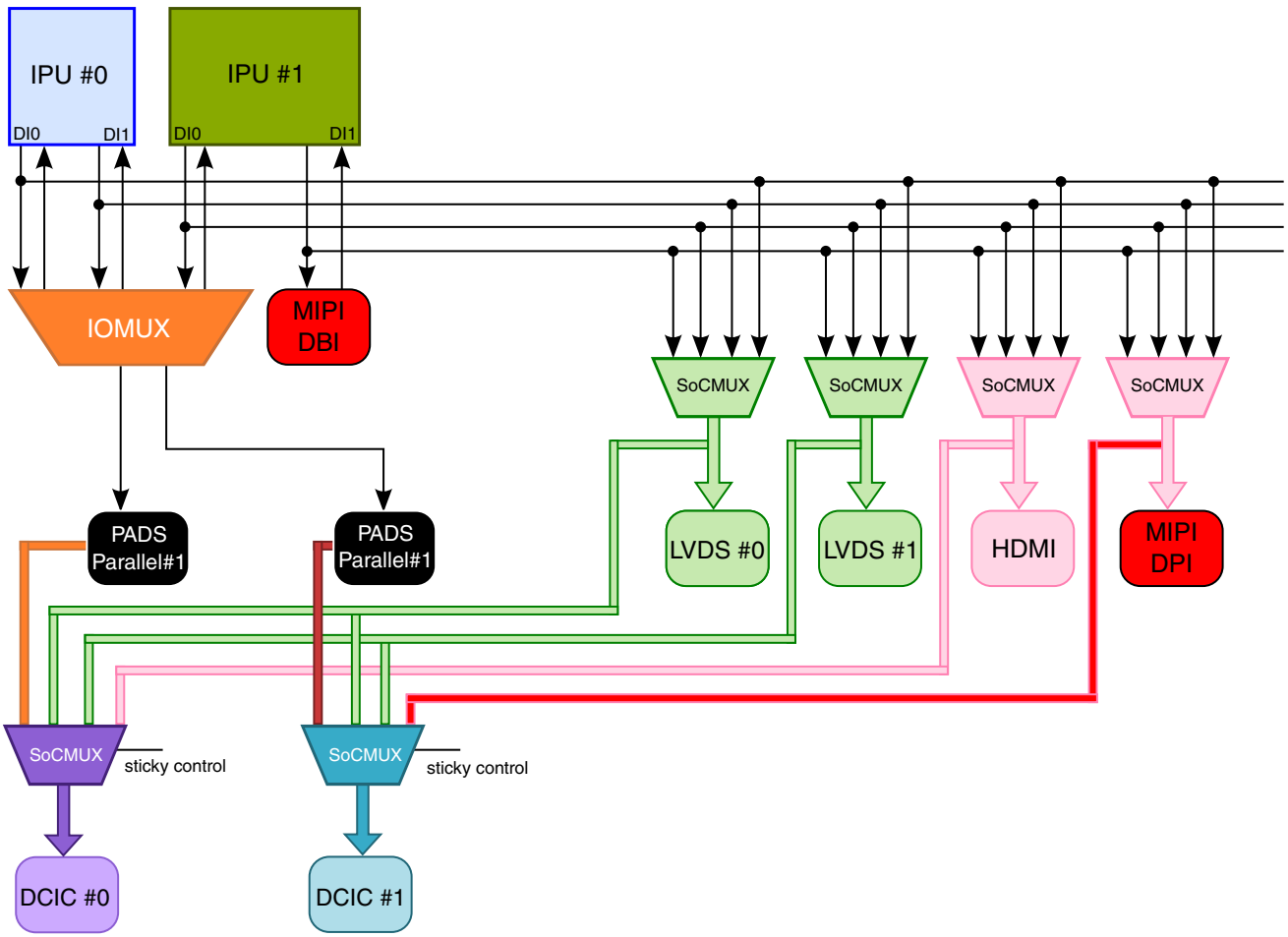


Figure 20-1. DCIC system integration example

20.1.1 Block Diagram

The figure below shows DCIC top level block diagram.

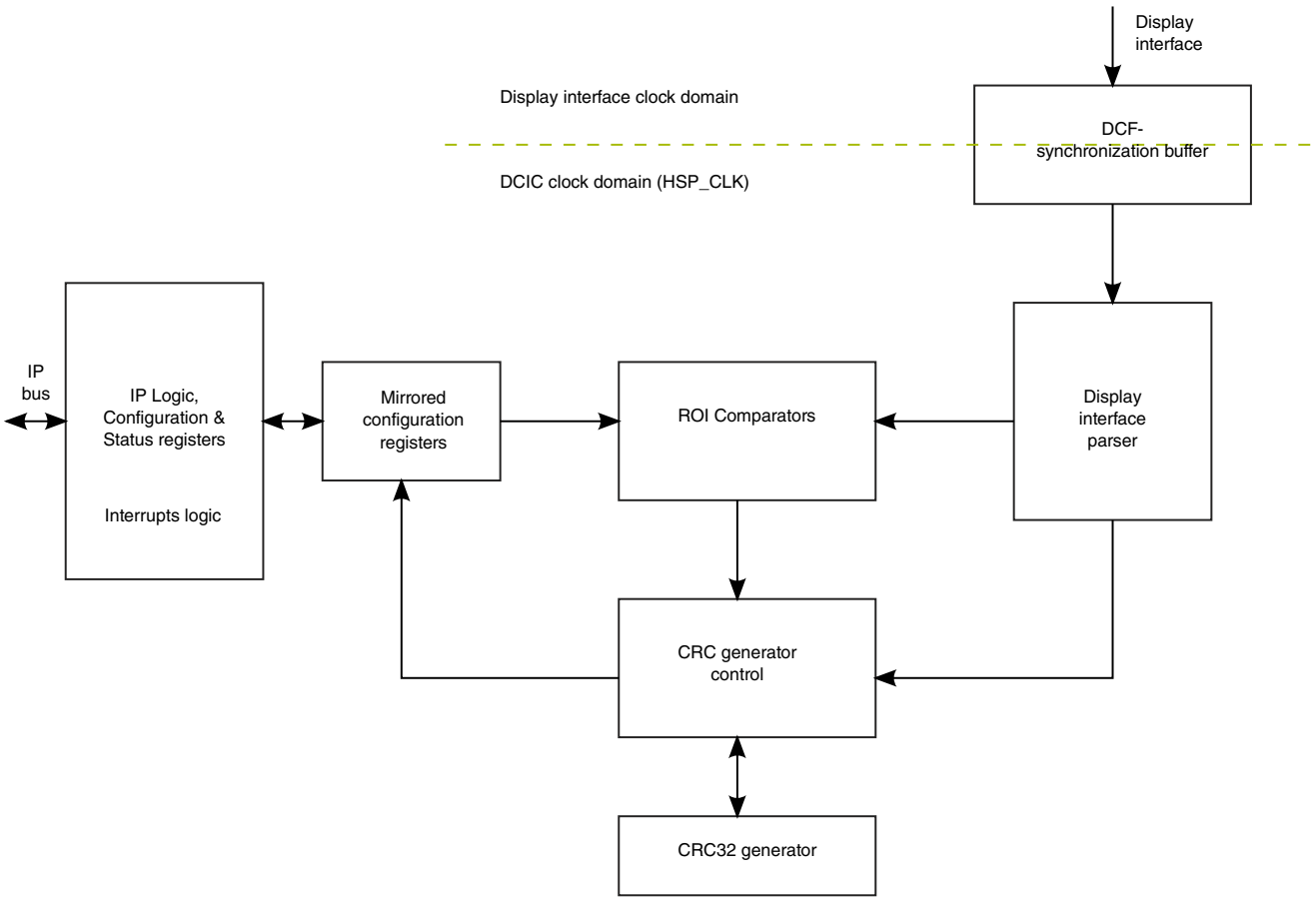


Figure 20-2. DCIC Block Diagram

20.1.2 Features

- Pixel clock up to 266 MHz
- Configurable polarity of Display Interface control signals
- 24-bit pixel data bus
- Up to 16 rectangular ROIs with a configurable location and size
- Independent CRC32 signature calculation for each ROI
- External controller mismatch indication signal

20.2 Functional Description

This section provides a complete functional description of the block.

20.2.1 Generic synchronous parallel display interface

The figure below depicts the LCD interface timing for a generic active matrix color TFT panel. In this figure signals are shown with negative polarity, but both polarity modes are supported by DCIC through configuration of appropriate bits of DCICCR register.

The sequence of events for active matrix interface timing is:

- IPP_DISP_CLK latches data into the panel on its negative edge (when positive polarity is selected). In active mode, IPP_DISP_CLK runs continuously.
- HSYNC causes the panel to start a new line.
- VSYNC causes the panel to start a new frame. It always encompasses at least one HSYNC pulse.
- DRDY acts like an output enable signal. This output enables the data to be shifted onto the display. When disabled, the data is invalid and the trace is off.

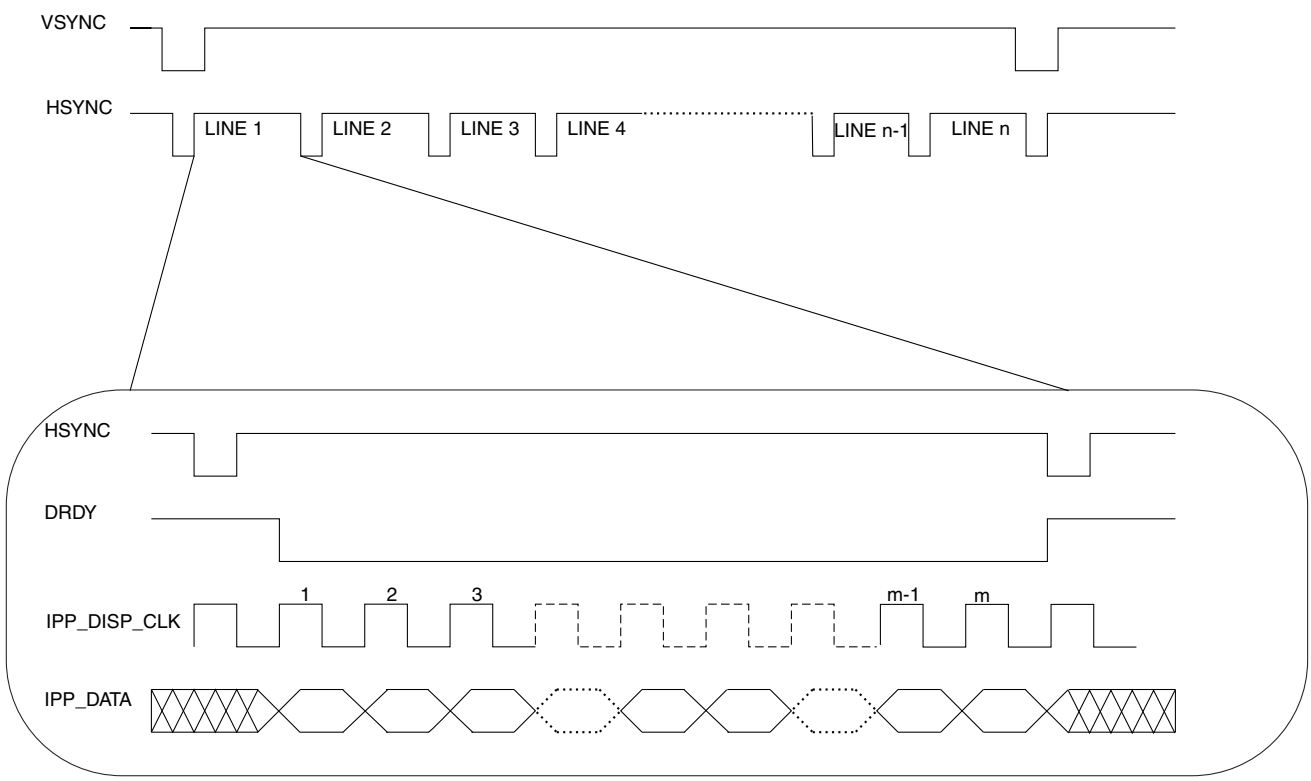


Figure 20-3. Interface Timing Diagram for TFT (Active Matrix) Panels

20.2.2 CRC Polynomial

DCIC uses the CRC32 polynomial to calculate the data signature.

Initial value for CRC calculation is 0x00000000.

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

CRC generator uses XOR-ed combination of input data bits and previous result bits.

20.2.3 Mode of operation

After DCIC is configured and enabled, providing IPU display interface is already enabled, first VSYNC is being watched. VSYNC assretion zeroizes main position counters and initializes CRC storage registers.

Each HSYNC assertion after first line in frame that holds valid pixel data (non blanking) will signal new line, i.e. increment Y-counter and zeroize X-counter. Each valid data cycle (DATA_EN asserted) will increment X-counter. Thus X & Y values used to configure the module should be the coordinates on the actual visual part of the screen without blanking intervals. X & Y position counters are constantly compared to ROIs configuration. When current position matches one of the enabled ROIs, it triggers the enablement of CRC generator, which will use the current pixel data and previous pixel CRC result for this ROI. The result of the new calculation stored separately for each ROI. Next VSYNC, apart of signalling new frame start, will cause sampling calaculated CRC values into configuration registers domain. These values then will be compared to expected signatures and appropriate status bits / interrupts will be asserted.

20.2.4 Interrupts

There are two maskable interrupts:

- Functional - Asserted when match results ready.
- Error - Asserted when there is a signature mismatch.

Both interrupts are generated immediately after completing the signature match check. Software should clean the interrupts by writing "1" to appropriate status bits (FI_STAT, ROI_MATCH_STAT). EI_STAT bit is a result of OR between all ROI_MATCH_STAT bits, hence it will be cleared automatically when these bits are clear.

Interrupt masks can be set/reset only while FREEZE_MASK bit isn't set. This bit can not be set back to zero.

20.2.5 Software reset

Disabling and enabling the module operation (IC_EN bit, DCIC Control Register (DCIC_DCICC)) will reset all the logic apart of configuration registers domain. Status bits of DCICS (DCIC Status Register (DCIC_DCICS)) register will also be cleared.

20.2.6 Clock domains

Asynchronous FIFO is being used to sample raw display interface signals and transfer them into DCIC fast clock domain (hsp_clk).

Display interface clock is always slower than DCIC fast clock.

IP registers are driven by the same fast clock and will be synchronized to system IP Bus clock by external IP_SYNC module.

20.2.7 External controller mismatch indication signal

Besides of interrupts to SoC core, DCIC provides an additional mismatch indication for external controller. The signal, if enabled, continuously oscillating at a rate which is an integer division of the main clock (hsp clock).

- When the status bit of the mismatch interrupt is set - i.e. from mismatch detection until the CPU clears the bit: division x16
- Otherwise: division x4

The indication signal is idle while integrity check is disabled.

20.2.8 Power saving

Disabling the module by clearing IC_EN bit will stop all module activities, including sampling of input signals into the FIFO.

20.2.9 System Considerations

- The DCIC always assumes a 24-bit pixel. For proper functionality with lower color depth, one must ensure that:
  - When the CPU calculates the reference signature, it applies the same mapping of a pixel to a 24-bit field as that performed by the IPU.

    - The display is connected to the appropriate pins.
    - The simplest mapping would be
        - Map the pixel to the data bus in the same way as for 24 bpp
        - Set the values at the extra LSBs to zero.
- Parameter updates
    - The reference signature can be freely updated from frame to frame, in coordination with the changing content (since it is used only once per frame, just before the interrupt)
    - Each ROI can be freely and independently enabled/disabled between frames (since the enable bit is double buffered)
    - The size and location of a ROI can be modified while it is disabled.
    - Display interface signals polarity can be changed only when the module is disabled.
- ROI (regions of interest) don't overlap, i.e. each pixel belongs to single ROI at most.
- DCIC operation is intended for monitoring relatively static portions of graphic interface, otherwise it will be complicated software task to keep up with frame synchronization and update expected signature for each frame.
- There should be no processing done at IPU on monitored ROIs, which can't be taken in consideration by software when calculating the expected signature.

20.3 DCIC Memory Map/Register Definition

Important: All write accesses have to be full word (32-bit) accesses. No error/abort will be responded in case of different access size, but no data will be written. Similarly, there will be no error response in case of access to undefined memory space.

Read access will return 32-bit data, which may be truncated on system level in case it wasn't full word acces.

DCIC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_4000	DCIC Control Register (DCIC1_DCICC)	32	R/W	0000_0070h	20.3.1/964
20E_4004	DCIC Interrupt Control Register (DCIC1_DCICIC)	32	R/W	0000_0003h	20.3.2/965
20E_4008	DCIC Status Register (DCIC1_DCICS)	32	w1c	0000_0000h	20.3.3/966
20E_4010	DCIC ROI Config Register m (DCIC1_DCICRC)	32	R/W	0000_0000h	20.3.4/967
20E_4014	DCIC ROI Size Register m (DCIC1_DCICRS)	32	R/W	0000_0000h	20.3.5/968

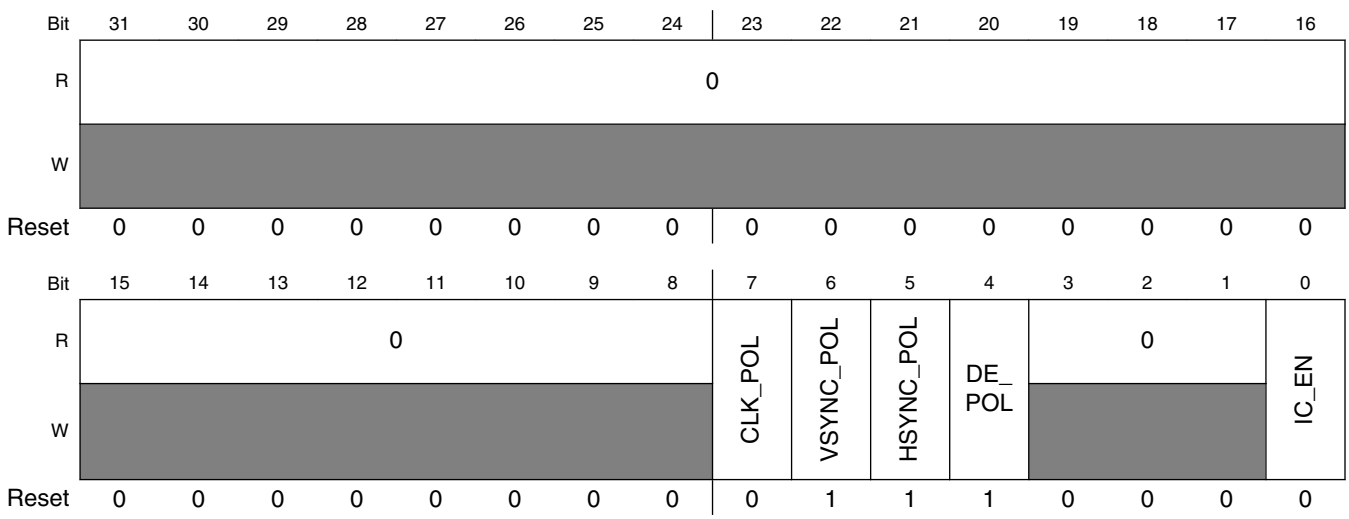
Table continues on the next page...

DCIC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_4018	DCIC ROI Reference Signature Register m (DCIC1_DCICRRS)	32	R/W	0000_0000h	20.3.6/969
20E_401C	DCIC ROI Calculated Signature m (DCIC1_DCICRCS)	32	R	0000_0000h	20.3.7/969
20E_8000	DCIC Control Register (DCIC2_DCICC)	32	R/W	0000_0070h	20.3.1/964
20E_8004	DCIC Interrupt Control Register (DCIC2_DCICIC)	32	R/W	0000_0003h	20.3.2/965
20E_8008	DCIC Status Register (DCIC2_DCICS)	32	w1c	0000_0000h	20.3.3/966
20E_8010	DCIC ROI Config Register m (DCIC2_DCICRC)	32	R/W	0000_0000h	20.3.4/967
20E_8014	DCIC ROI Size Register m (DCIC2_DCICRS)	32	R/W	0000_0000h	20.3.5/968
20E_8018	DCIC ROI Reference Signature Register m (DCIC2_DCICRRS)	32	R/W	0000_0000h	20.3.6/969
20E_801C	DCIC ROI Calculated Signature m (DCIC2_DCICRCS)	32	R	0000_0000h	20.3.7/969

20.3.1 DCIC Control Register (DCICx_DCICC)

Address: Base address + 0h offset



DCICx_DCICC field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 CLK_POL	DISP_CLK signal polarity. 0 Not inverted (default). 1 Inverted.
6 VSYNC_POL	VSYNC_IN signal polarity.

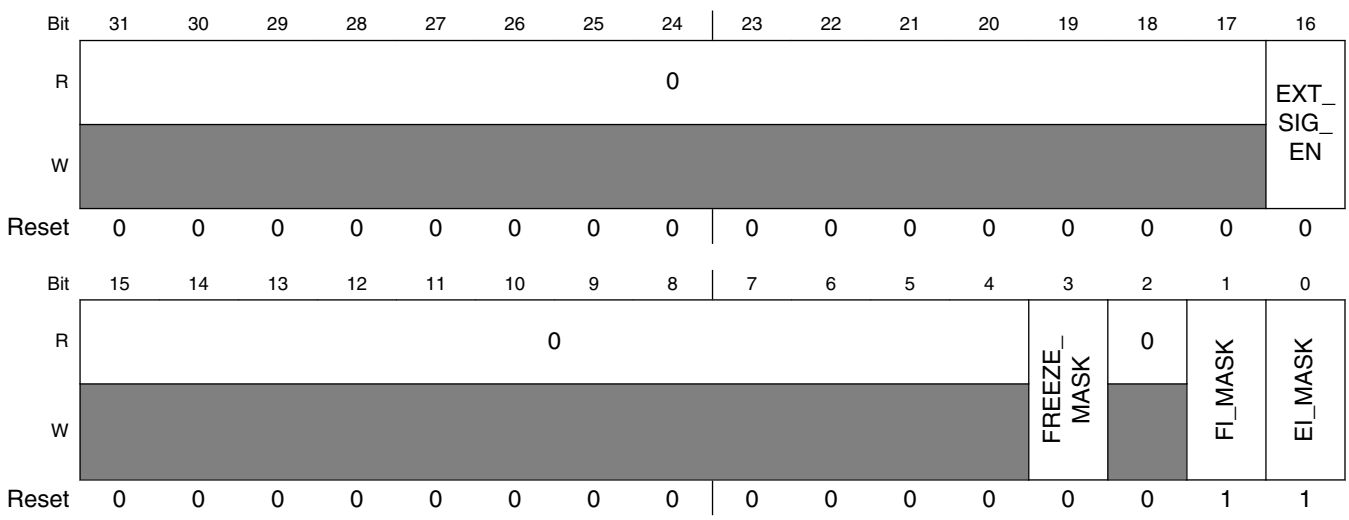
Table continues on the next page...

DCICx_DCICC field descriptions (continued)

Field	Description
	0 Active High. 1 Active Low (default).
5 HSYNC_POL	HSYNC_IN signal polarity. 0 Active High. 1 Active Low (default).
4 DE_POL	DATA_EN_IN signal polarity. 0 Active High. 1 Active Low (default).
3–1 Reserved	This read-only field is reserved and always has the value 0.
0 IC_EN	Integrity Check enable. Main enable switch. 0 Disabled 1 Enabled

20.3.2 DCIC Interrupt Control Register (DCICx_DCICIC)

Address: Base address + 4h offset



DCICx_DCICIC field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 EXT_SIG_EN	External controller mismatch indication signal. 0 Disabled (default) 1 Enabled

Table continues on the next page...

DCICx_DCICIC field descriptions (continued)

Field	Description
15–4 Reserved	This read-only field is reserved and always has the value 0.
3 FREEZE_MASK	Disable change of interrupt masks. "Sticky" bit which can be set once and cleared by reset only. 0 Masks change allowed (default) 1 Masks are frozen
2 Reserved	This read-only field is reserved and always has the value 0.
1 FI_MASK	Functional Interrupt mask. Can be changed only while FREEZE_MASK = 0. 0 Mask disabled - Interrupt assertion enabled 1 Mask enabled - Interrupt assertion disabled (default)
0 EI_MASK	Error Interrupt mask. Can be changed only while FREEZE_MASK = 0. 0 Mask disabled - Interrupt assertion enabled 1 Mask enabled - Interrupt assertion disabled (default)

20.3.3 DCIC Status Register (DCICx_DCICS)

Address: Base address + 8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0														FI_STAT	EI_STAT
W															w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

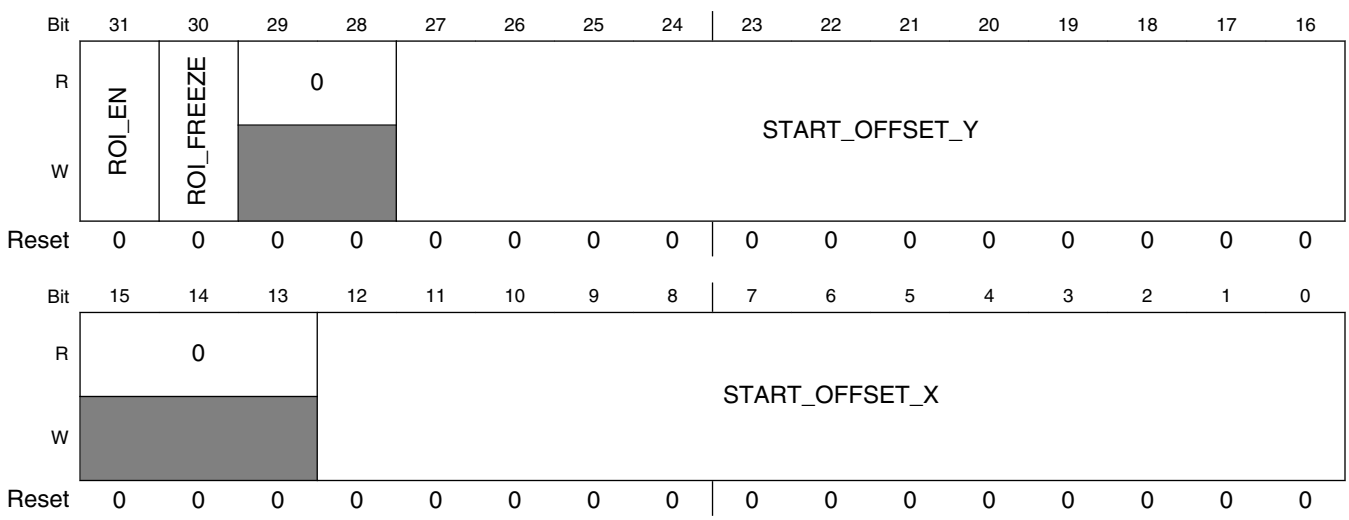
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	ROI_MATCH_STAT															
W	w1c															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DCICx_DCICS field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17 FI_STAT	Functional Interrupt status. Write "1" to clear. 0 No pending Interrupt 1 Pending Interrupt
16 EI_STAT	Error Interrupt status. Result of "OR" operation on ROI_MATCH_STAT[15:0] bits. Cleared when these bits are clear. 0 No pending Interrupt 1 Pending Interrupt
ROI_MATCH_STAT	Each set bit of this field indicates there was a mismatch at appropriate ROIs signature during the last frame. Valid only for active ROIs. Write "1" to clear. 0 ROI calculated CRC matches expected signature 1 Mismatch at ROI calculated CRC

20.3.4 DCIC ROI Config Register m (DCICx_DCICRC)

Address: Base address + 10h offset



DCICx_DCICRC field descriptions

Field	Description
31 ROI_EN	ROI #m tracking enable 0 Disabled 1 Enabled

Table continues on the next page...

DCICx_DCICRC field descriptions (continued)

Field	Description
30 ROI_FREEZE	When set, the only parameter of ROI #m that can be changed is reference signature. "Sticky" bit - can be set once and cleared by reset only. 0 ROI configuration can be changed 1 ROI configuration is frozen
29–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 START_ OFFSET_Y	Row number of ROIs upper-left corner (Y coordinate) Range: 0 to 2^12-1
15–13 Reserved	This read-only field is reserved and always has the value 0.
START_ OFFSET_X	Column number of ROIs upper-left corner (X coordinate) Range: 0 to 2^13-1

20.3.5 DCIC ROI Size Register m (DCICx_DCICRS)

Address: Base address + 14h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				END_OFFSET_Y												0			END_OFFSET_X											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DCICx_DCICRS field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 END_OFFSET_Y	Row number of ROIs lower-right corner (Y coordinate) Range: 1 to 2^12-1
15–13 Reserved	This read-only field is reserved and always has the value 0.
END_OFFSET_X	Column number of ROIs lower-right corner (X coordinate) Range: 1 to 2^13-1

20.3.6 DCIC ROI Reference Signature Register m (DCICx_DCICRRS)

Address: Base address + 18h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	REFERENCE_SIGNATURE																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DCICx_DCICRRS field descriptions

Field	Description
REFERENCE_ SIGNATURE	32-bit expected signature (CRC calculation result) for ROI #m

20.3.7 DCIC ROI Calculated Signature m (DCICx_DCICRCS)

Address: Base address + 1Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CALCULATED_SIGNATURE																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DCICx_DCICRCS field descriptions

Field	Description
CALCULATED_ SIGNATURE	32-bit actual signature (CRC calculation result) for ROI #m during the last frame. Updated automatically at the beginning of a next frame.

Chapter 21
Enhanced Configurable SPI (ECSPI)

21.1 Overview

The Enhanced Configurable Serial Peripheral Interface (ECSPI) is a full-duplex, synchronous, four-wire serial communication block.

The ECSPI contains a 64 x 32 receive buffer (RXFIFO) and a 64 x 32 transmit buffer (TXFIFO). With data FIFOs, the ECSPI allows rapid data communication with fewer software interrupts. The figure below shows a block diagram of the ECSPI.

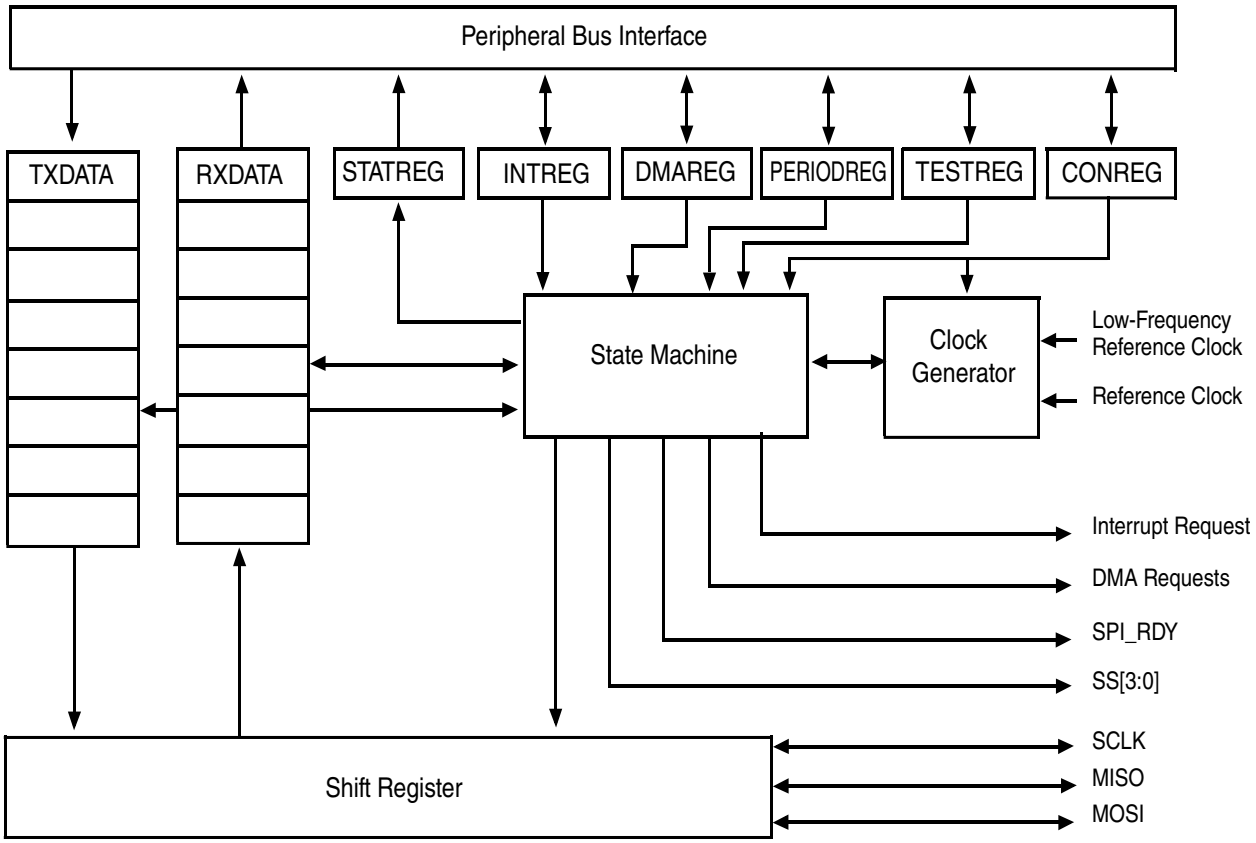


Figure 21-1. ECSPI Block Diagram

21.1.1 Features

Key features of the ECSPI include:

- Full-duplex synchronous serial interface
- Master/Slave configurable
- Four Chip Select (SS) signals to support multiple peripherals
- Transfer continuation function allows unlimited length data transfers
- 32-bit wide by 64-entry FIFO for both transmit and receive data
- Polarity and phase of the Chip Select (SS) and SPI Clock (SCLK) are configurable
- Direct Memory Access (DMA) support
- Max operation frequency up to the reference clock frequency.

21.1.2 Modes and Operations

The ECSPI supports the modes described in the indicated sections:

- Master Mode
- Slave Mode
- Low Power Modes

As described in Operations, the ECSPI supports the operations described in the indicated sections:

- Typical Master Mode
  - Master Mode with SPI_RDY
  - Master Mode with Wait States
  - Master Mode with SS_CTL[3:0] Control
  - Master Mode with Phase Control
- Typical Slave Mode

21.2 External Signals

The following table describes the external signals of ECSPI:

Table 21-1. ECSPI1 External Signals

Signal	Description	Pad	Mode	Direction
ECSPI1_MISO (MISO)	Master data in; slave data out	CSI0_DAT6	ALT2	IO
		DISP0_DAT22	ALT2	
		EIM_D17	ALT1	
		KEY_COL1	ALT0	
ECSPI1_MOSI (MOSI)	Master data out; slave data in	CSI0_DAT5	ALT2	IO
		DISP0_DAT21	ALT2	
		EIM_D18	ALT1	
		KEY_ROW0	ALT0	
ECSPI1_RDY (RDY)	SPI data ready signal	GPIO_19	ALT4	I
ECSPI1_SCLK (SCLK)	SPI clock signal	CSI0_DAT4	ALT2	IO
		DISP0_DAT20	ALT2	
		EIM_D16	ALT1	
		KEY_COL0	ALT0	
ECSPI1_SS0 (SS0)	Chip select signal	CSI0_DAT7	ALT2	IO
		DISP0_DAT23	ALT2	
		EIM_EB2	ALT1	
		KEY_ROW1	ALT0	

Table continues on the next page...

Table 21-1. ECSPI1 External Signals (continued)

Signal	Description	Pad	Mode	Direction
ECSPI1_SS1 (SS1)	Chip select signal	DISP0_DAT15	ALT2	IO
		EIM_D19	ALT1	
		KEY_COL2	ALT0	
ECSPI1_SS2 (SS2)	Chip select signal	EIM_D24	ALT3	IO
		KEY_ROW2	ALT0	
ECSPI1_SS3 (SS3)	Chip select signal	EIM_D25	ALT3	IO
		KEY_COL3	ALT0	

Table 21-2. ECSPI2 External Signals

Signal	Description	Pad	Mode	Direction
ECSPI2_MISO (MISO)	Master data in; slave data out	CSI0_DAT10	ALT2	IO
		DISP0_DAT17	ALT2	
		EIM_OE	ALT2	
ECSPI2_MOSI (MOSI)	Master data out; slave data in	CSI0_DAT9	ALT2	IO
		DISP0_DAT16	ALT2	
		EIM_CS1	ALT2	
ECSPI2_RDY (RDY)	SPI data ready signal	EIM_A25	ALT2	I
ECSPI2_SCLK (SCLK)	SPI clock signal	CSI0_DAT8	ALT2	IO
		DISP0_DAT19	ALT2	
		EIM_CS0	ALT2	
ECSPI2_SS0 (SS0)	Chip select signal	CSI0_DAT11	ALT2	IO
		DISP0_DAT18	ALT2	
		EIM_RW	ALT2	
ECSPI2_SS1 (SS1)	Chip select signal	DISP0_DAT15	ALT3	IO
		EIM_LBA	ALT2	
ECSPI2_SS2 (SS2)	Chip select signal	EIM_D24	ALT4	IO
ECSPI2_SS3 (SS3)	Chip select signal	EIM_D25	ALT4	IO

Table 21-3. ECSPI3 External Signals

Signal	Description	Pad	Mode	Direction
ECSPI3_MISO (MISO)	Master data in; slave data out	DISP0_DAT2	ALT2	IO
ECSPI3_MOSI (MOSI)	Master data out; slave data in	DISP0_DAT1	ALT2	IO
ECSPI3_RDY (RDY)	SPI data ready signal	DISP0_DAT7	ALT2	I
ECSPI3_SCLK (SCLK)	SPI clock signal	DISP0_DAT0	ALT2	IO
ECSPI3_SS0 (SS0)	Chip select signal	DISP0_DAT3	ALT2	IO
ECSPI3_SS1 (SS1)	Chip select signal	DISP0_DAT4	ALT2	IO

Table continues on the next page...

Table 21-3. ECSPI3 External Signals (continued)

Signal	Description	Pad	Mode	Direction
ECSPI3_SS2 (SS2)	Chip select signal	DISP0_DAT5	ALT2	IO
ECSPI3_SS3 (SS3)	Chip select signal	DISP0_DAT6	ALT2	IO

Table 21-4. ECSPI4 External Signals

Signal	Description	Pad	Mode	Direction
ECSPI4_MISO (MISO)	Master data in; slave data out	EIM_D22	ALT1	IO
ECSPI4_MOSI (MOSI)	Master data out; slave data in	EIM_D28	ALT2	IO
ECSPI4_RDY (RDY)	SPI data ready signal	EIM_EB3	ALT1	I
ECSPI4_SCLK (SCLK)	SPI clock signal	EIM_D21	ALT1	IO
ECSPI4_SS0 (SS0)	Chip select signal	EIM_D20	ALT1	IO
		EIM_D29	ALT2	
ECSPI4_SS1 (SS1)	Chip select signal	EIM_A25	ALT1	IO
ECSPI4_SS2 (SS2)	Chip select signal	EIM_D24	ALT1	IO
ECSPI4_SS3 (SS3)	Chip select signal	EIM_D25	ALT1	IO

Table 21-5. ECSPI5 External Signals

Signal	Description	Pad	Mode	Direction
ECSPI5_MISO (MISO)	Master data in; slave data out	SD1_DAT0	ALT1	IO
		SD2_DAT0	ALT1	
ECSPI5_MOSI (MOSI)	Master data out; slave data in	SD1_CMD	ALT1	IO
		SD2_CMD	ALT1	
ECSPI5_RDY (RDY)	SPI data ready signal	GPIO_7	ALT1	I
ECSPI5_SCLK (SCLK)	SPI clock signal	SD1_CLK	ALT1	IO
		SD2_CLK	ALT1	
ECSPI5_SS0 (SS0)	Chip select signal	SD1_DAT1	ALT1	IO
		SD2_DAT1	ALT1	
ECSPI5_SS1 (SS1)	Chip select signal	SD1_DAT2	ALT1	IO
		SD2_DAT2	ALT1	
ECSPI5_SS2 (SS2)	Chip select signal	SD1_DAT3	ALT1	IO
ECSPI5_SS3 (SS3)	Chip select signal	SD2_DAT3	ALT1	IO

Figure 21-2 shows the ECSPI in master mode connected to four external devices in a one-way communication link.

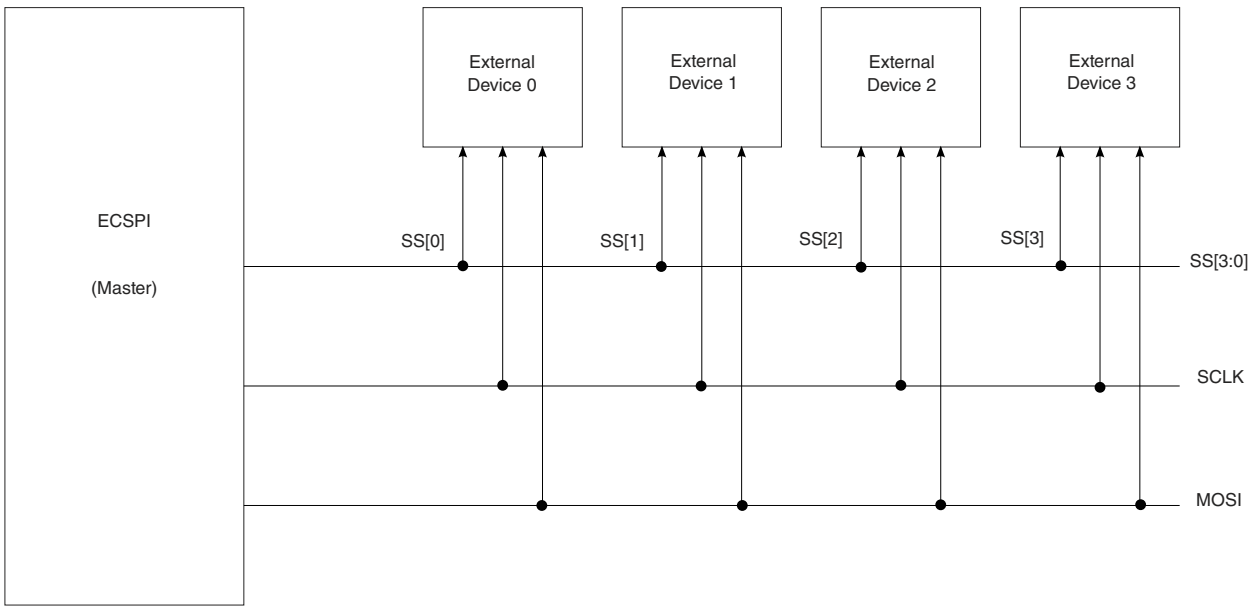


Figure 21-2. Example Connection Diagram

21.3 Clocks

The following table describes the clock sources for eCSPI. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 21-6. eCSPI Clocks

Clock name	Clock Root	Description
ipg_clk	ipg_clk_root	Peripheral clock
ipg_clk_32k	ckil_sync_clk_root	Low-frequency reference clock (32kHz)
ipg_clk_per	ecspi_clk_root	eCSPI module clock
ipg_clk_s	ipg_clk_root	Peripheral access clock

21.4 Functional Description

This section provides a complete functional description of the ECSPI. The figure found here shows the relationship of SCLK and data lines while ECSPI has been configured with different POL and PHA settings.

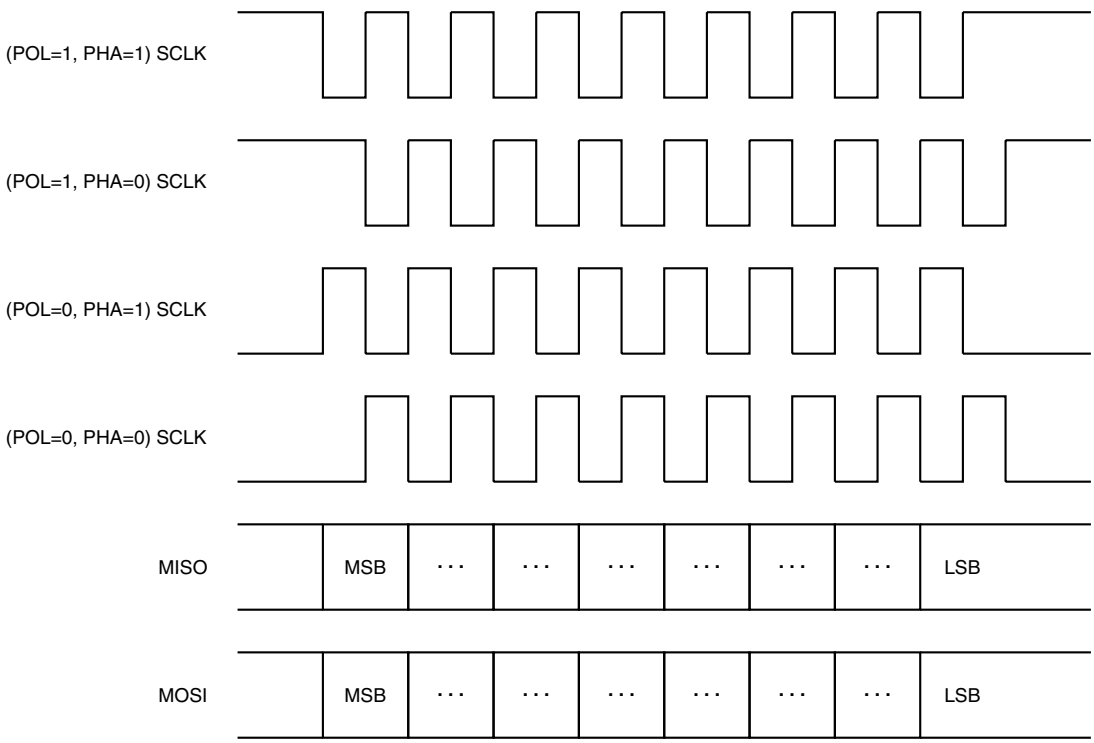


Figure 21-3. ECSPI SCLK, MISO, and MOSI Relationship

21.4.1 Master Mode

When the ECSPI is configured as a master, it uses a serial link to transfer data between the ECSPI and an external slave device.

One of the Chip Select (SS) signals and the clock signal (SCLK) are used to transfer data between two devices. If the external device is a transmit-only device, the ECSPI master's output port can be ignored and used for other purposes. In order to use the internal TXFIFO and RXFIFO, two auxiliary output signals, Chip Select (SS) and SPI_RDY, are used for data transfer rate control. Software can also configure the sample period control register to a fixed data transfer rate.

21.4.2 Slave Mode

When the ECSPI is configured as a slave, software can configure the ECSPI Control register to match the external SPI master's timing.

In this configuration, Chip Select ($\overline{SS}$) becomes an input signal, and is used to control data transfers through the Shift register, as well as to load/store the data FIFO.

Slave mode only supports the case when SSCTL (SSB_CTRL[x] bit) is cleared. The accurate burst length should always be specified using the BURST_LENGTH parameter. SSCTL (SSB_CTRL[x] bit) set to 1 is not supported in slave mode.

21.4.3 Low Power Modes

The ECSPI does not operate under low power mode.

It holds its operation when its clock is gated off in master mode. In slave mode, the ECSPI does not respond when its clock is gated off.

21.4.4 Operations

The information found here describes the ECSPI's operations.

21.4.4.1 Typical Master Mode

The ECSPI master uses the Chip Select (SS) signal to enable an external SPI device, and uses the SCLK signal to transfer data in and out of the Shift register.

The SPI_RDY enables fast data communication with fewer software interrupts. By programming the ECSPI_PERIODREG register accordingly, the ECSPI can be used for a fixed data transfer rate.

When the ECSPI is in Master mode the SS, SCLK, and MOSI are output signals, and the MISO signal is an input.

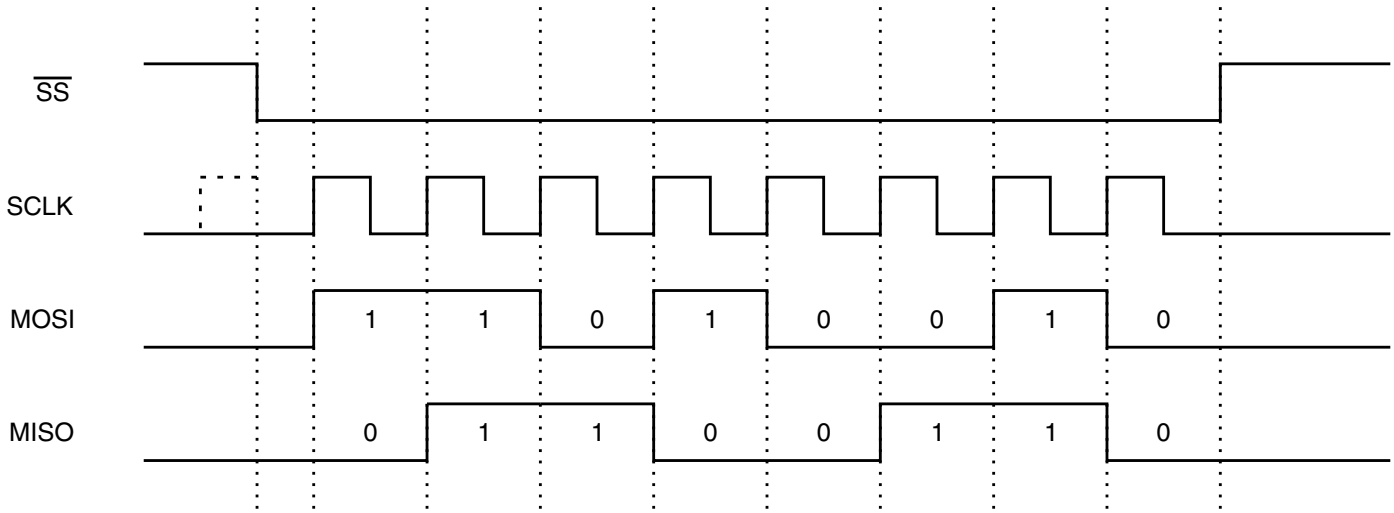


Figure 21-4. Typical SPI Burst (8-bit Transfer)

In the above figure, the Chip Select (SS) signal enables the selected external SPI device, and the SCLK synchronizes the data transfer. The MOSI and MISO signals change on rising edge of SCLK and the MISO signal is latched on the falling edge of the SCLK. The figure above shows a data of 0xD2 is shifted out, and a data of 0x66 is shifted in.

21.4.4.1.1 Master Mode with SPI_RDY

By default, the ECSPI does not use the SPI_RDY signal in master mode (MODE =1).

A SPI burst begins when the following events happen:

- The ECSPI is enabled, TXFIFO has data in it, and ECSPI_CONREG[XCH] bit or the ECSPI_CONREG[SMC] bit is set.
- When the SPI Data Ready Control (ECSPI_CONREG[DRCTL]) bits contains either 01 or 10, the SPI_RDY signal controls when a SPI burst starts.

A SPI burst is defined as a bus transaction that starts when the slave select is asserted and ends when the slave select is negated. The Chip Select (SS) signal will remain asserted until all the bits in a SPI burst are shifted out.

If ECSPI_CONREG[DRCTL] is set to 01, the SPI burst can be triggered only if a falling edge of the SPI_RDY signal has been detected.

The following figure shows the relationship between a SPI burst and the falling edge of SPI_RDY signal.

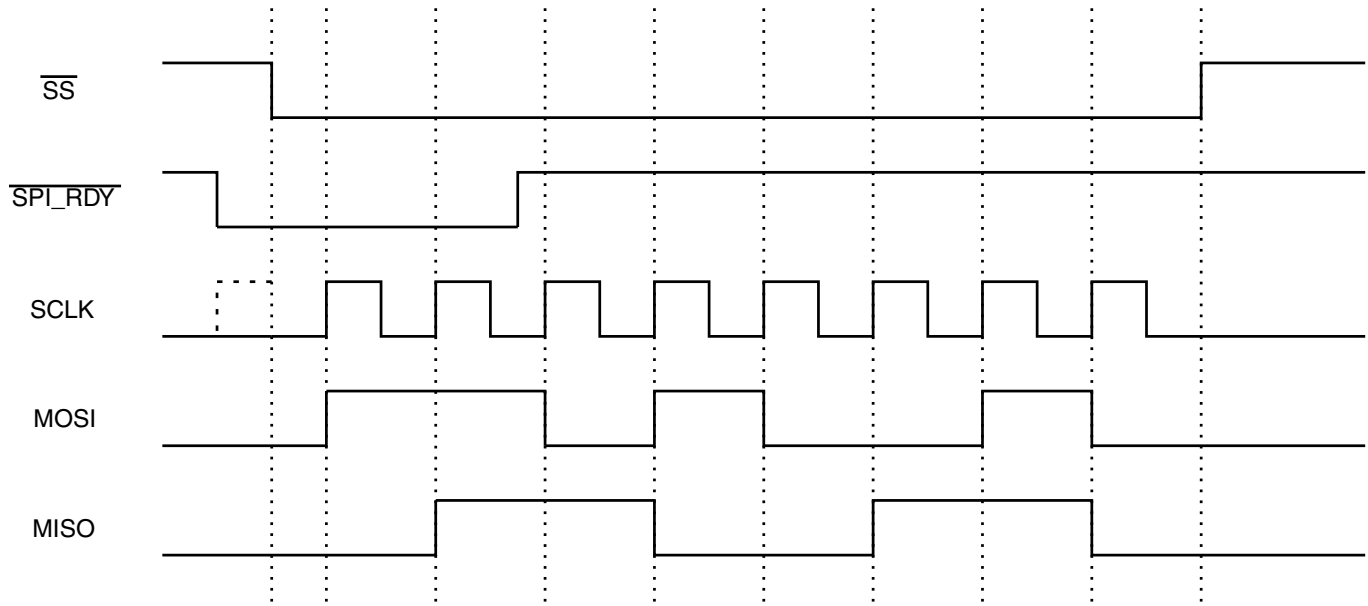


Figure 21-5. Relationship Between a SPI Burst and SPI_RDY: Falling-Edge Triggered

A SPI burst does not start until the falling edge of the SPI_RDY signal is detected. The next SPI burst starts when the next SPI_RDY falling edge is detected, after the last burst has finished.

If SPI Data Ready Control (ECSPI_CONREG[DRCTL]) is set to 10, the SPI burst can be triggered only if the SPI_RDY signal is low.

The following figure shows the relationship between a SPI burst and the SPI_RDY signal. The SPI burst does not begin until the SPI_RDY signal goes low. The ECSPI will keep transmitting SPI burst if the SPI_RDY signal remains low.

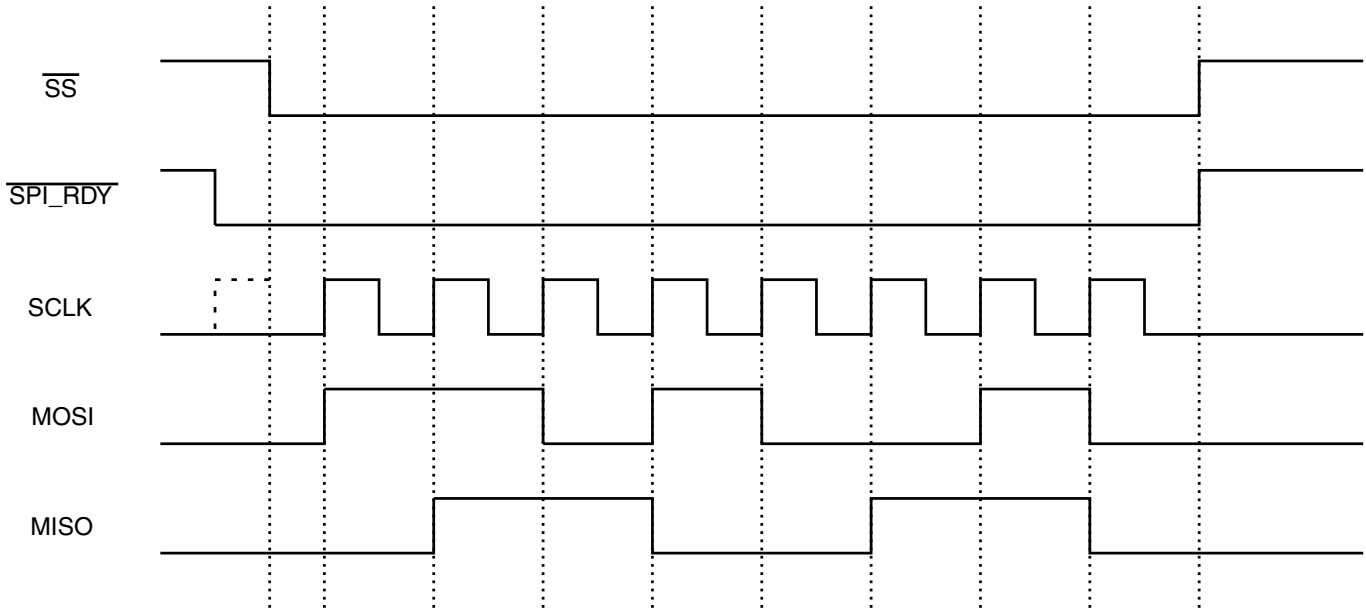


Figure 21-6. Relationship Between a SPI Burst and SPI_RDY: Low-Level Triggered

21.4.4.1.2 Master Mode with Wait States

Wait states can be inserted between SPI bursts. This provides a way for software to slow down the SPI burst to meet the timing requirements of a slower SPI device.

The following figure shows wait states inserted between SPI bursts.

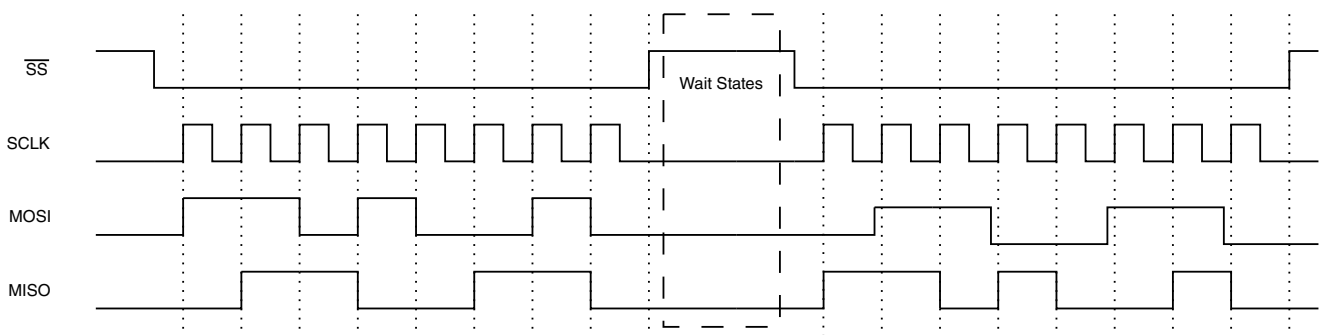


Figure 21-7. SPI Bursts with Wait States

In this case, the number of wait states is controlled by ECSPI_PERIODREG[SAMPLE PERIOD] and the wait states' clock source is selected by ECSPI_PERIODREG[CSRC].

21.4.4.1.3 Master Mode with SS_CTL[3:0] Control

The SPI SS Control (SS_CTL[3:0]) controls whether the current operation is single burst or multiple bursts.

When the SPI SS Wave Form Select (SS_CTL[3:0]) is set, the current operation is multiple bursts transfer. When the SPI SS Wave Form Select (SS_CTL[3:0]) bit is cleared, the current operation is single burst transfer. A SPI burst can contains multiple words as defined in the BURST LENGTH field of the ECSPI_CONREG register.

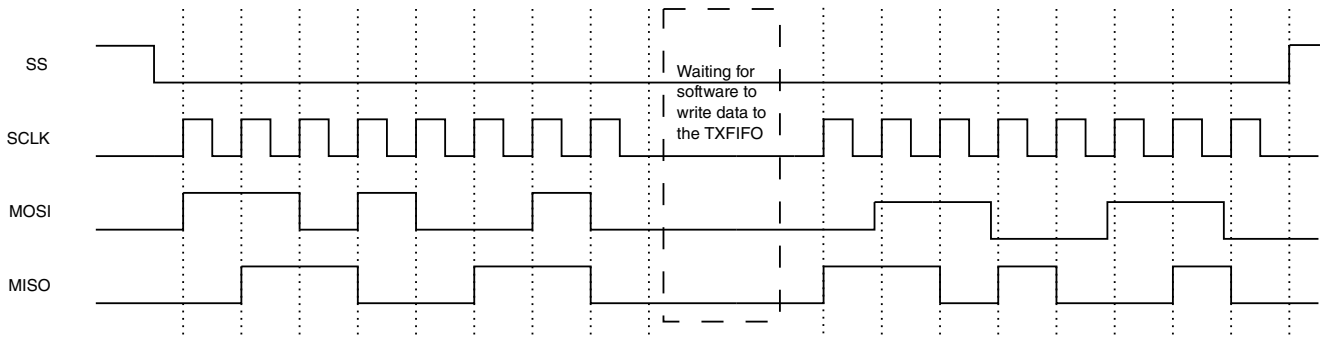


Figure 21-8. SPI Burst While SS_CTL[3:0] is Clear

In Figure 21-8, two 8-bit bursts in the TXFIFO have been combined and transmitted in one SPI burst. The maximum length of a single SPI burst is defined in the BURST LENGTH field of the ECSPI_CONREG control register. (Figure 21-8 corresponds to a BURST LENGTH of 8.) This provides a way for transferring a longer SPI burst by writing data into TXFIFO while the ECSPI is transmitting.

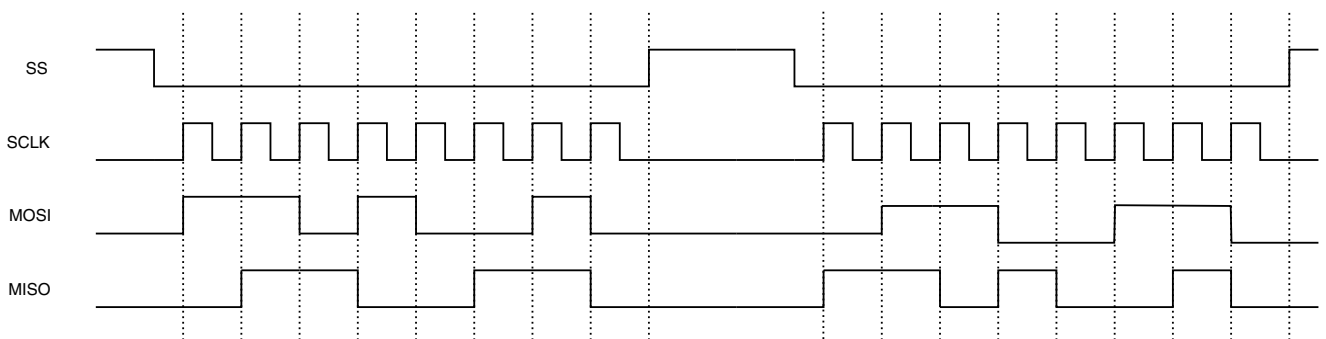


Figure 21-9. SPI Bursts While SS_CTL[3:0] is Set

In Figure 21-9, two FIFO entries are transmitted, one entry with each SPI burst. The ECSPI will continue to transmit SPI bursts until the TXFIFO is empty. When wait states can be inserted between SPI bursts, the SS will negate between SPI bursts until the wait states finish.

21.4.4.1.4 Master Mode with Phase Control

The Phase Control (ECSPI_CONREG[PHA]) bit controls how the transmit data shifts out and the receive data shifts in.

When the Phase control (ECSPI_CONREG[PHA]) bit is set, the transmit data will shift out on the rising edge of SCLK, and the receive data is latched on the falling edge of SCLK. The most-significant bit is output on the first rising SCLK edge.

When ECSPI_CONREG[PHA] is cleared, the transmit data is shifted out on the falling edge of SCLK and the receive data is latched on the rising edge of SCLK. The MSB is output when the host processor loads the transmitted data.

Inverting the SCLK polarity does not impact the edge-triggered operations because they are internal to the serial peripheral interface master. Figure 21-10 shows how SPI burst works with different POL and PHA configuration.

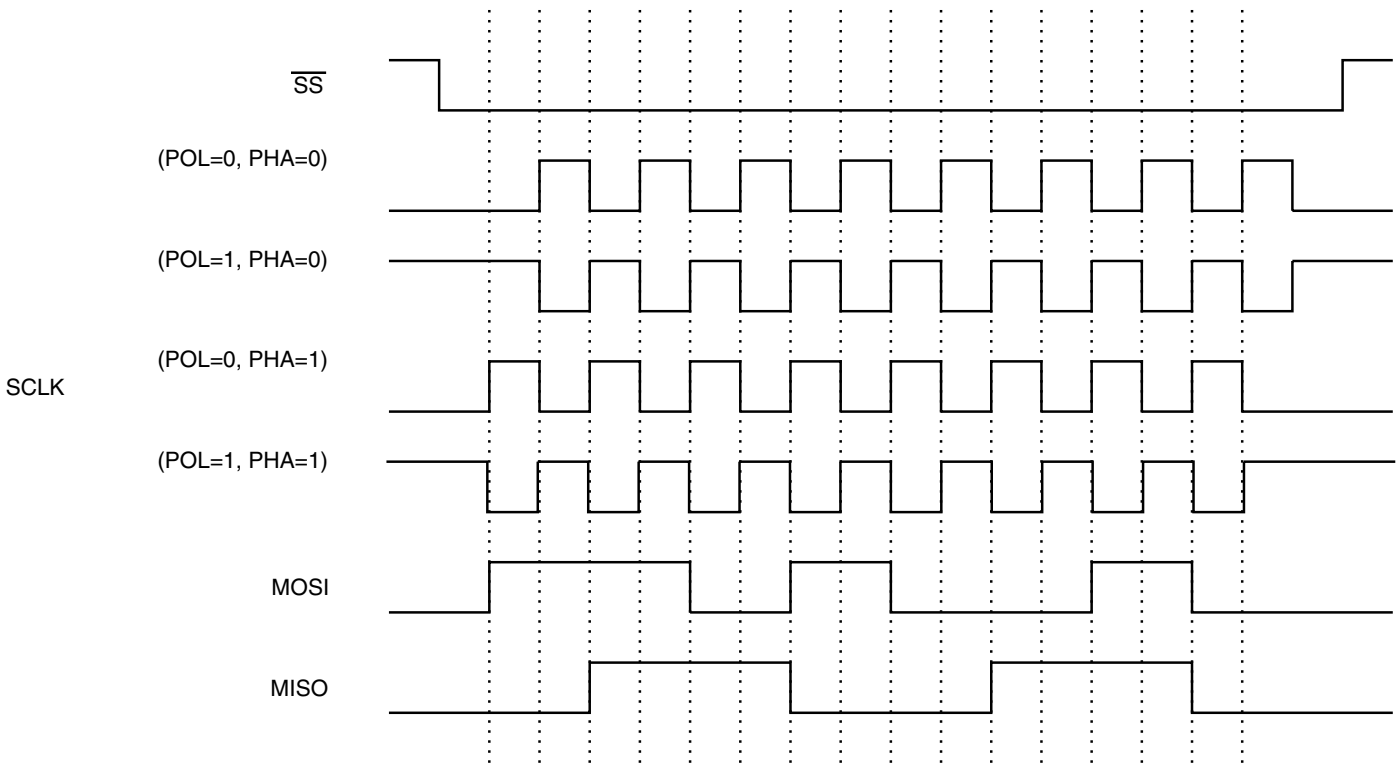


Figure 21-10. SPI Burst with Different POL and PHA Configurations

21.4.4.2 Typical Slave Mode

When the ECSPI is configured as a slave (Mode = 0), software can configure the ECSPI Control register to match the external SPI master's timing. In this configuration, SS becomes an input signal, and is used to latch data in and out of the internal data Shift registers, as well as to advance the data FIFO.

The SS, SCLK, and MOSI are inputs and MISO is output. Most of the timing diagrams are similar to the diagrams shown previously for the SPI in Master mode (Mode = 1), because the inputs come from a SPI master device.

However, the timing is different when SS is used to advance the data FIFO. When the SS_CTL[3:0] is set while the ECSPI is configured in Slave mode, the data FIFO will advance on the rising edge of the SS signal. When the polarity is reversed (SSPOL = 1), the data FIFO will advance on the falling edge of the SS signal.

The figure below shows a SPI burst in which the data FIFO is advanced by the rising edge of the SS signal.

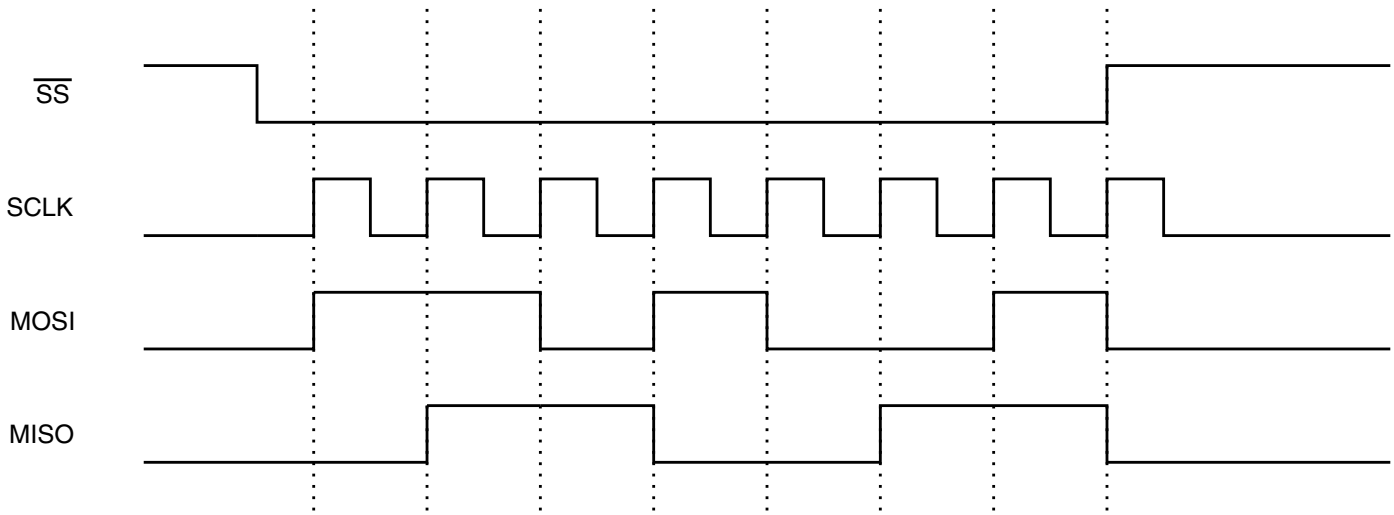


Figure 21-11. Advancing the Data FIFO on the Rising Edge of $\overline{SS}$

In the above case, only the most significant 7 bits are loaded to the RXFIFO.

21.4.5 Reset

Whenever a device reset occurs, a reset is performed on the ECSPI, resetting all registers to their default values.

Software can reset the block using the CONREG[EN] bit; see ECSPI.

21.4.6 Interrupts

Interrupt control provides a way to manage the ECSPI FIFOs:

- For transmitting data, software can enable the TXFIFO empty, TXFIFO data request, and TXFIFO full interrupts to maintain the TXFIFO using an interrupt service routine.
- For receiving data, software can enable the RXFIFO ready, RXFIFO data request, and RXFIFO full interrupts to retrieve data from the RXFIFO using an interrupt service routine.

Other interrupt sources can be used to control or debug the SPI bursts:

- The transfer-completed interrupt means that there is no data left in the TXFIFO and that the data in the Shift register has been shifted out.
- The RXFIFO overflow interrupt means that the RXFIFO received more than 64 words and will not accept any other words.

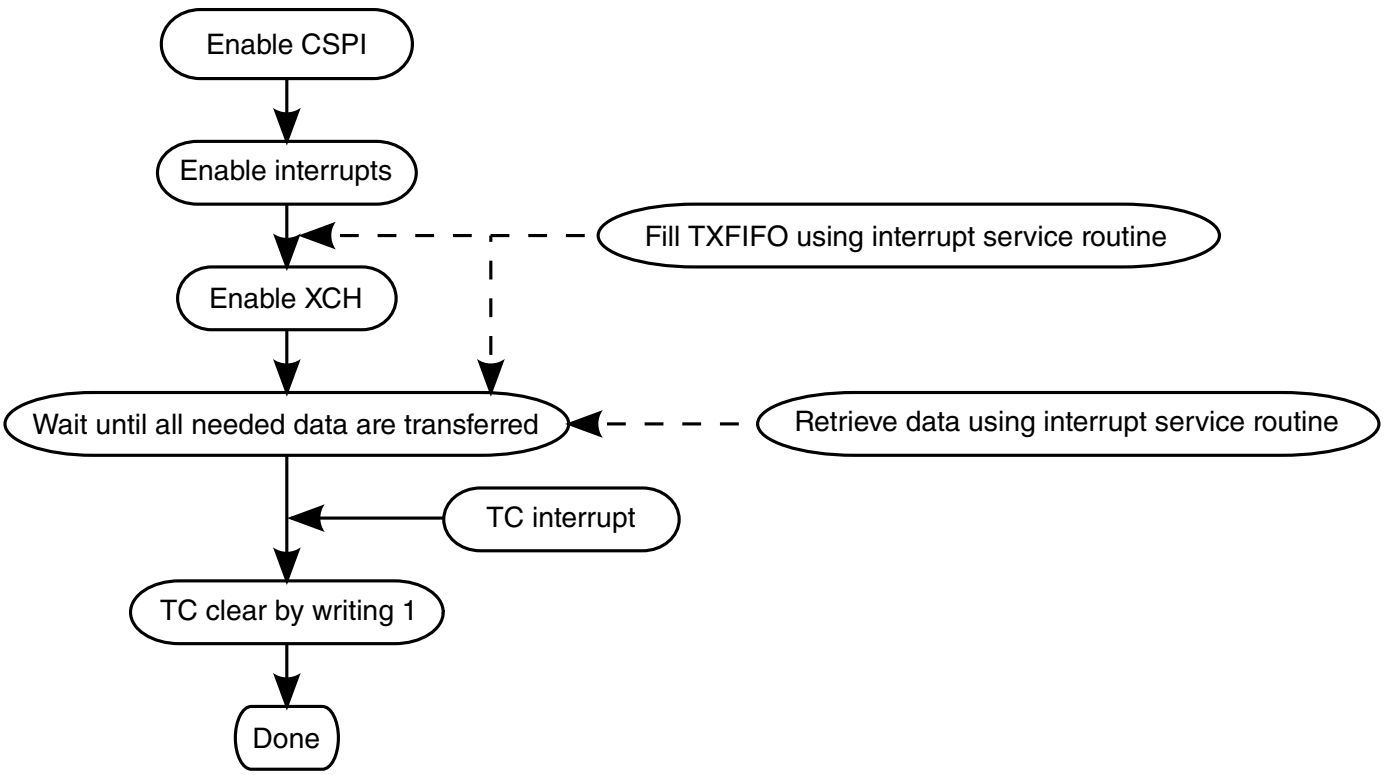


Figure 21-12. Program Sequence of SPI Burst Using Interrupt Control

21.4.7 DMA

DMA control provides another method to utilize the FIFOs in the ECSPI. By using DMA request and acknowledge signals, larger amounts of data can be transferred, and will reduce interrupts and host processor loading. When the appropriate conditions are matched, the block will send out a DMA request.

The DMA can deal with the following conditions:

- TXFIFO empty
- TXFIFO data request
- RXFIFO data request
- RXFIFO full

The figure below shows a program sequence of SPI bursts using DMA control.

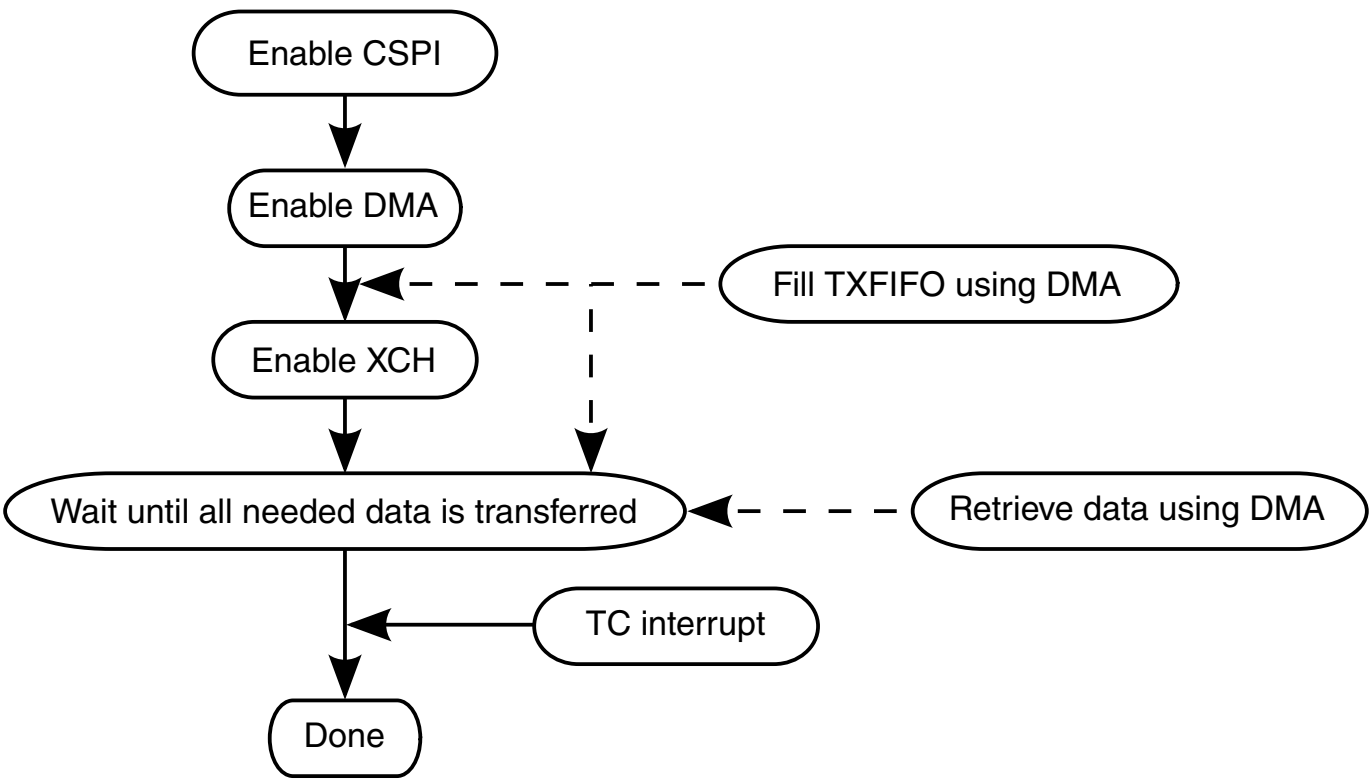


Figure 21-13. Program Sequence of SPI Burst Using DMA

21.4.8 Byte Order

The ECSPI does not support byte re-ordering in hardware.

21.5 Initialization

This section provides initialization information for ECSPI.

To initialize the block:

1. Clear the EN bit in ECSPI_CONREG to reset the block.
2. Enable the clocks for ECSPI.
3. Set the EN bit in ECSPI_CONREG to put ECSPI out of reset.
4. Configure corresponding IOMUX for ECSPI external signals.
5. Configure registers of ECSPI properly according to the specifications of the external SPI device.

21.6 Applications

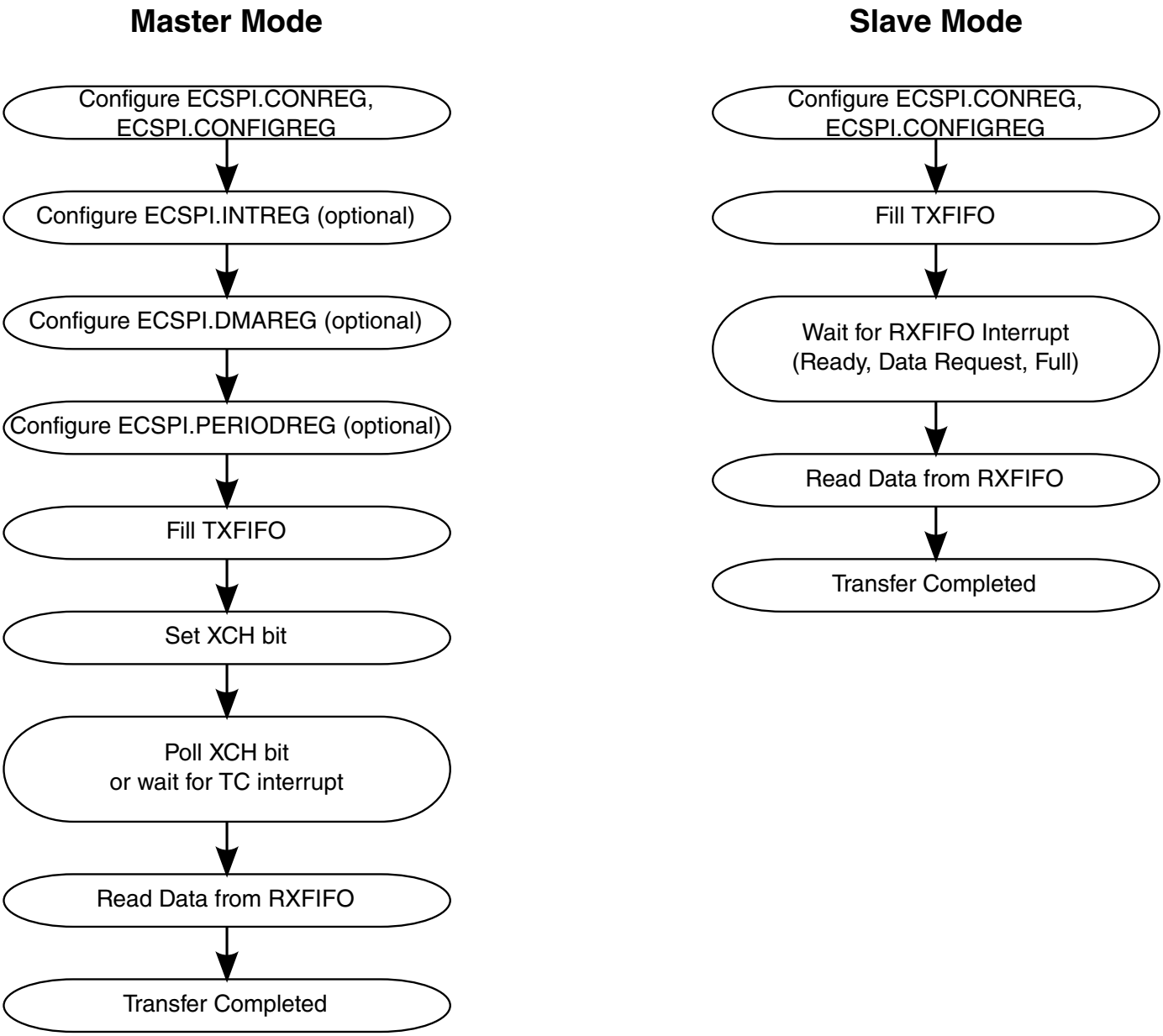


Figure 21-14. Flowchart of the ECSPI Operation

21.7 ECSPI Memory Map/Register Definition

This section includes the block memory map and detailed descriptions of all registers. For the base address of a particular block instantiation, see the system memory map.

ECSPI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
200_8000	Receive Data Register (ECSPI1_RXDATA)	32	R	0000_0000h	21.7.1/990
200_8004	Transmit Data Register (ECSPI1_TXDATA)	32	W	0000_0000h	21.7.2/991
200_8008	Control Register (ECSPI1_CONREG)	32	R/W	0000_0000h	21.7.3/991
200_800C	Config Register (ECSPI1_CONFIGREG)	32	R/W	0000_0000h	21.7.4/994
200_8010	Interrupt Control Register (ECSPI1_INTREG)	32	R/W	0000_0000h	21.7.5/996
200_8014	DMA Control Register (ECSPI1_DMAREG)	32	R/W	0000_0000h	21.7.6/997
200_8018	Status Register (ECSPI1_STATREG)	32	R/W	0000_0003h	21.7.7/999
200_801C	Sample Period Control Register (ECSPI1_PERIODREG)	32	R/W	0000_0000h	21.7.8/1000
200_8020	Test Control Register (ECSPI1_TESTREG)	32	R/W	0000_0000h	21.7.9/1002
200_8040	Message Data Register (ECSPI1_MSGDATA)	32	W	0000_0000h	21.7.10/ 1003
200_C000	Receive Data Register (ECSPI2_RXDATA)	32	R	0000_0000h	21.7.1/990
200_C004	Transmit Data Register (ECSPI2_TXDATA)	32	W	0000_0000h	21.7.2/991
200_C008	Control Register (ECSPI2_CONREG)	32	R/W	0000_0000h	21.7.3/991
200_C00C	Config Register (ECSPI2_CONFIGREG)	32	R/W	0000_0000h	21.7.4/994
200_C010	Interrupt Control Register (ECSPI2_INTREG)	32	R/W	0000_0000h	21.7.5/996
200_C014	DMA Control Register (ECSPI2_DMAREG)	32	R/W	0000_0000h	21.7.6/997
200_C018	Status Register (ECSPI2_STATREG)	32	R/W	0000_0003h	21.7.7/999
200_C01C	Sample Period Control Register (ECSPI2_PERIODREG)	32	R/W	0000_0000h	21.7.8/1000
200_C020	Test Control Register (ECSPI2_TESTREG)	32	R/W	0000_0000h	21.7.9/1002
200_C040	Message Data Register (ECSPI2_MSGDATA)	32	W	0000_0000h	21.7.10/ 1003
201_0000	Receive Data Register (ECSPI3_RXDATA)	32	R	0000_0000h	21.7.1/990
201_0004	Transmit Data Register (ECSPI3_TXDATA)	32	W	0000_0000h	21.7.2/991
201_0008	Control Register (ECSPI3_CONREG)	32	R/W	0000_0000h	21.7.3/991
201_000C	Config Register (ECSPI3_CONFIGREG)	32	R/W	0000_0000h	21.7.4/994
201_0010	Interrupt Control Register (ECSPI3_INTREG)	32	R/W	0000_0000h	21.7.5/996
201_0014	DMA Control Register (ECSPI3_DMAREG)	32	R/W	0000_0000h	21.7.6/997
201_0018	Status Register (ECSPI3_STATREG)	32	R/W	0000_0003h	21.7.7/999
201_001C	Sample Period Control Register (ECSPI3_PERIODREG)	32	R/W	0000_0000h	21.7.8/1000
201_0020	Test Control Register (ECSPI3_TESTREG)	32	R/W	0000_0000h	21.7.9/1002
201_0040	Message Data Register (ECSPI3_MSGDATA)	32	W	0000_0000h	21.7.10/ 1003
201_4000	Receive Data Register (ECSPI4_RXDATA)	32	R	0000_0000h	21.7.1/990
201_4004	Transmit Data Register (ECSPI4_TXDATA)	32	W	0000_0000h	21.7.2/991
201_4008	Control Register (ECSPI4_CONREG)	32	R/W	0000_0000h	21.7.3/991
201_400C	Config Register (ECSPI4_CONFIGREG)	32	R/W	0000_0000h	21.7.4/994
201_4010	Interrupt Control Register (ECSPI4_INTREG)	32	R/W	0000_0000h	21.7.5/996
201_4014	DMA Control Register (ECSPI4_DMAREG)	32	R/W	0000_0000h	21.7.6/997

Table continues on the next page...

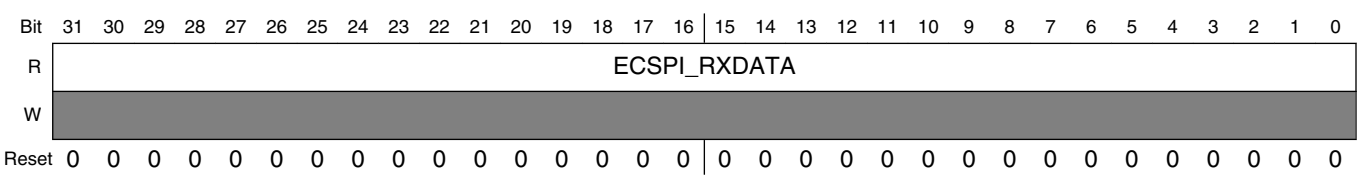
ECSPI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
201_4018	Status Register (ECSPI4_STATREG)	32	R/W	0000_0003h	21.7.7/999
201_401C	Sample Period Control Register (ECSPI4_PERIODREG)	32	R/W	0000_0000h	21.7.8/1000
201_4020	Test Control Register (ECSPI4_TESTREG)	32	R/W	0000_0000h	21.7.9/1002
201_4040	Message Data Register (ECSPI4_MSGDATA)	32	W	0000_0000h	21.7.10/ 1003
201_8000	Receive Data Register (ECSPI5_RXDATA)	32	R	0000_0000h	21.7.1/990
201_8004	Transmit Data Register (ECSPI5_TXDATA)	32	W	0000_0000h	21.7.2/991
201_8008	Control Register (ECSPI5_CONREG)	32	R/W	0000_0000h	21.7.3/991
201_800C	Config Register (ECSPI5_CONFIGREG)	32	R/W	0000_0000h	21.7.4/994
201_8010	Interrupt Control Register (ECSPI5_INTREG)	32	R/W	0000_0000h	21.7.5/996
201_8014	DMA Control Register (ECSPI5_DMAREG)	32	R/W	0000_0000h	21.7.6/997
201_8018	Status Register (ECSPI5_STATREG)	32	R/W	0000_0003h	21.7.7/999
201_801C	Sample Period Control Register (ECSPI5_PERIODREG)	32	R/W	0000_0000h	21.7.8/1000
201_8020	Test Control Register (ECSPI5_TESTREG)	32	R/W	0000_0000h	21.7.9/1002
201_8040	Message Data Register (ECSPI5_MSGDATA)	32	W	0000_0000h	21.7.10/ 1003

21.7.1 Receive Data Register (ECSPIx_RXDATA)

The Receive Data register (ECSPI_RXDATA) is a read-only register that forms the top word of the 64 x 32 receive FIFO. This register holds the data received from an external SPI device during a data transaction. Only word-sized read operations are allowed.

Address: Base address + 0h offset



ECSPIx_RXDATA field descriptions

Field	Description
ECSPI_RXDATA	Receive Data. This register holds the top word of the receive data FIFO. The FIFO is advanced for each read of this register. The data read is undefined when the Receive Data Ready (RR) bit in the Interrupt Control/Status register is cleared. Zeros are read when ECSPI is disabled.

21.7.2 Transmit Data Register (ECSPIx_TXDATA)

The Transmit Data (ECSPI_TXDATA) register is a write-only data register that forms the bottom word of the 64 x 32 TXFIFO. The TXFIFO can be written to as long as it is not full, even when the SPI Exchange bit (XCH) in ECSPI_CONREG is set. This allows software to write to the TXFIFO during a SPI data exchange process. Writes to this register are ignored when the ECSPI is disabled (ECSPI_CONREG[EN] bit is cleared).

Address: Base address + 4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R																																
W	ECSPI_TXDATA																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_TXDATA field descriptions

Field	Description
ECSPI_TXDATA	Transmit Data. This register holds the top word of data loaded into the FIFO. Data written to this register must be a word operation. The number of bits actually transmitted is determined by the BIT_COUNT field of the corresponding SPI Control register. If this field contains more bits than the number specified by BIT_COUNT, the extra bits are ignored. For example, to transfer 10 bits of data, a 32-bit word must be written to this register. Bits 9-0 are shifted out and bits 31-10 are ignored. When the ECSPI is operating in Slave mode, zeros are shifted out when the FIFO is empty. Zeros are read when ECSPI is disabled.

21.7.3 Control Register (ECSPIx_CONREG)

The Control Register (ECSPI_CONREG) allows software to enable the ECSPI , configure its operating modes, specify the divider value, and SPI_RDY control signal, and define the transfer length.

Address: Base address + 8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	BURST_LENGTH												CHANNEL_ SELECT		DRCTL	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PRE_DIVIDER				POST_DIVIDER				CHANNEL_MODE				SMC	XCH	HT	EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_CONREG field descriptions

Field	Description
31–20 BURST_ LENGTH	Burst Length. This field defines the length of a SPI burst to be transferred. The Chip Select (SS) will remain asserted until all bits in a SPI burst are shifted out. A maximum of 2^12 bits can be transferred in a single SPI burst. In master mode, it controls the number of bits per SPI burst. Since the shift register always loads 32-bit data from transmit FIFO, only the n least-significant (n = BURST LENGTH + 1) will be shifted out. The remaining bits will be ignored. In slave mode, only when SS_CTL is cleared, this field will take effect in the transfer. Number of Valid Bits in a SPI burst. 0x000 A SPI burst contains the 1 LSB in a word. 0x001 A SPI burst contains the 2 LSB in a word. 0x002 A SPI burst contains the 3 LSB in a word. ... 0x01F A SPI burst contains all 32 bits in a word. 0x020 A SPI burst contains the 1 LSB in first word and all 32 bits in second word. 0x021 A SPI burst contains the 2 LSB in first word and all 32 bits in second word. ... 0xFFE A SPI burst contains the 31 LSB in first word and 2^7 -1 words. 0xFFF A SPI burst contains 2^7 words.
19–18 CHANNEL_ SELECT	SPI CHANNEL SELECT bits. Select one of four external SPI Master/Slave Devices. In master mode, these two bits select the external slave devices by asserting the Chip Select (SSn) outputs. Only the selected Chip Select (SSn) signal can be active at a given time; the remaining three signals will be negated. 00 Channel 0 is selected. Chip Select 0 (SS0) will be asserted. 01 Channel 1 is selected. Chip Select 1 (SS1) will be asserted. 10 Channel 2 is selected. Chip Select 2 (SS2) will be asserted. 11 Channel 3 is selected. Chip Select 3 (SS3) will be asserted.
17–16 DRCTL	SPI Data Ready Control. This field selects the utilization of the $\overline{\text{SPI\_RDY}}$ signal in master mode. ECSPI checks this field before it starts an SPI burst. 00 The $\overline{\text{SPI\_RDY}}$ signal is a don't care. 01 Burst will be triggered by the falling edge of the SPI_RDY signal (edge-triggered). 10 Burst will be triggered by a low level of the SPI_RDY signal (level-triggered). 11 Reserved.
15–12 PRE_DIVIDER	SPI Pre Divider. ECSPI uses a two-stage divider to generate the SPI clock. This field defines the pre-divider of the reference clock. 0000 Divide by 1. 0001 Divide by 2. 0010 Divide by 3. ... 1101 Divide by 14. 1110 Divide by 15. 1111 Divide by 16.
11–8 POST_DIVIDER	SPI Post Divider. ECSPI uses a two-stage divider to generate the SPI clock. This field defines the post-divider of the reference clock using the equation: 2^{n} . 0000 Divide by 1.

Table continues on the next page...

ECSPIx_CONREG field descriptions (continued)

Field	Description
	0001 Divide by 2. 0010 Divide by 4. ... 1110 Divide by 2^{14}. 1111 Divide by 2^{15}.
7–4 CHANNEL_ MODE	SPI CHANNEL MODE selects the mode for each SPI channel. CHANNEL MODE[3] is for SPI channel 3. CHANNEL MODE[2] is for SPI channel 2. CHANNEL MODE[1] is for SPI channel 1. CHANNEL MODE[0] is for SPI channel 0. 0 Slave mode. 1 Master mode.
3 SMC	Start Mode Control. This bit applies only to channels configured in Master mode (CHANNEL MODE = 1). It controls how the ECSPI starts a SPI burst, either through the SPI exchange bit, or immediately when the TXFIFO is written to. 0 SPI Exchange Bit (XCH) controls when a SPI burst can start. Setting the XCH bit will start a SPI burst or multiple bursts. This is controlled by the SPI SS Wave Form Select (SS_CTL). Refer to XCH and SS_CTL descriptions. 1 Immediately starts a SPI burst when data is written in TXFIFO.
2 XCH	SPI Exchange Bit. This bit applies only to channels configured in Master mode (CHANNEL MODE = 1). If the Start Mode Control (SMC) bit is cleared, writing a 1 to this bit starts one SPI burst or multiple SPI bursts according to the SPI SS Wave Form Select (SS_CTL). The XCH bit remains set while either the data exchange is in progress, or when the ECSPI is waiting for an active input if SPIRDY is enabled through DRCTL. This bit is cleared automatically when all data in the TXFIFO and the shift register has been shifted out. 0 Idle. 1 Initiates exchange (write) or busy (read).
1 HT	Hardware Trigger Enable. This bit is used in master mode only. It enables hardware trigger (HT) mode. Note, HT mode is not supported by this product. 0 Disable HT mode. 1 Enable HT mode.
0 EN	SPI Block Enable Control. This bit enables the ECSPI. This bit must be set before writing to other registers or initiating an exchange. Writing zero to this bit disables the block and resets the internal logic with the exception of the ECSPI_CONREG. The block's internal clocks are gated off whenever the block is disabled. 0 Disable the block. 1 Enable the block.

21.7.4 Config Register (ECSPIx_CONFIGREG)

The Config Register (ECSPI_CONFIGREG) allows software to configure each SPI channel, configure its operating modes, specify the phase and polarity of the clock, configure the Chip Select (SS), and define the HT transfer length. Note, HT mode is not supported by this product.

Address: Base address + Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved			HT_LENGTH					SCLK_CTL				DATA_CTL				SS_POL				SS_CTL				SCLK_POL				SCLK_PHA			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_CONFIGREG field descriptions

Field	Description
31–29 -	This field is reserved. Reserved
28–24 HT_LENGTH	HT LENGTH. This field defines the message length in HT Mode. Note, HT mode is not supported by this product. The length in bits of one message is (HT LENGTH + 1).
23–20 SCLK_CTL	SCLK CTL. This field controls the inactive state of SCLK for each SPI channel. SCLK CTL[3] is for SPI channel 3. SCLK CTL[2] is for SPI channel 2. SCLK CTL[1] is for SPI channel 1. SCLK CTL[0] is for SPI channel 0. 0 Stay low. 1 Stay high.
19–16 DATA_CTL	DATA CTL. This field controls inactive state of the data line for each SPI channel. DATA CTL[3] is for SPI channel 3. DATA CTL[2] is for SPI channel 2. DATA CTL[1] is for SPI channel 1. DATA CTL[0] is for SPI channel 0. 0 Stay high. 1 Stay low.
15–12 SS_POL	SPI SS Polarity Select. In both Master and Slave modes, this field selects the polarity of the Chip Select (SS) signal. SS POL[3] is for SPI channel 3. SS POL[2] is for SPI channel 2. SS POL[1] is for SPI channel 1. SS POL[0] is for SPI channel 0.

Table continues on the next page...

ECSPIx_CONFIGREG field descriptions (continued)

Field	Description
	0 Active low. 1 Active high.
11–8 SS_CTL	SPI SS Wave Form Select. In master mode, this field controls the output wave form of the Chip Select (SS) signal when the SMC (Start Mode Control) bit is cleared. The SS_CTL are ignored if the SMC bit is set. SS CTL[3] is for SPI channel 3. SS CTL[2] is for SPI channel 2. SS CTL[1] is for SPI channel 1. SS CTL[0] is for SPI channel 0. In slave mode, this bit controls when the SPI burst is completed. An SPI burst is completed by the Chip Select (SS) signal edges. (SSPOL = 0: rising edge; SSPOL = 1: falling edge) The RXFIFO is advanced whenever a Chip Select (SS) signal edge is detected or the shift register contains 32-bits of valid data. 0 In master mode - only one SPI burst will be transmitted. 1 In master mode - Negate Chip Select (SS) signal between SPI bursts. Multiple SPI bursts will be transmitted. The SPI transfer will automatically stop when the TXFIFO is empty. 0 In slave mode - an SPI burst is completed when the number of bits received in the shift register is equal to (BURST LENGTH + 1). Only the n least-significant bits (n = BURST LENGTH[4:0] + 1) of the first received word are valid. All bits subsequent to the first received word in RXFIFO are valid. 1 In slave mode - an SPI burst is completed by the Chip Select (SS) signal edges. (SSPOL = 0: rising edge; SSPOL = 1: falling edge) The RXFIFO is advanced whenever a Chip Select (SS) signal edge is detected or the shift register contains 32-bits of valid data.
7–4 SCLK_POL	SPI Clock Polarity Control. This field controls the polarity of the SCLK signal. See Figure 21-10 for more information. SCLK_POL[3] is for SPI channel 3. SCLK_POL[2] is for SPI channel 2. SCLK_POL[1] is for SPI channel 1. SCLK_POL[0] is for SPI channel 0. 0 Active high polarity (0 = Idle). 1 Active low polarity (1 = Idle).
SCLK_PHA	SPI Clock/Data Phase Control. This field controls the clock/data phase relationship. See Figure 21-10 for more information. SCLK PHA[3] is for SPI channel 3. SCLK PHA[2] is for SPI channel 2. SCLK PHA[1] is for SPI channel 1. SCLK PHA[0] is for SPI channel 0. 0 Phase 0 operation. 1 Phase 1 operation.

21.7.5 Interrupt Control Register (ECSPIx_INTREG)

The Interrupt Control Register (ECSPI_INTREG) enables the generation of interrupts to the host processor. If the ECSPI is disabled, this register reads zero.

Address: Base address + 10h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								TCEN	ROEN	RFEN	RDREN	RREN	TFEN	TDREN	TEEN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_INTREG field descriptions

Field	Description
31–8 -	This field is reserved. Reserved
7 TCEN	Transfer Completed Interrupt enable. This bit enables the Transfer Completed Interrupt. 0 Disable 1 Enable
6 ROEN	RXFIFO Overflow Interrupt enable. This bit enables the RXFIFO Overflow Interrupt. 0 Disable 1 Enable
5 RFEN	RXFIFO Full Interrupt enable. This bit enables the RXFIFO Full Interrupt. 0 Disable 1 Enable
4 RDREN	RXFIFO Data Request Interrupt enable. This bit enables the RXFIFO Data Request Interrupt when the number of data entries in the RXFIFO is greater than RX_THRESHOLD. 0 Disable 1 Enable
3 RREN	RXFIFO Ready Interrupt enable. This bit enables the RXFIFO Ready Interrupt. 0 Disable 1 Enable

Table continues on the next page...

ECSPIx_INTREG field descriptions (continued)

Field	Description
2 TFEN	TXFIFO Full Interrupt enable. This bit enables the TXFIFO Full Interrupt. 0 Disable 1 Enable
1 TDREN	TXFIFO Data Request Interrupt enable. This bit enables the TXFIFO Data Request Interrupt when the number of data entries in the TXFIFO is less than or equal to TX_THRESHOLD. 0 Disable 1 Enable
0 TEEN	TXFIFO Empty Interrupt enable. This bit enables the TXFIFO Empty Interrupt. 0 Disable 1 Enable

21.7.6 DMA Control Register (ECSPIx_DMAREG)

The Direct Memory Access Control Register (ECSPI_DMAREG) provides software a way to use an on-chip DMA controller for ECSPI data. Internal DMA request signals enable direct data transfers between the ECSPI FIFOs and system memory. The ECSPI sends out DMA requests when the appropriate FIFO conditions are matched.

If the ECSPI is disabled, this register is read as 0.

Address: Base address + 14h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	RXTDEN	Reserved	RX_DMA_LENGTH						RXDEN	Reserved	RX_THRESHOLD					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved								TEDEN	Reserved	TX_THRESHOLD					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_DMAREG field descriptions

Field	Description
31 RXTDEN	RXFIFO TAIL DMA Request Enable. This bit enables an internal counter that is increased at each read of the RXFIFO. This counter is cleared automatically when it reaches RX DMA LENGTH. If the number of words remaining in the RXFIFO is greater than or equal to RX DMA LENGTH, a DMA request is generated even if it is less than or equal to RX_THRESHOLD. 0 Disable 1 Enable
30 -	This field is reserved. Reserved
29–24 RX_DMA_ LENGTH	RX DMA LENGTH. This field defines the burst length of a DMA operation. Applies only when RXTDEN is set.
23 RXDEN	RXFIFO DMA Request Enable. This bit enables/disables the RXFIFO DMA Request. 0 Disable 1 Enable
22 -	This field is reserved. Reserved
21–16 RX_ THRESHOLD	RX THRESHOLD. This field defines the FIFO threshold that triggers a RX DMA/INT request. A RX DMA/INT request is issued when the number of data entries in the RXFIFO is greater than RX_THRESHOLD.
15–8 -	This field is reserved. Reserved
7 TEDEN	TXFIFO Empty DMA Request Enable. This bit enables/disables the TXFIFO Empty DMA Request. 0 Disable 1 Enable
6 -	This field is reserved. Reserved
TX_ THRESHOLD	TX THRESHOLD. This field defines the FIFO threshold that triggers a TX DMA/INT request. A TX DMA/INT request is issued when the number of data entries in the TXFIFO is greater than TX_THRESHOLD.

21.7.7 Status Register (ECSPIx_STATREG)

The ECSPI Status Register (ECSPI_STATREG) reflects the status of the ECSPI's operating condition. If the ECSPI is disabled, this register reads 0x0000_0003.

Address: Base address + 18h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved								TC	RO	RF	RDR	RR	TF	TDR	TE
W									w1c	w1c						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

ECSPIx_STATREG field descriptions

Field	Description
31–8 -	This field is reserved. Reserved
7 TC	Transfer Completed Status bit. Writing 1 to this bit clears it. 0 Transfer in progress. 1 Transfer completed.
6 RO	RXFIFO Overflow. When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it. 0 RXFIFO has no overflow. 1 RXFIFO has overflowed.
5 RF	RXFIFO Full. This bit is set when the RXFIFO is full. 0 Not Full. 1 Full.
4 RDR	RXFIFO Data Request. 0 When RXTDE is set - Number of data entries in the RXFIFO is not greater than RX_THRESHOLD. 1 When RXTDE is set - Number of data entries in the RXFIFO is greater than RX_THRESHOLD or a DMA TAIL DMA condition exists. 0 When RXTDE is clear - Number of data entries in the RXFIFO is not greater than RX_THRESHOLD. 1 When RXTDE is clear - Number of data entries in the RXFIFO is greater than RX_THRESHOLD.
3 RR	RXFIFO Ready. This bit is set when one or more words are stored in the RXFIFO. 0 No valid data in RXFIFO. 1 More than 1 word in RXFIFO.
2 TF	TXFIFO Full. This bit is set when if the TXFIFO is full.

Table continues on the next page...

ECSPIx_STATREG field descriptions (continued)

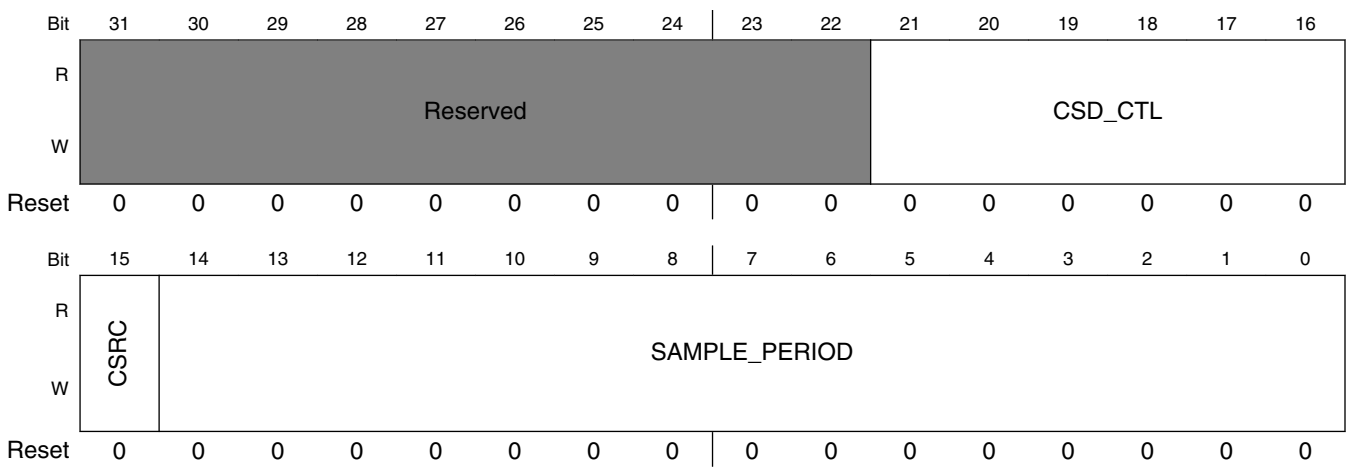
Field	Description
	0 TXFIFO is not Full. 1 TXFIFO is Full.
1 TDR	TXFIFO Data Request. 0 Number of empty slots in TXFIFO is greater than TX_THRESHOLD. 1 Number of empty slots in TXFIFO is not greater than TX_THRESHOLD.
0 TE	TXFIFO Empty. This bit is set if the TXFIFO is empty. 0 TXFIFO contains one or more words. 1 TXFIFO is empty.

21.7.8 Sample Period Control Register (ECSPIx_PERIODREG)

The Sample Period Control Register (ECSPI_PERIODREG) provides software a way to insert delays (wait states) between consecutive SPI transfers. Control bits in this register select the clock source for the sample period counter and the delay count indicating the number of wait states to be inserted between data transfers.

The delay counts apply only when the current channel is operating in Master mode (ECSPI_CONREG[CHANNEL MODE] = 1).ECSPI_PERIODREG also contains the CSD CTRL field used to insert a delay between the Chip Select's active edge and the first SPI Clock edge.

Address: Base address + 1Ch offset



ECSPIx_PERIODREG field descriptions

Field	Description
31–22 -	This field is reserved. Reserved

Table continues on the next page...

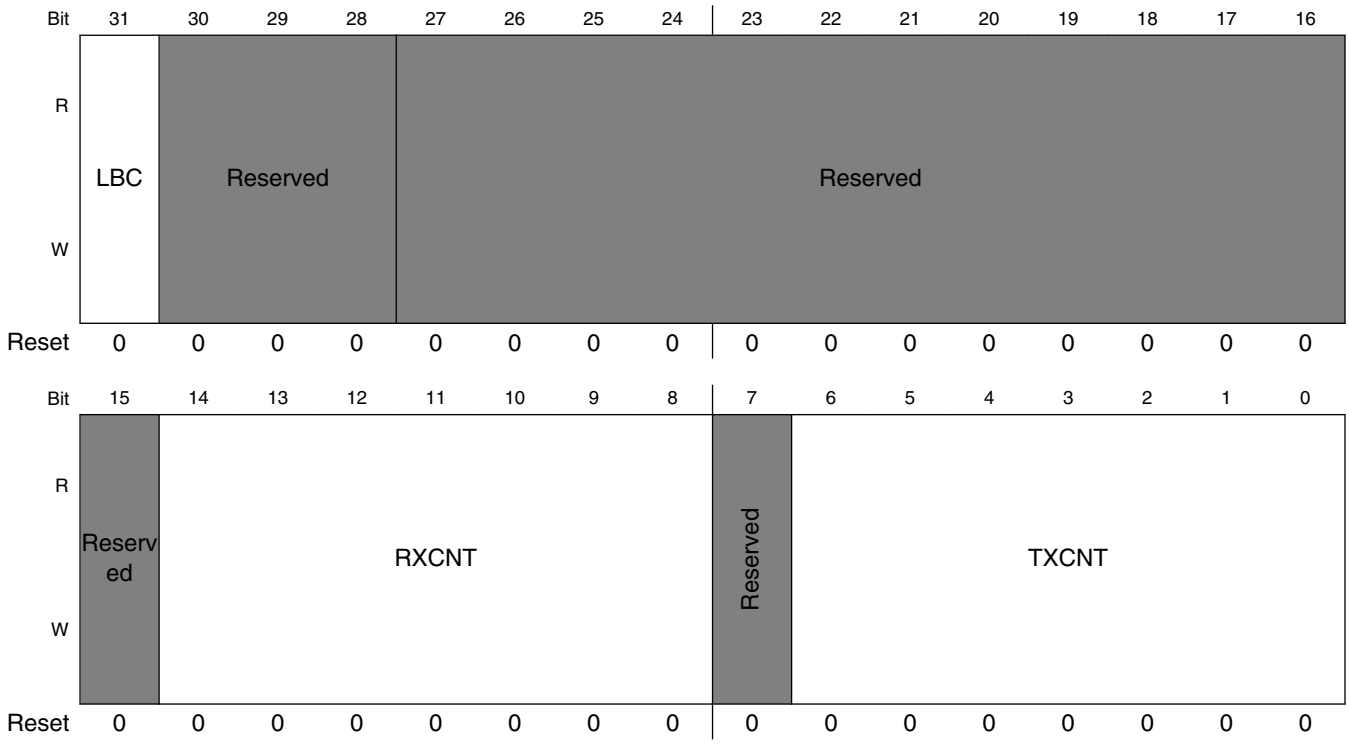
ECSPIx_PERIODREG field descriptions (continued)

Field	Description
21–16 CSD_CTL	Chip Select Delay Control bits. This field defines how many SPI clocks will be inserted between the chip select's active edge and the first SPI clock edge. The range is from 0 to 63.
15 CSRC	Clock Source Control. This bit selects the clock source for the sample period counter. 0 SPI Clock (SCLK) 1 Low-Frequency Reference Clock (32.768 KHz)
SAMPLE_ PERIOD	Sample Period Control. These bits control the number of wait states to be inserted in data transfers. During the idle clocks, the state of the SS output will operate according to the SS_CTL control field in the ECSPI_CONREG register. 0x0000 0 wait states inserted 0x0001 1 wait state inserted ... ... 0x7FFE 32766 wait states inserted 0x7FFF 32767 wait states inserted

21.7.9 Test Control Register (ECSPIx_TESTREG)

The Test Control Register (ECSPI_TESTREG) provides software a mechanism to internally connect the receive and transmit devices of the ECSPI, and monitor the contents of the receive and transmit FIFOs.

Address: Base address + 20h offset



ECSPIx_TESTREG field descriptions

Field	Description
31 LBC	Loop Back Control. This bit is used in Master mode only. When this bit is set, the ECSPI connects the transmitter and receiver sections internally, and the data shifted out from the most-significant bit of the shift register is looped back into the least-significant bit of the Shift register. In this way, a self-test of the complete transmit/receive path can be made. The output pins are not affected, and the input pins are ignored. 0 Not connected. 1 Transmitter and receiver sections internally connected for Loopback.
30–28 -	This field is reserved. Reserved, all bits should be ignored.
27–15 -	This field is reserved. Reserved
14–8 RXCNT	RXFIFO Counter. This field indicates the number of words in the RXFIFO.

Table continues on the next page...

ECSPIx_TESTREG field descriptions (continued)

Field	Description
7 -	This field is reserved. Reserved
TXCNT	TXFIFO Counter. This field indicates the number of words in the TXFIFO.

21.7.10 Message Data Register (ECSPIx_MSGDATA)

The Message Data Register (ECSPI_MSGDATA) forms the top word of the 16 x 32 MSG Data FIFO. Only word-size accesses are allowed for this register. Reads to this register return zero, and writes to this register store data in the MSG Data FIFO.

Address: Base address + 40h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R																																
W	ECSPI_MSGDATA																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ECSPIx_MSGDATA field descriptions

Field	Description
ECSPI_ MSGDATA	ECSPI_MSGDATA holds the top word of MSG Data FIFO. The MSG Data FIFO is advanced for each write of this register. The data read is zero. The data written to this register is stored in the MSG Data FIFO.

Chapter 22
External Interface Module (EIM)

22.1 Overview

The EIM handles the interface to devices external to the chip, including generation of chip selects, clock and control for external peripherals and memory. It provides asynchronous access to devices with SRAM-like interface and synchronous access to devices with NOR-Flash-like or PSRAM-like interface.

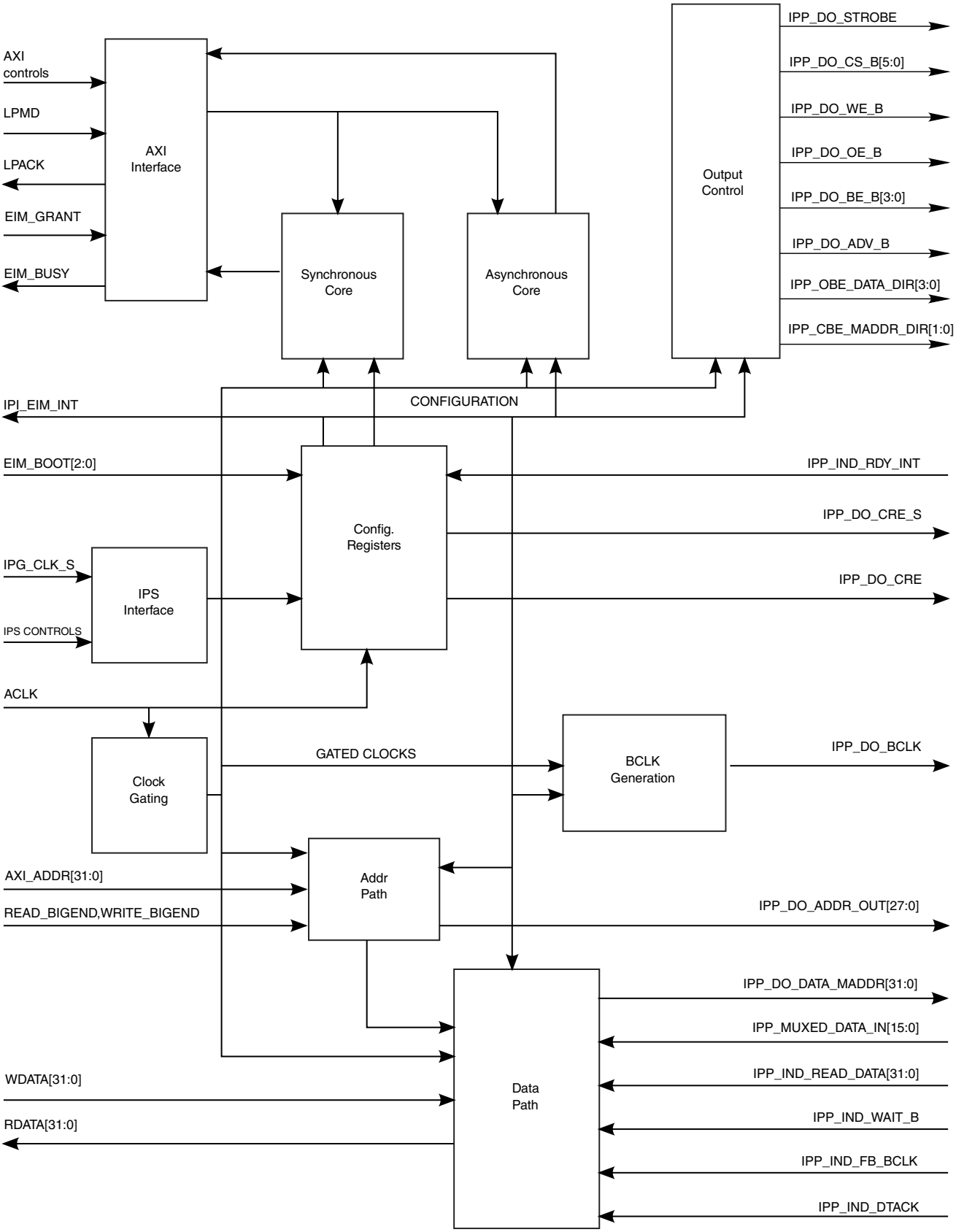


Figure 22-1. EIM Diagram

22.1.1 Features

- Four chip selects for external devices
  - Flexible address decoding. Each chip select memory space determined separately, according to VIA port configuration (see Chip Select Memory Map). Configurable Chip Select 0 base address (by VIA)
  - Individual select signal for each one of the memory space defined. Up to 6 memory spaces may be defined and programmed individually.
  - 26-bit external address bus, max memory size can be 128 MByte (1 Gigabit).
- Selectable Write Protection for each Chip Select
- Support for multiplexed address / data bus operation x16 and x32 port size
- Programmable Data Port Size for each Chip Select (x8, x16 and x32)
- Programmable Wait-State generator for each Chip Select, for write and read accesses separately
- Asynchronous accesses with programmable setup and hold times for control signals
- Support for Asynchronous page mode accesses (x16 and x32 port size)
- Independent synchronous Memory Burst Read Mode support for NOR-Flash and PSRAM memories (x16 and x32 port size)
- Independent synchronous Memory Burst Write Mode support for PSRAM and NOR-Flash like memories (CellularRAM™ from Micron, Infineon, and Cypress, OneNAND™ and utRAM™ from Samsung, and COSMORAM™ from Toshiba)
- Support of NAND-Flash devices with NOR-Flash like interface - MDOC™ (M-Systems), OneNAND™ (Samsung)
- Independent programmable variable/fix Latency support for read and write synchronous (burst) mode
- Support for Big Endian and Little Endian operation modes per access
- ARM AXI slave interface. One ID at a time support.
- External Interrupt support, RDY_INT signal function as external interrupt
- Boot from external device support according to boot signals, using RDY_INT signal
  - RDY signal support assertion after reset for MDOC™ (M-Systems) device
  - INT signal support assertion after reset for OneNAND™ (Samsung) device

22.1.2 Modes of Operation

The EIM has the following modes of operation:

- Asynchronous Mode
- Asynchronous Page Mode
- Multiplexed Address/Data mode
- Burst Clock Mode

- Low Power Modes
- Boot Mode

See details in the EIM Operational Modes.

22.1.2.1 Asynchronous Mode

This is a non-burst mode that is used for SRAM access. In this mode, a single data is read/written with each access (asserted address).

All controls' timings are controlled by preset values in Chip Select Configuration Registers.

22.1.2.2 Asynchronous Page Read Mode

Setting the APR bit causes the EIM to perform memory burst accesses by emulating page mode operation.

The external address asserts for each piece of data. The initial access timing is according to RWSC field, and the next address assertions timing is according to PAT field. When APR bit is set, RCSN OEN, RADVN and RBEN fields are ignored for burst access to the external device.

The page size can be set via the BL field to 2, 4, 8, 16, or 32 words (the word size is determined by the DSZ field).

22.1.2.3 Multiplexed Address/Data Mode

In this mode, multiplexing addresses and data bits on the same pins is supported for synchronous/asynchronous accesses to x8/x16/ x32 data width memory devices.

For more information about the pins that drive data/address in 8/16/32 non-muxed mode and 16/32 muxed mode, refer to the EIM Internal Module Multiplexing table in the EIM Internal Pads Allocation chapter of the datasheet.

Table 22-1. EIM multiplexing

Setup	Non Multiplexed Address/Data Mode							Multiplexed Address/ Data mode	
	8 Bit				16 Bit		32 Bit	16 Bit	32 Bit
	MUM = 0, DSZ = 100	MUM = 0, DSZ = 101	MUM = 0, DSZ = 110	MUM = 0, DSZ = 111	MUM = 0, DSZ = 001	MUM = 0, DSZ = 010	MUM = 0, DSZ = 011	MUM = 1, DSZ = 001	MUM = 1, DSZ = 011
A[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]	EIM_DA[15:0]
A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_A[25:16]	EIM_D[9:0]
D[7:0], EIM_EB0	EIM_D[7:0]	-	-	-	EIM_D[7:0]	-	EIM_D[7:0]	EIM_DA[7:0]	EIM_DA[7:0]
D[15:8], EIM_EB1	-	EIM_D[15:8]	-	-	EIM_D[15:8]	-	EIM_D[15:8]	EIM_DA[15:8]	EIM_DA[15:8]
D[23:16], EIM_EB2	-	-	EIM_D[23:16]	-	-	EIM_D[23:16]	EIM_D[23:16]	-	EIM_D[7:0]
D[31:24], EIM_EB3	-	-	-	EIM_D[31:24]	-	EIM_D[31:24]	EIM_D[31:24]	-	EIM_D[15:8]

22.1.2.4 Burst Clock Mode

The controller has the ability to support burst synchronous operations in various frequencies, depending on the frequency of the input clock supplied by the system (EIM clock).

The EIM clock can be divided by one, two, three or four, and its frequency can be changed according to the requirements. Variable and fix latency are supported for this mode, according to the external device requirements.

- Synchronous read mode. This is a burst mode, which is used for reading from Flash/ PSRAM memory devices. In this mode, after address assertion a burst of sequential data can be read. Data exchange is carried out according to BCLK being generated by EIM. An access is delayed according to external WAIT_B signal assertion (signal from the memory device).
- Synchronous write mode. A burst mode used for accessing external devices, which support synchronous write type of access (PSRAM protocol). In this mode, after address assertion a burst of sequential data can be written to the external device. Access may be delayed according to WAIT_B signal assertion (signal from the memory device) before first piece of data arrived to the external device.

NOTE

Maximum frequency of the EIM main clock is 133Mhz. It may be reduced by the system for special cases of external devices,

which demand a different frequency then integer division of the 133MHz clock.

22.1.2.5 Low Power Modes

The input clock is gated by ACT_CS bits. When all the ACT_CS are negated (all CS disable) the internal clock is turned off; awready/wready & arready signal are de-asserted and the master can't access the EIM.

22.1.2.6 Boot Mode

It is possible to perform a boot operation from external device located on CS0. The configuration of the relevant bits are done with boot mode signals according to the external device parameters (for example, port size and protocol assertion).

See System Boot for more details.

22.2 External Signals

The following table describes the external signals of EIM:

Table 22-2. EIM External Signals

Signal	Description	Pad	Mode	Direction
EIM_AD00	LSB multiplexed Address/Data Bus signal	EIM_DA0	ALT0	IO
EIM_AD01	LSB multiplexed Address/Data Bus signal	EIM_DA1	ALT0	IO
EIM_AD02	LSB multiplexed Address/Data Bus signal	EIM_DA2	ALT0	IO
EIM_AD03	LSB multiplexed Address/Data Bus signal	EIM_DA3	ALT0	IO
EIM_AD04	LSB multiplexed Address/Data Bus signal	EIM_DA4	ALT0	IO
EIM_AD05	LSB multiplexed Address/Data Bus signal	EIM_DA5	ALT0	IO
EIM_AD06	LSB multiplexed Address/Data Bus signal	EIM_DA6	ALT0	IO
EIM_AD07	LSB multiplexed Address/Data Bus signal	EIM_DA7	ALT0	IO
EIM_AD08	LSB multiplexed Address/Data Bus signal	EIM_DA8	ALT0	IO

Table continues on the next page...

Table 22-2. EIM External Signals (continued)

Signal	Description	Pad	Mode	Direction
EIM_AD09	LSB multiplexed Address/Data Bus signal	EIM_DA9	ALT0	IO
EIM_AD10	LSB multiplexed Address/Data Bus signal	EIM_DA10	ALT0	IO
EIM_AD11	LSB multiplexed Address/Data Bus signal	EIM_DA11	ALT0	IO
EIM_AD12	LSB multiplexed Address/Data Bus signal	EIM_DA12	ALT0	IO
EIM_AD13	LSB multiplexed Address/Data Bus signal	EIM_DA13	ALT0	IO
EIM_AD14	LSB multiplexed Address/Data Bus signal	EIM_DA14	ALT0	IO
EIM_AD15	LSB multiplexed Address/Data Bus signal	EIM_DA15	ALT0	IO
EIM_ADDR16	MSB Address Bus signal	EIM_A16	ALT0	O
EIM_ADDR17	MSB Address Bus signal	EIM_A17	ALT0	O
EIM_ADDR18	MSB Address Bus signal	EIM_A18	ALT0	O
EIM_ADDR19	MSB Address Bus signal	EIM_A19	ALT0	O
EIM_ADDR20	MSB Address Bus signal	EIM_A20	ALT0	O
EIM_ADDR21	MSB Address Bus signal	EIM_A21	ALT0	O
EIM_ADDR22	MSB Address Bus signal	EIM_A22	ALT0	O
EIM_ADDR23	MSB Address Bus signal	EIM_A23	ALT0	O
EIM_ADDR24	MSB Address Bus signal	EIM_A24	ALT0	O
EIM_ADDR25	MSB Address Bus signal	EIM_A25	ALT0	O
EIM_ADDR26	MSB Address Bus signal	NANDF_CS3	ALT3	O
EIM_BCLK	Burst Clock (BCLK). This active-high output signal is used to clock external burstcapable devices to synchronize the loading and incrementing of addresses and delivery of burst read and write data to/from the EIM. Its behavior is affected by the BCM field in the EIM_WCR and the SWR, SRD, BCD, and BCS fields of the EIM_CSxGCR1.	EIM_BCLK	ALT0	O
EIM_CRE	Used as CRE/PS for CellularRam memory. It is used for the Mode Register Set command. This signal can be configured as active low or active high. See CRE and CREP field descriptions of the EIM_CSxGCR1 registers.	NANDF_CS2	ALT3	O
EIM_CS0	Chip Selects. These signals are active-low. Behavior is affected by the RCSA and RCSN fields of the	EIM_CS0	ALT0	O

Table continues on the next page...

Table 22-2. EIM External Signals (continued)

Signal	Description	Pad	Mode	Direction
	EIM_CSxRCR1 registers and the WCSA and WCSN fields of the EIM_CSxWCR1registers.			
EIM_CS1	Chip Selects. These signals are active-low. Behavior is affected by the RCSA and RCSN fields of the EIM_CSxRCR1 registers and the WCSA and WCSN fields of the EIM_CSxWCR1registers.	EIM_CS1	ALT0	O
EIM_CS2	Chip Selects. These signals are active-low. Behavior is affected by the RCSA and RCSN fields of the EIM_CSxRCR1 registers and the WCSA and WCSN fields of the EIM_CSxWCR1registers.	DISP0_DAT18	ALT7	O
		SD2_DAT1	ALT2	
EIM_CS3	Chip Selects. These signals are active-low. Behavior is affected by the RCSA and RCSN fields of the EIM_CSxRCR1 registers and the WCSA and WCSN fields of the EIM_CSxWCR1registers.	DISP0_DAT19	ALT7	O
		SD2_DAT2	ALT2	
EIM_DATA00	MSB Data Bus signal	CSI0_DATA_EN	ALT1	IO
EIM_DATA01	MSB Data Bus signal	CSI0_VSYNC	ALT1	IO
EIM_DATA02	MSB Data Bus signal	CSI0_DAT4	ALT1	IO
EIM_DATA03	MSB Data Bus signal	CSI0_DAT5	ALT1	IO
EIM_DATA04	MSB Data Bus signal	CSI0_DAT6	ALT1	IO
EIM_DATA05	MSB Data Bus signal	CSI0_DAT7	ALT1	IO
EIM_DATA06	MSB Data Bus signal	CSI0_DAT8	ALT1	IO
EIM_DATA07	MSB Data Bus signal	CSI0_DAT9	ALT1	IO
EIM_DATA08	MSB Data Bus signal	CSI0_DAT12	ALT1	IO
EIM_DATA09	MSB Data Bus signal	CSI0_DAT13	ALT1	IO
EIM_DATA10	MSB Data Bus signal	CSI0_DAT14	ALT1	IO
EIM_DATA11	MSB Data Bus signal	CSI0_DAT15	ALT1	IO
EIM_DATA12	MSB Data Bus signal	CSI0_DAT16	ALT1	IO
EIM_DATA13	MSB Data Bus signal	CSI0_DAT17	ALT1	IO
EIM_DATA14	MSB Data Bus signal	CSI0_DAT18	ALT1	IO
EIM_DATA15	MSB Data Bus signal	CSI0_DAT19	ALT1	IO
EIM_DATA16	MSB Data Bus signal	EIM_D16	ALT0	IO
EIM_DATA17	MSB Data Bus signal	EIM_D17	ALT0	IO
EIM_DATA18	MSB Data Bus signal	EIM_D18	ALT0	IO
EIM_DATA19	MSB Data Bus signal	EIM_D19	ALT0	IO
EIM_DATA20	MSB Data Bus signal	EIM_D20	ALT0	IO
EIM_DATA21	MSB Data Bus signal	EIM_D21	ALT0	IO
EIM_DATA22	MSB Data Bus signal	EIM_D22	ALT0	IO

Table continues on the next page...

Table 22-2. EIM External Signals (continued)

Signal	Description	Pad	Mode	Direction
EIM_DATA23	MSB Data Bus signal	EIM_D23	ALT0	IO
EIM_DATA24	MSB Data Bus signal	EIM_D24	ALT0	IO
EIM_DATA25	MSB Data Bus signal	EIM_D25	ALT0	IO
EIM_DATA26	MSB Data Bus signal	EIM_D26	ALT0	IO
EIM_DATA27	MSB Data Bus signal	EIM_D27	ALT0	IO
EIM_DATA28	MSB Data Bus signal	EIM_D28	ALT0	IO
EIM_DATA29	MSB Data Bus signal	EIM_D29	ALT0	IO
EIM_DATA30	MSB Data Bus signal	EIM_D30	ALT0	IO
EIM_DATA31	MSB Data Bus signal	EIM_D31	ALT0	IO
EIM_DTACK_B	Data Acknowledge, asynchronous access. This input is used as a data acknowledge signal for single asynchronous accesses.	EIM_WAIT	ALT1	I
EIM_EB0	Byte Enable. These active-low output signals indicate valid data bytes for the current access. They may be configured to assert for write cycles only. EIM_EB[0] corresponds to DATA_OUT[7:0] For asynchronous write accesses, behavior is affected by the WBEA and WBEN fields of the EIM_CS1WCR1-EIM_CS5WCR1 Registers. On synchronous or asynchronous read accesses, these signals are always asserted at the start of the access and negated at end of the access.	EIM_EB0	ALT0	O
EIM_EB1	Byte Enable. These active-low output signals indicate valid data bytes for the current access. They may be configured to assert for write cycles only. EIM_EB[1] corresponds to DATA_OUT[15:8] For asynchronous write accesses, behavior is affected by the WBEA and WBEN fields of the EIM_CS1WCR1-EIM_CS5WCR1 Registers. On synchronous or asynchronous read accesses, these signals are always asserted at the start of the access and negated at end of the access.	EIM_EB1	ALT0	O

Table continues on the next page...

Table 22-2. EIM External Signals (continued)

Signal	Description	Pad	Mode	Direction
EIM_EB2	Byte Enable. These active-low output signals indicate valid data bytes for the current access. They may be configured to assert for write cycles only. EIM_EB[2] corresponds to DATA_OUT[23:16] For asynchronous write accesses, behavior is affected by the WBEA and WBEN fields of the EIM_CS1WCR1-EIM_CS5WCR1 Registers. On synchronous or asynchronous read accesses, these signals are always asserted at the start of the access and negated at end of the access.	EIM_EB2	ALT0	O
EIM_EB3	Byte Enable. These active-low output signals indicate valid data bytes for the current access. They may be configured to assert for write cycles only. EIM_EB[3] corresponds to DATA_OUT[31:24]. For asynchronous write accesses, behavior is affected by the WBEA and WBEN fields of the EIM_CS1WCR1-EIM_CS5WCR1 Registers. On synchronous or asynchronous read accesses, these signals are always asserted at the start of the access and negated at end of the access.	EIM_EB3	ALT0	O
EIM_LBA	Address Valid. This active-low output signal is asserted during burst mode accesses to cause the external burst capable device to load a new starting burst address. Assertion of LBA indicates that a valid address is present on the address bus. Its behavior is affected by the SWR, SRD, BCD, and BCS fields of the EIM_CSxGCR1 registers, the RADVA and RADVN fields of the EIM_CSxRCR1 registers, and the WADVA and WADVN fields of the EIM_CSxWCR1 registers. In asynchronous mode, LBA length is affected by the RADVA, WADVA, RADVN, and WADVN fields. Minimum length of LBA signal in all modes is one EIM clock cycle.	EIM_LBA	ALT0	O

Table continues on the next page...

Table 22-2. EIM External Signals (continued)

Signal	Description	Pad	Mode	Direction
EIM_OE	Output Enable. This active-low output signal indicates the bus access is a read and enables external devices to drive the data bus with read data. Its behavior is affected by the OEA and OEN bit fields in the Chip Select Configuration Registers.	EIM_OE	ALT0	O
EIM_RW	Memory Write Enable. This active-low output signal indicates the bus access is a write and enables external devices to sample the data bus. Its behavior is affected by the WEA and WEN bit fields in the Chip Select Configuration Registers.	EIM_RW	ALT0	O
EIM_WAIT	Ready/Busy/Wait signal. This active-low input signal is asserted by external burst capable devices which support fixed or variable latency of data. It is serviced in synchronous mode only (EIM_CSxGCR1[SWR, SRD] =1). WAIT will have a pull up resistor in pad. The signal indicates whether the External device is ready for data transaction or not. Busy cycles (or wait cycles) of the external device can occur at the start of a Burst access or at page boundary crossover. **NOTE:** For burst devices, WAIT output should be configured to change one cycle before data is ready (before delay). **NOTE:** Some External devices may not use this input signal for ready state indication (fix latency without WAIT signal monitoring). For these devices EIM should be configured accordingly (see RFL, WFL, and PSZ field descriptions). **NOTE:** This is same as what is shown in IP_IND_WAIT_B	EIM_WAIT	ALT0	I

22.2.1 Other Important Block I/O Signals Internal to the SoC

The following table provides a description of other signals which are internal to the that are important to understand the function of EIM.

Name	I/O	Description
EIM_FB_BCLK	Input	Burst Clock Feedback. This block input is used to sample read data during high transfer speeds. The signal provides feedback from the I/O pad of the BCLK output pin and tends to align more closely with data from the external memory device.
EIM_BOOT	Input	EIM Boot Configuration. These block inputs determine the reset state of DSZ[1:0] and MUM. A more detailed description can be found in Fusemap.
ACLK	Input	AXI clock, maximum frequency 133 Mhz
IPG_CLK_S	Input	EIM module IPG clock
RST_B	Input	Active low HW reset
EIM_WARM_RESET	Input	Warm Reset. If this signal is asserted the rst_b will reset only the internal FF and state machine while S/W registers will keep their current state.This signal is active high signal.

22.3 Clocks

The following table describes the clock sources for EIM. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 22-4. EIM Clocks

Clock name	Clock Root	Description
aclk	aclk_eim_slow_clk_root	EIM clock (main)
aclk_slow	aclk_eim_slow_clk_root	EIM clock (slow)
ipg_clk_s	ipg_clk_root	Peripheral access clock
aclk_exsc	aclk_eim_slow_clk_root	EIM clock (external device)

- ACLK: EIM clock (main clock, AXI clock) with a Max frequency of 133Mhz. Can be gated externally when there is no active AXI access.
- ACLK_SLOW: EIM all time running ACLK. Used for flip-flops that must be active even when EIM is in low power down mode to provide clock for lpack/lpmd registers, IP registers and IP to AXI sync registers.
- IPG_CLK_S: IPG clock for IP accesses. IP registers are activated by ACLK_SLOW clock.
- ACLK_EXSC: Clock created from EIM clock for External device usage. Integer division by 1, 2, 3 and 4 of the clock can be use with BCD bit field configuration, according to external devices demands. EIM clock frequency may be reduced for lower frequency support which cannot be achieved via BCD bit field.

22.4 Chip Select Memory Map

Table 22-5. EIM Chip Select Memory Map

Address	Space Size	Use	Access
EIM_NFC_BASE/ACT_CS/ADDRS0 inputs	128MB	CS0 memory region	R/W
ACT_CS/ADDRS1 inputs	128MB	CS1 memory region	R/W
ACT_CS/ADDRS2 inputs	128MB	CS2 memory region	R/W
ACT_CS/ADDRS3 inputs	128MB	CS3 memory region	R/W

22.5 Functional Description

This section provides the functional description for the EIM.

22.5.1 Bus Sizing Configuration

The EIM supports byte, half word and word operands allowing access to x8, x16, x32 ports. It can be address/data multiplexed in x16, x32 ports. The port size is programmable via the DSZ bit field in the corresponding Chip Select Configuration Register. An 8-bit port can reside in each one of the bytes of the data bus. A 16-bit port can reside on the lower 16 bits of the data bus, DATA_IN/OUT[15:0] or on the higher 16 bits of the data bus, DATA_IN/OUT[31:16].

In the case of a multi-cycle transfer, the lower two address bits (ADDR[1:0]) are incremented appropriately. The EIM address bus is configured according to DSZ bit field and AUS bits. There is either one bit (for x16 port size) or two bits (for x32 port size) right shift of the address bits (only when AUS=0) and no bit shift when AUS = 1 or DSZ[2] = 1.

The EIM has a data multiplexer which takes the four bytes of the AXI data bus and routes them to their required positions to properly interface to memory.

NOTE

A word access to or from a x16 port requires two external bus cycles to complete the transfer.

A word access to or from a x8 port requires four external bus cycles to complete the transfer.

22.5.1.1 8 BIT PORT SUPPORT

EIM has limited support for mot68000 & intel 386 protocols.

22.5.1.1.1 MOTOROLA 68000

EIM has limited support for mot68000 protocol. Only basic read or write asynchronous operations are supported.

The following operations are not supported:

- Read modify write
- Sync access
- All special accesses (ARM platform space, bus arbitration, bus control, bus error & reset operations)
- FC outputs

22.5.1.1.2 INTEL 386

EIM has limited support for intel 386 protocol. Only basic read or write async non-pipelined operations are supported.

The following operations are not supported:

- Other bus cycles (interrupt, halt & refresh)
- Bus lock
- M/IO, DC, LBA, NA, REFRESH & BS8 signals

22.5.2 EIM Operational Modes

Listed here are the main operational modes for EIM selected by control bit fields settings.

For details, see the bit field descriptions of SWR / SRD / MUM. All modes are supported in with 8-, 16- or 32-bit port configuration, according to DSZ bit field.

Table 22-6. EIM Operation Modes Field Settings

Control bit fields			Brief mode description
MUM	SRD	SWR	
0	0	0	Asynchronous write / Asynchronous read for APR=0 / Asynchronous page read for APR=1, none multiplexed
		1	Synchronous write/ Asynchronous read or APR=0 /

Table continues on the next page...

Table 22-6. EIM Operation Modes Field Settings (continued)

Control bit fields			Brief mode description
MUM	SRD	SWR	
			Asynchronous page read for APR=1,none multiplexed
	1	0	Asynchronous write/Synchronous read none multiplexed
		1	Synchronous write/read none multiplexed
1	0	0	Asynchronous write/read multiplexed
		1	Synchronous write/ Asynchronous read multiplexed
	1	0	Asynchronous write/Synchronous read multiplexed
		1	Synchronous write/read multiplexed

22.5.3 Burst Mode (Synchronous) Memory Operation

This mode is enabled for read or write access. Bit SWR sets the burst mode for write operations at the corresponding chip select and bit SRD sets it for read operation.

When this mode is set, the controller attempts to translate the Master burst accesses to memory burst accesses, being limited by the memory burst length, predefined by BL value, or memory and Master WRAP/INCR boundary crossing non-matching. Only the first address accessed is put by the controller on the external address bus in a memory burst sequence.

EIM may translate from some Master sequential accesses to one or several memory bursts, but not from two Master individual accesses to one memory burst.

For the first access in a memory burst sequence, the EIM asserts $\overline{\text{ADV}}$, causing the external burst device to latch the starting burst address; then toggle the burst clock (BCLK) for a predefined number of cycles in order to latch the first unit of data. Subsequent accessed data units can then be burst in fewer clock cycles, realizing an overall increase in bus bandwidth.

NOTE

The BCLK signal toggles only when burst access is executed toward the external device (BCM=1'b0 for normal mode use). It runs with a 50% duty cycle until the end of access is reached. When access is terminated, BCLK stops toggling.

Memory burst accesses are terminated by the EIM whenever it detects the following:

- The specific burst length has executed completely (end of access)
- Write access - missing data in write buffer (Master is delaying the data transfer toward the EIM)

- Next sequential access crosses boundary with unequal condition (wrap/increment, burst length) on the Master and memory
- Current memory burst length reached

22.5.4 Burst Clock Divisor (BCD)

In some cases, it may be necessary to slow the external bus in relation to the internal bus to allow accesses to burst devices that have a maximum operating frequency less than the operating frequency of the internal bus.

The internal bus frequency can be divided by one, two, three or four for presentation on the external bus in burst mode operation.

BCLK can only be set to integer divisions of the incoming clock frequency. To get a specific frequency on BCLK, configure the divider to change the incoming EIM clock accordingly.

By programming the BCD bit field to various values, two signals on the external bus are affected; $\overline{\text{ADV}}$ and BCLK. The $\overline{\text{ADV}}$ signal is asserted according to RADVA or WADVA bit fields programming, and is negated according to the formula mentioned in RADVN and WADVN bit fields description. The BCLK signal runs with a 50% duty cycle until the end of access is reached.

If BCM = 1, the BCLK runs at frequency according to GBCD bit field settings on every async memory access, regardless of the SWR and SRD bits configuration. Caution should be exercised when using BCM bit; GBCD bit field should be updated once and should not change when BCLK is toggling. The BCM bit is used mainly for system debug mode. It has no functional use of the EIM in normal mode.

22.5.5 Burst Clock Start (BCS)

In an effort to allow greater flexibility in achieving the minimum number of wait states on burst accesses, you can determine when you want the BCLK to start toggling after the start of access. This allows the BCLK to be skewed from point of data capture on the EIM clock by any number of EIM clock cycles.

Care must be exercised when setting BCS bit field in conjunction with the BCD and RWSC/WWSC bit fields. See the external timing diagrams in Burst (Synchronous Mode) Read Memory Accesses Timing Diagram - BCD=1 and Burst (Synchronous Mode) Read Memory Accesses Timing Diagram - BCD=0 for examples of how to use the BCS, BCD and RWSC/WWSC bit fields together.

22.5.6 Multiplexed Address/Data Mode Support

The control bit MUM allows support memory with multiplexed address/data bus both in asynchronous and in synchronous modes.

Caution should be exercised for using OEA/WEA & ADH bit fields. They should be configured according to the external device requirements, as it determines the time point of end of address phase and start of data phase.

22.5.7 Mixed Master/Memory Burst Modes Support

To provide mixed sequential/wrap accesses with different length, EIM interprets burst signal and generate additional $\overline{\text{ADV}}$ signals whenever there appear unequal address or burst boundary crossing condition.

BL bit field is used to notify EIM about current memory burst and wrap condition for properly external address generation. In case of non-matching boundaries in both the memory and Master access, EIM starts a new memory burst access by updating address from Master on address bus and generating $\overline{\text{ADV}}$ signal.

22.5.8 AXI (Master) Bus Cycles Support

The EIM uses an ARM AXI slave interface. It has a 32-bit bus and supports one access (one ID) at a time. No out of order or parallel accesses are supported.

The following AXI protocol signals are not supported:

- AWLOCK
- AWCACHE
- ARLOCK
- ARCACHE

ARID bus is sampled when:

- new read access is valid on the read address channel and is reflected on the RID bus output toward the master.

AWID bus is sampled when:

- new write access is valid on the write address channel and is reflected on the WID/BID bus output toward the master.

ARPROT and AWPROT signal are partially used. ARPROT[0] and AWPROT[0] bits are used for normal/privileged access detection. ARPROT[2:1] and AWPROT[2:1] are not used.

When sampling a valid access on both of the address channels, the read access will be performed first while write access is pending. After last data transfer completed, the pending write will be executed.

A new access may be executed one cycle after sampling a valid access on the read or write address channels, assuming there is no current access (back to back) which can cause a recovery or end of access penalty cycles, for write access, also assuming data is in write buffer for fast execution.

NOTE

- Only 32-bit word size accesses are supported for burst mode accesses.
- Only 8-bit (1 byte), 16-bit (2 byte) and 32-bit (4 byte) word size supported for single access.
- Maximum number of burst length is 16.
- According to AXI protocol, burst access should not cross 4 KB blocks. In case EIM gets an access that crosses the 4 KB, memory address calculation is invalid.

AXI transfers shown in the table below are also supported. These AXI cycles will be translated into the necessary cycles on the memory side. For example, for optimal operation in case ARM cache is configured to 8 beat burst with wrap, a synchronous flash and cellular RAM memory should be configured in 16 word wrap burst mode when using a 16-bit data port, and in 8 word wrap burst mode when using a 32-bit data port. EIM uses BL bit field to support different memory configurations. The controller splits the transaction when needed in some cases. See Table 22-8.

Table 22-7. AXI Burst Cycles Supported

Burst Length - Number of data transfers	Burst size - Bytes in transfer	Burst type	Description
1	1	INCR	Single transfer
1	2	INCR	Single transfer
1	4	INCR	Single transfer
2	4	WRAP	2-beat wrapping burst
4	4	WRAP	4-beat wrapping burst
8	4	WRAP	8-beat wrapping burst
16	4	WRAP	16-beat wrapping burst
2	4	INCR	2-beat incrementing burst

Table continues on the next page...

Table 22-7. AXI Burst Cycles Supported (continued)

Burst Length - Number of data transfers	Burst size - Bytes in transfer	Burst type	Description
3	4	INCR	3-beat incrementing burst
4	4	INCR	4-beat incrementing burst
5	4	INCR	5-beat incrementing burst
6	4	INCR	6-beat incrementing burst
7	4	INCR	7-beat incrementing burst
8	4	INCR	8-beat incrementing burst
9	4	INCR	9-beat incrementing burst
10	4	INCR	10-beat incrementing burst
11	4	INCR	11-beat incrementing burst
12	4	INCR	12-beat incrementing burst
13	4	INCR	13-beat incrementing burst
14	4	INCR	14-beat incrementing burst
15	4	INCR	15-beat incrementing burst
16	4	INCR	16-beat incrementing burst

Table 22-8. AXI to Memory Burst Splits Number

AXI Burst Type	Memory Burst Type Config.	# of accesses to X8 Memory Port size	# of accesses to X16 Memory Port size	# of accesses to X32 Memory Port size
INC16 Aligned Addr.	WRAP4	16	8	4
	Cont.	1	1	1
INC16 Unaligned Addr.	WRAP4	17	9	5
	Cont.	1	1	1
WRAP16 Aligned Addr.	WRAP16	4	2	1
	Cont.	1	1	1
WRAP16 Unaligned Addr.	WRAP16	5	3	1
	Cont.	2	2	2
INC8 Aligned Addr.	WRAP8	4	2	1
	WRAP16	2	1	1
INC8 Unaligned Addr.	WRAP8	4 or 5	2 or 3	2
	WRAP16	2 or 3	2	1 or 2
WRAP8 Aligned Addr.	WRAP16	2	1	1
	Cont.	1	1	1
WRAP8 Unaligned Addr.	WRAP16	2 or 3	1	2
	Cont.	2	2	2

22.5.9 WAIT_B Signal, RWSC and WWSC bit fields Usage

Most of the external devices supporting burst mode for write or read accesses provide a signal which indicates data is valid on the memory bus (a.k.a. handshake mode). For this mode, RFL and WFL bits should be cleared and RWSC/ WWSC bit fields indicate when the controller should start sampling this signal from the external device or, in other words, how many BCLK cycles should be masked.

For devices which do not use this signal or have a fixed latency ability, the RFL and WFL bits may be set for internal calculation regarding BCLK cycles penalty until data is valid (memory initial access time). For this mode, RWSC/ WWSC indicates when the data is ready for sampling by the controller (read access) or the external device (write access). There is separation between read and write accesses wait-state control. For read access, RWSC bit field is valid and WWSC bit field is ignored; for write access, WWSC is valid and RWSC is ignored.

22.5.10 IPS Register Interface

Access to the registers of the EIM, read or write, is made with IPS protocol signals. The system should avoid changing the registers while master/memory transaction is valid, as this can cause an unknown behavior of the controller.

Register access size is 32-bit as the register size definition, other size of access (byte or half word) is not supported.

22.5.11 MRS Set for PSRAM

Memory registers of PSRAM devices can be configured according to external signal, which indicates whether the access is to a memory array or memory register domain.

When the CRE bit is set, the following transactions to the external device will assert the CRE signal. The polarity of this signal is determined by the CREP bit for active low or active high assertion of the signal.

22.5.12 EIM Access Termination

EIM is monitoring the corresponding CSx control signal every time variable latency access or dtack access is performed toward the external device.

In variable latency accesses, the Watchdog Timer (WDOG-1) counts BCLK cycles. If it reaches the wdog_limit (according to the WDOG_LIMIT bit field in the WCR) before the device signals can drive/sample new data, the controller will terminate the access and generate an error response transfer toward the Master.

In dtack access, WDOG-1 counts ACLK cycles instead of BCLK and it reaches the wdog_limit before the device asserts the dtack signal, the controller will terminate the access and generate an error response transfer toward the Master.

WDOG-1 can by disabled by WDOG_EN bit in the WCR.

22.5.13 Error Conditions

The following conditions cause an error (AXI error or IPS error) response signal:

- AXI errors
  - Access to a disabled chip select - access to a mapped chip select address space where the CSEN bit in the corresponding chip select Configuration Register is clear
  - Access to a non mapped address - access to an address that is not mapped to any CS.
  - User access to a supervisor-protected chip select address space (the SP bit in the corresponding chip select Configuration Register is set)
  - User access in fixed mode access
  - User performs write access to write protected chip select
  - First write data ID and write address ID do not match. (No data is written to the memory.)
  - First Write Data ID and write address ID match but one or more of the other Write data IDs does not match the First Write data ID (data is written to memory according)
  - Access duration to external device from CSx signal assertion is 128/256/512/1024 cycles (access is terminated by the controller) - This error can be disabled be software.
- IPS errors
  - User read or write access to a reserved/non-valid address in the EIM Configuration Register

22.5.14 DTACK Mode

In DTACK mode, the EIM uses DTACK signal as an indication of when to end the access.

DTACK is an asynchronous edge/level sensitive signal. DTACK polarity is configurable by the DAP bit in CsxGCR2 (default value is 0).

In this case, EIM begins the access and after a few cycles (according DAPS field) and waits until DTACK (after synchronization) becomes asserted, then samples the data in read access and completes the current data access (see Figure 22-15, Figure 22-16 & Figure 22-17).

If more than one data is needed, CS will be negated between access (CSREC field is not zero) and the AXI burst access will be split into single accesses (see Figure 22-19).

22.5.15 RDY_INT Signal as Interrupt

The EIM has an external interrupt support. When INTEN bit in the WCR is set, signal RDY_INT is used as interrupt; its status is being reflected by INT bit and output signal.

It is cleared by writing one to the INT bit. When INTEN is cleared, the interrupt is disabled. This interrupt is a level interrupt and its polarity can be configured by the INTPOL bit in the WCR.

22.5.16 RDY_INT Signal as Ready After Reset Indication

This feature is used for boot propose from external devices based on NANDFlash array memory with NORFlash interface.

When ERRST bit is set, RDY_INT signal is monitored to determine ready after reset of the external device located on CS0.

The monitoring is taking place when CS0 is accessed for the first time. The access will be pending until assertion of the signal is detected. When detection occurs, ERRST bit is self-cleared and pending access is executed to the external device on CS0.

22.5.17 EIM_GRANT / EIM_BUSY Handshake Description

Prior to executing command to one of the external device (chip select), EIM assert EIM_BUSY signal (1'b1) and checks the EIM_GRANT signal status.

If EIM_GRANT signal is high, it indicates external data bus is not used by other slaves (NAND Flash Controller) and EIM may start to execute the access. If EIM_GRANT is low, EIM waits until it is set (1'b1) before executing the access.

EIM keeps EIM_BUSY signal set until it completes the access toward the external device.

Once EIM_GRANT signal is set, it can not be reset until EIM_BUSY signal is cleared by EIM.

NOTE

In 16-bit Muxed EIM doesn't use the data bus, therefore there is no sharing of the data bus with NFC. EIM doesn't wait for EIM_GRANT signal from NFC and doesn't assert the EIM_BUSY signal.

22.5.18 LPMD / LPACK Handshake Description

These signals are used for frequency and/or voltage change, and for entering low power mode during normal operation of the EIM. Before any change can take place, the controller and all the relevant external devices should be in idle state, which means no access or data transfer is in process.

LPMD input signal is asserted once EIM detects the assertion of LPMD, all ready signals of the AXI channels are negated, and EIM is not sampling new accesses. It finishes all the ongoing accesses and already pending ones. When EIM is in idle state, the LPACK output signal is asserted. EIM will stay in idle state and the LPACK signal will stay asserted until the LPMD signal is negated.

22.5.19 Endianness

Big and Little endianness are supported by the controller according to the following table.

Table 22-9. EIM Out/in Data in Case AXI Out/in Data is 0xB3B2B1B0

Endian mode	AXI access	AXI address [1:0]	Port size and used bits								
			Word port				Half word port			Byte port	
			[31:24]	[23:16]	[15:8]	[7:0]	External address [0]	[31:24] ([15:8])	[23:16] ([7:0])	External address [1:0]	[31:24] ([23:16]) ([15:8]) ([7:0])
Big	Word	0	0xB3	0xB2	0xB1	0xB0	0	0xB3	0xB2	0	0xB3
										1	0xB2
							1	0xB1	0xB0	2	0xB1
										3	0xB0

Table continues on the next page...

Table 22-9. EIM Out/in Data in Case AXI Out/in Data is 0xB3B2B1B0 (continued)

Endian mode	AXI access	AXI address [1:0]	Port size and used bits								
			Word port				Half word port			Byte port	
			[31:24]	[23:16]	[15:8]	[7:0]	External address [0]	[31:24] ([15:8])	[23:16] ([7:0])	External address [1:0]	[31:24] ([23:16]) ([15:8]) ([7:0])
	Half Word	0			0xB1	0xB0	0	0xB3	0xB2	0	0xB3
										1	0xB2
		2	0xB3	0xB2			1	0xB1	0xB0	2	0xB1
										3	0xB0
	Byte	0				0xB0	0		0xB3	0	0xB3
		1			0xB1			0xB2		1	0xB2
		2		0xB2			1		0xB1	2	0xB1
		3	0xB3					0xB0		3	0xB0
Little	Word	0	0xB3	0xB2	0xB1	0xB0	0	0xB1	0xB0	0	0xB0
										1	0xB1
							1	0xB3	0xB2	2	0xB2
										3	0xB3
	Half Word	0			0xB1	0xB0	0	0xB1	0xB0	0	0xB0
										1	0xB1
		2	0xB3	0xB2			1	0xB3	0xB2	2	0xB2
										3	0xB3
	Byte	0				0xB0	0		0xB0	0	0xB0
		1			0xB1			0xB1		1	0xB1
		2		0xB2			1		0xB2	2	0xB2
		3	0xB3					0xB3		3	0xB3

22.5.20 Strobe Signal Use

The strobe signal is toggling according to address/data valid condition on the external bus for read and write accesses, and for both synchronous and asynchronous modes.

At any time point when address/data is valid on the external bus, the strobe signal will generate a positive edge, which can be used to sample the external data and control signal.

NOTE

Strobe signal for read data is active (RL + 1) cycles after data on external bus is valid.

22.6 Initialization Information

22.6.1 Booting from EIM

EIM is ready to work with CS0 after the hardware reset, but it has been configured for very slow access (for boot purposes), with additional setup and hold time.

Other CSs are disabled by hardware reset. Therefore, all CSs must be properly initialized before use in writing values to the corresponding chip select configuration registers.

DSZ[1:0] and MUM fields are set according to EIM_BOOT [2:0] block inputs.

22.7 Typical Application

Application note uses following functions to illustrate EIM and memory accesses:

- WR16(address, data) is a 16 bit write access
- WR32(address, data) is a 32 bit write access
- RD16(address, data) is a 16 bit read access
- RD32(address, data) is a 32 bit read access
- WR_I(address, data, delta, counter) is a write data sequence, there data(i+1) = data(i) + delta
- COMMAND_SEQUENCE
- CHECK_STATUS

NOTE

COMMAND_SEQUENCE and CHECK_STATUS are described in AMD Flash Utility, Intel Sibley Flash Utility, MDOC Device Utility, Samsung OneNAND Utility, and Spansion Flash Utility.

All addresses are byte addresses. "CS0" is a Chip Select 0 base address. "EIM_" is a prefix of EIM's registers. 'h is a prefix of hexadecimal constant. "//" is a comment beginning. csba[cs] is a dimension of CS base addresses. "addr" means an address offset in current CS address space. Examples use CS0 address space, but it may apply to any CS except for boot mode functionality.

Configuration examples were verfied with the memory models listed below and may require some adjustments for other family members.

22.7.1 Access to Intel Sibley Flash

The following configurations are intended to Sibley family muxed and non-muxed devices.

22.7.1.1 Intel Sibley Flash Asynchronous Mode Configuration

- WR32('EIM_CS0GCR1,'h00210081);
- WR32('EIM_CS0RCR1,'h0e020000);
- WR32('EIM_CS0RCR2,'h00000000);
- WR32('EIM_CS0WCR1,'h0704a040);

22.7.1.2 Intel Sibley Flash Synchronous Mode Configuration

Configuration used for 133 MHz synchronous access to flash:

```
// Set memory to synchronous read mode
     WR16('CS0+('h5903<<1),'h0060);
     WR16('CS0+('h5903<<1),'h0003);
     WR16('CS0+('h0000<<1),'h00ff);
// Set EIM configuration to synchronous timing
     WR32('EIM_CS0GCR1,'h50214225);                    // 133 MHz
     WR32('EIM_CS0RCR1,'h0c000000);                    // 12 cycles on memory
```

Configuration used for 66 MHz synchronous access to muxed flash:

```
// Set memory to synchronous read mode
     WR16('CS0+('h3103<<1),'h0060);
     WR16('CS0+('h3103<<1),'h0003);
     WR16('CS0+('h0000<<1),'h00ff);
//-------------------------------------------------------------------------
// Set EIM configuration to synchronous timing
     WR32('EIM_CS0GCR1,'h5021122d);                    // 66 MHz
     WR32('EIM_CS0RCR1,'h07000000);                    // 7cycles on memory
```

22.7.1.3 Intel Sibley Flash Utility

```
// Single data word programming to addr
     WR16('CS0+addr,'h0060);                 // Unlock
     WR16('CS0+addr,'h00d0);
     WR16('CS0+addr, 'h0041);
     WR16('CS0+addr, data);
     WR16('CS0+caddr, 'h0070);                 // Read Status command
     while('CS0+data[7] == 0)                              // Wait / Polling
         RD16('CS0+addr, data);              // Read status
     RD16('CS0+addr, data);                  // Read status
     WR16('CS0+'h0000,'h00ff);
// Write buffer programming
     WR16('CS0+addr,'h0060);                 // Unlock
     WR16('CS0+addr,'h00d0);
     data    = 0;
```

```
WR16('CS0+addr, 'h0070);                 // Read Status command
while(data[7] == 0)                            // Wait
    RD16('CS0+addr, data);               // Read status
WR16('CS0+'h0000,'h00ff);
WR16('CS0+addr,'h00e9);             // Write Buffer command
WR16('CS0+addr,255);                // Word counter (<256)
for(i=0; i<'h200; i = i + 'h40)
   WR_I('CS0+addr+i, data+((i>2)*'h0010_0001), 'h0010_0001, 16); // Data
WR16('CS0+addr,'h00d0);             // Write Confirm command
data  = 0;
while(data[7] == 0)                          // Wait
   RD16('CS0+addr, data);           // Read status
RD16('CS0+addr, data);              // Read status
WR16('CS0+'h0000,'h00ff);
```

22.7.2 Access to MDOC Device

The following configurations are intended to MDOC H3 device.

22.7.2.1 MDOC Device Boot

To boot from the MDOC device the ERRST bit should be configured to 1, so that EIM will hold the first read access to CS0 until the MDOC asserts the RDY signal.

22.7.2.2 MDOC Device Asynchronous Mode Configuration

```
// Non-muxed mode
WR32('EIM_CS0GCR1,'h00410081);
WR32('EIM_CS0RCR1,'h0e121010);
WR32('EIM_CS0RCR2,'h00000000);
WR32('EIM_CS0WCR1,'h12092492);
// Muxed mode
WR32('EIM_CS0GCR1,'h00410081);
WR32('EIM_CS0RCR1,'h0e121010);
WR32('EIM_CS0RCR2,'h00000000);
WR32('EIM_CS0WCR1,'h12092492);
```

22.7.2.3 MDOC Device Utility

```
// Read Manufacturer ID and Device ID
    RE16('CS0+'h9400,'h4833);
    RE16('CS0+'h9422,'hb7cc);
```

22.7.3 Access to Micron PSRAM

The following configurations are intended to mt45w4mw16bfb_706.

22.7.3.1 Micron PSRAM Asynchronous Mode Configuration

```
// 16 bit memory
    WR32('EIM_CS0GCR1,'h403104b1);
    WR32('EIM_CS0RCR1,'h0b010000);
    WR32('EIM_CS0RCR2,'h00000008);
    WR32('EIM_CS0WCR1,'h0b040040);
// 32 bit memory
    WR32('EIM_CS0GCR1,'h403304b1);
    WR32('EIM_CS0RCR1,'h0f010000);
    WR32('EIM_CS0RCR2,'h00000008);
    WR32('EIM_CS0WCR1,'h0f040040);
```

22.7.3.2 Micron PSRAM Synchronous Mode Configuration

```
// 16 bit memory
    WR32('EIM_CS0GCR1,'h403104b1);
    WR32('EIM_CS0WCR1,'h0b040000);
    WR16('CS0+('h85947<<1),'h0040); // memory configuration
    WR32('EIM_CS0GCR1,'h4021_5487);// fixed latency memory wrap 4
    WR32('EIM_CS0RCR1,'h04000000);
    WR32('EIM_CS0RCR2,'h00000008);
    WR32('EIM_CS0WCR1,'h04000000);
// 32 bit memory
    WR32('EIM_CS0GCR1,'h6003_04f1);
    WR32('EIM_CS0WCR1,'h0b04_0000);
    WR32('CS0+('h85947<<2),'h0040); // memory configuration
    WR32('EIM_CS0GCR1,'h4003_1487); // var latency memory inc. page size 128
    WR32('EIM_CS0RCR1,'h04000000);
    WR32('EIM_CS0RCR2,'h00000008);
    WR32('EIM_CS0WCR1,'h04000000);
```

22.7.4 Access to Samsung OneNAND

Mentioned below are the configurations intended for Samsung OneNAND muxed and non-muxed devices.

22.7.4.1 Samsung OneNAND Boot

There are two ways to boot from Samsung OneNAND. In the first way, the ERRST bit is set to 0 and the user has to poll the interrupt status in the OneNAND interrupt register (or set interrupt handler there). In the second way, the ERRST bit is set to 1 and the user should enable the device interrupt output before the first read from CS0 access is issued.

Load sectors 2,3 to DataRAM, page 0 done in the next example:

- WR16('CS0+('hF241<<1),'h0); // Clear interrupt status
- WR16('CS0+('hF100<<1),'h0); // block[8:0] address
- WR16('CS0+('hF107<<1),'h2); // sector[1:0] and page[7:2] addresses
- WR16('CS0+('hF200<<1),'h802); // buffer[11:8] address and counter[1:0]

- WR16('CS0+('hF101<<1),'h0); // DDP choose
- WR16('CS0+('hF220<<1),'h0); // Set command

22.7.4.2 Samsung OneNAND Asynchronous Mode Configuration

```
// Non-muxed memory
    WR32('EIM_CS0GCR1,'h00410081);
    WR32('EIM_CS0RCR1,'h0b010000);
    WR32('EIM_CS0RCR2,'h00000000);
    WR32('EIM_CS0WCR1,'h0c092480);
// Muxed memory
    WR32('EIM_CS0GCR1,'h00410089);
    WR32('EIM_CS0RCR1,'h0b010000);
    WR32('EIM_CS0RCR2,'h00000000);
    WR32('EIM_CS0WCR1,'h0c092480);
```

22.7.4.3 Samsung OneNAND Synchronous Mode Configuration

Set memory and EIM to synchronous read mode is shown in the next example:

```
    WR16('CS0+('hF221<<1),'hc0e0); // Synchronous read, 4 clk latency
    WR32('EIM_CS0GCR1,'h50412405); // 44 MHz (non-muxed)
    WR32('EIM_CS0RCR1,'h05010000);
```

The muxed Samsung OneNAND supports synchronous write, too:

```
// Set memory & EIM to synchronous read and write mode
    WR16('CS0+('hF221<<1),'hc0f2); // Sync. read and write, 4 clk latency
    WR32('EIM_CS0GCR1,'h5041240f); // 44 MHz
    WR32('EIM_CS0RCR1,'h05010000);
    WR32('EIM_CS0WCR1,'h05040000);
```

22.7.4.4 Samsung OneNAND Utility

The following utility algorithms are used on the Samsung OneNAND:

```
// Unlock Block command
    WR16('CS0+('hF100<<1),'h0);        // DFS
    WR16('CS0+('hF100<<1),'h0);        // DBS
    WR16('CS0+('hF24c<<1),'h2);        // SBA - block number (2)
    WR16('CS0+('hF241<<1),'h0);        // Clear interrupt status
    WR16('CS0+('hF220<<1),'h23);       // Unlock command
    data ='h0;
    while(!(data &'h0004))             // Polling
       RD32('WIAR, data); // Read status
// Erase block command
    WR16('CS0+('hF100<<1),'h2);        // DFS and block ([8:0]) address
    WR16('CS0+('hF101<<1),'h0);        // DBS
    WR16('CS0+('hF241<<1),'h0);        // Clear interrupt status
              WR16('CS0+('hF220<<1),'h94);            // Erase command
    data ='h0;
    while(!(data &'h0004))                  // Wait
        RD32('WIAR, data); // Read status
// Program page command
    WR16('CS0+('hF100<<1),'h2);             // DFS and block[8:0] address
    WR16('CS0+('hF107<<1),'h0);             // sector[1:0] and page[7:2] addresses
```

```
          WR16('CS0+('hF200<<1),'h800);           // buffer[11:8] address and counter[1:0]
          WR16('CS0+('hF241<<1),'h0);             // Clear interrupt status
          WR16('CS0+('hF220<<1),'h80);            // Program command
          data ='h0;
          while(!(data &'h0004))                  // Wait
              RD32('WIAR, data); // Read status
```

22.7.5 Access to Samsung UtRAM

Below mentioned configurations are intended for Samsung UtRAM.

22.7.5.1 Samsung UtRAM Asynchronous Mode Configuration

```
          WR32('EIM_CS0GCR1,'h400104b1);
          WR32('EIM_CS0RCR1,'h0a010000);
          WR32('EIM_CS0RCR2,'h00000008);
          WR32('EIM_CS0WCR1,'h0b040040);
```

22.7.5.2 Samsung UtRAM Synchronous Mode Configuration

```
          RD16('CS0+('hff_ffff<<1),data);             // command sequence
          RD16('CS0+('hff_ffff<<1),data);
          RD16('CS0+('hff_ffff<<1),data);
          RD16('CS0+('hff_feff<<1),data);
          RD16('CS0+('h00_82a0<<1),data);             // memory sync. configuration
          WR32('EIM_CS0GCR1,'h4021_53b7);            // fixed latency memory wrap 32
          WR32('EIM_CS0RCR1,'h0500_0000);
          WR32('EIM_CS0WCR1,'h0300_0000);
```

22.7.6 Access to Spansion Flash

Below mentioned configurations are intended for Spansion Flash.

22.7.6.1 Spansion Flash Asynchronous Mode Configuration

```
          WR32('EIM_CS0GCR1,'h00410081);
          WR32('EIM_CS0RCR1,'h0a018000);
          WR32('EIM_CS0RCR2,'h00000000);
          WR32('EIM_CS0WCR1,'h0704a240);
WR16('CS0+('hF220<<1),'h94);            // Erase command
          data ='h0;
          while(!(data &'h0004))                  // Wait
              RD32('WIAR, data); // Read status
     // Program page command
          WR16('CS0+('hF100<<1),'h2);             // DFS and block[8:0] address
          WR16('CS0+('hF107<<1),'h0);             // sector[1:0] and page[7:2] addresses
          WR16('CS0+('hF200<<1),'h800);           // buffer[11:8] address and counter[1:0]
          WR16('CS0+('hF241<<1),'h0);             // Clear interrupt status
          WR16('CS0+('hF220<<1),'h80);            // Program command
          data ='h0;
```

```
        while(!(data &'h0004))                  // Wait
            RD32('WIAR, data); // Read status
```

22.7.6.2 Spansion Flash Synchronous Mode Configuration

```
        WR16('CS0+('h0555<<1),'h00aa);   // command sequence
        WR16('CS0+('h02aa<<1),'h0055);
        WR16('CS0+('h0555<<1),'hd0);
        WR16('CS0+('h0000<<1),'h1ec4);   // memory sync. configuration
        WR32('EIM_CS0GCR1,'h50411325); // 66 MHz
        WR32('EIM_CS0RCR1,'h05000000); // 5 cycles on memory
```

22.7.6.3 Spansion Flash Utility

```
    // Single word programming
        COMMAND_SEQUENCE(cs,16,'ha0);           // single word programming
        WR16('CS0+addr, data);
        CHECK_STATUS('CS0+addr,data,16,1,errst);
    // Write buffer programming
        COMMAND_SEQUENCE(0,16,'h25);              // write buffer programming
        WR16('CS0+addr,'h001f);                     // counter-1
        WR_I('CS0+addr, data, 'h0010_0001, 16); // data
        WR16('CS0+addr,'h0029);                     // write buffer to flash
        CHECK_STATUS('CS0+addr+'h3e,data[31:16]+'h00f0,16,1,errst);
```

There COMMAND_SEQUENCE and CHECK_STATUS are next functions:

```
    task COMMAND_SEQUENCE;
        input[2:0]    cs;
        input[7:0]    port_size;
        input[31:0]   code;
    begin
        if(port_size == 16)
           begin
           WR16(csba[cs]+('h0555<<1),'h00aa);
           WR16(csba[cs]+('h02aa<<1),'h0055);
           WR16(csba[cs]+('h0555<<1),code);
           end
        else
           begin
           WR32(csba[cs]+('h0555<<2),'h00aa);
           WR32(csba[cs]+('h02aa<<2),'h0055);
           WR32(csba[cs]+('h0555<<2),code);
           end
    end
    endtask
    task     CHECK_STATUS;
        input[31:0] addr;
        input[31:0] edata;
        input[7:0]   port_size;
        input[7:0]   opcode;
        output[7:0] errst;
        reg[31:0]    data;
        reg[31:0]    data3;
    begin
        errst = 0;
        data  = 0;
        data3 = 0;
while(!(data == edata) && !errst) // Wait operation
      begin: BR_EN
      RD16(addr, data);             // Read status
      if(data[7] !== edata[7])
```

```
            begin
            if(data[5] == 1)
                 begin
                 RD16(addr, data3);
                 RD16(addr, data);
                 if(data[6] !== data3[6])
                      begin
                      $display("CHECK_STATUS: Error timout on single data program");
                      errst = 1;
                      disable BR_EN;
                      end
                 end
            else
                 begin
                 if(opcode == 2)
                      if(data[1] == 1)
                          begin
                          RD16(addr, data3);
                          if(port_size == 32)
                             RD32(addr, data);
                          else
                             RD16(addr, data);
                          if(data[1] == 1 && data !== edata)
                             begin
                             $display("CHECK_STATUS: Error on write buffer");
                             errst =3;
                             disable BR_EN;
                             end
                          end
                 end
            end
        else
            begin
            RD16(addr, data3);
            if(port_size == 32)
                 RD32(addr, data);
            else
                 begin
                 RD16(addr, data);
                 edata[31:16] = 16'h0;
                 end
            if(data !== edata)
                 begin
                 $display("CHECK_STATUS: Error in data write on single data program");
                 errst =2;
                 disable BR_EN;
                 end
            end
        end
end
endtask
```

22.7.7 8 bit support

This section details the pin connections for Intel mode and Motorola mode.

Intel Mode - For intel mode use the following connection:

Table 22-10. Intel Mode pin connections

ARM platform Pin	EIM Pin	Notes
ADS#	IPP_DO_ADV_B	WAL = 1,RAL = 1

Table continues on the next page...

Table 22-10. Intel Mode pin connections (continued)

ARM platform Pin	EIM Pin	Notes
W/R	IPP_DO_BE_B	WBED = 1
WR#	WE#	
RD#	OE#	

Mot. Mode - For intel mode use the following connection:

Table 22-11. Motorola Mode pin connections

ARM platform Pin	EIM Pin	Notes
AS#	IPP_DO_CS_B	
R/W#	WE#	
LDS#	BE#	

22.8 External Bus Timing Diagrams

The following timing diagrams show the timing of accesses to memory or a peripheral with different timing parameters. All examples done for CS0, but are valid for any others chip select. BE means one from current used BE[3:0].

22.8.1 Asynchronous Read Memory Accesses Timing Diagram

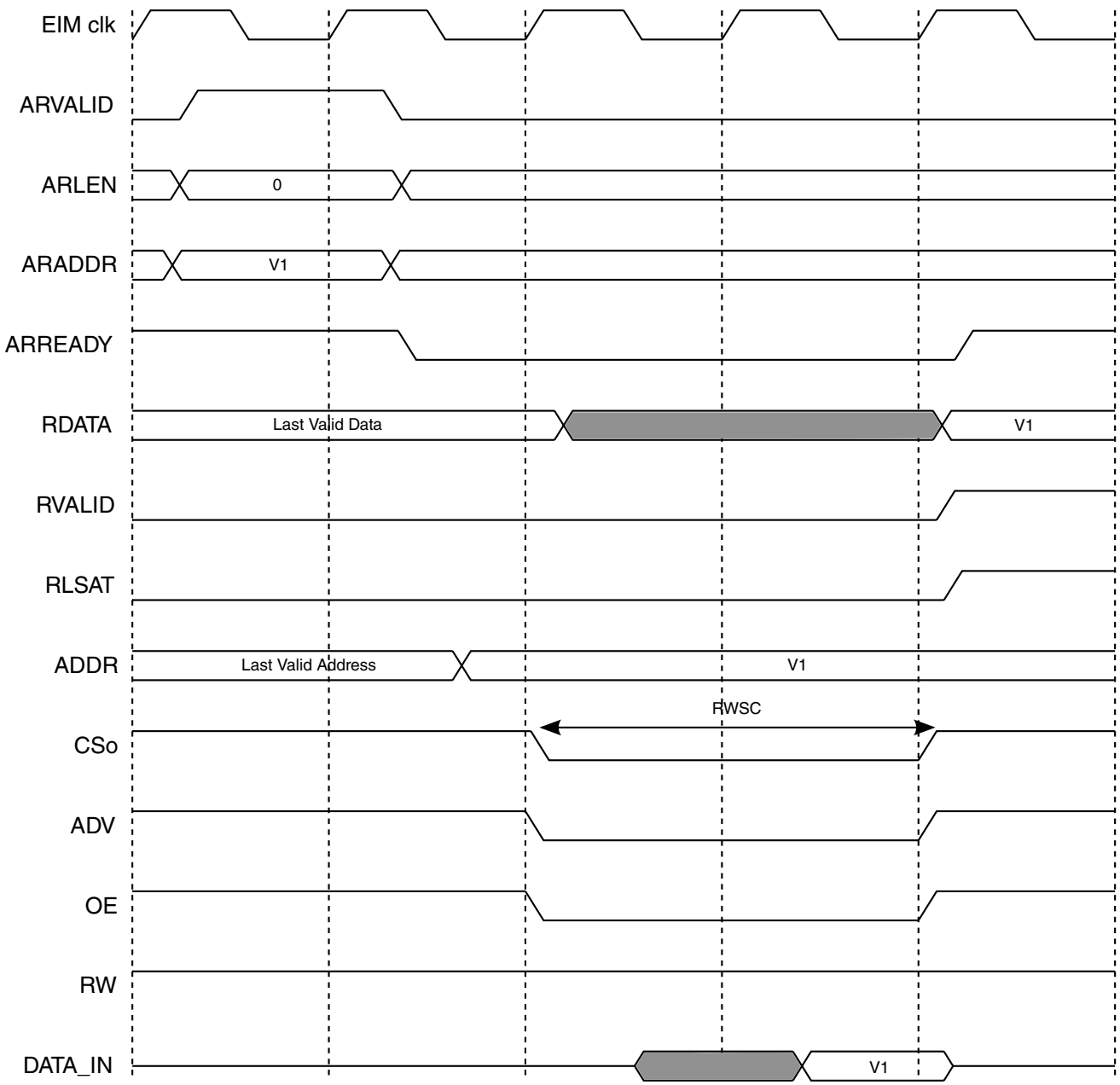


Figure 22-2. Read Access, RWSC=2,RCSA=0,OEA=0,RCSN=0,OEN=0, RAL=1

22.8.2 Asynchronous Write Memory Accesses Timing Diagram

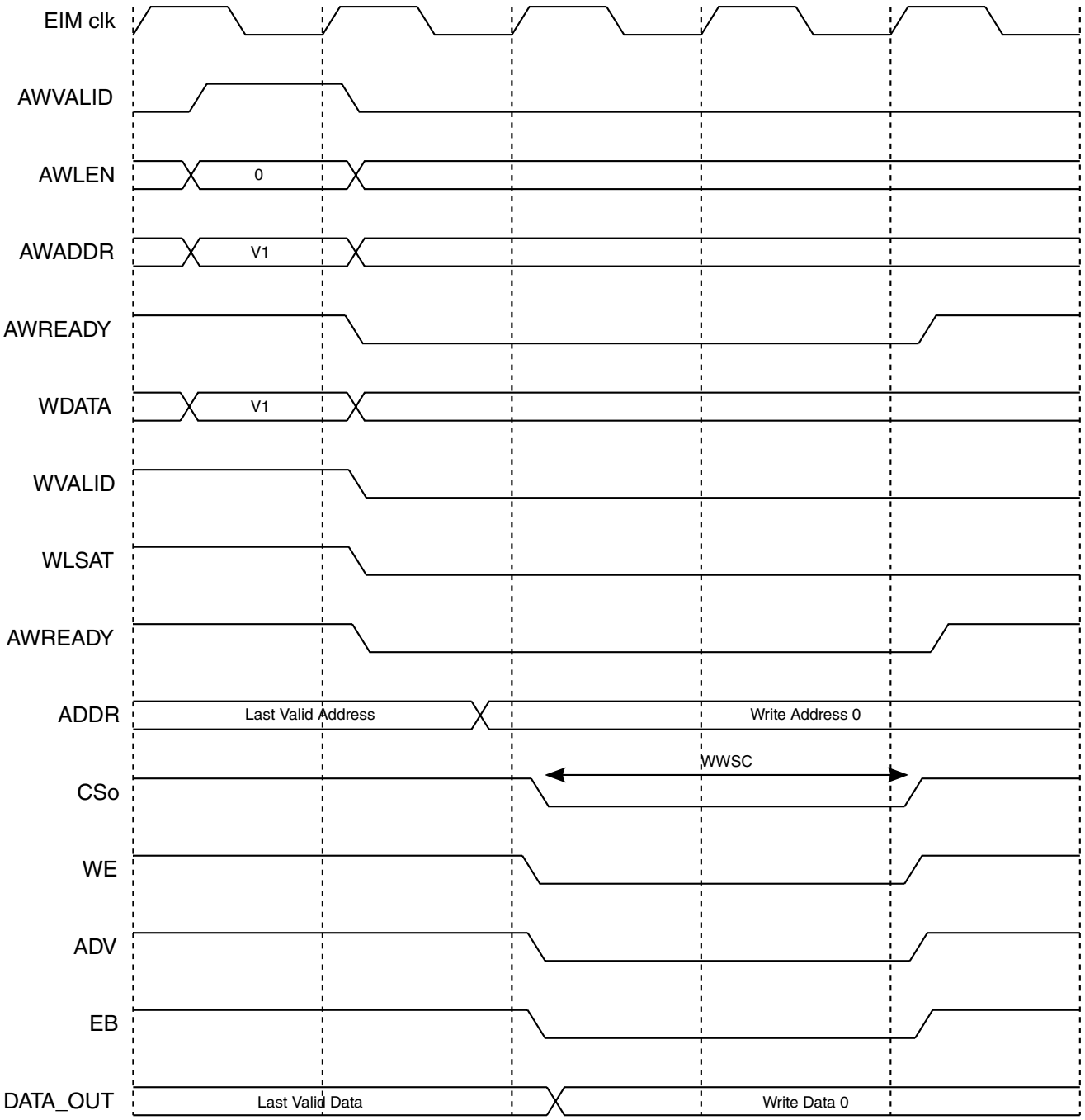


Figure 22-3. Write Access, WWSC=2,WCSA=0,WEA=0,WCSN=0,WEN=0,BEA=0, BEN=0, WAL=1

22.8.3 Asynchronous Read/Write Memory Accesses Timing Diagram

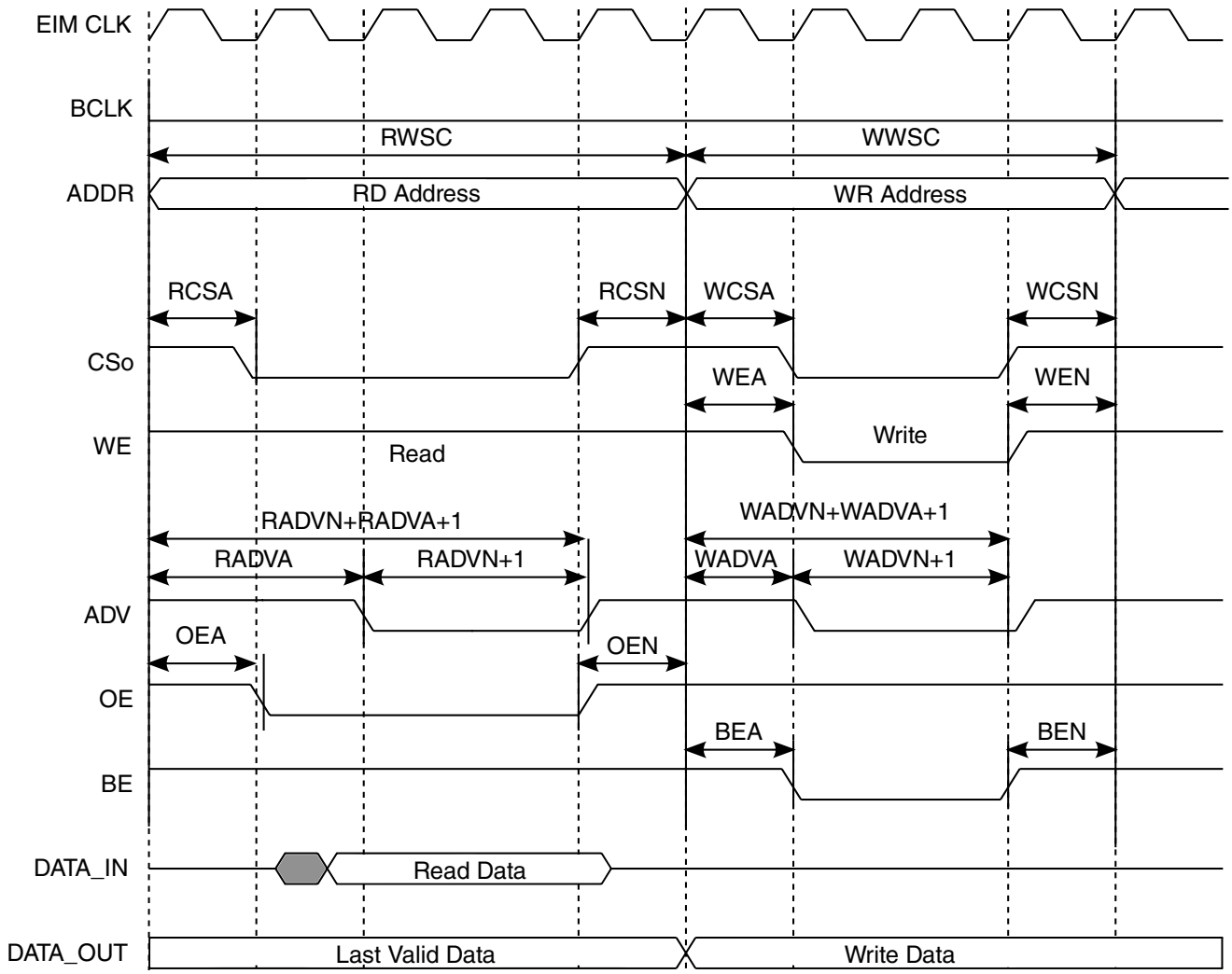


Figure 22-4.
RCSA=1,RADVA=2,OEA=1,RADVN=1,RCSN=1,OEN=1,WCSA=1,WEA=1,WADVA=1,BEA=1,WADVN=1,WCSN=1,WEN=1,BEN=1

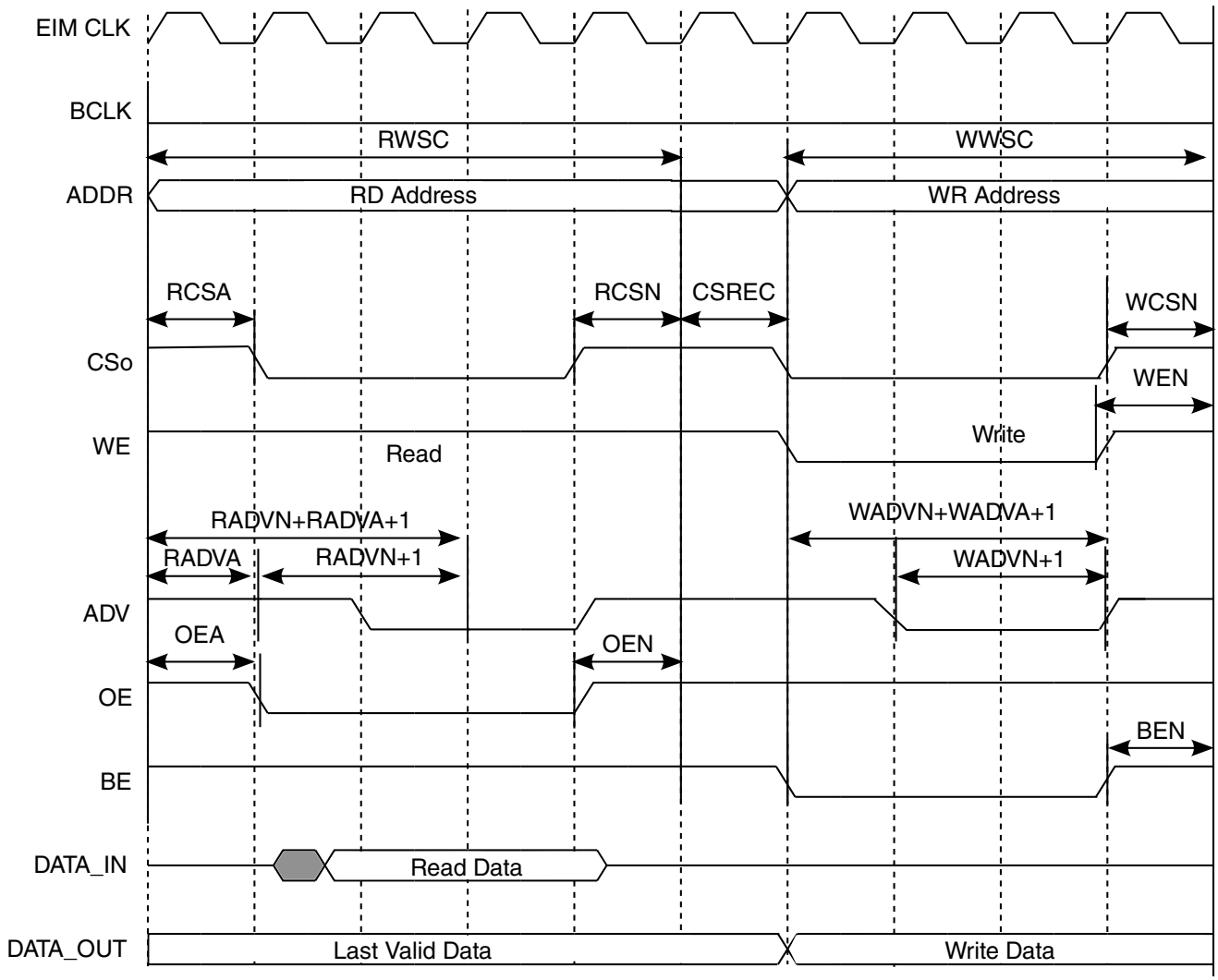


Figure 22-5.
RWSC=5,RCSA=1,RCSN=1,RADVA=1,RADVN=1,OEA=1,OEN=1,WWSC=4,WCSA=0,WCSN=1,WEA=0,WEN=1,WADVA=1,WADVN=1,BEA=0,BEN=1,CSREC=1

22.8.4 Asynchronous Read/Write Using RAL, WAL and CSREC

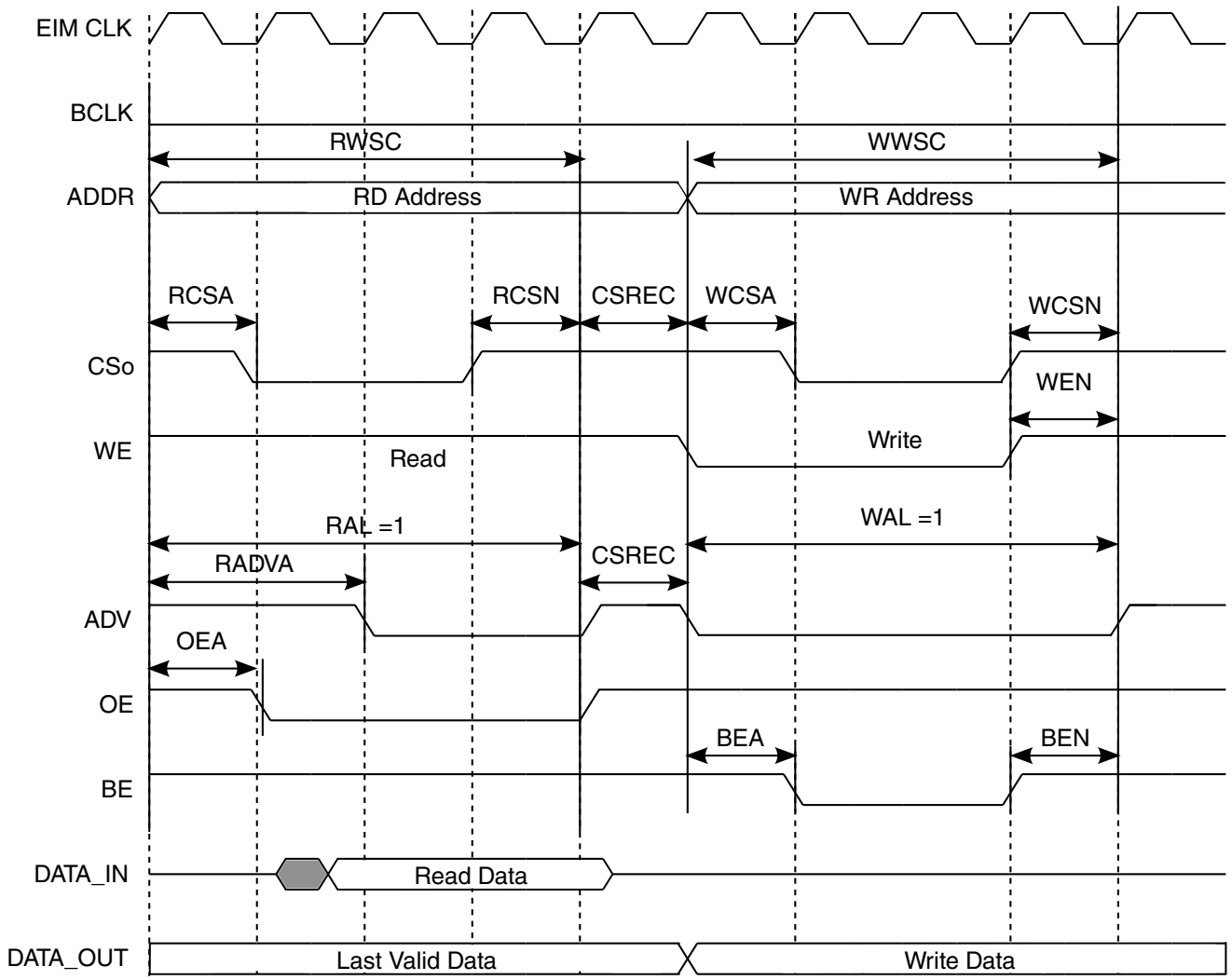


Figure 22-6.
RAL=1,RCSN=1,RADVA=2,OEA=1,RCSN=1,CSREC=1,WCSA=1,WEA=0,WADVA=0,BEA=1,WAL=1,WCSN=1,WEN=1,BEN=1

22.8.5 Consecutive Asynchronous Write Memory Accesses Timing Diagram

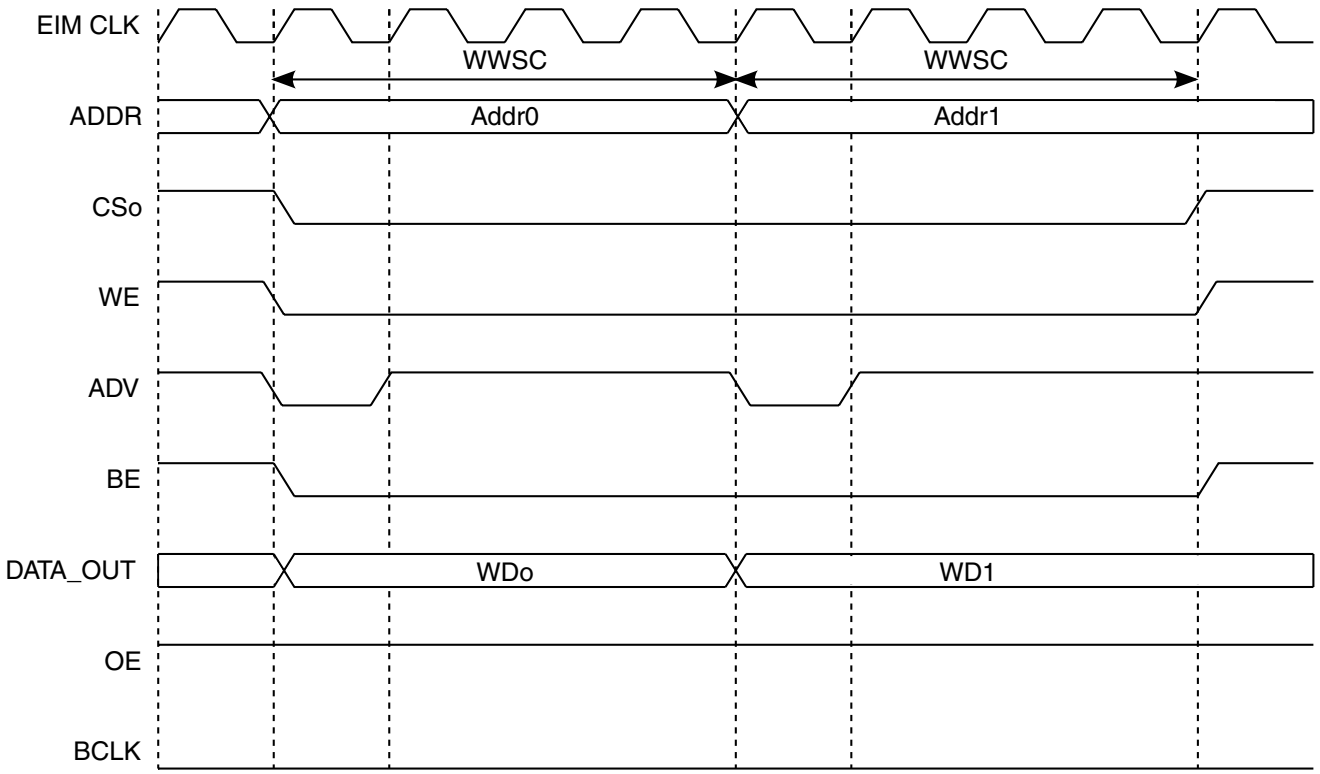


Figure 22-7. WWSC=4,WCSA=0,WEA=0,WADVA=0,BEA=0,WCSN=0,WEN=0,WADVN=0,BEN=0,CSREC=0

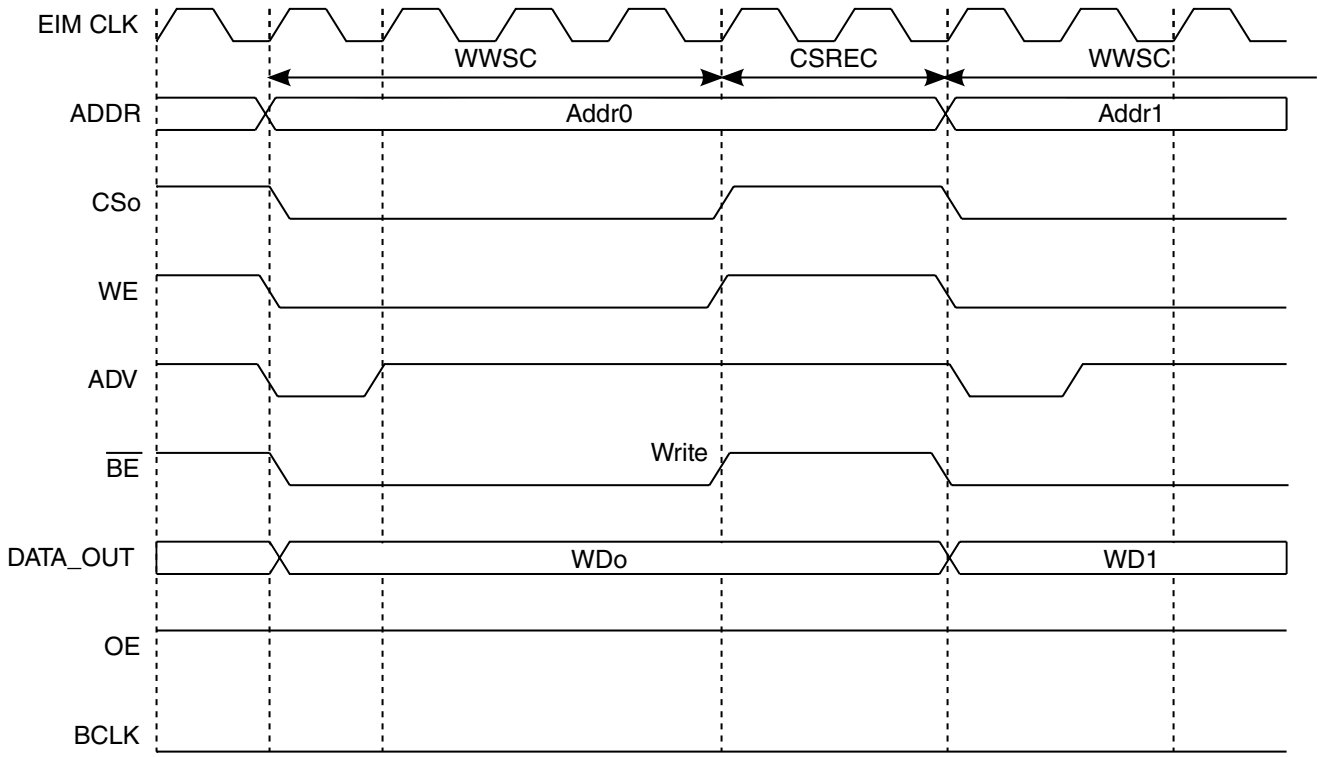


Figure 22-8. WWSC=4,WCSA=0,WEA=0,WADVA=0,BEA=0,WCSN=0,WEN=0,WADVN=0,BEN=0,CSREC=2

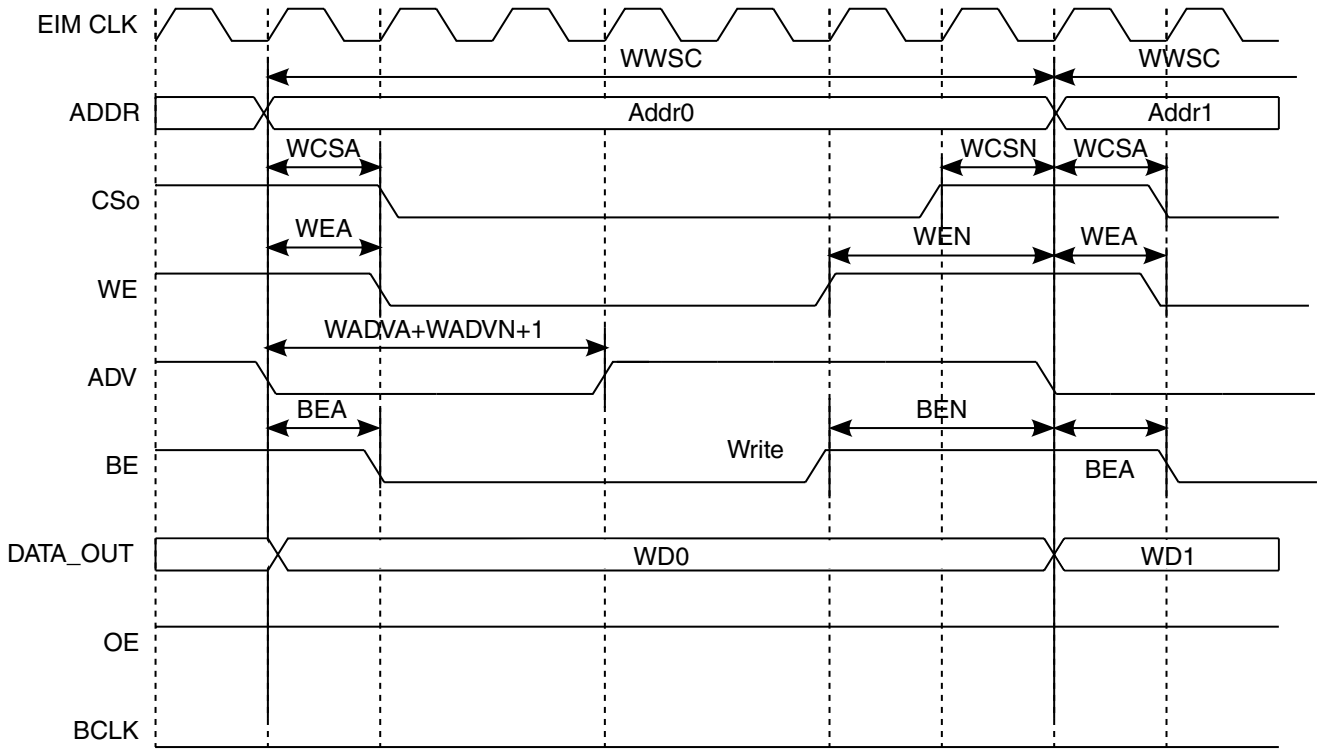


Figure 22-9.
WWSC=7,WCSA=1,WCSN=1,WEA=1,WEN=2,WADVA=0,WADVN=2,BEA=1,BEN=2

22.8.6 Consecutive Asynchronous Read Memory Accesses Timing Diagram

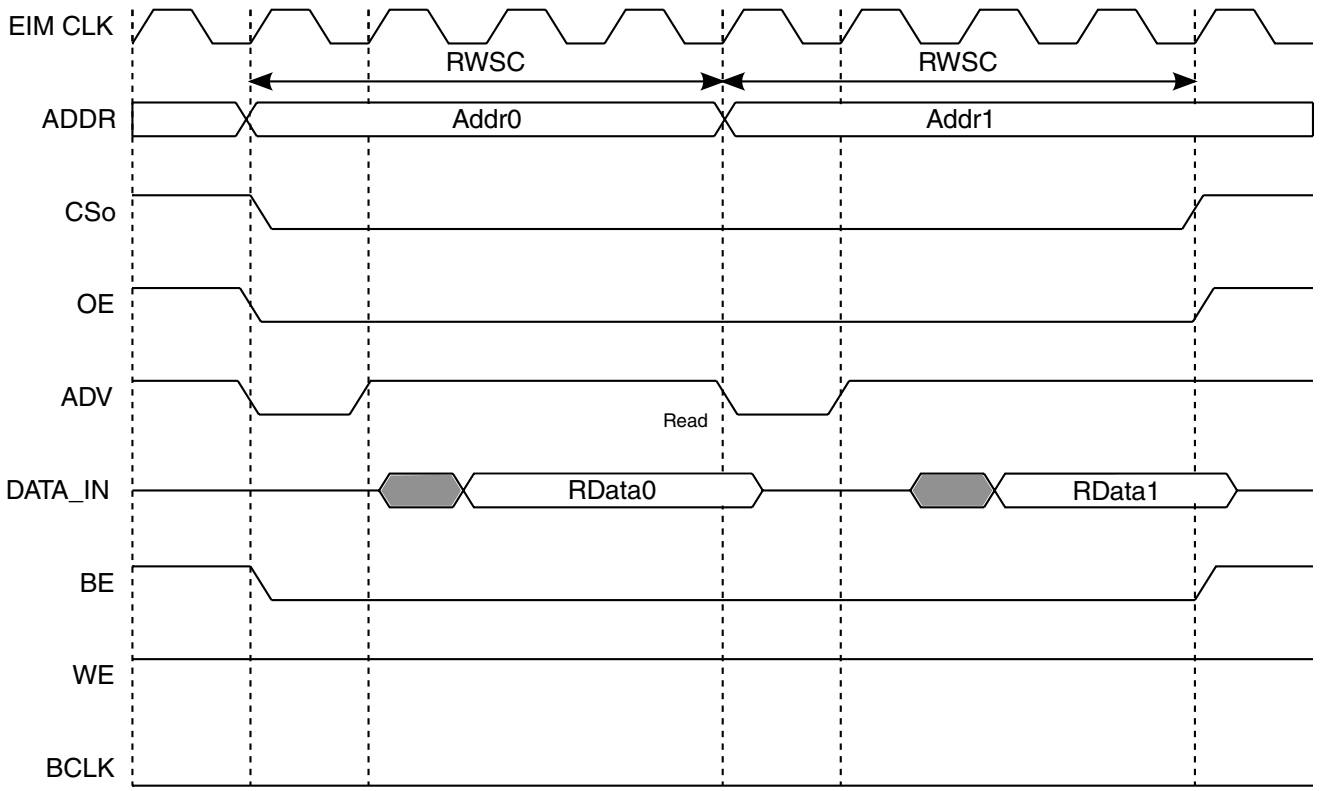


Figure 22-10. RWSC=4,RCSA=0,OEA=0,RADVA=0,RCSN=0,OEN=0,RADVN=0,CSREC=0

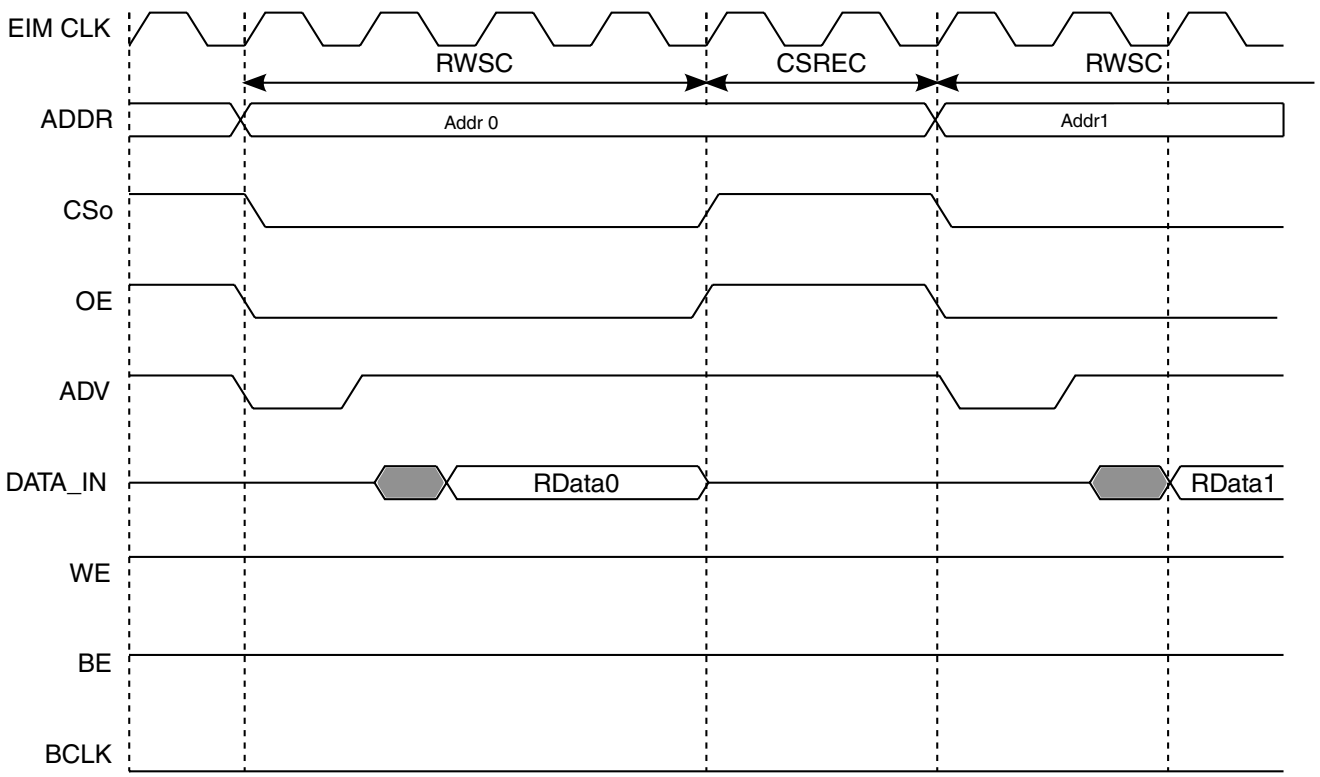


Figure 22-11. RWSC=4,RCSA=0,OEA=0,RADVA=0,RCSN=0,OEN=0,RADVN=0,CSREC=2

22.8.7 Burst (Synchronous Mode) Read Memory Accesses Timing Diagram - BCD=0

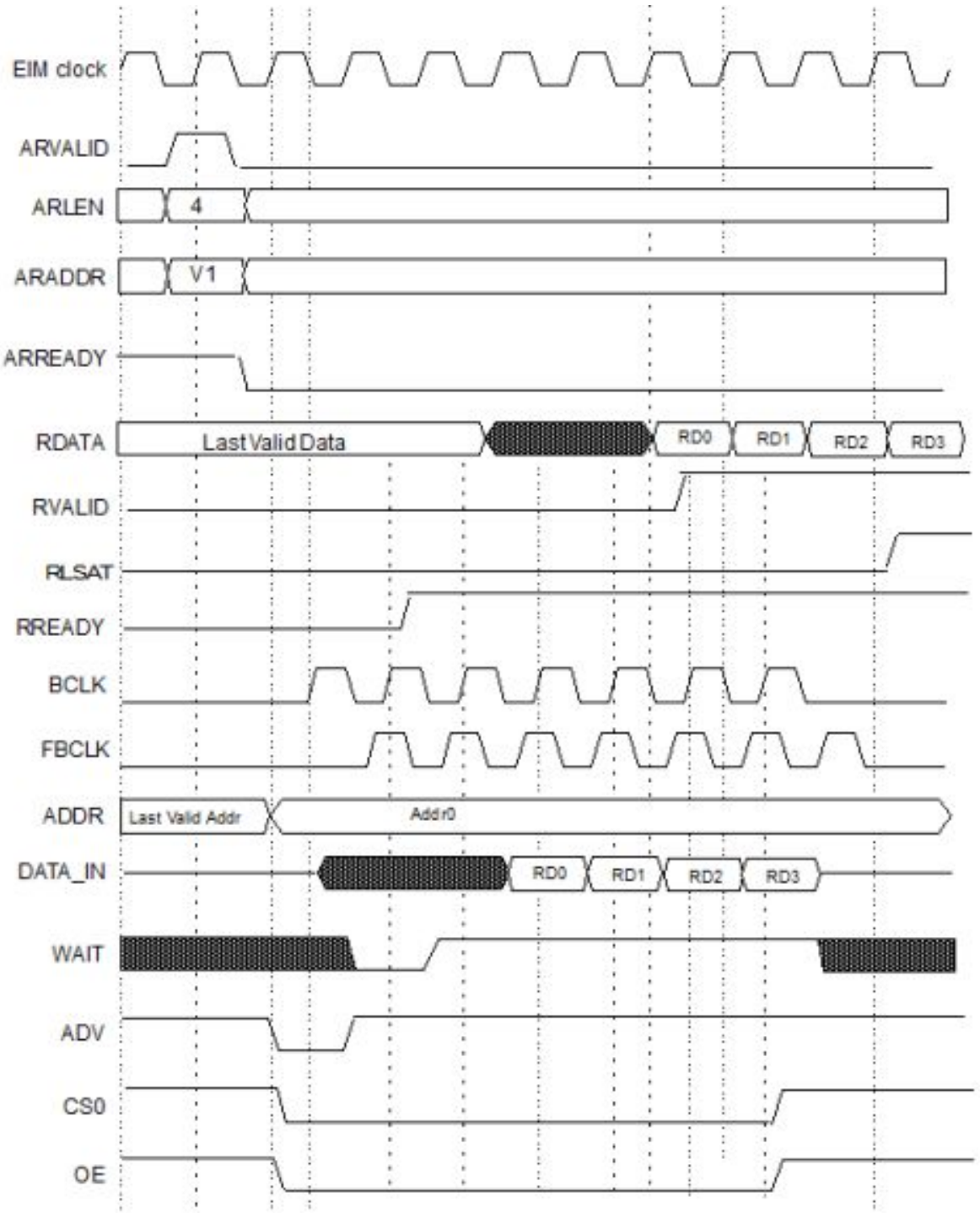


Figure 22-12. SRD=1,BCD=0,BCS=0,RWSC=1,RADVA=0,RADVN=0,RFL=0,RL=0

22.8.8 Burst (Synchronous Mode) Read Memory Accesses Timing Diagram - BCD=1

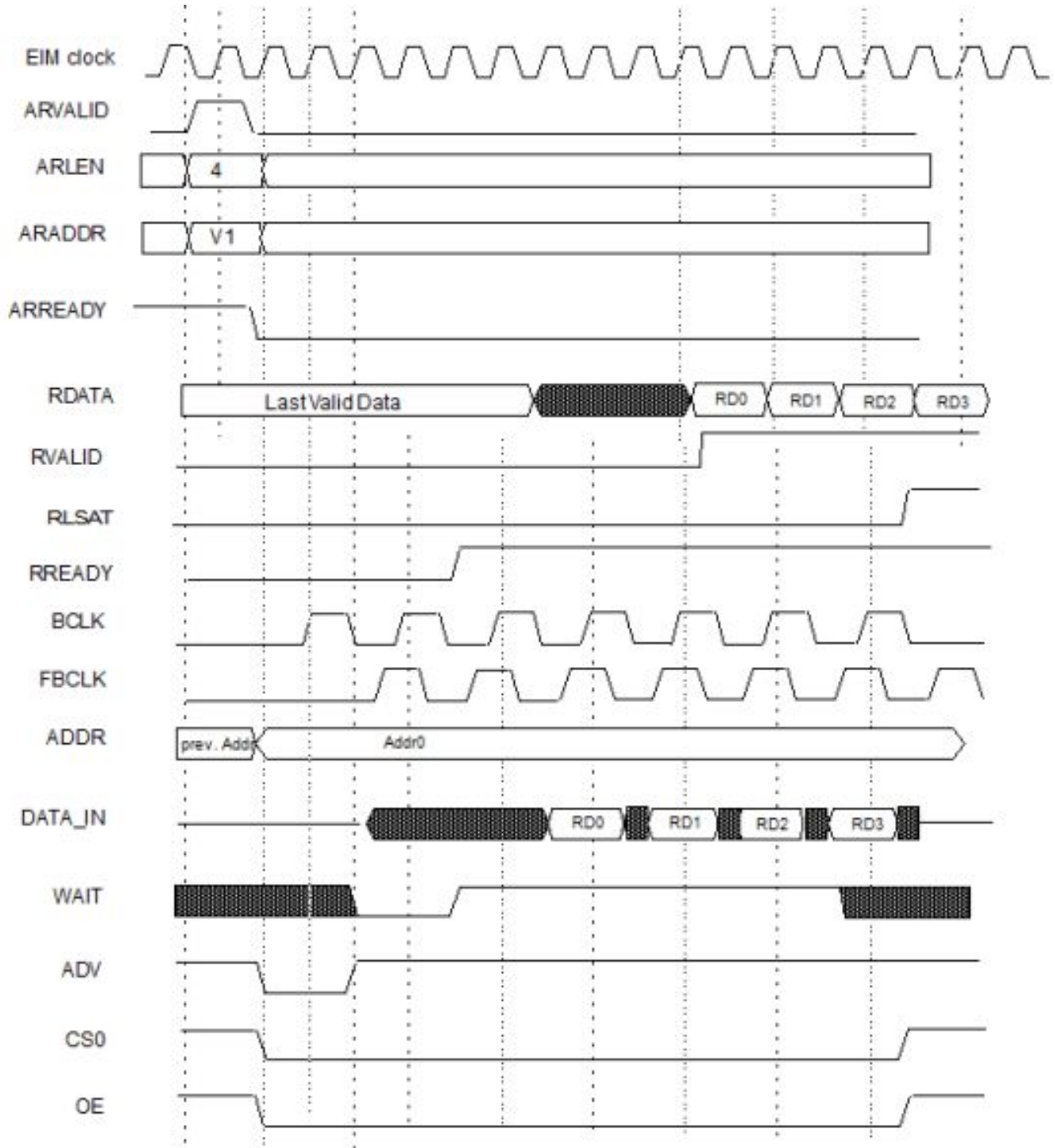
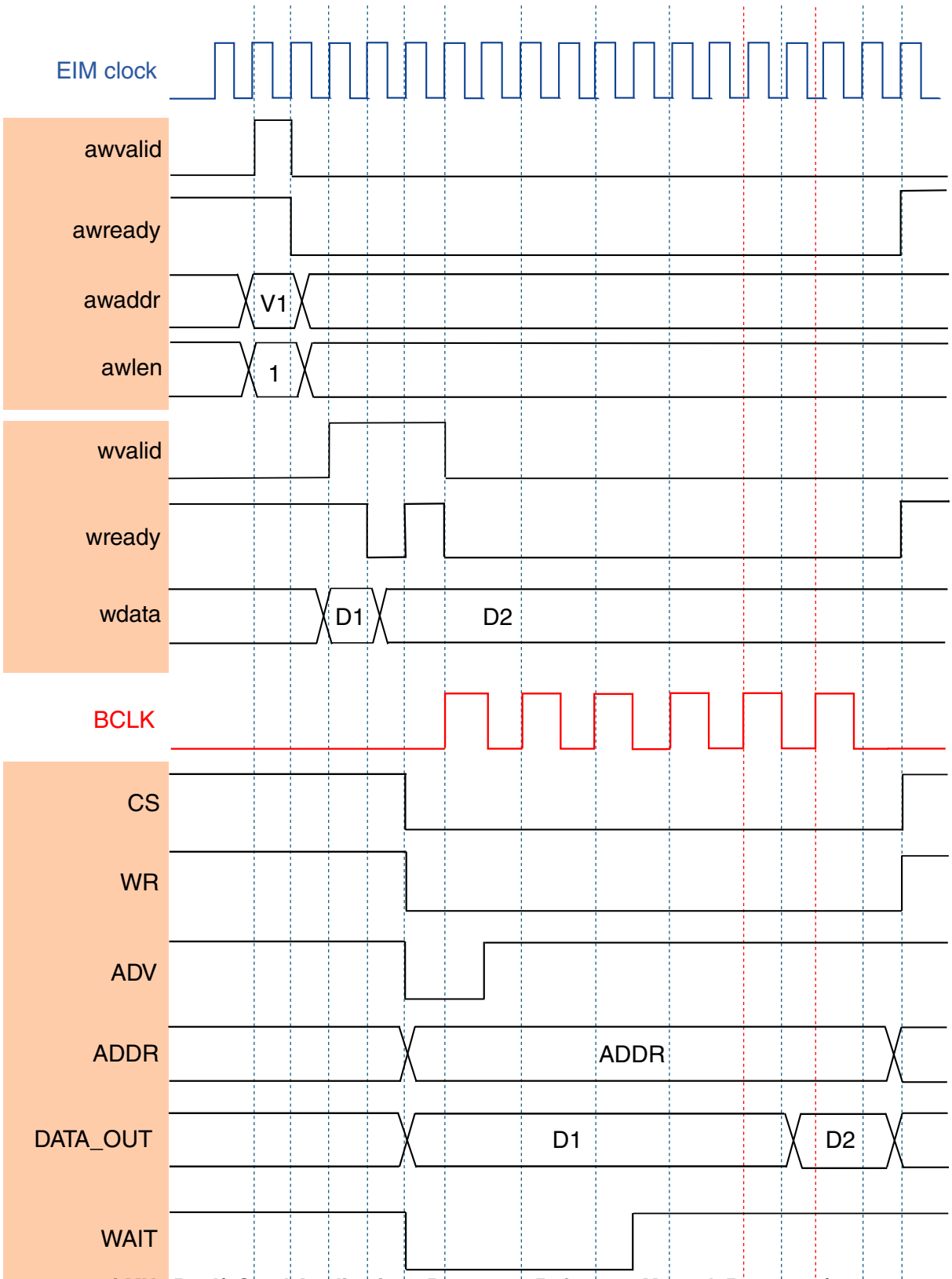


Figure 22-13. BCD=1, RL = 3

22.8.9 Burst (Synchronous Mode) Write Memory Access Timing - BCD=1



22.8.10 Asynchronous Page Mode Access

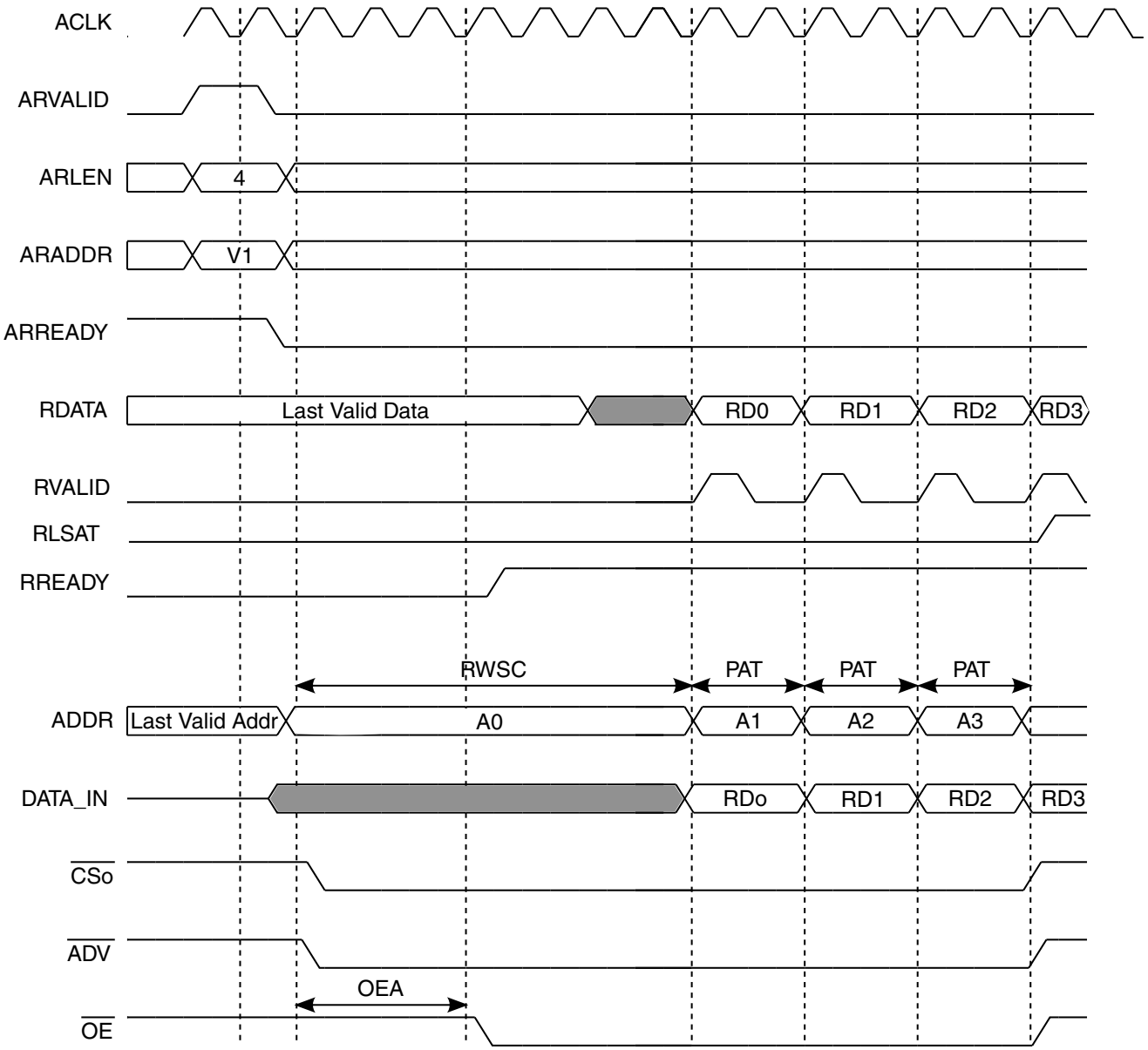


Figure 22-14. PAT = 2

22.8.11 DTACK Mode - AXI Single Access

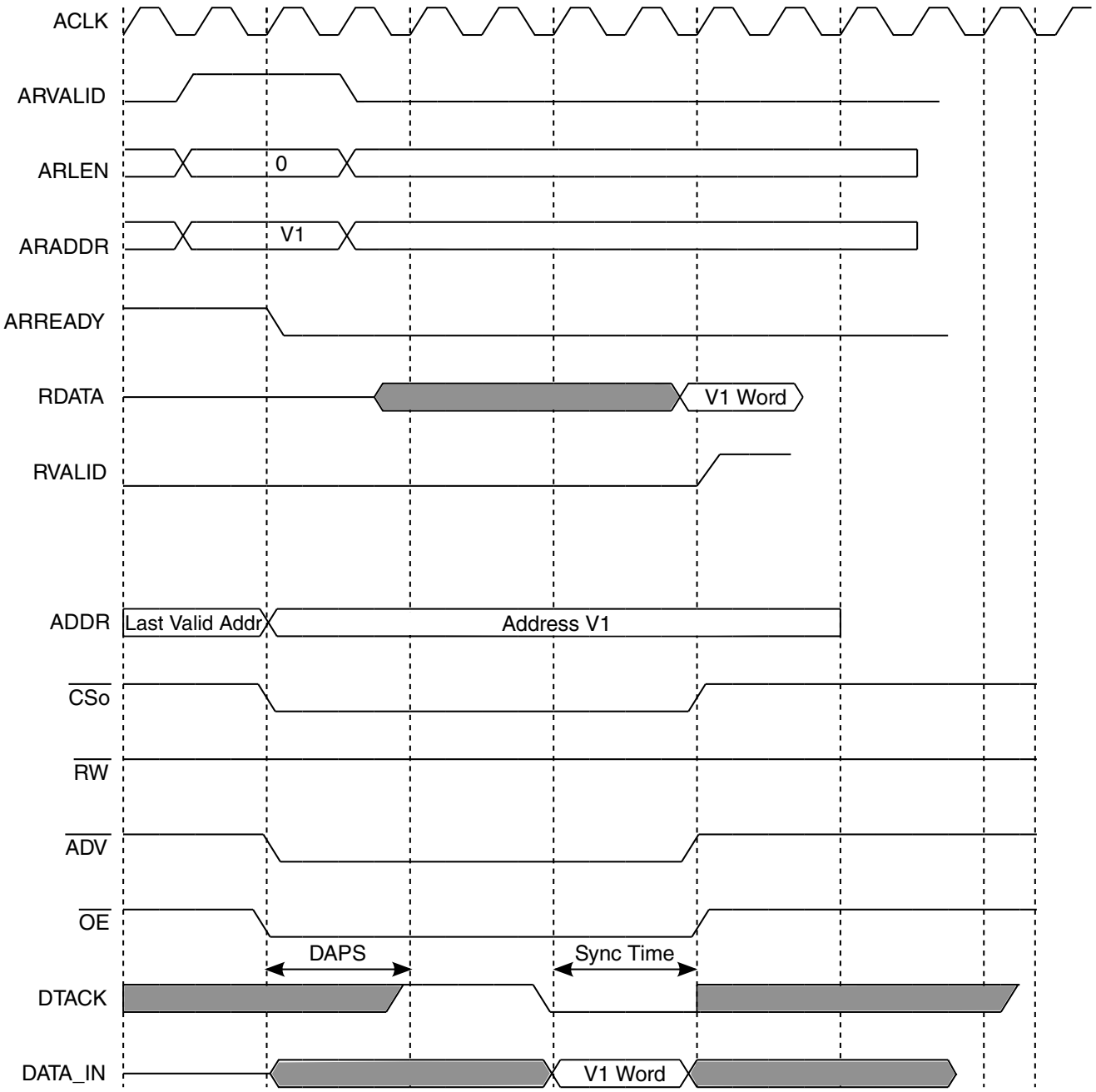


Figure 22-15. DAPS = 2

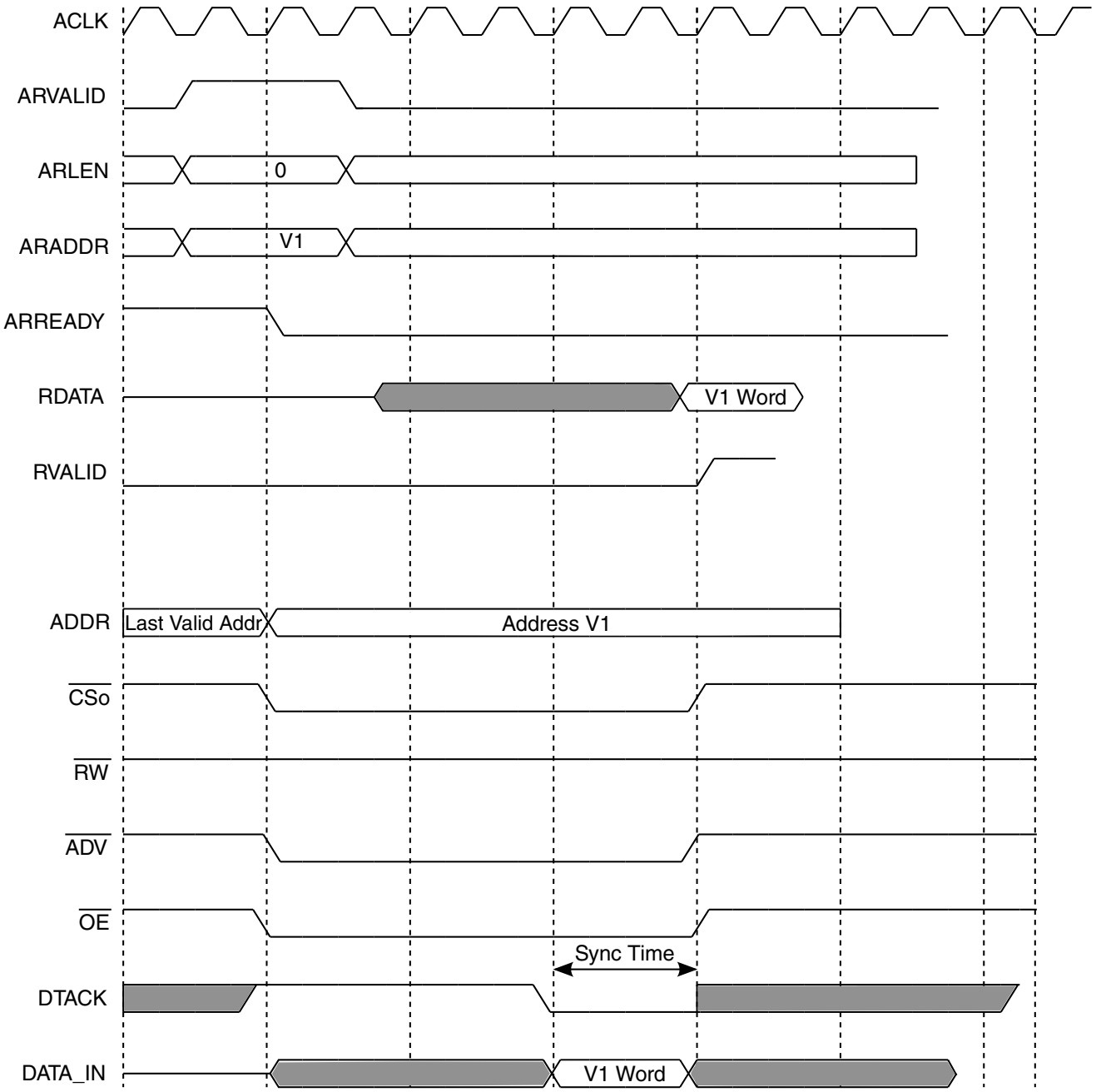


Figure 22-16. DAPS = 0

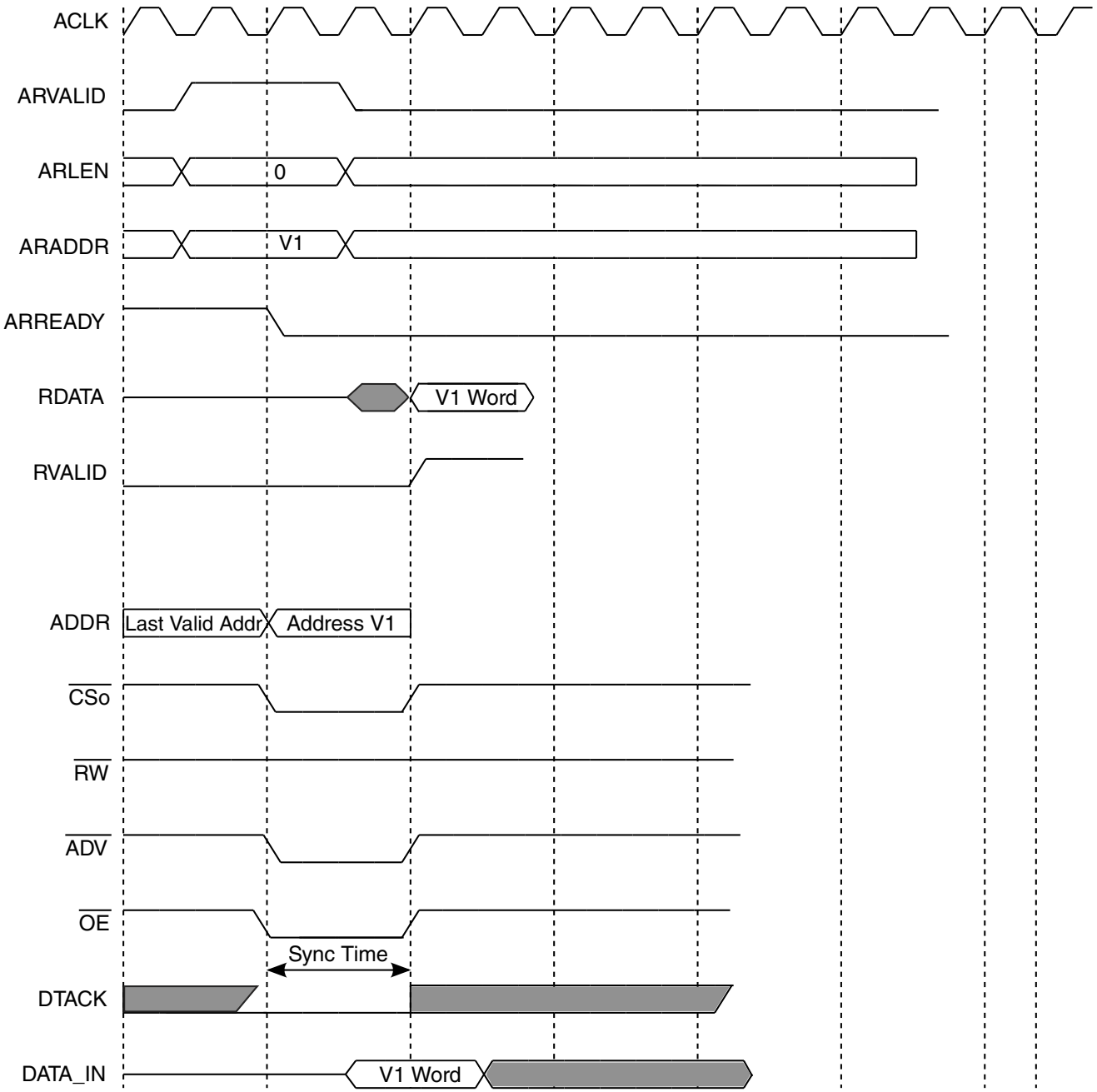


Figure 22-17. DAPS = 0

22.8.12 DTACK Mode - AXI Single Write Access

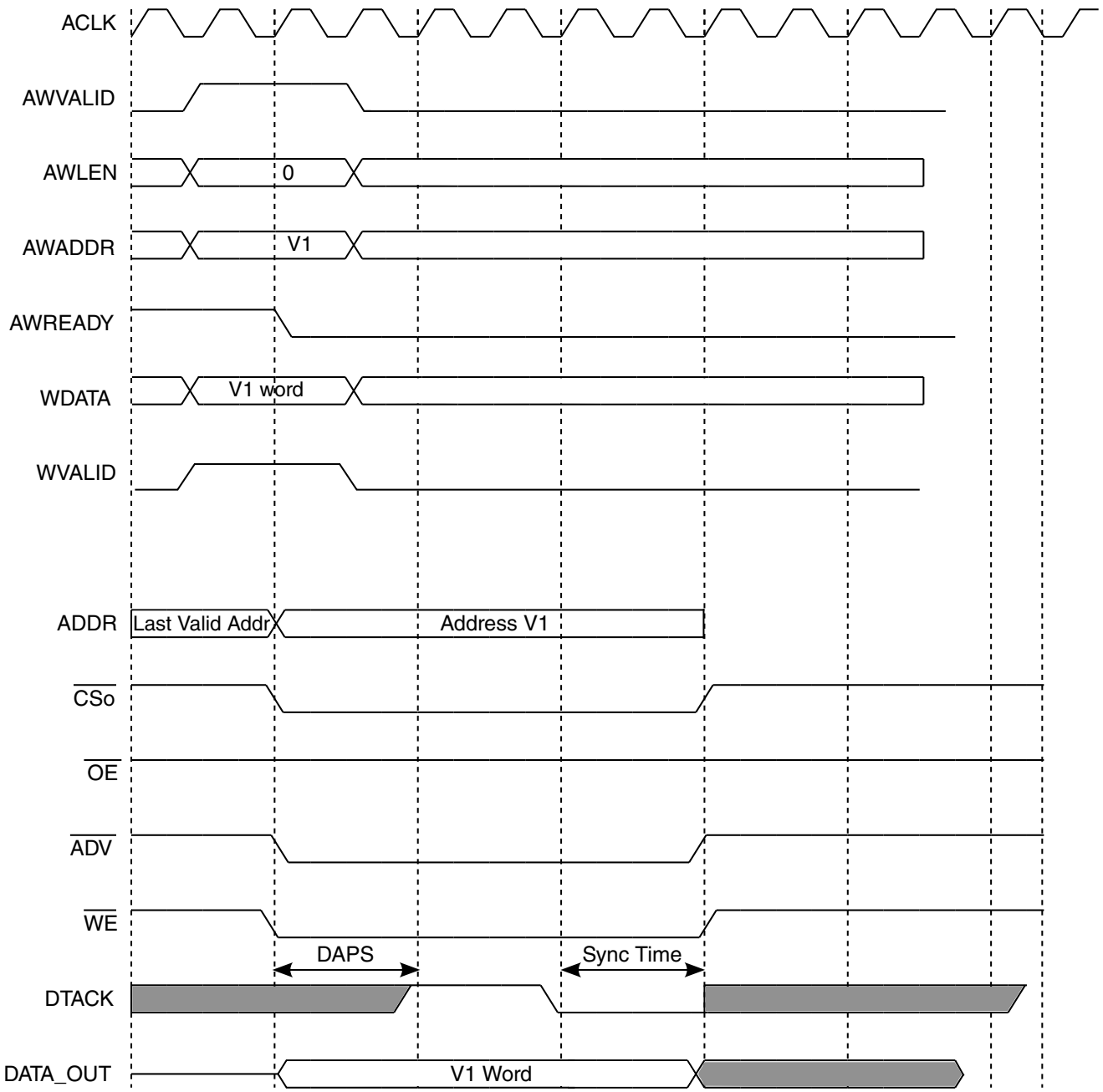


Figure 22-18. DAPS = 2

22.8.13 DTACK Mode - AXI Burst Access

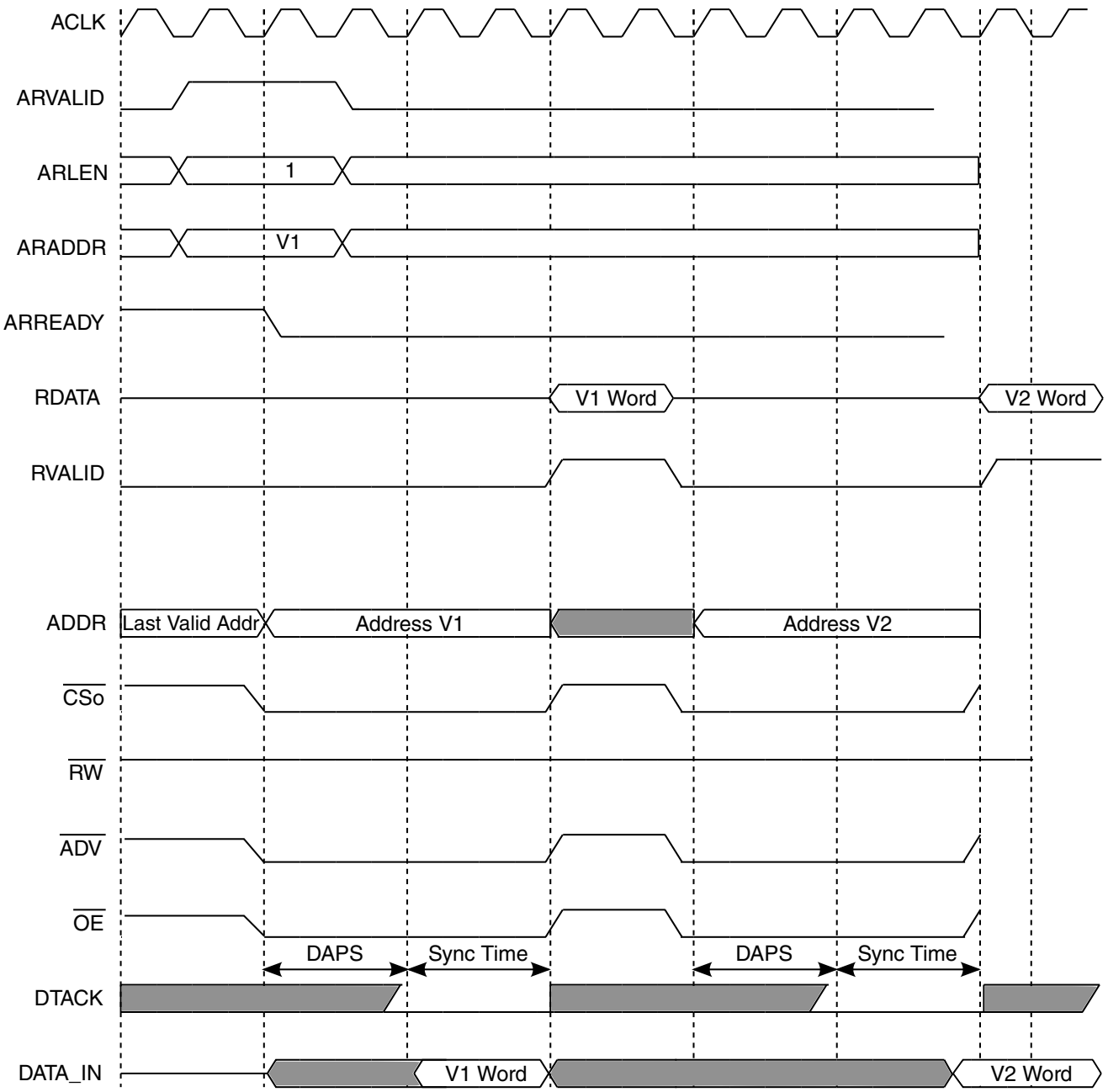


Figure 22-19. DAPS = 2 CSREC = 2

22.9 EIM Memory Map/Register Definition

The EIM includes 33 user-accessible 32-bit registers. The the EIM Configuration Register (EIM_WCR) contains control bits that configure the EIM for certain operation modes.

The 160 bits used to control Individual Chip Select are divided into five registers:

- Chip Select n General Configuration Register 1 (EIM_CSnGCR1)
- Chip Select n General Configuration Register 2 (EIM_CSnGCR2)
- Chip Select n Read Configuration Register 1 (EIM_CSnRCR1)
- Chip Select n Read Configuration Register 2 (EIM_CSnRCR2)
- Chip Select n Write Configuration Register (EIM_CSnWCR)

In addition there are 3 general registers: EIM_WCR, EIM_WIAR & EIM_EAR.

NOTE

- All EIM registers are sampled by IPG_CLK_S, therefore IPG_CLK_S must be active when accessing through IP bus.
- Read access from all registers (except EIM_WIAR & EIM_EAR) will generate one IPG_XFR_WAIT cycle.
- Read access from EIM_WIAR & EIM_EAR will generate six IPG_XFR_WAIT cycles.
- Write access to all registers (except EIM_EAR) will generate three IPG_XFR_WAIT cycles.
- Write access to EIM_EAR will generate six IPG_XFR_WAIT cycles.

EIM memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21B_8000	Chip Select n General Configuration Register 1 (EIM_CS0GCR1)	32	R/W	0061_0088h	22.9.1/1059
21B_8004	Chip Select n General Configuration Register 2 (EIM_CS0GCR2)	32	R/W	0000_1010h	22.9.2/1063
21B_8008	Chip Select n Read Configuration Register 1 (EIM_CS0RCR1)	32	R/W	1C00_2000h	22.9.3/1064
21B_800C	Chip Select n Read Configuration Register 2 (EIM_CS0RCR2)	32	R/W	0000_0000h	22.9.4/1067
21B_8010	Chip Select n Write Configuration Register 1 (EIM_CS0WCR1)	32	R/W	1C00_0000h	22.9.5/1068
21B_8014	Chip Select n Write Configuration Register 2 (EIM_CS0WCR2)	32	R/W	0000_0000h	22.9.6/1071
21B_8018	Chip Select n General Configuration Register 1 (EIM_CS1GCR1)	32	R/W	0061_0088h	22.9.1/1059

Table continues on the next page...

EIM memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21B_801C	Chip Select n General Configuration Register 2 (EIM_CS1GCR2)	32	R/W	0000_1010h	22.9.2/1063
21B_8020	Chip Select n Read Configuration Register 1 (EIM_CS1RCR1)	32	R/W	1C00_2000h	22.9.3/1064
21B_8024	Chip Select n Read Configuration Register 2 (EIM_CS1RCR2)	32	R/W	0000_0000h	22.9.4/1067
21B_8028	Chip Select n Write Configuration Register 1 (EIM_CS1WCR1)	32	R/W	1C00_0000h	22.9.5/1068
21B_802C	Chip Select n Write Configuration Register 2 (EIM_CS1WCR2)	32	R/W	0000_0000h	22.9.6/1071
21B_8030	Chip Select n General Configuration Register 1 (EIM_CS2GCR1)	32	R/W	0061_0088h	22.9.1/1059
21B_8034	Chip Select n General Configuration Register 2 (EIM_CS2GCR2)	32	R/W	0000_1010h	22.9.2/1063
21B_8038	Chip Select n Read Configuration Register 1 (EIM_CS2RCR1)	32	R/W	1C00_2000h	22.9.3/1064
21B_803C	Chip Select n Read Configuration Register 2 (EIM_CS2RCR2)	32	R/W	0000_0000h	22.9.4/1067
21B_8040	Chip Select n Write Configuration Register 1 (EIM_CS2WCR1)	32	R/W	1C00_0000h	22.9.5/1068
21B_8044	Chip Select n Write Configuration Register 2 (EIM_CS2WCR2)	32	R/W	0000_0000h	22.9.6/1071
21B_8048	Chip Select n General Configuration Register 1 (EIM_CS3GCR1)	32	R/W	0061_0088h	22.9.1/1059
21B_804C	Chip Select n General Configuration Register 2 (EIM_CS3GCR2)	32	R/W	0000_1010h	22.9.2/1063
21B_8050	Chip Select n Read Configuration Register 1 (EIM_CS3RCR1)	32	R/W	1C00_2000h	22.9.3/1064
21B_8054	Chip Select n Read Configuration Register 2 (EIM_CS3RCR2)	32	R/W	0000_0000h	22.9.4/1067
21B_8058	Chip Select n Write Configuration Register 1 (EIM_CS3WCR1)	32	R/W	1C00_0000h	22.9.5/1068
21B_805C	Chip Select n Write Configuration Register 2 (EIM_CS3WCR2)	32	R/W	0000_0000h	22.9.6/1071
21B_8090	EIM Configuration Register (EIM_WCR)	32	R/W	0000_0020h	22.9.7/1072
21B_8094	EIM IP Access Register (EIM_WIAR)	32	R/W	0000_0010h	22.9.8/1073
21B_8098	Error Address Register (EIM_EAR)	32	R/W	0000_0000h	22.9.9/1074

22.9.1 Chip Select n General Configuration Register 1 (EIM_CS*n*GCR1)

Address: 21B_8000h base + 0h offset + (24d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	PSZ				WP	GBC			AUS	CSREC			SP	DSZ		
Reset	0	0	0	0	0	0	0	0	0				0			

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	BCS		BCD		WC	BL			CREP	CRE	RFL	WFL	MUM	SRD	SWR	CSEN
Reset	0	0	0	0	0	0	0	0	1	0	0	0		0	0	0

EIM_CS*n*GCR1 field descriptions

Field	Description
31–28 PSZ	Page Size. This bit field indicates memory page size in words (word is defined by the DSZ field). PSZ is used when fix latency mode is applied, WFL=1 for sync. write accesses, RFL=1 for sync. Read accesses. When working in fix latency mode WAIT signal from the external device is not being monitored, PSZ is used to determine if page boundary is reached and renewal of access is preformed. This bit field is ignored when sync. Mode is disabled or fix latency mode is not being used for write or read access separately. It can be valid for both access type, read or write, or only for one type, according to configuration. PSZ is cleared by a hardware reset. 0000 8 words page size 0001 16 words page size 0010 32 words page size 0011 64 words page size 0100 128 words page size 0101 256 words page size 0110 512 words page size 0111 1024 (1k) words page size 1000 2048 (2k) words page size 1001 - 1111 Reserved
27 WP	Write Protect. This bit prevents writes to the address range defined by the corresponding chip select. WP is cleared by a hardware reset. 0 Writes are allowed in the memory range defined by chip. 1 Writes are prohibited. All attempts to write to an address mapped by this chip select result in a error response and no assertion of the chip select output.
26–24 GBC	Gap Between Chip Selects. This bit field, according to the settings shown below, determines the minimum time between end of access to the current chip select and start of access to different chip select. GBC is cleared by a hardware reset.

Table continues on the next page...

EIM_CS*n*GCR1 field descriptions (continued)

Field	Description
	Example settings: 000 minimum of 0 EIM clock cycles before next access from different chip select (async. mode only) 001 minimum of 1 EIM clock cycles before next access from different chip select 010 minimum of 2 EIM clock cycles before next access from different chip select 111 minimum of 7 EIM clock cycles before next access from different chip select
23 AUS	Address UnShifted. This bit indicates an unshifted mode for address assertion for the relevant chip select accesses. AUS bit is cleared by hardware reset. 0 Address shifted according to port size (DSZ config.) 1 Address unshifted
22–20 CSREC	CS Recovery. This bit field, according to the settings shown below, determines the minimum pulse width of CS, OE, and WE control signals before executing a new back to back access to the same chip select. CSREC is cleared by a hardware reset. **NOTE:** The reset value for EIM_CS0GCR1, CSREC[2:0] is 0b110. For EIM_CS1GCR1 - EIM_CS5GCR, the reset value is 0b000. Example settings: 000 0 EIM clock cycles minimum width of CS, OE and WE signals (read async. mode only) 001 1 EIM clock cycles minimum width of CS, OE and WE signals 010 2 EIM clock cycles minimum width of CS, OE and WE signals 111 7 EIM clock cycles minimum width of CS, OE and WE signals
19 SP	Supervisor Protect. This bit prevents accesses to the address range defined by the corresponding chip select when the access is attempted in the User mode. SP is cleared by a hardware reset. 0 User mode accesses are allowed in the memory range defined by chip select. 1 User mode accesses are prohibited. All attempts to access an address mapped by this chip select in User mode results in an error response and no assertion of the chip select output.
18–16 DSZ	Data Port Size. This bit field defines the width of an external device's data port as shown below. **NOTE:** Only async. access supported for 8 bit port. **NOTE:** The reset value for EIM_CS0GCR1, DSZ[2] = 0, DSZ[1:0] = EIM_BOOT[1:0]. For EIM_CS1GCR1 - EIM_CS5GCR1, the reset value is 0b001. 000 Reserved. 001 16 bit port resides on DATA[15:0] 010 16 bit port resides on DATA[31:16] 011 32 bit port resides on DATA[31:0] 100 8 bit port resides on DATA[7:0] 101 8 bit port resides on DATA[15:8] 110 8 bit port resides on DATA[23:16] 111 8 bit port resides on DATA[31:24]
15–14 BCS	Burst Clock Start. When SRD=1 or SWR=1,this bit field determines the number of EIM clock cycles delay from start of access before the first rising edge of BCLK is generated. When BCD=0 value of BCS=0 results in a half clock delay after the start of access. For other values of BCD a one clock delay after the start of access is applied, not an immediate assertion. BCS is cleared by a hardware reset. 00 0 EIM clock cycle additional delay

Table continues on the next page...

EIM_CS*n*GCR1 field descriptions (continued)

Field	Description
	01 1 EIM clock cycle additional delay 10 2 EIM clock cycle additional delay 11 3 EIM clock cycle additional delay
13–12 BCD	Burst Clock Divisor. This bit field contains the value used to program the burst clock divisor for BCLK generation. It is used to divide the internal EIMbus frequency. BCD is cleared by a hardware reset. **NOTE:** For other then the mentioned below frequency such as 104 MHz, EIM clock (input clock) should be adjust accordingly. 00 Divide EIM clock by 1 01 Divide EIM clock by 2 10 Divide EIM clock by 3 11 Divide EIM clock by 4
11 WC	Write Continuous. The WI bit indicates that write access to the memory are always continuous accesses regardless of the BL field value. WI is cleared by hardware reset. 0 Write access burst length occurs according to BL value. 1 Write access burst length is continuous.
10–8 BL	Burst Length. The BL bit field indicates memory burst length in words (word is defined by the DSZ field) and should be properly initialized for mixed wrap/increment accesses support. Continuous BL value corresponds to continuous burst length setting of the external memory device. For fix memory burst size, type is always wrap. In case not matching wrap boundaries in both the memory (BL field) and Master access on the current address, EIM update address on the external device address bus and regenerates the access. BL is cleared by a hardware reset. When APR=1, Page Read Mode is applied, BL determine the number of words within the read page burst. BL is cleared by a hardware reset for EIM_CS0GCR1 - EIM_CS5GCR1. 000 4 words Memory wrap burst length (read page burst size when APR = 1) 001 8 words Memory wrap burst length (read page burst size when APR = 1) 010 16 words Memory wrap burst length (read page burst size when APR = 1) 011 32 words Memory wrap burst length (read page burst size when APR = 1) 100 Continuous burst length (2 words read page burst size when APR = 1) 101 Reserved 110 Reserved 111 Reserved
7 CREP	Configuration Register Enable Polarity. This bit indicates CRE memory pin assertion state, active-low or active-high, while executing a memory register set command to the external device (PSRAM memory type). CREP is set by a hardware reset. **NOTE:** Whenever PSRAM is connected the CREP value must be correct also for accesses where CRE is disabled. For Non-PSRAM memory CREP value should be 1. 0 CRE signal is active low 1 CRE signal is active high
6 CRE	Configuration Register Enable. This bit indicates CRE memory pin state while executing a memory register set command to PSRAM external device. CRE is cleared by a hardware reset.

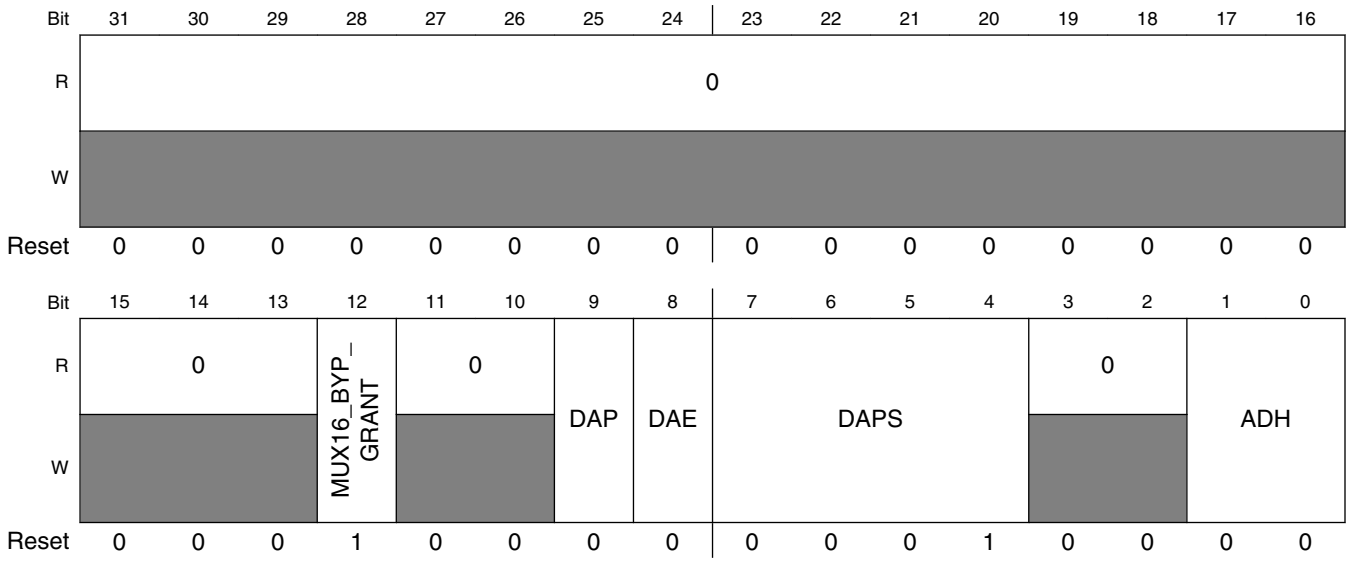
Table continues on the next page...

EIM_CS*n*GCR1 field descriptions (continued)

Field	Description
	0 CRE signal use is disable 1 CRE signal use is enable
5 RFL	Read Fix Latency. This bit field determine if the controller is monitoring the WAIT signal from the External device connected to the chip select (handshake mode - fix or variable data latency) or if it start sampling data according to RWSC field, it only valid in synchronous mode. RFL is cleared by a hardware reset. When RFL=1 Burst access is terminated on page boundary and resume on the following page according to BL bit field configuration, because WAIT signal is not monitored from the external device. 0 the External device WAIT signal is being monitored, and it reflect the external data bus state 1 the state of the External devices is determined internally (Fix latency mode only)
4 WFL	Write Fix Latency. This bit field determine if the controller is monitoring the WAIT signal from the External device connected to the chip select (handshake mode - fix or variable data latency) or if it start data transfer according to WWSC field, it only valid in synchronous mode. WFL is cleared by a hardware reset. When WFL=1 Burst access is terminated on page boundary and resume on the following page according to BL bit field configuration, because WAIT signal is not monitored from the external device 0 the External device WAIT signal is being monitored, and it reflect the external data bus state 1 the state of the External devices is determined internally (Fix latency mode only)
3 MUM	Multiplexed Mode. This bit determines the address/data multiplexed mode for asynchronous and synchronous accesses for 8 bit, 16 bit or 32 bit devices (DSZ config. dependent). **NOTE:** The reset value for EIM_CS0GCR1[MUM] = EIM_BOOT[2]. For EIM_CS1GCR1 - EIM_CS5GCR1 the reset value is 0. 0 Multiplexed Mode disable 1 Multiplexed Mode enable
2 SRD	Synchronous Read Data. This bit field determine the read accesses mode to the External device of the chip select. The External device should be configured to the same mode as this bit implicates. SRD is cleared by a hardware reset. **NOTE:** Sync. accesses supported only for 16/32 bit port. 0 read accesses are in Asynchronous mode 1 read accesses are in Synchronous mode
1 SWR	Synchronous Write Data. This bit field determine the write accesses mode to the External device of the chip select. The External device should be configured to the same mode as this bit implicates. SWR is cleared by a hardware reset. **NOTE:** Sync. accesses supported only for 16/32 bit port. 0 write accesses are in Asynchronous mode 1 write accesses are in Synchronous mode
0 CSEN	CS Enable. This bit controls the operation of the chip select pin. CSEN is set by a hardware reset for CSGCR0 to allow external boot operation. CSEN is cleared by a hardware reset to CSGCR1-CSGCR5. **NOTE:** Reset value for EIM_CS0GCR1 for CSEN is 1. For EIM_CS1GCR1-CS1GCR5 reset value is 0. 0 Chip select function is disabled; attempts to access an address mapped by this chip select results in an error respond and no assertion of the chip select output 1 Chip select is enabled, and is asserted when presented with a valid access.

22.9.2 Chip Select n General Configuration Register 2 (EIM_CS*n*GCR2)

Address: 21B_8000h base + 4h offset + (24d × i), where i=0d to 3d



EIM_CS*n*GCR2 field descriptions

Field	Description
31–13 Reserved	This read-only field is reserved and always has the value 0.
12 MUX16_BYP_GRANT	Muxed 16 bypass grant. This bit when asserted causes EIM to bypass the grant/ack. arbitration with NFC (only for 16 bit muxed mode accesses). 0 EIM waits for grant before driving a 16 bit muxed mode access to the memory. 1 EIM ignores the grant signal and immediately drives a 16 bit muxed mode access to the memory.
11–10 Reserved	This read-only field is reserved and always has the value 0.
9 DAP	Data Acknowledge Polarity. This bit indicates DTACK memory pin assertion state, active-low or active-high, while executing an async access using DTACK signal from the external device. DAP is cleared by a hardware reset. 0 DTACK signal is active high 1 DTACK signal is active low
8 DAE	Data Acknowledge Enable. This bit indicates external device is using DTACK pin as strobe/terminator of an async. access. DTACK signal may be used only in asynchronous single read (APR=0) or write accesses. DTACK poling start point is set by DAPS bit field. polarity of DTACK is set by DAP bit field. DAE is cleared by a hardware reset. 0 DTACK signal use is disable 1 DTACK signal use is enable
7–4 DAPS	Data Acknowledge Poling Start. This bit field determine the starting point of DTACK input signal polling. DAPS is used only in asynchronous single read or write accesses.

Table continues on the next page...

EIM_CS*n*GCR2 field descriptions (continued)

Field	Description
	NOTE: Since DTACK is an async. signal the start point of DTACK signal polling is at least 3 cycles after the start of access. DAPS is cleared by a hardware reset. Example settings: 0000 3 EIM clk cycle between start of access and first $\overline{\text{DTACK}}$ check 0001 4 EIM clk cycles between start of access and first $\overline{\text{DTACK}}$ check 0010 5 EIM clk cycles between start of access and first $\overline{\text{DTACK}}$ check 0111 10 EIM clk cycles between start of access and first $\overline{\text{DTACK}}$ check 1011 14 EIM clk cycles between start of access and first DTACK check 1111 18 EIM clk cycles between start of access and first DTACK check
3–2 Reserved	This read-only field is reserved and always has the value 0.
ADH	Address hold time - This bit field determine the address hold time after ADV negation when mum = 1 (muxed mode). When mum = 0 this bit has no effect. For read accesses the field determines when the pads direction will be switched. **NOTE:** Reset value for EIM_CS0GCR2 for ADH is 10. For EIM_CS1GCR2-EIM_CS5GCR2 reset value is 00. 00 0 cycle after ADV negation 01 1 cycle after ADV negation 10 2 cycle after ADV negation 11 Reserved

22.9.3 Chip Select n Read Configuration Register 1 (EIM_CS*n*RCR1)

Address: 21B_8000h base + 8h offset + (24d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0		RWSC						0	RADVA			RAL	RADVN		
W																
Reset	0	0							0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	OEA			0	OEN			0	RCSA			0	RCSN		
W																
Reset	0				0	0	0	0	0	0	0	0	0	0	0	0

EIM_CS*n*RCR1 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

EIM_CS*n*RCR1 field descriptions (continued)

Field	Description
29–24 RWSC	Read Wait State Control. This bit field programs the number of wait-states, according to the settings shown below, for synchronous or asynchronous read access to the external device connected to the chip select. When SRD=1 and RFL=0, RWSC indicates the number of burst clock (BCLK) cycles from the start of an access, before the controller can start sample data.Since WAIT signal can be asserted one cycle before the first data can be sampled, the controller starts evaluating the WAIT signal state one cycle before, this is referred as handshake mode or variable latency mode. When SRD=1 and RFL=1, RWSC indicates the number of burst clock (BCLK) cycles from the start of an access, until the external device is ready for data transfer, this is referred as fix latency mode. When SRD=0, RFL bit is ignored, RWSC indicates the asynchronous access length and the number of EIM clock cycles from the start of access until the external device is ready for data transfer. RWSC is cleared by a hardware reset. **NOTE:** The reset value for EIM_CS0RCR1, RWSC[5:0] = 0b011100. For CG1RCR1 - CS1RCR5 the reset value is 0b000000. Example settings: 000000 Reserved 000001 RWSC value is 1 000010 RWSC value is 2 111101 RWSC value is 61 111110 RWSC value is 62 111111 RWSC value is 63
23 Reserved	This read-only field is reserved and always has the value 0.
22–20 RADVA	ADV Assertion. This bit field determines when ADV signal is asserted for synchronous or asynchronous read modes according to the settings shown below. RADVA is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between beginning of access and ADV assertion 001 1 EIM clock cycles between beginning of access and ADV assertion 010 2 EIM clock cycles between beginning of access and ADV assertion 111 7 EIM clock cycles between beginning of access and ADV assertion
19 RAL	Read ADV Low. This bit field determine ADV signal negation time. When RAL=1, RADVN bit field is ignored and ADV signal will stay asserted until end of access. When RAL=0 negation of ADV signal is according to RADVN bit field configuration.
18–16 RADVN	ADV Negation. This bit field determines when ADV signal to memory is negated during read accesses. When SRD=1 (synchronous read mode), ADV negation occurs according to the following formula: (RADVN + RADVA + BCD + BCS + 1) EIM clock cycles from start of access. When asynchronous read mode is applied (SRD=0) and RAL=0 ADV negation occurs according to the following formula: (RADVN + RADVA + 1) EIM clock cycles from start of access. RADVN is cleared by a hardware reset. **NOTE:** the reset value for EIM_CS0RCR1[RADVN] = 2. For EIM_CS1RCR1 - EIM_CS5RCR1, the reset value is 0b000. **NOTE:** This field should be configured so ADV negation will occur before the end of access. For ADV negation at the same time with the end of access user should RAL bit.

Table continues on the next page...

EIM_CS*n*RCR1 field descriptions (continued)

Field	Description
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 OEA	OE Assertion. This bit field determines when OE signal are asserted during read cycles (synchronous or asynchronous mode), according to the settings shown below. OEA is cleared by a hardware reset. In muxed mode OE assertion occurs (OEA + RADVN + RADVA + ADH +1) EIM clock cycles from start of access. **NOTE:** The reset value for EIM_CS0RCR1[OEA] is 0b000 if EIM_BOOT[2] = 0. If EIM_BOOT[2] is 1, the reset value for EIM_CS0RCR1 is 0b010. The reset value of this field for EIM_CS1RCR1 - EIM_CS5RCR1 is 0b000. Example settings: 000 0 EIM clock cycles between beginning of access and OE assertion 001 1 EIM clock cycles between beginning of access and OE assertion 010 2 EIM clock cycles between beginning of access and OE assertion 111 7 EIM clock cycles between beginning of access and OE assertion
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 OEN	OE Negation. This bit field determines when OE signal is negated during read cycles in asynchronous single mode only (SRD=0 & APR = 0), according to the settings shown below. This bit field is ignored when SRD=1. OEN is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between end of access and OE negation 001 1 EIM clock cycles between end of access and OE negation 010 2 EIM clock cycles between end of access and OE negation 111 7 EIM clock cycles between end of access and OE negation
7 Reserved	This read-only field is reserved and always has the value 0.
6–4 RCSA	Read CS Assertion. This bit field determines when CS signal is asserted during read cycles (synchronous or asynchronous mode), according to the settings shown below. RCSA is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between beginning of read access and CS assertion 001 1 EIM clock cycles between beginning of read access and CS assertion 010 2 EIM clock cycles between beginning of read access and CS assertion 111 7 EIM clock cycles between beginning of read access and CS assertion
3 Reserved	This read-only field is reserved and always has the value 0.
RCSN	Read CS Negation. This bit field determines when CS signal is negated during read cycles in asynchronous single mode only (SRD=0 & APR = 0), according to the settings shown below. This bit field is ignored when SRD=1. RCSN is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between end of read access and CS negation 001 1 EIM clock cycles between end of read access and CS negation 010 2 EIM clock cycles between end of read access and CS negation 111 7 EIM clock cycles between end of read access and CS negation

22.9.4 Chip Select n Read Configuration Register 2 (EIM_CS*n*RCR2)

Address: 21B_8000h base + Ch offset + (24d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	APR	PAT			0		RL		0	RBEA			RBE	RBEN		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EIM_CS*n*RCR2 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15 APR	Asynchronous Page Read. This bit field determine the asynchronous read mode to the external device. When APR=0, the async. read access is done as single word (where word is defined by the DSZ field). when APR=1, the async. read access executed as page read. page size is according to BL field config., RCSN,RBEN,OEN and RADVN are being ignored. APR is cleared by a hardware reset for EIM_CS1GCR1 - EIM_CS5GCR1. **NOTE:** SRD=0 and MUM=0 must apply when APR=1
14–12 PAT	Page Access Time. This bit field is used in Asynchronous Page Read mode only (APR=1). the initial access is set by RWSC as in regular asynchronous mode. the consecutive address assertions width determine by PAT field according to the settings shown below. when APR=0 this field is ignored. PAT is cleared by a hardware reset for EIM_CS1GCR1 - EIM_CS5GCR1. 000 Address width is 2 EIM clock cycles 001 Address width is 3 EIM clock cycles 010 Address width is 4 EIM clock cycles 011 Address width is 5 EIM clock cycles 100 Address width is 6 EIM clock cycles 101 Address width is 7 EIM clock cycles 110 Address width is 8 EIM clock cycles 111 Address width is 9 EIM clock cycles
11–10 Reserved	This read-only field is reserved and always has the value 0.
9–8 RL	Read Latency. This bit field indicates cycle latency when executing a synchronous read operation. The fields holds the feedback clock loop delay in aclk cycle units. This field is cleared by a hardware reset. 00 Feedback clock loop delay is up to 1 cycle for BCD = 0 or 1.5 cycles for BCD != 0 01 Feedback clock loop delay is up to 2 cycles for BCD = 0 or 2.5 cycles for BCD != 0

Table continues on the next page...

EIM_CS*n*RCR2 field descriptions (continued)

Field	Description
	10 Feedback clock loop delay is up to 3 cycles for BCD = 0 or 3.5 cycles for BCD != 0 11 Feedback clock loop delay is up to 4 cycles for BCD = 0 or 4.5 cycles for BCD != 0
7 Reserved	This read-only field is reserved and always has the value 0.
6–4 RBEA	Read BE Assertion. This bit field determines when BE signal is asserted during read cycles (synchronous or asynchronous mode), according to the settings shown below. RBEA is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between beginning of read access and BE assertion 001 1 EIM clock cycles between beginning of read access and BE assertion 010 2 EIM clock cycles between beginning of read access and BE assertion 111 7 EIM clock cycles between beginning of read access and BE assertion
3 RBE	Read BE enable. This bit field determines if BE will be asserted during read access. 0 - BE are disabled during read access. 1- BE are enable during read access according to value of RBEA & RBEN bit fields.
RBEN	Read BE Negation. This bit field determines when BE signal is negated during read cycles in asynchronous single mode only (SRD=0 & APR=0), according to the settings shown below. This bit field is ignored when SRD=1. RBEN is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between end of read access and BE negation 001 1 EIM clock cycles between end of read access and BE negation 010 2 EIM clock cycles between end of read access and BE negation 111 7 EIM clock cycles between end of read access and BE negation

22.9.5 Chip Select n Write Configuration Register 1 (EIM_CS*n*WCR1)

Address: 21B_8000h base + 10h offset + (24d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	WAL	WBED	WWSC						WADVA			WADVN			WBEA	
Reset	0	0							0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	WBEA	WBEN			WEA				WEN			WCSA			WCSN	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EIM_CS*n*WCR1 field descriptions

Field	Description
31 WAL	Write ADV Low. This bit field determine ADV signal negation time in write accesses. When WAL=1, WADVN bit field is ignored and ADV signal will stay asserted until end of access. When WAL=0 negation of ADV signal is according to WADVN bit field configuration.
30 WBED	Write Byte Enable Disable. When asserted this bit prevent from IPP_DO_BE_B[x] to be asserted during write accesses.This bit is cleared by hardware reset.
29–24 WWSC	Write Wait State Control. This bit field programs the number of wait-states, according to the settings shown below, for synchronous or asynchronous write access to the external device connected to the chip select. When SWR=1 and WFL=0, WWSC indicates the number of burst clock (BCLK) cycles from the start of an access, before the memory can sample the first data.Since WAIT signal can be asserted one cycle before the first data can be sampled, the controller starts evaluating the WAIT signal state one cycle before, this is referred as handshake mode or variable latency mode. When SWR=1 and WFL=1, WWSC indicates the number of burst clock (BCLK) cycles from the start of an access, until the external device is ready for data transfer, this is referred as fix latency mode. When SWR=0, WFL bit is ignored, WWSC indicates the asynchronous access length and the number of EIM clock cycles from the start of access until the external device is ready for data transfer. WWSC is cleared by a hardware reset. **NOTE:** The reset value for EIM_CS0WCR1, WWSC[5:0] = 0b011100. For EIM_CS1WCR1 - EIM_CS5WCR1, the reset value of this field is 0b000000. Example settings: 000000 Reserved 000001 WWSC value is 1 000010 WWSC value is 2 000011 WWSC value is 3 111111 WWSC value is 63
23–21 WADVA	ADV Assertion. This bit field determines when ADV signal is asserted for synchronous or asynchronous write modes according to the settings shown below. WADVA is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between beginning of access and ADV assertion 001 1 EIM clock cycles between beginning of access and ADV assertion 010 2 EIM clock cycles between beginning of access and ADV assertion 111 7 EIM clock cycles between beginning of access and ADV assertion
20–18 WADVN	ADV Negation. This bit field determines when ADV signal to memory is negated during write accesses. When SWR=1 (synchronous write mode), ADV negation occurs according to the following formula: (WADVN + WADVA + BCD + BCS + 1) EIM clock cycles. When asynchronous read mode is applied (SWR=0) ADV negation occurs according to the following formula: (WADVN + WADVA + 1) EIM clock cycles. **NOTE:** Reset value for EIM_CS0WCR for WADVN is 2. For EIM_CS1WCR - EIM_CS5WCR reset value is 000. **NOTE:** This field should be configured so ADV negation will occur before the end of access. For ADV negation at the same time as the end of access, S/W should set the WAL bit.
17–15 WBEA	BE Assertion. This bit field determines when BE signal is asserted during write cycles in async. mode only (SWR=0), according to the settings shown below. BEA is cleared by a hardware reset.

Table continues on the next page...

EIM_CS*n*WCR1 field descriptions (continued)

Field	Description
	NOTE: Reset value for EIM_CS0WCR for WBEA is 2. For EIM_CS1WCR - EIM_CS5WCR reset value is 000. Example settings: 000 0 EIM clock cycles between beginning of access and BE assertion 001 1 EIM clock cycles between beginning of access and BE assertion 010 2 EIM clock cycles between beginning of access and BE assertion 111 7 EIM clock cycles between beginning of access and BE assertion
14–12 WBEN	BE[3:0] Negation. This bit field determines when BE[3:0] bus signal is negated during write cycles in async. mode only (SWR=0), according to the settings shown below. This bit field is ignored when SWR=1. BEN is cleared by a hardware reset. **NOTE:** Reset value for EIM_CS0WCR for WBEN is 2. For EIM_CS1WCR - EIM_CS5WCR reset value is 000. Example settings: 000 0 EIM clock cycles between end of access and WE negation 001 1 EIM clock cycles between end of access and WE negation 010 2 EIM clock cycles between end of access and WE negation 111 7 EIM clock cycles between end of access and WE negation
11–9 WEA	WE Assertion. This bit field determines when WE signal is asserted during write cycles (synchronous or asynchronous mode), according to the settings shown below. This bit field is ignored when executing a read access to the external device. WEA is cleared by a hardware reset. **NOTE:** Reset value for EIM_CS0WCR for WEA is 2. For EIM_CS1WCR - EIM_CS5WCR reset value is 000. Example settings: 000 0 EIM clock cycles between beginning of access and WE assertion 001 1 EIM clock cycles between beginning of access and WE assertion 010 2 EIM clock cycles between beginning of access and WE assertion 111 7 EIMclock cycles between beginning of access and WE assertion
8–6 WEN	WE Negation. This bit field determines when WE signal is negated during write cycles in asynchronous mode only (SWR=0), according to the settings shown below. This bit field is ignored when SWR=1. WEN is cleared by a hardware reset. **NOTE:** Reset value for EIM_CS0WCR for WEN is 2. For EIM_CS1WCR - EIM_CS5WCR reset value is 000. Example settings: 000 0 EIM clock cycles between beginning of access and WE assertion 001 1 EIM clock cycles between beginning of access and WE assertion 010 2 EIM clock cycles between beginning of access and WE assertion 111 7 EIM clock cycles between beginning of access and WE assertion
5–3 WCSA	Write CS Assertion. This bit field determines when CS signal is asserted during write cycles (synchronous or asynchronous mode), according to the settings shown below.this bit field is ignored when executing a read access to the external device. WCSA is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between beginning of write access and CS assertion

Table continues on the next page...

EIM_CS*n*WCR1 field descriptions (continued)

Field	Description
	001 1 EIM clock cycles between beginning of write access and CS assertion 010 2 EIM clock cycles between beginning of write access and CS assertion 111 7 EIMclock cycles between beginning of write access and CS assertion
WCSN	Write CS Negation. This bit field determines when CS signal is negated during write cycles in asynchronous mode only (SWR=0), according to the settings shown below. This bit field is ignored when SWR=1. WCSN is cleared by a hardware reset. Example settings: 000 0 EIM clock cycles between end of read access and CS negation 001 1 EIM clock cycles between end of read access and CS negation 010 2 EIM clock cycles between end of read access and CS negation 111 7 EIM clock cycles between end of read access and CS negation

22.9.6 Chip Select n Write Configuration Register 2 (EIM_CS*n*WCR2)

Address: 21B_8000h base + 14h offset + (24d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

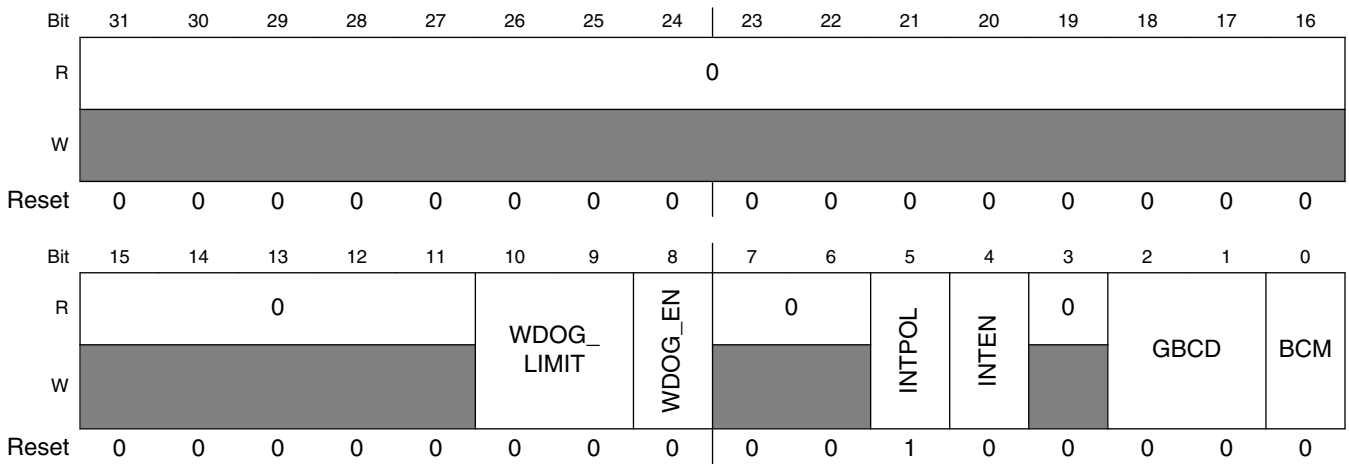
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															WBCDD
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EIM_CS*n*WCR2 field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 WBCDD	Write Burst Clock Divisor Decrement. If this bit is asserted and BCD value is 0 sync. write access will be preformed as if BCD value is 1.When this bit is negated or BCD value is not 0 this bit has no affect. This bit is cleared by hardware reset.

22.9.7 EIM Configuration Register (EIM_WCR)

Address: 21B_8000h base + 90h offset = 21B_8090h



EIM_WCR field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–9 WDOG_LIMIT	Memory Watch Dog (WDog) cycle limit. This bit field determines the number of BCLK cycles (ACLK cycles in dtack mode) before the wdog counter terminates the access and send an error response to the master. 00 128 BCLK cycles 01 256 BCLK cycles 10 512 BCLK cycles 11 1024 BCLK cycles
8 WDOG_EN	Memory WDog enable. This bit controls the operation of the wdog counter that terminates the EIM access. 0 Memory WDog is Disabled 1 Memory WDog is Enabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5 INTPOL	Interrupt Polarity. This bit field determines the polarity of the external device interrupt. 0 External interrupt polarity is active low 1 External interrupt polarity is active high
4 INTEN	Interrupt Enable. When this bit is set the External signal RDY_INT as active interrupt. When interrupt occurs, INT bit at the WCR will be set and t EIM_EXT_INT signal will be asserted correspondingly. This bit is cleared by a hardware reset. 0 External interrupt Disable 1 External interrupt Enable

Table continues on the next page...

EIM_WCR field descriptions (continued)

Field	Description
3 Reserved	This read-only field is reserved and always has the value 0.
2–1 GBCD	General Burst Clock Divisor. When BCM bit is set, this bit field contains the value used to program the burst clock divisor for Continuous BCLK generation. The other BCD bit fields for each chip select are ignored. It is used to divide the internal AXI bus frequency. When BCM=0 GBCD bit field has no influence. GBCD is cleared by a hardware reset. 00 Divide EIM clock by 1 01 Divide EIM clock by 2 10 Divide EIM clock by 3 11 Divide EIM clock by 4
0 BCM	Burst Clock Mode. This bit selects the burst clock mode of operation. It is used for system debug mode. BCM is cleared by a hardware reset. **NOTE:** The BCLK frequency in this mode is according to GBCD bit field. **NOTE:** The BCLK phase is opposite to the EIM clock in this mode if GBCD is 0. **NOTE:** This bit should be used only in async. accesses. No sync access can be executed if this bit is set. **NOTE:** When this bit is set bcd field shouldn't be configured to 0. 0 The burst clock runs only when accessing a chip select range with the SWR/SRD bits set. When the burst clock is not running it remains in a logic 0 state. When the burst clock is running it is configured by the BCD and BCS bit fields in the chip select Configuration Register. 1 The burst clock runs whenever ACLK is active (independent of chip select configuration)

22.9.8 EIM IP Access Register (EIM_WIAR)

Address: 21B_8000h base + 94h offset = 21B_8094h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											ACLK_EN	ERRST	INT	IPS_ACK	IPS_REQ
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

EIM_WIAR field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 ACLK_EN	ACLK enable. This bit gates the ACLK for the EIM except from FFs that get ipg_aclk_s. After reset ACLK is enabled. 0 ACLK is disabled 1 ACLK is enabled
3 ERRST	READY After Reset. This bit controls the initial ready/busy status for external devices on CS0 immediately after hardware reset. This is a sticky bit which is cleared once the RDY_INT signal is asserted by the external device. When ERRST = 1 the first fetch access from EIM to the external device located on CS0 will be pending until RDY_INT signal indicates that the external device is ready, then EIM will execute the access. 0 RDY_INT After Reset Disable 1 RDY_INT After Reset Enable
2 INT	Interrupt. This bit indicates interrupt assertion by an external device according to RDY_INT signal. When polling this bit, INT=0 indicates interrupt not occurred and INT=1 indicates assertion of the external device interrupt. This bit is cleared by a hardware reset.
1 IPS_ACK	IPS ACK. The EIM is ready for ips access. There is no active AXI access and no new AXI access is accepted till this bit is cleared. This bit is cleared by the master after it completes the ips accesses. 0 Master cannot access ips. 1 Master can access ips.
0 IPS_REQ	IPS request. The Master requests to access one of the IPS registers. During such access the EIM should not perform any AXI/memory accesses. The EIM finishes the AXI accesses that already starts and asserts the IPS_ACK bit. 0 No Master requests ips access 1 Master requests ips access

22.9.9 Error Address Register (EIM_EAR)

Address: 21B_8000h base + 98h offset = 21B_8098h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Error_ADDR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EIM_EAR field descriptions

Field	Description
Error_ADDR	Error Address. This bit field holds the AXI address of the last access that caused error. This register is read only register.

Chapter 23
10/100/1000-Mbps Ethernet MAC (ENET)

23.1 Introduction

The MAC-NET core, in conjunction with a 10/100/1000 MAC, implements layer 3 network acceleration functions. These functions are designed to accelerate the processing of various common networking protocols, such as IP, TCP, UDP, and ICMP, providing wire speed services to client applications.

23.2 Overview

The core implements a triple speed 10/100/1000 Mbit/s Ethernet MAC compliant with the IEEE802.3-2002 standard. The MAC layer provides compatibility with half- or full-duplex 10/100 Mbit/s Ethernet LANs and full-duplex gigabit Ethernet LANs.

The MAC operation is fully programmable and can be used in Network Interface Card (NIC), bridging, or switching applications. The core implements the remote network monitoring (RMON) counters according to IETF RFC 2819.

The core also implements a hardware acceleration block to optimize the performance of network controllers providing TCP/IP, UDP, and ICMP protocol services. The acceleration block performs critical functions in hardware, which are typically implemented with large software overhead.

The core implements programmable embedded FIFOs that can provide buffering on the receive path for lossless flow control.

Advanced power management features are available with magic packet detection and programmable power-down modes.

The programmable 10/100/1000 Ethernet MAC with IEEE 1588 integrates a standard IEEE 802.3 Ethernet MAC with a time-stamping module. The IEEE 1588 standard provides accurate clock synchronization for distributed control nodes for industrial automation applications.

23.2.1 Features

The MAC-NET core includes the following features.

23.2.1.1 Ethernet MAC features

- Implements the full 802.3 specification with preamble/SFD generation, frame padding generation, CRC generation and checking
- Supports zero-length preamble
- Dynamically configurable to support 10/100 Mbit/s and gigabit operation
- Supports 10/100 Mbit/s full-duplex and configurable half-duplex operation
- Supports gigabit full-duplex operation
- Compliant with the AMD magic packet detection with interrupt for node remote power management
- Seamless interface to commercial ethernet PHY devices via one of the following:
  - a 4-bit Media Independent Interface (MII) operating at 2.5/25 MHz.
  - a 4-bit non-standard MII-Lite (MII without the CRS and COL signals) operating at 2.5/25 MHz.
  - a 2-bit Reduced MII (RMII) operating at 50 MHz.
  - a (double data rate) 4-bit Reduced GMII (RGMII) operating at 125 MHz.
- Simple 64-Bit FIFO user-application interface
- CRC-32 checking at full speed with optional forwarding of the frame check sequence (FCS) field to the client
- CRC-32 generation and append on transmit or forwarding of user application provided FCS selectable on a per-frame basis
- In full-duplex mode:
  - Implements automated pause frame (802.3 x31A) generation and termination, providing flow control without user application intervention
  - Pause quanta used to form pause frames — dynamically programmable
  - Pause frame generation additionally controllable by user application offering flexible traffic flow control
  - Optional forwarding of received pause frames to the user application
  - Implements standard flow-control mechanism
- In half-duplex mode: provides full collision support, including jamming, backoff, and automatic retransmission

- Supports VLAN-tagged frames according to IEEE 802.1Q
- Programmable MAC address: Insertion on transmit; discards frames with mismatching destination address on receive (except broadcast and pause frames)
- Programmable promiscuous mode support to omit MAC destination address checking on receive
- Multicast and unicast address filtering on receive based on 64-entry hash table, reducing higher layer processing load
- Programmable frame maximum length providing support for any standard or proprietary frame length
- Statistics indicators for frame traffic and errors (alignment, CRC, length) and pause frames providing for IEEE 802.3 basic and mandatory management information database (MIB) package and remote network monitoring (RFC 2819)
- Simple handshake user application FIFO interface with fully programmable depth and threshold levels
- Provides separate status word for each received frame on the user interface providing information such as frame length, frame type, VLAN tag, and error information
- Multiple internal loopback options
- MDIO master interface for PHY device configuration and management with two programmable MDIO base addresses
- Supports legacy FEC buffer descriptors

23.2.1.2 IP protocol performance optimization features

- Operates on TCP/IP and UDP/IP and ICMP/IP protocol data or IP header only
- Enables wire-speed processing
- Supports IPv4 and IPv6
- Transparent passing of frames of other types and protocols
- Supports VLAN tagged frames according to IEEE 802.1q with transparent forwarding of VLAN tag and control field
- Automatic IP-header and payload (protocol specific) checksum calculation and verification on receive
- Automatic IP-header and payload (protocol specific) checksum generation and automatic insertion on transmit configurable on a per-frame basis
- Supports IP and TCP, UDP, ICMP data for checksum generation and checking
- Supports full header options for IPv4 and TCP protocol headers

- Provides IPv6 support to datagrams with base header only — datagrams with extension headers are passed transparently unmodifed/unchecked
- Provides statistics information for received IP and protocol errors
- Configurable automatic discard of erroneous frames
- Configurable automatic host-to-network (RX) and network-to-host (TX) byte order conversion for IP and TCP/UDP/ICMP headers within the frame
- Configurable padding remove for short IP datagrams on receive
- Configurable Ethernet payload alignment to allow for 32-bit word-aligned header and payload processing
- Programmable store-and-forward operation with clock and rate decoupling FIFOs

23.2.1.3 IEEE 1588 features

- Supports all IEEE 1588 frames
- Allows reference clock to be chosen independently of network speed
- Software-programmable precise time-stamping of ingress and egress frames
- Timer monitoring capabilities for system calibration and timing accuracy management
- Precise time-stamping of external events with programmable interrupt generation
- Programmable event and interrupt generation for external system control
- Supports hardware- and software-controllable timer synchronization
- Provides a 4-channel IEEE 1588 timer — each channel supports input capture and output compare using the 1588 counter

23.2.2 Block diagram

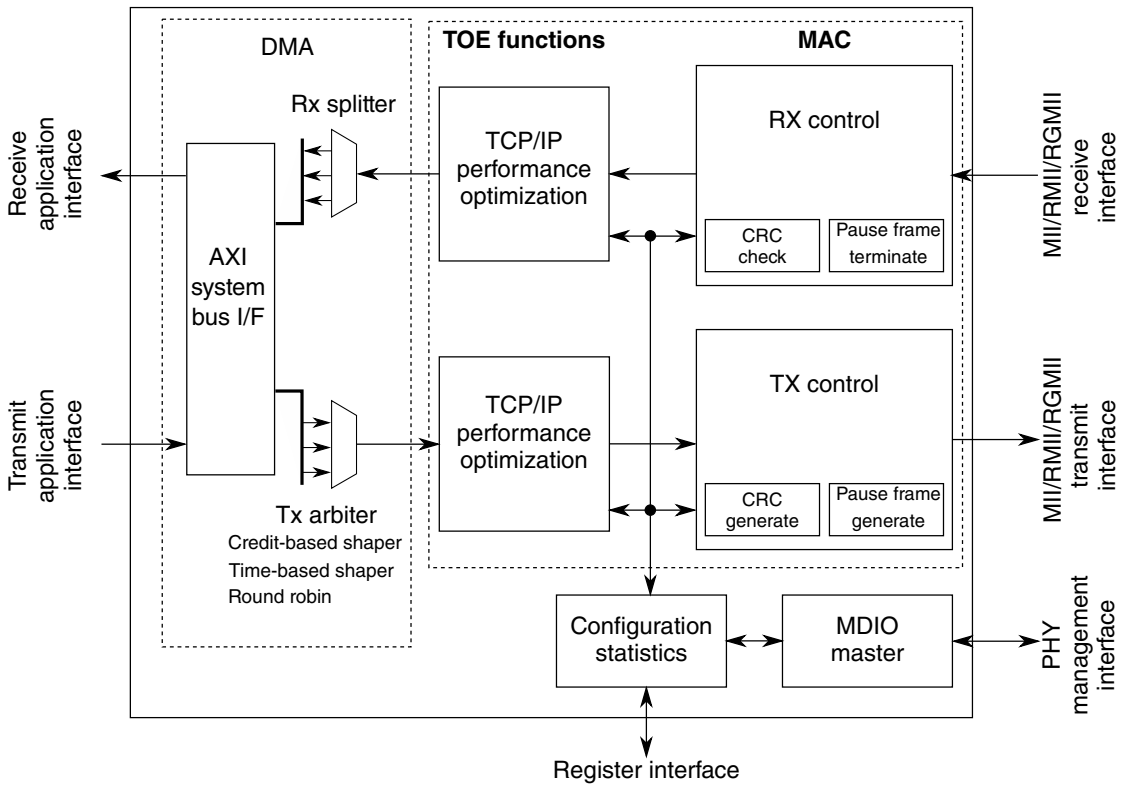


Figure 23-1. Ethernet MAC-NET core block diagram

23.3 External Signals

The table found here describes the external signals of ENET.

Table 23-1. ENET External Signals

Signal	Description	Mode	Pad	Alt Mode	Direction
ENET_1588_EVENT0_IN	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the	MII/RMII/RGMII	ENET_TXD1	ALT4	I

Table continues on the next page...

Table 23-1. ENET External Signals (continued)

Signal	Description	Mode	Pad	Alt Mode	Direction
	corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.				
ENET_1588_EVENT0_OUT	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.	MII/RMII/ RGMII	GPIO_19	ALT1	O
ENET_1588_EVENT1_IN	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.	MII/RMII/ RGMII	ENET_MDC	ALT4	I
ENET_1588_EVENT1_OUT	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.	MII/RMII/ RGMII	ENET_MDIO	ALT4	O
ENET_1588_EVENT2_IN	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected,	MII/RMII/ RGMII	GPIO_16	ALT1	I

Table continues on the next page...

Table 23-1. ENET External Signals (continued)

Signal	Description	Mode	Pad	Alt Mode	Direction
	the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.				
ENET_1588_EVENT2_OUT	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.	MII/RMII/ RGMII	ENET_RX_ER	ALT4	O
ENET_1588_EVENT3_IN	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software. When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.	MII/RMII/ RGMII	GPIO_17	ALT1	I
ENET_1588_EVENT3_OUT	Capture/compare block input/output event bus signal. When configured for capture and a rising edge is detected, the current timer value is latched and transferred into the corresponding ENET_TCCRn register for inspection by software.	MII/RMII/ RGMII	ENET_RXD1	ALT4	O

Table continues on the next page...

Table 23-1. ENET External Signals (continued)

Signal	Description	Mode	Pad	Alt Mode	Direction
	When configured for compare, the corresponding signal 1588_EVENT is asserted for one cycle when the timer reaches the compare value programmed in register ENET_TCCRn. An interrupt or DMA request can be triggered if the corresponding bit in ENET_TCSRn[TIE] or ENET_TCSRn[TDRE] is set.				
ENET_COL	Asserted upon detection of a collision and remains asserted while the collision persists. This signal is not defined for full duplex mode.	MII	KEY_ROW1	ALT1	I
ENET_CRS	Carrier sense. When asserted, indicates transmit or receive medium is not idle. In RMII mode, this signal is present on the RMII_CRS_DV pin.	MII	KEY_COL3	ALT1	I
ENET_MDC	Output clock provides a timing reference to the PHY for data transfers on the MDIO signal.	MII/RMII/ RGMII	ENET_MDC	ALT1	O
			KEY_COL2	ALT4	
ENET_MDIO	Transfers control information between the external PHY and the media-access controller. Data is synchronous to MDC. This signal is an input after reset.	MII/RMII/ RGMII	ENET_MDIO	ALT1	IO
			KEY_COL1	ALT1	
ENET_REF_CLK	In RMII mode, this signal is the reference clock for receive, transmit, and the control interface.	RMII	GPIO_16	ALT2	I
			RGMII_TX_CTL	ALT7	
ENET_RX_CLK	In MII mode, provides a timing reference for RX_EN, RX_DATA[3:0], and RX_ER.	MII	GPIO_18	ALT1	I
ENET_RX_DATA0	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	MII/RMII	ENET_RXD0	ALT1	I
ENET_RX_DATA1	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	MII/RMII	ENET_RXD1	ALT1	I
ENET_RX_DATA2	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	MII	KEY_COL2	ALT1	I
ENET_RX_DATA3	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	MII	KEY_COL0	ALT1	I

Table continues on the next page...

Table 23-1. ENET External Signals (continued)

Signal	Description	Mode	Pad	Alt Mode	Direction
ENET_RX_EN	Asserting this input indicates the PHY has valid nibbles present on the MII. RX_EN must remain asserted from the first recovered nibble of the frame through to the last nibble. Asserting RX_EN must start no later than the SFD and exclude any EOF. In RMII mode, this pin also generates the CRS signal.	MII/RMII	ENET_CRS_DV	ALT1	I
ENET_RX_ER	When asserted with RXDV, indicates the PHY detects an error in the current frame.	MII/RMII	ENET_RX_ER	ALT1	I
ENET_TX_CLK	Input clock, which provides a timing reference for TX_EN, TX_DATA[3:0], and TX_ER.	MII	ENET_REF_CLK	ALT1	I
ENET_TX_DATA0	Serial output Ethernet data. Only valid during TX_EN assertion.	MII/RMII	ENET_TXD0	ALT1	O
ENET_TX_DATA1	Serial output Ethernet data. Only valid during TX_EN assertion.	MII/RMII	ENET_TXD1	ALT1	O
ENET_TX_DATA2	Serial output Ethernet data. Only valid during TX_EN assertion.	MII	KEY_ROW2	ALT1	O
ENET_TX_DATA3	Serial output Ethernet data. Only valid during TX_EN assertion.	MII	KEY_ROW0	ALT1	O
ENET_TX_EN	Indicates when valid nibbles are present on the MII. This signal is asserted with the first nibble of a preamble and is deasserted before the first TX_CLK following the final nibble of the frame.	MII/RMII	ENET_TX_EN	ALT1	O
ENET_TX_ER	When asserted for one or more clock cycles while TXEN is also asserted, PHY sends one or more illegal symbols.	MII/RMII	GPIO_19	ALT6	O
RGMII_RD0	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	RGMII	RGMII_RD0	ALT1	I
RGMII_RD1	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	RGMII	RGMII_RD1	ALT1	I
RGMII_RD2	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	RGMII	RGMII_RD2	ALT1	I
RGMII_RD3	Contains the Ethernet input data transferred from the PHY to the media-access controller when RX_EN is asserted.	RGMII	RGMII_RD3	ALT1	I

Table continues on the next page...

Table 23-1. ENET External Signals (continued)

Signal	Description	Mode	Pad	Alt Mode	Direction
RGMII_RXC	In RGMII mode, provides a timing reference for RX_DATA[3:0] and RX_CTL.	RGMII	RGMII_RXC	ALT1	I
RGMII_RX_CTL	In RGMII mode, contains RX_EN on the rising edge of RGMII_RXC, and a logical derivative of RX_EN and RX_ER (RX_EN XOR RX_ER) on the falling edge of RGMII_RXC.	RGMII	RGMII_RX_CTL	ALT1	I
RGMII_TD0	Serial output Ethernet data. Only valid during TX_EN assertion.	RGMII	RGMII_TD0	ALT1	O
RGMII_TD1	Serial output Ethernet data. Only valid during TX_EN assertion.	RGMII	RGMII_TD1	ALT1	O
RGMII_TD2	Serial output Ethernet data. Only valid during TX_EN assertion.	RGMII	RGMII_TD2	ALT1	O
RGMII_TD3	Serial output Ethernet data. Only valid during TX_EN assertion.	RGMII	RGMII_TD3	ALT1	O
RGMII_TXC	In RGMII mode, provides a timing reference for TX_DATA[3:0] and TX_CTL.	RGMII	RGMII_TXC	ALT1	O
RGMII_TX_CTL	In RGMII mode, contains TX_EN on the rising edge of RGMII_TXC, and a logical derivative of TX_EN and TX_ER (TX_EN XOR TX_ER) on the falling edge of RGMII_TXC.	RGMII	RGMII_TX_CTL	ALT1	O

23.4 Clocks

The table found here describes the clock sources for ENET.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 23-2. ENET Clocks

Clock name	Clock Root	Description
ipg_clk	ahb_clk_root	Module clock
ipg_clk_mac0	ahb_clk_root	MAC peripheral clock
ipg_clk_mac0_s	ipg_clk_root	MAC peripheral access clock
ipg_clk_s	ipg_clk_root	Peripheral access clock
ipg_clk_time	ipg_clk_root	Peripheral clock
mac0_rxmem_clk	ahb_clk_root	MAC receive memory clock
mac0_txmem_clk	ahb_clk_root	MAC transmit memory clock

NOTE

The ENET module requires ahb_clk_root to be 125 MHz or greater.

23.5 Memory map/register definition

Reserved bits should be written with 0 and ignored on read to allow future extension. Unused registers read zero and a write has no effect.

The table found here summarizes the Ethernet registers.

Table 23-3. Register map summary

Offset Address	Section	Description
0x0000 – 0x01FF	Configuration	Core control and status registers
0x0200 – 0x03FF	Statistics counters	MIB and Remote Network Monitoring (RFC 2819) registers
0x0400 – 0x0430	1588 control	1588 adjustable timer (TSM) and 1588 frame control
0x0600 – 0x07FC	Capture/compare block	Registers for the capture/compare block

ENET memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_8004	Interrupt Event Register (ENET_EIR)	32	w1c	0000_0000h	23.5.1/1089
218_8008	Interrupt Mask Register (ENET_EIMR)	32	R/W	0000_0000h	23.5.2/1092
218_8010	Receive Descriptor Active Register (ENET_RDAR)	32	R/W	0000_0000h	23.5.3/1095
218_8014	Transmit Descriptor Active Register (ENET_TDAR)	32	R/W	0000_0000h	23.5.4/1096
218_8024	Ethernet Control Register (ENET_ECR)	32	R/W	F000_0000h	23.5.5/1097
218_8040	MII Management Frame Register (ENET_MMFR)	32	R/W	0000_0000h	23.5.6/1099
218_8044	MII Speed Control Register (ENET_MSCR)	32	R/W	0000_0000h	23.5.7/1100
218_8064	MIB Control Register (ENET_MIBC)	32	R/W	C000_0000h	23.5.8/1102
218_8084	Receive Control Register (ENET_RCR)	32	R/W	05EE_0001h	23.5.9/1104
218_80C4	Transmit Control Register (ENET_TCR)	32	R/W	0000_0000h	23.5.10/ 1107
218_80E4	Physical Address Lower Register (ENET_PALR)	32	R/W	0000_0000h	23.5.11/ 1109
218_80E8	Physical Address Upper Register (ENET_PAUR)	32	R/W	0000_8808h	23.5.12/ 1109
218_80EC	Opcode/Pause Duration Register (ENET_OPD)	32	R/W	0001_0000h	23.5.13/ 1110
218_8118	Descriptor Individual Upper Address Register (ENET_IAUR)	32	R/W	0000_0000h	23.5.14/ 1110

Table continues on the next page...

ENET memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_811C	Descriptor Individual Lower Address Register (ENET_IALR)	32	R/W	0000_0000h	23.5.15/ 1111
218_8120	Descriptor Group Upper Address Register (ENET_GAUR)	32	R/W	0000_0000h	23.5.16/ 1111
218_8124	Descriptor Group Lower Address Register (ENET_GALR)	32	R/W	0000_0000h	23.5.17/ 1112
218_8144	Transmit FIFO Watermark Register (ENET_TFWR)	32	R/W	0000_0000h	23.5.18/ 1112
218_8180	Receive Descriptor Ring Start Register (ENET_RDSR)	32	R/W	0000_0000h	23.5.19/ 1113
218_8184	Transmit Buffer Descriptor Ring Start Register (ENET_TDSR)	32	R/W	0000_0000h	23.5.20/ 1114
218_8188	Maximum Receive Buffer Size Register (ENET_MRBR)	32	R/W	0000_0000h	23.5.21/ 1115
218_8190	Receive FIFO Section Full Threshold (ENET_RSFL)	32	R/W	0000_0000h	23.5.22/ 1116
218_8194	Receive FIFO Section Empty Threshold (ENET_RSEM)	32	R/W	0000_0000h	23.5.23/ 1116
218_8198	Receive FIFO Almost Empty Threshold (ENET_RAEM)	32	R/W	0000_0004h	23.5.24/ 1117
218_819C	Receive FIFO Almost Full Threshold (ENET_RAFL)	32	R/W	0000_0004h	23.5.25/ 1117
218_81A0	Transmit FIFO Section Empty Threshold (ENET_TSEM)	32	R/W	0000_0000h	23.5.26/ 1118
218_81A4	Transmit FIFO Almost Empty Threshold (ENET_TAEM)	32	R/W	0000_0004h	23.5.27/ 1118
218_81A8	Transmit FIFO Almost Full Threshold (ENET_TAFL)	32	R/W	0000_0008h	23.5.28/ 1119
218_81AC	Transmit Inter-Packet Gap (ENET_TIPG)	32	R/W	0000_000Ch	23.5.29/ 1119
218_81B0	Frame Truncation Length (ENET_FTRL)	32	R/W	0000_07FFh	23.5.30/ 1120
218_81C0	Transmit Accelerator Function Configuration (ENET_TACC)	32	R/W	0000_0000h	23.5.31/ 1120
218_81C4	Receive Accelerator Function Configuration (ENET_RACC)	32	R/W	0000_0000h	23.5.32/ 1121
218_8204	Tx Packet Count Statistic Register (ENET_RMON_T_PACKETS)	32	R	0000_0000h	23.5.33/ 1122
218_8208	Tx Broadcast Packets Statistic Register (ENET_RMON_T_BC_PKT)	32	R	0000_0000h	23.5.34/ 1123
218_820C	Tx Multicast Packets Statistic Register (ENET_RMON_T_MC_PKT)	32	R	0000_0000h	23.5.35/ 1123
218_8210	Tx Packets with CRC/Align Error Statistic Register (ENET_RMON_T_CRC_ALIGN)	32	R	0000_0000h	23.5.36/ 1124

Table continues on the next page...

ENET memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_8214	Tx Packets Less Than Bytes and Good CRC Statistic Register (ENET_RMON_T_UNDERSIZE)	32	R	0000_0000h	23.5.37/ 1124
218_8218	Tx Packets GT MAX_FL bytes and Good CRC Statistic Register (ENET_RMON_T_OVERSIZE)	32	R	0000_0000h	23.5.38/ 1125
218_821C	Tx Packets Less Than 64 Bytes and Bad CRC Statistic Register (ENET_RMON_T_FRAG)	32	R	0000_0000h	23.5.39/ 1125
218_8220	Tx Packets Greater Than MAX_FL bytes and Bad CRC Statistic Register (ENET_RMON_T_JAB)	32	R	0000_0000h	23.5.40/ 1126
218_8224	Tx Collision Count Statistic Register (ENET_RMON_T_COL)	32	R	0000_0000h	23.5.41/ 1126
218_8228	Tx 64-Byte Packets Statistic Register (ENET_RMON_T_P64)	32	R	0000_0000h	23.5.42/ 1127
218_822C	Tx 65- to 127-byte Packets Statistic Register (ENET_RMON_T_P65TO127)	32	R	0000_0000h	23.5.43/ 1127
218_8230	Tx 128- to 255-byte Packets Statistic Register (ENET_RMON_T_P128TO255)	32	R	0000_0000h	23.5.44/ 1128
218_8234	Tx 256- to 511-byte Packets Statistic Register (ENET_RMON_T_P256TO511)	32	R	0000_0000h	23.5.45/ 1128
218_8238	Tx 512- to 1023-byte Packets Statistic Register (ENET_RMON_T_P512TO1023)	32	R	0000_0000h	23.5.46/ 1129
218_823C	Tx 1024- to 2047-byte Packets Statistic Register (ENET_RMON_T_P1024TO2047)	32	R	0000_0000h	23.5.47/ 1129
218_8240	Tx Packets Greater Than 2048 Bytes Statistic Register (ENET_RMON_T_P_GTE2048)	32	R	0000_0000h	23.5.48/ 1130
218_8244	Tx Octets Statistic Register (ENET_RMON_T_OCTETS)	32	R	0000_0000h	23.5.49/ 1130
218_824C	Frames Transmitted OK Statistic Register (ENET_IEEE_T_FRAME_OK)	32	R	0000_0000h	23.5.50/ 1130
218_8250	Frames Transmitted with Single Collision Statistic Register (ENET_IEEE_T_1COL)	32	R	0000_0000h	23.5.51/ 1131
218_8254	Frames Transmitted with Multiple Collisions Statistic Register (ENET_IEEE_T_MCOL)	32	R	0000_0000h	23.5.52/ 1131
218_8258	Frames Transmitted after Deferral Delay Statistic Register (ENET_IEEE_T_DEF)	32	R	0000_0000h	23.5.53/ 1132
218_825C	Frames Transmitted with Late Collision Statistic Register (ENET_IEEE_T_LCOL)	32	R	0000_0000h	23.5.54/ 1132
218_8260	Frames Transmitted with Excessive Collisions Statistic Register (ENET_IEEE_T_EXCOL)	32	R	0000_0000h	23.5.55/ 1133
218_8264	Frames Transmitted with Tx FIFO Underrun Statistic Register (ENET_IEEE_T_MACERR)	32	R	0000_0000h	23.5.56/ 1133
218_8268	Frames Transmitted with Carrier Sense Error Statistic Register (ENET_IEEE_T_CSERR)	32	R	0000_0000h	23.5.57/ 1134
218_8270	Flow Control Pause Frames Transmitted Statistic Register (ENET_IEEE_T_FDXFC)	32	R	0000_0000h	23.5.58/ 1134

Table continues on the next page...

ENET memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_8274	Octet Count for Frames Transmitted w/o Error Statistic Register (ENET_IEEE_T_OCTETS_OK)	32	R	0000_0000h	23.5.59/ 1135
218_8284	Rx Packet Count Statistic Register (ENET_RMON_R_PACKETS)	32	R	0000_0000h	23.5.60/ 1135
218_8288	Rx Broadcast Packets Statistic Register (ENET_RMON_R_BC_PKT)	32	R	0000_0000h	23.5.61/ 1136
218_828C	Rx Multicast Packets Statistic Register (ENET_RMON_R_MC_PKT)	32	R	0000_0000h	23.5.62/ 1136
218_8290	Rx Packets with CRC/Align Error Statistic Register (ENET_RMON_R_CRC_ALIGN)	32	R	0000_0000h	23.5.63/ 1137
218_8294	Rx Packets with Less Than 64 Bytes and Good CRC Statistic Register (ENET_RMON_R_UNDERSIZE)	32	R	0000_0000h	23.5.64/ 1137
218_8298	Rx Packets Greater Than MAX_FL and Good CRC Statistic Register (ENET_RMON_R_OVERSIZE)	32	R	0000_0000h	23.5.65/ 1138
218_829C	Rx Packets Less Than 64 Bytes and Bad CRC Statistic Register (ENET_RMON_R_FRAG)	32	R	0000_0000h	23.5.66/ 1138
218_82A0	Rx Packets Greater Than MAX_FL Bytes and Bad CRC Statistic Register (ENET_RMON_R_JAB)	32	R	0000_0000h	23.5.67/ 1139
218_82A8	Rx 64-Byte Packets Statistic Register (ENET_RMON_R_P64)	32	R	0000_0000h	23.5.68/ 1139
218_82AC	Rx 65- to 127-Byte Packets Statistic Register (ENET_RMON_R_P65TO127)	32	R	0000_0000h	23.5.69/ 1140
218_82B0	Rx 128- to 255-Byte Packets Statistic Register (ENET_RMON_R_P128TO255)	32	R	0000_0000h	23.5.70/ 1140
218_82B4	Rx 256- to 511-Byte Packets Statistic Register (ENET_RMON_R_P256TO511)	32	R	0000_0000h	23.5.71/ 1141
218_82B8	Rx 512- to 1023-Byte Packets Statistic Register (ENET_RMON_R_P512TO1023)	32	R	0000_0000h	23.5.72/ 1141
218_82BC	Rx 1024- to 2047-Byte Packets Statistic Register (ENET_RMON_R_P1024TO2047)	32	R	0000_0000h	23.5.73/ 1142
218_82C0	Rx Packets Greater than 2048 Bytes Statistic Register (ENET_RMON_R_P_GTE2048)	32	R	0000_0000h	23.5.74/ 1142
218_82C4	Rx Octets Statistic Register (ENET_RMON_R_OCTETS)	32	R	0000_0000h	23.5.75/ 1142
218_82C8	Frames not Counted Correctly Statistic Register (ENET_IEEE_R_DROP)	32	R	0000_0000h	23.5.76/ 1143
218_82CC	Frames Received OK Statistic Register (ENET_IEEE_R_FRAME_OK)	32	R	0000_0000h	23.5.77/ 1143
218_82D0	Frames Received with CRC Error Statistic Register (ENET_IEEE_R_CRC)	32	R	0000_0000h	23.5.78/ 1144
218_82D4	Frames Received with Alignment Error Statistic Register (ENET_IEEE_R_ALIGN)	32	R	0000_0000h	23.5.79/ 1144
218_82D8	Receive FIFO Overflow Count Statistic Register (ENET_IEEE_R_MACERR)	32	R	0000_0000h	23.5.80/ 1145

Table continues on the next page...

ENET memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_82DC	Flow Control Pause Frames Received Statistic Register (ENET_IEEE_R_FDXFC)	32	R	0000_0000h	23.5.81/ 1145
218_82E0	Octet Count for Frames Received without Error Statistic Register (ENET_IEEE_R_OCTETS_OK)	32	R	0000_0000h	23.5.82/ 1146
218_8400	Adjustable Timer Control Register (ENET_ATCR)	32	R/W	0000_0000h	23.5.83/ 1146
218_8404	Timer Value Register (ENET_ATVR)	32	R/W	0000_0000h	23.5.84/ 1148
218_8408	Timer Offset Register (ENET_ATOFF)	32	R/W	0000_0000h	23.5.85/ 1148
218_840C	Timer Period Register (ENET_ATPER)	32	R/W	3B9A_CA00h	23.5.86/ 1149
218_8410	Timer Correction Register (ENET_ATCOR)	32	R/W	0000_0000h	23.5.87/ 1149
218_8414	Time-Stamping Clock Period Register (ENET_ATINC)	32	R/W	0000_0000h	23.5.88/ 1150
218_8418	Timestamp of Last Transmitted Frame (ENET_ATSTMP)	32	R	0000_0000h	23.5.89/ 1150
218_8604	Timer Global Status Register (ENET_TGSR)	32	R/W	0000_0000h	23.5.90/ 1151
218_8608	Timer Control Status Register (ENET_TCSR0)	32	R/W	0000_0000h	23.5.91/ 1152
218_860C	Timer Compare Capture Register (ENET_TCCR0)	32	R/W	0000_0000h	23.5.92/ 1153
218_8610	Timer Control Status Register (ENET_TCSR1)	32	R/W	0000_0000h	23.5.91/ 1152
218_8614	Timer Compare Capture Register (ENET_TCCR1)	32	R/W	0000_0000h	23.5.92/ 1153
218_8618	Timer Control Status Register (ENET_TCSR2)	32	R/W	0000_0000h	23.5.91/ 1152
218_861C	Timer Compare Capture Register (ENET_TCCR2)	32	R/W	0000_0000h	23.5.92/ 1153
218_8620	Timer Control Status Register (ENET_TCSR3)	32	R/W	0000_0000h	23.5.91/ 1152
218_8624	Timer Compare Capture Register (ENET_TCCR3)	32	R/W	0000_0000h	23.5.92/ 1153

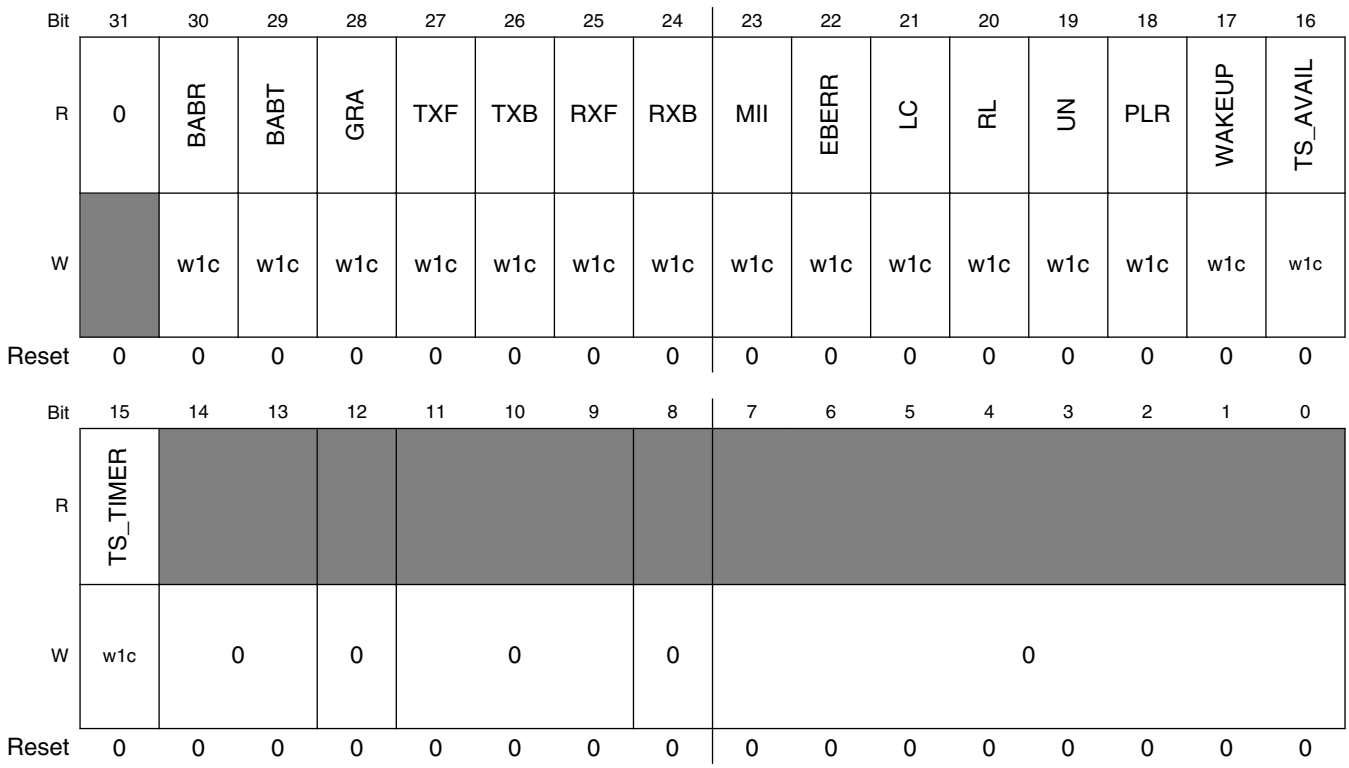
23.5.1 Interrupt Event Register (ENET_EIR)

When an event occurs that sets a bit in EIR, an interrupt occurs if the corresponding bit in the interrupt mask register (EIMR) is also set. Writing a 1 to an EIR bit clears it; writing 0 has no effect. This register is cleared upon hardware reset.

NOTE

TxBD[INT] and RxBD[INT] must be set to 1 to allow setting the corresponding EIR register flags in enhanced mode, ENET_ECR[EN1588] = 1. Legacy mode does not require these flags to be enabled.

Address: 218_8000h base + 4h offset = 218_8004h



ENET_EIR field descriptions

Field	Description
31 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
30 BABR	Babbling Receive Error Indicates a frame was received with length in excess of RCR[MAX_FL] bytes.
29 BABT	Babbling Transmit Error Indicates the transmitted frame length exceeds RCR[MAX_FL] bytes. Usually this condition is caused when a frame that is too long is placed into the transmit data buffer(s). Truncation does not occur.
28 GRA	Graceful Stop Complete This interrupt is asserted after the transmitter is put into a pause state after completion of the frame currently being transmitted. See Graceful Transmit Stop (GTS) for conditions that lead to graceful stop. **NOTE:** The GRA interrupt is asserted only when the TX transitions into the stopped state. If this bit is cleared by writing 1 and the TX is still stopped, the bit is not set again.

Table continues on the next page...

ENET_EIR field descriptions (continued)

Field	Description
27 TXF	Transmit Frame Interrupt Indicates a frame has been transmitted and the last corresponding buffer descriptor has been updated.
26 TXB	Transmit Buffer Interrupt Indicates a transmit buffer descriptor has been updated.
25 RXF	Receive Frame Interrupt Indicates a frame has been received and the last corresponding buffer descriptor has been updated.
24 RXB	Receive Buffer Interrupt Indicates a receive buffer descriptor is not the last in the frame has been updated.
23 MII	MII Interrupt. Indicates that the MII has completed the data transfer requested.
22 EBERR	Ethernet Bus Error Indicates a system bus error occurred when a uDMA transaction is underway. When this bit is set, ECR[ETHEREN] is cleared, halting frame processing by the MAC. When this occurs, software must ensure proper actions, possibly resetting the system, to resume normal operation.
21 LC	Late Collision Indicates a collision occurred beyond the collision window (slot time) in half-duplex mode. The frame truncates with a bad CRC and the remainder of the frame is discarded.
20 RL	Collision Retry Limit Indicates a collision occurred on each of 16 successive attempts to transmit the frame. The frame is discarded without being transmitted and transmission of the next frame commences. This error can only occur in half-duplex mode.
19 UN	Transmit FIFO Underrun Indicates the transmit FIFO became empty before the complete frame was transmitted. A bad CRC is appended to the frame fragment and the remainder of the frame is discarded.
18 PLR	Payload Receive Error Indicates a frame was received with a payload length error. See Frame Length/Type Verification: Payload Length Check for more information.
17 WAKEUP	Node Wakeup Request Indication Read-only status bit to indicate that a magic packet has been detected. Will act only if ECR[MAGICEN] is set.
16 TS_AVAIL	Transmit Timestamp Available Indicates that the timestamp of the last transmitted timing frame is available in the ATSTMP register.
15 TS_TIMER	Timestamp Timer The adjustable timer reached the period event. A period event interrupt can be generated if ATCR[PEREN] is set and the timer wraps according to the periodic setting in the ATPER register. Set the timer period value before setting ATCR[PEREN].
14–13 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.

Table continues on the next page...

ENET_EIR field descriptions (continued)

Field	Description
12 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
11–9 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
8 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.

23.5.2 Interrupt Mask Register (ENET_EIMR)

EIMR controls which interrupt events are allowed to generate actual interrupts. A hardware reset clears this register. If the corresponding bits in the EIR and EIMR registers are set, an interrupt is generated. The interrupt signal remains asserted until a 1 is written to the EIR field (write 1 to clear) or a 0 is written to the EIMR field.

Address: 218_8000h base + 8h offset = 218_8008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R		BABR	BABT	GRA	TXF	TXB	RXF	RXB	MII	EBERR	LC	RL	UN	PLR	WAKEUP	TS_AVAIL
W	0															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	TS_TIMER															
W		0		0		0		0				0				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_EIMR field descriptions

Field	Description
31 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
30 BABR	BABR Interrupt Mask Corresponds to interrupt source EIR[BABR] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR BABR field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.

Table continues on the next page...

ENET_EIMR field descriptions (continued)

Field	Description
	0 The corresponding interrupt source is masked. 1 The corresponding interrupt source is not masked.
29 BABT	BABT Interrupt Mask Corresponds to interrupt source EIR[BABT] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR BABT field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared. 0 The corresponding interrupt source is masked. 1 The corresponding interrupt source is not masked.
28 GRA	GRA Interrupt Mask Corresponds to interrupt source EIR[GRA] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR GRA field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared. 0 The corresponding interrupt source is masked. 1 The corresponding interrupt source is not masked.
27 TXF	TXF Interrupt Mask Corresponds to interrupt source EIR[TXF] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR TXF field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared. 0 The corresponding interrupt source is masked. 1 The corresponding interrupt source is not masked.
26 TXB	TXB Interrupt Mask Corresponds to interrupt source EIR[TXB] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR TXF field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared. 0 The corresponding interrupt source is masked. 1 The corresponding interrupt source is not masked.
25 RXF	RXF Interrupt Mask Corresponds to interrupt source EIR[RXF] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR RXF field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
24 RXB	RXB Interrupt Mask Corresponds to interrupt source EIR[RXB] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR RXB field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
23 MII	MII Interrupt Mask

Table continues on the next page...

ENET_EIMR field descriptions (continued)

Field	Description
	Corresponds to interrupt source EIR[MII] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR MII field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
22 EBERR	EBERR Interrupt Mask Corresponds to interrupt source EIR[EBERR] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR EBERR field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
21 LC	LC Interrupt Mask Corresponds to interrupt source EIR[LC] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR LC field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
20 RL	RL Interrupt Mask Corresponds to interrupt source EIR[RL] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR RL field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
19 UN	UN Interrupt Mask Corresponds to interrupt source EIR[UN] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR UN field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
18 PLR	PLR Interrupt Mask Corresponds to interrupt source EIR[PLR] and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR PLR field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
17 WAKEUP	WAKEUP Interrupt Mask Corresponds to interrupt source EIR[WAKEUP] register and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR WAKEUP field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
16 TS_AVAIL	TS_AVAIL Interrupt Mask Corresponds to interrupt source EIR[TS_AVAIL] register and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR TS_AVAIL field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.
15 TS_TIMER	TS_TIMER Interrupt Mask Corresponds to interrupt source EIR[TS_TIMER] register and determines whether an interrupt condition can generate an interrupt. At every module clock, the EIR samples the signal generated by the interrupting source. The corresponding EIR TS_TIMER field reflects the state of the interrupt signal even if the corresponding EIMR field is cleared.

Table continues on the next page...

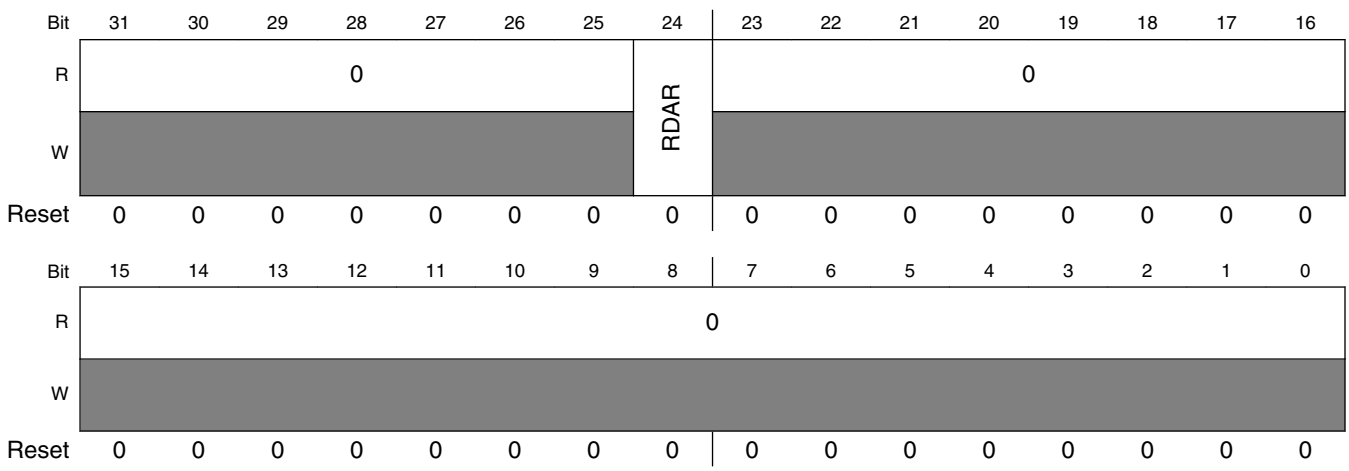
ENET_EIMR field descriptions (continued)

Field	Description
14–13 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
12 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
11–9 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
8 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.

23.5.3 Receive Descriptor Active Register (ENET_RDAR)

RDAR is a command register, written by the user, to indicate that the receive descriptor ring has been updated, that is, that the driver produced empty receive buffers with the empty bit set.

Address: 218_8000h base + 10h offset = 218_8010h



ENET_RDAR field descriptions

Field	Description
31–25 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
24 RDAR	Receive Descriptor Active Always set to 1 when this register is written, regardless of the value written. This field is cleared by the MAC device when no additional empty descriptors remain in the receive ring. It is also cleared when ECR[ETHEREN] transitions from set to cleared or when ECR[RESET] is set.

Table continues on the next page...

ENET_RDAR field descriptions (continued)

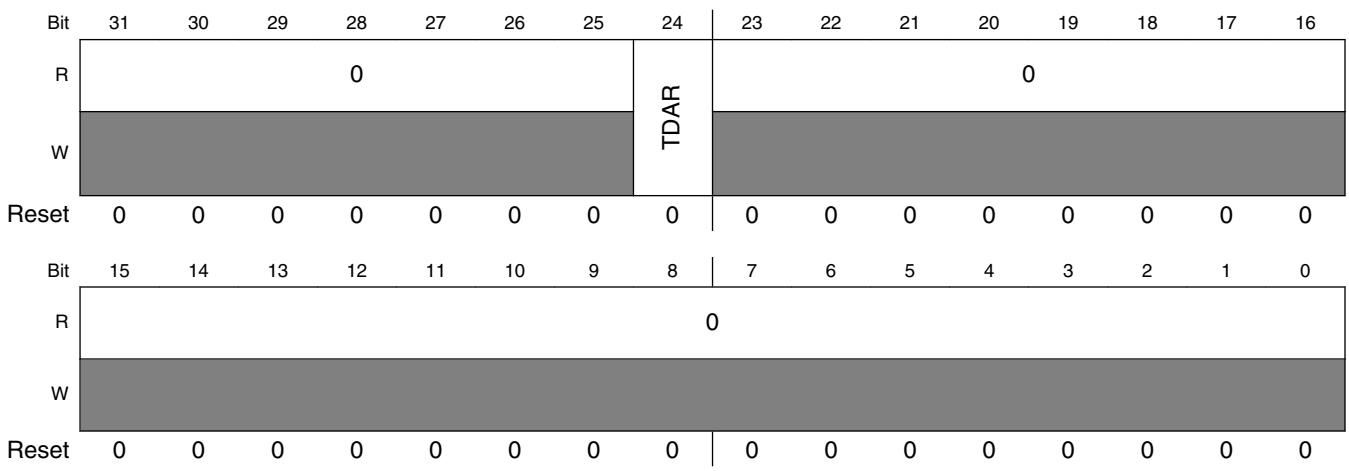
Field	Description
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.4 Transmit Descriptor Active Register (ENET_TDAR)

The TDAR is a command register that the user writes to indicate that the transmit descriptor ring has been updated, that is, that transmit buffers have been produced by the driver with the ready bit set in the buffer descriptor.

The TDAR register is cleared at reset, when ECR[ETHEREN] transitions from set to cleared, or when ECR[RESET] is set.

Address: 218_8000h base + 14h offset = 218_8014h



ENET_TDAR field descriptions

Field	Description
31–25 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
24 TDAR	Transmit Descriptor Active Always set to 1 when this register is written, regardless of the value written. This bit is cleared by the MAC device when no additional ready descriptors remain in the transmit ring. Also cleared when ECR[ETHEREN] transitions from set to cleared or when ECR[RESET] is set.
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.5 Ethernet Control Register (ENET_ECR)

ECR is a read/write user register, though hardware may also alter fields in this register. It controls many of the high level features of the Ethernet MAC, including legacy FEC support through the EN1588 field.

Address: 218_8000h base + 24h offset = 218_8024h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved				0	0	0	DBSWP	STOPEN	DBGEN	SPEED	EN1588	SLEEP	MAGICEN	ETHEREN	RESET
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_ECR field descriptions

Field	Description
31–12 Reserved	This field is reserved. This field must be set to F_0000h.
11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
8 DBSWP	Descriptor Byte Swapping Enable Swaps the byte locations of the buffer descriptors. 0 The buffer descriptor bytes are not swapped to support big-endian devices. 1 The buffer descriptor bytes are swapped to support little-endian devices.
7 STOPEN	STOPEN Signal Control Controls device behavior in doze mode. In doze mode, if this field is set then all the clocks of the ENET assembly are disabled, except the RMII /MII clock. Doze mode is similar to a conditional stop mode entry for the ENET assembly depending on ECR[STOPEN]. **NOTE:** If module clocks are gated in this mode, the module can still wake the system after receiving a magic packet in stop mode. MAGICEN must be set prior to entering sleep/stop mode.

Table continues on the next page...

ENET_ECR field descriptions (continued)

Field	Description
6 DBGEN	Debug Enable Enables the MAC to enter hardware freeze mode when the device enters debug mode. 0 MAC continues operation in debug mode. 1 MAC enters hardware freeze mode when the processor is in debug mode.
5 SPEED	Selects between 10/100 and 1000 Mbps modes of operation. 0 10/100 Mbps mode 1 1000 Mbps mode
4 EN1588	EN1588 Enable Enables enhanced functionality of the MAC. 0 Legacy FEC buffer descriptors and functions enabled. 1 Enhanced frame time-stamping functions enabled.
3 SLEEP	Sleep Mode Enable 0 Normal operating mode. 1 Sleep mode.
2 MAGICEN	Magic Packet Detection Enable Enables/disables magic packet detection. **NOTE:** MAGICEN is relevant only if the SLEEP field is set. If MAGICEN is set, changing the SLEEP field enables/disables sleep mode and magic packet detection. 0 Magic detection logic disabled. 1 The MAC core detects magic packets and asserts EIR[WAKEUP] when a frame is detected.
1 ETHEREN	Ethernet Enable Enables/disables the Ethernet MAC. When the MAC is disabled, the buffer descriptors for an aborted transmit frame are not updated. The uDMA, buffer descriptor, and FIFO control logic are reset, including the buffer descriptor and FIFO pointers. Hardware clears this field under the following conditions: • RESET is set by software • An error condition causes the EBERR field to set. **NOTE:** • ETHEREN must be set at the very last step during ENET configuration/setup/initialization, only *after* all other ENET-related registers have been configured. • If ETHEREN is cleared to 0 by software then next time ETHEREN is set, the EIR interrupts must cleared to 0 due to previous pending interrupts. 0 Reception immediately stops and transmission stops after a bad CRC is appended to any currently transmitted frame. 1 MAC is enabled, and reception and transmission are possible.
0 RESET	Ethernet MAC Reset When this field is set, it clears the ETHEREN field.

23.5.6 MII Management Frame Register (ENET_MMFR)

Writing to MMFR triggers a management frame transaction to the PHY device unless MSCR is programmed to zero.

If MSCR is changed from zero to non-zero during a write to MMFR, an MII frame is generated with the data previously written to the MMFR. This allows MMFR and MSCR to be programmed in either order if MSCR is currently zero.

If the MMFR register is written while frame generation is in progress, the frame contents are altered. Software must use the EIR[MII] interrupt indication to avoid writing to the MMFR register while frame generation is in progress.

Address: 218_8000h base + 40h offset = 218_8040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ST		OP		PA					RA					TA		DATA															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_MMFR field descriptions

Field	Description
31–30 ST	Start Of Frame Delimiter These fields must be programmed to 01 for a valid MII management frame.
29–28 OP	Operation Code Determines the frame operation. 00 Write frame operation, but not MII compliant. 01 Write frame operation for a valid MII management frame. 10 Read frame operation for a valid MII management frame. 11 Read frame operation, but not MII compliant.
27–23 PA	PHY Address Specifies one of up to 32 attached PHY devices.
22–18 RA	Register Address Specifies one of up to 32 registers within the specified PHY device.
17–16 TA	Turn Around This field must be programmed to 10 to generate a valid MII management frame.
DATA	Management Frame Data This is the field for data to be written to or read from the PHY register.

23.5.7 MII Speed Control Register (ENET_MSCR)

MSCR provides control of the MII clock (MDC pin) frequency and allows a preamble drop on the MII management frame.

The MII_SPEED field must be programmed with a value to provide an MDC frequency of less than or equal to 2.5 MHz to be compliant with the IEEE 802.3 MII specification. The MII_SPEED must be set to a non-zero value to source a read or write management frame. After the management frame is complete, the MSCR register may optionally be cleared to turn off MDC. The MDC signal generated has a 50% duty cycle except when MII_SPEED changes during operation. This change takes effect following a rising or falling edge of MDC.

If the internal module clock is 25 MHz, programming MII_SPEED to 0x4 results in an MDC as given in the following equation:

25 MHz / ((4 + 1) x 2) = 2.5 MHz

The following table shows the optimum values for MII_SPEED as a function of internal module clock frequency.

Table 23-12. Programming Examples for MSCR

Internal MAC clock frequency	MSCR [MII_SPEED]	MDC frequency
25 MHz	0x4	2.50 MHz
33 MHz	0x6	2.36 MHz
40 MHz	0x7	2.50 MHz
50 MHz	0x9	2.50 MHz
66 MHz	0xD	2.36 MHz

Address: 218_8000h base + 44h offset = 218_8044h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					HOLDTIME			DIS_ PRE	MII_SPEED						0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

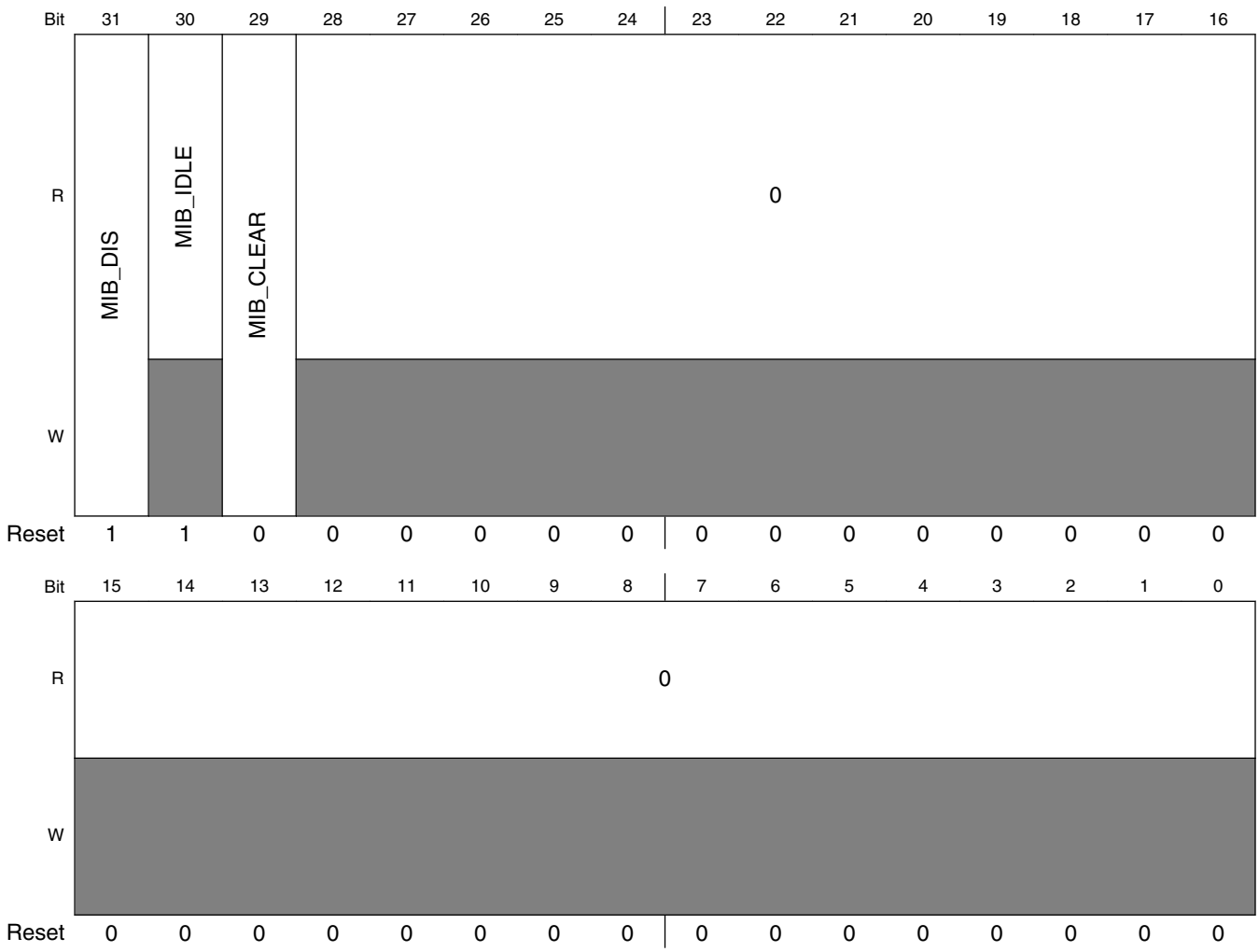
ENET_MSCR field descriptions

Field	Description
31–11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10–8 HOLDTIME	Hold time On MDIO Output IEEE802.3 clause 22 defines a minimum of 10 ns for the hold time on the MDIO output. Depending on the host bus frequency, the setting may need to be increased. 000 1 internal module clock cycle 001 2 internal module clock cycles 010 3 internal module clock cycles 111 8 internal module clock cycles
7 DIS_PRE	Disable Preamble Enables/disables prepending a preamble to the MII management frame. The MII standard allows the preamble to be dropped if the attached PHY devices do not require it. 0 Preamble enabled. 1 Preamble (32 ones) is not prepended to the MII management frame.
6–1 MII_SPEED	MII Speed Controls the frequency of the MII management interface clock (MDC) relative to the internal module clock. A value of 0 in this field turns off MDC and leaves it in low voltage state. Any non-zero value results in the MDC frequency of: 1/((MII_SPEED + 1) x 2) of the internal module clock frequency
0 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.8 MIB Control Register (ENET_MIBC)

MIBC is a read/write register controlling and observing the state of the MIB block. Access this register to disable the MIB block operation or clear the MIB counters. The MIB_DIS field resets to 1.

Address: 218_8000h base + 64h offset = 218_8064h



ENET_MIBC field descriptions

Field	Description
31 MIB_DIS	Disable MIB Logic If this control field is set, 0 MIB logic is enabled. 1 MIB logic is disabled. The MIB logic halts and does not update any MIB counters.

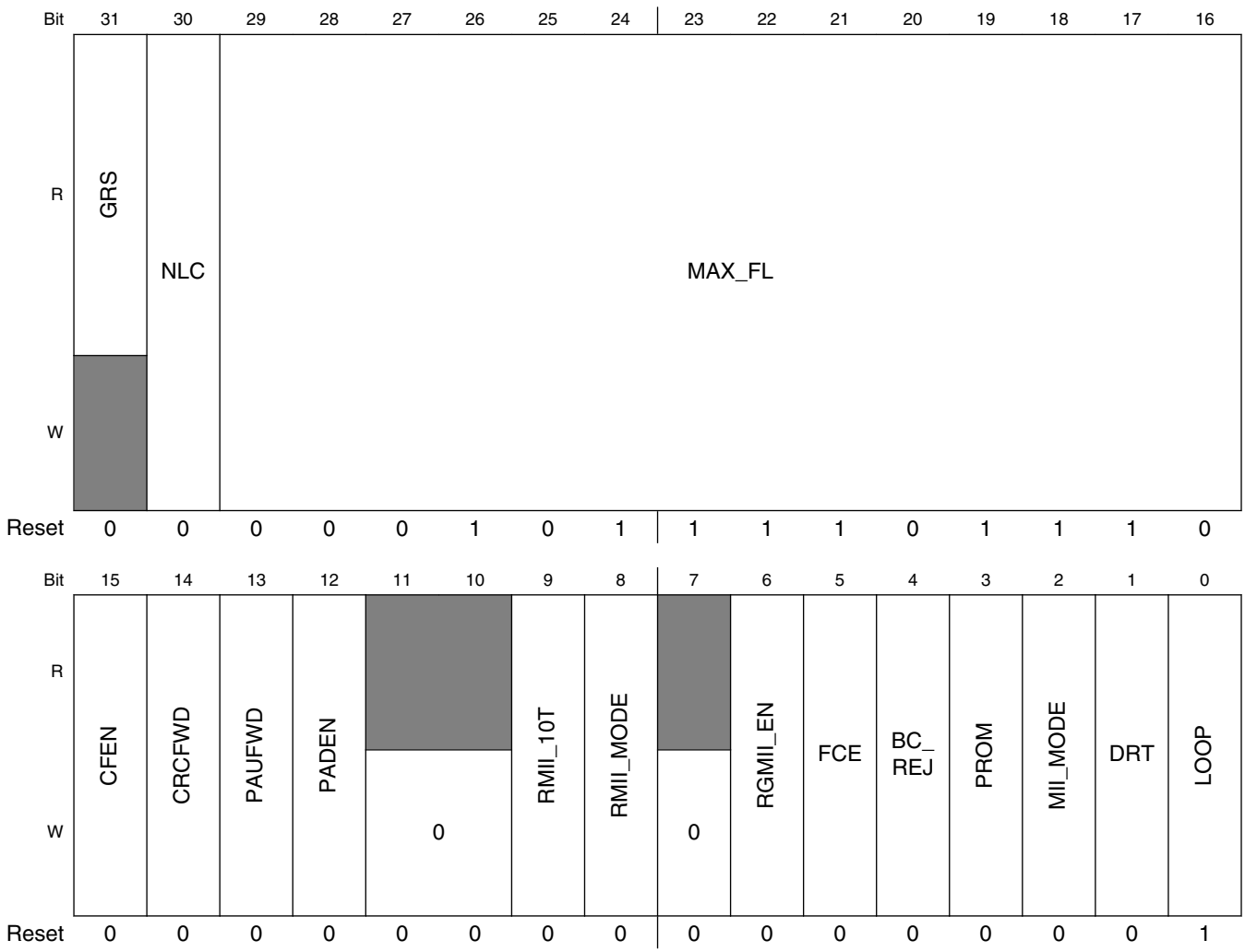
Table continues on the next page...

ENET_MIBC field descriptions (continued)

Field	Description
30 MIB_IDLE	MIB Idle 0 The MIB block is updating MIB counters. 1 The MIB block is not currently updating any MIB counters.
29 MIB_CLEAR	MIB Clear **NOTE:** This field is not self-clearing. To clear the MIB counters set and then clear this field. 0 See note above. 1 All statistics counters are reset to 0.
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.9 Receive Control Register (ENET_RCR)

Address: 218_8000h base + 84h offset = 218_8084h



ENET_RCR field descriptions

Field	Description
31 GRS	Graceful Receive Stopped Read-only status indicating that the MAC receive datapath is stopped.
30 NLC	Payload Length Check Disable Enables/disables a payload length check. 0 The payload length check is disabled. 1 The core checks the frame's payload length with the frame length/type field. Errors are indicated in the EIR[PLC] field.
29–16 MAX_FL	Maximum Frame Length

Table continues on the next page...

ENET_RCR field descriptions (continued)

Field	Description
	Resets to decimal 1518. Length is measured starting at DA and includes the CRC at the end of the frame. Transmit frames longer than MAX_FL cause the BABT interrupt to occur. Receive frames longer than MAX_FL cause the BABR interrupt to occur and set the LG field in the end of frame receive buffer descriptor. The recommended default value to be programmed is 1518 or 1522 if VLAN tags are supported.
15 CFEN	MAC Control Frame Enable Enables/disables the MAC control frame. 0 MAC control frames with any opcode other than 0x0001 (pause frame) are accepted and forwarded to the client interface. 1 MAC control frames with any opcode other than 0x0001 (pause frame) are silently discarded.
14 CRCFWD	Terminate/Forward Received CRC Specifies whether the CRC field of received frames is transmitted or stripped. **NOTE:** If padding function is enabled (PADEN = 1), CRCFWD is ignored and the CRC field is checked and always terminated and removed. 0 The CRC field of received frames is transmitted to the user application. 1 The CRC field is stripped from the frame.
13 PAUFWD	Terminate/Forward Pause Frames Specifies whether pause frames are terminated or forwarded. 0 Pause frames are terminated and discarded in the MAC. 1 Pause frames are forwarded to the user application.
12 PADEN	Enable Frame Padding Remove On Receive Specifies whether the MAC removes padding from received frames. 0 No padding is removed on receive by the MAC. 1 Padding is removed from received frames.
11–10 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
9 RMII_10T	Enables 10-Mbps mode of the RMII or RGMII . 0 100 Mbps operation. 1 10 Mbps operation.
8 RMII_MODE	RMII Mode Enable Specifies whether the MAC is configured for MII mode or RMII operation , when ECR[SPEED] is cleared . **NOTE:** Do not set both RCR[RGMII_EN] and RCR[RMII_MODE]. 0 MAC configured for MII mode. 1 MAC configured for RMII operation.
7 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
6 RGMII_EN	RGMII Mode Enable **NOTE:** Do not set both RCR[RGMII_EN] and RCR[RMII_MODE].

Table continues on the next page...

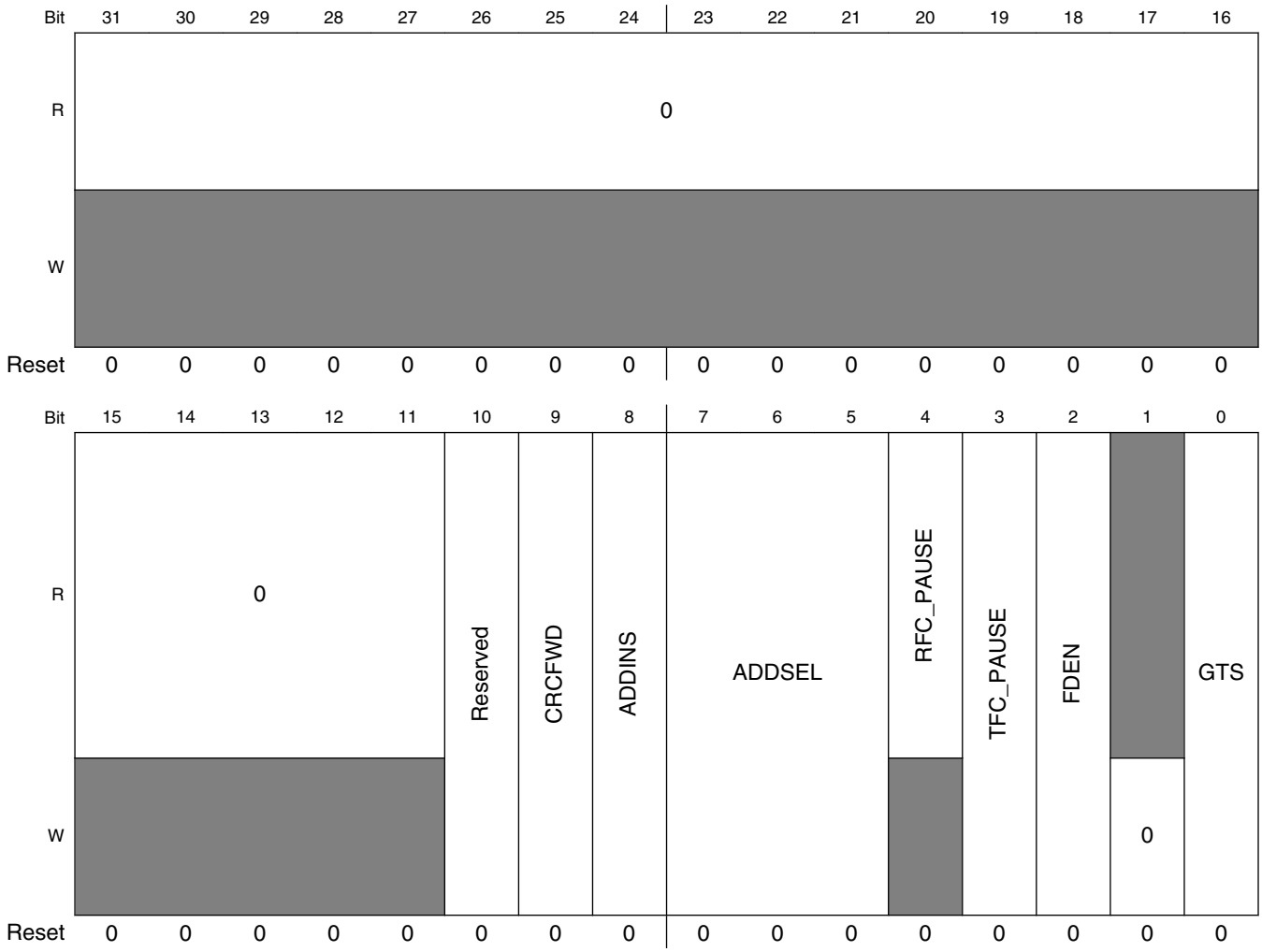
ENET_RCR field descriptions (continued)

Field	Description
	0 MAC configured for non-RGMII operation 1 MAC configured for RGMII operation. If ECR[SPEED] is set, the MAC is in RGMII 1000 Mbps mode. If ECR[SPEED] is cleared, the MAC is in RGMII 10/100 Mbps mode.
5 FCE	Flow Control Enable If set, the receiver detects PAUSE frames. Upon PAUSE frame detection, the transmitter stops transmitting data frames for a given duration.
4 BC_REJ	Broadcast Frame Reject If set, frames with destination address (DA) equal to 0xFFFF_FFFF_FFFF are rejected unless the PROM field is set. If BC_REJ and PROM are set, frames with broadcast DA are accepted and the MISS (M) is set in the receive buffer descriptor.
3 PROM	Promiscuous Mode All frames are accepted regardless of address matching. 0 Disabled. 1 Enabled.
2 MII_MODE	Media Independent Interface Mode This field must always be set. 0 Reserved. 1 MII or RMII mode, as indicated by the RMII_MODE field.
1 DRT	Disable Receive On Transmit 0 Receive path operates independently of transmit. Used for full-duplex or to monitor transmit activity in half-duplex mode. 1 Disable reception of frames while transmitting. Normally used for half-duplex mode.
0 LOOP	Internal Loopback This is an MII internal loopback, therefore MII_MODE must be written to 1 and RMII_MODE must be written to 0. 0 Loopback disabled. 1 Transmitted frames are looped back internal to the device and transmit MII output signals are not asserted. DRT must be cleared.

23.5.10 Transmit Control Register (ENET_TCR)

TCR is read/write and configures the transmit block. This register is cleared at system reset. FDEN can only be modified when ECR[ETHEREN] is cleared.

Address: 218_8000h base + C4h offset = 218_80C4h



ENET_TCR field descriptions

Field	Description
31–11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10 Reserved	This field is reserved. This field is read/write and must be set to 0.
9 CRCFWD	Forward Frame From Application With CRC

Table continues on the next page...

ENET_TCR field descriptions (continued)

Field	Description
	0 TxBD[TC] controls whether the frame has a CRC from the application. 1 The transmitter does not append any CRC to transmitted frames, as it is expecting a frame with CRC from the application.
8 ADDINS	Set MAC Address On Transmit 0 The source MAC address is not modified by the MAC. 1 The MAC overwrites the source MAC address with the programmed MAC address according to ADDSEL.
7–5 ADDSEL	Source MAC Address Select On Transmit If ADDINS is set, indicates the MAC address that overwrites the source MAC address. 000 Node MAC address programmed on PADDR1/2 registers. 100 Reserved. 101 Reserved. 110 Reserved.
4 RFC_PAUSE	Receive Frame Control Pause This status field is set when a full-duplex flow control pause frame is received and the transmitter pauses for the duration defined in this pause frame. This field automatically clears when the pause duration is complete.
3 TFC_PAUSE	Transmit Frame Control Pause Pauses frame transmission. When this field is set, EIR[GRA] is set. With transmission of data frames stopped, the MAC transmits a MAC control PAUSE frame. Next, the MAC clears TFC_PAUSE and resumes transmitting data frames. If the transmitter pauses due to user assertion of GTS or reception of a PAUSE frame, the MAC may continue transmitting a MAC control PAUSE frame. 0 No PAUSE frame transmitted. 1 The MAC stops transmission of data frames after the current transmission is complete.
2 FDEN	Full-Duplex Enable If this field is set, frames transmit independent of carrier sense and collision inputs. Only modify this bit when ECR[ETHEREN] is cleared.
1 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
0 GTS	Graceful Transmit Stop When this field is set, MAC stops transmission after any frame currently transmitted is complete and EIR[GRA] is set. If frame transmission is not currently underway, the GRA interrupt is asserted immediately. After transmission finishes, clear GTS to restart. The next frame in the transmit FIFO is then transmitted. If an early collision occurs during transmission when GTS is set, transmission stops after the collision. The frame is transmitted again after GTS is cleared. There may be old frames in the transmit FIFO that transmit when GTS is reasserted. To avoid this, clear ECR[ETHEREN] following the GRA interrupt.

23.5.11 Physical Address Lower Register (ENET_PALR)

PALR contains the lower 32 bits (bytes 0, 1, 2, 3) of the 48-bit address used in the address recognition process to compare with the destination address (DA) field of receive frames with an individual DA. In addition, this register is used in bytes 0 through 3 of the six-byte source address field when transmitting PAUSE frames. This register is not reset and you must initialize it.

Address: 218_8000h base + E4h offset = 218_80E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PADDR1																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_PALR field descriptions

Field	Description
PADDR1	Pause Address Bytes 0 (bits 31:24), 1 (bits 23:16), 2 (bits 15:8), and 3 (bits 7:0) of the 6-byte individual address are used for exact match and the source address field in PAUSE frames.

23.5.12 Physical Address Upper Register (ENET_PAUR)

PAUR contains the upper 16 bits (bytes 4 and 5) of the 48-bit address used in the address recognition process to compare with the destination address (DA) field of receive frames with an individual DA. In addition, this register is used in bytes 4 and 5 of the six-byte source address field when transmitting PAUSE frames. Bits 15:0 of PAUR contain a constant type field (0x8808) for transmission of PAUSE frames. The upper 16 bits of this register are not reset and you must initialize it.

Address: 218_8000h base + E8h offset = 218_80E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PADDR2																TYPE															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0

ENET_PAUR field descriptions

Field	Description
31–16 PADDR2	Bytes 4 (bits 31:24) and 5 (bits 23:16) of the 6-byte individual address used for exact match, and the source address field in PAUSE frames.
TYPE	Type Field In PAUSE Frames

Table continues on the next page...

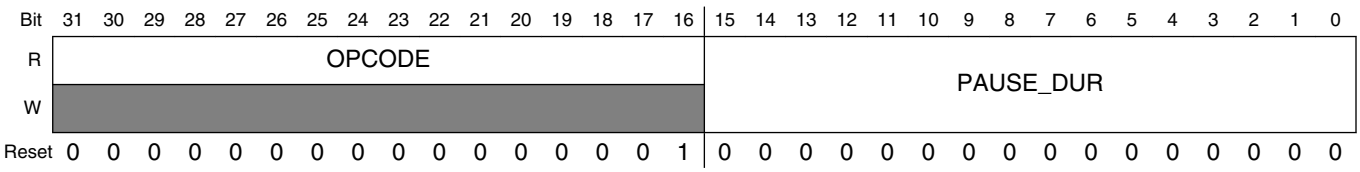
ENET_PAUR field descriptions (continued)

Field	Description
	These fields have a constant value of 0x8808.

23.5.13 Opcode/Pause Duration Register (ENET_OPD)

OPD is read/write accessible. This register contains the 16-bit opcode and 16-bit pause duration fields used in transmission of a PAUSE frame. The opcode field is a constant value, 0x0001. When another node detects a PAUSE frame, that node pauses transmission for the duration specified in the pause duration field. The lower 16 bits of this register are not reset and you must initialize it.

Address: 218_8000h base + ECh offset = 218_80ECh



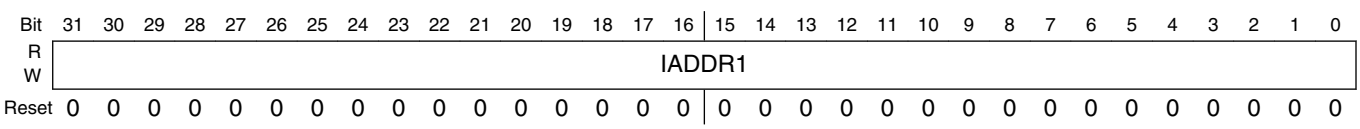
ENET_OPD field descriptions

Field	Description
31–16 OPCODE	Opcode Field In PAUSE Frames These fields have a constant value of 0x0001.
PAUSE_DUR	Pause Duration Pause duration field used in PAUSE frames.

23.5.14 Descriptor Individual Upper Address Register (ENET_IAUR)

IAUR contains the upper 32 bits of the 64-bit individual address hash table. The address recognition process uses this table to check for a possible match with the destination address (DA) field of receive frames with an individual DA. This register is not reset and you must initialize it.

Address: 218_8000h base + 118h offset = 218_8118h



ENET_IAUR field descriptions

Field	Description
IADDR1	Contains the upper 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a unicast address. Bit 31 of IADDR1 contains hash index bit 63. Bit 0 of IADDR1 contains hash index bit 32.

23.5.15 Descriptor Individual Lower Address Register (ENET_IALR)

IALR contains the lower 32 bits of the 64-bit individual address hash table. The address recognition process uses this table to check for a possible match with the DA field of receive frames with an individual DA. This register is not reset and you must initialize it.

Address: 218_8000h base + 11Ch offset = 218_811Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IADDR2																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IALR field descriptions

Field	Description
IADDR2	Contains the lower 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a unicast address. Bit 31 of IADDR2 contains hash index bit 31. Bit 0 of IADDR2 contains hash index bit 0.

23.5.16 Descriptor Group Upper Address Register (ENET_GAUR)

GAUR contains the upper 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a multicast address. You must initialize this register.

Address: 218_8000h base + 120h offset = 218_8120h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	GADDR1																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_GAUR field descriptions

Field	Description
GADDR1	Contains the upper 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a multicast address. Bit 31 of GADDR1 contains hash index bit 63. Bit 0 of GADDR1 contains hash index bit 32.

23.5.17 Descriptor Group Lower Address Register (ENET_GALR)

GALR contains the lower 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a multicast address. You must initialize this register.

Address: 218_8000h base + 124h offset = 218_8124h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	GADDR2																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_GALR field descriptions

Field	Description
GADDR2	Contains the lower 32 bits of the 64-bit hash table used in the address recognition process for receive frames with a multicast address. Bit 31 of GADDR2 contains hash index bit 31. Bit 0 of GADDR2 contains hash index bit 0.

23.5.18 Transmit FIFO Watermark Register (ENET_TFWR)

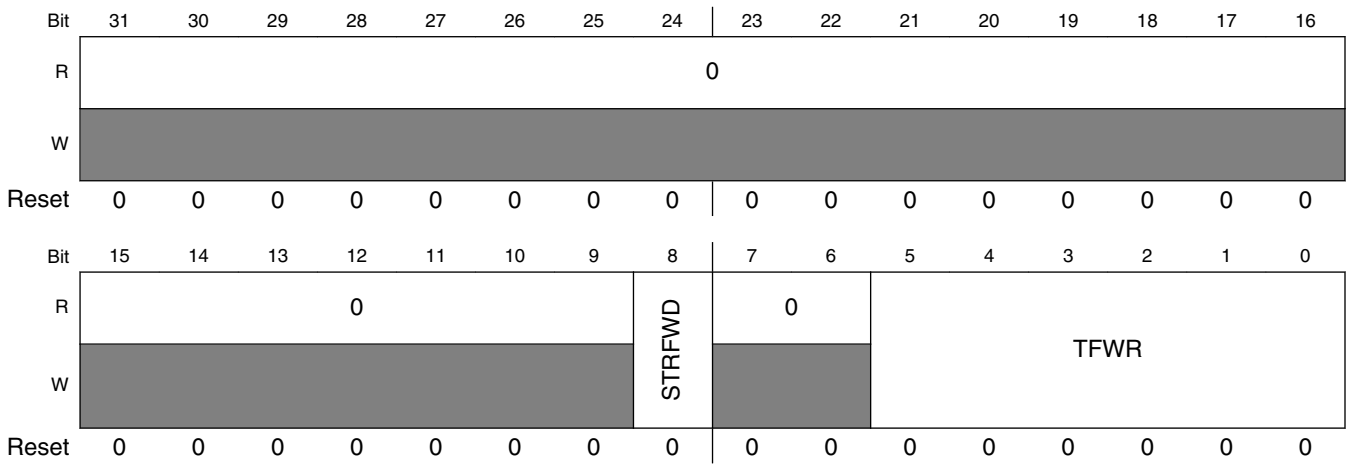
If TFWR[STRFWD] is cleared, TFWR[TFWR] controls the amount of data required in the transmit FIFO before transmission of a frame can begin. This allows you to minimize transmit latency (TFWR = 00 or 01) or allow for larger bus access latency (TFWR = 11) due to contention for the system bus. Setting the watermark to a high value minimizes the risk of transmit FIFO underrun due to contention for the system bus. The byte counts associated with the TFWR field may need to be modified to match a given system requirement. For example, worst case bus access latency by the transmit data DMA channel.

When the FIFO level reaches the value the TFWR field and when the STR_FWD is set to ‘0’, the MAC transmit control logic starts frame transmission even before the end-of-frame is available in the FIFO (cut-through operation).

If a complete frame has a size smaller than the threshold programmed with TFWR, the MAC also transmits the Frame to the line.

To enable store and forward on the Transmit path, set STR_FWD to ‘1’. In this case, the MAC starts to transmit data only when a complete frame is stored in the Transmit FIFO.

Address: 218_8000h base + 144h offset = 218_8144h



ENET_TFWR field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
8 STRFWD	Store And Forward Enable 0 Reset. The transmission start threshold is programmed in TFWR[TFWR]. 1 Enabled.
7–6 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TFWR	Transmit FIFO Write If TFWR[STRFWD] is cleared, this field indicates the number of bytes, in steps of 64 bytes, written to the transmit FIFO before transmission of a frame begins. **NOTE:** If a frame with less than the threshold is written, it is still sent independently of this threshold setting. The threshold is relevant only if the frame is larger than the threshold given. 000000 64 bytes written. 000001 64 bytes written. 000010 128 bytes written. 000011 192 bytes written. ... ... 111111 4032 bytes written.

23.5.19 Receive Descriptor Ring Start Register (ENET_RDSR)

RDSR points to the beginning of the circular receive buffer descriptor queue in external memory. This pointer must be 64-bit aligned (bits 2–0 must be zero); however, it is recommended to be 128-bit aligned, that is, evenly divisible by 16.

NOTE

This register must be initialized prior to operation

Address: 218_8000h base + 180h offset = 218_8180h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	R_DES_START															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	R_DES_START														0	
W														0		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RDSR field descriptions

Field	Description
31–3 R_DES_START	Pointer to the beginning of the receive buffer descriptor queue.
2 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.20 Transmit Buffer Descriptor Ring Start Register (ENET_TDSR)

TDSR provides a pointer to the beginning of the circular transmit buffer descriptor queue in external memory. This pointer must be 64-bit aligned (bits 2–0 must be zero); however, it is recommended to be 128-bit aligned, that is, evenly divisible by 16.

NOTE

This register must be initialized prior to operation.

Address: 218_8000h base + 184h offset = 218_8184h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	X_DES_START															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	X_DES_START														0	
W														0		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_TDSR field descriptions

Field	Description
31–3 X_DES_START	Pointer to the beginning of the transmit buffer descriptor queue.

Table continues on the next page...

ENET_TDSR field descriptions (continued)

Field	Description
2 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.21 Maximum Receive Buffer Size Register (ENET_MRBR)

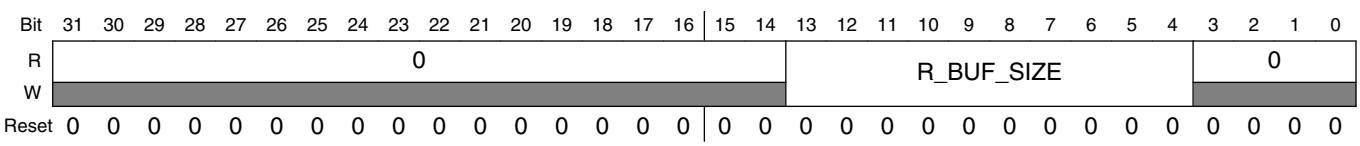
The MRBR is a user-programmable register that dictates the maximum size of all receive buffers. This value should take into consideration that the receive CRC is always written into the last receive buffer.

- To allow one maximum size frame per buffer, MRBR must be set to RCR[MAX_FL] or larger.
- To properly align the buffer, MRBR must be evenly divisible by 16. To ensure this, bits 3–0 are set to zero by the device.
- To minimize bus usage (descriptor fetches), set MRBR greater than or equal to 256 bytes.

NOTE

This register must be initialized before operation.

Address: 218_8000h base + 188h offset = 218_8188h

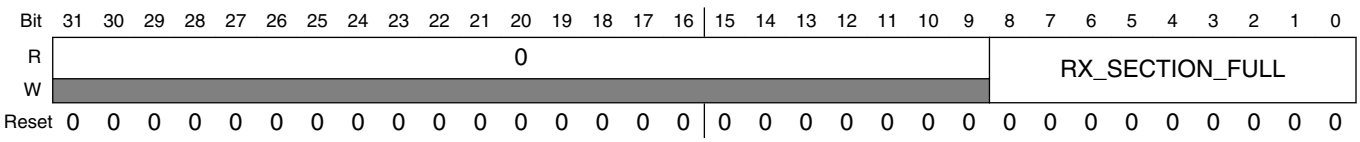


ENET_MRBR field descriptions

Field	Description
31–14 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
13–4 R_BUF_SIZE	Receive buffer size in bytes.
Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

23.5.22 Receive FIFO Section Full Threshold (ENET_RSFL)

Address: 218_8000h base + 190h offset = 218_8190h

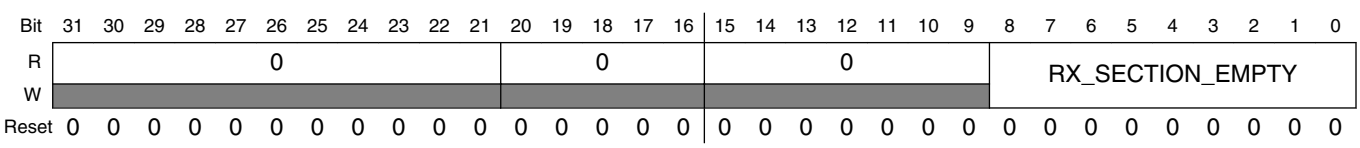


ENET_RSFL field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RX_SECTION_FULL	Value Of Receive FIFO Section Full Threshold Value, in 64-bit words, of the receive FIFO section full threshold. Clear this field to enable store and forward on the RX FIFO. When programming a value greater than 0 (cut-through operation), it must be greater than RAEM[RX_ALMOST_EMPTY]. When the FIFO level reaches the value in this field, data is available in the Receive FIFO (cut-through operation).

23.5.23 Receive FIFO Section Empty Threshold (ENET_RSEM)

Address: 218_8000h base + 194h offset = 218_8194h



ENET_RSEM field descriptions

Field	Description
31–21 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
20–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
15–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RX_SECTION_EMPTY	Value Of The Receive FIFO Section Empty Threshold Value, in 64-bit words, of the receive FIFO section empty threshold. When the FIFO has reached this level, a pause frame will be issued. A value of 0 disables automatic pause frame generation. When the FIFO level goes below the value programmed in this field, an XON pause frame is issued to indicate the FIFO congestion is cleared to the remote Ethernet client.

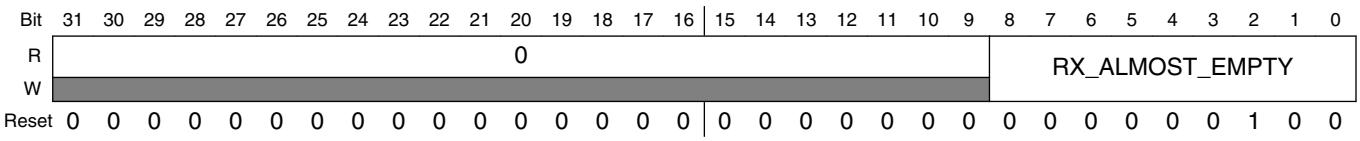
Table continues on the next page...

ENET_RSEM field descriptions (continued)

Field	Description
	NOTE: The section-empty threshold indications from both FIFOs are OR'ed to cause XOFF pause frame generation.

23.5.24 Receive FIFO Almost Empty Threshold (ENET_RAEM)

Address: 218_8000h base + 198h offset = 218_8198h

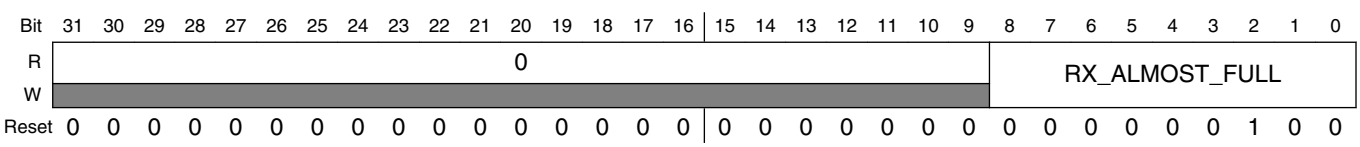


ENET_RAEM field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RX_ALMOST_EMPTY	Value Of The Receive FIFO Almost Empty Threshold Value, in 64-bit words, of the receive FIFO almost empty threshold. When the FIFO level reaches the value programmed in this field and the end-of-frame has not been received for the frame yet, the core receive read control stops FIFO read (and subsequently stops transferring data to the MAC client application). It continues to deliver the frame, if again more data than the threshold or the end-of-frame is available in the FIFO. A minimum value of 4 should be set.

23.5.25 Receive FIFO Almost Full Threshold (ENET_RAFL)

Address: 218_8000h base + 19Ch offset = 218_819Ch



ENET_RAFL field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RX_ALMOST_FULL	Value Of The Receive FIFO Almost Full Threshold Value, in 64-bit words, of the receive FIFO almost full threshold. When the FIFO level comes close to the maximum, so that there is no more space for at least RX_ALMOST_FULL number of words, the MAC stops writing data in the FIFO and truncates the received frame to avoid FIFO overflow. The corresponding error status will be set when the frame is delivered to the application. A minimum value of 4 should be set.

23.5.26 Transmit FIFO Section Empty Threshold (ENET_TSEM)

Address: 218_8000h base + 1A0h offset = 218_81A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TX_SECTION_EMPTY																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_TSEM field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TX_SECTION_ EMPTY	Value Of The Transmit FIFO Section Empty Threshold Value, in 64-bit words, of the transmit FIFO section empty threshold. See Transmit FIFO for more information.

23.5.27 Transmit FIFO Almost Empty Threshold (ENET_TAEM)

Address: 218_8000h base + 1A4h offset = 218_81A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TX_ALMOST_EMPTY																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

ENET_TAEM field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TX_ALMOST_ EMPTY	Value of Transmit FIFO Almost Empty Threshold Value, in 64-bit words, of the transmit FIFO almost empty threshold. When the FIFO level reaches the value programmed in this field, and no end-of-frame is available for the frame, the MAC transmit logic, to avoid FIFO underflow, stops reading the FIFO and transmits a frame with an MII error indication. See Transmit FIFO for more information. A minimum value of 4 should be set.

23.5.28 Transmit FIFO Almost Full Threshold (ENET_TAFL)

Address: 218_8000h base + 1A8h offset = 218_81A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																							TX_ALMOST_FULL								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

ENET_TAFL field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TX_ALMOST_FULL	Value Of The Transmit FIFO Almost Full Threshold Value, in 64-bit words, of the transmit FIFO almost full threshold. A minimum value of six is required . A recommended value of at least 8 should be set allowing a latency of two clock cycles to the application. If more latency is required the value can be increased as necessary (latency = TAFL - 5). When the FIFO level comes close to the maximum, so that there is no more space for at least TX_ALMOST_FULL number of words, the pin ff_tx_rdy is deasserted. If the application does not react on this signal, the FIFO write control logic, to avoid FIFO overflow, truncates the current frame and sets the error status. As a result, the frame will be transmitted with an GMII/MII error indication. See Transmit FIFO for more information. **NOTE:** A FIFO overflow is a fatal error and requires a global reset on the transmit datapath or at least deassertion of ETHEREN.

23.5.29 Transmit Inter-Packet Gap (ENET_TIPG)

Address: 218_8000h base + 1ACh offset = 218_81ACh

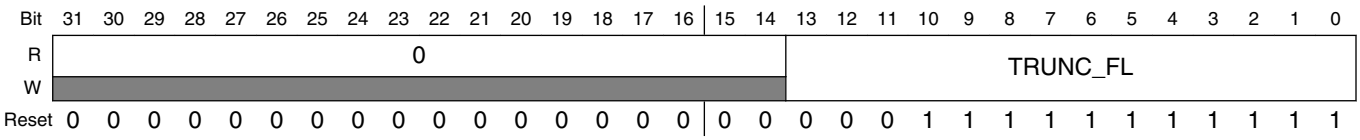
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																											IPG				
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0

ENET_TIPG field descriptions

Field	Description
31–5 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
IPG	Transmit Inter-Packet Gap Indicates the IPG, in bytes, between transmitted frames. Valid values range from 8 to 27. If value is less than 8, the IPG is 8. If value is greater than 27, the IPG is 27.

23.5.30 Frame Truncation Length (ENET_FTRL)

Address: 218_8000h base + 1B0h offset = 218_81B0h



ENET_FTRL field descriptions

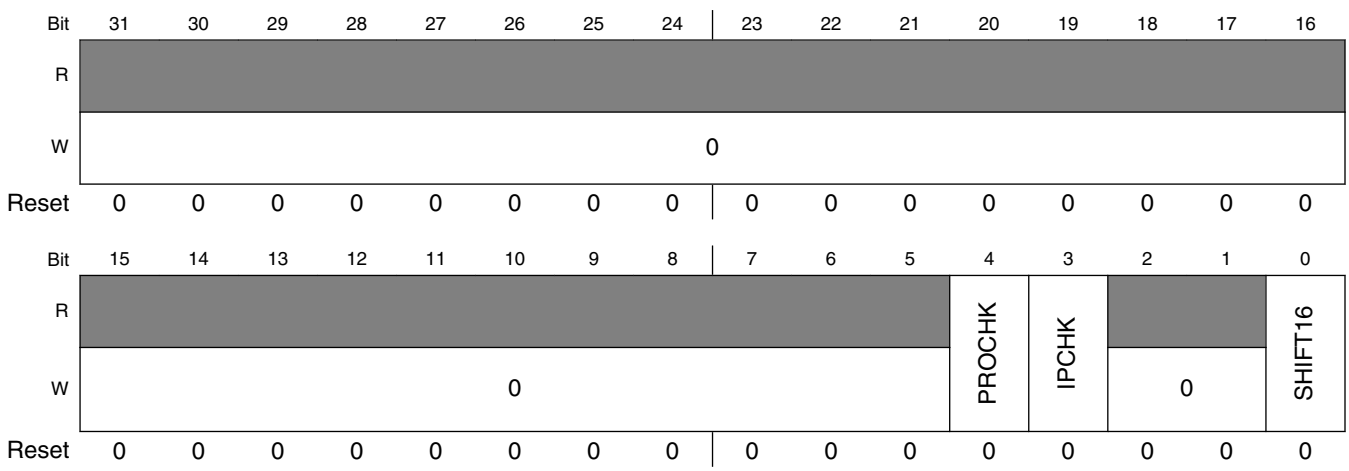
Field	Description
31–14 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TRUNC_FL	Frame Truncation Length Indicates the value a receive frame is truncated, if it is greater than this value. Must be greater than or equal to RCR[MAX_FL]. **NOTE:** Truncation happens at TRUNC_FL. However, when truncation occurs, the application (FIFO) may receive less data, guaranteeing that it never receives more than the set limit.

23.5.31 Transmit Accelerator Function Configuration (ENET_TACC)

TACC controls accelerator actions when sending frames. The register can be changed before or after each frame, but it must remain unmodified during frame writes into the transmit FIFO.

The TFWR[STRFWD] field must be set to use the checksum feature.

Address: 218_8000h base + 1C0h offset = 218_81C0h



ENET_TACC field descriptions

Field	Description
31–5 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
4 PROCHK	Enables insertion of protocol checksum. 0 Checksum not inserted. 1 If an IP frame with a known protocol is transmitted, the checksum is inserted automatically into the frame. The checksum field must be cleared. The other frames are not modified.
3 IPCHK	Enables insertion of IP header checksum. 0 Checksum is not inserted. 1 If an IP frame is transmitted, the checksum is inserted automatically. The IP header checksum field must be cleared. If a non-IP frame is transmitted the frame is not modified.
2–1 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
0 SHIFT16	TX FIFO Shift-16 0 Disabled. 1 Indicates to the transmit data FIFO that the written frames contain two additional octets before the frame data. This means the actual frame begins at bit 16 of the first word written into the FIFO. This function allows putting the frame payload on a 32-bit boundary in memory, as the 14-byte Ethernet header is extended to a 16-byte header.

23.5.32 Receive Accelerator Function Configuration (ENET_RACC)

Address: 218_8000h base + 1C4h offset = 218_81C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R																
W									0							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R									SHIFT16	LINEDIS				PRODIS	IPDIS	PADREM
W				0							0					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RACC field descriptions

Field	Description
31–8 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.

Table continues on the next page...

ENET_RACC field descriptions (continued)

Field	Description
7 SHIFT16	RX FIFO Shift-16 When this field is set, the actual frame data starts at bit 16 of the first word read from the RX FIFO aligning the Ethernet payload on a 32-bit boundary. **NOTE:** This function only affects the FIFO storage and has no influence on the statistics, which use the actual length of the frame received. 0 Disabled. 1 Instructs the MAC to write two additional bytes in front of each frame received into the RX FIFO.
6 LINEDIS	Enable Discard Of Frames With MAC Layer Errors 0 Frames with errors are not discarded. 1 Any frame received with a CRC, length, or PHY error is automatically discarded and not forwarded to the user application interface.
5–3 Reserved	This field is reserved. This write-only field is reserved. It must always be written with the value 0.
2 PRODIS	Enable Discard Of Frames With Wrong Protocol Checksum 0 Frames with wrong checksum are not discarded. 1 If a TCP/IP, UDP/IP, or ICMP/IP frame is received that has a wrong TCP, UDP, or ICMP checksum, the frame is discarded. Discarding is only available when the RX FIFO operates in store and forward mode (RSFL cleared).
1 IPDIS	Enable Discard Of Frames With Wrong IPv4 Header Checksum 0 Frames with wrong IPv4 header checksum are not discarded. 1 If an IPv4 frame is received with a mismatching header checksum, the frame is discarded. IPv6 has no header checksum and is not affected by this setting. Discarding is only available when the RX FIFO operates in store and forward mode (RSFL cleared).
0 PADREM	Enable Padding Removal For Short IP Frames 0 Padding not removed. 1 Any bytes following the IP payload section of the frame are removed from the frame.

23.5.33 Tx Packet Count Statistic Register (ENET_RMON_T_PACKETS)

Address: 218_8000h base + 204h offset = 218_8204h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_PACKETS field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.34 Tx Broadcast Packets Statistic Register (ENET_RMON_T_BC_PKT)

RMON Tx Broadcast Packets

Address: 218_8000h base + 208h offset = 218_8208h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_BC_PKT field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Broadcast packets

23.5.35 Tx Multicast Packets Statistic Register (ENET_RMON_T_MC_PKT)

Address: 218_8000h base + 20Ch offset = 218_820Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_MC_PKT field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Multicast packets

23.5.36 Tx Packets with CRC/Align Error Statistic Register (ENET_RMON_T_CRC_ALIGN)

Address: 218_8000h base + 210h offset = 218_8210h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_CRC_ALIGN field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packets with CRC/align error

23.5.37 Tx Packets Less Than Bytes and Good CRC Statistic Register (ENET_RMON_T_UNDERSIZE)

Address: 218_8000h base + 214h offset = 218_8214h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_UNDERSIZE field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.38 Tx Packets GT MAX_FL bytes and Good CRC Statistic Register (ENET_RMON_T_OVERSIZE)

Address: 218_8000h base + 218h offset = 218_8218h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_OVERSIZE field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.39 Tx Packets Less Than 64 Bytes and Bad CRC Statistic Register (ENET_RMON_T_FRAG)

.

Address: 218_8000h base + 21Ch offset = 218_821Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_FRAG field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.40 Tx Packets Greater Than MAX_FL bytes and Bad CRC Statistic Register (ENET_RMON_T_JAB)

Address: 218_8000h base + 220h offset = 218_8220h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_JAB field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.41 Tx Collision Count Statistic Register (ENET_RMON_T_COL)

Address: 218_8000h base + 224h offset = 218_8224h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_COL field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.42 Tx 64-Byte Packets Statistic Register (ENET_RMON_T_P64)

.

Address: 218_8000h base + 228h offset = 218_8228h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_P64 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.43 Tx 65- to 127-byte Packets Statistic Register (ENET_RMON_T_P65TO127)

Address: 218_8000h base + 22Ch offset = 218_822Ch

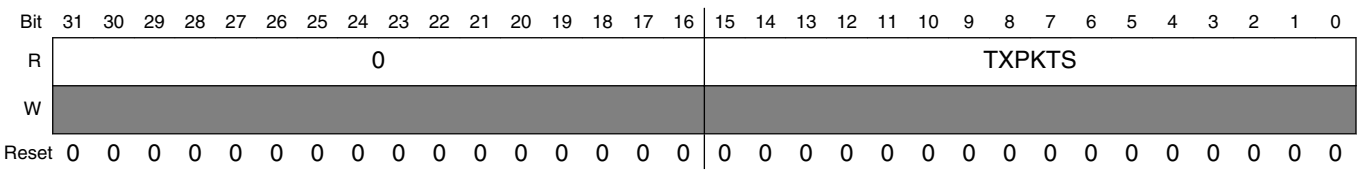
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_P65TO127 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.44 Tx 128- to 255-byte Packets Statistic Register (ENET_RMON_T_P128TO255)

Address: 218_8000h base + 230h offset = 218_8230h

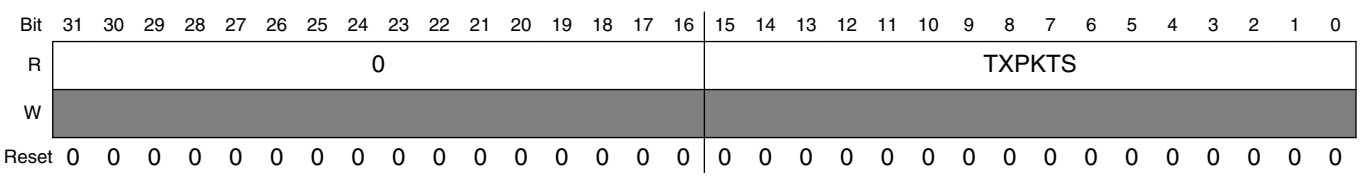


ENET_RMON_T_P128TO255 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.45 Tx 256- to 511-byte Packets Statistic Register (ENET_RMON_T_P256TO511)

Address: 218_8000h base + 234h offset = 218_8234h



ENET_RMON_T_P256TO511 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.46 Tx 512- to 1023-byte Packets Statistic Register (ENET_RMON_T_P512TO1023)

.

Address: 218_8000h base + 238h offset = 218_8238h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_P512TO1023 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.47 Tx 1024- to 2047-byte Packets Statistic Register (ENET_RMON_T_P1024TO2047)

Address: 218_8000h base + 23Ch offset = 218_823Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_P1024TO2047 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.48 Tx Packets Greater Than 2048 Bytes Statistic Register (ENET_RMON_T_P_GTE2048)

Address: 218_8000h base + 240h offset = 218_8240h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TXPKTS															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_P_GTE2048 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXPKTS	Packet count

23.5.49 Tx Octets Statistic Register (ENET_RMON_T_OCTETS)

Address: 218_8000h base + 244h offset = 218_8244h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	TXOCTS																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_T_OCTETS field descriptions

Field	Description
TXOCTS	Octet count

23.5.50 Frames Transmitted OK Statistic Register (ENET_IEEE_T_FRAME_OK)

Address: 218_8000h base + 24Ch offset = 218_824Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_FRAME_OK field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.51 Frames Transmitted with Single Collision Statistic Register (ENET_IEEE_T_1COL)

Address: 218_8000h base + 250h offset = 218_8250h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_1COL field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.52 Frames Transmitted with Multiple Collisions Statistic Register (ENET_IEEE_T_MCOL)

Address: 218_8000h base + 254h offset = 218_8254h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_MCOL field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.53 Frames Transmitted after Deferral Delay Statistic Register (ENET_IEEE_T_DEF)

Address: 218_8000h base + 258h offset = 218_8258h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_DEF field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.54 Frames Transmitted with Late Collision Statistic Register (ENET_IEEE_T_LCOL)

Address: 218_8000h base + 25Ch offset = 218_825Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_LCOL field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.55 Frames Transmitted with Excessive Collisions Statistic Register (ENET_IEEE_T_EXCOL)

Address: 218_8000h base + 260h offset = 218_8260h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_EXCOL field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.56 Frames Transmitted with Tx FIFO Underrun Statistic Register (ENET_IEEE_T_MACERR)

Address: 218_8000h base + 264h offset = 218_8264h

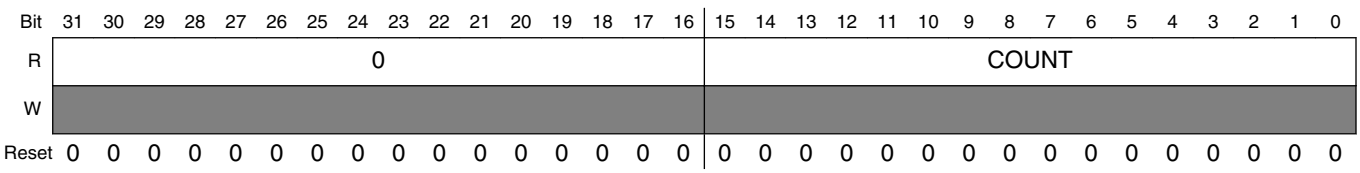
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_T_MACERR field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.57 Frames Transmitted with Carrier Sense Error Statistic Register (ENET_IEEE_T_CSERR)

Address: 218_8000h base + 268h offset = 218_8268h

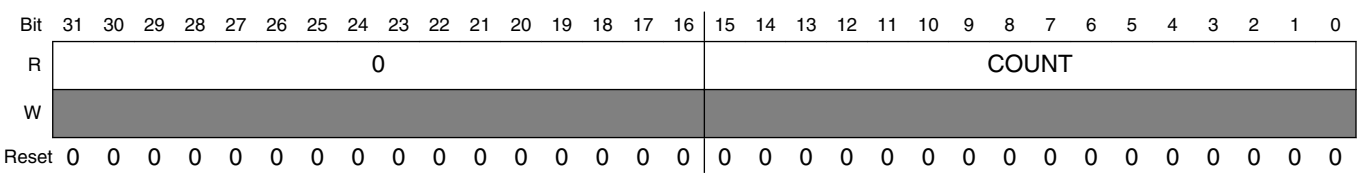


ENET_IEEE_T_CSERR field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.58 Flow Control Pause Frames Transmitted Statistic Register (ENET_IEEE_T_FDXFC)

Address: 218_8000h base + 270h offset = 218_8270h



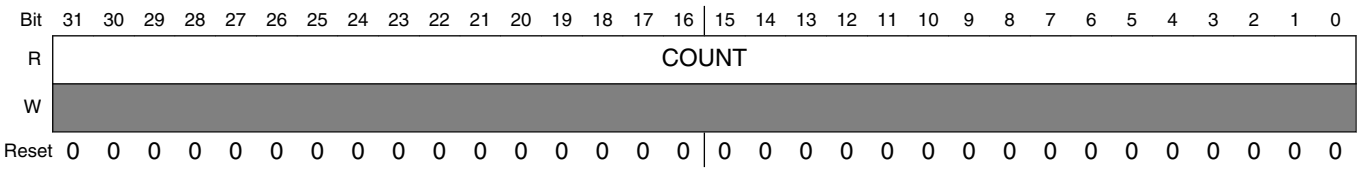
ENET_IEEE_T_FDXFC field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.59 Octet Count for Frames Transmitted w/o Error Statistic Register (ENET_IEEE_T_OCTETS_OK)

Counts total octets (includes header and FCS fields).

Address: 218_8000h base + 274h offset = 218_8274h

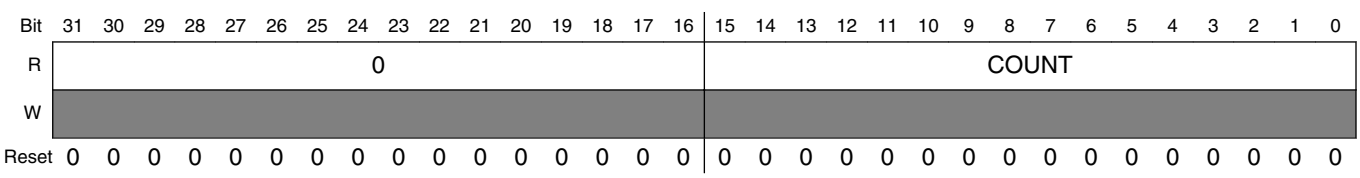


ENET_IEEE_T_OCTETS_OK field descriptions

Field	Description
COUNT	Octet count

23.5.60 Rx Packet Count Statistic Register (ENET_RMON_R_PACKETS)

Address: 218_8000h base + 284h offset = 218_8284h

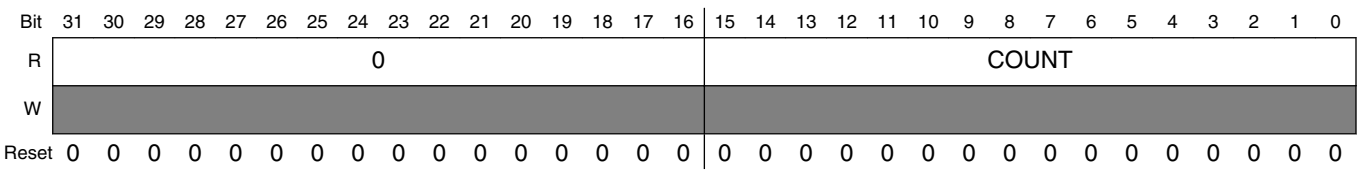


ENET_RMON_R_PACKETS field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.61 Rx Broadcast Packets Statistic Register (ENET_RMON_R_BC_PKT)

Address: 218_8000h base + 288h offset = 218_8288h

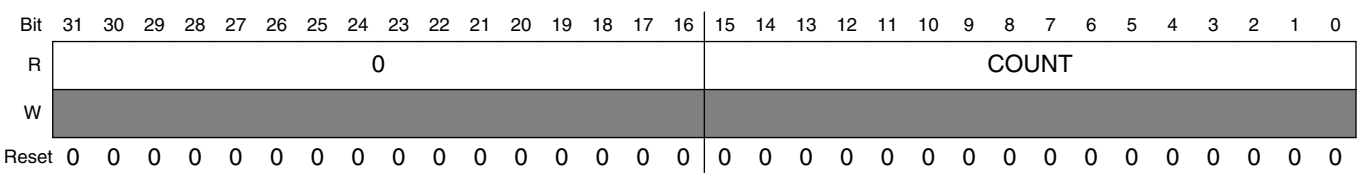


ENET_RMON_R_BC_PKT field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.62 Rx Multicast Packets Statistic Register (ENET_RMON_R_MC_PKT)

Address: 218_8000h base + 28Ch offset = 218_828Ch

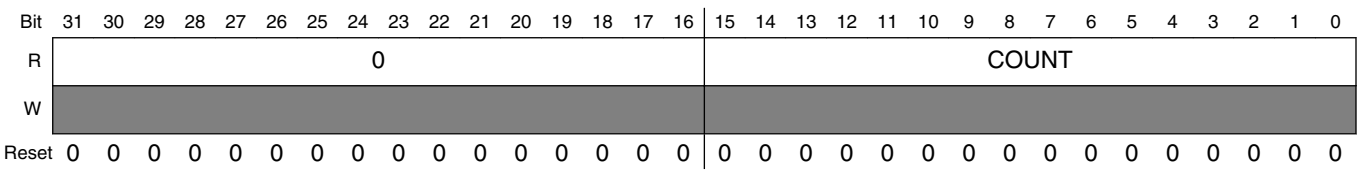


ENET_RMON_R_MC_PKT field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.63 Rx Packets with CRC/Align Error Statistic Register (ENET_RMON_R_CRC_ALIGN)

Address: 218_8000h base + 290h offset = 218_8290h

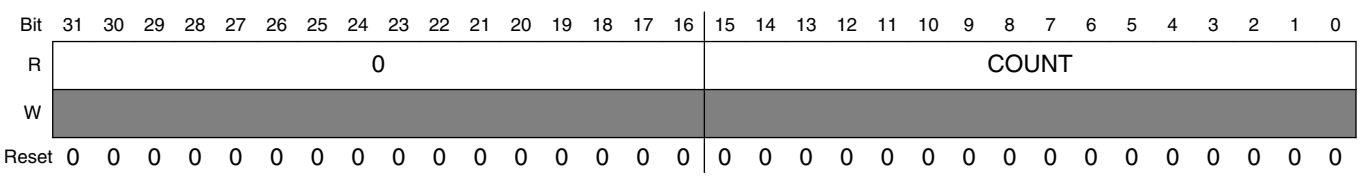


ENET_RMON_R_CRC_ALIGN field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.64 Rx Packets with Less Than 64 Bytes and Good CRC Statistic Register (ENET_RMON_R_UNDERSIZE)

Address: 218_8000h base + 294h offset = 218_8294h



ENET_RMON_R_UNDERSIZE field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.65 Rx Packets Greater Than MAX_FL and Good CRC Statistic Register (ENET_RMON_R_OVERSIZE)

Address: 218_8000h base + 298h offset = 218_8298h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_OVERSIZE field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.66 Rx Packets Less Than 64 Bytes and Bad CRC Statistic Register (ENET_RMON_R_FRAG)

Address: 218_8000h base + 29Ch offset = 218_829Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_FRAG field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.67 Rx Packets Greater Than MAX_FL Bytes and Bad CRC Statistic Register (ENET_RMON_R_JAB)

Address: 218_8000h base + 2A0h offset = 218_82A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_JAB field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.68 Rx 64-Byte Packets Statistic Register (ENET_RMON_R_P64)

Address: 218_8000h base + 2A8h offset = 218_82A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P64 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.69 Rx 65- to 127-Byte Packets Statistic Register (ENET_RMON_R_P65TO127)

Address: 218_8000h base + 2ACh offset = 218_82ACh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P65TO127 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.70 Rx 128- to 255-Byte Packets Statistic Register (ENET_RMON_R_P128TO255)

Address: 218_8000h base + 2B0h offset = 218_82B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P128TO255 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.71 Rx 256- to 511-Byte Packets Statistic Register (ENET_RMON_R_P256TO511)

Address: 218_8000h base + 2B4h offset = 218_82B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P256TO511 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.72 Rx 512- to 1023-Byte Packets Statistic Register (ENET_RMON_R_P512TO1023)

Address: 218_8000h base + 2B8h offset = 218_82B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P512TO1023 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.73 Rx 1024- to 2047-Byte Packets Statistic Register (ENET_RMON_R_P1024TO2047)

Address: 218_8000h base + 2BCh offset = 218_82BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P1024TO2047 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.74 Rx Packets Greater than 2048 Bytes Statistic Register (ENET_RMON_R_P_GTE2048)

Address: 218_8000h base + 2C0h offset = 218_82C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_RMON_R_P_GTE2048 field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Packet count

23.5.75 Rx Octets Statistic Register (ENET_RMON_R_OCTETS)

Address: 218_8000h base + 2C4h offset = 218_82C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COUNT																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

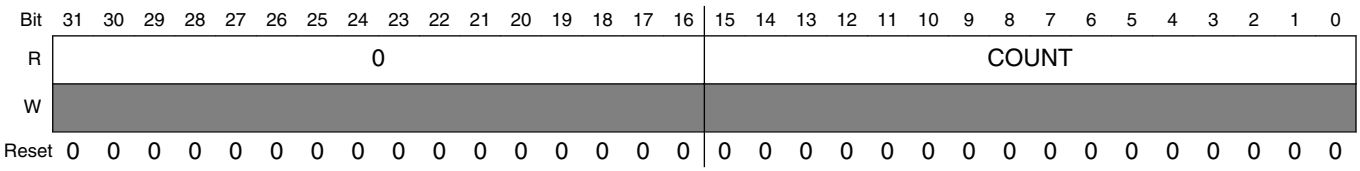
ENET_RMON_R_OCTETS field descriptions

Field	Description
COUNT	Octet count

23.5.76 Frames not Counted Correctly Statistic Register (ENET_IEEE_R_DROP)

Counter increments if a frame with invalid or missing SFD character is detected and has been dropped. None of the other counters increments if this counter increments.

Address: 218_8000h base + 2C8h offset = 218_82C8h

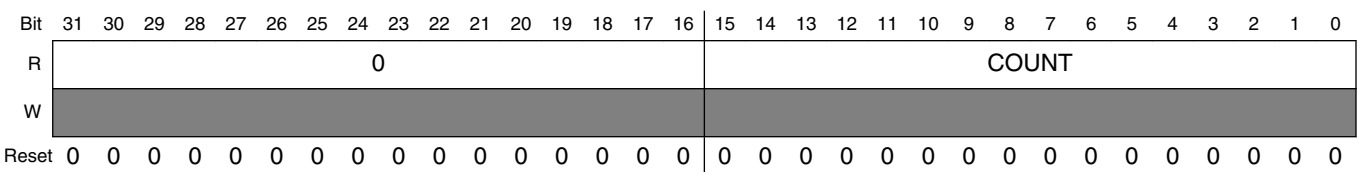


ENET_IEEE_R_DROP field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.77 Frames Received OK Statistic Register (ENET_IEEE_R_FRAME_OK)

Address: 218_8000h base + 2CCh offset = 218_82CCh



ENET_IEEE_R_FRAME_OK field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.78 Frames Received with CRC Error Statistic Register (ENET_IEEE_R_CRC)

Address: 218_8000h base + 2D0h offset = 218_82D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_R_CRC field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.79 Frames Received with Alignment Error Statistic Register (ENET_IEEE_R_ALIGN)

Address: 218_8000h base + 2D4h offset = 218_82D4h

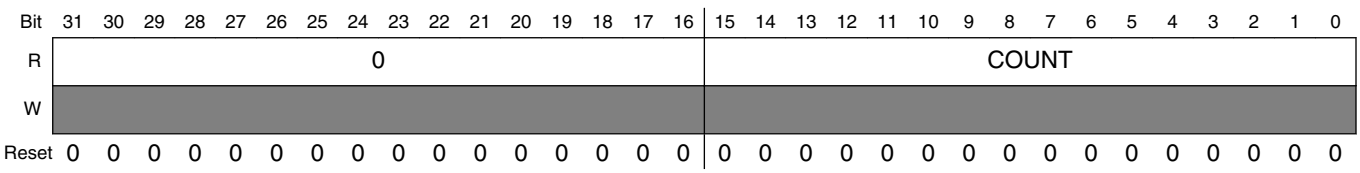
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COUNT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_IEEE_R_ALIGN field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Frame count

23.5.80 Receive FIFO Overflow Count Statistic Register (ENET_IEEE_R_MACERR)

Address: 218_8000h base + 2D8h offset = 218_82D8h

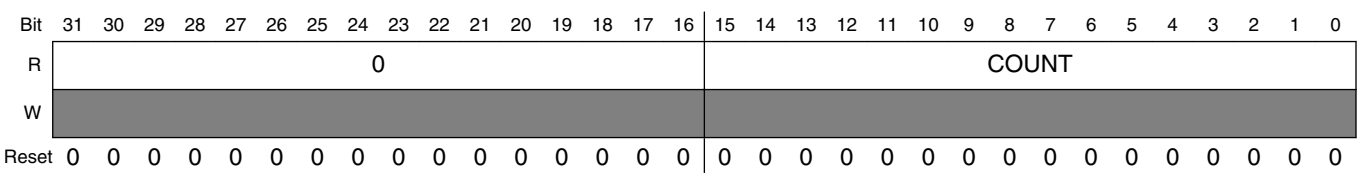


ENET_IEEE_R_MACERR field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Count

23.5.81 Flow Control Pause Frames Received Statistic Register (ENET_IEEE_R_FDXFC)

Address: 218_8000h base + 2DCh offset = 218_82DCh

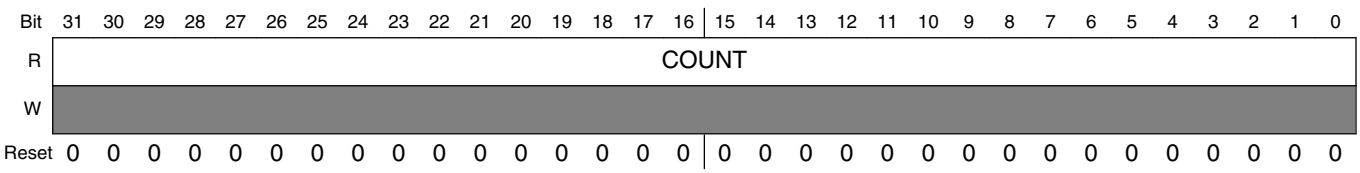


ENET_IEEE_R_FDXFC field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COUNT	Pause frame count

23.5.82 Octet Count for Frames Received without Error Statistic Register (ENET_IEEE_R_OCTETS_OK)

Address: 218_8000h base + 2E0h offset = 218_82E0h



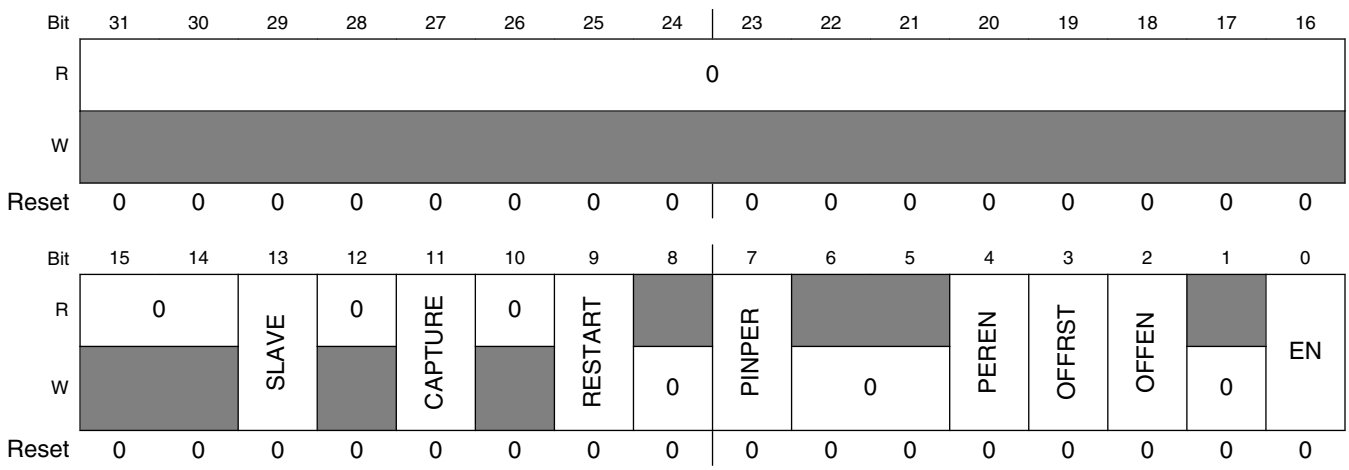
ENET_IEEE_R_OCTETS_OK field descriptions

Field	Description
COUNT	Octet count

23.5.83 Adjustable Timer Control Register (ENET_ATCR)

ATCR command fields can trigger the corresponding events directly. It is not necessary to preserve any of the configuration fields when a command field is set in the register, that is, no read-modify-write is required.

Address: 218_8000h base + 400h offset = 218_8400h



ENET_ATCR field descriptions

Field	Description
31–14 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.

Table continues on the next page...

ENET_ATCR field descriptions (continued)

Field	Description
13 SLAVE	Enable Timer Slave Mode 0 The timer is active and all configuration fields in this register are relevant. 1 The internal timer is disabled and the externally provided timer value is used. All other fields, except CAPTURE, in this register have no effect. CAPTURE can still be used to capture the current timer value.
12 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
11 CAPTURE	Capture Timer Value This field automatically clears to 0 after the command completes. **NOTE:** To ensure that the correct time value is read from the ATVR register, a minimum amount of time must elapse from issuing this command to reading the ATVR register. This minimum time is defined by the greater of either six register clock cycles or six 1588/timestamp clock cycles. 0 No effect. 1 The current time is captured and can be read from the ATVR register.
10 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
9 RESTART	Reset Timer Resets the timer to zero. This has no effect on the counter enable. If the counter is enabled when this field is set, the timer is reset to zero and starts counting from there. When set, all other fields are ignored during a write. This field automatically clears to 0 after the command completes. **NOTE:** The Reset Timer command requires at least 6 clock cycles of either the register clock or the 1588/timestamp clock, whichever is greater, to complete.
8 Reserved	This field is reserved.
7 PINPER	Enables event signal output assertion on period event. **NOTE:** Not all devices contain the event signal output. See the chip configuration details. 0 Disable. 1 Enable.
6–5 Reserved	This field is reserved.
4 PEREN	Enable Periodical Event 0 Disable. 1 A period event interrupt can be generated (EIR[TS_TIMER]) and the event signal output is asserted when the timer wraps around according to the periodic setting ATPER. The timer period value must be set before setting this bit. **NOTE:** Not all devices contain the event signal output. See the chip configuration details.
3 OFFRST	Reset Timer On Offset Event 0 The timer is not affected and no action occurs, besides clearing OFFEN, when the offset is reached. 1 If OFFEN is set, the timer resets to zero when the offset setting is reached. The offset event does not cause a timer interrupt.

Table continues on the next page...

ENET_ATCR field descriptions (continued)

Field	Description
2 OFFEN	Enable One-Shot Offset Event 0 Disable. 1 The timer can be reset to zero when the given offset time is reached (offset event). The field is cleared when the offset event is reached, so no further event occurs until the field is set again. The timer offset value must be set before setting this field.
1 Reserved	This field is reserved.
0 EN	Enable Timer 0 The timer stops at the current value. 1 The timer starts incrementing.

23.5.84 Timer Value Register (ENET_ATVR)

Address: 218_8000h base + 404h offset = 218_8404h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ATIME																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_ATVR field descriptions

Field	Description
ATIME	A write sets the timer. A read returns the last captured value. To read the current value, issue a capture command (set ATCR[CAPTURE]) prior to reading this register.

23.5.85 Timer Offset Register (ENET_ATOFF)

Address: 218_8000h base + 408h offset = 218_8408h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	OFFSET																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_ATOFF field descriptions

Field	Description
OFFSET	Offset value for one-shot event generation. When the timer reaches the value, an event can be generated to reset the counter. If the increment value in ATINC is given in true nanoseconds, this value is also given in true nanoseconds.

23.5.86 Timer Period Register (ENET_ATPER)

Address: 218_8000h base + 40Ch offset = 218_840Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PERIOD																															
Reset	0	0	1	1	1	0	1	1	1	0	0	1	1	0	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0

ENET_ATPER field descriptions

Field	Description
PERIOD	Value for generating periodic events. Each instance the timer reaches this value, the period event occurs and the timer restarts. If the increment value in ATINC is given in true nanoseconds, this value is also given in true nanoseconds. The value should be initialized to 1,000,000,000 (1×10^9) to represent a timer wrap around of one second. The increment value set in ATINC should be set to the true nanoseconds of the period of clock ts_clk, hence implementing a true 1 second counter.

23.5.87 Timer Correction Register (ENET_ATCOR)

Address: 218_8000h base + 410h offset = 218_8410h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	COR														
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

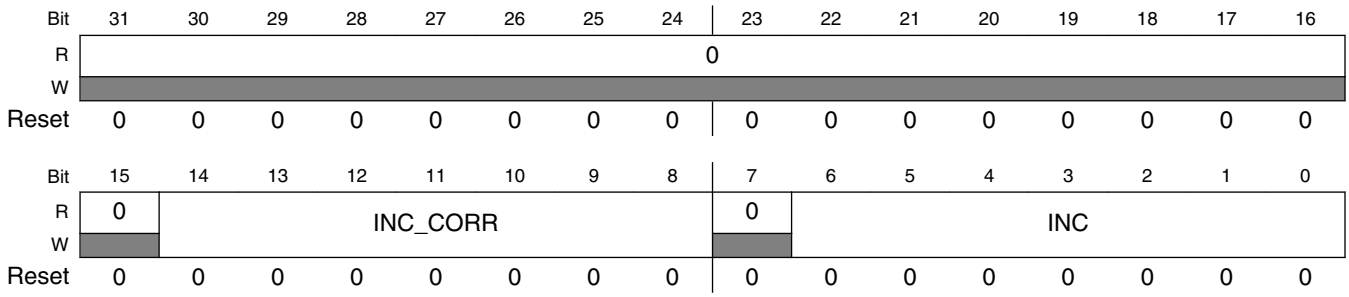
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	COR															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_ATCOR field descriptions

Field	Description
31 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
COR	Correction Counter Wrap-Around Value Defines after how many timer clock cycles (ts_clk) the correction counter should be reset and trigger a correction increment on the timer. The amount of correction is defined in ATINC[INC_CORR]. A value of 0 disables the correction counter and no corrections occur. **NOTE:** This value is given in clock cycles, not in nanoseconds as all other values.

23.5.88 Time-Stamping Clock Period Register (ENET_ATINC)

Address: 218_8000h base + 414h offset = 218_8414h

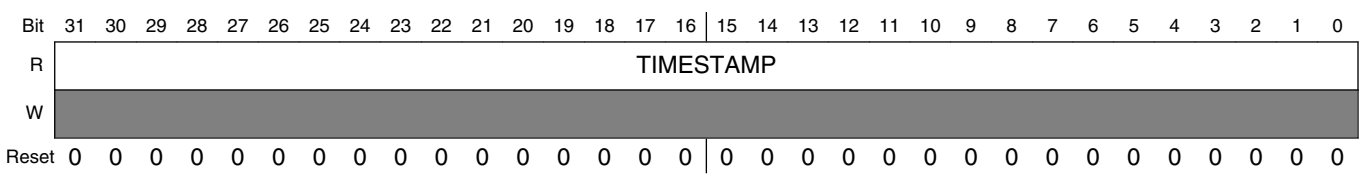


ENET_ATINC field descriptions

Field	Description
31–15 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
14–8 INC_CORR	Correction Increment Value This value is added every time the correction timer expires (every clock cycle given in ATCOR). A value less than INC slows down the timer. A value greater than INC speeds up the timer.
7 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
INC	Clock Period Of The Timestamping Clock (ts_clk) In Nanoseconds The timer increments by this amount each clock cycle. For example, set to 10 for 100 MHz, 8 for 125 MHz, 5 for 200 MHz. **NOTE:** For highest precision, use a value that is an integer fraction of the period set in ATPER.

23.5.89 Timestamp of Last Transmitted Frame (ENET_ATSTMP)

Address: 218_8000h base + 418h offset = 218_8418h



ENET_ATSTMP field descriptions

Field	Description
TIMESTAMP	Timestamp of the last frame transmitted by the core that had TxBD[TS] set . This register is only valid when EIR[TS_AVAIL] is set.

23.5.90 Timer Global Status Register (ENET_TGSR)

Address: 218_8000h base + 604h offset = 218_8604h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

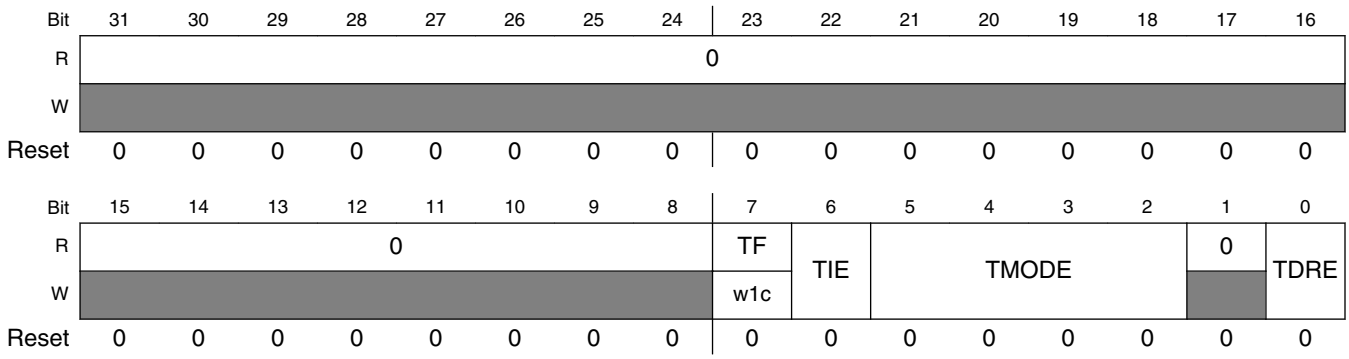
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0												TF3	TF2	TF1	TF0
W													w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_TGSR field descriptions

Field	Description
31–4 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
3 TF3	Copy Of Timer Flag For Channel 3 0 Timer Flag for Channel 3 is clear 1 Timer Flag for Channel 3 is set
2 TF2	Copy Of Timer Flag For Channel 2 0 Timer Flag for Channel 2 is clear 1 Timer Flag for Channel 2 is set
1 TF1	Copy Of Timer Flag For Channel 1 0 Timer Flag for Channel 1 is clear 1 Timer Flag for Channel 1 is set
0 TF0	Copy Of Timer Flag For Channel 0 0 Timer Flag for Channel 0 is clear 1 Timer Flag for Channel 0 is set

23.5.91 Timer Control Status Register (ENET_TCSR*n*)

Address: 218_8000h base + 608h offset + (8d × i), where i=0d to 3d



ENET_TCSR*n* field descriptions

Field	Description
31–8 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
7 TF	Timer Flag Sets when input capture or output compare occurs. This flag is double buffered between the module clock and 1588 clock domains. When this field is 1, it can be cleared to 0 by writing 1 to it. 0 Input Capture or Output Compare has not occurred 1 Input Capture or Output Compare has occurred
6 TIE	Timer Interrupt Enable 0 Interrupt is disabled 1 Interrupt is enabled
5–2 TMODE	Timer Mode Updating the Timer Mode field takes a few cycles to register because it is synchronized to the 1588 clock. The version of Timer Mode returned on a read is from the 1588 clock domain. When changing Timer Mode, always disable the channel and read this register to verify the channel is disabled first. 0000 Timer Channel is disabled. 0001 Timer Channel is configured for Input Capture on rising edge 0010 Timer Channel is configured for Input Capture on falling edge 0011 Timer Channel is configured for Input Capture on both edges 0100 Timer Channel is configured for Output Compare - software only 0101 Timer Channel is configured for Output Compare - toggle output on compare 0110 Timer Channel is configured for Output Compare - clear output on compare 0111 Timer Channel is configured for Output Compare - set output on compare 1000 Reserved 1010 Timer Channel is configured for Output Compare - clear output on compare, set output on overflow 10x1 Timer Channel is configured for Output Compare - set output on compare, clear output on overflow 1100 Reserved

Table continues on the next page...

ENET_TCSR*n* field descriptions (continued)

Field	Description
	1110 Timer Channel is configured for Output Compare - pulse output low on compare for one 1588 clock cycle 1111 Timer Channel is configured for Output Compare - pulse output high on compare for one 1588 clock cycle
1 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
0 TDRE	Timer DMA Request Enable 0 DMA request is disabled 1 DMA request is enabled

23.5.92 Timer Compare Capture Register (ENET_TCCR*n*)

Address: 218_8000h base + 60Ch offset + (8d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	TCC																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ENET_TCCR*n* field descriptions

Field	Description
TCC	Timer Capture Compare This register is double buffered between the module clock and 1588 clock domains. When configured for compare, the 1588 clock domain updates with the value in the module clock domain whenever the Timer Channel is first enabled and on each subsequent compare. Write to this register with the first compare value before enabling the Timer Channel. When the Timer Channel is enabled, write the second compare value either immediately, or at least before the first compare occurs. After each compare, write the next compare value before the previous compare occurs and before clearing the Timer Flag. The compare occurs one 1588 clock cycle after the IEEE 1588 Counter increments past the compare value in the 1588 clock domain. If the compare value is less than the value of the 1588 Counter when the Timer Channel is first enabled, then the compare does not occur until following the next overflow of the 1588 Counter. If the compare value is greater than the IEEE 1588 Counter when the 1588 Counter overflows, or the compare value is less than the value of the IEEE 1588 Counter after the overflow, then the compare occurs one 1588 clock cycle following the overflow. When configured for Capture, the value of the IEEE 1588 Counter is captured into the 1588 clock domain and then updated into the module clock domain, provided the Timer Flag is clear. Always read the capture value before clearing the Timer Flag.

23.6 Functional description

This section provides a complete functional description of the MAC-NET core.

23.6.1 Ethernet MAC frame formats

The IEEE 802.3 standard defines the Ethernet frame format as follows:

- Minimum length of 64 bytes
- Maximum length of 1518 bytes excluding the preamble and the start frame delimiter (SFD) bytes

An Ethernet frame consists of the following fields:

- Seven bytes preamble
- Start frame delimiter (SFD)
- Two address fields
- Length or type field
- Data field
- Frame check sequence (CRC value)
- Extension field is defined only for Gigabit Ethernet half-duplex implementations and is not supported by the MAC core

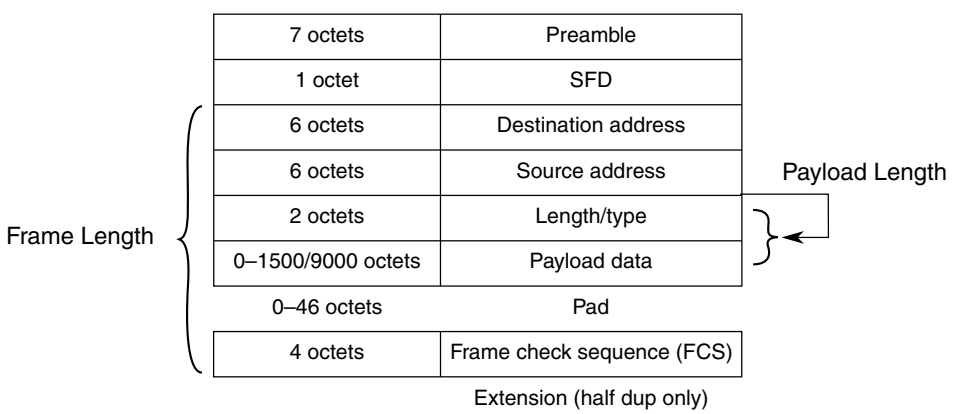


Figure 23-102. MAC frame format overview

Optionally, MAC frames can be VLAN-tagged with an additional four-byte field inserted between the MAC source address and the type/length field. VLAN tagging is defined by the IEEE P802.1q specification. VLAN-tagged frames have a maximum length of 1522 bytes, excluding the preamble and the SFD bytes.

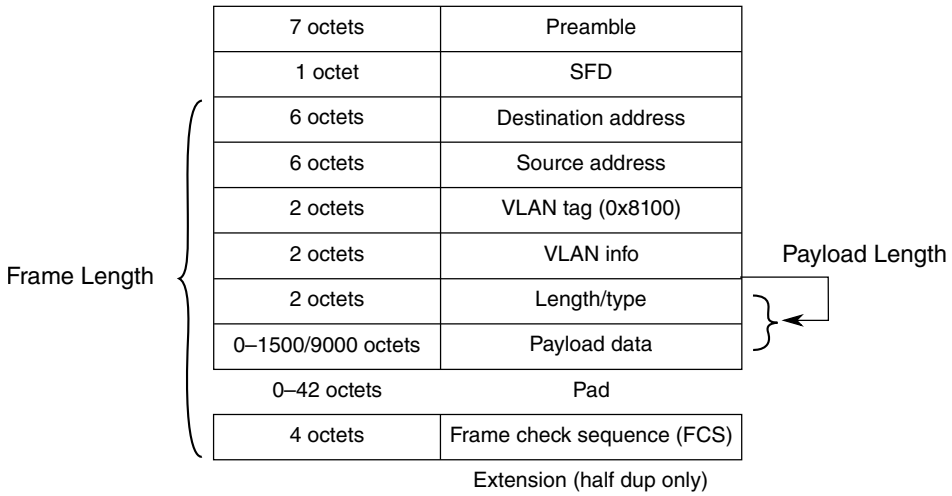


Figure 23-103. VLAN-tagged MAC frame format overview

Table 23-106. MAC frame definition

Term	Description
Frame length	Defines the length, in octets, of the complete frame without preamble and SFD. A frame has a valid length if it contains at least 64 octets and does not exceed the programmed maximum length.
Payload length	The length/type field indicates the length of the frame's payload section. The most significant byte is sent/received first. • If the length/type field is set to a value less than 46, the payload is padded so that the minimum frame length requirement (64 bytes) is met. For VLAN-tagged frames, a value less than 42 indicates a padded frame. • If the length/type field is set to a value larger than the programmed frame maximum length (e.g. 1518) it is interpreted as a type field.
Destination and source address	48-bit MAC addresses. The least significant byte is sent/received first and the first two least significant bits of the MAC address distinguish MAC frames, as detailed in MAC address check.

Note

Although the IEEE specification defines a maximum frame length, the MAC core provides the flexibility to program any value for the frame maximum length.

23.6.1.1 Pause Frames

The receiving device generates a pause frame to indicate a congestion to the emitting device, which should stop sending data.

Pause frames are indicated by the length/type set to 0x8808. The two first bytes of a pause frame following the type, defines a 16-bit opcode field set to 0x0001 always. A 16-bit pause quanta is defined in the frame payload bytes 2 (P1) and 3 (P2) as defined in the following table. The P1 pause quanta byte is the most significant.

Table 23-107. Pause Frame Format (Values in Hex)

1	2	3	4	5	6	7	8	9	10	11	12	13	14
55	55	55	55	55	55	55	D5	01	80	C2	00	00	01
Preamble							SFD	Multicast Destination Address					
15	16	17	18	19	20	21	22	23	24	25	26	27 –68	
00	00	00	00	00	00	88	08	00	01	hi	lo	00	
Source Address						Type		Opcode		P1	P2	pad (42)	
69	70	71	72										
26	6B	AE	0A										
CRC-32													

There is no payload length field found within a pause frame and a pause frame is always padded with 42 bytes (0x00).

If a pause frame with a pause value greater than zero (XOFF condition) is received, the MAC stops transmitting data as soon the current frame transfer is completed. The MAC stops transmitting data for the value defined in pause quanta. One pause quanta fraction refers to 512 bit times.

If a pause frame with a pause value of zero (XON condition) is received, the transmitter is allowed to send data immediately (see Full-duplex flow control operation for details).

23.6.1.2 Magic packets

A magic packet is a unicast, multicast, or broadcast packet, which carries a defined sequence in the payload section.

Magic packets are received and inspected only under specific conditions as described in Magic packet detection.

The defined sequence to decode a magic packet is formed with a synchronization stream which consists of six consecutive 0xFF bytes, and is followed by sequence of sixteen consecutive unicast MAC addresses of the node to be awakened.

This sequence can be located anywhere in the magic packet payload. The magic packet is formed with a standard Ethernet header, optional padding, and CRC.

23.6.2 IP and higher layers frame format

The following sections use the term datagram to describe the protocol specific data unit that is found within the payload section of its container entity.

For example, an IP datagram specifies the payload section of an Ethernet frame. A TCP datagram specifies the payload section within an IP datagram.

23.6.2.1 Ethernet types

IP datagrams are carried in the payload section of an Ethernet frame. The Ethernet frame type/length field discriminates several datagram types.

The following table lists the types of interest:

Table 23-108. Ethernet type value examples

Type	Description
0x8100	VLAN-tagged frame. The actual type is found 4 octets later in the frame.
0x0800	IPv4
0x0806	ARP
0x86DD	IPv6

23.6.2.2 IPv4 datagram format

The following figure shows the IP Version 4 (IPv4) header, which is located at the beginning of an IP datagram. It is organized in 32-bit words. The first byte sent/received is the leftmost byte of the first word (in other words, version/IHL field).

The IP header can contain further options, which are always padded if necessary to guarantee the payload following the header is aligned to a 32-bit boundary.

The IP header is immediately followed by the payload, which can contain further protocol headers (for example, TCP or UDP, as indicated by the protocol field value). The complete IP datagram is transported in the payload section of an Ethernet frame.

Table 23-109. IPv4 header format

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Version	IHL	TOS		Length			
Fragment ID				Flags	Fragment offset		
TTL		Protocol		Header checksum			
Source address							
Destination address							
Options							

Table 23-110. IPv4 header fields

Field name	Description
Version	4-bit IP version information. 0x4 for IPv4 frames.
IHL	4-bit Internet header length information. Determines number of 32-bit words found within the IP header. If no options are present, the default value is 0x5.
TOS	Type of service/DiffServ field.
Length	Total length of the datagram in bytes, including all octets of header and payload.
Fragment ID, flags, fragment offset	Fields used for IP fragmentation.
TTL	Time-to-live. In effect, is decremented at each router arrival. If zero, datagram must be discarded.
Protocol	Identifier of protocol that follows in the datagram.
Header checksum	Checksum of IP header. For computational purposes, this field's value is zero.
Source address	Source IP address.
Destination address	Destination IP address.

23.6.2.3 IPv6 datagram format

The following figure shows the IP version 6 (IPv6) header, which is located at the beginning of an IP datagram. It is organized in 32-bit words and has a fixed length of ten words (40 bytes). The next header field identifies the type of the header that follows the IPv6 header. It is defined similar to the protocol identifier within IPv4, with new definitions for identifying extension headers. These headers can be inserted between the IPv6 header and the protocol header, which will shift the protocol header accordingly.The accelerator currently only supports IPv6 without extension headers (in other words, the next header specifies TCP, UDP, or IMCP).

The first byte sent/received is the leftmost byte of the first word (in other words, version/ traffic class fields).

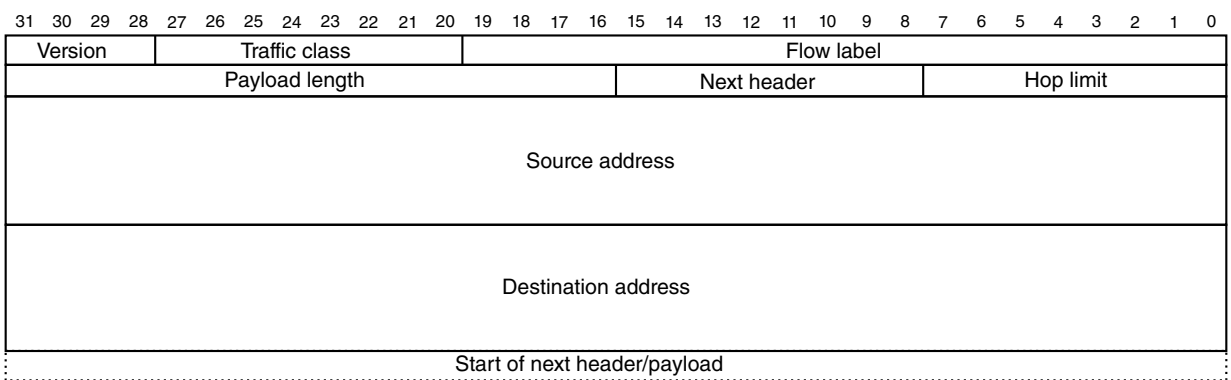


Figure 23-104. IPv6 header format

Table 23-111. IPv6 header fields

Field name	Description
Version	4-bit IP version information. 0x6 for all IPv6 frames.
Traffic class	8-bit field defining the traffic class.
Flow label	20-bit flow label identifying frames of the same flow.
Payload length	16-bit length of the datagram payload in bytes. It includes all octets following the IPv6 header.
Next header	Identifies the header that follows the IPv6 header. This can be the protocol header or any IPv6 defined extension header.
Hop limit	Hop counter, decremented by one by each station that forwards the frame. If hop limit is 0 the frame must be discarded.
Source address	128-bit IPv6 source address.
Destination address	128-bit IPv6 destination address.

23.6.2.4 Internet Control Message Protocol (ICMP) datagram format

An internet control message protocol (ICMP) is found following the IP header, if the protocol identifier is 1. The ICMP datagram has a four-octet header followed by additional message data.

Table 23-112. ICMP header format

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Type		Code		Checksum			
ICMP message data							

Table 23-113. IP header fields

Field name	Description
Type	8-bit type information
Code	8-bit code that is related to the message type
Checksum	16-bit one's complement checksum over the complete ICMP datagram

23.6.2.5 User Datagram Protocol (UDP) datagram format

A user datagram protocol header is found after the IP header, when the protocol identifier is 17.

The payload of the datagram is after the UDP header. The header byte order follows the conventions given for the IP header above.

Table 23-114. UDP header format

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Source port				Destination port			
Length				Checksum			

Table 23-115. UDP header fields

Field name	Description
Source port	Source application port
Destination port	Destination application port
Length	Length of user data which immediately follows the header, including the UDP header (that is, minimum value is 8)
Checksum	Checksum over the complete datagram and some IP header information

23.6.2.6 TCP datagram format

A TCP header is found following the IP header, when the protocol identifier has a value of 6.

The TCP payload immediately follows the TCP header.

Table 23-116. TCP header format

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Source port				Destination port			
Sequence number							
Acknowledgement number							
Offset	Reserved		Flags	Window			
Checksum				Urgent pointer			
Options							

Table 23-117. TCP header fields

Field name	Description
Source port	Source application port
Destination port	Destination application port
Sequence number	Transmit sequence number
Ack. number	Receive sequence number
Offset	Data offset, which is number of 32-bit words within TCP header — if no options selected, defaults to value of 5

Table continues on the next page...

Table 23-117. TCP header fields (continued)

Field name	Description
Flags	URG, ACK, PSH, RST, SYN, FIN flags
Window	TCP receive window size information
Checksum	Checksum over the complete datagram (TCP header and data) and IP header information
Options	Additional 32-bit words for protocol options

23.6.3 IEEE 1588 message formats

The following sections describe the IEEE 1588 message formats.

23.6.3.1 Transport encapsulation

The precision time protocol (PTP) datagrams are encapsulated in Ethernet frames using the UDP/IP transport mechanism, or optionally, with the newer 1588v2 directly in Ethernet frames (layer 2).

Typically, multicast addresses are used to allow efficient distribution of the synchronization messages.

23.6.3.1.1 UDP/IP

The 1588 messages (v1 and v2) can be transported using UDP/IP multicast messages.

Table 23-118 shows IP multicast groups defined for PTP. The table also shows their respective MAC layer multicast address mapping according to RFC 1112 (last three octets of IP follow the fixed value of 01-00-5E).

Table 23-118. UDP/IP multicast domains

Name	IP Address	MAC Address mapping
DefaultPTPdomain	224.0.1.129	01-00-5E-00-01-81
AlternatePTPdomain1	224.0.1.130	01-00-5E-00-01-82
AlternatePTPdomain2	224.0.1.131	01-00-5E-00-01-83
AlternatePTPdomain3	224.0.1.132	01-00-5E-00-01-84

Table 23-119. UDP port numbers

Message type	UDP port	Note
Event	319	Used for SYNC and DELAY_REQUEST messages
General	320	All other messages (for example, follow-up, delay-response)

23.6.3.1.2 Native Ethernet (PTPv2)

In addition to using UDP/IP frames, IEEE 1588v2 defines a native Ethernet frame format that uses ethertype = 0x88F7. The payload of the Ethernet frame immediately contains the PTP datagram, starting with the PTPv2 header.

Besides others, version 2 adds a peer delay mechanism to allow delay measurements between individual point-to-point links along a path over multiple nodes. The following multicast domains are also defined in PTPv2.

Table 23-120. PTPv2 multicast domains

Name	MAC address
Normal messages	01-1B-19-00-00-00
Peer delay messages	01-80-C2-00-00-0E

23.6.3.2 PTP header

All PTP frames contain a common header that determines the protocol version and the type of message, which defines the remaining content of the message.

All multi-octet fields are transmitted in big-endian order (the most significant byte is transmitted/received first).

The last four bits of versionPTP are at the same position (second byte) for PTPv1 and PTPv2 headers. This allows accurate identification by inspecting the first two bytes of the message.

23.6.3.2.1 PTPv1 header

Table 23-121. Common PTPv1 message header

Offset	Octets	Bits							
		7	6	5	4	3	2	1	0
0	2	versionPTP = 0x0001							
2	2	versionNetwork							
4	16	subdomain							
20	1	messageType							
21	1	sourceCommunicationTechnology							
22	6	sourceUuid							
28	2	sourcePortId							
30	2	sequenceId							

Table continues on the next page...

Table 23-121. Common PTPv1 message header (continued)

Offset	Octets	Bits							
		7	6	5	4	3	2	1	0
32	1	control							
33	1	0x00							
34	2	flags							
36	4	reserved							

The type of message is encoded in the messageType and control fields as shown in Table 23-122 :

Table 23-122. PTPv1 message type identification

messageType	control	Message Name	Message
0x01	0x0	SYNC	Event message
0x01	0x1	DELAY_REQ	Event message
0x02	0x2	FOLLOW_UP	General message
0x02	0x3	DELAY_RESP	General message
0x02	0x4	MANAGEMENT	General message
other	other	—	Reserved

The field sequenceId is used to non-ambiguously identify a message.

23.6.3.2.2 PTPv2 header

Table 23-123. Common PTPv2 message header

Offset	Octets	Bits							
		7	6	5	4	3	2	1	0
0	1	transportSpecific				messageId			
1	1	reserved				versionPTP = 0x2			
2	2	messageLength							
4	1	domainNumber							
5	1	reserved							
6	2	flags							
8	8	correctionField							
16	4	reserved							
20	10	sourcePortIdentity							
30	2	sequenceId							
32	1	control							
33	1	logMeanMessageInterval							

The type of message is encoded in the field messageId as follows:

Table 23-124. PTPv2 message type identification

messageId	Message name	Message
0x0	SYNC	Event message
0x1	DELAY_REQ	Event message
0x2	PATH_DELAY_REQ	Event message
0x3	PATH_DELAY_RESP	Event message
0x4–0x7	—	Reserved
0x8	FOLLOW_UP	General message
0x9	DELAY_RESP	General message
0xa	PATH_DELAY_FOLLOW_UP	General message
0xb	ANNOUNCE	General message
0xc	SIGNALING	General message
0xd	MANAGEMENT	General message

The PTPv2 flags field contains further details on the type of message, especially if one-step or two-step implementations are used. The one- or two-step implementation is controlled by the TWO_STEP bit in the first octet of the flags field as shown below. Reserved bits are cleared.

Table 23-125. PTPv2 message flags field definitions

Bit	Name	Description
0	ALTERNATE_MASTER	See IEEE 1588 Clause 17.4
1	TWO_STEP	1 Two-step clock 0 One-step clock
2	UNICAST	1 Transport layer address uses a unicast destination address 0 Multicast is used
3	—	Reserved
4	—	Reserved
5	Profile specific	
6	Profile specific	
7	—	Reserved

23.6.4 MAC receive

The MAC receive engine performs the following tasks:

- Check frame framing

- Remove frame preamble and frame SFD field
- Discard frame based on frame destination address field
- Terminate pause frames
- Check frame length
- Remove payload padding if it exists
- Calculate and verify CRC-32
- Write received frames in the core receive FIFO

If the MAC is programmed to operate in half-duplex mode, it will also check if the frame is received with a collision.

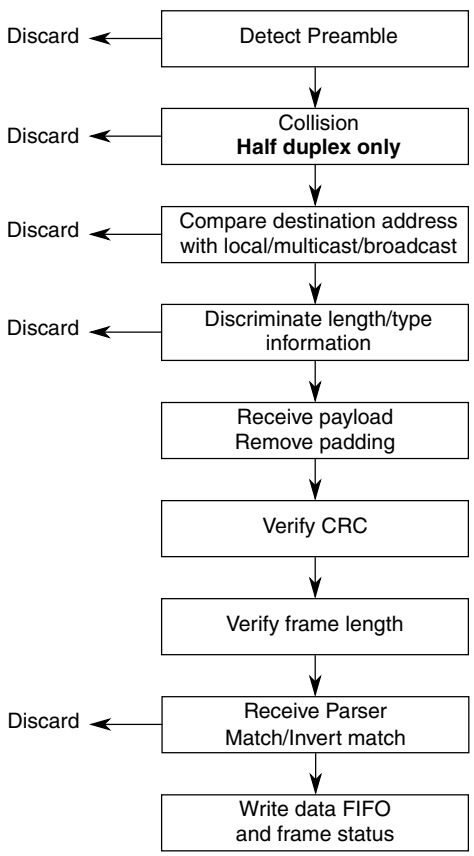


Figure 23-105. MAC receive flow

23.6.4.1 Collision detection in half-duplex mode

If the packet is received with a collision detected during reception of the first 64 bytes, the packet is discarded (if frame size was less than ~14 octets) or transmitted to the user application with an error and RxBD[CE] set.

23.6.4.2 Preamble processing

The IEEE 802.3 standard allows a maximum size of 56 bits (seven bytes) for the preamble, while the MAC core allows any preamble length, including zero length preamble.

The MAC core checks for the start frame delimiter (SFD) byte. If the next byte of the preamble, which is different from 0x55, is not 0xD5, the frame is discarded.

Although the IEEE specification dictates that the inner-packet gap should be at least 96 bits, the MAC core is designed to accept frames separated by only 64 10/100 Mbps operation (MII) bits.

The MAC core removes the preamble and SFD bytes.

23.6.4.3 MAC address check

The destination address bit 0 differentiates between multicast and unicast addresses.

- If bit 0 is 0, the MAC address is an individual (unicast) address.
- If bit 0 is 1, the MAC address defines a group (multicast) address.
- If all 48 bits of the MAC address are set, it indicates a broadcast address.

23.6.4.3.1 Unicast address check

If a unicast address is received, the destination MAC address is compared to the node MAC address programmed by the host in the PADDR1/2 registers.

If the destination address matches any of the programmed MAC addresses, the frame is accepted.

If Promiscuous mode is enabled (RCR[PROM] = 1) no address checking is performed and all unicast frames are accepted.

23.6.4.3.2 Multicast and unicast address resolution

The hash table algorithm used in the group and individual hash filtering operates as follows.

- The 48-bit destination address is mapped into one of 64 bits, represented by 64 bits in ENET*n*_GAUR/GALR (group address hash match) or ENET*n*_IAUR/IALR (individual address hash match).
- This mapping is performed by passing the 48-bit address through the on-chip 32-bit CRC generator and selecting the six most significant bits of the CRC-encoded result to generate a number between 0 and 63.
- The msb of the CRC result selects ENET*n*_GAUR (msb = 1) or ENET*n*_GALR (msb = 0).
- The five lsbs of the hash result select the bit within the selected register.
- If the CRC generator selects a bit set in the hash table, the frame is accepted; else, it is rejected.

For example, if eight group addresses are stored in the hash table and random group addresses are received, the hash table prevents roughly 56/64 (or 87.5%) of the group address frames from reaching memory. Those that do reach memory must be further filtered by the processor to determine if they truly contain one of the eight desired addresses.

The effectiveness of the hash table declines as the number of addresses increases.

The user must initialize the hash table registers. Use this CRC32 polynomial to compute the hash:

- $FCS(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x^{1} + 1$

If Promiscuous mode is enabled (ENET*n*_RCR[PROM] = 1) all unicast and multicast frames are accepted regardless of ENET*n*_GAUR/GALR and ENET*n*_IAUR/IALR settings.

23.6.4.3.3 Broadcast address reject

All broadcast frames are accepted if BC_REJ is cleared or ENET*n*_RCR[PROM] is set. If PROM is cleared when ENET*n*_RCR[BC_REJ] is set, all broadcast frames are rejected.

Table 23-126. Broadcast address reject programming

PROM	BC_REJ	Broadcast frames
0	0	Accepted
0	1	Rejected

Table continues on the next page...

Table 23-126. Broadcast address reject programming (continued)

PROM	BC_REJ	Broadcast frames
1	0	Accepted
1	1	Accepted

23.6.4.3.4 Miss-bit implementation

For higher layer filtering purposes, RxBD[M] indicates an address miss when the MAC operates in promiscuous mode and accepts a frame that would otherwise be rejected.

If a group/individual hash or exact match does not occur and Promiscuous mode is enabled (RCR[PROM] = 1), the frame is accepted and the M bit is set in the buffer descriptor; otherwise, the frame is rejected.

This means the status bit is set in any of the following conditions during Promiscuous mode:

- A broadcast frame is received when BC_REJ is set
- A unicast is received that does not match either:
  - Node address (PALR[PADDR1] and PAUR[PADDR2])
  - Hash table for unicast (IAUR[IADDR1] and IALR[IADDR2])
- A multicast is received that does not match the GAUR[GADDR1] and GALR[GADDR2] hash table entries

23.6.4.4 Frame length/type verification: payload length check

If the length/type is less than 0x600 and NLC is set, the MAC checks the payload length and reports any error in the frame status word and interrupt bit PLR.

If the length/type is greater than or equal to 0x600, the MAC interprets the field as a type and no payload length check is performed.

The length check is performed on VLAN and stacked VLAN frames. If a padded frame is received, no length check can be performed due to the extended frame payload because padded frames can never have a payload length error.

23.6.4.5 Frame length/type verification: frame length check

When the receive frame length exceeds MAX_FL bytes, the BABR interrupt is generated and the RxBD[LG] bit is set.

The frame is not truncated unless the frame length exceeds the value programmed in ENET*n*_FTRL[TRUNC_FL]. If the frame is truncated, RxBD[TR] is set. In addition, a truncated frame always has the CRC error indication set (RxBD[CR]).

23.6.4.6 VLAN frames processing

VLAN frames have a length/type field set to 0x8100 immediately followed by a 16-Bit VLAN control information field.

VLAN-tagged frames are received as normal frames because the VLAN tag is not interpreted by the MAC function, and are pushed complete with the VLAN tag to the user application. If the length/type field of the VLAN-tagged frame, which is found four octets later in the frame, is less than 42, the padding is removed. In addition, the frame status word (RxBD[NO]) indicates that the current frame is VLAN tagged.

23.6.4.7 Pause frame termination

The receive engine terminates pause frames and does not transfer them to the receive FIFO. The quanta is extracted and sent to the MAC transmit path via a small internal clock rate decoupling asynchronous FIFO.

The quanta is written only if a correct CRC and frame length are detected by the control state machine. If not, the quanta is discarded and the MAC transmit path is not paused.

Good pause frames are ignored if ENET*n*_RCR[FCE] is cleared and are forwarded to the client interface when ENET*n*_RCR[PAUFWD] is set.

23.6.4.8 CRC check

The CRC-32 field is checked and forwarded to the core FIFO interface if ENET*n*_RCR[CRCFWD] is cleared and ENET*n*_RCR[PADEN] is set.

When CRCFWD is set (regardless of PADEN), the CRC-32 field is checked and terminated (not transmitted to the FIFO).

The CRC polynomial, as specified in the 802.3 standard, is:

- $FCS(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$

The 32 bits of the CRC value are placed in the frame check sequence (FCS) field with the x^{31} term as right-most bit of the first octet. The CRC bits are thus received in the following order: x^{31}, x^{30},..., x^1, x^0.

If a CRC error is detected, the frame is marked invalid and RxBD[CR] is set.

23.6.4.9 Frame padding removal

When a frame is received with a payload length field set to less than 46 (42 for VLAN-tagged frames and 38 for frames with stacked VLANs), the zero padding can be removed before the frame is written into the data FIFO depending on the setting of ENET*n*_RCR[PADEN].

Note

If a frame is received with excess padding (in other words, the length field is set as mentioned above, but the frame has more than 64 octets) and padding removal is enabled, then the padding is removed as normal and no error is reported if the frame is otherwise correct (for example: good CRC, less than maximum length, and no other error).

23.6.5 MAC transmit

Frame transmission starts when the transmit FIFO holds enough data.

After a transfer starts, the MAC transmit function performs the following tasks:

- Generates preamble and SFD field before frame transmission
- Generates XOFF pause frames if the receive FIFO reports a congestion or if ENET*n*_TCR[TFC_PAUSE] is set with ENET*n*_OPD[PAUSE_DUR] set to a non-zero value
- Generates XON pause frames if the receive FIFO congestion condition is cleared or if TFC_PAUSE is set with PAUSE_DUR cleared
- Suspends Ethernet frame transfer (XOFF) if a non-zero pause quanta is received from the MAC receive path
- Adds padding to the frame if required

- Calculates and appends CRC-32 to the transmitted frame
- Sends the frame with correct inter-packet gap (IPG) (deferring)

When the MAC is configured to operate in half-duplex mode, the following additional tasks are performed:

- Collision detection
- Frame retransmit after back-off timer expires

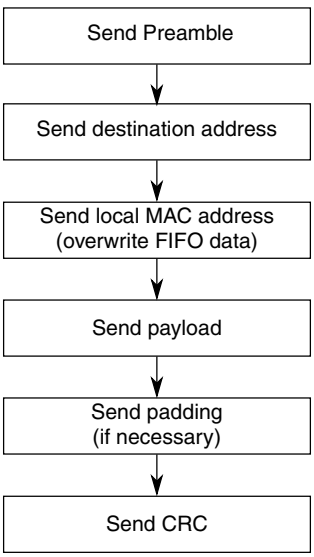


Figure 23-106. Frame transmit overview

23.6.5.1 Frame payload padding

The IEEE specification defines a minimum frame length of 64 bytes.

If the frame sent to the MAC from the user application has a size smaller than 60 bytes, the MAC automatically adds padding bytes (0x00) to comply with the Ethernet minimum frame length specification. Transmit padding is always performed and cannot be disabled.

If the MAC is not allowed to append a CRC (TxBD[TC] = 1), the user application is responsible for providing frames with a minimum length of 64 octets.

23.6.5.2 MAC address insertion

On each frame received from the core transmit FIFO interface, the source MAC address is either:

- Replaced by the address programmed in the PADDR1/2 fields (ENET*n*_TCR[ADDINS] = 1)
- Transparently forwarded to the Ethernet line (ENET*n*_TCR[ADDINS] = 0)

23.6.5.3 CRC-32 generation

The CRC-32 field is optionally generated and appended at the end of a frame.

The CRC polynomial, as specified in the 802.3 standard, is:

- $FCS(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x^{1} + 1$

The 32 bits of the CRC value are placed in the FCS field so that the x^{31} term is the right-most bit of the first octet. The CRC bits are thus transmitted in the following order: x^{31}, x^{30},..., x^{1}, x^{0}.

23.6.5.4 Inter-packet gap (IPG)

In full-duplex mode, after frame transmission and before transmission of a new frame, an IPG (programmed in ENET*n*_TIPG) is maintained. The minimum IPG can be programmed between 8 and 27 byte-times (64 and 216 bit-times).

In half-duplex mode, the core constantly monitors the line. Actual transmission of the data onto the network occurs only if it has been idle for a 96-bit time period, and any back-off time requirements have been satisfied. In accordance with the standard, the core begins to measure the IPG from CRS/GMII_CRS de-assertion.

23.6.5.5 Collision detection and handling — half-duplex operation only

A collision occurs on a half-duplex network when concurrent transmissions from two or more nodes take place. During transmission, the core monitors the line condition and detects a collision when the PHY device asserts COL.

When the core detects a collision while transmitting, it stops transmission of the data and transmits a 32-bit jam pattern. If the collision is detected during the preamble or the SFD transmission, the jam pattern is transmitted after completing the SFD, which results in a minimum 96-bit fragment. The jam pattern is a fixed pattern that is not compared to the actual frame CRC, and has a very low probability (0.532) of having a jam pattern identical to the CRC.

If a collision occurs before transmission of 64 bytes (including preamble and SFD), the MAC core waits for the backoff period and retransmits the packet data (stored in a 64-byte re-transmit buffer) that has already been sent on the line. The backoff period is generated from a pseudo-random process (truncated binary exponential backoff).

If a collision occurs after transmission of 64 bytes (including preamble and SFD), the MAC discards the remainder of the frame, optionally sets the LC interrupt bit, and sets TxBD[LCE].

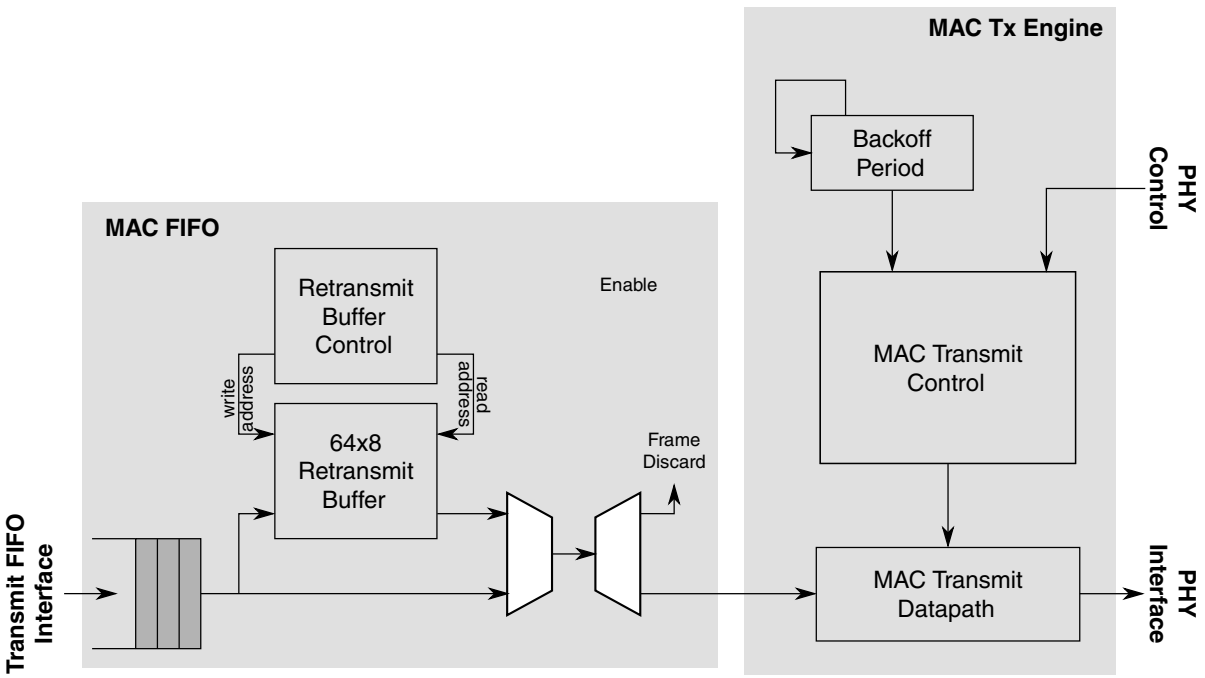


Figure 23-107. Packet re-transmit overview

The backoff time is represented by an integer multiple of slot times. One slot is equal to a 512-bit time period. The number of the delay slot times, before the n^{th} re-transmission attempt, is chosen as a uniformly-distributed random integer in the range:

- $0 < r < 2^k$
- $k = \min(n, N)$; where n is the number of retransmissions and $N = 10$

For example, after the first collision, the backoff period is 0 or 1 slot time. If a collision occurs on the first retransmission, the backoff period is 0, 1, 2, or 3, and so on.

The maximum backoff time (in 512-bit time slots) is limited by N = 10 as specified in the IEEE 802.3 standard.

If a collision occurs after 16 consecutive retransmissions, the core reports an excessive collision condition (ENET*n*_EIR[RL] interrupt field and TxBD[EE]) and discards the current packet from the FIFO.

In networks violating the standard requirements, a collision may occur after transmission of the first 64 bytes. In this case, the core stops the current packet transmission and discards the rest of the packet from the transmit FIFO. The core resumes transmission with the next packet available in the core transmit FIFO.

23.6.6 Full-duplex flow control operation

Three conditions are handled by the core's flow control engine:

- Remote device congestion — The remote device connected to the same Ethernet segment as the core reports congestion and requests that the core stop sending data.
- Core FIFO congestion — When the core's receive FIFO reaches a user-programmable threshold (RX section empty), the core sends a pause frame back to the remote device requesting the data transfer to stop.
- Local device congestion — Any device connected to the core can request (typically, via the host processor) the remote device to stop transmitting data.

23.6.6.1 Remote device congestion

When the MAC transmit control gets a valid pause quanta from the receive path and if ENET*n*_RCR[FCE] is set, the MAC transmit logic:

- Completes the transfer of the current frame.
- Stops sending data for the amount of time specified by the pause quanta in 512 bit time increments.
- Sets ENET*n*_TCR[RFC_PAUSE].

Frame transfer resumes when the time specified by the quanta expires and if no new quanta value is received, or if a new pause frame with a quanta value set to 0x0000 is received. The MAC also resets RFC_PAUSE to zero.

If ENET*n*_RCR[FCE] cleared, the MAC ignores received pause frames.

Optionally and independent of ENET*n*_RCR[FCE], pause frames are forwarded to the client interface if PAUFWD is set.

23.6.6.2 Local device/FIFO congestion

The MAC transmit engine generates pause frames when the local receive FIFO is not able to receive more than a pre-defined number of words (FIFO programmable threshold) or when pause frame generation is requested by the local host processor.

- To generate a pause frame, the host processor sets ENET*n*_TCR[TFC_PAUSE]. A single pause frame is generated when the current frame transfer is completed and TFC_PAUSE is automatically cleared. Optionally, an interrupt is generated.
- An XOFF pause frame is generated when the receive FIFO asserts its section empty flag (internal). An XOFF pause frame is generated automatically, when the current frame transfer completes.
- An XON pause frame is generated when the receive FIFO deasserts its section empty flag (internal). An XON pause frame is generated automatically, when the current frame transfer completes.

When an XOFF pause frame is generated, the pause quanta (payload byte P1 and P2) is filled with the value programmed in ENET*n*_OPD[PAUSE_DUR].

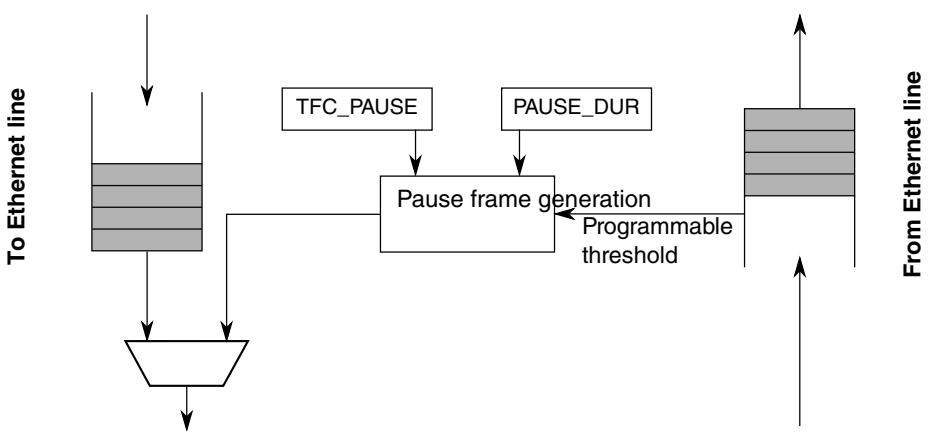


Figure 23-108. Pause frame generation overview

Note

Although the flow control mechanism should prevent any FIFO overflow on the MAC core receive path, the core receive FIFO is protected. When an overflow is detected on the receive FIFO,

the current frame is truncated with an error indication set in the frame status word. The frame should subsequently be discarded by the user application.

23.6.7 Magic packet detection

Magic packet detection wakes a node that is put in power-down mode by the node management agent. Magic packet detection is supported only if the MAC is configured in sleep mode.

23.6.7.1 Sleep mode

To put the MAC in Sleep mode, set ENET*n*_ECR[SLEEP]. At the same time ENET*n*_ECR[MAGICEN] should also be set to enable magic packet detection.

In addition, when the processor is in Stop mode, Sleep mode is entered, without affecting the ENET*n*_ECR register bits.

When the MAC is in Sleep mode:

- The transmit logic is disabled.
- The FIFO receive/transmit functions are disabled.
- The receive logic is kept in Normal mode, but it ignores all traffic from the line except magic packets. They are detected so that a remote agent can wake the node.

23.6.7.2 Magic packet detection

The core is designed to detect magic packets (see Magic packets) with the destination address set to:

- Any multicast address
- The broadcast address
- The unicast address programmed in PADDR1/2

When a magic packet is detected, EIR[WAKEUP] is set and none of the statistic registers are incremented.

23.6.7.3 Wakeup

When a magic packet is detected, indicated by ENET*n*_EIR[WAKEUP], ENET*n*_ECR[SLEEP] should be cleared to resume normal operation of the MAC. Clearing the SLEEP bit automatically masks ENET*n*_ECR[MAGICEN], disabling magic packet detection.

23.6.8 IP accelerator functions

The following sections describe the IP accelerator functions.

23.6.8.1 Checksum calculation

The IP and ICMP, TCP, UDP checksums are calculated with one's complement arithmetic summing up 16-bit values.

- For ICMP, the checksum is calculated over the complete ICMP datagram, in other words without IP header.
- For TCP and UDP, the checksums contain the header and data sections and values from the IP header, which can be seen as a pseudo-header that is not actually present in the data stream.

Table 23-127. IPv4 pseudo-header for checksum calculation

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Source address							
Destination address							
Zero		Protocol		TCP/UDP length			

Table 23-128. IPv6 pseudo-header for checksum calculation

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Source address							
Destination address							
TCP/UDP length							
Zero						Next header	

The TCP/UDP length value is the length of the TCP or UDP datagram, which is equal to the payload of an IP datagram. It is derived by subtracting the IP header length from the complete IP datagram length that is given in the IP header (IPv4), or directly taken from the IP header (IPv6). The protocol field is the corresponding value from the IP header. The Zero fields are all zeroes.

For IPv6, the complete 128-bit addresses are considered. The next header value identifies the upper layer protocol as either TCP or UDP. It may differ from the next header value of the IPv6 header if extension headers are inserted before the protocol header.

The checksum calculation uses 16-bit words in network byte order: The first byte sent/received is the MSB, and the second byte sent/received is the LSB of the 16-bit value to add to the checksum. If the frame ends on an odd number of bytes, a zero byte is appended for checksum calculation only, and is not actually transmitted.

23.6.8.2 Additional padding processing

According to IEEE 802.3, any Ethernet frame must have a minimum length of 64 octets.

The MAC usually removes padding on receive when a frame with length information is received. Because IP frames have a type value instead of length, the MAC does not remove padding for short IP frames, as it is not aware of the frame contents.

The IP accelerator function can be configured to remove the Ethernet padding bytes that might follow the IP datagram.

On transmit, the MAC automatically adds padding as necessary to fill any frame to a 64-byte length.

23.6.8.3 32-bit Ethernet payload alignment

The data FIFOs allow inserting two additional arbitrary bytes in front of a frame. This extends the 14-byte Ethernet header to a 16-byte header, which leads to alignment of the Ethernet payload, following the Ethernet header, on a 32-bit boundary.

This function can be enabled for transmit and receive independently with the corresponding SHIFT16 bits in the ENET*n*_TACC and ENET*n*_RACC registers.

When enabled, the valid frame data is arranged as shown in Table 23-129.

Table 23-129. 64-bit interface data structure with SHIFT16 enabled

63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0	

Table continues on the next page...

Table 23-129. 64-bit interface data structure with SHIFT16 enabled (continued)

Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0	Any value	Any value
Byte 13	Byte 12	Byte 11	Byte 10	Byte 9	Byte 8	Byte 7	Byte 6
...							

23.6.8.3.1 Receive processing

When ENET*n*_RACC[SHIFT16] is set, each frame is received with two additional bytes in front of the frame.

The user application must ignore these first two bytes and find the first byte of the frame in bits 23–16 of the first word from the RX FIFO.

Note

SHIFT16 must be set during initialization and kept set during the complete operation, because it influences the FIFO write behavior.

23.6.8.3.2 Transmit processing

When ENET*n*_TACC[SHIFT16] is set, the first two bytes of the first word written (bits 15–0) are discarded immediately by the FIFO write logic.

The SHIFT16 bit can be enabled/disabled for each frame individually if required, but can be changed only between frames.

23.6.8.4 Received frame discard

Because the receive FIFO must be operated in store and forward mode (ENET*n*_RSFL cleared), received frames can be discarded based on the following errors:

- The MAC function receives the frame with an error:
  - The frame has an invalid payload length
  - Frame length is greater than MAX_FL
  - Frame received with a CRC-32 error
  - Frame truncated due to receive FIFO overflow
  - Frame is corrupted as PHY signaled an error (RX_ERR asserted during reception)

- An IP frame is detected and the IP header checksum is wrong
- An IP frame with a valid IP header and a valid IP header checksum is detected, the protocol is known but the protocol-specific checksum is wrong

If one of the errors occurs and the IP accelerator function is configured to discard frames (ENET*n*_RACC), the frame is automatically discarded. Statistics are maintained normally and are not affected by this discard function.

23.6.8.5 IPv4 fragments

When an IPv4 IP fragment frame is received, only the IP header is inspected and its checksum verified. 32-bit alignment operates the same way on fragments as it does on normal IP frames, as specified above.

The IP fragment frame payload is not inspected for any protocol headers. As such, a protocol header would only exist in the very first fragment. To assist in protocol-specific checksum verification, the one's-complement sum is calculated on the IP payload (all bytes following the IP header) and provided with the frame status word.

The frame fragment status field (RxBD[FRAG]) is set to indicate a fragment reception, and the one's-complement sum of the IP payload is available in RxBD[Payload checksum].

Note

After all fragments have been received and reassembled, the application software can take advantage of the payload checksum delivered with the frame's status word to calculate the protocol-specific checksum of the datagram.

For example, if a TCP payload is delivered by multiple IP fragments, the application software can calculate the pseudo-header checksum value from the first fragment, and add the payload checksums delivered with the status for all fragments to verify the TCP datagram checksum.

23.6.8.6 IPv6 support

The following sections describe the IPv6 support.

23.6.8.6.1 Receive processing

An Ethernet frame of type 0x86DD identifies an IP Version 6 frame (IPv6) frame. If an IPv6 frame is received, the first IP header is inspected (first ten words), which is available in every IPv6 frame.

If the receive SHIFT16 function is enabled, the IP header is aligned on a 32-bit boundary allowing more efficient processing (see 32-bit Ethernet payload alignment).

For TCP and UDP datagrams, the pseudo-header checksum calculation is performed and verified.

To assist in protocol-specific checksum verification, the one's-complement sum is always calculated on the IP payload (all bytes following the IP header) and provided with the frame status word. For example, if extension headers were present, their sums can be subtracted in software from the checksum to isolate the TCP/UDP datagram checksum, if required.

23.6.8.6.2 Transmit processing

For IPv6 transmission, the SHIFT16 function is supported to process 32-bit aligned datagrams.

IPv6 has no IP header checksum; therefore, the IP checksum insertion configuration is ignored.

The protocol checksum is inserted only if the next header of the IP header is a known protocol (TCP, UDP, or ICMP). If a known protocol is detected, the checksum over all bytes following the IP header is calculated and inserted in the correct position.

The pseudo-header checksum calculation is performed for TCP and UDP datagrams accordingly.

23.6.9 Resets and stop controls

The following sections describe the resets and stop controls.

23.6.9.1 Hardware reset

To reset the Ethernet module, set ENET*n*_ECR[RESET].

23.6.9.2 Soft reset

When ENET*n*_ECR[ETHER_EN] is cleared during operation, the following occurs:

- DMA, buffer descriptor, and FIFO control logic are reset, including the buffer descriptor and FIFO pointers.
- A currently ongoing transmit is terminated by asserting GMII/TXER to the PHY.
- A currently ongoing transmit FIFO write from the application is terminated by stopping the write to the FIFO, and all further data from the application is ignored. All subsequent writes are ignored until re-enabled.
- A currently ongoing receive FIFO read is terminated. The RxBD has arbitrary values in this case.

23.6.9.3 Hardware freeze

When the processor enters debug mode and ECR[DBGEN] is set, the MAC enters a freeze state where it stops all transmit and receive activities gracefully.

The following happens when the MAC enters hardware freeze:

- A currently ongoing receive transaction on the receive application interface is completed as normal. No further frames are read from the FIFO.
- A currently ongoing transmit transaction on the transmit application interface is completed as normal (in other words, until writing end-of-packet (EOP)).
- A currently ongoing frame receive is completed normally, after which no further frames are accepted from the MII/GMII.
- A currently ongoing frame transmit is completed normally, after which no further frames are transmitted.

23.6.9.4 Graceful stop

During a graceful stop, any currently ongoing transactions are completed normally and no further frames are accepted. The MAC can resume from a graceful stop without the need for a reset (for example, clearing ETHER_EN is not required).

The following conditions lead to a graceful stop of the MAC transmit or receive datapaths.

23.6.9.4.1 Graceful transmit stop (GTS)

When gracefully stopped, the MAC is no longer reading frame data from the transmit FIFO and has completed any ongoing transmission.

In any of the following conditions, the transmit datapath stops after an ongoing frame transmission has been completed normally.

- ENET*n*_TCR[GTS] is set by software.
- ENET*n*_TCR[TFC_PAUSE] is set by software requesting a pause frame transmission. The status (and register bit) is cleared after the pause frame has been sent.
- A pause frame was received stopping the transmitter. The stopped situation is terminated when the pause timer expires or a pause frame with zero quanta is received.
- MAC is placed in Sleep mode by software or the processor entering Stop mode (see Sleep mode).
- The MAC is in Hardware Freeze mode.

When the transmitter has reached its stopped state, the following events occur:

- The GRA interrupt is asserted, when transitioned into stopped.
- In Hardware Freeze mode, the GRA interrupt does not wait for the application write completion and asserts when the transmit state machine (in other words, line side of TX FIFO) reaches its stopped state.

23.6.9.4.2 Graceful receive stop (GRS)

When gracefully stopped, the MAC is no longer writing frames into the receive FIFO.

The receive datapath stops after any ongoing frame reception has been completed normally, if any of the following conditions occur:

- MAC is placed in Sleep mode either by the software or the processor is in Stop mode). The MAC continues to receive frames and search for magic packets if enabled (see Magic packet detection). However, no frames are written into the receive FIFO, and therefore are not forwarded to the application.
- The MAC is in Hardware Freeze mode. The MAC does not accept any frames from the MII/GMII.

When the receive datapath is stopped, the following events occur:

- If the RX is in the stopped state, RCR[GRS] is set
- The GRA interrupt is asserted when the transmitter and receiver are stopped
- Any ongoing receive transaction to the application (RX FIFO read) continues normally until the frame end of package (EOP) is reached. After this, the following occurs:
  - When Sleep mode is active, all further frames are discarded, flushing the RX FIFO
  - In Hardware Freeze mode, no further frames are delivered to the application and they stay in the receive FIFO.

Note

The assertion of GRS does not wait for an ongoing FIFO read transaction on the application side of the FIFO (FIFO read).

23.6.9.4.3 Graceful stop interrupt (GRA)

The graceful stopped interrupt (GRA) is asserted for the following conditions:

- In Sleep mode, the interrupt asserts only after both TX and RX datapaths are stopped.
- In Hardware Freeze mode, the interrupt asserts only after both TX and RX datapaths are stopped.
- The MAC transmit datapath is stopped for any other condition (GTS, TFC_PAUSE, pause received).

The GRA interrupt is triggered only once when the stopped state is entered. If the interrupt is cleared while the stop condition persists, no further interrupt is triggered.

23.6.10 IEEE 1588 functions

To allow for IEEE 1588 or similar time synchronization protocol implementations, the MAC is combined with a time-stamping module to support precise time-stamping of incoming and outgoing frames. Set ENET*n*_ECR[EN1588] to enable 1588 support.

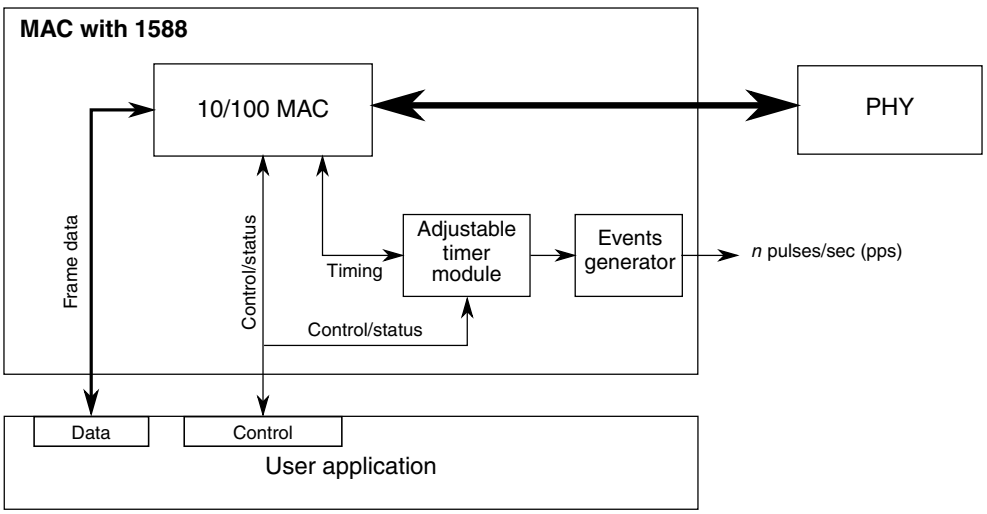


Figure 23-109. IEEE 1588 functions overview

23.6.10.1 Adjustable timer module

The adjustable timer module (TSM) implements the free-running counter (FRC), which generates the timestamps. The FRC operates with the time-stamping clock, which can be set to any value depending on your system requirements.

Through dedicated correction logic, the timer can be adjusted to allow synchronization to a remote master and provide a synchronized timing reference to the local system. The timer can be configured to cause an interrupt after a fixed time period, to allow synchronization of software timers or perform other synchronized system functions.

The timer is typically used to implement a period of one second; hence, its value ranges from 0 to (1×10^9)-1. The period event can trigger an interrupt, and software can maintain the seconds and hours time values as necessary.

23.6.10.1.1 Adjustable timer implementation

The adjustable timer consists of a programmable counter/accumulator and a correction counter. The periods of both counters and their increment rates are freely configurable, allowing very fine tuning of the timer.

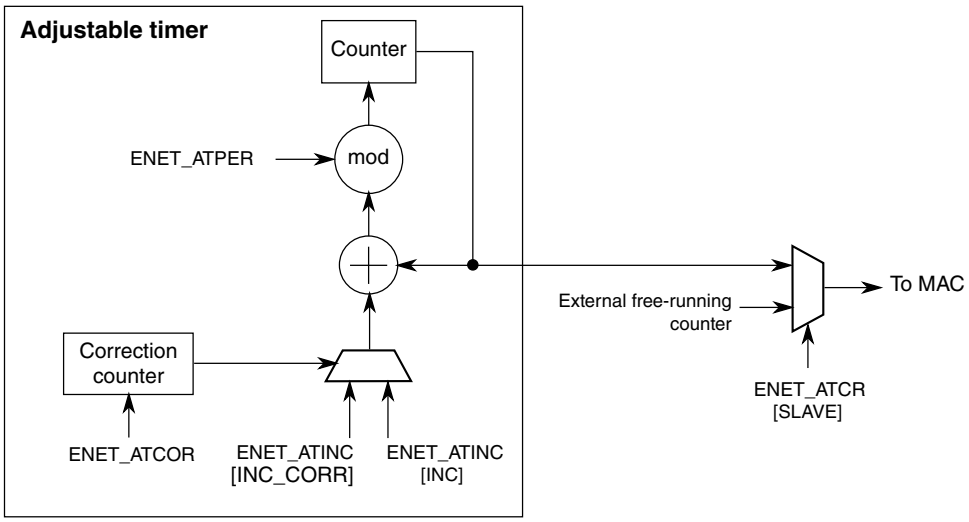


Figure 23-110. Adjustable timer implementation detail

The counter produces the current time. During each time-stamping clock cycle, a constant value is added to the current time as programmed in ENET*n*_ATINC. The value depends on the chosen time-stamping clock frequency. For example, if it operates at 125 MHz, setting the increment to eight represents 8 ns.

The period, configured in ENET*n*_ATPER, defines the modulo when the counter wraps. In a typical implementation, the period is set to 1×10^9 so that the counter wraps every second, and hence all timestamps represent the absolute nanoseconds within the one second period. When the period is reached, the counter wraps to start again respecting the period modulo. This means it does not necessarily start from zero, but instead the counter is loaded with the value (Current + Inc $-(1 \times 10^9)$), assuming the period is set to 1×10^9.

The correction counter operates fully independently, and increments by one with each time-stamping clock cycle. When it reaches the value configured in ENET*n*_ATCOR, it restarts and instructs the timer once to increment by the correction value, instead of the normal value.

The normal and correction increments are configured in ENET*n*_ATINC. To speed up the timer, set the correction increment more than the normal increment value. To slow down the timer, set the correction increment less than the normal increment value.

The correction counter only defines the distance of the corrective actions, not the amount. This allows very fine corrections and low jitter (in the range of 1 ns) independent of the chosen clock frequency.

By enabling slave mode (ENET*n*_ATCR[SLAVE] = 1), the timer is ignored and the current time is externally provided from one of the external modules. See the Chip Configuration details for which clock source is used. This is useful if multiple modules

within the system must operate from a single timer. When slave mode is enabled, you still must set ENET*n*_ATINC[INC] to the value of the master, since it is used for internal comparisons.

23.6.10.2 Transmit timestamping

Only 1588 event frames need to be time-stamped on transmit. The client application (for example, the MAC driver) should detect 1588 event frames and set TxBD[TS] together with the frame.

If TxBD[TS] is set, the MAC records the timestamp for the frame in ENET*n*_ATSTMP. ENET*n*_EIR[TS_AVAIL] is set to indicate that a new timestamp is available.

Software implements a handshaking procedure by setting TxBD[TS] when it transmits the frame for which a timestamp is needed, and then waits for ENET*n*_EIR[TS_AVAIL] to determine when the timestamp is available. The timestamp is then read from ENET*n*_ATSTMP. This is done for all event frames. Other frames do not use TxBD[TS] and, therefore, do not interfere with the timestamp capture.

23.6.10.3 Receive timestamping

When a frame is received, the MAC latches the value of the timer when the frame's start of frame delimiter (SFD) field is detected, and provides the captured timestamp on RxBD[1588 timestamp]. This is done for all received frames.

23.6.10.4 Time synchronization

The adjustable timer module is available to synchronize the local clock of a node to a remote master. It implements a free running 32-bit counter, and also contains an additional correction counter.

The correction counter increases or decreases the rate of the free running counter, enabling very fine granular changes of the timer for synchronization, yet adding only very low jitter when performing corrections.

The application software implements, in a slave scenario, the required control algorithm, setting the correction to compensate for local oscillator drifts and locking the timer to the remote master clock on the network.

The timer and all timestamp-related information should be configured to show the true nanoseconds value of a second (in other words, the timer is configured to have a period of one second). Hence, the values range from 0 to (1×10^9)–1. In this application, the seconds counter is implemented in software using an interrupt function that is executed when the nanoseconds counter wraps at 1×10^9.

23.6.10.5 Input capture and output compare

The Input Capture Output Compare block can be used to provide precise hardware timing for input and output events.

23.6.10.5.1 Input capture

The TCCR*n* capture registers latch the time value when the corresponding external event occurs. An event can be a rising-, falling-, or either-edge of one of the 1588_TMR*n* signals. An event will cause the corresponding TCSR*n*[TF] timer flag to be set, indicating that an input capture has occurred. If the corresponding interrupt is enabled with the TCSR*n*[TIE] field, an interrupt can be generated.

23.6.10.5.2 Output compare

The TCCR*n* compare registers are loaded with the time at which the corresponding event should occur. When the ENET free-running counter value matches the output compare reference value in the TCCR*n* register, the corresponding flag, TCSR*n*[TF], is set, indicating that an output compare has occurred. The corresponding interrupt, if enabled by TCSR*n*[TIE], will be generated, and the corresponding 1588_TMR*n* output signal will be asserted according to TCSR*n*[TMODE].

23.6.10.5.3 DMA requests

A DMA request can be enabled by setting TCSR*n*[TDRE]. The corresponding DMA request is generated when the TCSR*n*[TF] timer flag is set. When the DMA has completed, the corresponding TCSRn[TF] flag is cleared.

23.6.11 FIFO thresholds

The core FIFO thresholds are fully programmable to dynamically change the FIFO operation.

For example, store and forward transfer can be enabled by a simple change in the FIFO threshold registers.

The thresholds are defined in 64-bit words.

23.6.11.1 Receive FIFO

Four programmable thresholds are available, which can be set to any value to control the core operation as follows.

Table 23-130. Receive FIFO thresholds definition

Register	Description
ENET*n*_RSFL [RX_SECTION_FULL]	When the FIFO level reaches the ENET*n*_RSFL value, the MAC status signal is asserted to indicate that data is available in the receive FIFO (cut-through operation). Once asserted, if the FIFO empties below the threshold set with ENET*n*_RAEM and if the end-of-frame is not yet stored in the FIFO, the status signal is deasserted again. If a frame has a size smaller than the threshold (in other words, an end-of-frame is available for the frame), the status is also asserted. To enable store and forward on the receive path, clear ENET*n*_RSFL. The MAC status signal is asserted only when a complete frame is stored in the receive FIFO. When programming a non-zero value to ENET*n*_RSFL (cut-through operation) it should be greater than ENET*n*_RAEM.
ENET*n*_RAEM [RX_ALMOST_EMPTY]	When the FIFO level reaches the ENET*n*_RAEM value, and the end-of-frame has not been received, the core receive read control stops the FIFO read (and subsequently stops transferring data to the MAC client application). It continues to deliver the frame, if again more data than the threshold or the end-of-frame is available in the FIFO. Set ENET*n*_RAEM to a minimum of six.
ENET*n*_RAFL [RX_ALMOST_FULL]	When the FIFO level approaches the maximum and there is no more space remaining for at least ENETn_RAFL number of words, the MAC control logic stops writing data in the FIFO and truncates the receive frame to avoid FIFO overflow. The corresponding error status is set when the frame is delivered to the application. Set ENET*n*_RAFL to a minimum of 4.
ENET*n*_RSEM [RX_SECTION_EMPTY]	When the FIFO level reaches the ENET*n*_RSEM value, an indication is sent to the MAC transmit logic, which generates an XOFF pause frame. This indicates FIFO congestion to the remote Ethernet client. When the FIFO level goes below the value programmed in ENET*n*_RSEM, an indication is sent to the MAC transmit logic, which generates an XON pause frame. This indicates the FIFO congestion is cleared to the remote Ethernet client. Clearing ENET*n*_RSEM disables any pause frame generation.

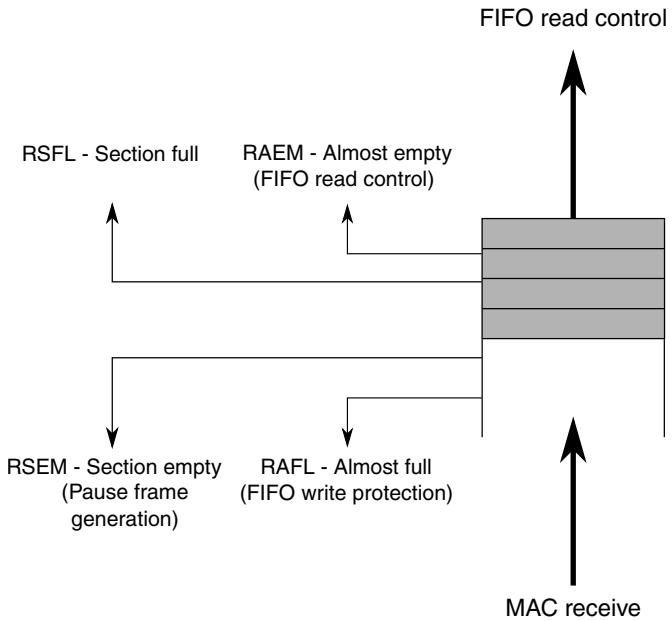


Figure 23-111. Receive FIFO overview

23.6.11.2 Transmit FIFO

Four programmable thresholds are available which control the core operation as described below.

Table 23-131. Transmit FIFO thresholds definition

Register	Description
ENET*n*_TAEM [TX_ALMOST _EMPTY]	When the FIFO level reaches the ENET*n*_TAEM value and no end-of-frame is available for the frame, the MAC transmit logic avoids a FIFO underflow by stopping FIFO reads and transmitting the Ethernet frame with an MII error indication. Set ENET*n*_TAEM to a minimum of 4.
ENET*n*_TAFL [TX_ALMOST _FULL]	When the FIFO level approaches the maximum, so that there is no more space for at least ENET*n*_TAFL number of words, the MAC deasserts its control signal to the application. If the application does not react on this signal, the FIFO write control logic avoids FIFO overflow by truncating the current frame and setting the error status. As a result, the frame is transmitted with an GMII/MII error indication. Set ENET*n*_TAFL to a minimum of 4. Larger values allow more latency for the application to react on the MAC control signal deassertion, before the frame is truncated. A typical setting is 8, which offers 3–4 clock cycles of latency to the application to react on the MAC control signal deassertion.
ENET*n*_TSEM [TX_SECTION _EMPTY]	When the FIFO level reaches the ENET*n*_TSEM value, a MAC status signal is deasserted to indicate that the transmit FIFO is getting full. This gives the ENET module an indication to slow or stop its write transaction to avoid a buffer overflow. This is a pure indication function to the application. It has no effect within the MAC. When ENET*n*_TSEM is 0, the signal is never deasserted.
ENET*n*_TFWR	When the FIFO level reaches the ENET*n*_TFWR value and when STRFWD is cleared, the MAC transmit control logic starts frame transmission before the end-of-frame is available in the FIFO (cut-through operation).

Table 23-131. Transmit FIFO thresholds definition

Register	Description
	If a complete frame has a size smaller than the ENET*n*_TFWR threshold, the MAC also transmits the frame to the line. To enable store and forward on the transmit path, set STRFWD. In this case, the MAC starts to transmit data only when a complete frame is stored in the transmit FIFO.

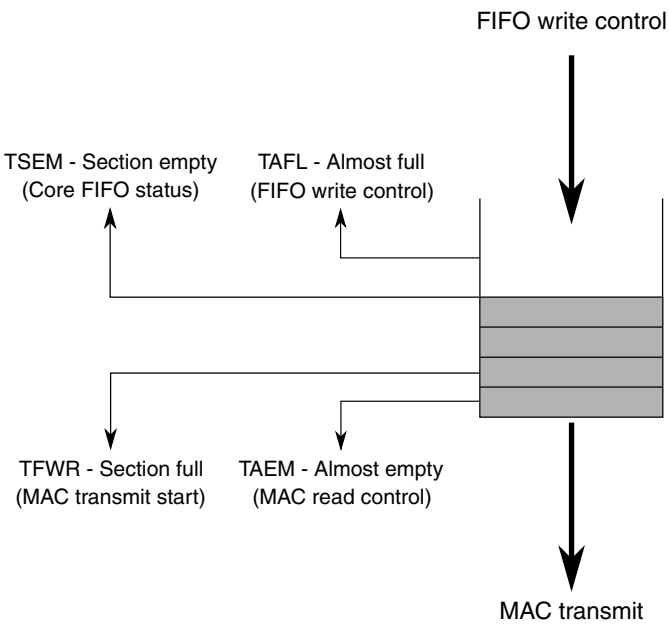


Figure 23-112. Transmit FIFO overview

23.6.12 Loopback options

The core implements external and internal loopback options, which are controlled by the ENET*n*_RCR register fields found here.

The core implements external and internal loopback options, which are controlled by the following ENET*n*_RCR register fields:

Table 23-132. Loopback options

Register field	Description
LOOP	Internal MII loopback. The MAC transmit is returned to the MAC receive. No data is transmitted to the external interfaces. In MII internal loopback, MII_TXCLK and MII_RXCLK must be provided with a clock signal (2.5 MHz for 10 Mbit/s, and 25 MHz for 100 Mbit/s))

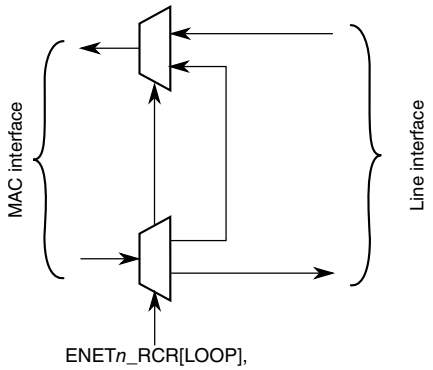


Figure 23-113. Loopback options

23.6.13 Legacy buffer descriptors

To support the Ethernet controller on previous Freescale devices, legacy FEC buffer descriptors are available. To enable legacy support, clear ENET*n*_ECR[1588EN].

NOTE

- The legacy buffer descriptor tables show the byte order for chips. DBSWP must be set to after reset to enable mode.

23.6.13.1 Legacy receive buffer descriptor

The following table shows the legacy FEC receive buffer descriptor. contains the descriptions for each field.

23.6.13.2 Legacy transmit buffer descriptor

The following table shows the legacy FEC transmit buffer descriptor. contains the descriptions for each field.

23.6.14 Enhanced buffer descriptors

This section provides a description of the enhanced operation of the driver/DMA via the buffer descriptors.

It is followed by a detailed description of the receive and transmit descriptor fields. To enable the enhanced features, set ENET*n*_ECR[1588EN].

NOTE

The enhanced buffer descriptor tables show the byte order for chips. DBSWP must be set to after reset to enable mode.

23.6.14.1 Enhanced receive buffer descriptor

The following table shows the enhanced uDMA receive buffer descriptor. contains the descriptions for each field.

23.6.14.2 Enhanced transmit buffer descriptor

23.6.15 Client FIFO application interface

The FIFO interface is completely asynchronous from the Ethernet line, and the transmit and receive interface can operate at a different clock rate.

All transfers to/from the user application are handled independently of the core operation, and the core provides a simple interface to user applications based on a two-signal handshake.

23.6.15.1 Data structure description

The data structure defined in the following tables for the FIFO interface must be respected to ensure proper data transmission on the Ethernet line. Byte 0 is sent to and received from the line first.

Table 23-133. FIFO interface data structure

	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
Word 0	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
Word 1	Byte 15	Byte 14	Byte 13	Byte 12	Byte 11	Byte 10	Byte 9	Byte 8
...	...							

The size of a frame on the FIFO interface may not be a modulo of 64-bit.

The user application may not care about the Ethernet frame formats in full detail. It needs to provide and receive an Ethernet frame with the following structure:

- Ethernet MAC destination address

- Ethernet MAC source address
- Optional 802.1q VLAN tag (VLAN type and info field)
- Ethernet length/type field
- Payload

Frames on the FIFO interface do not contain preamble and SFD fields, which are inserted and discarded by the MAC on transmit and receive, respectively.

- On receive, CRC and frame padding can be stripped or passed through transparently.
- On transmit, padding and CRC can be provided by the user application, or appended automatically by the MAC independently for each frame. No size restrictions apply.

Note

On transmit, if ENET*n*_TCR[ADDINS] is set, bytes 6–11 of each frame can be set to any value, since the MAC overwrites the bytes with the MAC address programmed in the ENET*n*_PAUR and ENET*n*_PALR registers.

Table 23-134. FIFO interface frame format

Byte number	Field
0–5	Destination MAC address
6–11	Source MAC address
12–13	Length/type field
14–N	Payload data

VLAN-tagged frames are supported on both transmit and receive, and implement additional information (VLAN type and info).

Table 23-135. FIFO interface VLAN frame format

Byte number	Field
0–5	Destination MAC address
6–11	Source MAC address
12–15	VLAN tag and info
16–17	Length/type field
18–N	Payload data

Note

The standard defines that the LSB of the MAC address is sent/received first, while for all the other header fields — in other words, length/type, VLAN tag, VLAN info, and pause quanta — the MSB is sent/received first.

23.6.15.2 Data structure examples

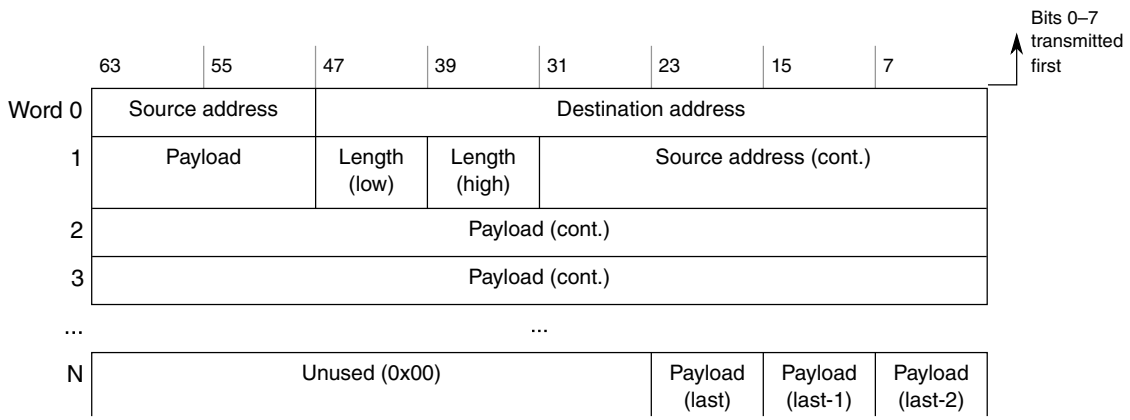


Figure 23-114. Normal Ethernet frame 64-bit mapping example

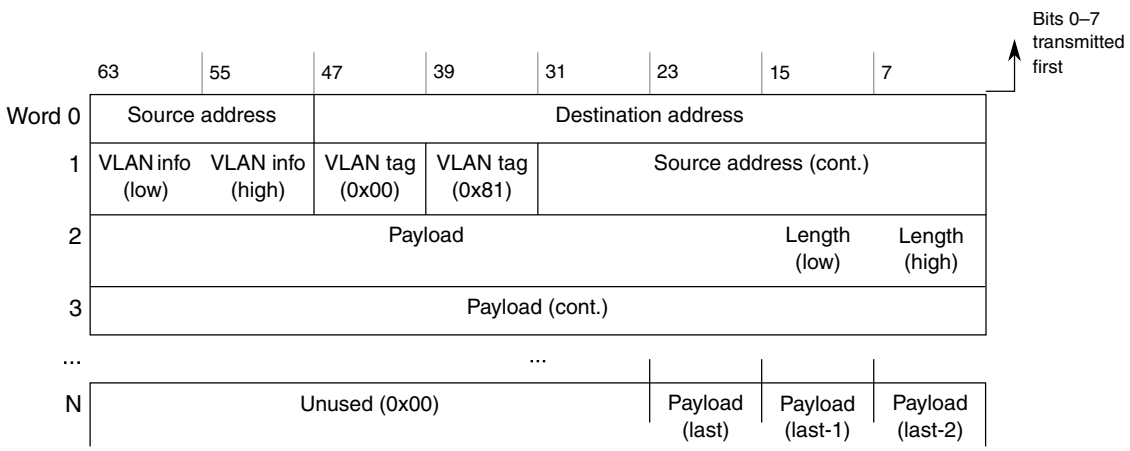


Figure 23-115. VLAN-tagged frame 64-bit mapping example

If CRC forwarding is enabled (CRCFWD = 0), the last four valid octets of the frame contain the FCS field. The non-significant bytes of the last word can have any value.

23.6.15.3 Frame status

A MAC layer status word and an accelerator status word is available in the receive buffer descriptor.

See Enhanced buffer descriptors for details.

The status is available with each frame with the last data of the frame.

If the frame status contains a MAC layer error (for example, CRC or length error), RxBD[ME] is also set with the last data of the frame.

23.6.16 FIFO protection

The following sections describe the FIFO protection mechanisms.

23.6.16.1 Transmit FIFO underflow

During a frame transfer, when the transmit FIFO reaches the almost empty threshold with no end-of-frame indication stored in the FIFO, the MAC logic:

- Stops reading data from the FIFO
- Asserts the MII error signal (MII_TXER) (1 in Figure 23-116) to indicate that the fragment already transferred is not valid
- Deasserts the MII transmit enable signal (MII_TXEN) to terminate the frame transfer (2)

After an underflow, when the application completes the frame transfer (3), the MAC transmit logic discards any new data available in the FIFO until the end of packet is reached (4) and sets the enhanced TxBD[UE] field.

The MAC starts to transfer data on the MII interface when the application sends a new frame with a start of frame indication (5).

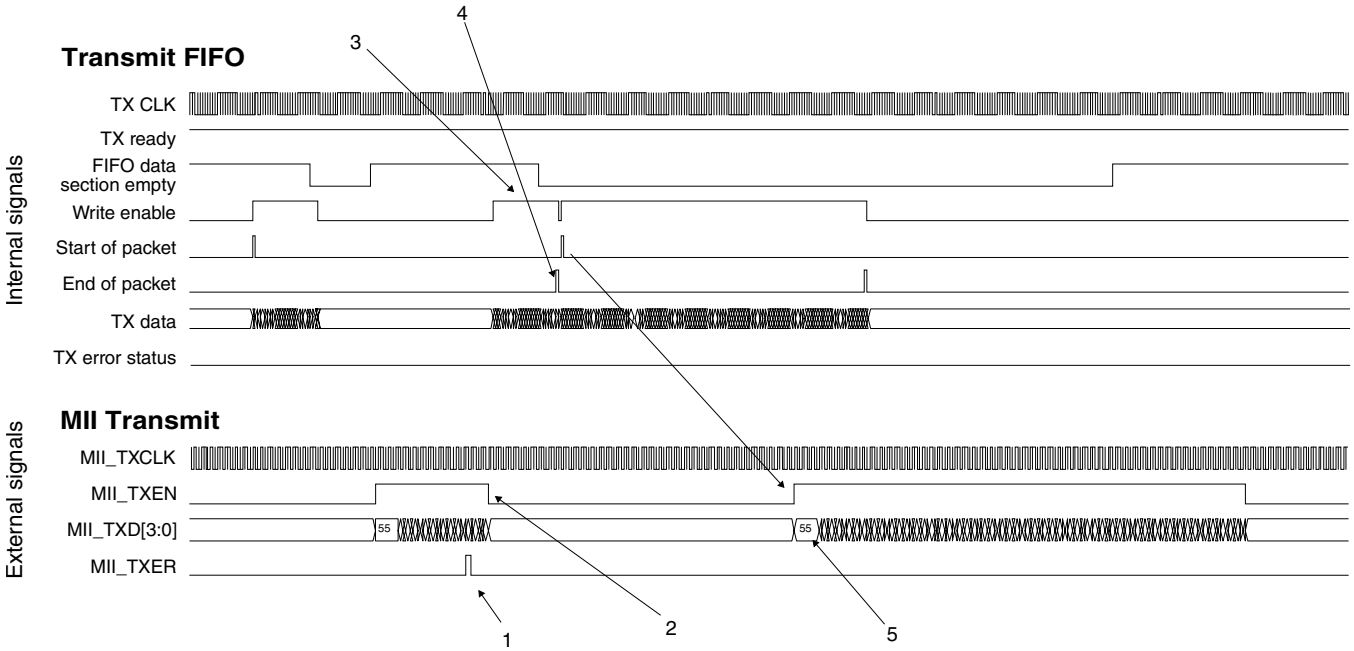


Figure 23-116. Transmit FIFO underflow protection

23.6.16.2 Transmit FIFO overflow

On the transmit path, when the FIFO reaches the programmable almost full threshold, the internal MAC ready signal is deasserted. The application should stop sending new data.

However, if the application keeps sending data, the transmit FIFO overflows, corrupting contents that were previously stored. The core logic sets the enhanced TxBD[OE] field for the next frame transmitted to indicate this overflow occurence.

Note

Overflow is a fatal error and must be addressed by resetting the core or clearing ENET*n*_ECR[ETHER_EN], to clear the FIFOs and prepare for normal operation again.

23.6.16.3 Receive FIFO overflow

During a frame reception, if the client application is not able to receive data (1), the MAC receive control truncates the incoming frame when the FIFO reaches the programmable almost-full threshold to avoid an overflow.

The frame is subsequently received on the FIFO interface with an error indication (enhanced RxBD[ME] field set together with receive end-of-packet) (2) with the truncation error status field set (3).

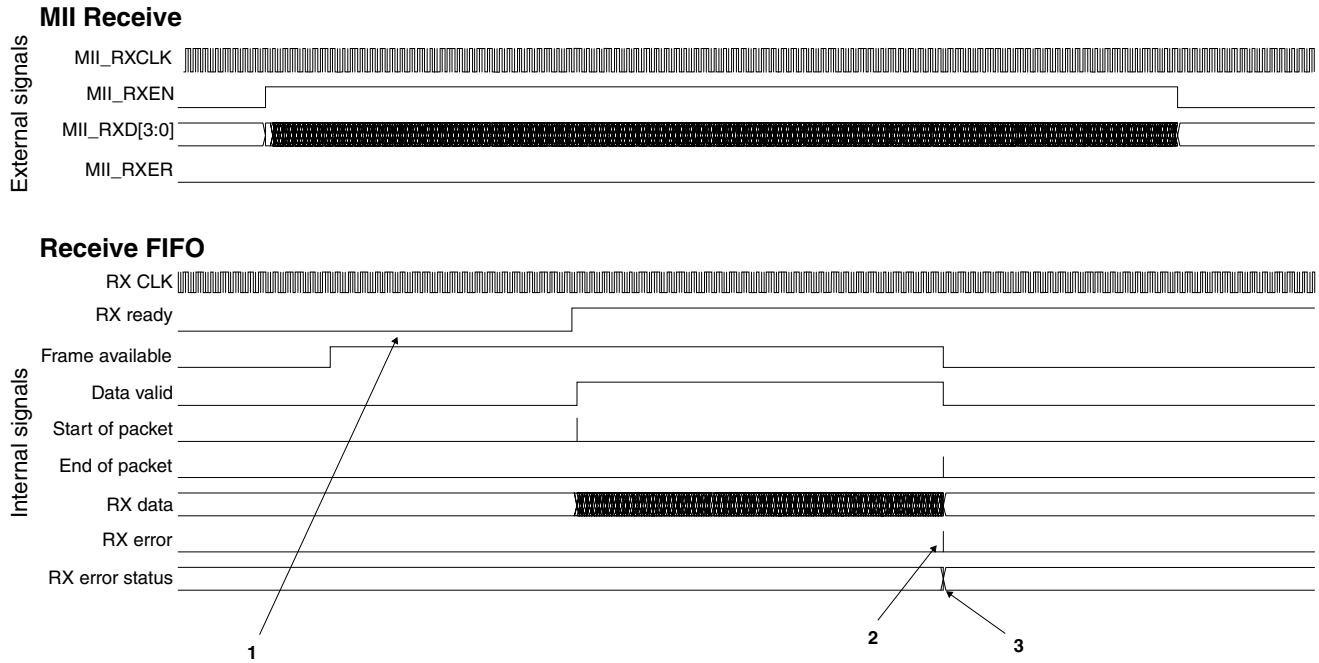


Figure 23-117. Receive FIFO overflow protection

23.6.17 PHY management interface

The MDIO interface is a two-wire management interface. The MDIO management interface implements a standardized method to access the PHY device management registers.

The core implements a master MDIO interface, which can be connected to up to 32 PHY devices.

23.6.17.1 MDIO frame format

The core MDIO master controller communicates with the slave (PHY device) using frames that are defined in the following table.

A complete frame has a length of 64 bits made up of an optional 32-bit preamble, 14-bit command, 2-bit bus direction change, and 16-bit data. Each bit is transferred on the rising edge of the MDIO clock (MDC signal).

The core PHY management interface supports the standard MDIO specification (IEEE803.2 Clause 22).

Table 23-136. MDIO frame formats (read/write)

Type	Command					TA	Data		Idle
	PRE	ST	OP	Addr1	Addr2		MSB	LSB	
Read	1…1	01	10	xxxxx	xxxxx	Z0	xxxxxxxxxxxxxxxx		Z
Write	1…1	01	01	xxxxx	xxxxx	10	xxxxxxxxxxxxxxxx		Z

Table 23-137. MDIO frame field descriptions

Field	Description
PRE	Preamble. 32 bits of logical ones sent prior to every transaction when ENET*n*_MSCR[DIS_PRE] is cleared. If DIS_PRE is set, the preamble is not generated.
ST	Start indication, programmed with ENET*n*_MMFR[ST] • Standard MDIO (Clause 22): 01
OP	Opcode defines if a read or write operation is performed, programmed with ENET*n*_MMFR[OP]. 01 Write operation 10 Read operation
Addr1	The PHY device address, programmed with ENET*n*_MMFR[PA]. Up to 32 devices can be addressed.
Addr2	Register address, programmed with ENET*n*_MMFR[RA]. Each PHY can implement up to 32 registers.
TA	Turnaround time, programmed with ENET*n*_MMFR[TA]. Two bit-times are reserved for read operations to switch the data bus from write to read. The PHY device presents its register contents in the data phase and drives the bus from the second bit of the turnaround phase.
Data	16 bits of data, set to ENET*n*_MMFR[DATA]. Written to or read from the PHY
Idle	The MDIO data signal is tri-stated between frames.

23.6.17.2 MDIO clock generation

The MDC clock is generated from the internal bus clock divided by the value programmed in ENET*n*_MSCR[MII_SPEED].

23.6.17.3 MDIO operation

To perform an MDIO access, set the MDIO command register (ENET*n*_MMFR) according to the description provided in MII Management Frame Register (ENETn_MMFR).

To check when the programmed access completes, read the ENET*n*_EIR[MII] field.

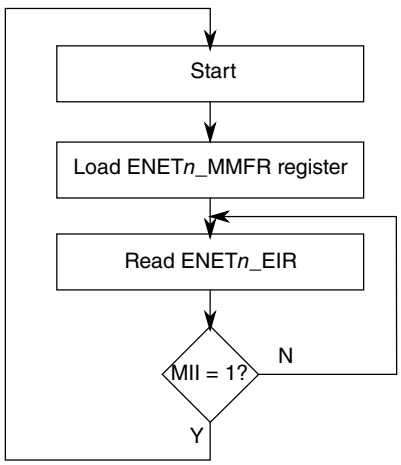


Figure 23-118. MDIO access overview

23.6.18 Ethernet interfaces

The following Ethernet interfaces are implemented:

- Fast Ethernet MII (Media Independent Interface)
- RMII 10/100 using interface converters/gaskets
- RGMII 10/100/1000 by way of interface converters/gaskets

The following table shows how to configure ENET registers to select each interface.

Mode	ECR[SLEEP]	RCR[RMII_10T]	RCR[RMII_MODE]	RCR[RGMII_EN]
MII - 10 Mbit/s	0	—	0	0
MII - 100 Mbit/s	0	—	0	0
RMII - 10 Mbit/s	0	1	1	0
RMII - 100 Mbit/s	0	0	1	0
RGMII - 10 Mbit/s	0	1	0	1
RGMII - 100 Mbit/s	0	0	0	1
RGMII - 1000 Mbit/s	1	—	0	1

23.6.18.1 RMII interface

In RMII receive mode, for normal reception following assertion of CRS_DV, RXD[1:0] is 00 until the receiver determines that the receive event has a proper start-of-stream delimiter (SSD).

The preamble appears (RXD[1:0]=01) and the MACs begin capturing data following detection of SFD.

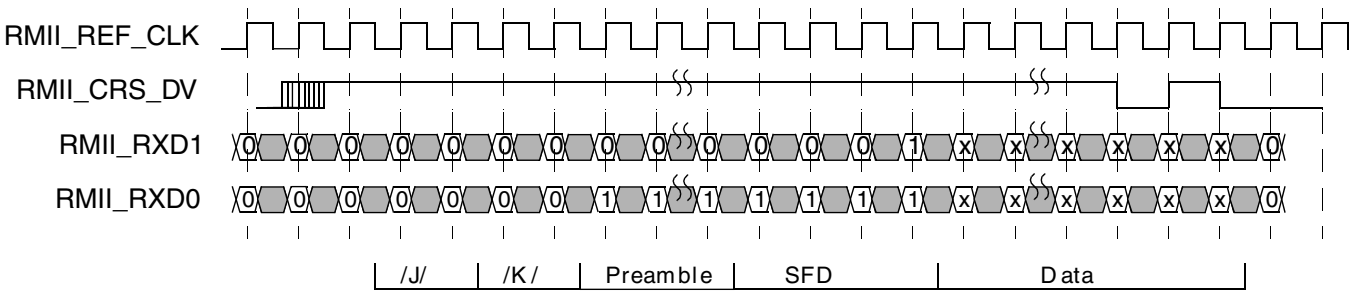


Figure 23-119. RMII receive operation

If a false carrier is detected (bad SSD), then RXD[1:0] is 10 until the end of the receive event. This is a unique pattern since a false carrier can only occur at the beginning of a packet where the preamble is decoded (RXD[1:0] = 01).

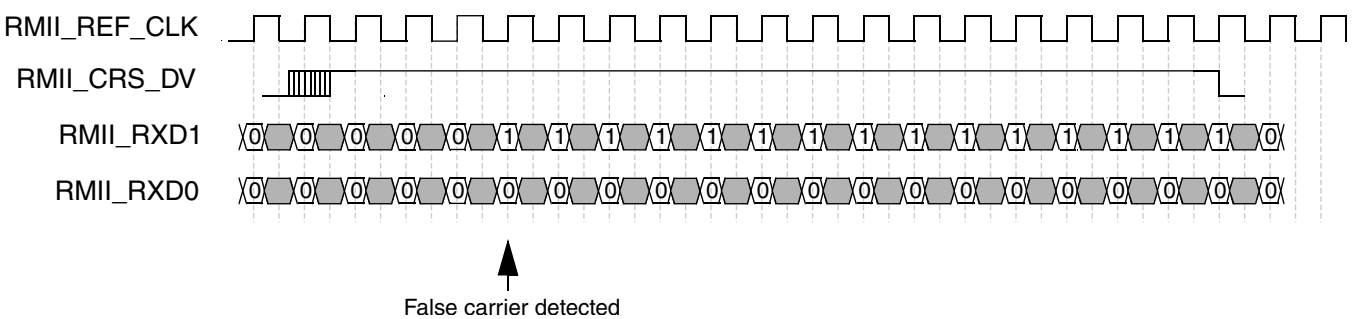


Figure 23-120. RMII receive operation with false carrier

In RMII transmit mode, TXD[1:0] provides valid data for each REF_CLK period while TXEN is asserted.

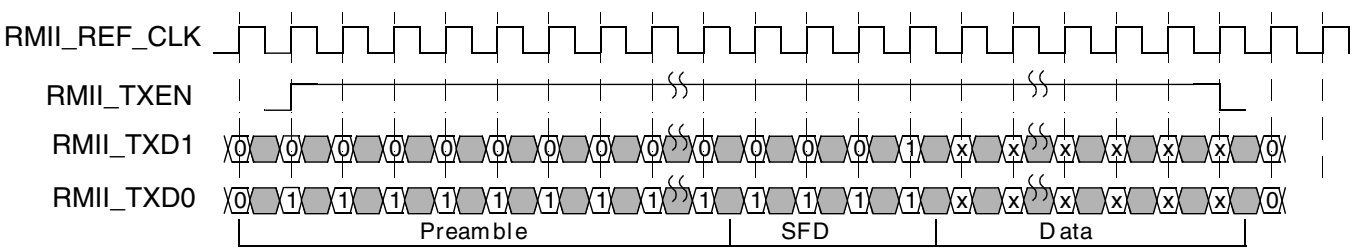


Figure 23-121. RMII transmit operation

23.6.18.2 RGMII interface

In RGMII modes, the data and control information is multiplexed by taking advantage of both edges of the reference clocks.

The data signals contain the lower four data bits on the rising edge and the upper four bits on the falling edge. The control signals are multiplexed into a single clock cycle using the same technique.

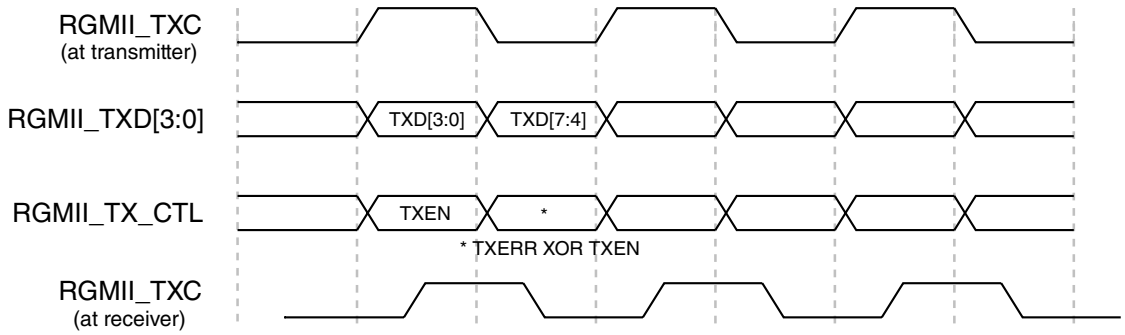


Figure 23-122. RGMII transmit operation

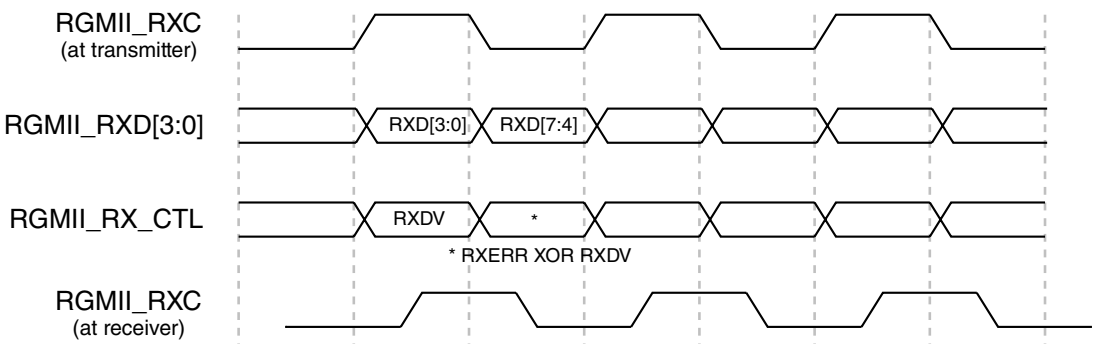


Figure 23-123. RGMII receive operation

23.6.18.3 MII Interface — transmit

On transmit, all data transfers are synchronous to MII_TXCLK rising edge. The MII data enable signal MII_TXEN is asserted to indicate the start of a new frame, and remains asserted until the last byte of the frame is present on the MII_TXD[3:0] bus.

Between frames, MII_TXEN remains deasserted.

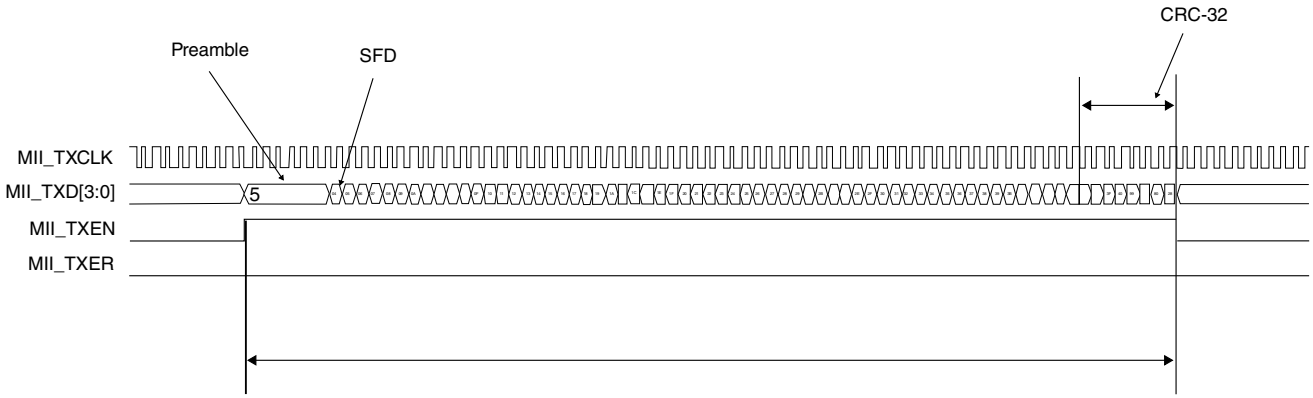


Figure 23-124. MII transmit operation

If a frame is received on the FIFO interface with an error (for example, RxBD[ME] set) the frame is subsequently transmitted with the MII_TXER error signal for one clock cycle at any time during the packet transfer.

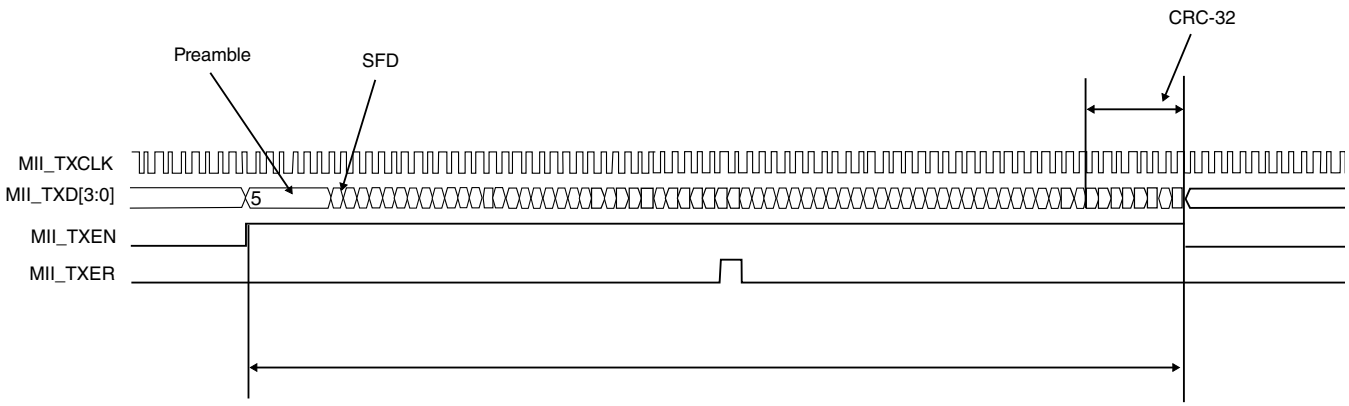


Figure 23-125. MII transmit operation — errored frame

23.6.18.3.1 Transmit with collision — half-duplex

When a collision is detected during a frame transmission (MII_COL asserted), the MAC stops the current transmission, sends a 32-bit jam pattern, and re-transmits the current frame.

(See Collision detection in half-duplex mode for details)

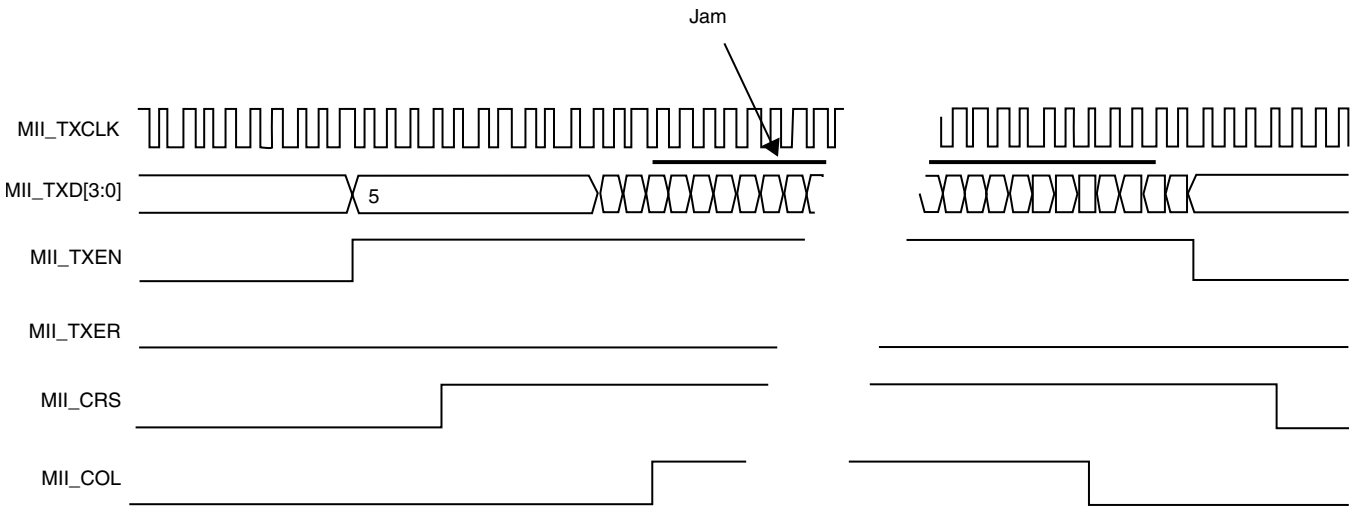


Figure 23-126. MII transmit operation — transmission with collision

23.6.18.4 MII interface — receive

On receive, all signals are sampled on the MII_RXCLK rising edge. The MII data enable signal, MII_RXDV, is asserted by the PHY to indicate the start of a new frame and remains asserted until the last byte of the frame is present on MII_RXD[3:0] bus.

Between frames, MII_RXDV remains deasserted.

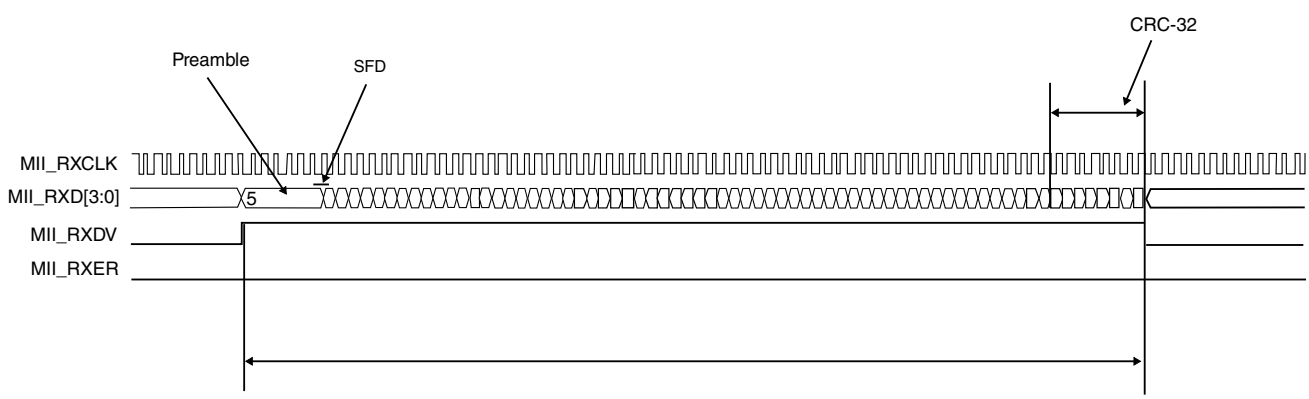


Figure 23-127. MII receive operation

If the PHY detects an error on the frame received from the line, the PHY asserts the MII error signal, MII_RXER, for at least one clock cycle at any time during the packet transfer.

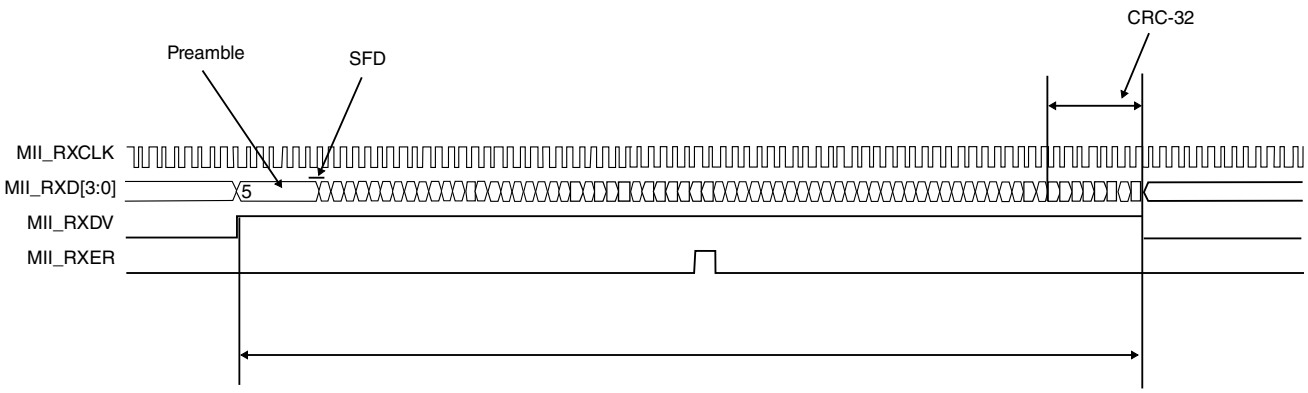


Figure 23-128. MII receive operation — errored frame

A frame received on the MII interface with a PHY error indication is subsequently transferred on the FIFO interface with RxBD[ME] set.

Chapter 24
Enhanced Periodic Interrupt Timer (EPIT)

24.1 Overview

EPIT is a 32-bit set-and-forget timer that is capable of providing precise interrupts at regular intervals with minimal processor intervention. EPIT begins counting after it is enabled by software.

The following figure shows the EPIT block diagram.

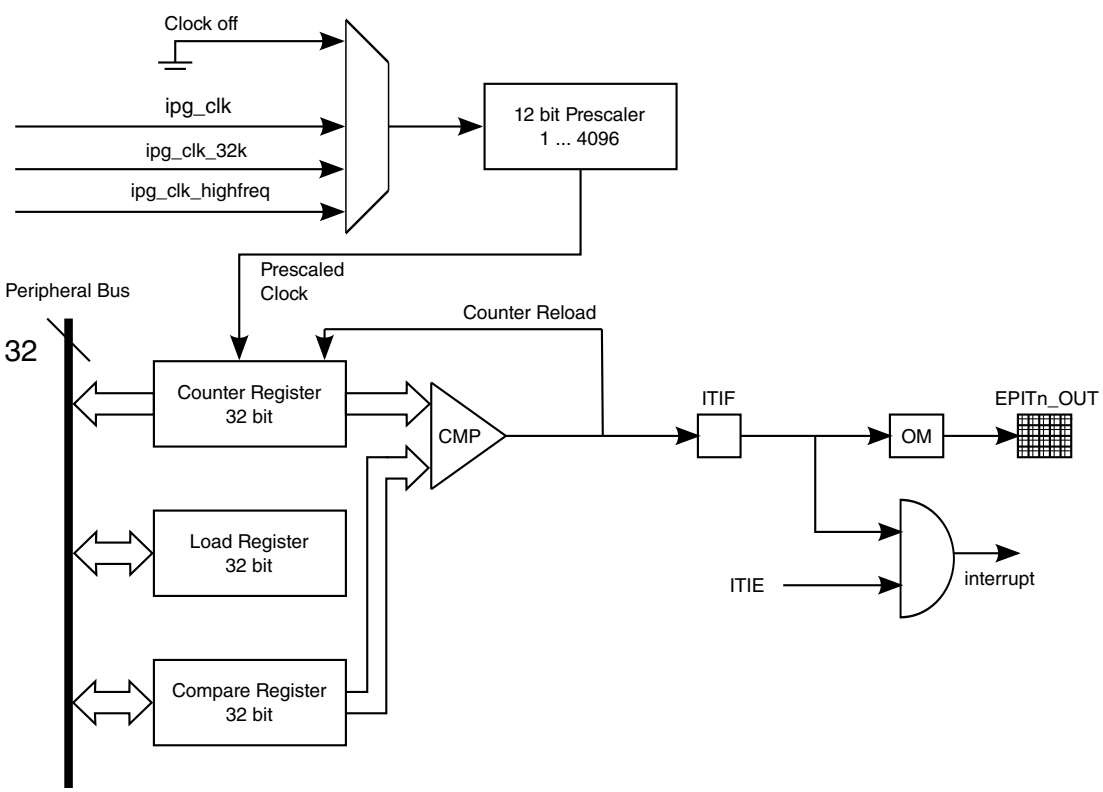


Figure 24-1. EPIT block diagram

24.1.1 EPIT features

EPIT has the following key features:

- 32-bit down counter with clock source selection
- 12-bit prescaler for division of input clock frequency
- Counter value that can be programmed on the fly
- Can be programmed to be active during low-power and debug modes
- Interrupt generation when counter reaches the compare value

24.1.2 EPIT modes and operations

EPIT supports the following modes: set-and-forget and free running. See the following sections for more information.

- Operating in set-and-forget mode
- Operating in free-running mode

See Operations for a description of the operations that EPIT supports.

24.2 External signals

The following table describes EPIT's I/O signals.

Table 24-1. EPIT External Signals

Signal	Description	Pad	Mode	Direction
EPIT1_OUT	Output 1 pin at chip boundary for indicating the occurrence of an output compare event through a specified transition.	EIM_D19	ALT6	O
		GPIO_0	ALF4	
		GPIO_7	ALT2	
EPIT2_OUT	Output 2 pin at chip boundary for indicating the occurrence of an output compare event through a specified transition.	EIM_D20	ALT6	O
		GPIO_8	ALT2	

24.3 Clocks

The table found here describes the clock sources for EPIT.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 24-2. EPIT Clocks

Clock name	Clock Root	Description
ipg_clk	ipg_clk_root	Peripheral clock
ipg_clk_32k	ckil_sync_clk_root	Low-frequency reference clock (32kHz)
ipg_clk_highfreq	perclk_clk_root	High-frequency reference clock
ipg_clk_s	ipg_clk_root	Peripheral access clock

The clock that feeds the prescaler can be selected from among the following sources:

- **High-frequency reference clock (ipg_clk_highfreq)**

  This clock is provided by the Clock Control Module (CCM). This clock remains on during low-power mode when the peripheral clock is turned off, allowing EPIT to use this clock in low-power mode. In normal mode, the CCM synchronizes this clock to ahb_clk; in low-power mode, CCM switches to an unsynchronized version.

- **Low-frequency reference clock (ipg_clk_32k)**

  This 32 kHz reference clock is provided by the CCM. This clock remains on in low-power mode when the peripheral clock is turned off, so EPIT can use this clock during low-power mode. In normal mode, the CCM synchronizes this clock to ahb_clk; in low-power mode, CCM switches to an unsynchronized version. This clock is derived from the external 32kHz crystal.

- **Peripheral clock (ipg_clk)**

  This is the peripheral clock (PER Clock) which is provided (and optionally gated) by the CCM. This clock is typically used in normal operations. In low-power modes, if the EPIT is programmed to be disabled (via STOPEN or WAITEN), then the peripheral clock can be switched off.

The clock input source is determined by the CLKSRC field in the control register. The clock input to the prescaler can also be disabled by setting CLKSRC to 0b00. **This field value should only be changed after first disabling the EPIT by clearing the EN bit in the EPIT_EPITCR**. For other programming requirements that apply while changing clock source, refer section Change of Clock Source.

The PRESCALER field in the control register is used to select the divide ratio of the input clock that drives the main counter. The prescaler can divide the input clock by a value between 1 and 4096. A change in the value of the PRESCALER field is immediately reflected on its output clock frequency. The following figure shows the timing for a change in the prescaler value.

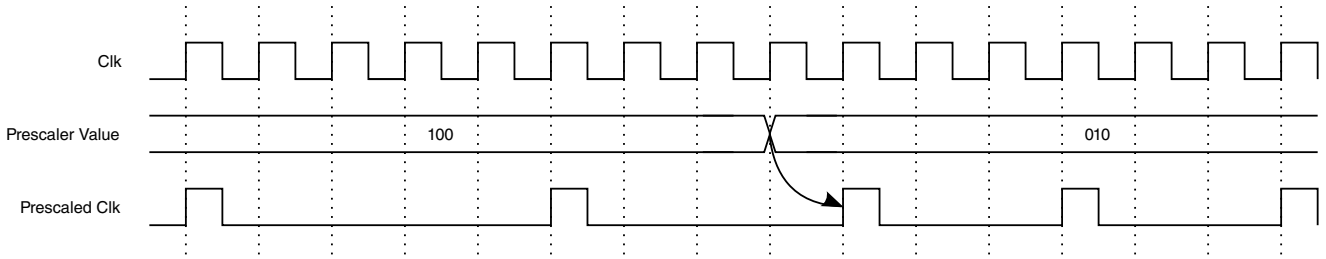


Figure 24-2. Prescaler Value Change Diagram

24.4 Functional Description

This section provides a complete functional description of the block.

24.4.1 Operating modes

EPIT can operate in either set-and-forget or free-running mode. Use EPIT_CR[RLD] to select the desired mode.

24.4.1.1 Operating in set-and-forget mode

To select this mode of operation, set the RLD bit in the control register (EPIT_CR).

In this mode, the counter obtains its data from the load register (EPIT_LR); it cannot be written to directly from the block data bus. Whenever the counter reaches zero, the value in EPIT_LR is loaded into the counter. This value is then decremented to zero.

To directly initialize the counter instead of waiting for the count to reach zero, set the EPIT counter overwrite enable bit (EPIT_CR[IOVW]) and write to EPIT_LR with the required initialization value.

24.4.1.2 Operating in free-running mode

To select this mode of operation, clear the RLD bit.

In this mode, the counter rolls over from 0000 0000h to FFFF FFFFh without reloading from the modulus register. After rolling over, the counter continues counting down.

To directly initalize the counter, set the EPIT counter overwrite enable bit (EPIT_EPITCL[IOVW]) and write to EPIT_EPITLR with the required initialization value.

24.4.2 Operations

EPIT has a single 32-bit down counter, which starts counting when the block is enabled by software.

The start value of the counter is loaded from the EPIT load register, which can be written to at any time by the processor. The value in the compare register determines the time that the interrupt occurs.

When EPIT is disabled (EN = 0), both the main counter and the prescaler counter freeze their count at their current count values. When EPIT is re-enabled (EN = 1), the ENMOD bit, which is a RW bit, decides the counter value:

- If ENMOD is set, the main counter is loaded with the load value (If RLD = 1)/ FFFF FFFFh (If RLD = 0) and the prescaler counter is reset (000h).
- If ENMOD is cleared, both main counter and prescaler counter restart counting from their frozen values.

If EPIT is programmed to be disabled in a low-power mode (STOP/WAIT), both the main counter and the prescaler counter freeze at their current count values when EPIT enters low-power mode. When EPIT exits the low-power mode, both the main counter and the prescaler counter start counting from their frozen values regardless of the ENMOD bit.

A hardware reset resets all EPIT registers to their respective reset values. There is a software reset which has the same effect on all registers except for the EN, ENMOD, STOPEN and WAITEN bits in the control register. The state of these bits are not affected by software reset. A software reset can be asserted even when the EPIT is disabled.

24.4.3 Compare Event

When the programmed value of EPIT_EPITCMPR matches the value in EPIT_EPITCNR a compare status flag is set, and an interrupt is generated if the OCIEN bit is set in the control register.

The compare output pin is set, cleared, toggled, or not affected at all depending on the setting of the output mode (OM) bits in the control register. If an interrupt is required at rollover (when the counter value reaches 0x0000_0000 and the new value is loaded) then the compare register value should be set equal to the load register value in set-and-forget mode, or equal to 0xFFFF_FFFF in free-running mode.

The following figure shows the timing for a compare event and interrupt.

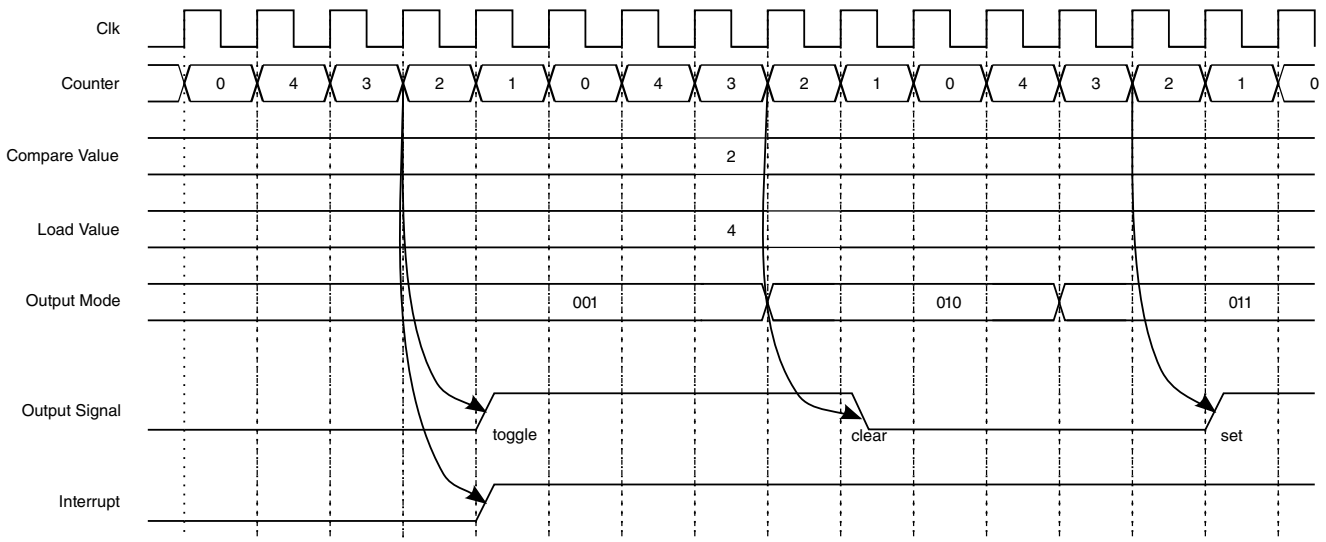


Figure 24-3. Compare Event and Interrupt Timing Diagram

EPIT will generate a compare event in the next count if the EPITx_CNR from the previous count equals the new EPITx_CMPR configured before re-enabling the EPIT in the next count. Even in case a new start counter value was updated in EPITx_LR before re-enabling the EPIT for the next round. To avoid this, configure the EPITx_CMPR to previous EPITx_CNR+1. Or, in set and forget mode, configure EPITx_LR with IOVW=1, before disabling EPIT. Also can do an extra disable/enable iteration to clear OCIF and update EPITx_CNR.

24.4.3.1 Counter Value Overwrite

The EPIT counter value can be overwritten to acquire a desired value at any point of time. The procedure for this is to set the IOVW bit in the control register and then write the desired value into the load register.

This results in the load register acquiring that value and also the counter being overwritten with it. If the EPIT is running the counter resumes counting from the overwritten value.

24.4.3.2 Low-Power Mode Behavior

The EPIT timer's behavior in low-power modes depends on which clock source is being used.

If the selected clock source is available and the corresponding low-power enable bit is set, then the EPIT continues to function in the low-power mode. If the EPIT is programmed to be disabled in a low-power mode (STOP/WAIT), then main counter and the prescaler counter freeze at the current count values when the EPIT enters low-power mode. When the EPIT exits the low-power mode, both main counter and prescaler counter start counting from their frozen values irrespective of the ENMOD bit.

24.4.3.3 Debug Mode Behavior

In debug mode, the user has the option to run or halt the EPIT timers. If the DBGEN bit is reset in the EPIT Control Register, the timer is halted.

When debug mode is exited, the timer operation reverts to what it was prior to entering debug mode.

24.5 Initialization/ Application Information

24.5.1 Change of Clock Source

The CLKSRC field in EPIT_EPITCR determines the clock source. This field value should be changed only after disabling the EPIT (EN = 0).

Below is the software sequence which must be followed while changing clock source.

1. Disable the EPIT - set EN=0 in EPIT_EPITCR.
2. Disable EPIT ouput - program OM=00 in the EPIT_EPITCR.
3. Disable EPIT interrupts.
4. Program CLKSRC to desired clock source in EPIT_EPITCR.
5. Clear the EPIT status register (EPIT_EPITSR), that is, write "1" to clear (w1c).
6. Set ENMOD= 1 in the EPIT_EPITCR, to bring the EPIT Counter to defined state (EPIT_EPITLR value or 0xFFFF_FFFF).
7. Enable EPIT - set (EN=1) in the EPIT_EPITCR
8. Enable the EPIT interrupts.

24.6 EPIT Memory Map/Register Definition

The EPIT includes five user-accessible 32-bit registers. The following table summarizes these registers and their addresses.

Peripheral bus write access to the EPIT control register (EPITCR) and the EPIT load register (EPITLR) results in one cycle of wait state, while other valid peripheral bus accesses are with 0 wait state.

EPIT memory map

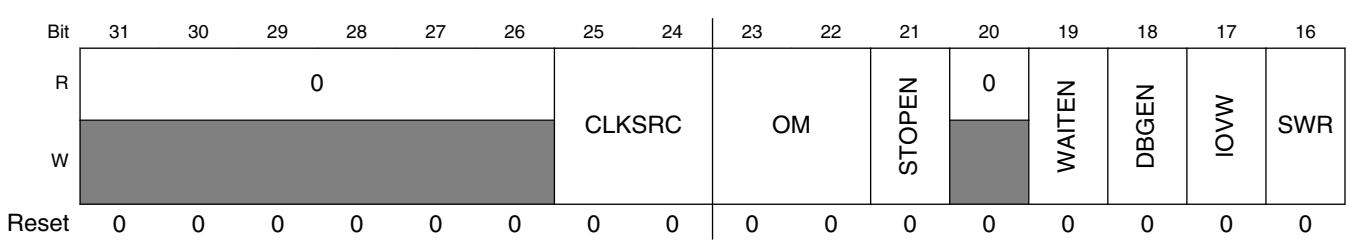
Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20D_0000	Control register (EPIT1_CR)	32	R/W	0000_0000h	24.6.1/1214
20D_0004	Status register (EPIT1_SR)	32	R/W	0000_0000h	24.6.2/1217
20D_0008	Load register (EPIT1_LR)	32	R/W	FFFF_FFFFh	24.6.3/1217
20D_000C	Compare register (EPIT1_CMPR)	32	R/W	0000_0000h	24.6.4/1218
20D_0010	Counter register (EPIT1_CNR)	32	R	FFFF_FFFFh	24.6.5/1218
20D_4000	Control register (EPIT2_CR)	32	R/W	0000_0000h	24.6.1/1214
20D_4004	Status register (EPIT2_SR)	32	R/W	0000_0000h	24.6.2/1217
20D_4008	Load register (EPIT2_LR)	32	R/W	FFFF_FFFFh	24.6.3/1217
20D_400C	Compare register (EPIT2_CMPR)	32	R/W	0000_0000h	24.6.4/1218
20D_4010	Counter register (EPIT2_CNR)	32	R	FFFF_FFFFh	24.6.5/1218

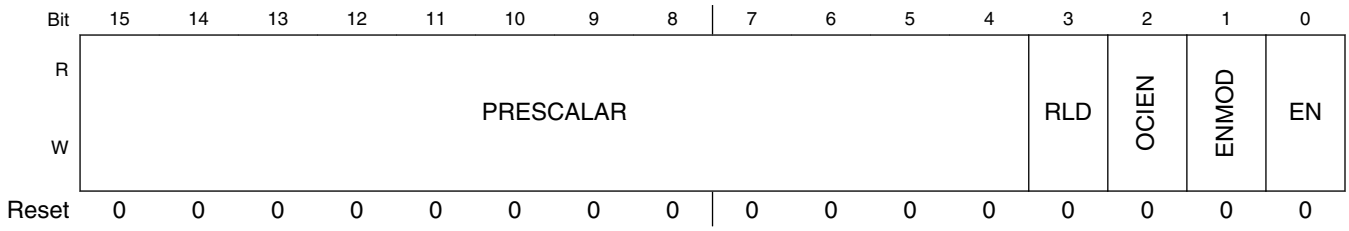
24.6.1 Control register (EPITx_CR)

The EPIT control register (EPIT_CR) is used to configure the operating settings of the EPIT. It contains the clock division prescaler value and also the interrupt enable bit. Additionally, it contains other control bits which are described below.

Peripheral Bus Write access to EPIT Control Register (EPIT_CR) results in one cycle of the wait state, while other valid peripheral bus accesses are with 0 wait state.

Address: Base address + 0h offset





EPITx_CR field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25–24 CLKSRC	Select clock source These bits determine which clock input is to be selected for running the counter. This field value should only be changed when the EPIT is disabled by clearing the EN bit in this register. For other programming requirements while changing clock source, refer to Change of Clock Source. 00 Clock is off 01 Peripheral clock 10 High-frequency reference clock 11 Low-frequency reference clock
23–22 OM	EPIT output mode.This bit field determines the mode of EPIT output on the output pin. 00 EPIT output is disconnected from pad 01 Toggle output pin 10 Clear output pin 11 Set output pin
21 STOPEN	EPIT stop mode enable. This read/write control bit enables the operation of the EPIT during stop mode. This bit is reset by a hardware reset and unaffected by software reset. 0 EPIT is disabled in stop mode 1 EPIT is enabled in stop mode
20 Reserved	This read-only field is reserved and always has the value 0.
19 WAITEN	This read/write control bit enables the operation of the EPIT during wait mode. This bit is reset by a hardware reset. A software reset does not affect this bit. 0 EPIT is disabled in wait mode 1 EPIT is enabled in wait mode
18 DBGEN	This bit is used to keep the EPIT functional in debug mode. When this bit is cleared, the input clock is gated off in debug mode.This bit is reset by hardware reset. A software reset does not affect this bit. 0 Inactive in debug mode 1 Active in debug mode
17 IOVW	EPIT counter overwrite enable. This bit controls the counter data when the modulus register is written. When this bit is set, all writes to the load register overwrites the counter contents and the counter starts subsequently counting down from the programmed value. 0 Write to load register does not result in counter value being overwritten. 1 Write to load register results in immediate overwriting of counter value.

Table continues on the next page...

EPITx_CR field descriptions (continued)

Field	Description
16 SWR	Software reset. The EPIT is reset when this bit is set to 1. It is a self clearing bit. This bit is set when the block is in reset state and is cleared when the reset procedure is over. Setting this bit resets all the registers to their reset values, except for the EN, ENMOD, STOPEN, WAITEN and DBGEN bits in this control register 0 EPIT is out of reset 1 EPIT is undergoing reset
15–4 PRESCALAR	Counter clock prescaler value. This bit field determines the prescaler value by which the clock is divided before it goes to the counter 0x000 Divide by 1 0x001 Divide by 2... 0xFFF Divide by 4096
3 RLD	Counter reload control. This bit is cleared by hardware reset. It decides the counter functionality, whether to run in free-running mode or set-and-forget mode. 0 When the counter reaches zero it rolls over to 0xFFFF_FFFF (free-running mode) 1 When the counter reaches zero it reloads from the modulus register (set-and-forget mode)
2 OCIEN	Output compare interrupt enable. This bit enables the generation of interrupt on occurrence of compare event. 0 Compare interrupt disabled 1 Compare interrupt enabled
1 ENMOD	EPIT enable mode. When EPIT is disabled (EN=0), both main counter and prescaler counter freeze their count at current count values. ENMOD bit is a r/w bit that determines the counter value when the EPIT is enabled again by setting EN bit. If ENMOD bit is set, then main counter is loaded with the load value (If RLD=1)/ 0xFFFF_FFFF (If RLD=0) and prescaler counter is reset, when EPIT is enabled (EN=1). If ENMOD is programmed to 0 then both main counter and prescaler counter restart counting from their frozen values when EPIT is enabled (EN=1). If EPIT is programmed to be disabled in a low-power mode (STOP/WAIT/ DEBUG), then both the main counter and the prescaler counter freeze at their current count values when EPIT enters low-power mode. When EPIT exits the low-power mode, both main counter and prescaler counter start counting from their frozen values irrespective of the ENMOD bit. This bit is reset by a hardware reset. A software reset does not affect this bit. 0 Counter starts counting from the value it had when it was disabled. 1 Counter starts count from load value (RLD=1) or 0xFFFF_FFFF (If RLD=0)
0 EN	This bit enables the EPIT. EPIT counter and prescaler value when EPIT is enabled (EN = 1), is dependent upon ENMOD and RLD bit as described for ENMOD bit. It is recommended that all registers be properly programmed before setting this bit. This bit is reset by a hardware reset. A software reset does not affect this bit. 0 EPIT is disabled 1 EPIT is enabled

24.6.2 Status register (EPITx_SR)

The EPIT status register (EPIT_SR) has a single status bit for the output compare event. The bit is a write 1 to clear bit.

Address: Base address + 4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															OCIF
W																w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EPITx_SR field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 OCIF	Output compare interrupt flag. This bit is the interrupt flag that is set when the content of counter equals the content of the compare register (EPIT_CMPR). The bit is a write 1 to clear bit. 0 Compare event has not occurred 1 Compare event occurred

24.6.3 Load register (EPITx_LR)

The EPIT load register (EPIT_LR) contains the value that is to be loaded into the counter when EPIT counter reaches zero if the RLD bit in EPIT_CR is set. If the IOVW bit in the EPIT_CR is set then a write to this register overwrites the value of the EPIT counter register in addition to updating this registers value. This overwrite feature is active even if the RLD bit is not set.

Address: Base address + 8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	LOAD																															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

EPITx_LR field descriptions

Field	Description
LOAD	Load value. Value that is loaded into the counter at the start of each count cycle.

24.6.4 Compare register (EPITx_CMPR)

The EPIT compare register (EPIT_CMPR) holds the value that determines when a compare event is generated.

Address: Base address + Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	COMPARE																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

EPITx_CMPR field descriptions

Field	Description
COMPARE	Compare Value. When the counter value equals this bit field value a compare event is generated.

24.6.5 Counter register (EPITx_CNR)

The EPIT counter register (EPIT_CNR) contains the current count value and can be read at any time without disturbing the counter. This is a read-only register and any attempt to write into it generates a transfer error. But if the IOVW bit in EPIT_CR is set, the value of this register can be overwritten with a write to EPIT_LR. This change is reflected when this register is subsequently read.

Address: Base address + 10h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COUNT																															
W																																
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

EPITx_CNR field descriptions

Field	Description
COUNT	Counter value. This contains the current value of the counter.

Chapter 25
Enhanced Serial Audio Interface (ESAI)

25.1 Overview

The Enhanced Serial Audio Interface (ESAI) provides a full-duplex serial port for serial communication with a variety of serial devices, including industry-standard codecs, Sony/Phillips Digital Interface (SPDIF) transceivers, and other DSPs.

The ESAI consists of independent transmitter and receiver sections, each section with its own clock generator. It is a superset of the 56300 Family ESSI peripheral and of the 56000 Family SAI peripheral.

All serial transfers in the module are synchronized to a clock. Additional synchronization signals are used to delineate the word frames. The normal mode of operation is used to transfer data at a periodic rate, one word per period. The network mode is similar in that it is also intended for periodic transfers; however, it supports up to 32 words (time slots) per period. This mode can be used to build time division multiplexed (TDM) networks. In contrast, the on-demand mode is intended for non-periodic transfers of data and to transfer data serially at high speed when the data becomes available.

The following figure shows the ESAI block diagram.

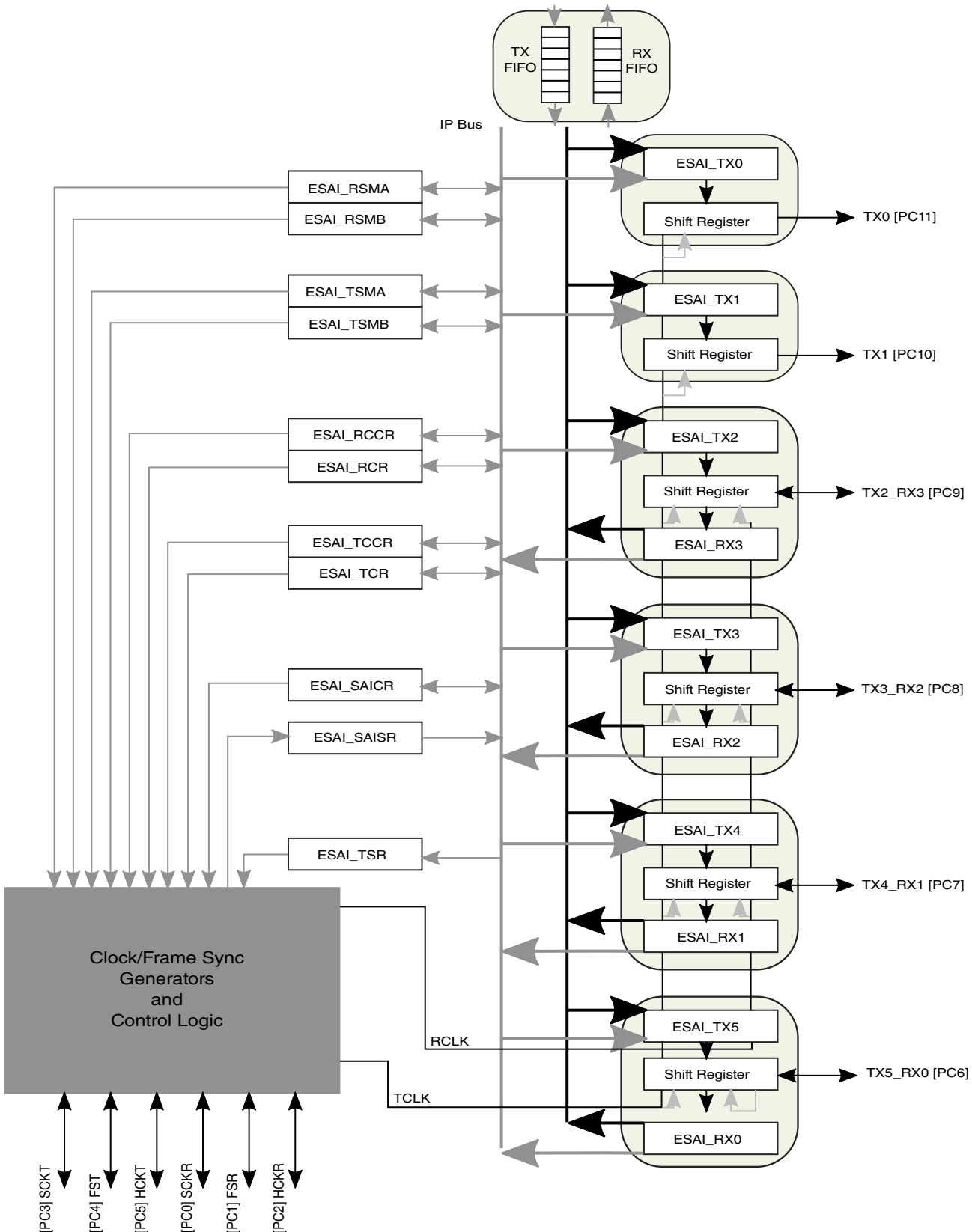


Figure 25-1. ESAI Block Diagram

25.1.1 Features

- Independent (asynchronous mode) or shared (synchronous mode) transmit and receive sections with separate or shared internal/external clocks and frame syncs, operating in Master or Slave mode.
- Up to six transmitters and four receivers with TX2_RX3, TX3_RX2, TX4_RX1, and TX5_RX0 pins shared by transmitters 2 to 5 and receivers 0 to 3. TX0 AND TX1 pins are used by transmitters 0 and 1 only.
- Programmable data interface modes such as I2S, LSB aligned, MSB aligned
- Programmable word length (8, 12, 16, 20 or 24bits)
- Flexible selection between system clock or external oscillator as input clock source, programmable internal clock divider and frame sync generation
- AC97 support
- Time Slot Mask Registers for reduced ARM platform overhead (for both Transmit and Receive)
- 128-word Transmit FIFO shared by six transmitters
- 128-word Receive FIFO shared by four receivers

25.1.2 Modes of Operation

ESAI has three basic operating modes and many data/operation formats.

ESAI operating mode are selected by the ESAI control registers (ESAI_TCCR, ESAI_TCR, ESAI_RCCR, ESAI_RCR, and ESAI_SAICR). The main operating modes are described in the following section.

25.1.2.1 Normal/Network/On-Demand Mode Selection

Selecting between the normal mode and network mode is accomplished by clearing or setting the TMOD0-TMOD1 bits in the ESAI_TCR register for the transmitter section, as well as in the RMOD0-RMOD1 bits in the ESAI_RCR register for the receiver section.

For normal mode, the ESAI functions with one data word of I/O per frame (per enabled transmitter or receiver). The normal mode is typically used to transfer data to or from a single device.

For the network mode, 2 to 32 time slots per frame may be selected. During each frame, 0 to 32 data words of I/O may be received or transmitted. In either case, the transfers are periodic. The frame sync signal indicates the first time slot in the frame. Network mode is typically used in time division multiplexed (TDM) networks of codecs, DSPs with multiple words per frame, or multi-channel devices.

Selecting the network mode and setting the frame rate divider to zero (DC=00000) selects the on-demand mode. This special case does not generate a periodic frame sync. A frame sync pulse is generated only when data is available to transmit. The on-demand mode requires that the transmit frame sync be internal (output) and the receive frame sync be external (input). Therefore, for simplex operation, the synchronous mode could be used; however, for full-duplex operation, the asynchronous mode must be used. Data transmission that is data driven is enabled by writing data into each TX. Although the ESAI is double buffered, only one word can be written to each TX, even if the transmit shift register is empty. The receive and transmit interrupts function as usual using TDE and RDF; however, transmit underruns are impossible for on-demand transmission and are disabled.

25.1.2.2 Synchronous/Asynchronous Operating Modes

The transmit and receive sections of the ESAI may be synchronous or asynchronous, that is, the transmitter and receiver sections may use common clock and synchronization signals (synchronous operating mode), or they may have their own separate clock and sync signals (asynchronous operating mode).

The SYN bit in the ESAI_SAICR register selects synchronous or asynchronous operation. Because the ESAI is designed to operate either synchronously or asynchronously, separate receive and transmit interrupts are provided.

When SYN is cleared, the ESAI transmitter and receiver clocks and frame sync sources are independent. If SYN is set, the ESAI transmitter and receiver clocks and frame sync come from the transmitter section (either external or internal sources).

Data clock and frame sync signals can be generated internally by the ARM Core or may be obtained from external sources. If internally generated, the ESAI clock generator is used to derive high frequency clock, bit clock and frame sync signals from the ARM Core internal system clock.

25.1.2.3 Frame Sync Selection

The frame sync can be either a bit-long or word-long signal.

The transmitter frame format is defined by the TFSL bit in the ESAI_TCR register. The receiver frame format is defined by the RFSL bit in the ESAI_RCR register.

1. In the word-long frame sync format, the frame sync signal is asserted during the entire word data transfer period. This frame sync length is compatible with codecs, SPI serial peripherals, serial A/D and D/A converters, shift registers and telecommunication PCM serial I/O.
2. In the bit-long frame sync format, the frame sync signal is asserted for one bit clock immediately before the data transfer period. This frame sync length is compatible with Intel and National components, codecs and telecommunication PCM serial I/O.

The relative timing of the word length frame sync as referred to the data word is specified by the TFSR bit in the ESAI_TCR register for the transmitter section and by the RFSR bit in the ESAI_RCR register for the receive section. The word length frame sync may be generated (or expected) with the first bit of the data word, or with the last bit of the previous word. TFSR and RFSR are ignored when a bit length frame sync is selected.

Polarity of the frame sync signal may be defined as positive (asserted high) or negative (asserted low). The TFSP bit in the ESAI_TCCR register specifies the polarity of the frame sync for the transmitter section. The RFSP bit in the ESAI_RCCR register specifies the polarity of the frame sync for the receiver section.

The ESAI receiver looks for a receive frame sync leading edge (trailing edge if RFSP is set) only when the previous frame is completed. If the frame sync goes high before the frame is completed (or before the last bit of the frame is received in the case of a bit frame sync or a word length frame sync with RFSR set), the current frame sync is not recognized, and the receiver is internally disabled until the next frame sync. Frames do not have to be adjacent, that is, a new frame sync does not have to immediately follow the previous frame. Gaps of arbitrary periods can occur between frames. Enabled transmitters are tri-stated during these gaps.

When operating in the synchronous mode (SYN=1), all clocks including the frame sync are generated by the transmitter section.

25.1.2.4 Shift Direction Selection

Some data formats, such as those used by codecs, specify MSB first while other data formats, such as the AES-EBU digital audio interface, specify LSB first.

The MSB/LSB first selection is made by programming RSHFD bit in the ESAI_RCR register for the receiver section and by programming the TSHFD bit in the ESAI_TCR register for the transmitter section.

25.2 External Signals

Three to twelve pins are required for operation, depending on the operating mode selected and the number of transmitters and receivers enabled.

The TX0 and TX1 pins are used by transmitters 0 and 1 only. The TX2_RX3, TX3_RX2, TX4_RX1, and TX5_RX0 pins are shared by transmitters 2 to 5 with receivers 0 to 3. The actual mode of operation is selected under software control. All transmitters operate fully synchronized under control of the same transmitter clock signals. All receivers operate fully synchronized under control of the same receiver clock signals.

The following table describes the external signals of ESAI:

Table 25-1. ESAI External Signals

Signal	Description	Pad	Mode	Direction
ESAI_RX_CLK	RX serial bit clock for the ESAI interface. The direction can be programmed.	ENET_MDIO	ALT2	IO
		GPIO_1	ALT0	
ESAI_RX_FS	RX frame sync signal for the ESAI interface.	ENET_REF_CLK	ALT2	IO
		GPIO_9	ALT0	
ESAI_RX_HF_CLK	RX high frequency clock for the ESAI interface.	ENET_RX_ER	ALT2	IO
		GPIO_3	ALT0	
ESAI_TX0	Used for transmitting data from the ESAI_TX0 serial transmit shift register.	GPIO_17	ALT0	IO
		NANDF_CS2	ALT2	
ESAI_TX1	Used for transmitting data from the ESAI_TX1 serial transmit shift register.	GPIO_18	ALT0	IO
		NANDF_CS3	ALT2	
ESAI_TX2_RX3	Used as TX2 for transmitting data from the ESAI_TX2 serial transmit shift register when programmed as a transmitter pin Used as the RX3 signal for receiving serial data to the ESAI_RX3 serial receive shift register when programmed as a receiver pin	ENET_TXD1	ALT2	IO
		GPIO_5	ALT0	
ESAI_TX3_RX2	Used as TX3 for transmitting data from the ESAI_TX3 serial transmit shift register when programmed as a transmitter pin Used as the RX2 signal for receiving serial data to the ESAI_RX2 serial receive shift register when programmed as a receiver pin	ENET_TX_EN	ALT2	IO
		GPIO_16	ALT0	

Table continues on the next page...

Table 25-1. ESAI External Signals (continued)

Signal	Description	Pad	Mode	Direction
ESAI_TX4_RX1	Used as TX4 for transmitting data from the ESAI_TX4 serial transmit shift register when programmed as a transmitter pin Used as the RX1 signal for receiving serial data to the ESAI_RX1 serial receive shift register when programmed as a receiver pin	ENET_TXD0	ALT2	IO
		GPIO_7	ALT0	
ESAI_TX5_RX0	Used as TX5 for transmitting data from the ESAI_TX5 serial transmit shift register when programmed as a transmitter pin Used as the RX0 signal for receiving serial data to the ESAI_RX0 serial receive shift register when programmed as a receiver pin	ENET_MDC	ALT2	IO
		GPIO_8	ALT0	
ESAI_TX_CLK	TX serial bit clock for the ESAI interface. The direction can be programmed.	ENET_CRS_DV	ALT2	IO
		GPIO_6	ALT0	
ESAI_TX_FS	Frame sync for both the transmitters and receivers in the synchronous mode (SYN=1) and for the transmitters only in asynchronous mode	ENET_RXD1	ALT2	IO
		GPIO_2	ALT0	
ESAI_TX_HF_CLK	TX high frequency clock for the ESAI interface.	ENET_RXD0	ALT2	IO
		GPIO_4	ALT0	

25.2.1 Serial Transmit 0 Data Pin

TX0 is used for transmitting data from the ESAI_TX0 serial transmit shift register.

TX0 is an output when data is being transmitted from the ESAI_TX0 shift register. In the on-demand mode with an internally generated bit clock, the TX0 pin becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX0 may be programmed as a disconnected pin (PC11) when the ESAI TX0 function is not being used. (See Table 25-48.)

25.2.2 Serial Transmit 1 Data Pin

TX1 is used for transmitting data from the ESAI_TX1 serial transmit shift register.

TX1 is an output when data is being transmitted from the ESAI_TX1 shift register. In the on-demand mode with an internally generated bit clock, the TX1 pin becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX1 may be programmed as a disconnected pin (PC10) when the ESAI TX1 function is not being used. (See Table 25-48.)

25.2.3 Serial Transmit 2/Receive 3 Data Pin

TX2_RX3 is used as the TX2 for transmitting data from the ESAI_TX2 serial transmit shift register when programmed as a transmitter pin, or as the RX3 signal for receiving serial data to the ESAI_RX3 serial receive shift register when programmed as a receiver pin.

TX2_RX3 is an input when data is being received by the ESAI_RX3 shift register. TX2_RX3 is an output when data is being transmitted from the ESAI_TX2 shift register. In the on-demand mode with an internally generated bit clock, the TX2_RX3 pin becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX2_RX3 may be programmed as a disconnected pin (PC9) when the ESAI TX2 and RX3 functions are not being used. (See Table 25-48.)

25.2.4 Serial Transmit 3/Receive 2 Data Pin

TX3_RX2 is used as the TX3 signal for transmitting data from the ESAI_TX3 serial transmit shift register when programmed as a transmitter pin, or as the RX2 signal for receiving serial data to the ESAI_RX2 serial receive shift register when programmed as a receiver pin.

TX3_RX2 is an input when data is being received by the ESAI_RX2 shift register. TX3_RX2 is an output when data is being transmitted from the ESAI_TX3 shift register. In the on-demand mode with an internally generated bit clock, the TX3_RX2 pin

becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX3_RX2 may be programmed as a disconnected pin (PC8) when the ESAI TX3 and RX2 functions are not being used. (See Table 25-48.)

25.2.5 Serial Transmit 4/Receive 1 Data Pin

TX4_RX1 is used as the TX4 signal for transmitting data from the ESAI_TX4 serial transmit shift register when programmed as transmitter pin, or as the RX1 signal for receiving serial data to the RX1 serial receive shift register when programmed as a receiver pin.

TX4_RX1 is an input when data is being received by the ESAI_RX1 shift register. TX4_RX1 is an output when data is being transmitted from the ESAI_TX4 shift register. In the on-demand mode with an internally generated bit clock, the TX4_RX1 pin becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX4_RX1 may be programmed as a disconnected pin (PC7) when the ESAI TX4 and RX1 functions are not being used. (See Table 25-48.)

25.2.6 Serial Transmit 5/Receive 0 Data Pin

TX5_RX0 is used as the TX5 signal for transmitting data from the ESAI_TX5 serial transmit shift register when programmed as transmitter pin, or as the RX0 signal for receiving serial data to the ESAI_RX0 serial shift register when programmed as a receiver pin.

TX5_RX0 is an input when data is being received by the ESAI_RX0 shift register. TX5_RX0 is an output when data is being transmitted from the ESAI_TX5 shift register. In the on-demand mode with an internally generated bit clock, the TX5_RX0 pin becomes high impedance for a full clock period after the last data bit has been transmitted, assuming another data word does not follow immediately. If a data word follows immediately, there is no high-impedance interval.

TX5_RX0 may be programmed as a disconnected pin (PC6) when the ESAI TX5 and RX0 functions are not being used. (See Table 25-48.)

25.2.7 Receiver Serial Clock

SCKR is a bidirectional pin providing the receivers serial bit clock for the ESAI interface.

The direction of this pin is determined by the RCKD bit in the ESAI_RCCR register. The SCKR operates as a clock input or output used by all the enabled receivers in the asynchronous mode (SYN = 0), or as serial flag 0 pin in the synchronous mode (SYN = 1).

When this pin is configured as serial flag pin, its direction is determined by the RCKD bit in the ESAI_RCCR register. When configured as the output flag OF0, this pin reflects the value of the OF0 bit in the ESAI_SAICR register, and the data in the OF0 bit shows up at the pin synchronized to the frame sync being used by the transmitter and receiver sections. When this pin is configured as the input flag IF0, the data value at the pin is stored in the IF0 bit in the ESAI_SAISR register, synchronized by the frame sync in normal mode or the slot in network mode.

SCKR may be programmed as a disconnected pin (PC0) when the ESAI SCKR function is not being used. (See Table 25-48.)

NOTE

Although the external ESAI serial clocks can be independent of and asynchronous to the internal ipg_clk_esai ESAI system clock, the external ESAI serial clock frequency cannot exceed 1/4 of the ipg_clk_esai and each external ESAI serial clock phase must exceed the minimum of 2 x 1/igp_clk_esai.

For SCKR pin mode definitions, see Table 25-39.

The table below provides a list of asynchronous-mode receiver clock sources. For more information about EXTAL/ESAI clocking control bits (ERI, ERO), refer to ESAI Control Register (ESAI_ECR).

Table 25-2. Receiver Clock Sources (Asynchronous Mode Only)

RHCKD	RFSD	RCKD	ERI	ERO	Receiver Bit Clock Source	OUTPUTS		
0	0	0	N/A	N/A	SCKR	-	-	-
0	0	1	N/A	N/A	HCKR	-	-	SCKR
0	1	0	N/A	N/A	SCKR	-	FSR	-
0	1	1	N/A	N/A	HCKR	-	FSR	SCKR
1	0	0	0	0	SCKR	HCKR	-	-

Table continues on the next page...

Table 25-2. Receiver Clock Sources (Asynchronous Mode Only) (continued)

RHCKD	RFSD	RCKD	ERI	ERO	Receiver Bit Clock Source	OUTPUTS		
1	0	0	0	1	SCKR	HCKR	-	-
1	0	0	1	0	SCKR	HCKR	-	-
1	0	0	1	1	SCKR	HCKR	-	-
1	0	1	0	0	Fsys[1]	HCKR	-	SCKR
1	0	1	0	1	Fsys	HCKR	-	SCKR
1	0	1	1	0	EXTAL[2]	HCKR	-	SCKR
1	0	1	1	1	EXTAL	HCKR	-	SCKR
1	1	0	0	0	SCKR	HCKR	FSR	-
1	1	0	0	1	SCKR	HCKR	FSR	-
1	1	0	1	0	SCKR	HCKR	FSR	-
1	1	0	1	1	SCKR	HCKR	FSR	-
1	1	1	0	0	Fsys	HCKR	FSR	SCKR
1	1	1	0	1	Fsys	HCKR	FSR	SCKR
1	1	1	1	0	EXTAL	HCKR	FSR	SCKR
1	1	1	1	1	EXTAL	HCKR	FSR	SCKR

Fsys = ipg_clk_esai

EXTAL is the on-chip clock source other than ipg_clk_esai ESAI system clock, and it is from esai_clk_root in CCM.

25.2.8 Transmitter Serial Clock

SCKT is a bidirectional pin providing the transmitters serial bit clock for the ESAI interface.

The direction of this pin is determined by the TCKD bit in the ESAI_TCCR register. The SCKT is a clock input or output used by all the enabled transmitters in the asynchronous mode (SYN = 0) or by all the enabled transmitters and receivers in the synchronous mode (SYN = 1).

The following table provides a list of asynchronous-mode transmitter clock sources.

Table 25-3. Transmitter Clock Sources (Asynchronous Mode Only)

THCKD	TFSD	TCKD	ETI	ETO	Transmitter Bit Clock Source	OUTPUTS		
0	0	0	N/A	N/A	SCKT	-	-	-
0	0	1	N/A	N/A	HCKT	-	-	SCKT

Table continues on the next page...

Table 25-3. Transmitter Clock Sources (Asynchronous Mode Only) (continued)

THCKD	TFSD	TCKD	ETI	ETO	Transmitter Bit Clock Source	OUTPUTS		
0	1	0	N/A	N/A	SCKT	-	FST	-
0	1	1	N/A	N/A	HCKT	-	FST	SCKT
1	0	0	0	0	SCKT	HCKT	-	-
1	0	0	0	1	SCKT	HCKT	-	-
1	0	0	1	0	SCKT	HCKT	-	-
1	0	0	1	1	SCKT	HCKT	-	-
1	0	1	0	0	Fsys[1]	HCKT	-	SCKT
1	0	1	0	1	Fsys	HCKT	-	SCKT
1	0	1	1	0	EXTAL[2]	HCKT	-	SCKT
1	0	1	1	1	EXTAL	HCKT	-	SCKT
1	1	0	0	0	SCKR	HCKT	FST	-
1	1	0	0	1	SCKR	HCKT	FST	-
1	1	0	1	0	SCKR	HCKT	FST	-
1	1	0	1	1	SCKR	HCKT	FST	-
1	1	1	0	0	Fsys	HCKT	FST	SCKT
1	1	1	0	1	Fsys	HCKT	FST	SCKT
1	1	1	1	0	EXTAL	HCKT	FST	SCKT
1	1	1	1	1	EXTAL	HCKT	FST	SCKT

Fsys = ipg_clk_esai

EXTAL is the on-chip clock sources other than ipg_clk_easi ESAI system clock, and it is from esai_clk_root in CCM

SCKT may be programmed as a disconnected pin (PC3) when the ESAI SCKT function is not being used. (See Table 25-48.)

For more information about EXTAL/ESAI clocking control bits (ETI, ETO), see ESAI Control Register (ESAI_ECR).

NOTE

Although the external ESAI serial clocks can be independent of and asynchronous to the internal ipg_clk_esai ESAI system clock, the external ESAI serial clock frequency cannot exceed 1/4 of the ipg_clk_esai and each external ESAI serial clock phase must exceed the minimum of 2 x 1/ipg_clk_esai.

25.2.9 Frame Sync for Receiver

FSR is a bidirectional pin providing the receivers frame sync signal for the ESAI interface. The direction of this pin is determined by the RFSD bit in ESAI_RCR register.

In the asynchronous mode (SYN=0), the FSR pin operates as the frame sync input or output used by all the enabled receivers. In the synchronous mode (SYN=1), it operates as either the serial flag 1 pin (TEBE=0), or as the transmitter external buffer enable control (TEBE=1, RFSD=1). For FSR pin mode definitions, see Table 25-40; for receiver clock signals, see Table 25-2.

When this pin is configured as serial flag pin, its direction is determined by the RFSD bit in the ESAI_RCCR register. When configured as the output flag OF1, this pin reflects the value of the OF1 bit in the ESAI_SAICR register, and the data in the OF1 bit shows up at the pin synchronized to the frame sync being used by the transmitter and receiver sections. When configured as the input flag IF1, the data value at the pin is stored in the IF1 bit in the ESAI_SAISR register, synchronized by the frame sync in normal mode or the slot in network mode.

FSR may be programmed as a disconnected pin (PC1) when the ESAI FSR function is not being used. (See Table 25-48.)

25.2.10 Frame Sync for Transmitter

FST is a bidirectional pin providing the frame sync for both the transmitters and receivers in the synchronous mode (SYN=1) and for the transmitters only in asynchronous mode

(SYN=0) (see Table 25-3). The direction of this pin is determined by the TFSD bit in the ESAI_TCCR register. When configured as an output, this pin is the internally generated frame sync signal. When configured as an input, this pin receives an external frame sync signal for the transmitters (and the receivers in synchronous mode).

FST may be programmed as a disconnected pin (PC4) when the ESAI FST function is not being used. (See Table 25-48.)

25.2.11 High Frequency Clock for Transmitter

HCKT is a bidirectional pin providing the transmitters high frequency clock for the ESAI interface.

The direction of this pin is determined by the THCKD bit in the ESAI_TCCR register. In the asynchronous mode (SYN=0), the HCKT pin operates as the high frequency clock input or output used by all enabled transmitters. In the synchronous mode (SYN=1), it

operates as the high frequency clock input or output used by all enabled transmitters and receivers. When programmed as input this pin is used as an alternative high frequency clock source to the ESAI transmitter rather than the ARM Core main clock. When programmed as output it can serve as a high frequency sample clock (to external DACs for example) or as an additional system clock (see Table 25-3).

HCKT may be programmed as a disconnected pin (PC5) when the ESAI HCKT function is not being used. (See Table 25-48.)

25.2.12 High Frequency Clock for Receiver

HCKR is a bidirectional pin providing the receivers high frequency clock for the ESAI interface.

The direction of this pin is determined by the RHCKD bit in the ESAI_RCCR register. In the asynchronous mode (SYN=0), the HCKR pin operates as the high frequency clock input or output used by all the enabled receivers. In the synchronous mode (SYN=1), it operates as the serial flag 2 pin. For HCKR pin mode definitions, see Table 25-41; for receiver clock signals, see Table 25-2.

When this pin is configured as serial flag pin, its direction is determined by the RHCKD bit in the ESAI_RCCR register. When configured as the output flag OF2, this pin reflects the value of the OF2 bit in the ESAI_SAICR register, and the data in the OF2 bit shows up at the pin synchronized to the frame sync being used by the transmitter and receiver sections. When configured as the input flag IF2, the data value at the pin is stored in the IF2 bit in the ESAI_SAISR register, synchronized by the frame sync in normal mode or the slot in network mode.

HCKR may be programmed as a disconnected pin (PC2) when the ESAI HCKR function is not being used. (See Table 25-48.)

25.2.13 Serial I/O Flags

Three ESAI pins (FSR, SCKR and HCKR) are available as serial I/O flags when the ESAI is operating in the synchronous mode (SYN=1).

Their operation is controlled by RCKD, RFSD, TEBE bits in the ESAI_RCR, ESAI_RCCR and ESAI_SAICR registers.The output data bits (OF2, OF1 and OF0) and the input data bits (IF2, IF1 and IF0) are double buffered to/from the HCKR, FSR and SCKR pins. Double buffering the flags keeps them in sync with the TX and RX data lines.

Each flag can be separately programmed. Flag 0 (SCKR pin) direction is selected by RCKD, RCKD=1 for output and RCKD=0 for input. Flag 1 (FSR pin) is enabled when the pin is not configured as external transmitter buffer enable (TEBE=0) and its direction is selected by RFSD, RFSD=1 for output and RFSD=0 for input. Flag 2 (HCKR pin) direction is selected by RHCKD, RHCKD=1 for output and RHCKD=0 for input.

When programmed as input flags, the SCKR, FSR and HCKR logic values, respectively, are latched at the same time as the first bit of the receive data word is sampled. Because the input was latched, the signal on the input flag pin (SCKR, FSR or HCKR) can change without affecting the input flag until the first bit of the next receive data word. When the received data words are transferred to the receive data registers, the input flag latched values are then transferred to the IF0, IF1 and IF2 bits in the SAISR register, where they may be read by software.

When programmed as output flags, the SCKR, FSR and HCKR logic values are driven by the contents of the OF0, OF1 and OF2 bits in the ESAI_SAICR register respectively, and they are driven when the transmit data registers are transferred to the transmit shift registers. The value on SCKR, FSR and HCKR is stable from the time the first bit of the transmit data word is transmitted until the first bit of the next transmit data word is transmitted. Software may change the OF0-OF2 values thus controlling the SCKR, FSR and HCKR pin values for each transmitted word. The normal sequence for setting output flags when transmitting data is as follows: wait for TDE (transmitter empty) to be set; first write the flags, and then write the transmit data to the transmit registers. OF0, OF1, and OF2 are double buffered so that the flag states appear on the pins when the transmit data is transferred to the transmit shift register, that is, the flags are synchronous with the data.

25.3 Clocks

The table found here describes the clock sources for ESAI.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 25-4. ESAI Clocks

Clock name	Clock Root	Description
extal_clk	esai_clk_root	ESAI system clock
ipg_clk_esai	ahb_clk_root	Bus clock
ipg_clk_s	ipg_clk_root	Peripheral access clock
mem_clk	ahb_clk_root	Mem clock

25.4 Functional Description

This section provides a complete functional description of the block.

25.4.1 ESAI After Reset

Hardware or software reset clears the port control register bits and the port direction control register bits, which configure all ESAI I/O pins as disconnected and both ESAI FIFOs are also in reset state.

The ESAI is in personal reset state while all ESAI pins are programmed as disconnected, and it is active only if at least one of the ESAI I/O pins is programmed as an ESAI pin.

25.4.2 ESAI Interrupt Requests

The ESAI can generate eight different interrupt requests

(ordered from the highest to the lowest priority):

1. **ESAI Receive Data with Exception Status**

   Occurs when the receive exception interrupt is enabled (REIE=1 in the RCR register), at least one of the enabled receive data registers is full (RDF=1) and a receiver overrun error has occurred (ROE=1 in the SAISR register). ROE is cleared by first reading the SAISR and then reading all the enabled receive data registers.

2. **ESAI Receive Even Data**

   Occurs when the receive even slot data interrupt is enabled (REDIE=1), at least one of the enabled receive data registers is full (RDF=1), the data is from an even slot (REDF=1) and no exception has occurred (ROE=0 or REIE=0).

   Reading all enabled receiver data registers clears RDF and REDF.

3. **ESAI Receive Data**

   Occurs when the receive interrupt is enabled (RIE=1), at least one of the enabled receive data registers is full (RDF=1), no exception has occurred (ROE=0 or REIE=0) and no even slot interrupt has occurred (REDF=0 or REDIE=0). Reading all enabled receiver data registers clears RDF.

4. **ESAI Receive Last Slot Interrupt**

Occurs, if enabled (RLIE=1), after the last slot of the frame ended (in network mode only) regardless of the receive mask register setting. The receive last slot interrupt may be used for resetting the receive mask slot register, reconfiguring the DMA channels and reassigning data memory pointers. Using the receive last slot interrupt guarantees that the previous frame was serviced with the previous setting and the new frame is serviced with the new setting without synchronization problems. Note that the maximum receive last slot interrupt service time should not exceed N-1 ESAI bits service time (where N is the number of bits in a slot).

5. **ESAI Transmit Data with Exception Status**

   Occurs when the transmit exception interrupt is enabled (TEIE=1), at least one transmit data register of the enabled transmitters is empty (TDE=1) and a transmitter underrun error has occurred (TUE=1). TUE is cleared by first reading the SAISR and then writing to all the enabled transmit data registers, or to the TSR register.

6. **ESAI Transmit Last Slot Interrupt**

   Occurs, if enabled (TLIE=1), at the start of the last slot of the frame in network mode regardless of the transmit mask register setting. The transmit last slot interrupt may be used for resetting the transmit mask slot register, reconfiguring the DMA channels and reassigning data memory pointers. Using the transmit last slot interrupt guarantees that the previous frame was serviced with the previous setting and the new frame is serviced with the new setting without synchronization problems. Note that the maximum transmit last slot interrupt service time should not exceed N-1 ESAI bits service time (where N is the number of bits in a slot).

7. **ESAI Transmit Even Data**

   Occurs when the transmit even slot data interrupt is enabled (TEDIE=1), at least one of the enabled transmit data registers is empty (TDE=1), the slot is an even slot (TEDE=1) and no exception has occurred (TUE=0 or TEIE=0). Writing to all the TX registers of the enabled transmitters or to TSR clears this interrupt request.

8. **ESAI Transmit Data**

   Occurs when the transmit interrupt is enabled (TIE=1), at least one of the enabled transmit data registers is empty (TDE=1), no exception has occurred (TUE=0 or TEIE=0) and no even slot interrupt has occurred (TEDE=0 or TEDIE=0). Writing to all the TX registers of the enabled transmitters, or to the TSR clears this interrupt request.

25.4.3 ESAI DMA Requests from the FIFOs

The ESAI can generate two different DMA requests:

1. ESAI Transmit FIFO Empty - Asserts when the number of empty slots in the ESAI transmit FIFO exceeds the threshold programmed in the ESAI Transmit FIFO Configuration Register (TFCR). Automatically negates when the number of empty slots is less than the threshold programmed in the ESAI Transmit FIFO Configuration Register.
2. ESAI Receive FIFO Full - Asserts when the number of data words in the ESAI receive FIFO exceeds the threshold programmed in the ESAI Receive FIFO Configuration Register (RFCR). Automatically negates when the number of words is less than the threshold programmed in the ESAI Receive FIFO Configuration Register.

25.4.4 ESAI Transmit and Receive Shift Registers

25.4.4.1 ESAI Transmit Shift Registers

The transmit shift registers contain the data being transmitted

(Figure 25-2 and Figure 25-3).

Data is shifted out to the serial transmit data pins by the selected (internal/external) bit clock when the associated frame sync I/O is asserted.

The number of bits shifted out before the shift registers are considered empty and may be written to again can be 8, 12, 16, 20, 24 or 32 bits (determined by the slot length control bits in the TCR register). Data is shifted out of these registers MSB first if TSHFD=0 and LSB first if TSHFD=1.

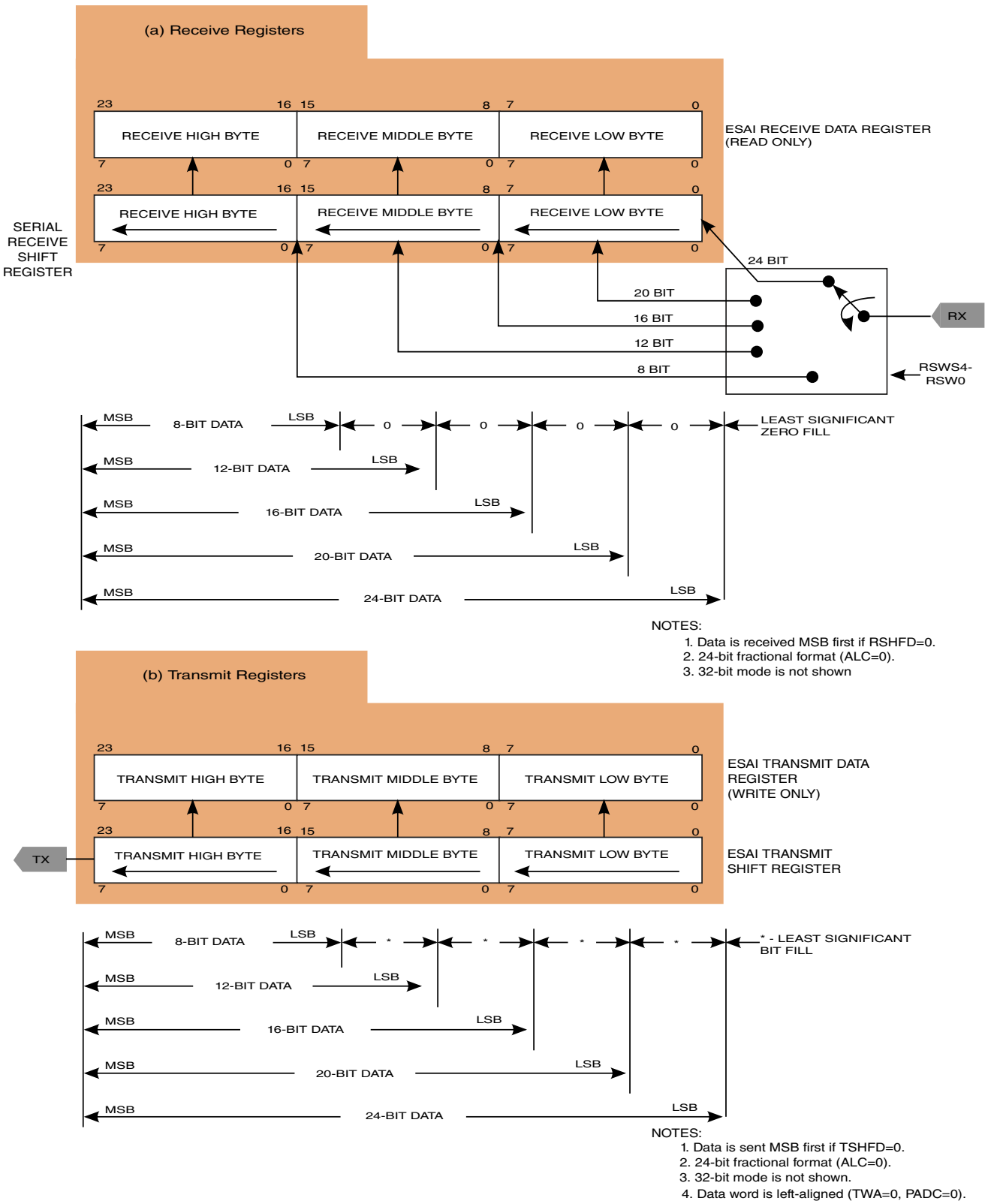


Figure 25-2. ESAI Data Path Programming Model ([R/T]SHFD=0)

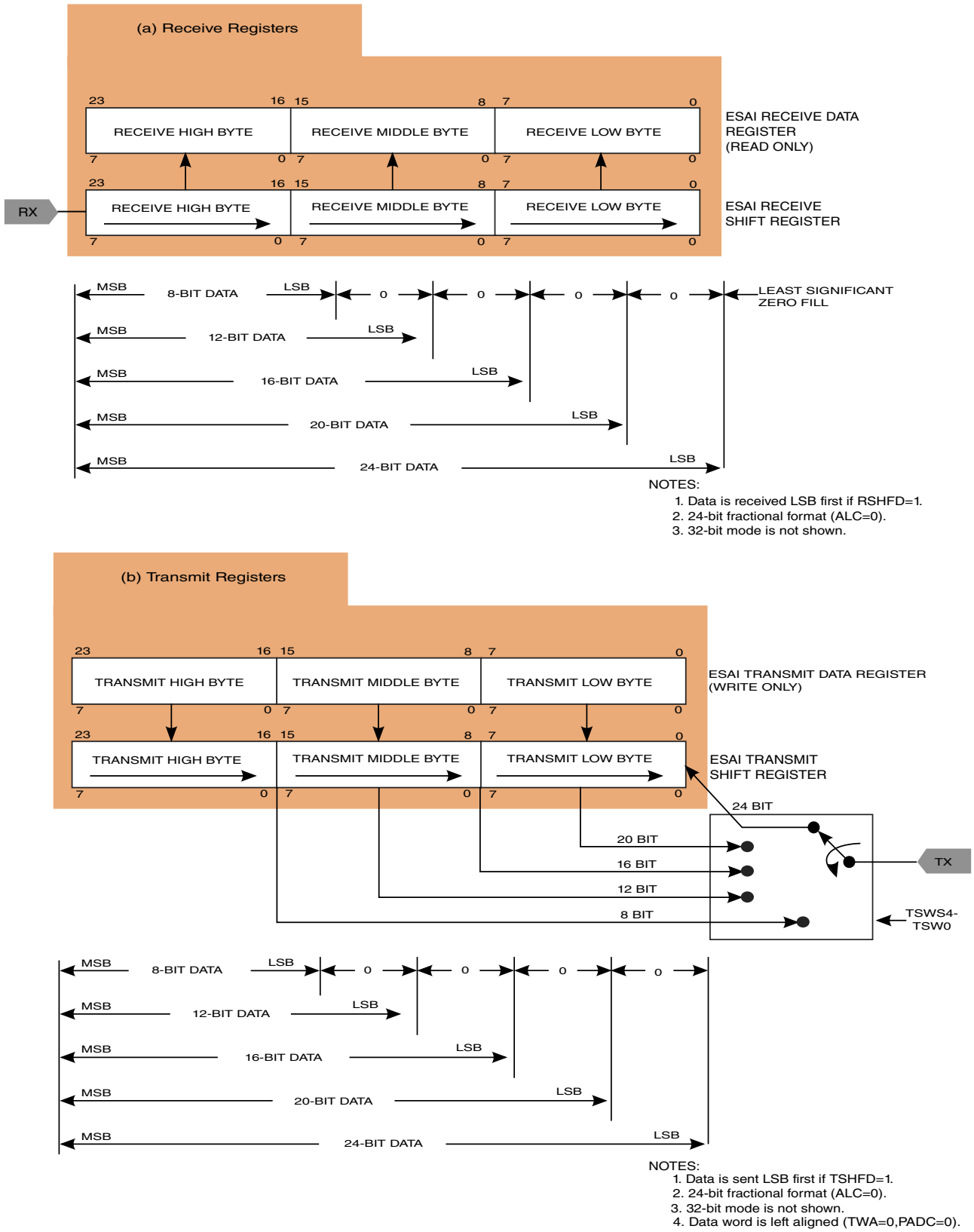


Figure 25-3. ESAI Data Path Programming Model ([R/T]SHFD=1)

25.4.4.2 ESAI Receive Shift Registers

The receive shift registers (Figure 25-2 and Figure 25-3) receive the incoming data from the serial receive data pins. Data is shifted in by the selected (internal/external) bit clock when the associated frame sync I/O is asserted. Data is assumed to be received MSB first if RSHFD=0 and LSB first if RSHFD=1. Data is transferred to the ESAI receive data registers after 8, 12, 16, 20, 24, or 32 serial clock cycles were counted, depending on the slot length control bits in the ESAI_RCR register.

25.5 Initialization Information

25.5.1 ESAI Initialization

The correct way to initialize the ESAI is as follows:

1. Enable the ESAI logic clock by asserting bit 0 of ESAI Control Register (ESAI_ECR[0]).
2. Hardware, software, ESAI individual reset. Note that asserting bit 1 of ESAI Control Register only reset the ESAI core logic, including configuration registers, but not the ESAI FIFOs.
3. Reset ESAI FIFOs by asserting bit 1 of ESAI_TFCR and ESAI_RFCR.
4. Program ESAI control and time slot registers.(The transmit/receive enable bits of TCR/RCR should not be set.)
5. Program ESAI FIFOs via TFCR and RFCR. (Enable Transmit/Receive FIFO, enable transmitters/receivers, transmit initialization and set Transmit FIFO/Receive FIFO watermark.)
6. Write initial words to ESAI Transmit Data Register (ESAI_ETDR), at least one word per enabled transmitter slot but as many as desired. For example 4 channels with 2 slot-per-channel are enabled, then 8 words need to be written into ESAI_ETDR
7. Remove ESAI personal reset by configuring ESAI_PCRC and ESAI_PRRC.
8. Enabled Transmitters/Receivers in ESAI_TCR/ESAI_RCR.

During program execution, all ESAI pins may be defined disconnected, causing the ESAI to stop serial activity and enter the individual reset state.

All status bits of the interface are set to their reset state however, the control bits are not affected. This procedure allows the programmer to reset the ESAI separately from the other internal peripherals. During individual reset, internal DMA accesses to the data registers of the ESAI are not valid and data read is undefined.

The programmer must use an individual ESAI reset when changing the ESAI control registers (except for TEIE, REIE, TLIE, RLIE, TIE, RIE, TE0-TE5, RE0-RE3) to ensure proper operation of the interface.

NOTE

If the ESAI receiver section is already operating with some of the receivers and enabling additional receivers on the fly, that is, without first putting the ESAI receiver in the personal reset state by setting their REx control bits, it will result in erroneous data being received as the first data word for the newly enabled receivers.

25.5.2 ESAI Initialization Examples

25.5.2.1 Initializing the ESAI using Personal Reset

1. Enable the ESAI logic clock by setting bit 0 of ESAI Control Register(ESAI_ECR[0]).
2. The ESAI should be in its personal reset state (ESAI_PCRC = 0x000 and ESAI_PRRC = 0x000). In the personal reset state, both the transmitter and receiver sections of the ESAI are simultaneously reset. The TPR bit in the ESAI_TCR register may be used to reset just the transmitter section. The RPR bit in the ESAI_RCR register may be used to reset just the receiver section.
3. Configure the control registers (ESAI_TCCR, ESAI_TCR, ESAI_RCCR, ESAI_RCR) and ESAI FIFOs configuration Registers (ESAI_TFCR, ESAI_RFCR) according to the operating mode, but do not enable transmitters (TE5-TE0 = 0x0) or receivers (RE3-RE0 = 0x0). It is possible to set the interrupt enable bits which are in use during the operation (no interrupt occurs).
4. Enable the ESAI by setting the ESAI_PCRC and ESAI_PRRC register bits according to pins which are in use during operation.
5. Write initial words to ESAI Transmit Data Register (ESAI_ETDR), at least one word per enabled transmitter slot but as many as desired. For example 4 channels with 2 slot-per-channel are enabled, then 8 words need to be written into ESAI_ETDR. This step is needed even if DMA is used to service the transmitters.
6. Enable the transmitters and receivers.
7. From now on ESAI can be serviced either by polling, interrupts, or DMA.

Operation proceeds as follows:

- For internally generated clock and frame sync, these signals are active immediately after ESAI is enabled (step 4 above).

- Data is received only when one of the receive enable (REx) bits is set and after the occurrence of frame sync signal (either internally or externally generated).
- Data is transmitted only when the transmitter enable (TEx) bit is set and after the occurrence of frame sync signal (either internally or externally generated). The transmitter outputs remain tri-stated after TEx bit is set until the frame sync occurs.

25.5.2.2 Initializing the ESAI Transmitter Section

1. It is assumed that the ESAI is operational; that is, at least one pin is defined as an ESAI pin.
2. Enable the ESAI logic clock by setting bit 0 of ESAI Control Register(ESAI_ECR[0])
3. The transmitter section should be in its individual reset state (TPR = 1) and also reset the ESAI Transmit FIFO (ESAI_TFCR[1] = 1).
4. Configure the control registers ESAI_TCCR and ESAI_TCR according to the operating mode, configure the Transmit FIFO Configuration Register (bring transmit FIFO out of reset, enable Transmit FIFO, enable transmitters, transmit initialization and set watermark). Make sure to clear the transmitter enable bits (TE0-TE5). TPR must remain set.
5. Take the transmitter section out of the individual reset state by clearing TPR.
6. Write initial words to ESAI Transmit Data Register (ESAI_ETDR), at least one word per enabled transmitter slot but as many as desired. For example 4 channels with 2 slot-per-channel are enabled, then 8 words need to be written into ESAI_ETDR
7. Enable the transmitters by setting their TE bits.
8. Data is transmitted only when the transmitter enable (TEx) bit is set and after the occurrence of frame sync signal (either internally or externally generated). The transmitter outputs remain tri-stated after TEx bit is set until the frame sync occurs.
9. From now on the transmitters are operating and can be serviced either by polling, interrupts, or DMA.

25.5.2.3 Initializing the ESAI Receiver Section

1. It is assumed that the ESAI is operational; that is, at least one pin is defined as an ESAI pin.
2. Enable the ESAI logic clock by setting bit 0 of ESAI Control Register (ESAI_ECR[0])
3. The receiver section should be in its individual reset state (RPR = 1) and also reset the ESAI Receive FIFO (ESAI_RFCR[1] = 1).
4. Configure the control registers ESAI_RCCR and ESAI_RCR according to the operating mode, configure the Receive FIFO Configuration Register (bring receive

FIFO out of reset, enable Receive FIFO, receivers, and set watermark). Making sure to clear the receiver enable bits (RE0-RE3). RPR must remain set.
5. Take the receiver section out of the individual reset state by clearing RPR.
6. Enable the receivers by setting their RE bits.
7. From now on the receivers are operating and can be serviced either by polling, interrupts, or DMA.

25.6 ESAI Memory Map/Register Definition

ESAI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
202_4000	ESAI Transmit Data Register (ESAI_ETDR)	32	W (always reads 0)	0000_0000h	25.6.1/1243
202_4004	ESAI Receive Data Register (ESAI_ERDR)	32	R	0000_0000h	25.6.2/1244
202_4008	ESAI Control Register (ESAI_ECR)	32	R/W	0000_0000h	25.6.3/1244
202_400C	ESAI Status Register (ESAI_ESR)	32	R	0000_0000h	25.6.4/1245
202_4010	Transmit FIFO Configuration Register (ESAI_TFCR)	32	R/W	0000_0000h	25.6.5/1247
202_4014	Transmit FIFO Status Register (ESAI_TFSR)	32	R	0000_0000h	25.6.6/1249
202_4018	Receive FIFO Configuration Register (ESAI_RFCR)	32	R/W	0000_0000h	25.6.7/1250
202_401C	Receive FIFO Status Register (ESAI_RFSR)	32	R	0000_0000h	25.6.8/1251
202_4080	Transmit Data Register n (ESAI_TX0)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_4084	Transmit Data Register n (ESAI_TX1)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_4088	Transmit Data Register n (ESAI_TX2)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_408C	Transmit Data Register n (ESAI_TX3)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_4090	Transmit Data Register n (ESAI_TX4)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_4094	Transmit Data Register n (ESAI_TX5)	32	W (always reads 0)	0000_0000h	25.6.9/1252
202_4098	ESAI Transmit Slot Register (ESAI_TSR)	32	W (always reads 0)	0000_0000h	25.6.10/ 1253

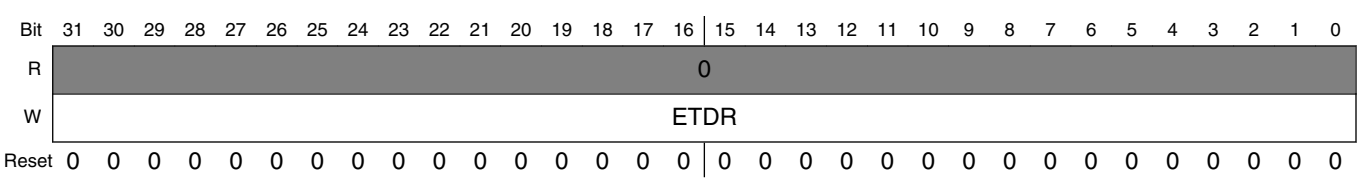
Table continues on the next page...

ESAI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
202_40A0	Receive Data Register n (ESAI_RX0)	32	R	0000_0000h	25.6.11/ 1253
202_40A4	Receive Data Register n (ESAI_RX1)	32	R	0000_0000h	25.6.11/ 1253
202_40A8	Receive Data Register n (ESAI_RX2)	32	R	0000_0000h	25.6.11/ 1253
202_40AC	Receive Data Register n (ESAI_RX3)	32	R	0000_0000h	25.6.11/ 1253
202_40CC	Serial Audio Interface Status Register (ESAI_SAISR)	32	R	0000_0000h	25.6.12/ 1254
202_40D0	Serial Audio Interface Control Register (ESAI_SAICR)	32	R/W	0000_0000h	25.6.13/ 1256
202_40D4	Transmit Control Register (ESAI_TCR)	32	R/W	0000_0000h	25.6.14/ 1259
202_40D8	Transmit Clock Control Register (ESAI_TCCR)	32	R/W	0000_0000h	25.6.15/ 1266
202_40DC	Receive Control Register (ESAI_RCR)	32	R/W	0000_0000h	25.6.16/ 1270
202_40E0	Receive Clock Control Register (ESAI_RCCR)	32	R/W	0000_0000h	25.6.17/ 1274
202_40E4	Transmit Slot Mask Register A (ESAI_TSMA)	32	R/W	0000_FFFFh	25.6.18/ 1277
202_40E8	Transmit Slot Mask Register B (ESAI_TSMB)	32	R/W	0000_FFFFh	25.6.19/ 1278
202_40EC	Receive Slot Mask Register A (ESAI_RSMA)	32	R/W	0000_FFFFh	25.6.20/ 1279
202_40F0	Receive Slot Mask Register B (ESAI_RSMB)	32	R/W	0000_FFFFh	25.6.21/ 1280
202_40F8	Port C Direction Register (ESAI_PRRC)	32	R/W	0000_0000h	25.6.22/ 1281
202_40FC	Port C Control Register (ESAI_PCRC)	32	R/W	0000_0000h	25.6.23/ 1281

25.6.1 ESAI Transmit Data Register (ESAI_ETDR)

Address: 202_4000h base + 0h offset = 202_4000h

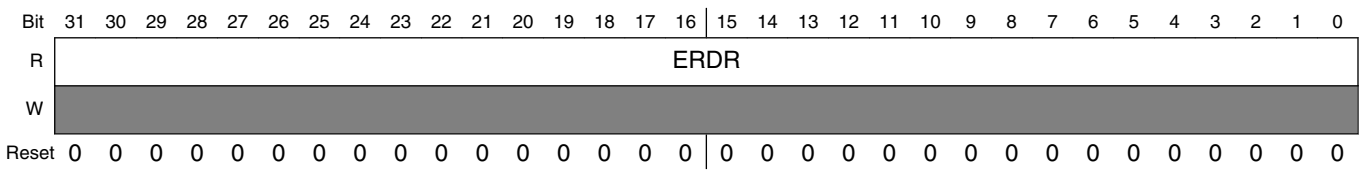


ESAI_ETDR field descriptions

Field	Description
ETDR	ESAI Transmit Data Register. Writing to this register stores the data written into the ESAI Transmit FIFO. Writing to this register when the Transmit FIFO is full causes the data written to be lost (the existing data within the FIFO is not overwritten). When multiple ESAI transmitters are enabled, the data for each transmitter must be interleaved from lowest transmitter to highest transmitter (for example, if transmitters 0, 2 and 3 are enabled then data must be written as follows: transmitter #0, transmitter #2, transmitter #3, transmitter #0, transmitter #2, transmitter #3, transmitter #0, etc). Data within the ESAI Transmit FIFO is passed to the ESAI transmit shifter registers as defined by the Transmit Word Alignment configuration bits.

25.6.2 ESAI Receive Data Register (ESAI_ERDR)

Address: 202_4000h base + 4h offset = 202_4004h

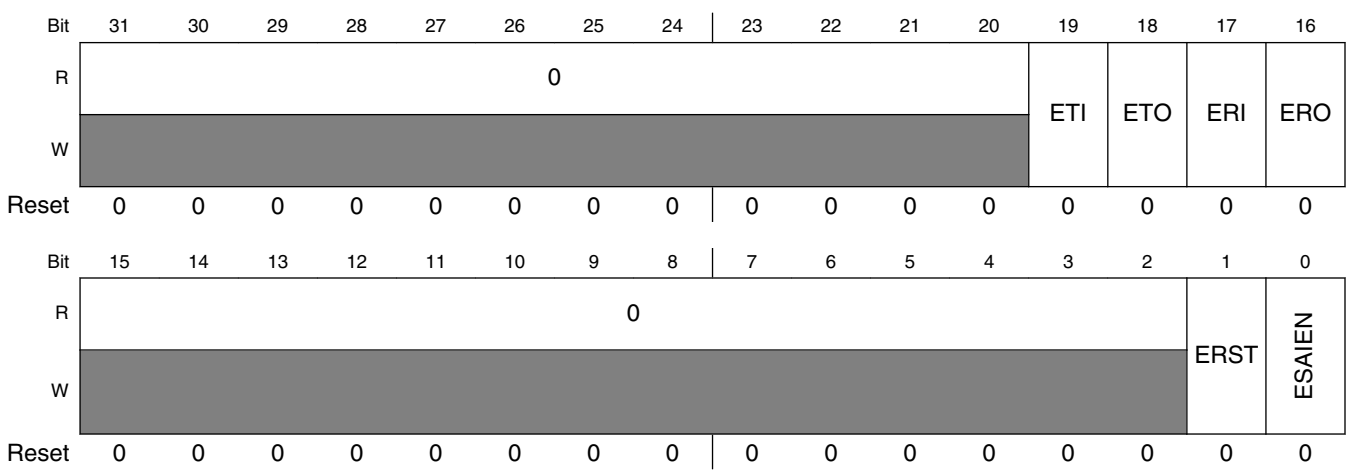


ESAI_ERDR field descriptions

Field	Description
ERDR	ESAI Receive Data Register. Reading this register returns the data within the ESAI Receive FIFO. Reading this register when the Receive FIFO is empty returns the last valid data word. When multiple ESAI receivers are enabled, the data for each receiver is interleaved from lowest receiver to highest receiver (for example, if receivers 0, 2 and 3 are enabled then data is returned as follows: receiver #0, receiver #2, receiver #3, receiver #0, receiver #2, receiver #3, receiver #0, etc). Data is passed from the ESAI receive shift registers to the ESAI Receive FIFO as defined by the Receiver Word Alignment configuration bits either zero or sign-extended based on the Receive Extension control bit.

25.6.3 ESAI Control Register (ESAI_ECR)

Address: 202_4000h base + 8h offset = 202_4008h



ESAI_ECR field descriptions

Field	Description
31–20 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
19 ETI	EXTAL Transmitter In. Mux EXTAL in place of the High Frequency Transmitter Clock input pin. HCKT can still be used to drive a divided down EXTAL or as GPIO. 0 HCKT pin has normal function. 1 EXTAL muxed into HCKT input.
18 ETO	EXTAL Transmitter Out. Drive the EXTAL input on the High Frequency Transmitter Clock pin. 0 HCKT pin has normal function. 1 EXTAL driven onto HCKT pin.
17 ERI	EXTAL Receiver In. Mux EXTAL in place of the High Frequency Receiver Clock input pin. HCKR can still be used to drive a divided down EXTAL or as GPIO. 0 HCKR pin has normal function. 1 EXTAL muxed into HCKR input.
16 ERO	EXTAL Receiver Out. Drive the EXTAL input on the High Frequency Receiver Clock pin. 0 HCKR pin has normal function. 1 EXTAL driven onto HCKR pin.
15–2 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
1 ERST	ESAI Reset. Reset the ESAI core logic (including configuration registers) but not the ESAI FIFOs. 0 ESAI not reset. 1 ESAI reset.
0 ESAIEN	ESAI Enable. Enables/disables the ESAI logic clock. Enable the ESAI before reading or writing other ESAI registers. 0 ESAI disabled. 1 ESAI enabled.

25.6.4 ESAI Status Register (ESAI_ESR)

Address: 202_4000h base + Ch offset = 202_400Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					TINIT	RFF	TFE	TLS	TDE	TED	TD	RLS	RDE	RED	RD
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_ESR field descriptions

Field	Description
31–11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10 TINIT	Transmit Initialization. Indicates that the Transmit FIFO is writing the first word for each enabled transmitter into the Transmit Data Registers. This bit sets when the Transmit FIFO is enabled (provided Transmit Initialization is enabled) and clears after the Transmit Data Registers have been initialized. The Transmit Enable bits in the Transmit Control Register should not be set until this flag has cleared. 0 Transmitter has finished initializing the Transmit Data Registers (or Transmit FIFO is not enabled or Transmit Initialization is not enabled). 1 Transmitter has not finished initializing the Transmit Data Registers.
9 RFF	Receive FIFO Full. Indicates that the number of data words in the Receive FIFO has equaled or exceeded the Receive FIFO Watermark. This flag also drives the ESAI Receiver DMA request line. ESAI FIFO DMA requests see ESAI DMA Requests from the FIFOs. 0 Number of words in Receive FIFO less than Receive FIFO watermark. 1 Number of words in Receive FIFO is equal to or greater than Receive FIFO watermark.
8 TFE	Transmit FIFO Empty. Indicates that the number of empty slots in the Transmit FIFO has met or exceeded the Transmit FIFO Watermark. This flag also drives the ESAI Transmitter DMA request line. ESAI FIFO DMA request see ESAI DMA Requests from the FIFOs. 0 Number of empty slots in Transmit FIFO less than Transmit FIFO watermark. 1 Number of empty slots in Transmit FIFO is equal to or greater than Transmit FIFO watermark.
7 TLS	Transmit Last Slot. Reading this register when TLS is set will negate the Transmit Last Slot interrupt. 0 TLS is not the highest priority active interrupt. 1 TLS is the highest priority active interrupt.
6 TDE	Transmit Data Exception. 0 TDE is not the highest priority active interrupt. 1 TDE is the highest priority active interrupt.
5 TED	Transmit Even Data. 0 TED is not the highest priority active interrupt. 1 TED is the highest priority active interrupt.
4 TD	Transmit Data. 0 TD is not the highest priority active interrupt. 1 TD is the highest priority active interrupt.
3 RLS	Receive Last Slot. Reading this register when RLS is set will negate the Receive Last Slot interrupt. 0 RLS is not the highest priority active interrupt. 1 RLS is the highest priority active interrupt.
2 RDE	Receive Data Exception. 0 RDE is not the highest priority active interrupt. 1 RDE is the highest priority active interrupt.
1 RED	Receive Even Data. 0 RED is not the highest priority active interrupt. 1 RED is the highest priority active interrupt.

Table continues on the next page...

ESAI_ESR field descriptions (continued)

Field	Description
0 RD	Receive Data. 0 RD is not the highest priority active interrupt. 1 RD is the highest priority active interrupt.

25.6.5 Transmit FIFO Configuration Register (ESAI_TFCR)

Address: 202_4000h base + 10h offset = 202_4010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	0												TIEN	TWA[2:0]		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	TFWM[7:0]								TE5	TE4	TE3	TE2	TE1	TE0	TFR	TFE
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_TFCR field descriptions

Field	Description
31–20 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
19 TIEN	Transmitter Initialization Enable. Enables the initialization of the Transmit Data Registers when the Transmitter FIFO is enabled. TIEN=1 is recommended. 0 Transmit Data Registers are not initialized from the FIFO once the Transmit FIFO is enabled. Software must manually initialize the Transmit Data Registers separately. 1 Transmit Data Registers are initialized from the FIFO once the Transmit FIFO is enabled.
18–16 TWA[2:0]	Transmit Word Alignment. Configures the alignment of the data written into the ESAI Transmit Data Register and then passed to the relevant 24 bit Transmit shift register. 000 MSB of data is bit 31. Data bits 7-0 are ignored when passed to transmit shift register. 001 MSB of data is bit 27. Data bits 3-0 are ignored when passed to transmit shift register. 010 MSB of data is bit 23. 011 MSB of data is bit 19. Bottom 4 bits of transmit shift register are zeroed. 100 MSB of data is bit 15. Bottom 8 bits of transmit shift register are zeroed. 101 MSB of data is bit 11. Bottom 12 bits of transmit shift register are zeroed. 110 MSB of data is bit 7. Bottom 16 bits of transmit shift register are zeroed. 111 MSB of data is bit 3. Bottom 20 bits of transmit shift register are zeroed.
15–8 TFWM[7:0]	Transmit FIFO Watermark. These bits configure the threshold at which the Transmit FIFO Empty flag will set. The TFE is set when the number of empty slots in the Transmit FIFO equal or exceed the selected threshold.
7 TE5	Transmitter #5 FIFO Enable. This bit enables transmitter #5 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled.

Table continues on the next page...

ESAI_TFCR field descriptions (continued)

Field	Description
	0 Transmitter #5 is not using the Transmit FIFO. 1 Transmitter #5 is using the Transmit FIFO.
6 TE4	Transmitter #4 FIFO Enable. This bit enables transmitter #4 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled. 0 Transmitter #4 is not using the Transmit FIFO. 1 Transmitter #4 is using the Transmit FIFO.
5 TE3	Transmitter #3 FIFO Enable. This bit enables transmitter #3 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled. 0 Transmitter #3 is not using the Transmit FIFO. 1 Transmitter #3 is using the Transmit FIFO.
4 TE2	Transmitter #2 FIFO Enable. This bit enables transmitter #2 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled. 0 Transmitter #2 is not using the Transmit FIFO. 1 Transmitter #2 is using the Transmit FIFO.
3 TE1	Transmitter #1 FIFO Enable. This bit enables transmitter #1 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled. 0 Transmitter #1 is not using the Transmit FIFO. 1 Transmitter #1 is using the Transmit FIFO.
2 TE0	Transmitter #0 FIFO Enable. This bit enables transmitter #0 to use the Transmit FIFO. Do not change this bit when the Transmitter FIFO is enabled. 0 Transmitter #0 is not using the Transmit FIFO. 1 Transmitter #0 is using the Transmit FIFO.
1 TFR	Transmit FIFO Reset. This bit resets the Transmit FIFO pointers. 0 Transmit FIFO not reset. 1 Transmit FIFO reset.
0 TFE	Transmit FIFO Enable. This bit enables the use of the Transmit FIFO. 0 Transmit FIFO disabled. 1 Transmit FIFO enabled.

25.6.6 Transmit FIFO Status Register (ESAI_TFSR)

Address: 202_4000h base + 14h offset = 202_4014h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	NTFO[2:0]			0	NTFI[2:0]			TFCNT[7:0]							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_TFSR field descriptions

Field	Description
31–15 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
14–12 NTFO[2:0]	Next Transmitter FIFO Out. Indicates which Transmit Data Register receives the top word of the Transmit FIFO. This will usually equal the lowest enabled transmitter, unless the transmit FIFO is empty. 000 Transmitter #0 receives next word from the Transmit FIFO. 001 Transmitter #1 receives next word from the Transmit FIFO. 010 Transmitter #2 receives next word from the Transmit FIFO. 011 Transmitter #3 receives next word from the Transmit FIFO. 100 Transmitter #4 receives next word from the Transmit FIFO. 101 Transmitter #5 receives next word from the Transmit FIFO. 110 Reserved. 111 Reserved.
11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10–8 NTFI[2:0]	Next Transmitter FIFO In. Indicates which transmitter receives the next word written to the FIFO. 000 Transmitter #0 receives next word written to the Transmit FIFO. 001 Transmitter #1 receives next word written to the Transmit FIFO. 010 Transmitter #2 receives next word written to the Transmit FIFO. 011 Transmitter #3 receives next word written to the Transmit FIFO. 100 Transmitter #4 receives next word written to the Transmit FIFO. 101 Transmitter #5 receives next word written to the Transmit FIFO. 110 Reserved. 111 Reserved.
TFCNT[7:0]	Transmit FIFO Counter. These bits indicate the number of data words stored in the Transmit FIFO.

25.6.7 Receive FIFO Configuration Register (ESAI_RFCR)

Address: 202_4000h base + 18h offset = 202_4018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												REXT	RWA[2:0]		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RFWM[7:0]								0		RE3	RE2	RE1	RE0	RFR	RFE
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_RFCR field descriptions

Field	Description
31–20 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
19 REXT	Receive Extension. Enables the receive data to be returned sign extended when the Receive Word Alignment is configured to return data where the MSB is not aligned with bit 31. 0 Receive data is zero extended. 1 Receive data is sign extended.
18–16 RWA[2:0]	Receive Word Alignment. Configures the alignment of the data passed from the relevant 24 bit Receive shift register and read out the ESAI Receive Data Register. 000 MSB of data is at bit 31. Data bits 7-0 are zeroed. 001 MSB of data is at bit 27. Data bits 3-0 are zeroed. 010 MSB of data is at bit 23. 011 MSB of data is at bit 19. Data bits 3-0 from receive shift register are ignored. 100 MSB of data is at bit 15. Data bits 7-0 from receive shift register are ignored. 101 MSB of data is at bit 11. Data bits 11-0 from receive shift register are ignored. 110 MSB of data is at bit 7. Data bits 15-0 from receive shift register are ignored. 111 MSB of data is at bit 3. Data bits 19-0 from receive shift register are ignored.
15–8 RFWM[7:0]	Receive FIFO Watermark. These bits configure the threshold at which the Receive FIFO Full flag will set. The RFF is set when the number of words in the Receive FIFO equal or exceed the selected threshold. It can be set to a non-zero value.
7–6 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
5 RE3	Receiver #3 FIFO Enable. This bit enables receiver #3 to use the Receive FIFO. Do not change this bit when the Receiver FIFO is enabled. 0 Receiver #3 is not using the Receive FIFO. 1 Receiver #3 is using the Receive FIFO.
4 RE2	Receiver #2 FIFO Enable. This bit enables receiver #2 to use the Receive FIFO. Do not change this bit when the Receiver FIFO is enabled. 0 Receiver #2 is not using the Receive FIFO. 1 Receiver #2 is using the Receive FIFO.

Table continues on the next page...

ESAI_RFCR field descriptions (continued)

Field	Description
3 RE1	Receiver #1 FIFO Enable. This bit enables receiver #1 to use the Receive FIFO. Do not change this bit when the Receiver FIFO is enabled. 0 Receiver #1 is not using the Receive FIFO. 1 Receiver #1 is using the Receive FIFO.
2 RE0	Receiver #0 FIFO Enable. This bit enables receiver #0 to use the Receive FIFO. Do not change this bit when the Receiver FIFO is enabled. 0 Receiver #0 is not using the Receive FIFO. 1 Receiver #0 is using the Receive FIFO.
1 RFR	Receive FIFO Reset. This bit resets the Receive FIFO pointers. 0 Receive FIFO not reset. 1 Receive FIFO reset.
0 RFE	Receive FIFO Enable. This bit enables the use of the Receive FIFO. 0 Receive FIFO disabled. 1 Receive FIFO enabled.

25.6.8 Receive FIFO Status Register (ESAI_RFSR)

Address: 202_4000h base + 1Ch offset = 202_401Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0		NRFI[1:0]		0		NRFO[1:0]		RFCNT[7:0]							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_RFSR field descriptions

Field	Description
31–14 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
13–12 NRFI[1:0]	Next Receiver FIFO In. Indicates which Receiver Data Register the Receive FIFO will load next. This will usually equal the lowest enabled receiver, unless the receive FIFO is full. 00 Receiver #0 returns next word to the Receive FIFO. 01 Receiver #1 returns next word to the Receive FIFO. 10 Receiver #2 returns next word to the Receive FIFO. 11 Receiver #3 returns next word to the Receive FIFO.

Table continues on the next page...

ESAI_RFSR field descriptions (continued)

Field	Description
11–10 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
9–8 NRFO[1:0]	Next Receiver FIFO Out. Indicates which receiver returns the top word of the Receive FIFO. 00 Receiver #0 returns next word from the Receive FIFO. 01 Receiver #1 returns next word from the Receive FIFO. 10 Receiver #2 returns next word from the Receive FIFO. 11 Receiver #3 returns next word from the Receive FIFO.
RFCNT[7:0]	Receive FIFO Counter. These bits indicate the number of data words stored in the Receive FIFO.

25.6.9 Transmit Data Register n (ESAI_TX*n*)

ESAI_TX5, ESAI_TX4, ESAI_TX3, ESAI_TX2, ESAI_TX1 and ESAI_TX0 are 32-bit write-only registers. Data to be transmitted is written into these registers and is automatically transferred to the transmit shift registers (Figure 25-2 and Figure 25-3). The data written (8, 12, 16, 20, or 24 bits) should occupy the most significant portion of the TXn according to the ALC control bit setting. The unused bits (least significant portion and the 8 most significant bits when ALC=1) of the TXn are don't care bits. The Core is interrupted whenever the TXn becomes empty if the transmit data register empty interrupt has been enabled.

Address: 202_4000h base + 80h offset + (4d × i), where i=0d to 5d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								0																							
W									TXn[23:0]																							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_TX*n* field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TXn[23:0]	Stores the data to be transmitted and is automatically transferred to the transmit shift registers. See ESAI Transmit Shift Registers.

25.6.10 ESAI Transmit Slot Register (ESAI_TSR)

Address: 202_4000h base + 98h offset = 202_4098h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								0																							
W									TSR[23:0]																							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_TSR field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TSR[23:0]	The write-only Transmit Slot Register (ESAI_TSR) is effectively a null data register that is used when the data is not to be transmitted in the available transmit time slot. The transmit data pins of all the enabled transmitters are in the high-impedance state for the respective time slot where TSR has been written. The Transmitter External Buffer Enable pin (FSR pin when SYN=1, TEBE=1, RFSD=1) disables the external buffers during the slot when the ESAI_TSR register has been written.

25.6.11 Receive Data Register n (ESAI_RX*n*)

ESAI_RX3, ESAI_RX2, ESAI_RX1, and ESAI_RX0 are 32-bit read-only registers that accept data from the receive shift registers when they become full (Figure 25-2 and Figure 25-3). The data occupies the most significant portion of the receive data registers, according to the ALC control bit setting. The unused bits (least significant portion and 8 most significant bits when ALC=1) read as zeros. The Core is interrupted whenever RXn becomes full if the associated interrupt is enabled.

Address: 202_4000h base + A0h offset + (4d × i), where i=0d to 3d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								RXn[23:0]																							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

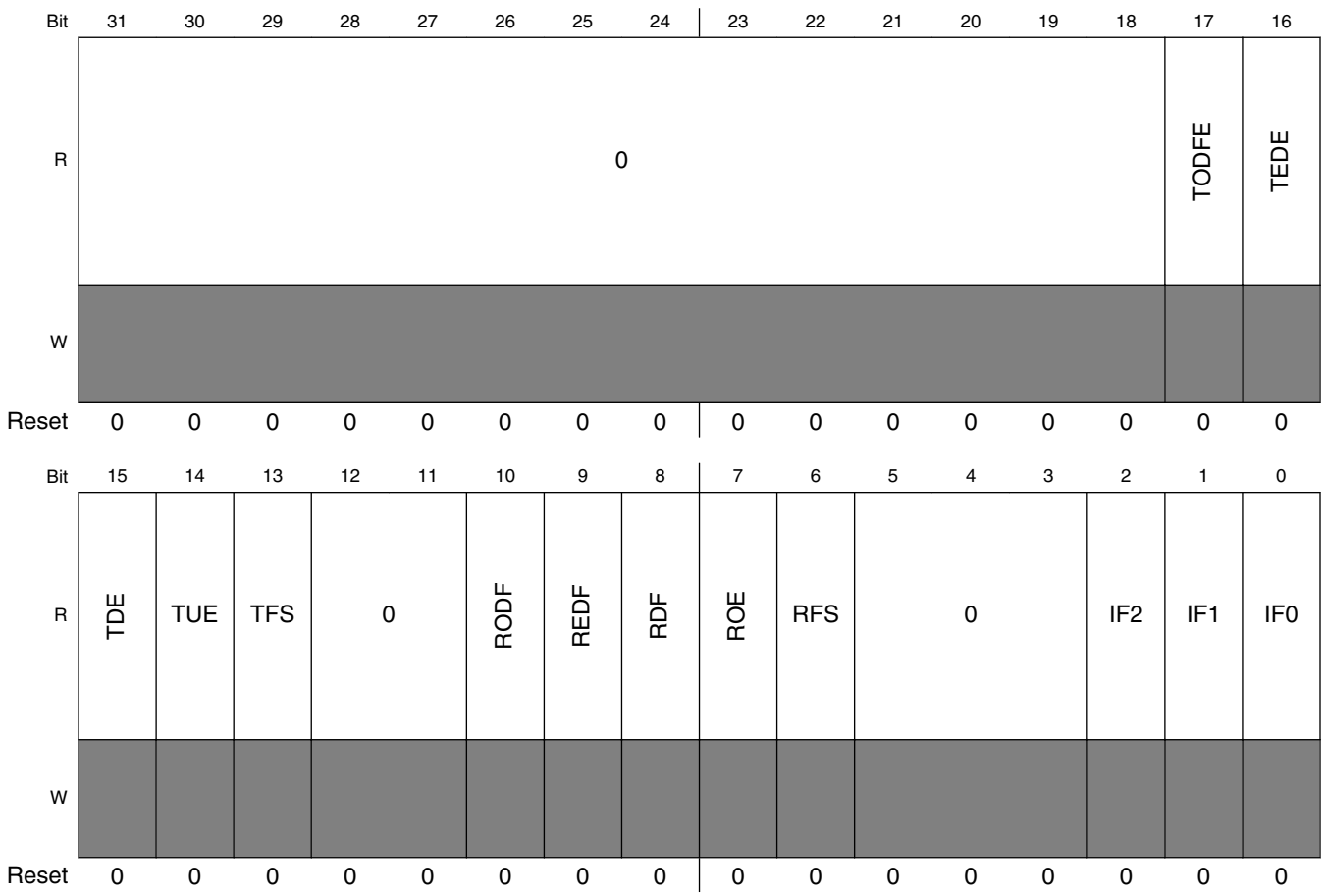
ESAI_RX*n* field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RXn[23:0]	Accept data from the receive shift registers when they become full See ESAI Receive Shift Registers

25.6.12 Serial Audio Interface Status Register (ESAI_SAISR)

The Status Register (ESAI_SAISR) is a read-only status register used by the ARM Core to read the status and serial input flags of the ESAI.

Address: 202_4000h base + CCh offset = 202_40CCh



ESAI_SAISR field descriptions

Field	Description
31–18 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
17 TODFE	ESAI_SAISR Transmit Odd-Data Register Empty. When set, TODFE indicates that the enabled transmitter data registers became empty at the beginning of an odd time slot. Odd time slots are all odd-numbered slots (1, 3, 5, and so on). Time slots are numbered from zero to N-1, where N is the number of time slots in the frame. This flag is set when the contents of the transmit data register of all the enabled transmitters are transferred to the transmit shift registers; it is also set for a TSR disabled time slot period in network mode (as if data were being transmitted after the TSR was written). When set, TODFE indicates that data should be written to all the TX registers of the enabled transmitters or to the transmit slot register (ESAI_TSR). TODE is cleared when the Core writes to all the transmit data registers of the

Table continues on the next page...

ESAI_SAISR field descriptions (continued)

Field	Description
	enabled transmitters, or when the Core writes to the TSR to disable transmission of the next time slot. If TIE is set, an ESAI transmit data interrupt request is issued when TODFE is set. Hardware, software, ESAI individual reset clear TODFE.
16 TEDE	ESAI_SAISR Transmit Even-DataRegister Empty. When set, TEDE indicates that the enabled transmitter data registers became empty at the beginning of an even time slot. Even time slots are all even-numbered slots (0, 2, 4, 6, etc.). Time slots are numbered from zero to N-1, where N is the number of time slots in the frame. The zero time slot is considered even. This flag is set when the contents of the transmit data register of all the enabled transmitters are transferred to the transmit shift registers; it is also set for a TSR disabled time slot period in network mode (as if data were being transmitted after the TSR was written). When set, TEDE indicates that data should be written to all the TX registers of the enabled transmitters or to the transmit slot register (ESAI_TSR). TEDE is cleared when the Core writes to all the transmit data registers of the enabled transmitters, or when the Core writes to the TSR to disable transmission of the next time slot. If TIE is set, an ESAI transmit data interrupt request is issued when TEDE is set. Hardware, software, ESAI individual reset clear TEDE.
15 TDE	ESAI_SAISR Transmit Data Register Empty. TDE is set when the contents of the transmit data register of all the enabled transmitters are transferred to the transmit shift registers; it is also set for a TSR disabled time slot period in network mode (as if data were being transmitted after the TSR was written). When set, TDE indicates that data should be written to all the TX registers of the enabled transmitters or to the transmit slot register (ESAI_TSR). TDE is cleared when the Core writes to all the transmit data registers of the enabled transmitters, or when the Core writes to the TSR to disable transmission of the next time slot. If TIE is set, an ESAI transmit data interrupt request is issued when TDE is set. Hardware, software, ESAI individual reset clear TDE.
14 TUE	ESAI_SAISR Transmit Underrun Error Flag. TUE is set when at least one of the enabled serial transmit shift registers is empty (no new data to be transmitted) and a transmit time slot occurs. When a transmit underrun error occurs, the previous data (which is still present in the TX registers that were not written) is retransmitted. If TEIE is set, an ESAI transmit data with exception (underrun error) interrupt request is issued when TUE is set. Hardware, software, ESAI individual reset clear TUE. TUE is also cleared by reading the ESAI_SAISR with TUE set, followed by writing to all the enabled transmit data registers or to ESAI_TSR.
13 TFS	ESAI_SAISR Transmit Frame Sync Flag. When set, TFS indicates that a transmit frame sync occurred in the current time slot. TFS is set at the start of the first time slot in the frame and cleared during all other time slots. Data written to a transmit data register during the time slot when TFS is set is transmitted (in network mode), if the transmitter is enabled, during the second time slot in the frame. TFS is useful in network mode to identify the start of a frame. TFS is cleared by hardware, software, ESAI individual reset. TFS is valid only if at least one transmitter is enabled, that is, one or more of TE0, TE1, TE2, TE3, TE4 and TE5 are set. (In normal mode, TFS always reads as a one when transmitting data because there is only one time slot per frame - the "frame sync" time slot)
12–11 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
10 RODF	ESAI_SAISR Receive Odd-Data Register Full. When set, RODF indicates that the received data in the receive data registers of the enabled receivers have arrived during an odd time slot when operating in the network mode. Odd time slots are all odd-numbered slots (1, 3, 5, and so on). Time slots are numbered from zero to N-1, where N is the number of time slots in the frame. RODF is set when the contents of the receive shift registers are transferred to the receive data registers. RODF is cleared when the Core reads all the enabled receive data registers or cleared by hardware, software, ESAI individual resets.
9 REDF	ESAI_SAISR Receive Even-Data Register Full. When set, REDF indicates that the received data in the receive data registers of the enabled receivers have arrived during an even time slot when operating in the network mode. Even time slots are all even-numbered slots (0, 2, 4, 6, and so on). Time slots are numbered from zero to N-1, where N is the number of time slots in the frame. The zero time slot is considered even. REDF is set when the contents of the receive shift registers are transferred to the receive data registers. REDF is cleared when the Core reads all the enabled receive data registers or

Table continues on the next page...

ESAI_SAISR field descriptions (continued)

Field	Description
	cleared by hardware, software, ESAI individual resets. If REDIE is set, an ESAI receive even slot data interrupt request is issued when REDF is set.
8 RDF	ESAI_SAISR Receive Data Register Full. RDF is set when the contents of the receive shift register of an enabled receiver is transferred to the respective receive data register. RDF is cleared when the Core reads the receive data register of all enabled receivers or cleared by hardware, software, ESAI individual reset. If RIE is set, an ESAI receive data interrupt request is issued when RDF is set.
7 ROE	ESAI_SAISR Receive Overrun Error Flag. The ROE flag is set when the serial receive shift register of an enabled receiver is full and ready to transfer to its receiver data register (RXn) and the register is already full (RDF=1). If REIE is set, an ESAI receive data with exception (overrun error) interrupt request is issued when ROE is set. Hardware, software, ESAI individual reset clear ROE. ROE is also cleared by reading the SAISR with ROE set, followed by reading all the enabled receive data registers.
6 RFS	ESAI_SAISR Receive Frame Sync Flag. When set, RFS indicates that a receive frame sync occurred during reception of the words in the receiver data registers. This indicates that the data words are from the first slot in the frame. When RFS is clear and a word is received, it indicates (only in the network mode) that the frame sync did not occur during reception of that word. RFS is cleared by hardware, software, ESAI individual reset. RFS is valid only if at least one of the receivers is enabled (REx=1). (In normal mode, RFS always reads as a one when reading data because there is only one time slot per frame - the "frame sync" time slot)
5–3 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
2 IF2	ESAI_SAISR Serial Input Flag 2. The IF2 bit is enabled only when the HCKR pin is defined as ESAI in the Port Control Register, SYN=1 and RHCKD=0, indicating that HCKR is an input flag and the synchronous mode is selected. Data present on the HCKR pin is latched during reception of the first received data bit after frame sync is detected. The IF2 bit is updated with this data when the receive shift registers are transferred into the receiver data registers. IF2 reads as a zero when it is not enabled. Hardware, software, ESAI individual reset clear IF2.
1 IF1	ESAI_SAISR Serial Inout Flag 1. The IF1 bit is enabled only when the FSR pin is defined as ESAI in the Port Control Register, SYN =1, RFSD=0 and TEBE=0, indicating that FSR is an input flag and the synchronous mode is selected. Data present on the FSR pin is latched during reception of the first received data bit after frame sync is detected. The IF1 bit is updated with this data when the receiver shift registers are transferred into the receiver data registers. IF1 reads as a zero when it is not enabled. Hardware, software, ESAI individual reset clear IF1.
0 IF0	ESAI_SAISR Serial Input Flag 0. The IF0 bit is enabled only when the SCKR pin is defined as ESAI in the Port Control Register, SYN=1 and RCKD=0, indicating that SCKR is an input flag and the synchronous mode is selected. Data present on the SCKR pin is latched during reception of the first received data bit after frame sync is detected. The IF0 bit is updated with this data when the receiver shift registers are transferred into the receiver data registers. IF0 reads as a zero when it is not enabled. Hardware, software, ESAI individual reset clear IF0.

25.6.13 Serial Audio Interface Control Register (ESAI_SAICR)

The read/write Common Control Register (ESAI_SAICR) contains control bits for functions that affect both the receive and transmit sections of the ESAI.

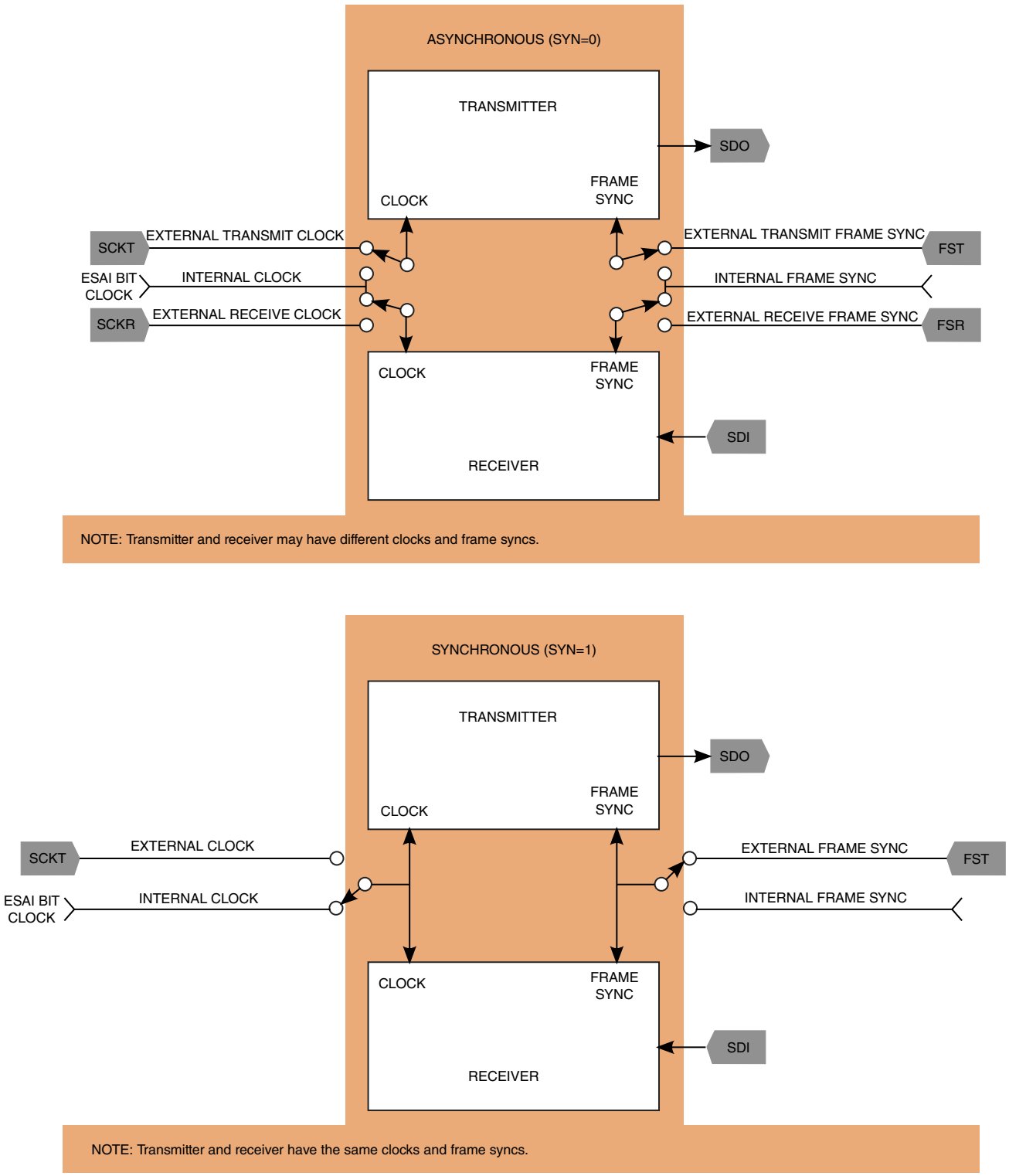


Figure 25-27. SAICR SYN Bit Operation

Address: 202_4000h base + D0h offset = 202_40D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							ALC	TEBE	SYN	0			OF2	OF1	OF0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_SAICR field descriptions

Field	Description
31–9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
8 ALC	ESAI_SAICR Alignment Control. The ESAI is designed for 24-bit fractional data, thus shorter data words are left aligned to the MSB (bit 23). Some applications use 16-bit fractional data. In those cases, shorter data words may be left aligned to bit 15. The Alignment Control (ALC) bit supports these applications. If ALC is set, transmitted and received words are left aligned to bit 15 in the transmit and receive shift registers. If ALC is cleared, transmitted and received word are left aligned to bit 23 in the transmit and receive shift registers. While ALC is set, 20-bit and 24-bit words may not be used, and word length control should specify 8-, 12-, or 16-bit words; otherwise, results are unpredictable.
7 TEBE	ESAI_SAICR Transmit External Buffer Enable. The Transmitter External Buffer Enable (TEBE) bit controls the function of the FSR pin when in the synchronous mode. If the ESAI is configured for operation in the synchronous mode (SYN=1), and TEBE is set while FSR pin is configured as an output (RFSD=1), the FSR pin functions as the transmitter external buffer enable control to enable the use of an external buffers on the transmitter outputs. If TEBE is cleared, the FSR pin functions as the serial I/O flag 1. See Port C Control Register for a summary of the effects of TEBE on the FSR pin.
6 SYN	ESAI_SAICR Synchronous Mode Selection. The Synchronous Mode Selection (SYN) bit controls whether the receiver and transmitter sections of the ESAI operate synchronously or asynchronously with respect to each other (see Port C Control Register). When SYN is cleared, the asynchronous mode is chosen and independent clock and frame sync signals are used for the transmit and receive sections. When SYN is set, the synchronous mode is chosen and the transmit and receive sections use common clock and frame sync signals. When in the synchronous mode (SYN=1), the transmit and receive sections use the transmitter section clock generator as the source of the clock and frame sync for both sections. Also, the receiver clock pins SCKR, FSR and HCKR now operate as I/O flags. Refer to Table 25-39, Table 25-40, and Table 25-41 for the effects of SYN on the receiver clock pins.
5–3 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
2 OF2	ESAI_SAICR Serial Output Flag 2. The Serial Output Flag 2 (OF2) is a data bit used to hold data to be send to the OF2 pin. When the ESAI is in the synchronous clock mode (SYN=1), the HCKR pin is configured as the ESAI flag 2. If the receiver high frequency clock direction bit (RHCKD) is set, the HCKR pin is the output flag OF2, and data present in the OF2 bit is written to the OF2 pin at the beginning of the frame in normal mode or at the beginning of the next time slot in network mode.
1 OF1	ESAI_SAICR Serial Output Flag 1. The Serial Output Flag 1 (OF1) is a data bit used to hold data to be send to the OF1 pin. When the ESAI is in the synchronous clock mode (SYN=1), the FSR pin is configured as the ESAI flag 1. If the receiver frame sync direction bit (RFSD) is set and the TEBE bit is cleared, the FSR pin is the output flag OF1, and data present in the OF1 bit is written to the OF1 pin at the beginning of the frame in normal mode or at the beginning of the next time slot in network mode.

Table continues on the next page...

ESAI_SAICR field descriptions (continued)

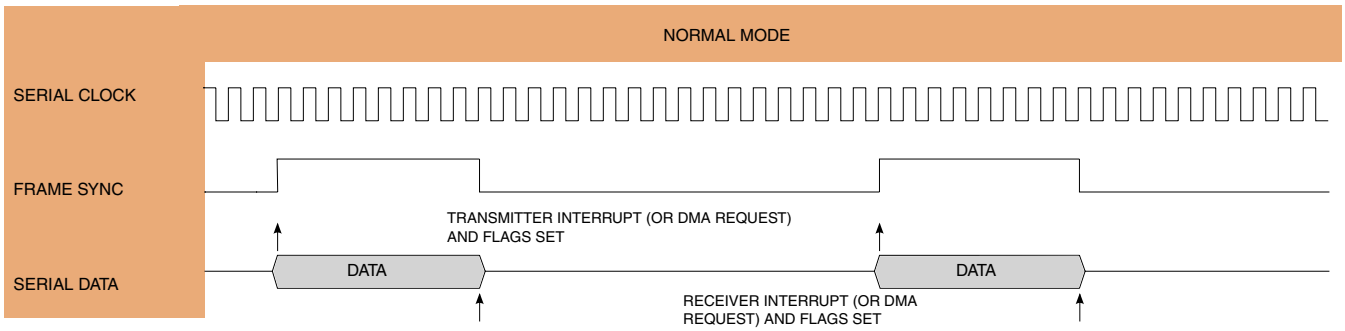
Field	Description
0 OF0	ESAI_SAICR Serial Output Flag 0. The Serial Output Flag 0 (OF0) is a data bit used to hold data to be send to the OF0 pin. When the ESAI is in the synchronous clock mode (SYN=1), the SCKR pin is configured as the ESAI flag 0. If the receiver serial clock direction bit (RCKD) is set, the SCKR pin is the output flag OF0, and data present in the OF0 bit is written to the OF0 pin at the beginning of the frame in normal mode or at the beginning of the next time slot in network mode.

25.6.14 Transmit Control Register (ESAI_TCR)

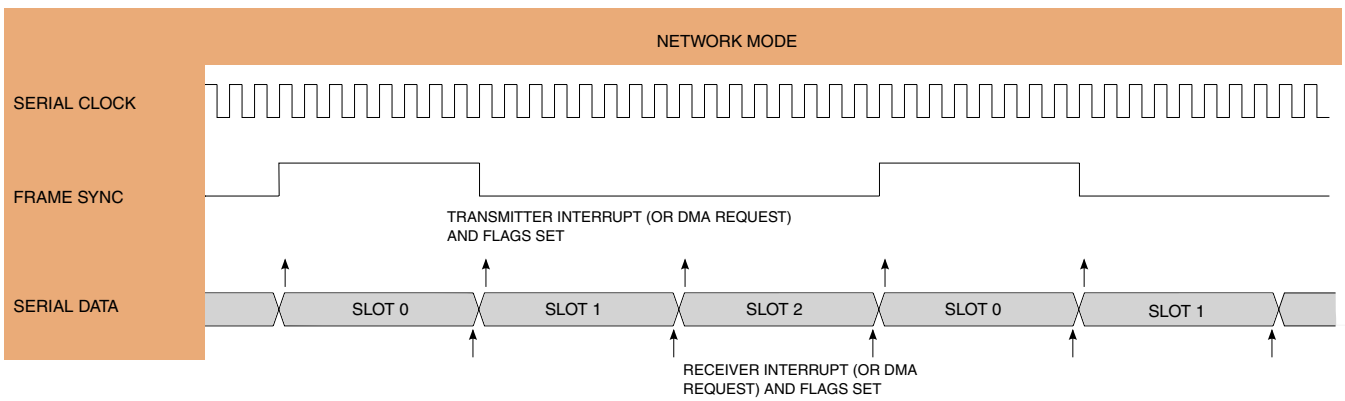
The read/write Transmit Control Register (ESAI_TCR) controls the ESAI transmitter section. Interrupt enable bits for the transmitter section are provided in this control register. Operating modes are also selected in this register.

Table 25-30. Transmit Network Mode Selection

TMOD1	TMOD0	TDC4-TDC0	Transmitter Network Mode
0	0	0x0-0x1F	Normal Mode
0	1	0x0	On-Demand Mode
0	1	0x1-0x1F	Network Mode
1	0	X	Reserved
1	1	0x0C	AC97



NOTE: Interrupts occur and data is transferred once per frame sync.



NOTE: Interrupts occur and a word may be transferred at every time slot.

Figure 25-29. Normal and Network Operation

Table 25-31. ESAI Transmit Slot and Word Length Selection

TSWS4	TSWS3	TSWS2	TSWS1	TSWS0	SLOT LENGTH	WORD LENGTH
0	0	0	0	0	8	8
0	0	1	0	0	12	8
0	0	0	0	1		12
0	1	0	0	0	16	8
0	0	1	0	1		12
0	0	0	1	0		16
0	1	1	0	0	20	8
0	1	0	0	1		12
0	0	1	1	0		16
0	0	0	1	1		20
1	0	0	0	0	24	8
0	1	1	0	1		12
0	1	0	1	0		16
0	0	1	1	1		20

Table continues on the next page...

Table 25-31. ESAI Transmit Slot and Word Length Selection (continued)

TSWS4	TSWS3	TSWS2	TSWS1	TSWS0	SLOT LENGTH	WORD LENGTH
1	1	1	1	0		24
1	1	0	0	0	32	8
1	0	1	0	1		12
1	0	0	1	0		16
0	1	1	1	1		20
1	1	1	1	1		24
0	1	0	1	1	Reserved	
0	1	1	1	0		
1	0	0	0	1		
1	0	0	1	1		
1	0	1	0	0		
1	0	1	1	0		
1	0	1	1	1		
1	1	0	0	1		
1	1	0	1	0		
1	1	0	1	1		
1	1	1	0	0		
1	1	1	0	1		

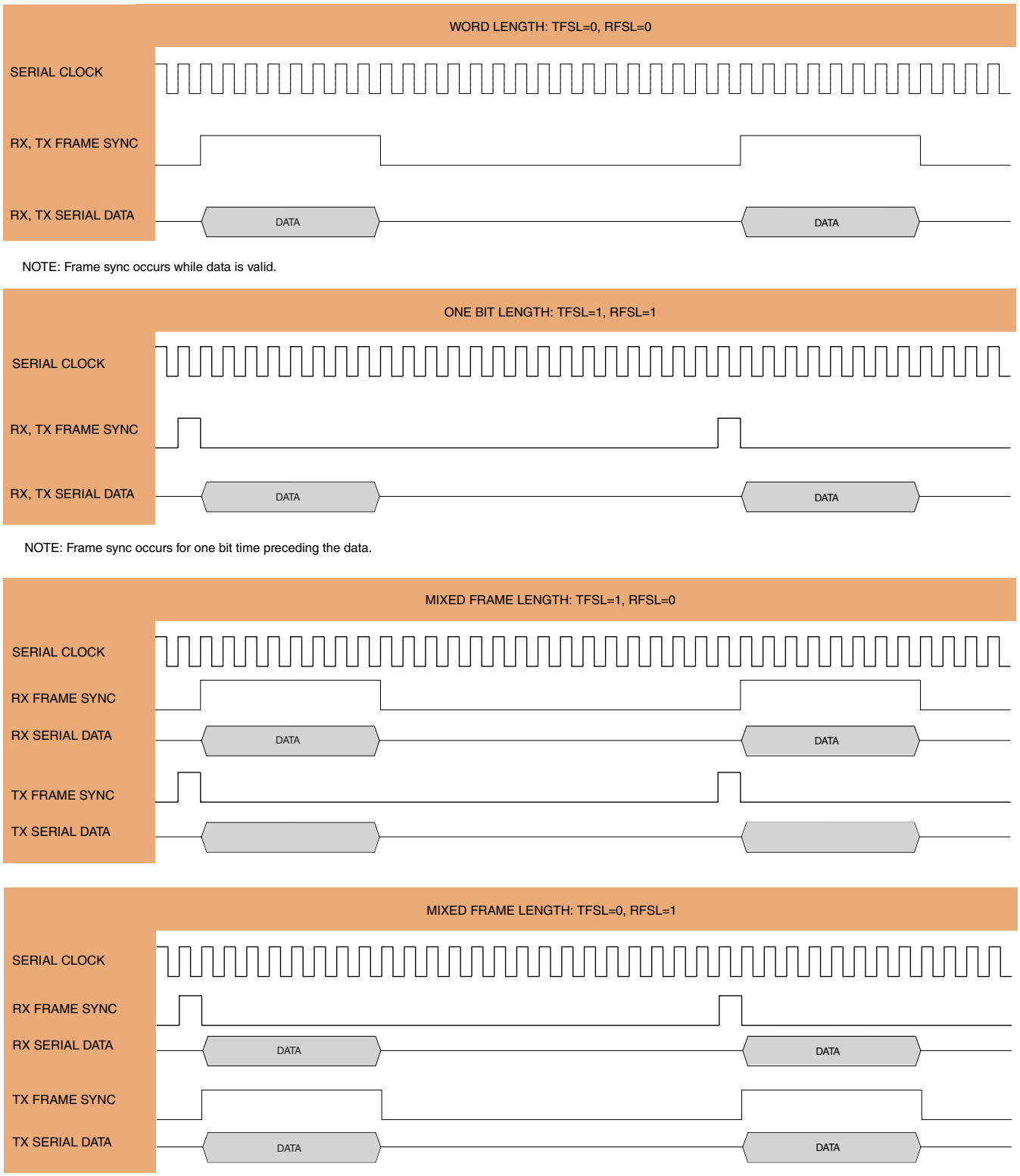
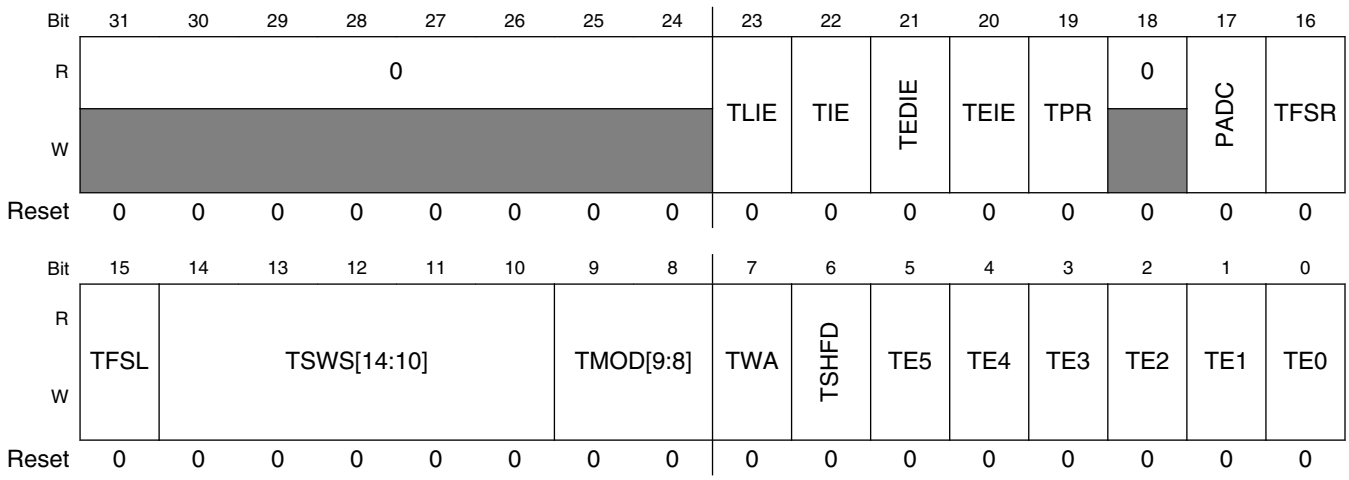


Figure 25-30. Frame Length Selection

Address: 202_4000h base + D4h offset = 202_40D4h



ESAI_TCR field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
23 TLIE	ESAI_TCR Transmit Last Slot Interrupt Enable. TLIE enables an interrupt at the beginning of last slot of a frame in network mode. When TLIE is set the Core is interrupted at the start of the last slot in a frame in network mode regardless of the transmit mask register setting. When TLIE is cleared the transmit last slot interrupt is disabled. TLIE is disabled when TDC[4:0]=0x00000 (on-demand mode). The use of the transmit last slot interrupt is described in ESAI Interrupt Requests.
22 TIE	ESAI_TCR Transmit Interrupt Enable. The Core is interrupted when TIE and the TDE flag in the ESAI_SAISR status register are set. When TIE is cleared, this interrupt is disabled. Writing data to all the data registers of the enabled transmitters or to ESAI_TSR clears TDE, thus clearing the interrupt. Transmit interrupts with exception have higher priority than normal transmit data interrupts, therefore if exception occurs (TUE is set) and TEIE is set, the ESAI requests an ESAI transmit data with exception interrupt from the interrupt controller.
21 TEDIE	ESAI_TCR Transmit Even Slot Data Interrupt Enable. The TEDIE control bit is used to enable the transmit even slot data interrupts. If TEDIE is set, the transmit even slot data interrupts are enabled. If TEDIE is cleared, the transmit even slot data interrupts are disabled. A transmit even slot data interrupt request is generated if TEDIE is set and the TEDE status flag in the ESAI_SAISR status register is set. Even time slots are all even-numbered time slots (0, 2, 4, etc.) when operating in network mode. The zero time slot in the frame is marked by the frame sync signal and is considered to be even. Writing data to all the data registers of the enabled transmitters or to ESAI_TSR clears the TEDE flag, thus servicing the interrupt. Transmit interrupts with exception have higher priority than transmit even slot data interrupts, therefore if exception occurs (TUE is set) and TEIE is set, the ESAI requests an ESAI transmit data with exception interrupt from the interrupt controller.
20 TEIE	ESAI_TCR Transmit Exception Interrupt Enable. When TEIE is set, the Core is interrupted when both TDE and TUE in the ESAI_SAISR status register are set. When TEIE is cleared, this interrupt is disabled. Reading the ESAI_SAISR status register followed by writing to all the data registers of the enabled transmitters clears TUE, thus clearing the pending interrupt.
19 TPR	ESAI_TCR Transmit Section Personal Reset. The TPR control bit is used to put the transmitter section of the ESAI in the personal reset state. The receiver section is not affected. When TPR is cleared, the transmitter section may operate normally. When TPR is set, the transmitter section enters the personal reset state immediately. When in the personal reset state, the status bits are reset to the same state as after hardware reset. The control bits are not affected by the personal reset state. The transmitter data pins are tri-stated while in the personal reset state; if a stable logic level is desired, the transmitter data pins should be defined as GPIO outputs, or external pull-up or pull-down resistors should be used. The

Table continues on the next page...

ESAI_TCR field descriptions (continued)

Field	Description
	transmitter clock outputs drive zeroes while in the personal reset state. Note that to leave the personal reset state by clearing TPR, the procedure described in ESAI Initialization Examples should be followed.
18 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
17 PADC	ESAI_TCR Transmit Zero Padding Control. When PADC is cleared, zero padding is disabled. When PADC is set, zero padding is enabled. PADC, in conjunction with the TWA control bit, determines the way that padding is done for operating modes where the word length is less than the slot length. See the TWA bit description in bit 7 for more details. Because the data word is shorter than the slot length, the data word is extended until achieving the slot length, according to the following rule: If the data word is left-aligned (TWA=0), and zero padding is disabled (PADC=0), the last data bit is repeated after the data word has been transmitted. If zero padding is enabled (PADC=1), zeroes are transmitted after the data word has been transmitted. If the data word is right-aligned (TWA=1), and zero padding is disabled (PADC=0), the first data bit is repeated before the transmission of the data word. If zero padding is enabled (PADC=1), zeroes are transmitted before the transmission of the data word.
16 TFSR	ESAI_TCR Transmit Frame Sync Relative Timing. TFSR determines the relative timing of the transmit frame sync signal as referred to the serial data lines, for a word length frame sync only (TFSL=0). When TFSR is cleared the word length frame sync occurs together with the first bit of the data word of the first slot. When TFSR is set the word length frame sync starts one serial clock cycle earlier, that is, together with the last bit of the previous data word.
15 TFSL	ESAI_TCR Transmit Frame Sync Length. The TFSL bit selects the length of frame sync to be generated or recognized. If TFSL is cleared, a word-length frame sync is selected. If TFSL is set, a 1-bit clock period frame sync is selected. See Figure 1-21 for examples of frame length selection.
14–10 TSWS[14:10]	ESAI_TCR Tx Slot and Word Length Select (TSWS4-TSWS0). The TSWS4-TSWS0 bits are used to select the length of the slot and the length of the data words being transferred through the ESAI. The word length must be equal to or shorter than the slot length. The possible combinations are shown in Table 25-31. See also the ESAI data path programming model in Figure 25-2 and Figure 25-3.
9–8 TMOD[9:8]	ESAI_TCR Transmit Network Mode Control (TMOD1-TMOD0). The TMOD1 and TMOD0 bits are used to define the network mode of ESAI transmitters, as shown in Table 25-31. In the normal mode, the frame rate divider determines the word transfer rate - one word is transferred per frame sync during the frame sync time slot, as shown in Figure 25-29. In network mode, it is possible to transfer a word for every time slot, as shown in Figure 25-29. For further details, refer to Modes of Operation In order to comply with AC-97 specifications, TSWS4-TSWS0 should be set to 00011 (20-bit slot, 20-bit word length), TFSL and TFSR should be cleared, and TDC4-TDC0 should be set to 0x0C (13 words in frame). If TMOD[1:0]=0b11 and the above recommendations are followed, the first slot and word will be 16 bits long, and the next 12 slots and words will be 20 bits long, as required by the AC97 protocol.
7 TWA	ESAI_TCR Transmit Word Alignment Control. The Transmitter Word Alignment Control (TWA) bit defines the alignment of the data word in relation to the slot. This is relevant for the cases where the word length is shorter than the slot length. If TWA is cleared, the data word is left-aligned in the slot frame during transmission. If TWA is set, the data word is right-aligned in the slot frame during transmission. Because the data word is shorter than the slot length, the data word is extended until achieving the slot length, according to the following rule: If the data word is left-aligned (TWA=0), and zero padding is disabled (PADC=0), the last data bit is repeated after the data word has been transmitted. If zero padding is enabled (PADC=1), zeroes are transmitted after the data word has been transmitted. If the data word is right-aligned (TWA=1), and zero padding is disabled (PADC=0), the first data bit is repeated before the transmission of the data word. If zero padding is enabled (PADC=1), zeroes are transmitted before the transmission of the data word.

Table continues on the next page...

ESAI_TCR field descriptions (continued)

Field	Description
6 TSHFD	ESAI_TCR Transmit Shift Direction. The TSHFD bit causes the transmit shift registers to shift data out MSB first when TSHFD equals zero or LSB first when TSHFD equals one (see Figure 25-2 and Figure 25-3 .
5 TE5	ESAI_TCR ESAI Transmit 5 Enable. TE5 enables the transfer of data from ESAI_TX5 to the transmit shift register #5. When TE5 is set and a frame sync is detected, the transmit #5 portion of the ESAI is enabled for that frame. When TE5 is cleared, the transmitter #5 is disabled after completing transmission of data currently in the ESAI transmit shift register. Data can be written to ESAI_TX5 when TE5 is cleared but the data is not transferred to the transmit shift register #5. The SDO5/SDI0 pin is the data input pin for ESAI_RX0 if TE5 is cleared and RE0 in the ESAI_RCR register is set. If both RE0 and TE5 are cleared, the transmitter and receiver are disabled, and the pin is tri-stated. Both RE0 and TE5 should not be set at the same time. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE5 and setting it again disables the transmitter #5 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO5/SDI0 pin remains in the high-impedance state. The on-demand mode transmit enable sequence can be the same as the normal mode, or TE5 can be left enabled.
4 TE4	ESAI_TCR ESAI Transmit 4 Enable. TE4 enables the transfer of data from ESAI_TX4 to the transmit shift register #4. When TE4 is set and a frame sync is detected, the transmit #4 portion of the ESAI is enabled for that frame. When TE4 is cleared, the transmitter #4 is disabled after completing transmission of data currently in the ESAI transmit shift register. Data can be written to ESAI_TX4 when TE4 is cleared but the data is not transferred to the transmit shift register #4. The SDO4/SDI1 pin is the data input pin for ESAI_RX1 if TE4 is cleared and RE1 in the RCR register is set. If both RE1 and TE4 are cleared, the transmitter and receiver are disabled, and the pin is tri-stated. Both RE1 and TE4 should not be set at the same time. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE4 and setting it again disables the transmitter #4 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO4/SDI1 pin remains in the high-impedance state. The on-demand mode transmit enable sequence can be the same as the normal mode, or TE4 can be left enabled.
3 TE3	ESAI_TCR ESAI Transmit 3 Enable. TE3 enables the transfer of data from ESAI_TX3 to the transmit shift register #3. When TE3 is set and a frame sync is detected, the transmit #3 portion of the ESAI is enabled for that frame. When TE3 is cleared, the transmitter #3 is disabled after completing transmission of data currently in the ESAI transmit shift register. Data can be written to ESAI_TX3 when TE3 is cleared but the data is not transferred to the transmit shift register #3. The SDO3/SDI2 pin is the data input pin for ESAI_RX2 if TE3 is cleared and RE2 in the ESAI_RCR register is set. If both RE2 and TE3 are cleared, the transmitter and receiver are disabled, and the pin is tri-stated. Both RE2 and TE3 should not be set at the same time. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE3 and setting it again disables the transmitter #3 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO3/SDI2 pin remains in the high-impedance state. The on-demand mode transmit enable sequence can be the same as the normal mode, or TE3 can be left enabled.
2 TE2	ESAI_TCR ESAI Transmit 2 Enable. TE2 enables the transfer of data from ESAI_TX2 to the transmit shift register #2. When TE2 is set and a frame sync is detected, the transmit #2 portion of the ESAI is enabled for that frame. When TE2 is cleared, the transmitter #2 is disabled after completing transmission of data

Table continues on the next page...

ESAI_TCR field descriptions (continued)

Field	Description
	currently in the ESAI transmit shift register. Data can be written to ESAI_TX2 when TE2 is cleared but the data is not transferred to the transmit shift register #2. The SDO2/SDI3 pin is the data input pin for ESAI_RX3 if TE2 is cleared and RE3 in the ESAI_RCR register is set. If both RE3 and TE2 are cleared, the transmitter and receiver are disabled, and the pin is tri-stated. Both RE3 and TE2 should not be set at the same time. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE2 and setting it again disables the transmitter #2 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO2/SDI3 pin remains in the high-impedance state. The on-demand mode transmit enable sequence can be the same as the normal mode, or TE2 can be left enabled.
1 TE1	ESAI_TCR ESAI Transmit 1 Enable. TE1 enables the transfer of data from TX1 to the transmit shift register #1. When TE1 is set and a frame sync is detected, the transmit #1 portion of the ESAI is enabled for that frame. When TE1 is cleared, the transmitter #1 is disabled after completing transmission of data currently in the ESAI transmit shift register. The SDO1 output is tri-stated, and any data present in TX1 is not transmitted, that is, data can be written to TX1 with TE1 cleared, but data is not transferred to the transmit shift register #1. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE1 and setting it again disables the transmitter #1 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO1 pin remains in the high-impedance state. The on-demand mode transmit enable sequence can be the same as the normal mode, or TE1 can be left enabled.
0 TE0	ESAI_TCR ESAI Transmit 0 Enable. TE0 enables the transfer of data from ESAI_TX0 to the transmit shift register #0. When TE0 is set and a frame sync is detected, the transmit #0 portion of the ESAI is enabled for that frame. When TE0 is cleared, the transmitter #0 is disabled after completing transmission of data currently in the ESAI transmit shift register. The SDO0 output is tri-stated, and any data present in ESAI_TX0 is not transmitted, that is, data can be written to ESAI_TX0 with TE0 cleared, but data is not transferred to the transmit shift register #0. The normal mode transmit enable sequence is to write data to one or more transmit data registers before setting TEx. The normal transmit disable sequence is to clear TEx, TIE and TEIE after TDE equals one. In the network mode, the operation of clearing TE0 and setting it again disables the transmitter #0 after completing transmission of the current data word until the beginning of the next frame. During that time period, the SDO0 pin remains in the high-impedance state.The on-demand mode transmit enable sequence can be the same as the normal mode, or TE0 can be left enabled.

25.6.15 Transmit Clock Control Register (ESAI_TCCR)

The read/write Transmitter Clock Control Register (ESAI_TCCR) controls the ESAI transmitter clock generator bit and frame sync rates, the bit clock and high frequency clock sources and the directions of the HCKT, FST and SCKT signals. In the synchronous mode (SYN=1), the bit clock defined for the transmitter determines the receiver bit clock as well. ESAI_TCCR also controls the number of words per frame for the serial data. Hardware and software reset clear all the bits of the ESAI_TCCR register.

Care should be taken in asynchronous mode whenever the frame sync clock (FSR, FST) is not sourced directly from its associated bit clock (SCKR, SCKT). Proper phase relationships must be maintained between these clocks in order to guarantee proper operation of the ESAI.

NOTE

ARM Core clock is ipg_clk_esai in block ESAI which is from CCM's ahb_clk_root.

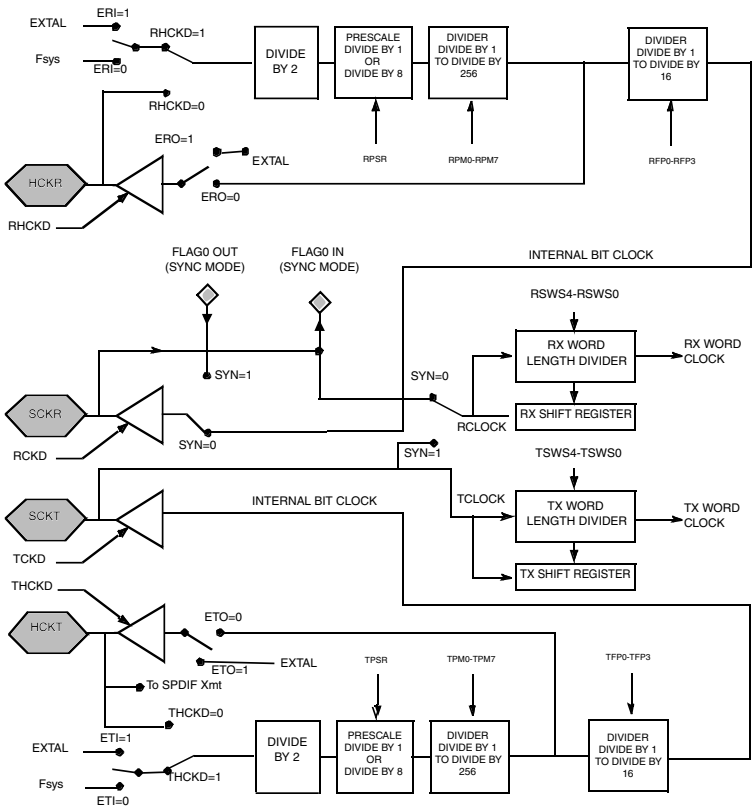


Figure 25-32. ESAI Clock Generator Functional Block Diagram

NOTE

1. ETI, ETO, ERI and ERO bit descriptions are covered in ESAI Control Register (ESAI_ECR). 2. Fsys is the ESAI system 133 MHz clock. 3. EXTAL is the on-chip clock sources other than ESAI system 133MHz clock.

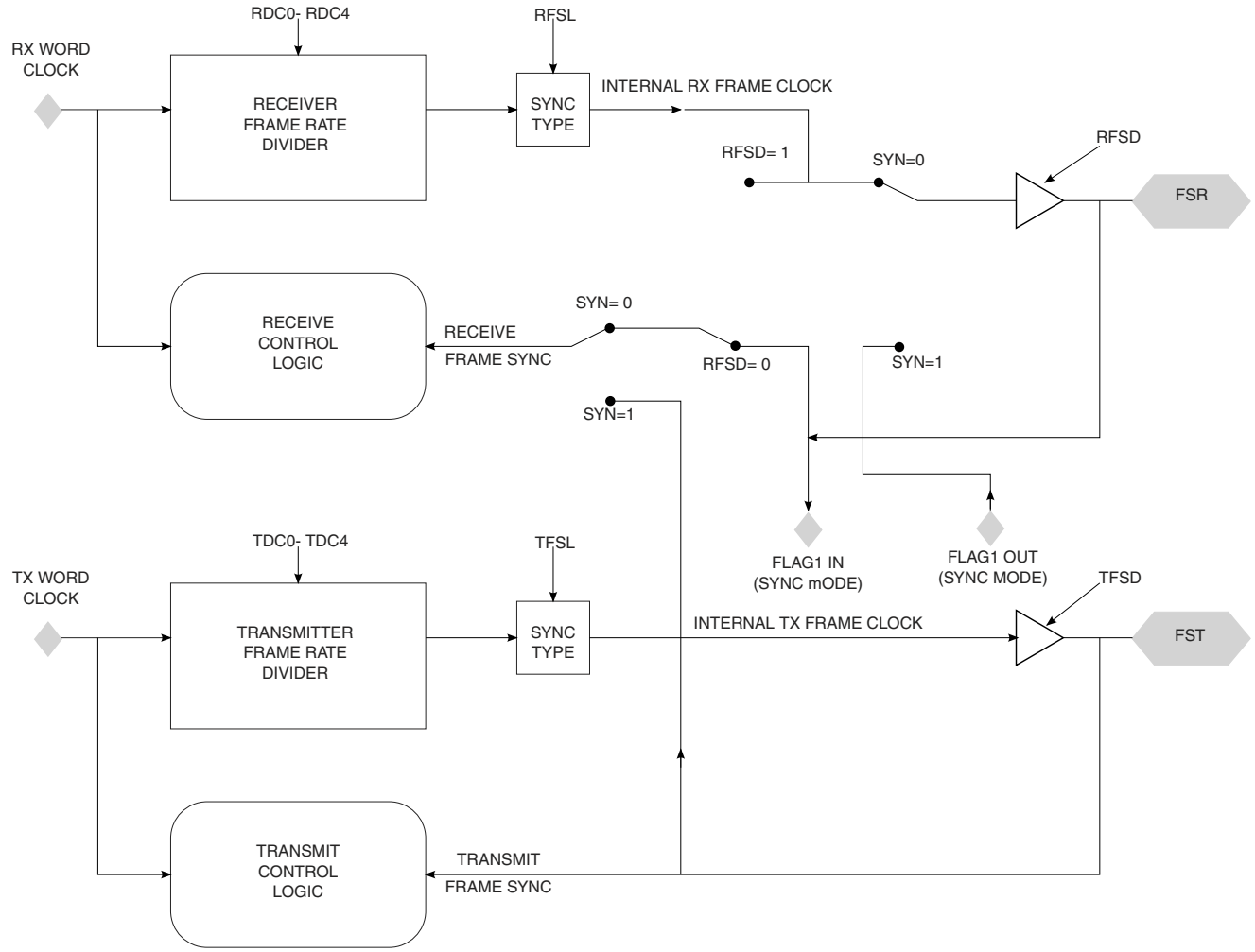
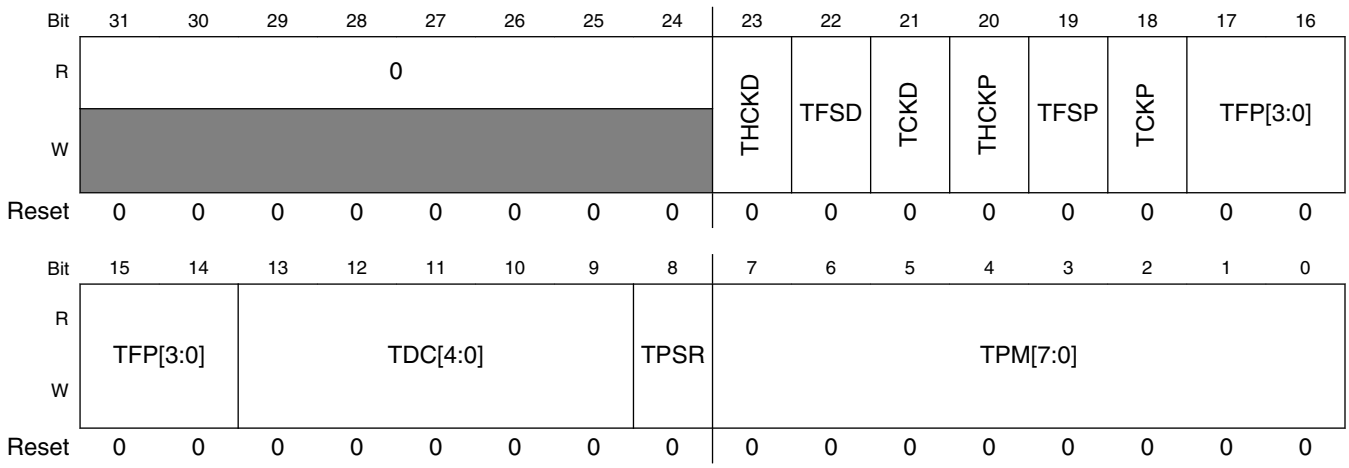


Figure 25-33. ESAI Frame Sync Generator Functional Block Diagram

Table 25-33. Transmitter High Frequency Clock Divider

TFP3-TFP0	Divide Ratio
0x0	1
0x1	2
0x2	3
0x3	4
...	...
0xF	16

Address: 202_4000h base + D8h offset = 202_40D8h



ESAI_TCCR field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
23 THCKD	ESAI_TCCR Transmit High Frequency Clock Direction. THCKD controls the direction of the HCKT pin. When THCKD is cleared, HCKT is an input; when THCKD is set, HCKT is an output (see Table 25-3).
22 TFSD	ESAI_TCCR Transmit Frame Sync Signal Direction. TFSD controls the direction of the FST pin. When TFSD is cleared, FST is an input; when TFSD is set, FST is an output (see Table 25-3).
21 TCKD	ESAI_TCCR Transmit Clock Source Direction. The Transmitter Clock Source Direction (TCKD) bit selects the source of the clock signal used to clock the transmit shift registers in the asynchronous mode (SYN=0) and the transmit shift registers and the receive shift registers in the synchronous mode (SYN=1). When TCKD is set, the internal clock source becomes the bit clock for the transmit shift registers and word length divider and is the output on the SCKT pin. When TCKD is cleared, the clock source is external; the internal clock generator is disconnected from the SCKT pin, and an external clock source may drive this pin (see Table 25-3).
20 THCKP	ESAI_TCCR Transmit High Frequency Clock Polarity The Transmitter High Frequency Clock Polarity (THCKP) bit controls the polarity of the HCKT. 0 - Normal polarity 1 - Inverted polarity
19 TFSP	ESAI_TCCR Transmit Frame Sync Polarity. The Transmitter Frame Sync Polarity (TFSP) bit determines the polarity of the transmit frame sync signal. When TFSP is cleared, the frame sync signal polarity is positive, that is, the frame start is indicated by a high level on the frame sync pin. When TFSP is set, the frame sync signal polarity is negative, that is, the frame start is indicated by a low level on the frame sync pin.
18 TCKP	ESAI_TCCR Transmit Clock Polarity. The Transmitter Clock Polarity (TCKP) bit controls on which bit clock edge data and frame sync are clocked out and latched in. If TCKP is cleared the data and the frame sync are clocked out on the rising edge of the transmit bit clock and latched in on the falling edge of the transmit bit clock. If TCKP is set the falling edge of the transmit clock is used to clock the data out and frame sync and the rising edge of the transmit clock is used to latch the data and frame sync in.
17–14 TFP[3:0]	ESAI_TCCR Tx High Frequency Clock Divider. The TFP3-TFP0 bits control the divide ratio of the transmitter high frequency clock to the transmitter serial bit clock when the source of the high frequency clock and the bit clock is the internal ARM Core clock. When the HCKT input is being driven from an external high frequency clock, the TFP3-TFP0 bits specify an additional division ratio in the clock divider chain. Table 25-33 shows the specification for the divide ratio. Figure 25-32 shows the ESAI high frequency clock generator functional diagram.

Table continues on the next page...

ESAI_TCCR field descriptions (continued)

Field	Description
13–9 TDC[4:0]	ESAI_TCCR Tx Frame Rate Divider Control. The TDC4-TDC0 bits control the divide ratio for the programmable frame rate dividers used to generate the transmitter frame clocks. In network mode, this ratio may be interpreted as the number of words per frame minus one. The divide ratio may range from 2 to 32 (TDC[4:0]=0x00001 to 0x11111) for network mode. A divide ratio of one (TDC[4:0]=0x00000) in network mode is a special case (on-demand mode). In normal mode, this ratio determines the word transfer rate. The divide ratio may range from 1 to 32 (TDC[4:0]=0x00000 to 0x11111) for normal mode. In normal mode, a divide ratio of 1 (TDC[4:0]=0x00000) provides continuous periodic data word transfers. A bit-length frame sync (TFSL=1) must be used in this case. The ESAI frame sync generator functional diagram is shown in Figure 25-33
8 TPSR	ESAI_TCCR Transmit Prescaler Range. The TPSR bit controls a fixed divide-by-eight prescaler in series with the variable prescaler. This bit is used to extend the range of the prescaler for those cases where a slower bit clock is desired. When TPSR is set, the fixed prescaler is bypassed. When TPSR is cleared, the fixed divide-by-eight prescaler is operational (see Figure 25-32). The maximum internally generated bit clock frequency is Fsys/4; the minimum internally generated bit clock frequency is Fsys/(2 x 8 x 256 x 16)=Fsys/65536. (Do not use the combination TPSR=1, TPM7-TPM0=0x00, and TFP3-TFP0=0x0 which causes synchronization problems when using the internal ARM Core clock as source (TCKD=1 or THCKD=1))
TPM[7:0]	ESAI_TCCR Transmit Prescale Modulus Select. The TPM7-TPM0 bits specify the divide ratio of the prescale divider in the ESAI transmitter clock generator. A divide ratio from 1 to 256 (TPM[7:0]=0x00 to 0xFF) may be selected. The bit clock output is available at the transmit serial bit clock (SCKT) pin. The bit clock output is also available internally for use as the bit clock to shift the transmit and receive shift registers. The ESAI transmit clock generator functional diagram is shown in Figure 25-32.

25.6.16 Receive Control Register (ESAI_RCR)

The read/write Receive Control Register (ESAI_RCR) controls the ESAI receiver section. Interrupt enable bits for the receivers are provided in this control register. The receivers are enabled in this register (0,1,2 or 3 receivers can be enabled) if the input data pin is not used by a transmitter. Operating modes are also selected in this register.

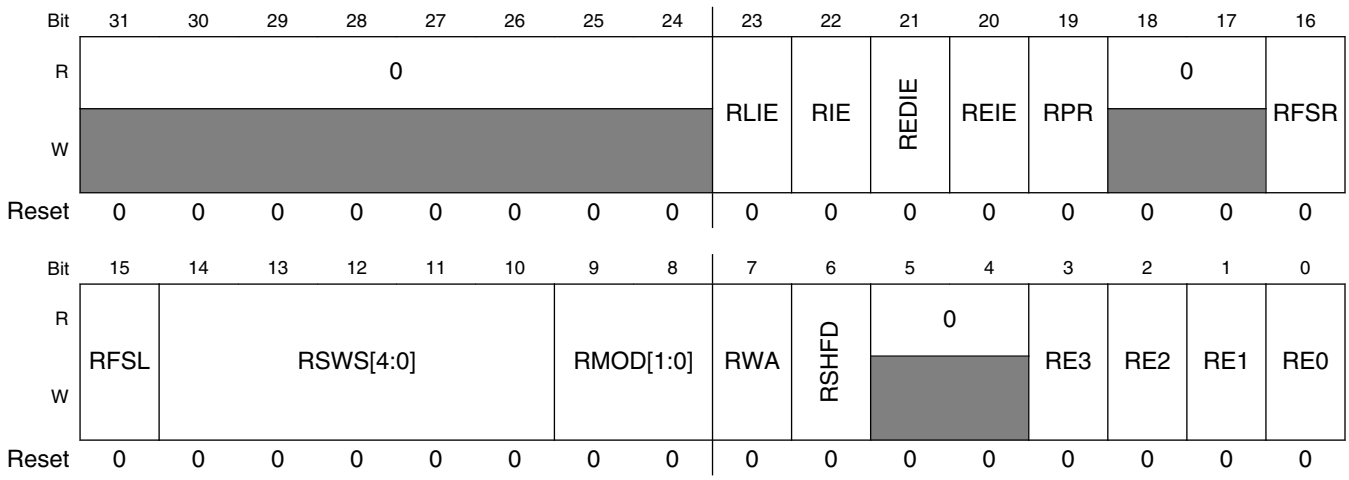
Table 25-35. ESAI Receive Network Mode Selection

RMOD1	RMOD0	RDC4-RDC0	Receiver Network Mode
0	0	0x0-0x1F	Normal Mode
0	1	0x0	On-Demand Mode
0	1	0x1-0x1F	Network Mode
1	0	X	Reserved
1	1	0x0C	AC97

Table 25-36. ESAI Receive Slot and Word Length Selection

RSWS4	RSWS3	RSWS2	RSWS1	RSWS0	SLOT LENGTH	WORD LENGTH
0	0	0	0	0	8	8
0	0	1	0	0	12	8
0	0	0	0	1		12
0	1	0	0	0	16	8
0	0	1	0	1		12
0	0	0	1	0		16
0	1	1	0	0	20	8
0	1	0	0	1		12
0	0	1	1	0		16
0	0	0	1	1		20
1	0	0	0	0	24	8
0	1	1	0	1		12
0	1	0	1	0		16
0	0	1	1	1		20
1	1	1	1	0		24
1	1	0	0	0	32	8
1	0	1	0	1		12
1	0	0	1	0		16
0	1	1	1	1		20
1	1	1	1	1		24
0	1	0	1	1	Reserved	
0	1	1	1	0		
1	0	0	0	1		
1	0	0	1	1		
1	0	1	0	0		
1	0	1	1	0		
1	0	1	1	1		
1	1	0	0	1		
1	1	0	1	0		
1	1	0	1	1		
1	1	1	0	0		
1	1	1	0	1		

Address: 202_4000h base + DCh offset = 202_40DCh



ESAI_RCR field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
23 RLIE	ESAI_RCR Receive Last Slot Interrupt Enable. RLIE enables an interrupt after the last slot of a frame ended in network mode only. When RLIE is set the Core is interrupted after the last slot in a frame ended regardless of the receive mask register setting. When RLIE is cleared the receive last slot interrupt is disabled. Hardware and software reset clear RLIE. RLIE is disabled when RDC[4:0]=00000 (on-demand mode). The use of the receive last slot interrupt is described in ESAI Interrupt Requests.
22 RIE	ESAI_RCR Receive Interrupt Enable. The Core is interrupted when RIE and the RDF flag in the ESAI_SAISR status register are set. When RIE is cleared, this interrupt is disabled. Reading the receive data registers of the enabled receivers clears RDF, thus clearing the interrupt. Receive interrupts with exception have higher priority than normal receive data interrupts, therefore if exception occurs (ROE is set) and REIE is set, the ESAI requests an ESAI receive data with exception interrupt from the interrupt controller.
21 REDIE	ESAI_RCR Receive Even Slot Data Interrupt Enable. The REDIE control bit is used to enable the receive even slot data interrupts. If REDIE is set, the receive even slot data interrupts are enabled. If REDIE is cleared, the receive even slot data interrupts are disabled. A receive even slot data interrupt request is generated if REDIE is set and the REDF status flag in the ESAI_SAISR status register is set. Even time slots are all even-numbered time slots (0, 2, 4, etc.) when operating in network mode. The zero time slot is marked by the frame sync signal and is considered to be even. Reading all the data registers of the enabled receivers clears the REDF flag, thus servicing the interrupt. Receive interrupts with exception have higher priority than receive even slot data interrupts, therefore if exception occurs (ROE is set) and REIE is set, the ESAI requests an ESAI receive data with exception interrupt from the interrupt controller.
20 REIE	ESAI_RCR Receive Exception Interrupt Enable. When REIE is set, the Core is interrupted when both RDF and ROE in the ESAI_SAISR status register are set. When REIE is cleared, this interrupt is disabled. Reading the ESAI_SAISR status register followed by reading the enabled receivers data registers clears ROE, thus clearing the pending interrupt.
19 RPR	ESAI_RCR Receiver Section Personal Reset. The RPR control bit is used to put the receiver section of the ESAI in the personal reset state. The transmitter section is not affected. When RPR is cleared, the receiver section may operate normally. When RPR is set, the receiver section enters the personal reset state immediately. When in the personal reset state, the status bits are reset to the same state as after hardware reset.The control bits are not affected by the personal reset state.The receiver data pins are disconnected while in the personal reset state.

Table continues on the next page...

ESAI_RCR field descriptions (continued)

Field	Description
	NOTE: To leave the personal reset state by clearing RPR, the procedure described in ESAI Initialization Examples should be followed.
18–17 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
16 RFSR	ESAI_RCR Receiver Frame Sync Relative Timing. RFSR determines the relative timing of the receive frame sync signal as referred to the serial data lines, for a word length frame sync only. When RFSR is cleared the word length frame sync occurs together with the first bit of the data word of the first slot. When RFSR is set the word length frame sync starts one serial clock cycle earlier, that is, together with the last bit of the previous data word.
15 RFSL	ESAI_RCR Receiver Frame Sync Length. The RFSL bit selects the length of the receive frame sync to be generated or recognized. If RFSL is cleared, a word-length frame sync is selected. If RFSL is set, a 1-bit clock period frame sync is selected. Refer to Figure 25-30 for examples of frame length selection.
14–10 RSWS[4:0]	ESAI_RCR Receiver Slot and Word Select. The RSWS4-RSWS0 bits are used to select the length of the slot and the length of the data words being received through the ESAI. The word length must be equal to or shorter than the slot length. The possible combinations are shown in Table 25-36. See also the ESAI data path programming model in Figure 25-2 and Figure 25-3.
9–8 RMOD[1:0]	ESAI_RCR Receiver Network Mode Control. The RMOD1 and RMOD0 bits are used to define the network mode of the ESAI receivers, as shown in Table 25-35. In the normal mode, the frame rate divider determines the word transfer rate - one word is transferred per frame sync during the frame sync time slot, as shown in Figure 25-29. In network mode, it is possible to transfer a word for every time slot, as shown in Figure 25-29. For more details, see Modes of Operation. In order to comply with AC-97 specifications, RSWS4-RSWS0 should be set to 0x00011 (20-bit slot, 20-bit word); RFSL and RFSR should be cleared, and RDC4-RDC0 should be set to 0x0C (13 words in frame).
7 RWA	ESAI_RCR Receiver Word Alignment Control. The Receiver Word Alignment Control (RWA) bit defines the alignment of the data word in relation to the slot. This is relevant for the cases where the word length is shorter than the slot length. If RWA is cleared, the data word is assumed to be left-aligned in the slot frame. If RWA is set, the data word is assumed to be right-aligned in the slot frame. If the data word is shorter than the slot length, the data bits which are not in the data word field are ignored. For data word lengths of less than 24 bits, the data word is right-extended with zeroes before being stored in the receive data registers.
6 RSHFD	ESAI_RCR Receiver Shift Direction. The RSHFD bit causes the receiver shift registers to shift data in MSB first when RSHFD is cleared or LSB first when RSHFD is set (see Figure 25-2 and Figure 25-3).
5–4 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
3 RE3	ESAI_RCR ESAI Receiver 3 Enable. When RE3 is set and TE2 is cleared, the ESAI receiver 3 is enabled and samples data at the SDO2/SDI3 pin. ESAI_TX2 and ESAI_RX3 should not be enabled at the same time (RE3=1 and TE2=1). When RE3 is cleared, receiver 3 is disabled by inhibiting data transfer into ESAI_RX3. If this bit is cleared while receiving a data word, the remainder of the word is shifted in and transferred to the ESAI_RX3 data register. If RE3 is set while some of the other receivers are already in operation, the first data word received in ESAI_RX3 will be invalid and must be discarded.
2 RE2	ESAI_RCR ESAI Receiver 2 Enable. When RE2 is set and TE3 is cleared, the ESAI receiver 2 is enabled and samples data at the SDO3/SDI2 pin. ESAI_TX3 and ESAI_RX2 should not be enabled at the same time (RE2=1 and TE3=1). When RE2 is cleared, receiver 2 is disabled by inhibiting data transfer into ESAI_RX2. If this bit is cleared while receiving a data word, the remainder of the word is shifted in and transferred to the ESAI_RX2 data register. If RE2 is set while some of the other receivers are already in operation, the first data word received in ESAI_RX2 will be invalid and must be discarded.

Table continues on the next page...

ESAI_RCR field descriptions (continued)

Field	Description
1 RE1	ESAI_RCR ESAI Receiver 1 Enable. When RE1 is set and TE4 is cleared, the ESAI receiver 1 is enabled and samples data at the SDO4/SDI1 pin. ESAI_TX4 and ESAI_RX1 should not be enabled at the same time (RE1=1 and TE4=1). When RE1 is cleared, receiver 1 is disabled by inhibiting data transfer into ESAI_RX1. If this bit is cleared while receiving a data word, the remainder of the word is shifted in and transferred to the ESAI_RX1 data register. If RE1 is set while some of the other receivers are already in operation, the first data word received in ESAI_RX1 will be invalid and must be discarded.
0 RE0	ESAI_RCR ESAI Receiver 0 Enable. When RE0 is set and TE5 is cleared, the ESAI receiver 0 is enabled and samples data at the SDO5/SDI0 pin. ESAI_TX5 and ESAI_RX0 should not be enabled at the same time (RE0=1 and TE5=1). When RE0 is cleared, receiver 0 is disabled by inhibiting data transfer into ESAI_RX0. If this bit is cleared while receiving a data word, the remainder of the word is shifted in and transferred to the ESAI_RX0 data register. If RE0 is set while some of the other receivers are already in operation, the first data word received in ESAI_RX0 will be invalid and must be discarded.

25.6.17 Receive Clock Control Register (ESAI_RCCR)

The read/write Receiver Clock Control Register (ESAI_RCCR) controls the ESAI receiver clock generator bit and frame sync rates, word length, and number of words per frame for the serial data. The ESAI_RCCR control bits are described in the following paragraphs.

NOTE

ARM Core clock is ipg_clk_esai in block ESAI which is from CCM's ahb_clk_root.

Table 25-38. Receiver High Frequency Clock Divider

RFP3-RFP0	Divide Ratio
0x0	1
0x1	2
0x2	3
0x3	4
...	...
0xF	16

Table 25-39. SCKR Pin Definition Table

Control Bits		SCKR PIN
SYN	RCKD	
0	0	SCKR input

Table continues on the next page...

Table 25-39. SCKR Pin Definition Table (continued)

Control Bits		SCKR PIN
SYN	RCKD	
0	1	SCKR output
1	0	IF0
1	1	OF0

Table 25-40. FSR Pin Definition Table

Control Bits			FSR Pin
SYN	TEBE	RFSD	
0	X	0	FSR input
0	X	1	FSR output
1	0	0	IF1
1	0	1	OF1
1	1	0	reserved
1	1	1	Transmitter Buffer Enable

Table 25-41. HCKR Pin Definition Table

Control Bits		HCKR PIN
SYN	RHCKD	
0	0	HCKR input
0	1	HCKR output
1	0	IF2
1	1	OF2

Address: 202_4000h base + E0h offset = 202_40E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0								RHCKD	RFSD	RCKD	RHCKP	RFSP	RCKP	RFP[3:0]	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RFP[3:0]			RDC[4:0]				RPSR	RPM[7:0]							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_RCCR field descriptions

Field	Description
31–24 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
23 RHCKD	ESAI_RCCR Receiver High Frequency Clock Direction. The Receiver High Frequency Clock Direction (RHCKD) bit selects the source of the receiver high frequency clock when in the asynchronous mode (SYN=0) and the IF2/OF2 flag direction in the synchronous mode (SYN=1). In the asynchronous mode, when RHCKD is set, the internal clock generator becomes the source of the receiver high frequency clock and is the output on the HCKR pin. In the asynchronous mode, when RHCKD is cleared, the receiver high frequency clock source is external; the internal clock generator is disconnected from the HCKR pin, and an external clock source may drive this pin. When RHCKD is cleared, HCKR is an input; when RHCKD is set, HCKR is an output. In the synchronous mode when RHCKD is set, the HCKR pin becomes the OF2 output flag. If RHCKD is cleared, the HCKR pin becomes the IF2 input flag. Refer to Table 25-2 and Table 25-41.
22 RFSD	ESAI_RCCR Receiver Frame Sync Signal Direction. The Receiver Frame Sync Signal Direction (RFSD) bit selects the source of the receiver frame sync signal when in the asynchronous mode (SYN=0) and the IF1/OF1/Transmitter Buffer Enable flag direction in the synchronous mode (SYN=1). In the asynchronous mode, when RFSD is set, the internal clock generator becomes the source of the receiver frame sync and is the output on the FSR pin. In the asynchronous mode, when RFSD is cleared, the receiver frame sync source is external; the internal clock generator is disconnected from the FSR pin, and an external clock source may drive this pin. In the synchronous mode when RFSD is set, the FSR pin becomes the OF1 output flag or the Transmitter Buffer Enable, according to the TEBE control bit. If RFSD is cleared, the FSR pin becomes the IF1 input flag. Refer to Table 25-2 and Table 25-40 .
21 RCKD	ESAI_RCCR Receiver Clock Source Direction. The Receiver Clock Source Direction (RCKD) bit selects the source of the clock signal used to clock the receive shift register in the asynchronous mode (SYN=0) and the IF0/OF0 flag direction in the synchronous mode (SYN=1). In the asynchronous mode, when RCKD is set, the internal clock source becomes the bit clock for the receive shift registers and word length divider and is the output on the SCKR pin. In the asynchronous mode when RCKD is cleared, the clock source is external; the internal clock generator is disconnected from the SCKR pin, and an external clock source may drive this pin. In the synchronous mode when RCKD is set, the SCKR pin becomes the OF0 output flag. If RCKD is cleared, the SCKR pin becomes the IF0 input flag. Refer to Table 25-2 and Table 25-39.
20 RHCKP	ESAI_RCCR Receiver High Frequency Clock Polarity. The Receiver High Frequency Clock Polarity (RHCKP) bit controls on which bit clock edge data and frame sync are clocked out and latched in. If RHCKP is cleared the data and the frame sync are clocked out on the rising edge of the receive high frequency bit clock and the frame sync is latched in on the falling edge of the receive bit clock. If RHCKP is set the falling edge of the receive clock is used to clock the data and frame sync out and the rising edge of the receive clock is used to latch the frame sync in.
19 RFSP	ESAI_RCCR Receiver Frame Sync Polarity. The Receiver Frame Sync Polarity (RFSP) determines the polarity of the receive frame sync signal. When RFSP is cleared the frame sync signal polarity is positive, that is, the frame start is indicated by a high level on the frame sync pin. When RFSP is set the frame sync signal polarity is negative, that is, the frame start is indicated by a low level on the frame sync pin.
18 RCKP	The Receiver Clock Polarity (RCKP) bit controls on which bit clock edge data and frame sync are clocked out and latched in. If RCKP is cleared the data and the frame sync are clocked out on the rising edge of the receive bit clock and the frame sync is latched in on the falling edge of the receive bit clock. If RCKP is set the falling edge of the receive clock is used to clock the data and frame sync out and the rising edge of the receive clock is used to latch the frame sync in.
17–14 RFP[3:0]	ESAI_RCCR Rx High Frequency Clock Divider. The RFP3-RFP0 bits control the divide ratio of the receiver high frequency clock to the receiver serial bit clock when the source of the receiver high frequency clock and the bit clock is the internal Arm Core clock. When the HCKR input is being driven

Table continues on the next page...

ESAI_RCCR field descriptions (continued)

Field	Description
	from an external high frequency clock, the RFP3-RFP0 bits specify an additional division ration in the clock divider chain. Table 25-38 provides the specification of the divide ratio. Figure 25-32 shows the ESAI high frequency generator functional diagram.
13–9 RDC[4:0]	ESAI_RCCR Rx Frame Rate Divider Control. The RDC4-RDC0 bits control the divide ratio for the programmable frame rate dividers used to generate the receiver frame clocks. In network mode, this ratio may be interpreted as the number of words per frame minus one. The divide ratio may range from 2 to 32 (RDC[4:0]=0x00001 to 0x11111) for network mode. A divide ratio of one (RDC[4:0]=0x00000) in network mode is a special case (on-demand mode). In normal mode, this ratio determines the word transfer rate. The divide ratio may range from 1 to 32 (RDC[4:0]=0x00000 to 0x11111) for normal mode. In normal mode, a divide ratio of one (RDC[4:0]=0x00000) provides continuous periodic data word transfers. A bit-length frame sync (RFSL=1) must be used in this case. The ESAI frame sync generator functional diagram is shown in Figure 25-33 .
8 RPSR	ESAI_RCCR Receiver Prescaler Range. The RPSR controls a fixed divide-by-eight prescaler in series with the variable prescaler. This bit is used to extend the range of the prescaler for those cases where a slower bit clock is desired. When RPSR is set, the fixed prescaler is bypassed. When RPSR is cleared, the fixed divide-by-eight prescaler is operational (see Figure 25-32). The maximum internally generated bit clock frequency is Fsys/4, the minimum internally generated bit clock frequency is Fsys/(2 x 8 x 256 x 16)=Fsys/65536. (Do not use the combination RPSR=1 and RPM7-RPM0 =0x00, which causes synchronization problems when using the internal Core clock as source (RHCKD=1 or RCKD=1))
RPM[7:0]	ESAI_RCCR Receiver Prescale Modulus Select. The RPM7-RPM0 bits specify the divide ratio of the prescale divider in the ESAI receiver clock generator. A divide ratio from 1 to 256 (RPM[7:0]=0x00 to 0xFF) may be selected. The bit clock output is available at the receiver serial bit clock (SCKR) pin. The bit clock output is also available internally for use as the bit clock to shift the receive shift registers. The ESAI receive clock generator functional diagram is shown in Figure 25-32.

25.6.18 Transmit Slot Mask Register A (ESAI_TSMA)

The Transmit Slot Mask Register A together with Transmit Slot Mask Register B (ESAI_TSMA and ESAI_TSMB) are two read/write registers used by the transmitters in network mode to determine for each slot whether to transmit a data word and generate a transmitter empty condition (TDE=1), or to tri-state the transmitter data pins. Fields ESAI_TSMA [TS[15:0]] and ESAI_TSMB [TS[31:16]] are concatenated to form the 32-bit field TS[31:0]. Bit number n in TS[31:0] is the enable/disable control bit for transmission in slot number n.

Address: 202_4000h base + E4h offset = 202_40E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																TS[15:0]															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

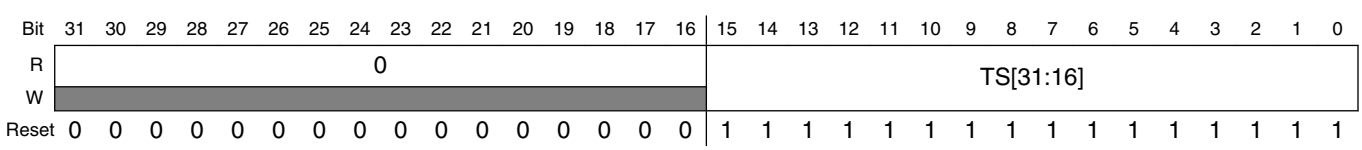
ESAI_TSMA field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TS[15:0]	When bit number N in ESAI_TSMA is cleared, all the transmit data pins of the enabled transmitters are tri-stated during transmit time slot number N. The data is still transferred from the transmit data registers to the transmit shift registers but neither the TDE nor the TUE flags are set. This means that during a disabled slot, no transmitter empty interrupt is generated. The Core is interrupted only for enabled slots. Data that is written to the transmit data registers when servicing this request is transmitted in the next enabled transmit time slot. When bit number N in ESAI_TSMA register is set, the transmit sequence is as usual: data is transferred from the TX registers to the shift registers and transmitted during slot number N, and the TDE flag is set. Using the slot mask in ESAI_TSMA does not conflict with using TSR. Even if a slot is enabled in ESAI_TSMA, the user may choose to write to TSR instead of writing to the transmit data registers TXn. This causes all the transmit data pins of the enabled transmitters to be tri-stated during the next slot. Data written to the ESAI_TSMA affects the next frame transmission. The frame being transmitted is not affected by this data and would comply to the last ESAI_TSMA setting. Data read from ESAI_TSMA returns the last written data. After hardware or software reset, the ESAI_TSMA register is preset to 0x0000FFFF, which means that all 16 possible slots are enabled for data transmission. When operating in normal mode, bit 0 of the ESAI_TSMA register must be set, otherwise no output is generated.

25.6.19 Transmit Slot Mask Register B (ESAI_TSMB)

The Transmit Slot Mask Register B together with Transmit Slot Mask Register A (ESAI_TSMA and ESAI_TSMB) are two read/write registers used by the transmitters in network mode to determine for each slot whether to transmit a data word and generate a transmitter empty condition (TDE=1), or to tri-state the transmitter data pins. Fields ESAI_TSMA [TS[15:0]] and ESAI_TSMB [TS[31:16]] are concatenated to form the 32-bit field TS[31:0]. Bit number n in TS[31:0] is the enable/disable control bit for transmission in slot number n.

Address: 202_4000h base + E8h offset = 202_40E8h



ESAI_TSMB field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
TS[31:16]	When bit number N in ESAI_TSMB is cleared, all the transmit data pins of the enabled transmitters are tri-stated during transmit time slot number N. The data is still transferred from the transmit data registers to the transmit shift registers but neither the TDE nor the TUE flags are set. This means that during a disabled slot, no transmitter empty interrupt is generated. The Core is interrupted only for enabled slots. Data that is written to the transmit data registers when servicing this request is transmitted in the next enabled transmit time slot. When bit number N in ESAI_TSMB register is set, the transmit sequence is as usual: data is transferred from the TX registers to the shift registers and transmitted during slot number N, and the TDE flag is set. Using the slot mask in ESAI_TSMB does not conflict with using TSR. Even if a slot is enabled in TSMB, the user may chose to write to TSR instead of writing to the transmit data registers TXn. This causes all the transmit data pins of the enabled transmitters to be tri-stated during the next slot. Data written to the ESAI_TSMB affects the next frame transmission. The frame being transmitted is not affected by this data and would comply to the last ESAI_TSMB setting. Data read from ESAI_TSMB returns the last written data. After hardware or software reset, the ESAI_TSMB register is preset to 0x0000FFFF, which means that all 16 possible slots are enabled for data transmission.

25.6.20 Receive Slot Mask Register A (ESAI_RSMA)

The Receive Slot Mask Register A together with Receive Slot Mask Register B (ESAI_RSMA and ESAI_RSMB) are two read/write registers used by the receiver in network mode to determine for each slot whether to receive a data word and generate a receiver full condition (RDF=1), or to ignore the received data. Fields ESAI_RSMA [RS[15:0]] and ESAI_RSMB [RS31:16]] are concatenated to form the 32-bit field RS[31:0]. Bit number n in RS[31:0] is an enable/disable control bit for receiving data in slot number n.

Address: 202_4000h base + ECh offset = 202_40ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																RS[15:0]															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

ESAI_RSMA field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RS[15:0]	When bit number N in the ESAI_RSMA register is cleared, the data from the enabled receivers input pins are shifted into their receive shift registers during slot number N. The data is not transferred from the

Table continues on the next page...

ESAI_RSMA field descriptions (continued)

Field	Description
	receive shift registers to the receive data registers, and neither the RDF nor the ROE flag is set. This means that during a disabled slot, no receiver full interrupt is generated. The Core is interrupted only for enabled slots. When bit number N in the ESAI_RSMA is set, the receive sequence is as usual: data which is shifted into the enabled receivers shift registers is transferred to the receive data registers and the RDF flag is set. Data written to the ESAI_RSMA affects the next received frame. The frame being received is not affected by this data and would comply to the last ESAI_RSMA setting. Data read from ESAI_RSMA returns the last written data. After hardware or software reset, the ESAI_RSMA register is preset to 0x0000FFFF, which means that all 16 possible slots are enabled for data reception. When operating in normal mode, bit 0 of the ESAI_RSMA register must be set to one, otherwise no input is received.

25.6.21 Receive Slot Mask Register B (ESAI_RSMB)

The Receive Slot Mask Register B together with Receive Slot Mask Register A (ESAI_RSMA and ESAI_RSMB) are two read/write registers used by the receiver in network mode to determine for each slot whether to receive a data word and generate a receiver full condition (RDF=1), or to ignore the received data. Fields ESAI_RSMA [RS[15:0]] and ESAI_RSMB [RS31:16]] are concatenated to form the 32-bit field RS[31:0]. Bit number n in RS[31:0] is an enable/disable control bit for receiving data in slot number n.

Address: 202_4000h base + F0h offset = 202_40F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																RS[31:16]															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

ESAI_RSMB field descriptions

Field	Description
31–16 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
RS[31:16]	When bit number N in the ESAI_RSMB register is cleared, the data from the enabled receivers input pins are shifted into their receive shift registers during slot number N. The data is not transferred from the receive shift registers to the receive data registers, and neither the RDF nor the ROE flag is set. This means that during a disabled slot, no receiver full interrupt is generated. The Core is interrupted only for enabled slots. When bit number N in the ESAI_RSMB is set, the receive sequence is as usual: data which is shifted into the enabled receivers shift registers is transferred to the receive data registers and the RDF flag is set.

Table continues on the next page...

ESAI_RSMB field descriptions (continued)

Field	Description
	Data written to the ESAI_RSMB affects the next received frame. The frame being received is not affected by this data and would comply to the last ESAI_RSMB setting. Data read from ESAI_RSMB returns the last written data. After hardware or software reset, the ESAI_RSMB register is preset to 0x0000FFFF, which means that all 16 possible slots are enabled for data reception.

25.6.22 Port C Direction Register (ESAI_PRRC)

There are two registers to control the ESAI personal reset status: Port C Direction Register (ESAI_PRRC) and Port C Control Register (ESAI_PCRC).

The read/write 32-bit Port C Direction Register (ESAI_PRRC) in conjunction with the Port C Control Register (ESAI_PCRC) controls the functionality of the ESAI personal reset state. Table 25-48 provides the port pin configurations. Hardware and software reset clear all ESAI_PRRC bits.

Address: 202_4000h base + F8h offset = 202_40F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																PDC[11:0]															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ESAI_PRRC field descriptions

Field	Description
31–12 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
PDC[11:0]	See Table 25-48.

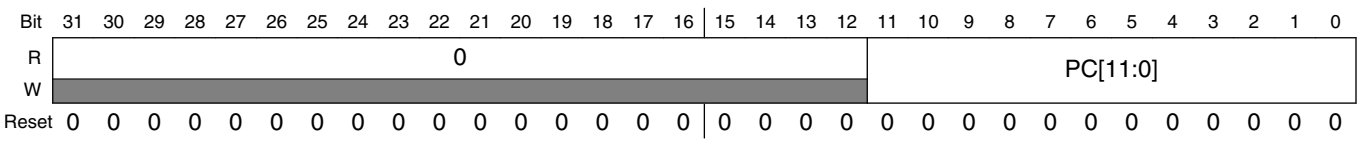
25.6.23 Port C Control Register (ESAI_PCRC)

The read/write 32-bit Port C Control Register (ESAI_PCRC) in conjunction with the Port C Direction Register (ESAI_PRRC) controls the functionality of the ESAI personal reset state. Each of the PC(11:0) bits controls the functionality of the corresponding port pin. Table 25-48 provides the port pin configurations. Hardware and software reset clear all ESAI_PCRC bits.

Table 25-48. PCRC and PRRC Bits Functionality

PDC[i]	PC[i]	Port Pin[i] Function
0	0	Disconnected
1	1	ESAI

Address: 202_4000h base + FCh offset = 202_40FCh



ESAI_PCRC field descriptions

Field	Description
31–12 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
PC[11:0]	See Table 25-48.

Chapter 26
Flexible Controller Area Network (FLEXCAN)

26.1 Overview

The Flexible Controller Area Network (FLEXCAN) module is a communication controller implementing the CAN protocol according to the CAN 2.0B protocol specification.

The CAN protocol was primarily designed to be used as a vehicle serial data bus meeting the specific requirements of this field: real-time processing, reliable operation in the EMI environment of a vehicle, cost-effectiveness and required bandwidth. The FLEXCAN module is a full implementation of the CAN protocol specification, which supports both standard and extended message frames. 64 Message Buffers are supported.

26.1.1 Block Diagram

A general block diagram is shown in the figure below,which describes the main sub-blocks implemented in the FLEXCAN module, including the associated memory for storing Mailboxes, Rx Global Mask Registers, Rx Individual Mask Registers, Rx FIFO and Rx FIFO ID Filters.

Support for 64 Mailboxes and 6-deep Rx FIFO is provided. The functions of the sub-modules are described in subsequent sections.

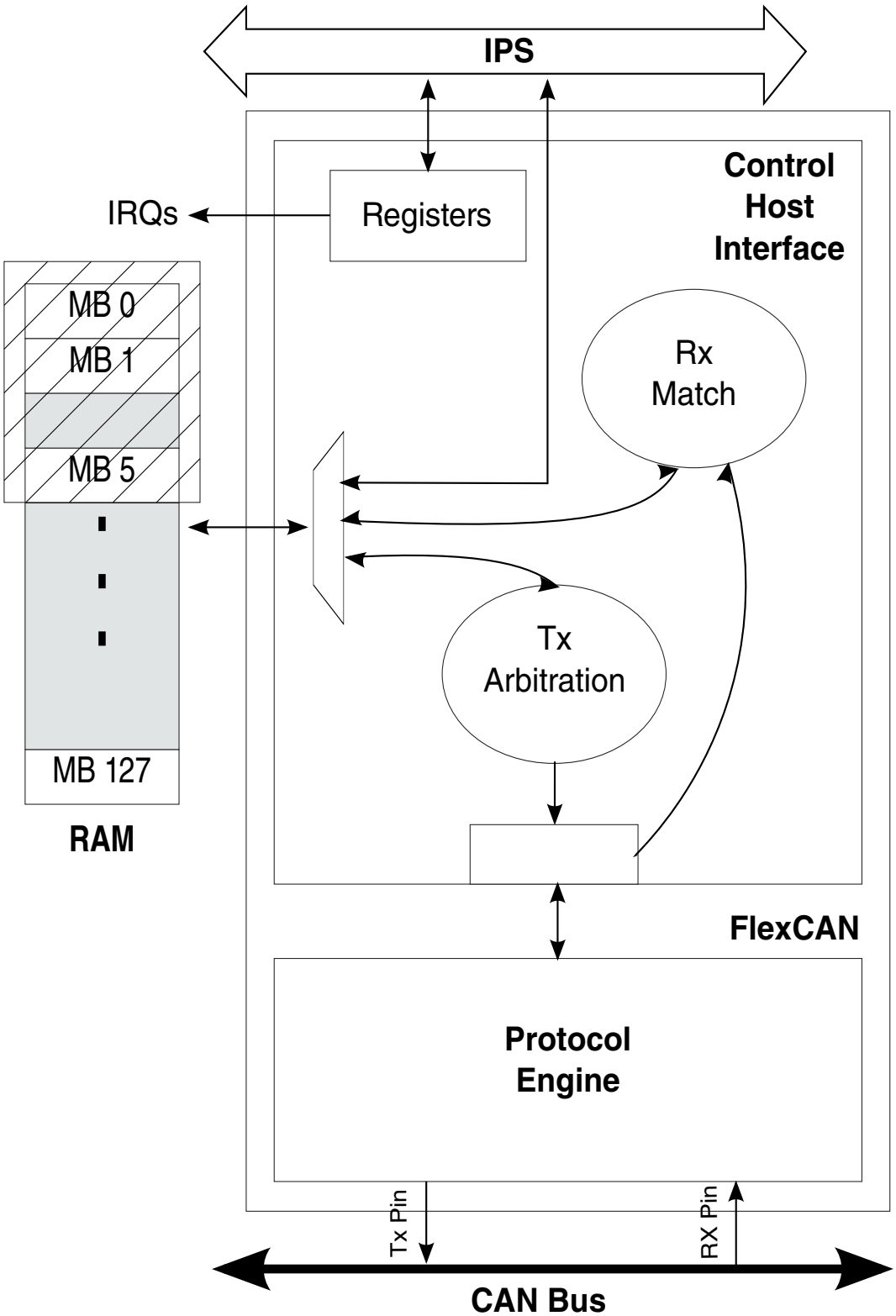


Figure 26-1. FLEXCAN Block Diagram

26.1.2 FLEXCAN Module Features

The FLEXCAN module includes these distinctive legacy features:

- Version 2.0B
  - Standard data and remote frames
  - Extended data and remote frames
  - Zero to eight bytes data length
  - Programmable bit rate up to 1 Mb/sec
  - Content-related addressing
- Flexible Mailboxes of eight bytes data length
- Each Mailbox is configurable as Rx or Tx, all supporting standard and extended messages
- Individual Rx Mask Registers per Mailbox
- Full featured Rx FIFO with storage capacity for 6 frames and internal pointer handling
- Transmission abort capability
- Powerful Rx FIFO ID filtering, capable of matching incoming IDs against either 128 extended, 256 standard or 512 partial (8 bits) IDs, with up to 32 individual masking capability
- 100% backwards compatibility with previous FLEXCAN version
- Unused structures space can be used as general purpose RAM space
- Listen only mode capability
- Programmable loop-back mode supporting self-test operation
- Programmable transmission priority scheme: lowest ID, lowest buffer number or highest priority
- Time Stamp based on 16-bit free-running timer
- Global network time, synchronized by a specific message
- Maskable interrupts independent of the transmission medium (an external transceiver is assumed)
- Short latency time due to an arbitration scheme for high-priority messages
- Low power modes, with programmable wake up on bus activity
- Configurable Glitch filter width to filter the noise on CAN bus when waking up
- Remote request frames may be handled automatically or by software.
- ID filter configuration in Normal Mode
- CAN bit time settings and configuration bits can only be written in Freeze Mode
- Tx mailbox status (Lowest priority buffer or empty buffer)
- SYNC bit status to inform that the module is synchronous with CAN bus
- CRC status for transmitted message
- Selectable priority between Mailboxes and Rx FIFO during matching process

26.1.3 Modes of Operation

The FLEXCAN module has four functional modes: Normal Mode (User and Supervisor), Freeze Mode, Listen-Only Mode and Loop-Back Mode. There are also two low power modes: Disable Mode and Stop Mode.

- Normal Mode (User or Supervisor):

In Normal Mode, the module operates receiving and/or transmitting message frames, errors are handled normally and all the CAN Protocol functions are enabled. User and Supervisor Modes differ in the access to some restricted control registers.

- Freeze Mode:

It is enabled when the FRZ bit in the MCR Register is asserted. If enabled, Freeze Mode is entered when the HALT bit in MCR is set or when Debug Mode is requested at MCU level and the FRZ_ACK bit in the MCR Register is asserted by the FlexCAN . In this mode, no transmission or reception of frames is done and synchronicity to the CAN bus is lost. See Freeze Mode for more information.

- Listen-Only Mode:

The module enters this mode when the LOM bit in the Control Register is asserted. In this mode, transmission is disabled, all error counters are frozen and the module operates in a CAN Error Passive mode. Only messages acknowledged by another CAN station will be received. If FLEXCAN detects a message that has not been acknowledged, it will flag a BIT0 error (without changing the REC), as if it was trying to acknowledge the message.

- Loop-Back Mode:

The module enters this mode when the LPB bit in the Control Register is asserted. In this mode, FLEXCAN performs an internal loop back that can be used for self test operation. The bit stream output of the transmitter is internally fed back to the receiver input. The FLEXCAN_RX input pin is ignored and the FLEXCAN_TX output goes to the recessive state (logic '1'). FLEXCAN behaves as it normally does when transmitting and treats its own transmitted message as a message received from a remote node. In this mode, FLEXCAN ignores the bit sent during the ACK slot in the CAN frame acknowledge field to ensure proper reception of its own message. Both transmit and receive interrupts are generated.

- Module Disable Mode:

This low power mode is entered when the MDIS bit in the MCR Register is asserted and the LPM_ACK is asserted by the FlexCAN. When disabled, the module requests to disable the clocks to the CAN Protocol Engine and Controller Host Interface sub-modules. Exit from this mode is done by negating the MDIS bit in the MCR Register. See Module Disable Mode for more information.

- Stop Mode:

This low power mode is entered when Stop Mode is requested at ARM level and the LPM_ACK bit in the MCR Register is asserted by the FlexCAN. When in Stop Mode, the module puts itself in an inactive state and then informs the ARM that the clocks can be shut down globally. Exit from this mode happens when the Stop Mode request is removed or when activity is detected on the CAN bus and the Self Wake Up mechanism is enabled. See Stop Mode for more information.

26.2 External Signals

The FLEXCAN module has two I/O signals.

Table 26-1. FLEXCAN External Signals

Signal	Description	Pad	Mode	Direction
FLEXCAN1_RX	FLEXCAN receive pin. This pin is the receive pin from the CAN bus transceiver. Dominant state is represented by logic level '0'. Recessive state is represented by logic level '1'.	GPIO_8	ALT3	I
		KEY_ROW2	ALT2	
		SD3_CLK	ALT2	
FLEXCAN1_TX	FLEXCAN transmit pin. This pin is the transmit pin to the CAN bus transceiver. Dominant state is represented by logic level '0'. Recessive state is represented by logic level '1'.	GPIO_7	ALT3	O
		KEY_COL2	ALT2	
		SD3_CMD	ALT2	
FLEXCAN2_RX	FLEXCAN receive pin. This pin is the receive pin from the CAN bus transceiver. Dominant state is represented by logic level '0'. Recessive state is represented by logic level '1'.	KEY_ROW4	ALT0	I
		SD3_DAT1	ALT2	
FLEXCAN2_TX	FLEXCAN transmit pin. This pin is the transmit pin to the CAN bus transceiver. Dominant state is represented by logic level '0'. Recessive state is represented by logic level '1'.	KEY_COL4	ALT0	O
		SD3_DAT0	ALT2	

26.3 Clocks

The table found here describes the clock sources for FLEXCAN.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 26-2. FLEXCAN Clocks

Clock name	Clock Root	Description
ipg_clk	ipg_clk_root	Peripheral clock
ipg_clk_chi	ipg_clk_root	CHI clock
ipg_clk_pe	can_clk_root	Protocol Engine clock
ipg_clk_pe_nogate	can_clk_root	Protocol Engine clock (no gating)
ipg_clk_s	ipg_clk_root	Peripheral access clock
mem_ram_CLK	ipg_clk_root	RAM clock

26.4 Message Buffer Structure

Message Buffer Address: Base + 0x0080-0x047C

The Message Buffer structure used by the FLEXCAN module is represented in the figure found here.

Both Extended and Standard Frames (29-bit Identifier and 11-bit Identifier, respectively) used in the CAN specification are represented.

Table 26-3. Message Buffer Structure

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0					CODE					SRR	IDE	RTR	DLC				TIME STAMP															
0x4	PRIO				ID Standard												ID Extended															
0x8	DATA BYTE 0								DATA BYTE 1								DATA BYTE 2								DATA BYTE 3							
0xC	DATA BYTE 4								DATA BYTE 5								DATA BYTE 6								DATA BYTE 7							

CODE - Message Buffer Code

This 4-bit field can be accessed (read or write) by the CPU and by the FLEXCAN module itself, as part of the message buffer matching and arbitration process. The encoding is shown in the following tables. See Functional Description for additional information.

Table 26-4. Message Buffer Code for Rx buffers

CODE Description	Rx Code BEFORE receive New Frame	SRV[1]	Rx Code AFTER successful reception[2]	RRS[3]	Comment
0b0000: INACTIVE- MB is not active.	INACTIVE	-	-	-	MB does not participate in the matching process.
0b0100: EMPTY - MB is active and empty.	EMPTY	-	FULL	-	When a frame is received successfully (after move-in process. Refer to Move-in for details), the CODE field is automatically updated to FULL.
0b0010: FULL - MB is full.	FULL	Yes	FULL	-	The act of reading the C/S word followed by unlocking the MB (SRV) does not make the code return to EMPTY. It remains FULL. If a new frame is moved to the MB after the MB was serviced, the code still remains FULL. Refer to Matching Process for matching details related to FULL code.
		No	OVERRUN	-	If the MB is FULL and a new frame is moved to this MB before the CPU services it, the CODE field is automatically updated to OVERRUN. Refer to Matching Process for details about overrun behavior.
0b0110: OVERRUN - MB is being overwritten into a full buffer.	OVERRUN	Yes	FULL	-	If the CODE field indicates OVERRUN and CPU has serviced the MB, when a new frame is moved to the MB, the code returns to FULL.
		No	OVERRUN	-	If the CODE field already indicates OVERRUN, and another new frame must be moved, the MB will be overwritten again, and the code will remain OVERRUN. Refer to Matching Process for details about overrun behavior.
0b1010: RANSWER[4] - A frame was configured to recognize a Remote Request Frame and transmit a Response Frame in return.	RANSWER	-	TANSWER(0b1110)	0	A Remote Answer was configured to recognize a remote request frame received, after that a MB is set to transmit a response frame. The code is automatically changed to TANSWER (0b1110). Refer to Matching Process for details. If CTRL2[RRS] is negated, transmit a response frame whenever a remote request frame with the same ID is received.
			-	1	This code is ignored during matching and arbitration process. Refer to Matching Process for details.

Table continues on the next page...

Table 26-4. Message Buffer Code for Rx buffers (continued)

CODE Description	Rx Code BEFORE receive New Frame	SRV[1]	Rx Code AFTER successful reception[2]	RRS[3]	Comment
CODE[0]=1b1: BUSY - FlexCAN is updating the contents of the MB. The CPU must not access the MB.	BUSY[5]	-	FULL	-	Indicates that the MB is being updated, it will be negated automatically and does not interfere on the next CODE.
			OVERRUN	-	

1. SRV: Serviced MB. MB was read and unlocked by reading TIMER or other MB.
2. A frame is considered successful reception after the frame to be moved to MB (move-in process). Refer to Move-in for details)
3. Remote Request Stored bit from CTRL2 register. Refer to CTRL2 for details.
4. Code 4'b1010 is not considered as a Tx and a MB with this code should not to be aborted.
5. Note that for Tx MBs, the BUSY bit should be ignored upon read, except when AEN bit is set in the MCR register. If this bit is asserted, the corresponding MB does not participate in the matching process.

Table 26-5. Message Buffer Code for Tx buffers

CODE Description	Tx Code BEFORE tx frame	MB RTR	Tx Code AFTER successful transmission	Comment
0b1000: INACTIVE - MB is not active	INACTIVE	-	-	MB does not participate in the arbitration process.
0b1001: ABORT - MB is aborted	ABORT	-	-	MB does not participate in the arbitration process. .
0b1100: DATA - MB is a Tx Data Frame (MB RTR must be 0)	DATA	0	INACTIVE	Transmit data frame unconditionally once. After transmission, the MB automatically returns to the INACTIVE state.
0b1100: REMOTE - MB is a Tx Remote Request Frame (MB RTR must be 1)	REMOTE	1	EMPTY	Transmit remote request frame unconditionally once. After transmission, the MB automatically becomes an Rx Empty MB with the same ID.
0b1110: TANSWER - MB is a Tx Response Frame from an incoming Remote Request Frame	TANSWER	-	RANSWER	This is an intermediate code that is automatically written to the MB by the CHI as a result of match to a remote request frame. The remote response frame will be transmitted unconditionally once and then the code will automatically return to RANSWER (0b1010). The CPU can also write this code with the same effect. The remote response frame can be either a data frame or another remote request frame depending on the RTR bit value. Refer to Matching Process and Arbitration process for details.

SRR - Substitute Remote Request

Fixed recessive bit, used only in extended format. It must be set to '1' by the user for transmission (Tx Buffers) and will be stored with the value received on the CAN bus for Rx receiving buffers. It can be received as either recessive or dominant. If FLEXCAN receives this bit as dominant, then it is interpreted as arbitration loss.

1= Recessive value is compulsory for transmission in Extended Format frames

0= Dominant is not a valid value for transmission in Extended Format frames

IDE - ID Extended Bit

This bit identifies whether the frame format is standard or extended. It is also used as part of the reception filter.

1= Frame format is extended

0= Frame format is standard

RTR - Remote Transmission Request

This bit affects the behavior of Remote Frames and is part of the reception filter. Refer to the tables above and RRS bit in Control 2 Register (FLEXCAN_CTRL2) for additional details.

If FLEXCAN transmits this bit as '1' (recessive) and receives it as '0' (dominant), it is interpreted as arbitration loss. If this bit is transmitted as '0' (dominant), then if it is received as '1' (recessive), the FLEXCAN module treats it as bit error. If the value received matches the value transmitted, it is considered as a successful bit transmission.

1= Indicates the current MB has a Remote Frame to be transmitted if MB is Tx. If the MB is Rx then incoming Remote Request Frames may be stored.

0= Indicates the current MB has a Data Frame to be transmitted. In Rx MB it may be considered in matching processes.

DLC - Length of Data in Bytes

This 4-bit field is the length (in bytes) of the Rx or Tx data, which is located in offset 0x08 through 0x0F of the MB space (see the first table above). In reception, this field is written by the FLEXCAN module, copied from the DLC (Data Length Code) field of the received frame. In transmission, this field is written by the ARM and corresponds to the DLC field value of the frame to be transmitted. When RTR=1, the Frame to be transmitted is a Remote Frame and does not include the data field, regardless of the Length field. The DLC field indicates which DATA BYTEs are valid as shown in the table below.

TIME STAMP - Free-Running Counter Time Stamp

This 16-bit field is a copy of the Free-Running Timer, captured for Tx and Rx frames at the time when the beginning of the Identifier field appears on the CAN bus.

PRIO - Local priority

This 3-bit field is only used when MCR[LPRIO_EN] bit is asserted and it only makes sense for Tx mailboxes. These bits are not transmitted. They are appended to the regular ID to define the transmission priority. See Arbitration process.

ID - Frame Identifier

In Standard Frame format, only the 11 most significant bits (28 to 18) are used for frame identification in both receive and transmit cases. The 18 least significant bits are ignored. In Extended Frame format, all bits are used for frame identification in both receive and transmit cases.

DATA BYTE 0-7 - Data Field

Up to eight bytes can be used for a data frame.

For Rx frames, the data is stored as it is received from the CAN bus. DATA BYTE (n) is valid only if *n* is less than DLC as shown in the table below.

For Tx frames, the CPU prepares the data field to be transmitted within the frame.

Table 26-6. DATA BYTEs validity

DLC	Valid DATA BYTEs
0	none
1	DATA BYTE 0
2	DATA BYTE 0-1
3	DATA BYTE 0-2
4	DATA BYTE 0-3
5	DATA BYTE 0-4
6	DATA BYTE 0-5
7	DATA BYTE 0-6
8	DATA BYTE 0-7

26.5 Rx FIFO Structure

When the MCR[RFEN] bit is set, the memory area from $80 to $DC (which is normally occupied by MBs 0 to 5) is used by the reception FIFO engine.

The region 0x80-0x8C contains the output of the FIFO which must be read by the CPU as a Message Buffer. This output contains the oldest message received and not read yet. The region 0x90-0xDC is reserved for internal use of the FIFO engine.

An additional memory area, that starts at 0xE0 and may extend up to 0x2DC (normally occupied by MBs 6 up to 37) depending on the CTRL2[RFFN] field setting, contains the ID Filter Table (configurable from 8 to 128 memory positions) that specifies filtering criteria for accepting frames into the FIFO.Table 26-7 shows the Rx FIFO data structure.

Each ID Filter Table Element occupies an entire 32-bit word and can be compounded by one, two or four Identifier Acceptance Filters (IDAF) depending on the MCR[IDAM] field setting. Table 26-8, Table 26-9 and Table 26-10 show the IDAF indexation. Table 26-11 show the three different formats that the IDAF can assume, depending on the MCR[IDAM] field setting. Note that all elements of the table must have the same format. See Rx FIFO for more information.

Out of reset, the ID Filter Table flexible memory area defaults to 0xE0 and only extends to 0xFC, which corresponds to MBs 6 to 7 for RFFN=0.

Table 26-7. Rx FIFO Structure

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x80										SRR	IDE	RTR	DLC				TIME STAMP															
0x84										ID Standard															ID Extended							
0x88	Data Byte 0												Data Byte 1												Data Byte 2							
0x8C	Data Byte 4												Data Byte 5												Data Byte 6							
0x90 to 0xDC	Reserved																															
0xE0	ID Filter Table Element 0																															
0xE4	ID Filter Table Element 1																															
0xE8 to 0x2D4	ID Filter Table Elements 2 through 125																															
0x2D8	ID Filter Table Element 126																															
0x2DC	ID Filter Table Element 127																															

Table 26-8. Position of ID Filter Table Elements in format A

Element	31																												0
0	IDAF0																												
1	IDAF1																												
2 through 125	Identifier Acceptance Filter 2 through 125																												
126	IDAF126																												
127	IDAF127																												

Table 26-9. Position of ID Filter Table Elements in format B

Element	31															16	15												0
0	IDAF0																IDAF1												
1	IDAF2																IDAF3												
2 through 125	Identifier Acceptance Filter 4 through 251																												
126	IDAF252																IDAF253												
127	IDAF254																IDAF255												

Table 26-10. Position of ID Filter Table Elements in format C

Element	31							24	23							16	15							8	7						0
0	IDAF0								IDAF1								IDAF2								IDAF3						
1	IDAF4								IDAF5								IDAF6								IDAF7						
2 through 125	Identifier Acceptance Filter 8 through 503																														
126	IDAF504								IDAF505								IDAF506								IDAF507						
127	IDAF508								IDAF509								IDAF510								IDAF511						

Table 26-11. Identifier Acceptance Filter Format A,B and C

Format	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A	RTR	IDE	RXIDA (Standard = 29-19, Extended = 29-1)																													
B	RTR	IDE	RXIDB_0 (Standard = 29-19, Extended = 29-16)														RTR	IDE	RXIDB_1 (Standard = 13-3, Extended = 13-0)														
C	RXIDC_0								RXIDC_1								RXIDC_2								RXIDC_3							

Table 26-11. Identifier Acceptance Filter Format A,B and C

	(Std/Ext = 31-24)	(Std/Ext = 23-16)	(Std/Ext = 15-8)	(Std/Ext = 7-0)

RTR - Remote Frame

This bit specifies whether Remote Request Frames are accepted into the FIFO if they match the target ID in Formats A and B. If Format C is chosen the acceptance does not depend on whether the frame is a Remote Request Frame or not.

1= Remote Frames can be accepted and data frames are rejected

0= Remote Frames are rejected and data frames can be accepted

IDE - Extended Frame

Specifies if either Extended or Standard Format frames are accepted into the FIFO if they match the target ID in Formats A and B. If Format C is chosen the acceptance does not depend on whether the frame is of the Extended or Standard Format.

1= Extended frames can be accepted and standard frames are rejected

0= Extended frames are rejected and standard frames can be accepted

RXIDA - Rx Frame Identifier (Format A)

Specifies an ID to be used as acceptance criteria for the FIFO. In the Standard Format (IDAF's or incoming frame's IDE bit is negated), only the 11 most significant bits (29 to 19) are used for frame identification. In the Extended Format (both IDAF's and incoming frame's IDE are asserted), all bits are used.

RXIDB_0, RXIDB_1 - Rx Frame Identifier (Format B)

Specifies an ID to be used as acceptance criteria for the FIFO. In the Standard Format (IDAF's or incoming frame's IDE bit is negated), the 11 most significant bits (29 to 19 and 13 to 3) are used for frame identification. In the Extended Format (both IDAF's and incoming frame's IDE are asserted), all 14 bits of the field are compared with the 14 most significant bits of the Identifier of the incoming frame. The 15 least significant bits of the Identifier of an incoming Extended Format frame do not affect the acceptance.

RXIDC_0, RXIDC_1, RXIDC_2, RXIDC_3 - Rx Frame Identifier (Format C)

Specifies an ID to be used as acceptance criteria for the FIFO. In both Standard Format and Extended Format, all 8 bits of the field are compared to the 8 most significant bits of the Identifier of the incoming frame. The 3 least significant bits of the Identifier of an incoming Standard Format frame and the 21 least significant bits of the Identifier of an incoming Extended Format frame do not affect the acceptance.

26.6 Functional Description

This section provides a complete functional description of the block.

26.6.1 Functional Overview

The FLEXCAN module is a CAN protocol engine with a very flexible mailbox system for transmitting and receiving CAN frames.

The mailbox system consists of a set of 64 Message Buffers (MB) that store configuration and control data, time stamp, message ID and data (see Message Buffer Structure). The memory corresponding to the first 38 MBs can be configured to support a FIFO reception scheme with a powerful ID filtering mechanism, capable of checking incoming frames against a table of IDs (up to 128 extended IDs or 256 standard IDs or 512 8-bit ID slices), with individual mask register for up to 32 ID Filter Table elements. Simultaneous reception through FIFO and mailbox is supported. For mailbox reception, a *matching* algorithm makes it possible to store received frames only into MBs that have the same ID. A masking scheme makes it possible to match the ID programmed on the MB with a range of Identifiers on received CAN frames. For transmission, an *arbitration* algorithm decides the prioritization of MBs to be transmitted based on the message ID (optionally augmented by 3 local priority bits) or the MB ordering.

Before proceeding with the functional description, an important concept must be explained. A Message Buffer is said to be "active" at a given time if it can participate in both the Matching and Arbitration processes. An Rx MB with a 0b0000 code is inactive (refer to Table 26-4). Similarly, a Tx MB with a 0b1000 or 0b1001 code is also inactive (refer to Table 26-5).

26.6.2 Transmit Process

In order to transmit a CAN frame, the CPU must prepare a Message Buffer for transmission by executing the procedure found here.

1. Check if the respective interruption bit is set and clear it.
2. If the MB is active (transmission pending), write the ABORT code (0b1001) to the CODE field of the Control and Status word to request an abortion of the transmission. Wait for the corresponding IFLAG to be asserted by polling the IFLAG register or by the interrupt request if enabled by the respective IMASK. Then read back the CODE field to check if the transmission was aborted or transmitted (see

Transmission Abort Mechanism). If backwards compatibility is desired (MCR[AEN] bit negated), just write the INACTIVE code (0b1000) to the CODE field to inactivate the MB but then the pending frame may be transmitted without notification (see Message Buffer Inactivation).

3. Write the ID word.
4. Write the data bytes.
5. Write the DLC, Control and Code fields of the Control and Status word to activate the MB.

Once the MB is activated, it will participate into the arbitration process and eventually be transmitted according to its priority.

At the end of the successful transmission, the value of the Free Running Timer at the time of the second bit of frame's Identifier field is written into the MB's Time Stamp field, the CODE field in the Control and Status word is updated, the CRC Register is updated, a status flag is set in the Interrupt Flag Register and an interrupt is generated if allowed by the corresponding Interrupt Mask Register bit. The new CODE field after transmission depends on the code that was used to activate the MB in step four (see Table 26-4 and Table 26-5 in Message Buffer Structure)

When the Abort feature is enabled (MCR[AEN] bit is asserted), after the Interrupt Flag is asserted for a Mailbox configured as transmit buffer, the Mailbox is blocked, therefore the CPU is not able to update it until it negates the Interrupt Flag. It means that the CPU must clear the corresponding IFLAG before starting to prepare this MB for a new transmission or reception.

26.6.3 Arbitration process

The arbitration process scans the Mailboxes searching the Tx one that holds the message to be sent in the next opportunity. This Mailbox is called the *arbitration winner*.

The scan starts from the lowest Mailbox number and runs toward the higher ones.

The arbitration process is triggered in the following events:

- From the CRC field of the CAN frame. The start point depends on the CTRL2[TASD] field value. See Control 2 Register (FLEXCAN_CTRL2) for details.
- During the error delimiter field of the CAN frame
- During the Overload Delimiter field of a CAN frame.
- When the winner is inactivated and the CAN bus has still not reached the first bit of the Intermission field.
- When ARM write to the C/S word of a winner MB and the CAN bus has still not reached the first bit of the Intermission field.

- When CHI is in Idle state and ARM writes to the C/S word of any MB.
- When FlexCAN exits Bus Off state
- Upon leaving Freeze Mode or Low Power Mode

If the arbitration process does not manage to evaluate all Mailboxes before the CAN bus has reached the first bit of the Intermission field the temporary arbitration winner is invalidated and the FlexCAN will not compete for the CAN bus in the next opportunity.

The arbitration process selects the winner among the active Tx Mailboxes at the end of the scan according to both CTRL1[LBUF] and MCR[LPRIO_EN] bits settings.

26.6.3.1 Lowest Mailbox number first

If CTRL1[LBUF] bit is asserted the first (lowest number) active Tx Mailbox found is the arbitration winner. MCR[LPRIO_EN] bit has no effect when CTRL1[LBUF] is asserted.

26.6.3.2 Highest Mailbox priority first

If CTRL1[LBUF] bit is negated then the arbitration process searches the active Tx Mailbox with the highest priority, and this Mailbox would have a higher probability to win the arbitration on CAN bus.

The sequence of bits considered for this arbitration is called the *arbitration value* of the Mailbox. The highest priority Tx Mailbox is the one that has the least arbitration value among all Tx Mailboxes.

If two or more Mailboxes have equivalent arbitration values the lowest Mailbox number is the arbitration winner.

The composition of the arbitration value depends on MCR[LPRIO_EN] bit setting.

26.6.3.2.1 Local Priority disabled

If MCR[LPRIO_EN] bit is negated the arbitration value is built in the exact sequence of bits as they would be transmitted in a CAN frame (see Table 26-12) in such a way that the Local Priority is disabled.

Table 26-12. Composition of the arbitration value when Local Priority is disabled

Format	Mailbox Arbitration Value (32 bits)				
Standard (IDE = 0)	Standard ID (11 bits)	RTR (1 bit)	IDE (1 bit)	- (18 bits)	- (1 bit)

Table continues on the next page...

Table 26-12. Composition of the arbitration value when Local Priority is disabled (continued)

Extended (IDE = 1)	Extended ID[28:18] (11 bits)	SRR (1 bit)	IDE (1 bit)	Extended ID[17:0] (18 bits)	RTR (1 bit)

26.6.3.2.2 Local Priority enabled

If Local Priority is desired MCR[LPRIO_EN] must be asserted.

In this case the Mailbox PRIO field is included at the very left of the arbitration value (see the table below).

Table 26-13. Composition of the arbitration value when Local Priority is enabled

Format	Mailbox Arbitration Value (35 bits)					
Standard (IDE = 0)	PRIO (3 bits)	Standard ID (11 bits)	RTR (1 bit)	IDE (1 bit)	- (18 bits)	- (1 bit)
Extended (IDE = 1)	PRIO (3 bits)	Extended ID[28:18](11 bits)	SRR (1 bit)	IDE (1 bit)	Extended ID[17:0] (18 bits)	RTR (1 bit)

As the PRIO field is the most significant part of the arbitration value Mailboxes with low PRIO values have higher priority than Mailboxes with high PRIO values regardless the rest of their arbitration values.

Note that the PRIO field is not part of the frame on the CAN bus. Its purpose is only to affect the internal arbitration process.

Once the arbitration winner is found, its content is copied to a hidden auxiliary MB called Tx Serial Message Buffer (Tx SMB), which has the same structure as a normal MB but is not user accessible. This operation is called "move-out" and after it is done, write access to the corresponding MB is blocked (if the AEN bit in MCR is asserted). The write access is released in the following events:

- After the MB is transmitted
- FlexCAN enters in Freeze Mode or Bus Off
- FlexCAN loses the bus arbitration or there is an error during the transmission

At the first opportunity window on the CAN bus, the message on the Tx SMB is transmitted according to the CAN protocol rules. FlexCAN transmits up to eight data bytes, even if the DLC (Data Length Code) field value is greater than that.

Arbitration process can be triggered in the following situations:

- During Rx and Tx frames from CAN CRC field to end of frame. Arbitration start point depends on instantiation parameters NUMBER_OF_MB and TASD. Additionally, TASD value may be changed (see Control 2 Register (FLEXCAN_CTRL2)) to optimize the arbitration start point.
- During CAN Bus Off state from TX_ERR_CNT=124 to 128. Arbitration start point depends on instantiation parameters NUMBER_OF_MB and TASD. Additionally, TASD value may be changed (see Control 2 Register (FLEXCAN_CTRL2)) to optimize the arbitration start point.
- During C/S write by CPU in Bus Idle. First C/S write starts arbitration process and a second C/S write during this same arbitration restarts the process. If other C/S writes are performed, Tx arbitration process is pending. If there is no arbitration winner after arbitration process has finished, then TX arbitration machine begins a new arbitration process.
- Arbitration winner deactivation during a valid arbitration window.
- Upon Leave Freeze Mode. If there is a re-synchronization during WaitForBusIdle arbitration process is restarted.

Arbitration process stops in the following situation:

- All Mailboxes were scanned.
- A Tx active Mailbox is found in case of Lowest Buffer feature enabled.
- Arbitration winner inactivation or abort during any arbitration process.
- There was not enough time to finish Tx arbitration process. For instance, a deactivation was performed near the end of frame). In this case arbitration process is pending.
- Error or Overload flag in the bus.
- Low Power or Freeze Mode request in Idle state

Arbitration is considered pending as described below:

- It was not possible to finish arbitration process in time.
- C/S write during arbitration if write is performed in a MB which number is lower than the Tx arbitration pointer.
- Any C/S write if there is no Tx Arbitration process in progress.
- Rx Match has just updated a Rx Code to Tx Code.
- Entering Bus off state.

C/S write during arbitration has the following effect:

- If C/S write is performed in the arbitration winner, a new process is restarted immediately.
- C/S write during arbitration if write is performed in a MB which number is higher than the Tx arbitration pointer.

26.6.4 Receive Process

To be able to receive CAN frames into a Mailbox, the CPU must prepare it for reception by executing the steps listed here.

1. If the Mailbox is active (either Tx or Rx) inactivate the Mailbox (see Message Buffer Inactivation), preferably with a *safe inactivation* (see Transmission Abort Mechanism);
2. Write the ID word;
3. Write the EMPTY code (0b0100) to the CODE field of the Control and Status word to activate the Mailbox.

Once the Mailbox is activated in the third step, it will be able to receive frames that match the programmed filter. At the end of a successful reception, the Mailbox is updated by the *move-in process* (see Move-in) as follows:

1. The received Data field (8 bytes at most) is stored;
2. The received Identifier field is stored;
3. The value of the Free Running Timer at the time of the second bit of frame's Identifier field is written into the Mailbox Time Stamp field;
4. The received SRR, IDE, RTR and DLC fields are stored;
5. The CODE field in the Control and Status word is updated. (see Table 26-4 and Table 26-5 in Section Message Buffer Structure)
6. A status flag is set in the Interrupt Flag Register and an interrupt is generated if allowed by the corresponding Interrupt Mask Register bit.

The recommended way for the CPU servicing (read) the frame received in a Mailbox is using the following procedure:

1. Read the Control and Status word of that Mailbox;
2. Check if the BUSY bit is deasserted, indicating that the Mailbox is locked. Repeat step 1) while it is asserted. See Message Buffer Lock Mechanism;
3. Read the contents of the Mailbox. Once Mailbox is locked now, its contents won't be modified by FlexCAN Move-in processes. See Move-in;
4. Acknowledge the proper flag at IFLAG registers;
5. Read the Free Running Timer. It is optional but recommended to unlock Mailbox as soon as possible and make it available for reception.

The CPU should synchronize to frame reception by the status flag bit for the specific Mailbox in one of the IFLAG Registers and not by the CODE field of that Mailbox. Polling the CODE field does not work because once a frame was received and the CPU services the Mailbox (by reading the C/S word followed by unlocking the Mailbox), the

CODE field will not return to EMPTY. It will remain FULL. If the CPU tries to workaround this behavior by writing to the C/S word to force an EMPTY code after reading the Mailbox without a prior *safe inactivation*, a newly received message matching the filter of that Mailbox may be lost.

In summary: never do polling by reading directly the C/S word of the Mailboxes. Instead, read the IFLAG registers.

Note that the received frame's Identifier field is always stored in the matching Mailbox, thus the contents of the ID field in a Mailbox may change if the match was due to masking. Note also that FlexCAN does receive frames transmitted by itself if there exists a matching Rx Mailbox, provided the MCR[SRX_DIS] bit is not asserted. If MCR[SRX_DIS] bit is asserted, FlexCAN will not store messages transmitted by itself in any MB, even if it contains a matching MB, and no interrupt flag or interrupt signal will be generated due to the frame reception.

To be able to receive CAN messages through the Rx FIFO, the CPU must enable and configure the Rx FIFO during Freeze Mode (see Rx FIFO). Upon receiving the Frames Available in Rx FIFO interrupt (see Interrupt Masks 1 Register (FLEXCAN_IMASK1), bit IFLAG[BUF5I] - Frames available in Rx FIFO), the CPU should service the received frame using the following procedure:

1. Read the Control and Status word (optional - needed only if a mask was used for IDE and RTR bits);
2. Read the ID field (optional - needed only if a mask was used);
3. Read the Data field;
4. Read the RXFIR register (optional);
5. Clear the Frames Available in Rx FIFO interrupt by writing 1 to IFLAG[BUF5I] bit (mandatory - releases the MB and allows the CPU to read the next Rx FIFO entry)

26.6.5 Matching Process

The matching process scans the MB memory looking for Rx MBs programmed with the same ID as the one received from the CAN bus.

If the FIFO is enabled, the priority of scanning can be selected between Mailboxes and FIFO filters. In any case, the matching starts from the lowest number Message Buffer toward the higher ones. If no match is found within the first structure then the other is scanned subsequently. In the event that the FIFO is full, the matching algorithm will always look for a matching MB outside the FIFO region.

As the frame is being received, it is stored in a hidden auxiliary MB called Rx Serial Message Buffer (Rx SMB).

The matching process start point depends on the following conditions:

- if the received frame is a remote frame, the start point is the CRC field of the frame;
- if the received frame is a data frame with DLC field equal to zero, the start point is the CRC field of the frame;
- if the received frame is a data frame with DLC field different than zero, the start point is the DATA field of the frame;

If a matching ID is found in the FIFO table or in one of the Mailboxes, the contents of the SMB will be transferred to the FIFO or to the matched Mailbox by the move-in process. If any CAN protocol error is detected then no match results will be transferred to the FIFO or to the matched Mailbox at the end of reception.

The matching process scans all matching elements of both Rx FIFO (if enabled) and active Rx Mailboxes (CODE is EMPTY, FULL, OVERRUN or RANSWER) in search of a successful comparison with the matching elements of the Rx SMB that is receiving the frame on the CAN bus. The SMB has the same structure of a Mailbox. The reception structures (Rx FIFO or Mailboxes) associated with the matching elements that had a successful comparison are the *matched structures* . The *matching winner* is selected at the end of the scan among those matched structures and depends on conditions described ahead. Please, refer to the following table for details.

Table 26-14. Matching architecture

Structure	SMB[RTR]	CTRL2[RRS]	CTRL2[EACEN]	MB[IDE]	MB[RTR]	MB[ID][1]	MB[CODE]
Mailbox	0	-	0	cmp[2]	no_cmp[3]	cmp_msk[4]	EMPTY or FULL or OVERRUN
Mailbox	0	-	1	cmp_msk	cmp_msk	cmp_msk	EMPTY or FULL or OVERRUN
Mailbox	1	0	-	cmp	no_cmp	cmp	RANSWER
Mailbox	1	1	0	cmp	no_cmp	cmp_msk	EMPTY or FULL or OVERRUN
Mailbox	1	1	1	cmp_msk	cmp_msk	cmp_msk	EMPTY or FULL or OVERRUN
FIFO[5]	-	-	-	cmp_msk	cmp_msk	cmp_msk	-

1. For Mailbox structure, If SMB[IDE] is asserted, the ID is 29 bits (ID Standard + ID Extended). In case of SMB[IDE] to be negated, the ID is only 11 bits (ID Standard). Please, refer to Message Buffer Structure for ID details. For FIFO structure, the ID depends on IDAM. Please, refer to Rx FIFO Structure for IDAM details.
2. cmp: Compares the SMB contents with the MB contents regardless the masks.
3. no_cmp: The SMB contents are not compared with the MB contents.
4. cmp_msk: Compares the SMB contents with MB contents taking into account the masks.

5. SMB[IDE] and SMB[RTR] are not taken into account when IDAM is type C.

A reception structure is *free-to-receive* when any of the following conditions is satisfied:

- the CODE field of the Mailbox is EMPTY;
- the CODE field of the Mailbox is either FULL or OVERRUN and it has already been serviced (the C/S word was read by the ARM and unlocked as described in Message Buffer Lock Mechanism);
- the CODE field of the Mailbox is either FULL or OVERRUN and an inactivation is performed. (see Message Buffer Inactivation)
- the Rx FIFO is not full.

The scan order for Mailboxes and Rx FIFO is from the matching element with lowest number to the higher ones.

The matching winner search for Mailboxes is affected by the MCR[IRMQ] bit. If it is negated the matching winner is the first matched Mailbox regardless if it is free-to-receive or not. If it is asserted, the matching winner is selected according to the priority below:

1. the first free-to-receive matched Mailbox;
2. the last non free-to-receive matched Mailbox.

It is possible to select the priority of scan between Mailboxes and Rx FIFO by the CTRL2[MRP] bit.

If the selected priority is Rx FIFO first:

- if the Rx FIFO is a matched structure and is free-to-receive then the Rx FIFO is the matching winner regardless of the scan for Mailboxes;
- otherwise (the Rx FIFO is not a matched structure or is not free-to-receive), then the matching winner is searched among Mailboxes as described above.

If the selected priority is Mailboxes first:

- if a free-to-receive matched Mailbox is found, it is the matching winner regardless the scan for Rx FIFO;
- if no matched Mailbox is found, then the matching winner is searched in the scan for the Rx FIFO;
- if both conditions above are not satisfied and a non free-to-receive matched Mailbox is found then the matching winner determination is conditioned by the MCR[IRMQ] bit:
  - if MCR[IRMQ] bit is negated the matching winner is the first matched Mailbox;
  - if MCR[IRMQ] bit is asserted the matching winner is the Rx FIFO if it is a free-to-receive matched structure, otherwise the matching winner is the last non free-to-receive matched Mailbox.

Please, refer to the table below for a summary of matching possibilities.

If a non-safe Mailbox inactivation (see Message Buffer Inactivation) occurs during matching process and the Mailbox inactivated is the temporary matching winner then the temporary matching winner is invalidated. The matching elements scan is not stopped nor restarted, it continues normally. The consequence is that the current matching process works as if the matching elements compared before the inactivation did not exist, therefore a message may be lost.

Suppose, for example, that the FIFO is disabled, IRMQ is enabled and there are two MBs with the same ID, and FlexCAN starts receiving messages with that ID. Let us say that these MBs are the second and the fifth in the array. When the first message arrives, the matching algorithm will find the first match in MB number 2. The code of this MB is EMPTY, so the message is stored there. When the second message arrives, the matching algorithm will find MB number 2 again, but it is not "free-to-receive", so it will keep looking and find MB number 5 and store the message there. If yet another message with the same ID arrives, the matching algorithm finds out that there are no matching MBs that are "free-to-receive", so it decides to overwrite the last matched MB, which is number 5. In doing so, it sets the CODE field of the MB to indicate OVERRUN.

Table 26-15. Matching Possibilities and Resulting Reception Structures

RFEN	IRMQ	MRP	Matched in MB	Matched in FIFO	Reception Structure	Description
No FIFO, only MB, match is always MB first						
0	0	X[1]	None[2]	-[3]	None	Frame lost by no match
0	0	X	Free[4]	-	FirstMB	
0	1	X	None	-	None	Frame lost by no match
0	1	X	Free	-	FirstMB	
0	1	X	NotFree	-	LastMB	Overrun
FIFO enabled, no match in FIFO is as if FIFO does not exist						
1	0	X	None	None[5]	None	Frame lost by no match
1	0	X	Free	None	FirstMB	
1	1	X	None	None	None	Frame lost by no match
1	1	X	Free	None	FirstMB	
1	1	X	NotFree	None	LastMB	Overrun
FIFO enabled, Queue disabled						
1	0	0	X	NotFull[6]	FIFO	
1	0	0	None	Full[7]	None	Frame lost by FIFO full (FIFO Overflow)
1	0	0	Free	Full	FirstMB	
1	0	0	NotFree	Full	FirstMB	
1	0	1	None	NotFull	FIFO	
1	0	1	None	Full	None	Frame lost by FIFO full (FIFO Overflow)

Table continues on the next page...

Table 26-15. Matching Possibilities and Resulting Reception Structures (continued)

RFEN	IRMQ	MRP	Matched in MB	Matched in FIFO	Reception Structure	Description
1	0	1	Free	X	FirstMB	
1	0	1	NotFree	X	FirtsMB	Overrun
FIFO enabled, Queue enabled						
1	1	0	X	NotFull	FIFO	
1	1	0	None	Full	None	Frame lost by FIFO full (FIFO Overflow)
1	1	0	Free	Full	FirstMB	
1	1	0	NotFree	Full	LastMB	Overrun
1	1	1	None	NotFull	FIFO	
1	1	1	Free	X	FirstMB	
1	1	1	NotFree	NotFull	FIFO	
1	1	1	NotFree	Full	LastMB	Overrun

1. It is a don't care condition.
2. Matched in MB "None" means that the frame has not matched any MB (free-to-receive or non-free-to-receive).
3. It is a forbidden condition.
4. Matched in MB "Free" means that the frame matched at least one MB free-to-receive regardless it has matched MBs non-free-to-receive.
5. Matched in FIFO "None" means that the frame has not matched any filter in FIFO. It is as the FIFO didn't exist (CTRL2[RFEN]=0).
6. Matched in FIFO "NotFull" means that the frame has matched a FIFO filter and has empty slots to receive it.
7. Matched in FIFO "Full" means that the frame has matched a FIFO filter but couldn't store it because it has no empty slots to receive it.

The ability to match the same ID in more than one MB can be exploited to implement a reception queue (in addition to the full featured FIFO) to allow more time for ARM to service the MBs. By programming more than one MB with the same ID, received messages will be queued into the MBs. ARM can examine the Time Stamp field of the MBs to determine the order in which the messages arrived.

Matching to a range of IDs is possible by using ID Acceptance Masks. FlexCAN supports individual masking per MB. Please refer to Rx Mailboxes Global Mask Register (FLEXCAN_RXMGMASK) . During the matching algorithm, if a mask bit is asserted, then the corresponding ID bit is compared. If the mask bit is negated, the corresponding ID bit is "don't care". Please note that the Individual Mask Registers are implemented in RAM, so they are not initialized out of reset. Also, they can only be programmed while the module is in Freeze Mode , otherwise they are blocked by hardware.

FlexCAN also supports an alternate masking scheme with only four mask registers (RGXMASK, RX14MASK, RX15MASK and RXFGMASK) for backward compatibility. This alternate masking scheme is enabled when the IRMQ bit in the MCR Register is negated.

26.6.6 Move Process

There are two types of move process, namely move-in and move-out.

26.6.6.1 Move-in

The move-in process is the copy of a message received by an Rx SMB to a Rx Mailbox or FIFO that has matched it. If the move destination is the Rx FIFO, attributes of the message are also copied to the RXFIR FIFO. Each Rx SMB has its own move-in process, but only one is performed at a given time as described ahead.

The move-in starts only when the message held by the Rx SMB has a corresponding matching winner (see Matching Process) and all of the following conditions are true:

- the CAN bus has reached or let past either:
  - the second bit of Intermission field next to the frame that carried the message that is in the Rx SMB;
  - the first bit of an overload frame next to the frame that carried the message that is in the Rx SMB;
- there is no ongoing matching process;
- the destination Mailbox is not locked by ARM;
- there is no ongoing move-in process from another Rx SMB. If more than one move-in processes are to be started at the same time both are performed and the newest substitutes the oldest.

The term *pending move-in* is used throughout the document and stands for a move-to-be that still does not satisfy all of the aforementioned conditions.

The move-in is cancelled and the Rx SMB is able to receive another message if any of the following conditions is satisfied:

- the destination Mailbox is inactivated after the CAN bus has reached the first bit of Intermission field next to the frame that carried the message and its matching process has finished;
- there is a previous pending move-in to the same destination Mailbox.
- the Rx SMB is receiving a frame transmitted by the FlexCAN itself and the self-reception is disabled;
- any CAN protocol error is detected.

Note that the pending move-in is not cancelled if the module enters in Freeze or Low Power Mode. It only stays on hold waiting for exiting Low Power Mode and to be unlocked. If an MB is unlocked during Freeze Mode, the move-in happens immediately.

The move-in process consists of the following steps:

1. if the message is destined to the Rx FIFO, push IDHIT into the RXFIR FIFO;
2. reads the words DATA0-3 and DATA4-7 from the Rx SMB;
3. writes it in the words DATA0-3 and DATA4-7 of the Rx Mailbox;
4. reads the words Control/Status and ID from the Rx SMB;
5. writes it in the words Control/Status and ID of the Rx Mailbox, updating the CODE field according to Table 26-4.

The move-in process is not atomic, in such a way that it is immediately cancelled by the inactivation of the destination Mailbox (see Message Buffer Inactivation) and in this case the Mailbox may be left partially updated, thus incoherent. The exception is if the move-in destination is an Rx FIFO Message Buffer, then the process cannot be cancelled.

The BUSY Bit (least significant bit of the CODE field) of the destination Message Buffer is asserted while the move-in is being performed in such a way that ARM beware that the Message Buffer content is temporarily incoherent.

26.6.6.2 Move-out

The move-out process is the copy of the content from a Tx Mailbox to the Tx SMB when a message for transmission is available (see Arbitration process).

The move-out occurs in the following conditions:

- the first bit of Intermission field;
- during Bus off field when TX Error Counter is in the 124 to 128 range;
- during BusIdle field
- during Wait For Bus Idle field

The move-out process is not atomic. Only ARM has priority to access the memory concurrently out of BusIdle state. In BusIdle, the move-out has the lowest priority to the concurrent memory accesses.

26.6.7 Data Coherence

In order to maintain data coherency and FlexCAN proper operation, the ARM must obey the rules described in

Any form of ARM accessing an MB structure within FlexCAN other than those specified may cause FlexCAN to behave in an unpredictable way.

26.6.7.1 Transmission Abort Mechanism

The abort mechanism provides a safe way to request the abortion of a pending transmission. A feedback mechanism is provided to inform ARM if the transmission was aborted or if the frame could not be aborted and was transmitted instead.

In order to abort a transmission, ARM must write a specific abort code (0b1001) to the CODE field of the Control and Status word. The active MBs configured as transmission must be aborted first and then they may be updated. If the abort code is written to a Mailbox that is currently being transmitted, or to a Mailbox that was already loaded into the SMB for transmission, the write operation is blocked and the MB is kept active, but the abort request is captured and kept pending until one of the following conditions are satisfied:

- The module loses the bus arbitration
- There is an error during the transmission
- The module is put into Freeze Mode
- The module enters in BusOff state
- There is an overload frame

If none of conditions above are reached, the MB is transmitted correctly, the interrupt flag is set in the IFLAG register and an interrupt to the ARM is generated (if enabled). The abort request is automatically cleared when the interrupt flag is set. In the other hand, if one of the above conditions is reached, the frame is not transmitted, therefore the abort code is written into the CODE field, the interrupt flag is set in the IFLAG and an interrupt is (optionally) generated to ARM.

If ARM writes the ABORT code before the transmission begins internally, then the write operation is not blocked, therefore the MB is updated and the interrupt flag is set. In this way ARM just needs to read the abort code to make sure the active MB was *safely inactivated* . Although the AEN bit is asserted and ARM wrote the abort code, in this case the MB is inactivated and not aborted, because the transmission did not start yet. One Mailbox is only aborted when the abort request is captured and kept pending until one of the previous conditions are satisfied.

The abort procedure can be summarized as follows:

1. ARM checks the corresponding IFLAG and clears it, if asserted.
2. ARM writes 0b1001 into the CODE field of the C/S word.
3. ARM waits for the corresponding IFLAG indicating that the frame was either transmitted or aborted.
4. ARM reads the CODE field to check if the frame was either transmitted (CODE=0b1000) or aborted (CODE=0b1001).

5. It is necessary to clear the corresponding IFLAG in order to allow the MB to be reconfigured.

26.6.7.2 Message Buffer Inactivation

Inactivation is a mechanism provided to protect the Mailbox against updates by the FlexCAN internal processes, thus allowing ARM to rely on Mailbox data coherence after having updated it, even in Normal Mode.

If a Mailbox is inactivated it does not participate neither in the arbitration nor in the matching process until it is reactivated. See Transmit Process and Receive Process for more detailed instruction on how to inactivate and reactivate a Mailbox.

In order to inactivate a Mailbox ARM must update its CODE field to INACTIVE (either 0b0000 or 0b1000).

As the user is not able to synchronize the CODE field update with the FlexCAN internal processes an inactivation can lead to undesirable results:

- a frame in the bus that matches the filtering of the inactivated Rx Mailbox may be lost without notice, even if there are other Mailboxes with the same filter;
- a frame containing the message within the inactivated Tx Mailbox may be transmitted without notice.

In order to eliminate such risk and perform a *safe inactivation* ARM must use the following mechanism along with the inactivation itself:

- for Tx Mailboxes, the Transmission Abort (see Transmission Abort Mechanism);

The inactivation automatically unlocks the Mailbox (see Message Buffer Lock Mechanism).

Message Buffers that are part of the Rx FIFO cannot be inactivated. There is no write protection on FIFO region by FlexCAN. ARM must keep the data coherence into FIFO region when RFEN is asserted.

26.6.7.3 Message Buffer Lock Mechanism

Besides MB inactivation, FlexCAN has another data coherence mechanism for the receive process. When ARM reads the Control and Status word of an Rx MB with codes FULL or OVERRUN, FlexCAN assumes that ARM wants to read the whole MB in an

atomic operation, and thus it sets an internal lock flag for that MB. The lock is released when ARM reads the Free Running Timer (global unlock operation), or when it reads the Control and Status word of another MB regardless of its code or when ARM writes into C/S word from locked MB. The MB locking is done to prevent a new frame to be written into the MB while ARM is reading it.

The locking mechanism only applies to Rx MBs that are not part of FIFO and have a code different than INACTIVE (0b0000) or EMPTY[1] (0b0100). Also, Tx MBs can not be locked.

Suppose, for example, that the FIFO is disabled and the second and the fifth MBs of the array are programmed with the same ID, and FlexCAN has already received and stored messages into these two MBs. Suppose now that the ARM decides to read MB number 5 and at the same time another message with the same ID is arriving. When ARM reads the Control and Status word of MB number 5, this MB is locked. The new message arrives and the matching algorithm finds out that there are no "free-to-receive" MBs, so it decides to override MB number 5. However, this MB is locked, so the new message can not be written there. It will remain in the SMB waiting for the MB to be unlocked, and only then will be written to the MB. If the MB is not unlocked in time and yet another new message with the same ID arrives, then the new message overwrites the one on the SMB and there will be no indication of lost messages either in the CODE field of the MB or in the Error and Status Register.

While the message is being moved-in from the SMB to the MB, the BUSY bit on the CODE field is asserted. If ARM reads the Control and Status word and finds out that the BUSY bit is set, it should defer accessing the MB until the BUSY bit is negated.

If the BUSY bit is asserted or if the MB is empty, then reading the Control and Status word does not lock the MB.

Inactivation takes precedence over locking. If ARM inactivates a locked Rx Mailbox, then its lock status is negated and the Mailbox is marked as invalid for the current matching round. Any pending message on the SMB will not be transferred anymore to the Mailbox.An MB is unlocked when ARM reads the Free Running Timer Register (see Free Running Timer Register (FLEXCAN_TIMER)), or the C/S word of another MB.

Lock and unlock mechanisms have the same functionality in both Normal and Freeze modes.

An unlock during Normal or Freeze mode results in the move-in of the pending message. However, the move-in is postponed if an unlock occurs during any of the low power modes (see in Modes of Operation specific information on Module Disable or Stop modes) and it will take place only when the module resumes to Normal or Freeze modes.

1. In previous FlexCAN versions, reading the C/S word locks the MB even if it is EMPTY. This behavior is maintained when the IRMQ bit is negated.

26.6.8 Rx FIFO

The receive-only FIFO is enabled by asserting the RFEN bit in the MCR.

The reset value of this bit is zero to maintain software backward compatibility with previous versions of the module that did not have the FIFO feature. The FIFO is 6-message deep, therefore when the FIFO is enabled, the memory region occupied by the first 6 Message Buffers is reserved for use of the FIFO engine (see Rx FIFO Structure). ARM can read the received messages sequentially, in the order they were received, by repeatedly reading a Message Buffer structure at the output of the FIFO.

The IFLAG[BUF5I] (Frames available in Rx FIFO) is asserted when there is at least one frame available to be read from the FIFO. An interrupt is generated if it is enabled by the corresponding mask bit. Upon receiving the interrupt, ARM can read the message (accessing the output of the FIFO as a Message Buffer) and the RXFIR register and then clear the interrupt. If there are more messages in the FIFO the act of clearing the interrupt updates the output of the FIFO with the next message and update the RXFIR with the attributes of that message, reissuing the interrupt to ARM. Otherwise, the flag remains negated. The output of the FIFO is only valid when the IFLAG[BUF5I] is asserted.

The IFLAG[BUF6I] (Rx FIFO Warning) is asserted when the number of unread messages within the Rx FIFO is increased to 5 from 4 due to the reception of a new one, meaning that the Rx FIFO is almost full. The flag remains asserted until ARM clears it.

The IFLAG[BUF7I] (Rx FIFO Overflow) is asserted when an incoming message was lost because the Rx FIFO is full. Note that the flag will not be asserted when the Rx FIFO is full and the message was captured by a Mailbox. The flag remains asserted until the ARM clears it.

Clearing one of those three flags does not affect the state of the other two.

An interrupt is generated if an IFLAG bit is asserted and the corresponding mask bit is asserted too.

A powerful filtering scheme is provided to accept only frames intended for the target application, thus reducing the interrupt servicing work load. The filtering criteria is specified by programming a table of up to 128 32-bit registers, according to CTRL2[RFFN] setting, that can be configured to one of the following formats (see also Rx FIFO Structure):

- Format A: 128 IDAFs (extended or standard IDs including IDE and RTR)
- Format B: 256 IDAFs (standard IDs or extended 14-bit ID slices including IDE and RTR)
- Format C: 512 IDAFs (standard or extended 8-bit ID slices)

Every frame available in the FIFO has a corresponding IDHIT (Identifier Acceptance Filter Hit Indicator) that can be read by accessing the RXFIR register. The RXFIR[IDHIT] field refers to the message at the output of the FIFO and is valid whilst the IFLAG[BUF5I] flag is asserted. The RXFIR register must be read only before clearing the flag, which guarantees that the information refers to the correct frame within the FIFO.

Up to thirty two elements of the ID Filter Table are individually affected by the Individual Mask Registers (RXIMR0 - RXIMR31), according to CTRL2[RFFN] setting (refer to Control 2 Register (FLEXCAN_CTRL2)), allowing very powerful filtering criteria to be defined. If the MCR[IRMQ] bit is negated (or if the RXIMR are not available for the particular MCU), then the FIFO ID Filter Table is affected by RXFGMASK.

26.6.9 CAN Protocol Related Features

26.6.9.1 Remote Frames

Remote frame is a special kind of frame. The user can program a mailbox to be a Remote Request Frame by writing the mailbox as Transmit with the RTR bit set to '1'. After the remote request frame is transmitted successfully, the mailbox becomes a Receive Message Buffer, with the same ID as before.

When a remote request frame is received by FlexCAN, it can be treated in three ways, depending on Remote Request Storing (CTRL2[RRS]) and Rx FIFO Enable (MCR[RFEN]) bits:

- If RRS is negated the frame's ID is compared to the IDs of the Transmit Message Buffers with the CODE field 0b1010. If there is a matching ID, then this mailbox frame will be transmitted. Note that if the matching mailbox has the RTR bit set, then FlexCAN will transmit a remote frame as a response. The received remote request frame is not stored in a receive buffer. It is only used to trigger a transmission of a frame in response. The mask registers are not used in remote frame matching, and all ID bits (except RTR) of the incoming received frame should match. In the case that a remote request frame was received and matched a mailbox, this message buffer immediately enters the internal arbitration process, but is considered as normal Tx mailbox, with no higher priority. The data length of this frame is independent of the DLC field in the remote frame that initiated its transmission.
- If RRS is asserted the frame's ID is compared to the IDs of the receive mailboxes with the CODE field 0b0100, 0b0010 or 0b0110. If there is a matching ID, then this

mailbox will store the remote frame in the same fashion of a data frame. No automatic remote response frame will be generated. The mask registers are used in the matching process.

- If RFEN is asserted FlexCAN will not generate an automatic response for remote request frames that match the FIFO filtering criteria. If the remote frame matches one of the target IDs, it will be stored in the FIFO and presented to the ARM. Note that for filtering formats A and B, it is possible to select whether remote frames are accepted or not. For format C, remote frames are always accepted (if they match the ID).Remote Request Frames are considered as normal frames, and generate a FIFO overflow when a successful reception occurs and the FIFO is already full.

26.6.9.2 Overload Frames

FLEXCAN does transmit overload frames due to detection of following conditions on CAN bus:

- Detection of a dominant bit in the first/second bit of Intermission
- Detection of a dominant bit at the 7th bit (last) of End of Frame field (Rx frames)
- Detection of a dominant bit at the 8th bit (last) of Error Frame Delimiter or Overload Frame Delimiter

26.6.9.3 Time Stamp

The value of the Free Running Timer is sampled at the beginning of the Identifier field on the CAN bus, and is stored at the end of "move-in" in the TIME STAMP field, providing network behavior with respect to time.

Note that the Free Running Timer can be reset upon a specific frame reception, enabling network time synchronization. Refer to TSYN description in Control 1 Register (FLEXCAN_CTRL1).

26.6.9.4 Protocol Timing

The FLEXCAN module supports a variety of means to setup bit timing parameters that are required by the CAN protocol. The Control Register has various fields used to control bit timing parameters: PRESDIV, PROPSEG, PSEG1, PSEG2 and RJW. See Control 1 Register (FLEXCAN_CTRL1) .

The PRESDIV field controls a prescaler that generates the Serial Clock (Sclock), whose period defines the 'time quantum' used to compose the CAN waveform. A time quantum is the atomic unit of time handled by the CAN engine.

$$f_{Tq} = \frac{f_{CANCLK}}{(\text{Prescaler value})}$$

A bit time is subdivided into three segments[2] (reference Table 26-16):

- SYNC_SEG: This segment has a fixed length of one time quantum. Signal edges are expected to happen within this section
- Time Segment 1: This segment includes the Propagation Segment and the Phase Segment 1 of the CAN standard. It can be programmed by setting the PROPSEG and the PSEG1 fields of the CTRL Register so that their sum (plus 2) is in the range of 4 to 16 time quanta
- Time Segment 2: This segment represents the Phase Segment 2 of the CAN standard. It can be programmed by setting the PSEG2 field of the CTRL Register (plus 1) to be 2 to 8 time quanta long

$$\text{Bit Rate} = \frac{f_{Tq}}{(\text{number of Time Quanta})}$$

2. For further explanation of the underlying concepts please refer to ISO/DIS 11519-1, Section 10.3. Reference also the Bosch CAN 2.0A/B protocol specification dated September 1991 for bit timing.

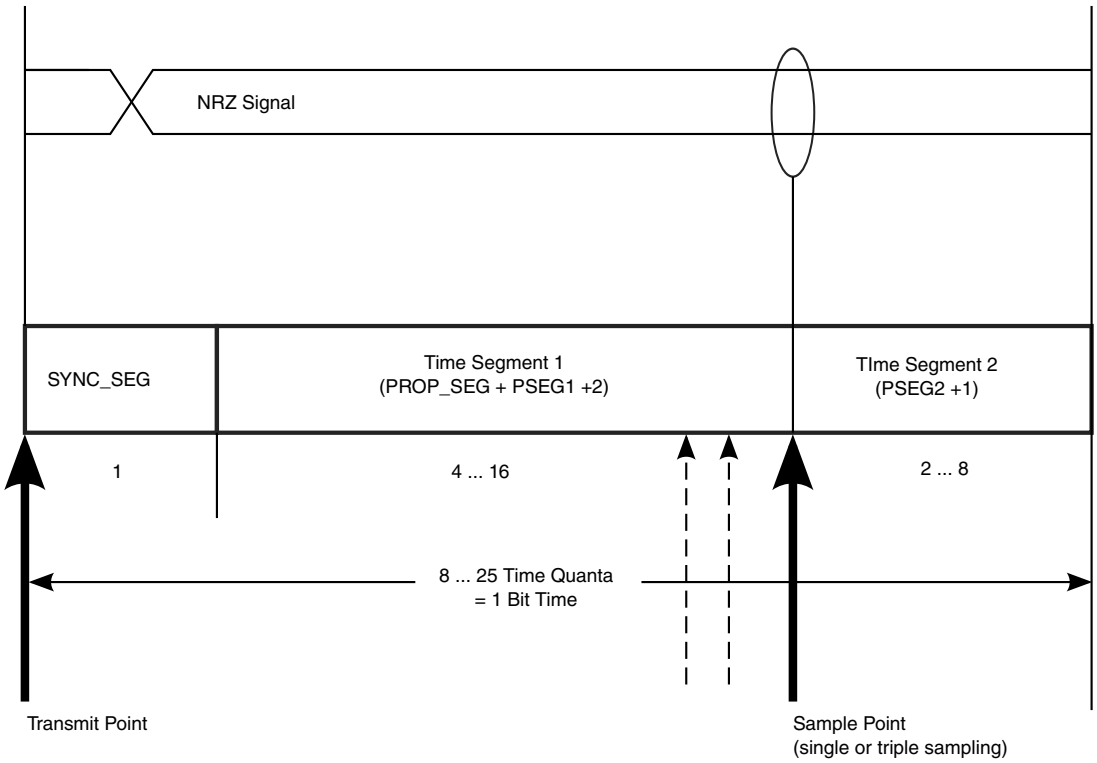


Figure 26-2. Segments within the Bit Time

Whenever CAN bit is used as a measure of duration (e.g. MCR[FRZ_ACK] and MCR[LPM_ACK] in Module Configuration Register (FLEXCAN_MCR)), the number of peripheral clocks in one CAN bit can be calculated as:

$$NCCP = \frac{f_{sys} \times [1 + (PSEG1 + 1) + (PSEG2 + 1)] \times (PRESDIV + 1)}{f_{CANCLK}}$$

where:

NCCP is the number of peripheral clocks in one CAN bit;

f_{CANCLK} is the Protocol Engine (PE) Clock in Hz;

f_{SYS} is the frequency of operation of the system (CHI) clock, in Hz;

PSEG1 is the value in CTRL1[PSEG1] field;

PSEG2 is the value in CTRL1[PSEG2] field;

PROPSEG is the value in CTRL1[PROPSEG] field;

PRESDIV is the value in CTRL1[PRESDIV] field.

For example, 180 CAN bits = 180 x NCCP peripheral clock periods.

Figure 26-2 gives an overview of the CAN compliant segment settings and the related parameter values.

Table 26-16. Time Segment Syntax

Syntax	Description
SYNC_SEG	System expects transitions to occur on the bus during this period.
Transmit Point	A node in transmit mode transfers a new value to the CAN bus at this point.
Sample Point	A node samples the bus at this point. If the three samples per bit option is selected, then this point marks the position of the third sample.

Table 26-17. CAN Standard Compliant Bit Time Segment Settings

Time Segment 1	Time Segment 2	Re-synchronization Jump Width
5 .. 10	2	1 .. 2
4 .. 11	3	1 .. 3
5 .. 12	4	1 .. 4
6 .. 13	5	1 .. 4
7 .. 14	6	1 .. 4
8 .. 15	7	1 .. 4
9 .. 16	8	1 .. 4

26.6.9.5 Arbitration and Matching Timing

During normal reception and transmission of frames, the matching, arbitration, move-in and move-out processes are executed during certain time windows inside the CAN frame, as shown in the following figures.

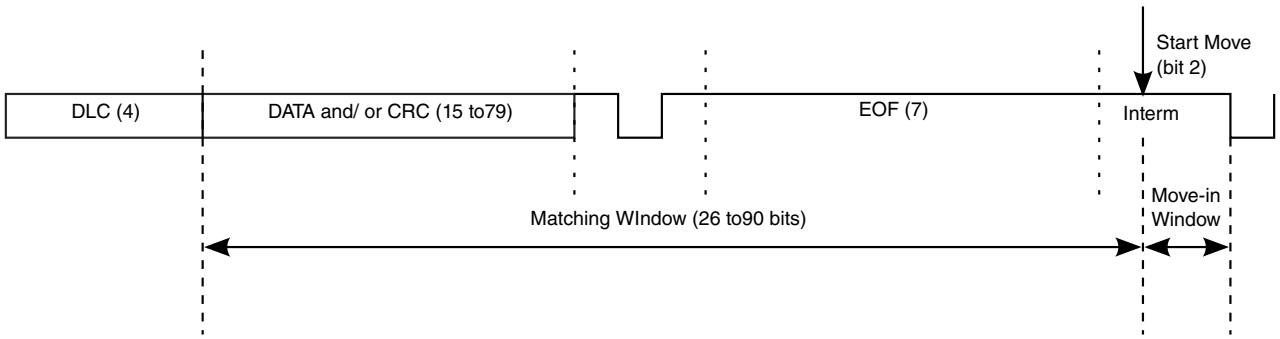


Figure 26-3. Matching and Move-In Time Windows

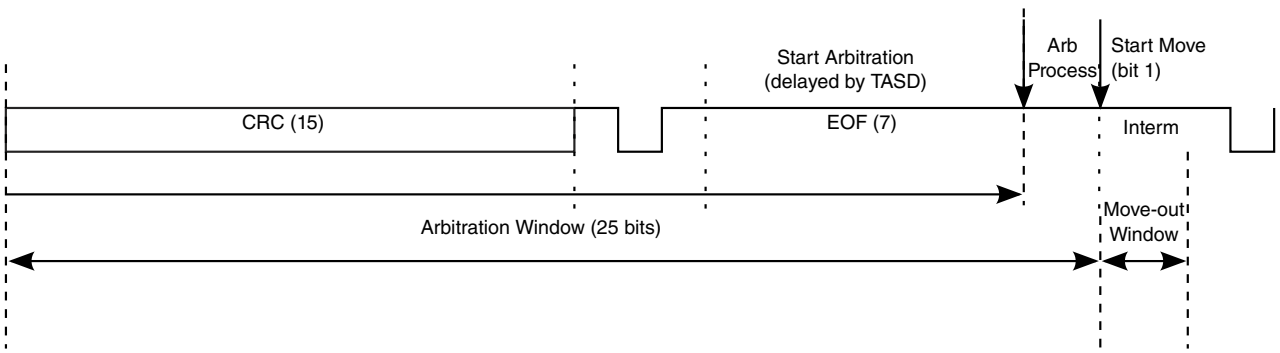


Figure 26-4. Arbitration and Move-Out Time Windows

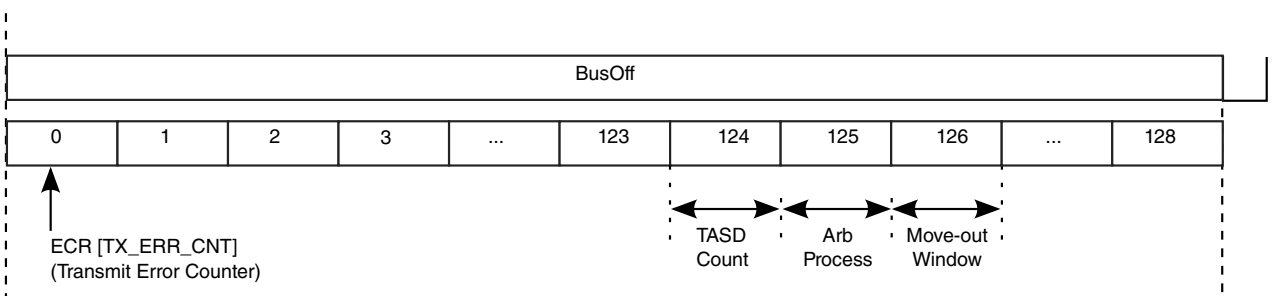


Figure 26-5. Arbitration at the end of Bus Off and Move-Out Time Windows

When doing matching and arbitration, FlexCAN needs to scan the whole Message Buffer memory during the available time window. In order to have sufficient time to do that, the following requirements must be observed:

- A valid CAN bit timing must be programmed, as indicated in Table 26-17
- The peripheral clock frequency can not be smaller than the oscillator clock frequency, i.e. the PLL can not be programmed to divide down the oscillator clock
- There must be a minimum ratio between the peripheral clock frequency and the CAN bit rate, as specified in the following table.

Table 26-18. Minimum Ratio Between Peripheral Clock Frequency and CAN Bit Rate

Number of Message Buffers	RFEN	Minimum Number of Peripheral Clocks per CAN bit
16 and 32	0	16
64	0	25
16	1	16
32	1	17
64	1	30

A direct consequence of the first requirement is that the minimum number of time quanta per CAN bit must be 8, so the oscillator clock frequency should be at least 8 times the CAN bit rate. The minimum frequency ratio specified in Table 26-18 can be achieved by choosing a high enough peripheral clock frequency when compared to the oscillator clock frequency, or by adjusting one or more of the bit timing parameters (PRESDIV, PROPSEG, PSEG1, PSEG2). As an example, taking the case of 64 MBs, if the oscillator and peripheral clock frequencies are equal and the CAN bit timing is programmed to have 8 time quanta per bit, then the prescaler factor (PRESDIV + 1) should be at least 2. For prescaler factor equal to one and CAN bit timing with 8 time quanta per bit, the ratio between peripheral and oscillator clock frequencies should be at least 2.

26.6.10 Modes of Operation Details

The FlexCAN module has four functional modes (Normal Mode, Freeze Mode, Listen-Only Mode and Loop-Back Mode) and two low power modes (Disable Mode and Stop Mode).

See in Modes of Operation an introductory description of all these modes of operation. The following sub-sections bring functional details on Freeze mode and the low power modes.

26.6.10.1 Freeze Mode

This mode is requested by ARM through the assertion of the HALT bit in the MCR Register or when the MCU is put into Debug Mode . In both cases it is also necessary that the FRZ bit is asserted in the MCR Register and the module is not in any of the low power modes (Disable, Stop). The acknowledgement is obtained through the assertion by the FlexCAN of FRZ_ACK bit in the same register. The ARM must only consider the FlexCAN in Freeze Mode when both request and acknowledgement conditions are satisfied.

When Freeze Mode is requested during transmission or reception, FlexCAN does the following:

- Waits to be in either Intermission, Passive Error, Bus Off or Idle state
- Waits for all internal activities like arbitration, matching, move-in and move-out to finish. Pending move-in is not taken in account
- Ignores the FLEXCAN_RX input pin and drives the FLEXCAN_TX pin as recessive
- Stops the prescaler, thus halting all CAN protocol activities
- Grants write access to the Error Counters Register, which is read-only in other modes
- Sets the NOT_RDY and FRZ_ACK bits in MCR

After requesting Freeze Mode, the user must wait for the FRZ_ACK bit to be asserted in MCR before executing any other action, otherwise FlexCAN may operate in an unpredictable way. In Freeze mode, all memory mapped registers are accessible, except for CTRL1[CLK_SRC] bit that can be read but cannot be written.

Exiting Freeze Mode is done in one of the following ways:

- ARM negates the FRZ bit in the MCR Register
- The ARM is removed from Debug Mode and the HALT bit is negated

The FRZ_ACK bit is negated after protocol engine recognizes the negation of freeze request. Once out of Freeze Mode, FlexCAN tries to re-synchronize to the CAN bus by waiting for 11 consecutive recessive bits.

26.6.10.2 Module Disable Mode

This low power mode is normally used to temporarily disable a complete FlexCAN block, with no power consumption. It is requested by the ARM through the assertion of the MDIS bit in the MCR Register and the acknowledgement is obtained through the assertion by the FlexCAN of the LPM_ACK bit in the same register. The ARM must only consider the FlexCAN in Disable Mode when both request and acknowledgement conditions are satisfied.

If the module is disabled during Freeze Mode, it requests to disable the clocks to the PE and CHI sub-modules, sets the LPM_ACK bit and negates the FRZ_ACK bit. The ability to shut down the clocks depends on how FlexCAN is integrated into the MCU. If the module is disabled during transmission or reception, FlexCAN does the following:

- Waits to be in either Idle or Bus Off state, or else waits for the third bit of Intermission and then checks it to be recessive
- Waits for all internal activities like arbitration, matching, move-in and move-out to finish. Pending move-in is not taken in account
- Ignores its FLEXCAN_RX input pin and drives its FLEXCAN_TX pin as recessive

- May shut down the clocks to the PE and CHI sub-modules, depending on how FlexCAN is integrated into the MCU
- Sets the NOT_RDY and LPM_ACK bits in MCR

The Bus Interface Unit continues to operate, enabling the ARM to access memory mapped registers, except the Rx Mailboxes Global Mask Registers, the Rx Buffer 14 Mask Register, the Rx Buffer 15 Mask Register, the Rx FIFO Global Mask Register. The Rx FIFO Information Register, the Message Buffers, the Rx Individual Mask Registers, and the reserved words within RAM may not be accessed when the module is in Disable Mode depending on how FlexCAN RAM is integrated into the ARM. Exiting from this mode is done by negating the MDIS bit by ARM, which make FlexCAN requests to resume the clocks and negates the LPM_ACK bit after CAN protocol engine recognizes the negation of disable mode requested by ARM.

26.6.10.3 Stop Mode

This is a system low power mode in which system clocks can be stopped for maximum power savings.. To enter stop mode, the CPU should manually assert a global Stop Mode request (see the CAN1_STOP_REQ and CAN2_STOP_REQ bit in the register IOMUXC_GPR13)and check the acknowledgement asserted by the FlexCAN (see the CAN1_STOP_ARK and CAN2_STOP_ARK in the register IOMUXC_GPR4) . The CPU must only consider the FlexCAN in Stop Mode when both request and acknowledgement conditions are satisfied.

If FlexCAN receives the global Stop Mode request during Freeze Mode, it sets the LPM_ACK bit, negates the FRZ_ACK bit and then sends the Stop Acknowledge signal to the CPU, in order to shut down the clocks globally. If Stop Mode is requested during transmission or reception, FlexCAN does the following:

- Waits to be in either Idle or Bus Off state, or else waits for the third bit of Intermission and checks it to be recessive
- Waits for all internal activities like arbitration, matching, move-in and move-out to finish.Pending move-in is not taken in account
- Ignores its FLEXCAN_RX input pin and drives its FLEXCAN_TX pin as recessive
- Sets the NOT_RDY and LPM_ACK bits in MCR
- Sends a Stop Acknowledge signal to the CPU, so that it can shut down the clocks globally

Exiting Stop Mode is done in one of the following ways:

- ARM resuming the clocks and removing the Stop Mode request
- ARM resuming the clocks and Stop Mode request as a result of the Self Wake mechanism

In the Self Wake mechanism, if the SLF_WAK bit in MCR Register was set at the time FlexCAN entered Stop Mode, then upon detection of a recessive to dominant transition on the CAN bus, FlexCAN sets the WAK_INT bit in the ESR Register and, if enabled by the WAK_MSK bit in MCR, generates a Wake Up interrupt to the ARM. Upon receiving the interrupt, the ARM should resume the clocks and remove the Stop Mode request manually. FlexCAN will then wait for 11 consecutive recessive bits to synchronize to the CAN bus. As a consequence, it will not receive the frame that woke it up.

The sensitivity to CAN bus activity can be modified by applying a low-pass filter function to the FLEXCAN_RX input line while in Stop Mode. See the WAK_SRC bit in Module Configuration Register (FLEXCAN_MCR) . This feature can be used to protect FlexCAN from waking up due to short glitches on the CAN bus lines. Such glitches can result from electromagnetic interference within noisy environments.

26.6.11 Interrupts

The module can generate up to 70 interrupt sources (64 interrupts due to message buffers and 6 interrupts due to Ored interrupts from MBs, Bus Off, Error, Tx Warning, Rx Warning and Wake Up)).

The number of actual sources depends on the configured number of message buffers.

Each one of the message buffers can be an interrupt source, if its corresponding IMASK bit is set. There is no distinction between Tx and Rx interrupts for a particular buffer, under the assumption that the buffer is initialized for either transmission or reception. Each of the buffers has assigned a flag bit in the IFLAG Registers. The bit is set when the corresponding buffer completes a successful transmission/reception and is cleared when the ARM writes it to '1' (unless another interrupt is generated at the same time).

If the Rx FIFO is enabled (bit RFEN on MCR set), the interrupts corresponding to MBs 0 to 7 have a different behavior. Bit 7 of the IFLAG1 becomes the "FIFO Overflow" flag; bit 6 becomes the FIFO Warning flag, bit 5 becomes the "Frames Available in FIFO flag" and bits 4-0 are unused. See Interrupt Flags 1 Register (FLEXCAN_IFLAG1) for more information.

A combined interrupt for all MBs is also generated by an Or of all the interrupt sources from MBs. This interrupt gets generated when any of the Mailboxes or FIFO generates an interrupt. The ARM must read the IFLAG Registers to determine which MB or FIFO caused the interrupt.

The other 5 interrupt sources (Bus Off, Error, Tx Warning, Rx Warning and Wake Up generate interrupts like the MB ones, and can be read from both the Error and Status Register 1 and 2. The Bus Off, Error, Tx Warning and Rx Warning interrupt mask bits are located in the Control 1 Register and the Wake-Up interrupt mask bit is located in the MCR.

26.7 Initialization/Application Information

This section provide instructions for initializing the FLEXCAN module.

26.7.1 FLEXCAN Initialization Sequence

The FLEXCAN module may be reset in two ways:

- SOC level hard reset which resets all memory mapped registers asynchronously
- SOFT_RST bit in MCR, which resets some of the memory mapped registers synchronously

Soft reset is synchronous and has to follow an internal request/acknowledge procedure across clock domains. Therefore, it may take some time to fully propagate its effects. The SOFT_RST bit remains asserted while soft reset is pending, so software can poll this bit to know when the reset has completed. Also, soft reset can not be applied while clocks are shut down in any of the low power modes. The low power mode should be exited and the clocks resumed before applying soft reset.

After the module is enabled (MDIS bit negated), FLEXCAN automatically goes to Freeze Mode. In Freeze Mode, FLEXCAN is un-synchronized to the CAN bus, the HALT and FRZ bits in MCR Register are set, the internal state machines are disabled and the FRZ_ACK and NOT_RDY bits in the MCR Register are set. The FLEXCAN_TX pin is in recessive state and FLEXCAN does not initiate any transmission or reception of CAN frames. Note that the Message Buffers and the Rx Individual Mask Registers are not affected by reset, so they are not automatically initialized.

For any configuration change/initialization it is required that FLEXCAN is put into Freeze Mode. The following is a generic initialization sequence applicable to the FLEXCAN module:

- Initialize the Module Configuration Register
  - Enable the individual filtering per MB and reception queue features by setting the IRMQ bit
  - Enable the warning interrupts by setting the WRN_EN bit

    - If required, disable frame self reception by setting the SRX_DIS bit
    - Enable the FIFO by setting the RFEN bit
    - Enable the abort mechanism by setting the AEN bit
    - Enable the local priority feature by setting the LPRIO_EN bit
- Initialize the Control Register
    - Determine the bit timing parameters: PROPSEG, PSEG1, PSEG2, RJW
    - Determine the bit rate by programming the PRESDIV field
    - Determine the internal arbitration mode (LBUF bit)
- Initialize the Message Buffers
    - The Control and Status word of all Message Buffers must be initialized
    - If FIFO was enabled, the 8-entry ID table must be initialized
    - Other entries in each Message Buffer should be initialized as required
- Initialize the Rx Individual Mask Registers
- Set required interrupt mask bits in the IMASK Registers (for all MB interrupts), in CTRL Register (for Bus Off and Error interrupts) and in MCR Register for Wake-Up interrupt
- Negate the HALT bit in MCR

Starting with the last event, FLEXCAN attempts to synchronize to the CAN bus.

26.8 FLEXCAN Memory Map/Register Definition

The complete memory map for a FLEXCAN module with 64 MBs capability is shown in the following table. Each individual register is identified by its complete name and the corresponding mnemonic. The access type can be Supervisor (S) or Unrestricted (U). Most of the registers can be configured to have either Supervisor or Unrestricted access by programming the SUPV bit in the MCR Register. The MCR register allows only Supervisor access regardless the SUPV bit state.

The FLEXCAN module stores CAN messages for transmission and reception using a Mailboxes and Rx FIFO structure.

FLEXCAN memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_0000	Module Configuration Register (FLEXCAN1_MCR)	32	R/W	5980_000Fh	26.8.1/1326
209_0004	Control 1 Register (FLEXCAN1_CTRL1)	32	R/W	0000_0000h	26.8.2/1331
209_0008	Free Running Timer Register (FLEXCAN1_TIMER)	32	R/W	0000_0000h	26.8.3/1334

Table continues on the next page...

FLEXCAN memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_0010	Rx Mailboxes Global Mask Register (FLEXCAN1_RXMGMASK)	32	R/W	FFFF_FFFFh	26.8.4/1334
209_0014	Rx Buffer 14 Mask Register (FLEXCAN1_RX14MASK)	32	R/W	FFFF_FFFFh	26.8.5/1335
209_0018	Rx Buffer 15 Mask Register (FLEXCAN1_RX15MASK)	32	R/W	FFFF_FFFFh	26.8.6/1336
209_001C	Error Counter Register (FLEXCAN1_ECR)	32	R/W	0000_0000h	26.8.7/1337
209_0020	Error and Status 1 Register (FLEXCAN1_ESR1)	32	R/W	0000_0000h	26.8.8/1338
209_0024	Interrupt Masks 2 Register (FLEXCAN1_IMASK2)	32	R/W	0000_0000h	26.8.9/1342
209_0028	Interrupt Masks 1 Register (FLEXCAN1_IMASK1)	32	R/W	0000_0000h	26.8.10/ 1342
209_002C	Interrupt Flags 2 Register (FLEXCAN1_IFLAG2)	32	R/W	0000_0000h	26.8.11/ 1343
209_0030	Interrupt Flags 1 Register (FLEXCAN1_IFLAG1)	32	R/W	0000_0000h	26.8.12/ 1343
209_0034	Control 2 Register (FLEXCAN1_CTRL2)	32	R/W	0000_0000h	26.8.13/ 1345
209_0038	Error and Status 2 Register (FLEXCAN1_ESR2)	32	R	0000_0000h	26.8.14/ 1351
209_0044	CRC Register (FLEXCAN1_CRCR)	32	R	0000_0000h	26.8.15/ 1353
209_0048	Rx FIFO Global Mask Register (FLEXCAN1_RXFGMASK)	32	R/W	FFFF_FFFFh	26.8.16/ 1354
209_004C	Rx FIFO Information Register (FLEXCAN1_RXFIR)	32	R	0000_0000h	26.8.17/ 1355
209_0880	Rx Individual Mask Registers (FLEXCAN1_RXIMR0_RXIMR63)	32	R/W	0000_0000h	26.8.18/ 1356
209_09E0	Glitch Filter Width Registers (FLEXCAN1_GFWR)	32	R/W	0000_007Fh	26.8.19/ 1356
209_4000	Module Configuration Register (FLEXCAN2_MCR)	32	R/W	5980_000Fh	26.8.1/1326
209_4004	Control 1 Register (FLEXCAN2_CTRL1)	32	R/W	0000_0000h	26.8.2/1331
209_4008	Free Running Timer Register (FLEXCAN2_TIMER)	32	R/W	0000_0000h	26.8.3/1334
209_4010	Rx Mailboxes Global Mask Register (FLEXCAN2_RXMGMASK)	32	R/W	FFFF_FFFFh	26.8.4/1334
209_4014	Rx Buffer 14 Mask Register (FLEXCAN2_RX14MASK)	32	R/W	FFFF_FFFFh	26.8.5/1335
209_4018	Rx Buffer 15 Mask Register (FLEXCAN2_RX15MASK)	32	R/W	FFFF_FFFFh	26.8.6/1336
209_401C	Error Counter Register (FLEXCAN2_ECR)	32	R/W	0000_0000h	26.8.7/1337
209_4020	Error and Status 1 Register (FLEXCAN2_ESR1)	32	R/W	0000_0000h	26.8.8/1338
209_4024	Interrupt Masks 2 Register (FLEXCAN2_IMASK2)	32	R/W	0000_0000h	26.8.9/1342
209_4028	Interrupt Masks 1 Register (FLEXCAN2_IMASK1)	32	R/W	0000_0000h	26.8.10/ 1342
209_402C	Interrupt Flags 2 Register (FLEXCAN2_IFLAG2)	32	R/W	0000_0000h	26.8.11/ 1343

Table continues on the next page...

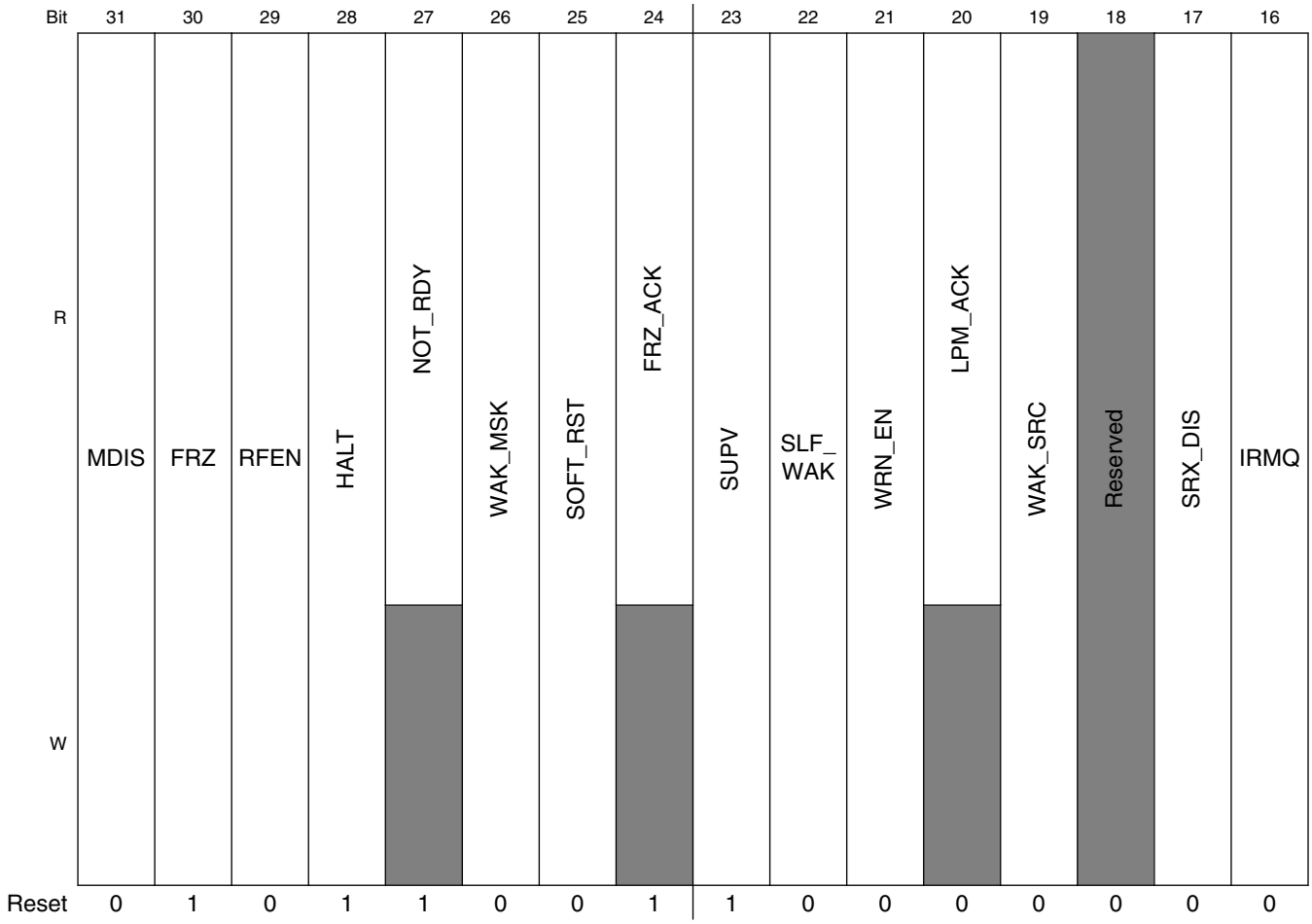
FLEXCAN memory map (continued)

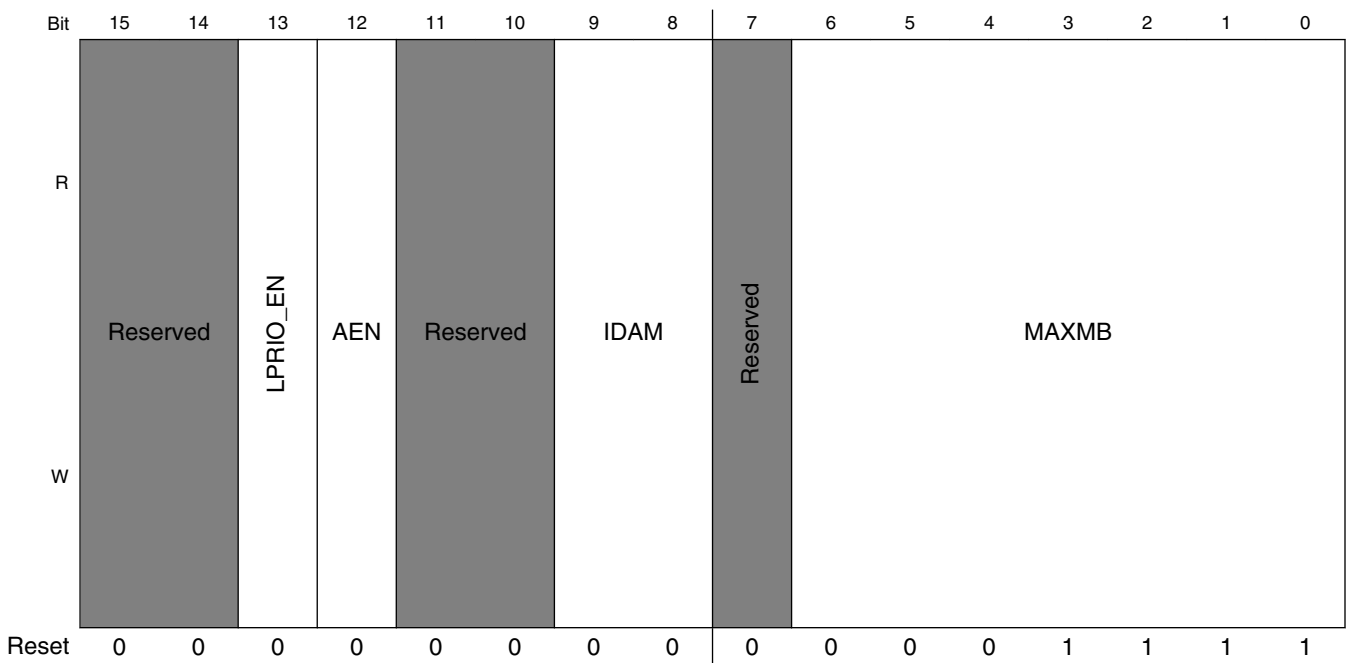
Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_4030	Interrupt Flags 1 Register (FLEXCAN2_IFLAG1)	32	R/W	0000_0000h	26.8.12/ 1343
209_4034	Control 2 Register (FLEXCAN2_CTRL2)	32	R/W	0000_0000h	26.8.13/ 1345
209_4038	Error and Status 2 Register (FLEXCAN2_ESR2)	32	R	0000_0000h	26.8.14/ 1351
209_4044	CRC Register (FLEXCAN2_CRCR)	32	R	0000_0000h	26.8.15/ 1353
209_4048	Rx FIFO Global Mask Register (FLEXCAN2_RXFGMASK)	32	R/W	FFFF_FFFFh	26.8.16/ 1354
209_404C	Rx FIFO Information Register (FLEXCAN2_RXFIR)	32	R	0000_0000h	26.8.17/ 1355
209_4880	Rx Individual Mask Registers (FLEXCAN2_RXIMR0_RXIMR63)	32	R/W	0000_0000h	26.8.18/ 1356
209_49E0	Glitch Filter Width Registers (FLEXCAN2_GFWR)	32	R/W	0000_007Fh	26.8.19/ 1356

26.8.1 Module Configuration Register (FLEXCANx_MCR)

This register defines global system configurations, such as the module operation mode (e.g., low power) and maximum message buffer configuration.

Address: Base address + 0h offset





FLEXCANx_MCR field descriptions

Field	Description
31 MDIS	This bit controls whether FLEXCAN is enabled or not. When disabled, FLEXCAN shuts down the clocks to the CAN Protocol Interface and Message Buffer Management sub-modules. This is the only bit in MCR not affected by soft reset. See Module Disable Mode for more information. 1 Disable the FLEXCAN module 0 Enable the FLEXCAN module
30 FRZ	The FRZ bit specifies the FLEXCAN behavior when the HALT bit in the MCR Register is set or when Debug Mode is requested at ARM level. When FRZ is asserted, FLEXCAN is enabled to enter Freeze Mode. Negation of this bit field causes FLEXCAN to exit from Freeze Mode. 1 Enabled to enter Freeze Mode 0 Not enabled to enter Freeze Mode
29 RFEN	This bit controls whether the Rx FIFO feature is enabled or not. When RFEN is set, MBs 0 to 5 cannot be used for normal reception and transmission because the corresponding memory region (0x80-0xDC) is used by the FIFO engine as well as additional MBs (up to 32, depending on CTRL2[RFFN] setting) which are used as Rx FIFO ID Filter Table elements.RFEN also impacts the definition of the minimum number of peripheral clocks per CAN bit as described in Table 26-18 (see Arbitration and Matching Timing).This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 FIFO enabled 0 FIFO not enabled
28 HALT	Assertion of this bit puts the FLEXCAN module into Freeze Mode. The ARM should clear it after initializing the Message Buffers and Control Register. No reception or transmission is performed by FLEXCAN before this bit is cleared. Freeze Mode can not be entered while FLEXCAN is in any of the low power modes.See Freeze Mode for more information 1 Enters Freeze Mode if the FRZ bit is asserted. 0 No Freeze Mode request.
27 NOT_RDY	This read-only bit indicates that FLEXCAN is either in Disable Mode, Stop Mode or Freeze Mode. It is negated once FLEXCAN has exited these modes. 1 FLEXCAN module is either in Disable Mode, Stop Mode or Freeze Mode 0 FLEXCAN module is either in Normal Mode, Listen-Only Mode or Loop-Back Mode
26 WAK_MSK	This bit enables the Wake Up Interrupt generation. 1 Wake Up Interrupt is enabled 0 Wake Up Interrupt is disabled
25 SOFT_RST	When this bit is asserted, FlexCAN resets its internal state machines and some of the memory mapped registers. The following registers are reset: MCR (except the MDIS bit), TIMER, ECR, ESR1, ESR2, IMASK1, IMASK2, IFLAG1, IFLAG2 and CRCR. Configuration registers that control the interface to the CAN bus are not affected by soft reset. The following registers are unaffected: CTRL1, CTRL2, RXIMR0_RXIMR63, RXGMASK, RX14MASK, RX15MASK, RXFGMASK, RXFIR and all Message Buffers The SOFT_RST bit can be asserted directly by the ARM when it writes to the MCR Register. It may take some time to fully propagate its effect. The SOFT_RST bit remains asserted while reset is pending, and is automatically negated when reset completes. Therefore, software can poll this bit to know when the soft reset has completed. Soft reset cannot be applied while clocks are shut down in any of the low power modes. The module should be first removed from low power mode, and then soft reset can be applied.

Table continues on the next page...

FLEXCANx_MCR field descriptions (continued)

Field	Description
	1 Reset the registers 0 No reset request
24 FRZ_ACK	This read-only bit indicates that FLEXCAN is in Freeze Mode and its prescaler is stopped. The Freeze Mode request cannot be granted until current transmission or reception processes have finished. Therefore the software can poll the FRZ_ACK bit to know when FLEXCAN has actually entered Freeze Mode. If Freeze Mode request is negated, then this bit is negated once the FLEXCAN prescaler is running again. If Freeze Mode is requested while FLEXCAN is in any of the low power modes, then the FRZ_ACK bit will only be set when the low power mode is exited. See Freeze Mode for more information 1 FLEXCAN in Freeze Mode, prescaler stopped 0 FLEXCAN not in Freeze Mode, prescaler running
23 SUPV	This bit configures some of the FLEXCAN registers to be either in Supervisor or User Mode. Reset value of this bit is '1', so the affected registers start with Supervisor access allowance only. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 FlexCAN is in Supervisor Mode. Affected registers allow only Supervisor access. Unrestricted access behaves as though the access was done to an unimplemented register location 0 FlexCAN is in User Mode. Affected registers allow both Supervisor and Unrestricted accesses
22 SLF_WAK	This bit enables the Self Wake Up feature when FLEXCAN is in Stop Mode. If this bit had been asserted by the time FLEXCAN entered Stop Mode, then FLEXCAN will look for a recessive to dominant transition on the bus during these modes. If a transition from recessive to dominant is detected during Stop Mode, then FLEXCAN generates, if enabled to do so, a Wake Up interrupt to the ARM so that it can resume the clocks globally and FlexCAN can request to resume the clocks. This bit can not be written while the module is in Stop Mode. 1 FLEXCAN Self Wake Up feature is enabled 0 FLEXCAN Self Wake Up feature is disabled
21 WRN_EN	When asserted, this bit enables the generation of the TWRN_INT and RWRN_INT flags in the Error and Status Register. If WRN_EN is negated, the TWRN_INT and RWRN_INT flags will always be zero, independent of the values of the error counters, and no warning interrupt will ever be generated.This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 TWRN_INT and RWRN_INT bits are set when the respective error counter transition from <96 to ≥ 96. 0 TWRN_INT and RWRN_INT bits are zero, independent of the values in the error counters.
20 LPM_ACK	This read-only bit indicates that FLEXCAN is either in Disable Mode or Stop Mode. Either of these low power modes can not be entered until all current transmission or reception processes have finished, so the ARM can poll the LPM_ACK bit to know when FLEXCAN has actually entered low power mode. See Module Disable Mode, and Stop Mode for more information 1 FLEXCAN is either in Disable Mode, or Stop mode 0 FLEXCAN not in any of the low power modes
19 WAK_SRC	This bit defines whether the integrated low-pass filter is applied to protect the FLEXCAN_RX input from spurious wake up. See Stop Mode for more information. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 FLEXCAN uses the filtered FLEXCAN_RX input to detect recessive to dominant edges on the CAN bus 0 FLEXCAN uses the unfiltered FLEXCAN_RX input to detect recessive to dominant edges on the CAN bus.
18 -	This field is reserved. Reserved

Table continues on the next page...

FLEXCANx_MCR field descriptions (continued)

Field	Description
17 SRX_DIS	This bit defines whether FlexCAN is allowed to receive frames transmitted by itself. If this bit is asserted, frames transmitted by the module will not be stored in any MB, regardless if the MB is programmed with an ID that matches the transmitted frame, and no interrupt flag or interrupt signal will be generated due to the frame reception. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Self reception disabled 0 Self reception enabled
16 IRMQ	This bit indicates whether Rx matching process will be based either on individual masking and queue or on masking scheme with RXMGMASK, RX14MASK and RX15MASK, RXFGMASK. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Individual Rx masking and queue feature are enabled. 0 Individual Rx masking and queue feature are disabled.For backward compatibility, the reading of C/S word locks the MB even if it is EMPTY.
15–14 -	This field is reserved. Reserved
13 LPRIO_EN	This bit is provided for backwards compatibility reasons. It controls whether the local priority feature is enabled or not. It is used to extend the ID used during the arbitration process. With this extended ID concept, the arbitration process is done based on the full 32-bit word, but the actual transmitted ID still has 11-bit for standard frames and 29-bit for extended frames.This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Local Priority enabled 0 Local Priority disabled
12 AEN	This bit is supplied for backwards compatibility reasons. When asserted, it enables the Tx abort feature. This feature guarantees a safe procedure for aborting a pending transmission, so that no frame is sent in the CAN bus without notification. This bit can only be written in Freeze mode as it is blocked by hardware in other modes.Write Abort code into Rx Mailboxes can cause unpredictable results when the MCR[AEN] is asserted. 1 Abort enabled 0 Abort disabled
11–10 -	This field is reserved. Reserved
9–8 IDAM	This 2-bit field identifies the format of the elements of the Rx FIFO filter table, as shown below. Note that all elements of the table are configured at the same time by this field (they are all the same format). See Rx FIFO Structure. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 00 Format A One full ID (standard or extended) per ID filter Table element. 01 Format B Two full standard IDs or two partial 14-bit extended IDs per ID filter Table element. 10 Format C Four partial 8-bit IDs (standard or extended) per ID filter Table element. 11 Format D All frames rejected.
7 -	This field is reserved. Reserved
MAXMB	This 7-bit field defines the number of the last Message Buffers that will take part in the matching and arbitration processes. The reset value (0x0F) is equivalent to 16 MB configuration. This field can only be written in Freeze Mode as it is blocked by hardware in other modes Number of the last MB = MAXMB.

Table continues on the next page...

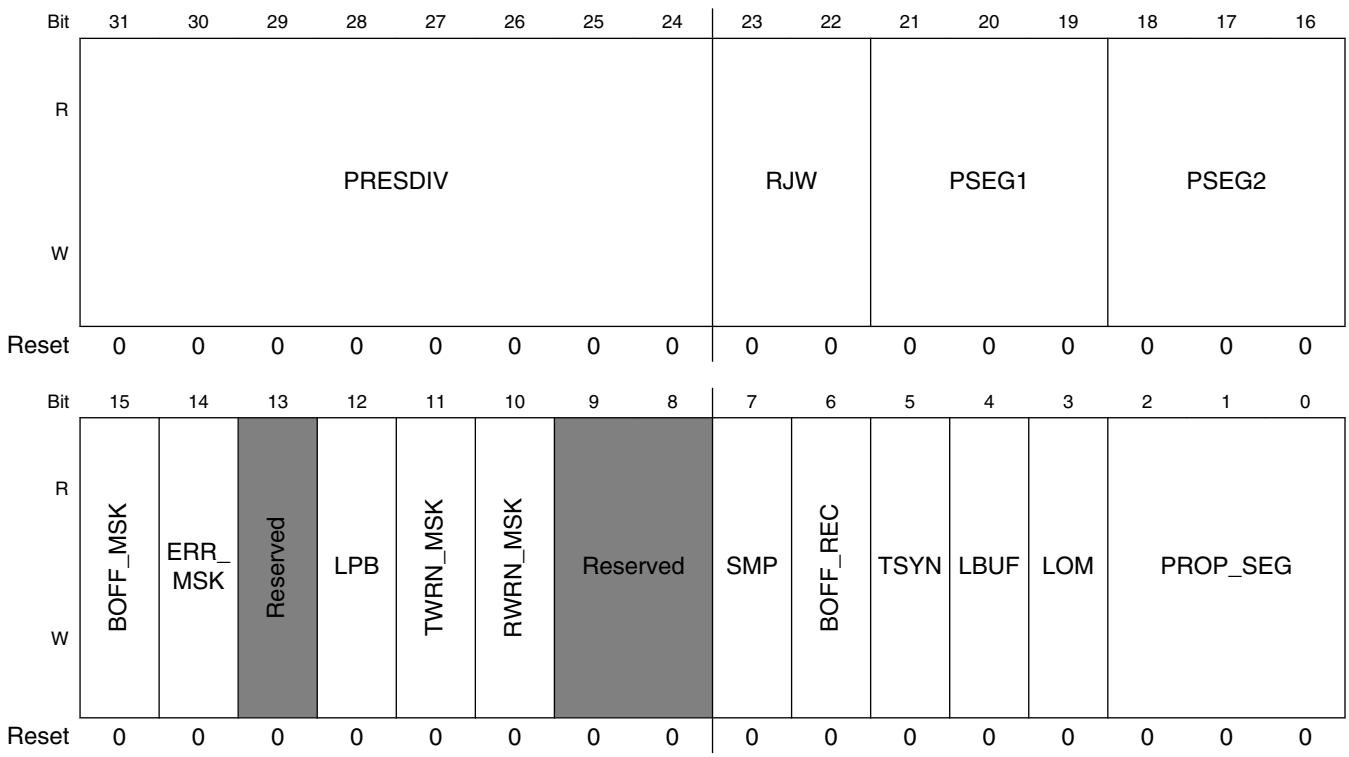
FLEXCANx_MCR field descriptions (continued)

Field	Description
	NOTE: Additionally, the value of MAXMB must encompass the FIFO size defined by CTRL2[RFFN] MAXMB also impacts the definition of the minimum number of peripheral clocks per CAN bit as described in Table 26-18 (see Arbitration and Matching Timing).

26.8.2 Control 1 Register (FLEXCANx_CTRL1)

This register is defined for specific FLEXCAN control features related to the CAN bus, such as bit-rate, programmable sampling point within an Rx bit, Loop Back Mode, Listen Only Mode, Bus Off recovery behavior and interrupt enabling (Bus-Off, Error, Warning). It also determines the Division Factor for the clock prescaler.

Address: Base address + 4h offset



FLEXCANx_CTRL1 field descriptions

Field	Description
31–24 PRESDIV	This 8-bit field defines the ratio between the PE clock frequency and the Serial Clock (Sclock) frequency. The Sclock period defines the time quantum of the CAN protocol. For the reset value, the Sclock frequency is equal to the PE clock frequency. The Maximum value of this register is 0xFF, that gives a minimum Sclock frequency equal to the PE clock frequency divided by 256.For more information refer to Protocol Timing. This field can only be written in Freeze mode as it is blocked by hardware in other modes.

Table continues on the next page...

FLEXCANx_CTRL1 field descriptions (continued)

Field	Description
	Sclock frequency = CPI clock frequency / (PRESDIV+1)
23–22 RJW	This 2-bit field defines the maximum number of time quanta[1] that a bit time can be changed by one re-synchronization. The valid programmable values are 0-3. This field can only be written in Freeze mode as it is blocked by hardware in other modes Resync Jump Width = RJW + 1.
21–19 PSEG1	This 3-bit field defines the length of Phase Buffer Segment 1 in the bit time. The valid programmable values are 0-7. This field can only be written in Freeze mode as it is blocked by hardware in other modes Phase Buffer Segment 1 = (PSEG1 + 1) x Time-Quanta.
18–16 PSEG2	This 3-bit field defines the length of Phase Buffer Segment 2 in the bit time. The valid programmable values are 1-7. This field can only be written in Freeze mode as it is blocked by hardware in other modes Phase Buffer Segment 2 = (PSEG2 + 1) x Time-Quanta.
15 BOFF_MSK	This bit provides a mask for the Bus Off Interrupt. 1 Bus Off interrupt enabled 0 Bus Off interrupt disabled
14 ERR_MSK	This bit provides a mask for the Error Interrupt. 1 Error interrupt enabled 0 Error interrupt disabled
13 -	This field is reserved. Reserved
12 LPB	This bit configures FlexCAN to operate in Loop-Back Mode. In this mode, FlexCAN performs an internal loop back that can be used for self test operation. The bit stream output of the transmitter is fed back internally to the receiver input. The FLEXCAN_RX input pin is ignored and the FLEXCAN_TX output goes to the recessive state (logic '1'). FlexCAN behaves as it normally does when transmitting, and treats its own transmitted message as a message received from a remote node. In this mode, FlexCAN ignores the bit sent during the ACK slot in the CAN frame acknowledge field, generating an internal acknowledge bit to ensure proper reception of its own message. Both transmit and receive interrupts are generated. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Loop Back enabled 0 Loop Back disabled
11 TWRN_MSK	This bit provides a mask for the Tx Warning Interrupt associated with the TWRN_INT flag in the Error and Status Register. This bit is read as zero when MCR[WRN_EN] bit is negated. This bit can only be written if MCR[WRN_EN] bit is asserted. 1 Tx Warning Interrupt enabled 0 Tx Warning Interrupt disabled
10 RWRN_MSK	This bit provides a mask for the Rx Warning Interrupt associated with the RWRN_INT flag in the Error and Status Register. This bit is read as zero when MCR[WRN_EN] bit is negated. This bit can only be written if MCR[WRN_EN] bit is asserted. 1 Rx Warning Interrupt enabled 0 Rx Warning Interrupt disabled
9–8 -	This field is reserved. Reserved
7 SMP	This bit defines the sampling mode of CAN bits at the FLEXCAN_RX. This bit can only be written in Freeze mode as it is blocked by hardware in other modes.

Table continues on the next page...

FLEXCANx_CTRL1 field descriptions (continued)

Field	Description
	1 Three samples are used to determine the value of the received bit: the regular one (sample point) and 2 preceding samples, a majority rule is used 0 Just one sample is used to determine the bit value
6 BOFF_REC	This bit defines how FLEXCAN recovers from Bus Off state. If this bit is negated, automatic recovering from Bus Off state occurs according to the CAN Specification 2.0B. If the bit is asserted, automatic recovering from Bus Off is disabled and the module remains in Bus Off state until the bit is negated by the user. If the negation occurs before 128 sequences of 11 recessive bits are detected on the CAN bus, then Bus Off recovery happens as if the BOFF_REC bit had never been asserted. If the negation occurs after 128 sequences of 11 recessive bits occurred, then FLEXCAN will re-synchronize to the bus by waiting for 11 recessive bits before joining the bus. After negation, the BOFF_REC bit can be re-asserted again during Bus Off, but it will only be effective the next time the module enters Bus Off. If BOFF_REC was negated when the module entered Bus Off, asserting it during Bus Off will not be effective for the current Bus Off recovery. 1 Automatic recovering from Bus Off state disabled 0 Automatic recovering from Bus Off state enabled, according to CAN Spec 2.0 part B
5 TSYN	This bit enables a mechanism that resets the free-running timer each time a message is received in Message Buffer 0. This feature provides means to synchronize multiple FLEXCAN stations with a special "SYNC" message (i.e., global network time). If the RFEN bit in MCR is set (FIFO enabled), the first available Mailbox, according to CTRL2[RFFN] setting, is used for timer synchronization instead of MB0.This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Timer Sync feature enabled 0 Timer Sync feature disabled
4 LBUF	This bit defines the ordering mechanism for Message Buffer transmission. When asserted, the LPRIO_EN bit does not affect the priority arbitration.This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Lowest number buffer is transmitted first 0 Buffer with highest priority is transmitted first
3 LOM	This bit configures FLEXCAN to operate in Listen Only Mode. In this mode, transmission is disabled, all error counters are frozen and the module operates in a CAN Error Passive mode . Only messages acknowledged by another CAN station will be received. If FLEXCAN detects a message that has not been acknowledged, it will flag a BIT0 error (without changing the REC), as if it was trying to acknowledge the message. Listen-Only Mode acknowledgement can be obtained by the state of ESR1[FLT_CONF] field which is Passive Error when Listen-Only Mode is entered. There can be some delay between the Listen-Only Mode request and acknowledge. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 FLEXCAN module operates in Listen Only Mode 0 Listen Only Mode is deactivated
PROP_SEG	This 3-bit field defines the length of the Propagation Segment in the bit time. The valid programmable values are 0-7. This field can only be written in Freeze mode as it is blocked by hardware in other modes Propagation Segment Time = (PROPSEG + 1) * Time-Quanta. Time-Quantum = one Sclock period.

1. One time quantum is equal to the Sclock period.

26.8.3 Free Running Timer Register (FLEXCANx_TIMER)

This register represents a 16-bit free running counter that can be read and written by the ARM. The timer starts from $0000 after Reset, counts linearly to $FFFF, and wraps around.

The timer is clocked by the FLEXCAN bit-clock (which defines the baud rate on the CAN bus). During a message transmission/reception, it increments by one for each bit that is received or transmitted. When there is no message on the bus, it counts using the previously programmed baud rate. During Freeze Mode, disable, and stop mode, the timer is not incremented.

The timer value is captured at the beginning of the identifier field of any frame on the CAN bus. This captured value is written into the Time Stamp entry in a message buffer after a successful reception or transmission of a message.

If bit CTRL1[TSYN] is asserted the Timer is reset whenever a message is received in the first available Mailbox, according to CTRL2[RFFN] setting.

ARM can write to this register anytime. However, if the write occurs at the same time that the Timer is being reset by a reception in the first Mailbox, then the write value is discarded.

Reading this register affects the Mailbox Unlocking procedure. For additional details, refer to Message Buffer Lock Mechanism.

Address: Base address + 8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved																TIMER															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_TIMER field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
TIMER	TIMER

26.8.4 Rx Mailboxes Global Mask Register (FLEXCANx_RXMGMASK)

RXMGMASK is provided for legacy support. Asserting the MCR[IRMQ] bit causes the RXMGMASK Register to have no effect on the module operation.

RXMGMASK is used to mask the filter fields of all Rx MBs, excluding MBs 14-15, which have individual mask registers.

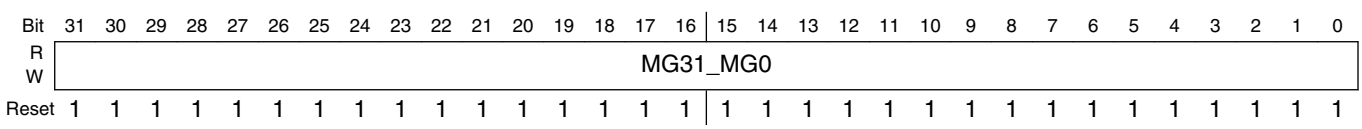
This register can only be written in Freeze mode as it is blocked by hardware in other modes.

Table 26-49. Rx Mailboxes Global Mask usage

SMB[RTR][1]	CTRL2[RRS]	CTRL2[EACEN]	Mailbox filter fields			
			MB[RTR]	MB[IDE]	MB[ID]	reserved
0	-	0	- Note[2]	- Note[3]	MG[28:0]	MG[31:29]
0	-	1	MG[31]	MG[30]	MG[28:0]	MG[29]
1	0	-	-	-	-	MG[31:0]
1	1	0	-	-	MG[28:0]	MG[31:29]
1	1	1	MG[31]	MG[30]	MG[28:0]	MG[29]

1. RTR bit of the Incoming Frame. It is saved into an auxiliary MB called Rx Serial Message Buffer (Rx SMB).
2. If CTRL2[EACEN] bit is negated the RTR bit of Mailbox is never compared with the RTR bit of the Incoming Frame (Rx SMB[RTR]).
3. If CTRL2[EACEN] bit is negated the IDE bit of Mailbox is always compared with the IDE bit of the Incoming Frame (Rx SMB[IDE]).

Address: Base address + 10h offset



FLEXCANx_RXMGMASK field descriptions

Field	Description
MG31_MG0	These bits mask the Mailbox filter bits as shown in the figure above. Note that the alignment with the ID word of the Mailbox is not perfect as the two most significant MG bits affect the fields RTR and IDE which are located in the Control and Status word of the Mailbox. Rx Mailboxes Global Mask Register (FLEXCAN_RXMGMASK) shows in detail which MG bits mask each Mailbox filter field. 1 The corresponding bit in the filter is checked against the one received 0 the corresponding bit in the filter is "don't care"

26.8.5 Rx Buffer 14 Mask Register (FLEXCANx_RX14MASK)

RX14MASK is provided for legacy support, asserting the MCR[IRMQ] bit causes the RX14MASK to have no effect on the module operation.

RX14MASK is used to mask the filter fields of Message Buffer 14.

This register can only be programmed while the module is in Freeze Mode as it is blocked by hardware in other modes.

Address: Base address + 14h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RX14M31_RX14M0																															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

FLEXCANx_RX14MASK field descriptions

Field	Description
RX14M31_ RX14M0	These bits mask Mailbox 14 filter bits in the same fashion as RXMGMASK masks other Mailboxes filters (see Rx Mailboxes Global Mask Register (FLEXCAN_RXMGMASK)) 1 The corresponding bit in the filter is checked 0 the corresponding bit in the filter is "don't care"

26.8.6 Rx Buffer 15 Mask Register (FLEXCANx_RX15MASK)

RX15MASK is provided for legacy support, asserting the MCR[IRMQ] bit causes the RX15MASK Register to have no effect on the module operation.

RX15MASK is used to mask the filter fields of Message Buffer 15.

This register can only be programmed while the module is in Freeze Mode as it is blocked by hardware in other modes.

Address: Base address + 18h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RX15M31_RX15M0																															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

FLEXCANx_RX15MASK field descriptions

Field	Description
RX15M31_ RX15M0	These bits mask Mailbox 15 filter bits in the same fashion as RXMGMASK masks other Mailboxes filters (see Rx Mailboxes Global Mask Register (FLEXCAN_RXMGMASK)). 1 The corresponding bit in the filter is checked 0 the corresponding bit in the filter is "don't care"

26.8.7 Error Counter Register (FLEXCANx_ECR)

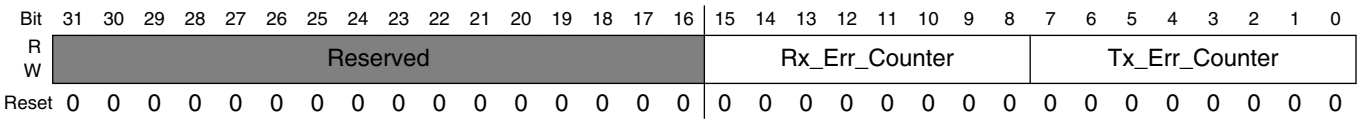
This register has 2 8-bit fields reflecting the value of two FLEXCAN error counters: Transmit Error Counter (Tx_Err_Counter field) and Receive Error Counter (Rx_Err_Counter field). The rules for increasing and decreasing these counters are described in the CAN protocol and are completely implemented in the FLEXCAN module. Both counters are read only except in Freeze Mode, where they can be written by the ARM.

FLEXCAN responds to any bus state as described in the protocol, e.g. transmit 'Error Active' or 'Error Passive' flag, delay its transmission start time ('Error Passive') and avoid any influence on the bus when in 'Bus Off' state. The following are the basic rules for FLEXCAN bus state transitions.

- If the value of Tx_Err_Counter or Rx_Err_Counter increases to be greater than or equal to 128, the FLT_CONF field in the Error and Status Register is updated to reflect 'Error Passive' state.
- If the FLEXCAN state is 'Error Passive', and either Tx_Err_Counter or Rx_Err_Counter decrements to a value less than or equal to 127 while the other already satisfies this condition, the FLT_CONF field in the Error and Status Register is updated to reflect 'Error Active' state.
- If the value of Tx_Err_Counter increases to be greater than 255, the FLT_CONF field in the Error and Status Register is updated to reflect 'Bus Off' state, and an interrupt may be issued. The value of Tx_Err_Counter is then reset to zero.
- If FLEXCAN is in 'Bus Off' state, then Tx_Err_Counter is cascaded together with another internal counter to count the 128th occurrences of 11 consecutive recessive bits on the bus. Hence, Tx_Err_Counter is reset to zero and counts in a manner where the internal counter counts 11 such bits and then wraps around while incrementing the Tx_Err_Counter. When Tx_Err_Counter reaches the value of 128, the FLT_CONF field in the Error and Status Register is updated to be 'Error Active' and both error counters are reset to zero. At any instance of dominant bit following a stream of less than 11 consecutive recessive bits, the internal counter resets itself to zero without affecting the Tx_Err_Counter value.
- If during system start-up, only one node is operating, then its Tx_Err_Counter increases in each message it is trying to transmit, as a result of acknowledge errors (indicated by the ACK_ERR bit in the Error and Status Register). After the transition to 'Error Passive' state, the Tx_Err_Counter does not increment anymore by acknowledge errors. Therefore the device never goes to the 'Bus Off' state.
- If the Rx_Err_Counter increases to a value greater than 127, it is not incremented further, even if more errors are detected while being a receiver. At the next

successful message reception, the counter is set to a value between 119 and 127 to resume to 'Error Active' state.

Address: Base address + 1Ch offset



FLEXCANx_ECR field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
15–8 Rx_Err_Counter	Rx_Err_Counter
Tx_Err_Counter	Tx_Err_Counter

26.8.8 Error and Status 1 Register (FLEXCANx_ESR1)

This register reflects various error conditions, some general status of the device and it is the source of four interrupts to the ARM.

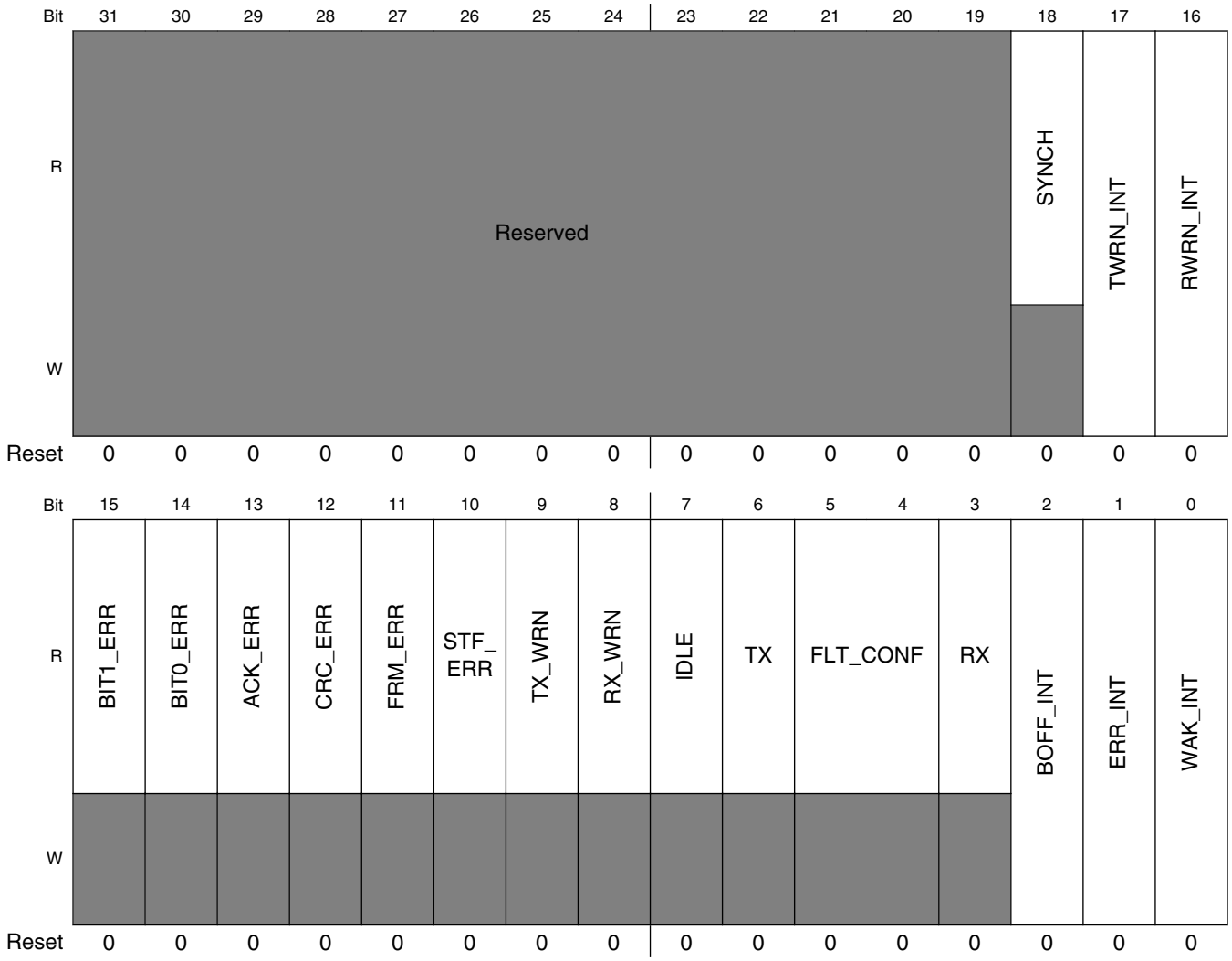
The ARM read action clears bits 15-10, therefore the reported *error conditions*(bits 15-10) are those that occurred since the last time the ARM read this register. Bits 9-3 are status bits.

Some bits in this register are read-only and some are not.

Table 26-54. FlexCAN State

SYNCH	IDLE	TX	RX	FlexCAN state
0	0	0	0	Not synchronized to CAN bus
1	1	x	x	Idle
1	0	1	0	Transmitting
1	0	0	1	Receiving
other combinations				Reserved

Address: Base address + 20h offset



FLEXCANx_ESR1 field descriptions

Field	Description
31–19 -	This field is reserved. Reserved
18 SYNCH	This read-only flag indicates whether the FlexCAN is synchronized to the CAN bus and able to participate in the communication process. It is set and cleared by the FlexCAN. Refer to Table 26-29 1 FlexCAN is synchronized to the CAN bus 0 FlexCAN is not synchronized to the CAN bus
17 TWRN_INT	If the WRN_EN bit in MCR is asserted, the TWRN_INT bit is set when the TX_WRN flag transition from '0' to '1', meaning that the Tx error counter reached 96. If the corresponding mask bit in the Control Register (TWRN_MSK) is set, an interrupt is generated to the ARM. This bit is cleared by writing it to '1'. When WRN_EN is negated, this flag is masked. ARM must clear this flag before disabling the bit. Otherwise it will be set when the WRN_EN is set again. Writing '0' has no effect. This flag is not generated during "Bus Off" state. This bit is not updated during Freeze mode. 1 The Tx error counter transition from < 96 to >= 96 0 No such occurrence

Table continues on the next page...

FLEXCANx_ESR1 field descriptions (continued)

Field	Description
16 RWRN_INT	If the WRN_EN bit in MCR is asserted, the RWRN_INT bit is set when the RX_WRN flag transition from '0' to '1', meaning that the Rx error counters reached 96. If the corresponding mask bit in the Control Register (RWRN_MSK) is set, an interrupt is generated to the ARM. This bit is cleared by writing it to '1'. When WRN_EN is negated, this flag is masked. ARM must clear this flag before disabling the bit. Otherwise it will be set when the WRN_EN is set again. Writing '0' has no effect. This bit is not updated during Freeze mode. 1 The Rx error counter transition from < 96 to >= 96 0 No such occurrence
15 BIT1_ERR	This bit indicates when an inconsistency occurs between the transmitted and the received bit in a message. This bit is not set by a transmitter in case of arbitration field or ACK slot, or in case of a node sending a passive error flag that detects dominant bits. 1 At least one bit sent as recessive is received as dominant 0 No such occurrence
14 BIT0_ERR	This bit indicates when an inconsistency occurs between the transmitted and the received bit in a message. 1 At least one bit sent as dominant is received as recessive 0 No such occurrence
13 ACK_ERR	This bit indicates that an Acknowledge Error has been detected by the transmitter node, i.e., a dominant bit has not been detected during the ACK SLOT. 1 An ACK error occurred since last read of this register 0 No such occurrence
12 CRC_ERR	This bit indicates that a CRC Error has been detected by the receiver node, i.e., the calculated CRC is different from the received. 1 A CRC error occurred since last read of this register. 0 No such occurrence
11 FRM_ERR	This bit indicates that a Form Error has been detected by the receiver node, i.e., a fixed-form bit field contains at least one illegal bit. 1 A Form Error occurred since last read of this register 0 No such occurrence
10 STF_ERR	This bit indicates that a Stuffing Error has been detected. 1 A Stuffing Error occurred since last read of this register. 0 No such occurrence.
9 TX_WRN	This bit indicates when repetitive errors are occurring during message transmission. 1 TX_Err_Counter ≥ 96 0 No such occurrence
8 RX_WRN	This bit indicates when repetitive errors are occurring during message reception. 1 Rx_Err_Counter ≥ 96 0 No such occurrence
7 IDLE	This bit indicates when CAN bus is in IDLE state.Refer to Table 26-29.

Table continues on the next page...

FLEXCANx_ESR1 field descriptions (continued)

Field	Description
	1 CAN bus is now IDLE 0 No such occurrence
6 TX	This bit indicates if FLEXCAN is transmitting a message.Refer to Table 26-29. 1 FLEXCAN is transmitting a message 0 FLEXCAN is receiving a message
5–4 FLT_CONF	If the LOM bit in the Control Register is asserted, after some delay that depends on the CAN bit timing the FLT_CONF field will indicate "Error Passive". The very same delay affects the way how FLT_CONF reflects an update to ECR register by the ARM. It may be necessary up to one CAN bit time to get them coherent again. Since the Control Register is not affected by soft reset, the FLT_CONF field will not be affected by soft reset if the LOM bit is asserted. This 2-bit field indicates the Confinement State of the FLEXCAN module, as shown in below: 00 Error Active 01 Error Passive 1x Bus off
3 RX	This bit indicates if FlexCAN is receiving a message. Refer to Table 26-29. 1 FLEXCAN is transmitting a message 0 FLEXCAN is receiving a message
2 BOFF_INT	This bit is set when FLEXCAN enters 'Bus Off' state. If the corresponding mask bit in the Control Register (BOFF_MSK) is set, an interrupt is generated to the ARM. This bit is cleared by writing it to '1'. Writing '0' has no effect. 1 FLEXCAN module entered 'Bus Off' state 0 No such occurrence
1 ERR_INT	This bit indicates that at least one of the Error Bits (bits 15-10) is set. If the corresponding mask bit in the Control Register (ERR_MSK) is set, an interrupt is generated to the ARM. This bit is cleared by writing it to '1'.Writing '0' has no effect. 1 Indicates setting of any Error Bit in the Error and Status Register 0 No such occurrence
0 WAK_INT	When FLEXCAN is Stop Mode and a recessive to dominant transition is detected on the CAN bus and if the WAK_MSK bit in the MCR Register is set, an interrupt is generated to the ARM. This bit is cleared by writing it to '1'. When SLF_WAK is negated, this flag is masked. ARM must clear this flag before disabling the bit. Otherwise it will be set when the SLF_WAK is set again. Writing '0' has no effect 1 Indicates a recessive to dominant transition received on the CAN bus when the FLEXCAN module is in Stop Mode 0 No such occurrence

26.8.9 Interrupt Masks 2 Register (FLEXCANx_IMASK2)

This register allows any number of a range of 32 Message Buffer Interrupts to be enabled or disabled. It contains one interrupt mask bit per buffer, enabling the ARM to determine which buffer generates an interrupt after a successful transmission or reception (i.e. when the corresponding IFLAG2 bit is set).

Address: Base address + 24h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	BUF63M_BUF32M																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_IMASK2 field descriptions

Field	Description
BUF63M_ BUF32M	Each bit enables or disables the respective FLEXCAN Message Buffer (MB32 to MB63) Interrupt. Setting or clearing a bit in the IMASK2 Register can assert or negate an interrupt request, if the corresponding IFLAG2 bit is set. 1 The corresponding buffer Interrupt is enabled 0 The corresponding buffer Interrupt is disabled

26.8.10 Interrupt Masks 1 Register (FLEXCANx_IMASK1)

This register allows to enable or disable any number of a range of 32 Message Buffer Interrupts. It contains one interrupt mask bit per buffer, enabling the ARM to determine which buffer generates an interrupt after a successful transmission or reception (i.e., when the corresponding IFLAG1 bit is set).

Address: Base address + 28h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	BUF31M_BUF0M																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_IMASK1 field descriptions

Field	Description
BUF31M_BUF0M	Each bit enables or disables the respective FLEXCAN Message Buffer (MB0 to MB31) Interrupt. Setting or clearing a bit in the IMASK1 Register can assert or negate an interrupt request, if the corresponding IFLAG1 bit is set

FLEXCANx_IMASK1 field descriptions (continued)

Field	Description
	1 The corresponding buffer Interrupt is enabled 0 The corresponding buffer Interrupt is disabled

26.8.11 Interrupt Flags 2 Register (FLEXCANx_IFLAG2)

This register defines the flags for 32 Message Buffer interrupts. It contains one interrupt flag bit per buffer. Each successful transmission or reception sets the corresponding IFLAG2 bit. If the corresponding IMASK2 bit is set, an interrupt will be generated. The interrupt flag must be cleared by writing it to '1'. Writing '0' has no effect.Before updating MCR[MAXMB] field, ARM must treat the IFLAG2 bits which MB value is greater than the MCR[MAXMB] to be updated, otherwise they will keep set and be inconsistent with the amount of MBs available.

Address: Base address + 2Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	BUF63I_BUF32I																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_IFLAG2 field descriptions

Field	Description
BUF63I_BUF32I	Each bit flags the respective FLEXCAN Message Buffer (MB32 to MB63) interrupt. 1 The corresponding buffer has successfully completed transmission or reception 0 No such occurrence

26.8.12 Interrupt Flags 1 Register (FLEXCANx_IFLAG1)

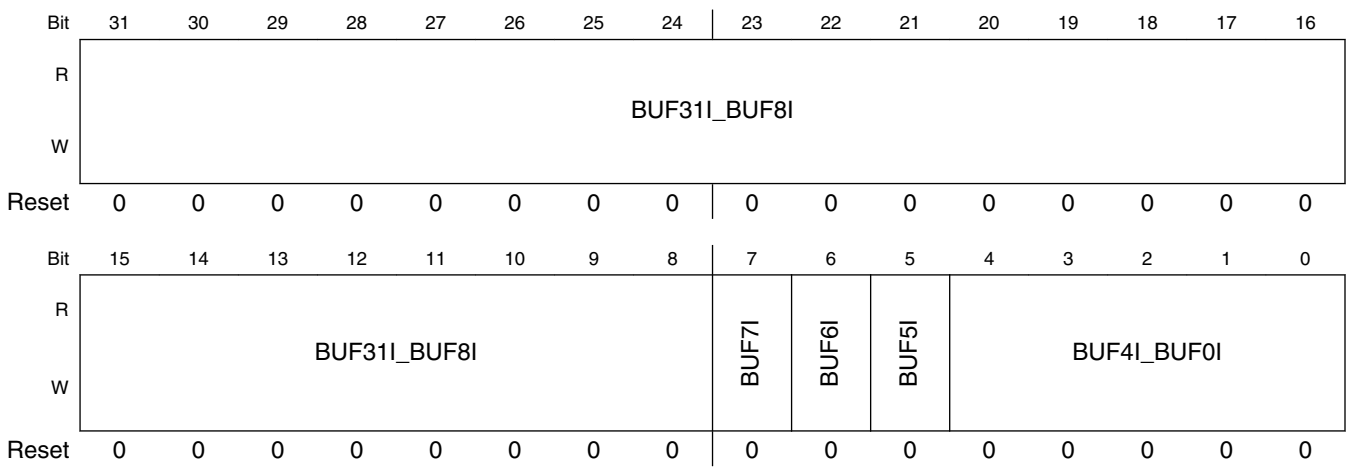
This register defines the flags for 32 Message Buffer interrupts and FIFO interrupts. It contains one interrupt flag bit per buffer. Each successful transmission or reception sets the corresponding IFLAG1 bit. If the corresponding IMASK1 bit is set, an interrupt will be generated. The Interrupt flag must be cleared by writing it to '1'. Writing '0' has no effect.

When the RFEN bit in the MCR is set (Rx FIFO enabled), the function of the 8 least significant interrupt flags (BUF7I - BUF0I) is changed to support the FIFO operation. BUF7I, BUF6I and BUF5I indicate operating conditions of the FIFO, while BUF4I to BUF0I are not used. Before enabling the RFEN, ARM must service the IFLAGS asserted

in the Rx FIFO region (see Rx FIFO). Otherwise, these IFLAGS will mistakenly show the related MBs now belonging to FIFO as having contents to be serviced. When the RFEN is negated, the FIFO flags must be cleared. The same care must be taken when a RFFN value is selected extending Rx FIFO filters beyond MB7 (see Control 2 Register (FLEXCAN_CTRL2)). For example, when RFFN is 0x8, the MB0-23 range is occupied by Rx FIFO filters and related IFLAGS must be cleared.

Before updating MCR[MAXMB] field, ARM must service the IFLAG1 which MB value is greater than the MCR[MAXMB] to be updated, otherwise they will keep set and be inconsistent with the amount of MBs available.

Address: Base address + 30h offset



FLEXCANx_IFLAG1 field descriptions

Field	Description
31–8 BUF31I_BUF8I	Each bit flags the respective FLEXCAN Message Buffer (MB8 to MB31) interrupt. 1 The corresponding MB has successfully completed transmission or reception 0 No such occurrence
7 BUF7I	If the Rx FIFO is not enabled, this bit flags the interrupt for MB7. If the MCR[RFEN] bit is asserted, this flag indicates that a message was lost because Rx FIFO is full. Note that the flag will not be asserted when the Rx FIFO is full and the message was captured by a Mailbox. This flag is cleared by the FlexCAN whenever the bit MCR[RFEN] is changed by ARM writes. 1 MB7 completed transmission/reception or FIFO overflow 0 No such occurrence
6 BUF6I	If the Rx FIFO is not enabled, this bit flags the interrupt for MB6. If the MCR[RFEN] bit is asserted, this flag indicates when the number of unread messages within the Rx FIFO is increased to 5 from 4 due to the reception of a new one, meaning that the Rx FIFO is almost full. Note that if the flag is cleared while the number of unread messages is greater than 4 it will not assert again until the number of unread messages within the Rx FIFO is decreased to equal or less than 4. This flag is cleared by the FlexCAN whenever the bit MCR[RFEN] is changed by ARM writes.

Table continues on the next page...

FLEXCANx_IFLAG1 field descriptions (continued)

Field	Description
	1 MB6 completed transmission/reception or FIFO almost full 0 No such occurrence
5 BUF5I	If the Rx FIFO is not enabled, this bit flags the interrupt for MB5. If the Rx FIFO is enabled, this flag indicates that at least one frame is available to be read from the Rx FIFO. This flag is cleared by the FlexCAN whenever the bit MCR[RFEN] is changed by ARM writes. 1 MB5 completed transmission/reception or frames available in the FIFO 0 No such occurrence
BUF4I_BUF0I	If the Rx FIFO is not enabled, these bits flag the interrupts for MB0 to MB4. If the Rx FIFO is enabled, these flags are not used and must be considered as reserved locations. These flags are cleared by the FlexCAN whenever the bit MCR[RFEN] is changed by ARM writes. 1 Corresponding MB completed transmission/reception 0 No such occurrence

26.8.13 Control 2 Register (FLEXCANx_CTRL2)

This register contains control bits for CAN errors, FIFO features and mode selection.

Table 26-60. Rx FIFO Filters

RFFN[3:0]	Number of Rx FIFO filters	Message Buffers occupied by Rx FIFO and ID Filter Table	Remaining Available Mailboxes[1]	Rx FIFO ID Filter Table Elements Affected by Rx Individual Masks[2]	Rx FIFO ID Filter Table Elements Affected by Rx FIFO Global Mask[2]
0x0	8	MB 0-7	MB 8-63	Elements 0-7	none
0x1	16	MB 0-9	MB 10-63	Elements 0-9	Elements 10-15
0x2	24	MB 0-11	MB 12-63	Elements 0-11	Elements 12-23
0x3	32	MB 0-13	MB 14-63	Elements 0-13	Elements 14-31
0x4	40	MB 0-15	MB 16-63	Elements 0-15	Elements 16-39
0x5	48	MB 0-17	MB 18-63	Elements 0-17	Elements 18-47
0x6	56	MB 0-19	MB 20-63	Elements 0-19	Elements 20-55
0x7	64	MB 0-21	MB 22-63	Elements 0-21	Elements 22-63
0x8	72	MB 0-23	MB 24-63	Elements 0-23	Elements 24-71
0x9	80	MB 0-25	MB 26-63	Elements 0-25	Elements 26-79
0xA	88	MB 0-27	MB 28-63	Elements 0-27	Elements 28-87
0xB	96	MB 0-29	MB 30-63	Elements 0-29	Elements 30-95
0xC	104	MB 0-31	MB 32-63	Elements 0-31	Elements 32-103
0xD	112	MB 0-33	MB 34-63	Elements 0-31	Elements 32-111
0xE	120	MB 0-35	MB 36-63	Elements 0-31	Elements 32-119
0xF	128	MB 0-37	MB 38-63	Elements 0-31	Elements 32-127

1. The number of the last remaining available mailboxes is defined by the MCR[MAXMB] field.
2. If Rx Individual Mask Registers are not enabled then all Rx FIFO filters are affected by the Rx FIFO Global Mask.

Each group of eight filters occupies a memory space equivalent to two Message Buffers which means that the more filters are implemented the less Mailboxes will be available.

Considering that the Rx FIFO occupies the memory space originally reserved for MB0-5, RFFN should be programmed with a value corresponding to a number of filters not greater than the number of available memory words which can be calculated as follows:

$$(SETUP\_MB - 6) \times 4$$

where SETUP_MB is MAXMB.

The number of remaining Mailboxes available will be:

$$SETUP\_MB - 8 - (RFFN \times 2)$$

If the Number of Rx FIFO Filters programmed through RFFN exceeds the SETUP_MB value, the exceeding ones will not be functional.Unshaded regions in Table 26-36 indicate the valid combinations of MAXMB, RFEN and RFFN, shaded regions are not functional.

Table 26-61. Valid Combinations of MAXMB, RFEN and RFFN

RFFN	0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RFEN	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MAXMB																	
0 - 6																	
7 - 8																	
9 - 10																	
11 - 12																	
13 -14																	
15 - 16																	
17 -18																	
19 - 20																	
21 - 22																	

Table continues on the next page...

Table 26-61. Valid Combinations of MAXMB, RFEN and RFFN (continued)

RFFN	0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RFEN	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
MAXMB																	
23 - 24																	
25 - 26																	
27 - 28																	
29 -30																	
31 - 32																	
33 - 34																	
35 - 36																	
37 - 63																	

Address: Base address + 34h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R		Reserved		WRMFRZ	RFEN				TASD					MRP	RRS	EACEN
W	0															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_CTRL2 field descriptions

Field	Description
31 -	must be written as 0
30–29 -	This field is reserved. Reserved

Table continues on the next page...

FLEXCANx_CTRL2 field descriptions (continued)

Field	Description
28 WRMFRZ	Enable unrestricted write access to FlexCAN memory in Freeze mode. This bit can only be written in Freeze mode and has no effect out of Freeze mode. 1 Enable unrestricted write access to FlexCAN memory 0 Keep the write access restricted in some regions of FlexCAN memory
27–24 RFEN	This 4-bit field defines the number of Rx FIFO filters according to Table 26-35. The maximum selectable number of filters is determined by the ARM. This field can only be written in Freeze mode as it is blocked by hardware in other modes. RFFN defines a number of Message Buffers occupied by Rx FIFO and ID Filter (see Table 26-35) that **may not exceed** the number of available Mailboxes present in module, defined by MCR[MAXMB].Default RFFN value is 0x0, which leads to a total of 8 Rx FIFO filters, occupies the first 8 Message Buffers (MB 0-7) and makes available the next Message Buffers (MB 8-63) for Mailboxes. As a second example, when RFFN is set to 0xD, there will be 112 Rx FIFO filters, located in MB 0-33, and MB 34-63 are available for Mailboxes. Notice that, in this case, individual masks (RXIMR) will just cover Rx FIFO filters in 0-31 range, and filters 32-111 will use RXFGMASK. In case of reducing the number of last Message Buffers, MCR[MAXMB] (see Module Configuration Register (FLEXCAN_MCR)) can be adjusted by the application to minimum of 33, in order to give room to the Rx FIFO and its ID Filter Table defined by RFFN. On the contrary, if the application sets MCR[MAXMB] to 16, for instance, the maximum RFFN is limited to 0x4. RFFN also impacts the definition of the minimum number of peripheral clocks per CAN bit as described in Table 26-18 (see Arbitration and Matching Timing).
23–19 TASD	This 5-bit field indicates how many CAN bits the Tx arbitration process start point can be delayed from the first bit of CRC field on CAN bus. This field can only be written in Freeze mode as it is blocked by hardware in other modes. This field is useful to optimize the transmit performance based on factors such as: peripheral/serial clock ratio, CAN bit timing and number of MBs. The duration of an arbitration process, in terms of CAN bits, is directly proportional to the number of available MBs and CAN baud rate and inversely proportional to the peripheral clock frequency. The optimal arbitration timing is that in which the last MB is scanned right before the first bit of the Intermission field of a CAN frame. Therefore, if there are few MBs and the system/serial clock ratio is high and the CAN baud rate is low then the arbitration can be delayed and vice-versa. If TASD is 0 then the arbitration start is not delayed, thus ARM has less time to configure a Tx MB for the next arbitration, but more time is reserved for arbitration. In the other hand, if TASD is 24 then ARM can configure a Tx MB later and less time is reserved for arbitration. If too little time is reserved for arbitration the FlexCAN may be not able to find winner MBs in time to compete with other nodes for the CAN bus. If the arbitration ends too much time before the first bit of Intermission field then there is a chance that ARM reconfigure some Tx MBs and the winner MB is not the best to be transmitted. The reset value is different on various platforms, according to their peripheral clock frequency, number of MBs and target CAN baud rate. The optimal configuration for TASD can be calculated as:

Table continues on the next page...

FLEXCANx_CTRL2 field descriptions (continued)

Field	Description
	$$TASD = 25 - \left\{ \frac{f_{CANCLK} \times [MAXMB + 3 - (RFEN \times 8) - (RFEN \times RFFN \times 2)] \times 2}{f_{sys} \times [1 + (PSEG1 + 1) + (PSEG2 + 1)] \times (PRESDIV + 1)} \right\}$$ where: f_{CANCLK} is the Protocol Engine (PE) Clock in Hz; PE clock is derrived from CAN_CLK_ROOT in CCM. See Clock Root Generator f_{SYS} is the peripheral clock in Hz; MAXMB is the value in CTRL1[MAXMB] field; RFEN is the value in CTRL1[RFEN] bit; RFFN is the value in CTRL2[RFFN] field; PSEG1 is the value in CTRL1[PSEG1] field; PSEG2 is the value in CTRL1[PSEG2] field; PROPSEG is the value in CTRL1[PROPSEG] field; PRESDIV is the value in CTRL1[PRESDIV] field. Please refer to Arbitration process and Protocol Timing for more details.
18 MRP	If this bit is set the matching process starts from the Mailboxes and if no match occurs the matching continues on the Rx FIFO. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Matching starts from Mailboxes and continues on Rx FIFO 0 Matching starts from Rx FIFO and continues on Mailboxes
17 RRS	If this bit is asserted Remote Request Frame is submitted to a matching process and stored in the corresponding Message Buffer in the same fashion of a Data Frame. No automatic Remote Response Frame will be generated. If this bit is negated the Remote Request Frame is submitted to a matching process and an automatic Remote Response Frame is generated if a Message Buffer with CODE=0b1010 is found with the same ID. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Remote Request Frame is stored 0 Remote Response Frame is generated
16 EACEN	This bit controls the comparison of IDE and RTR bits within Rx Mailboxes filters with their corresponding bits in the incoming frame by the matching process. This bit does not affect matching for Rx FIFO. This bit can only be written in Freeze mode as it is blocked by hardware in other modes. 1 Enables the comparison of both Rx Mailbox filter's IDE and RTR bit with their corresponding bits within the incoming frame. Mask bits do apply. 0 Rx Mailbox filter's IDE bit is always compared and RTR is never compared despite mask bits.
-	This field is reserved.

Table continues on the next page...

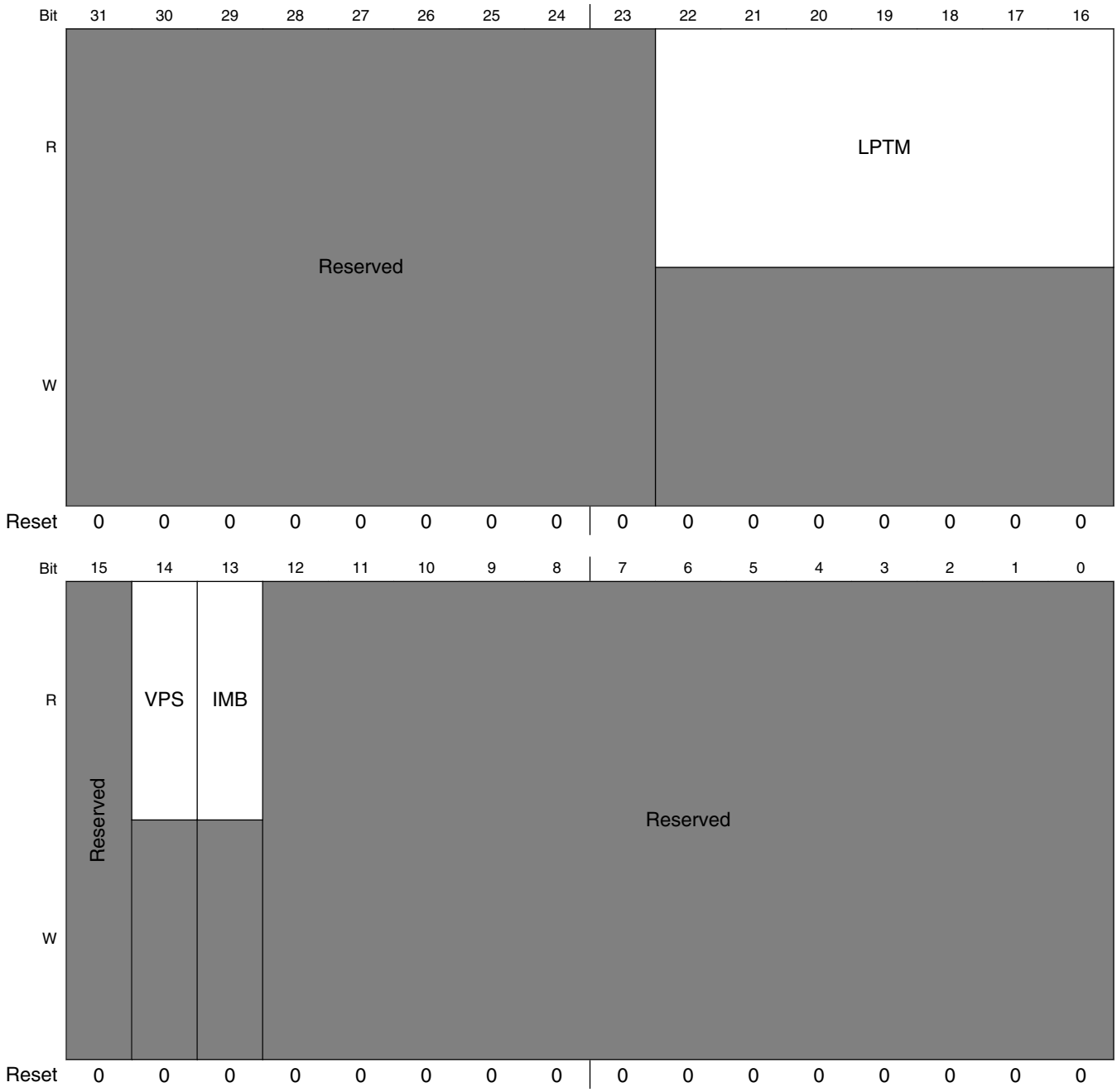
FLEXCANx_CTRL2 field descriptions (continued)

Field	Description
	Reserved

26.8.14 Error and Status 2 Register (FLEXCANx_ESR2)

This register reflects various interrupt flags and some general status.

Address: Base address + 38h offset



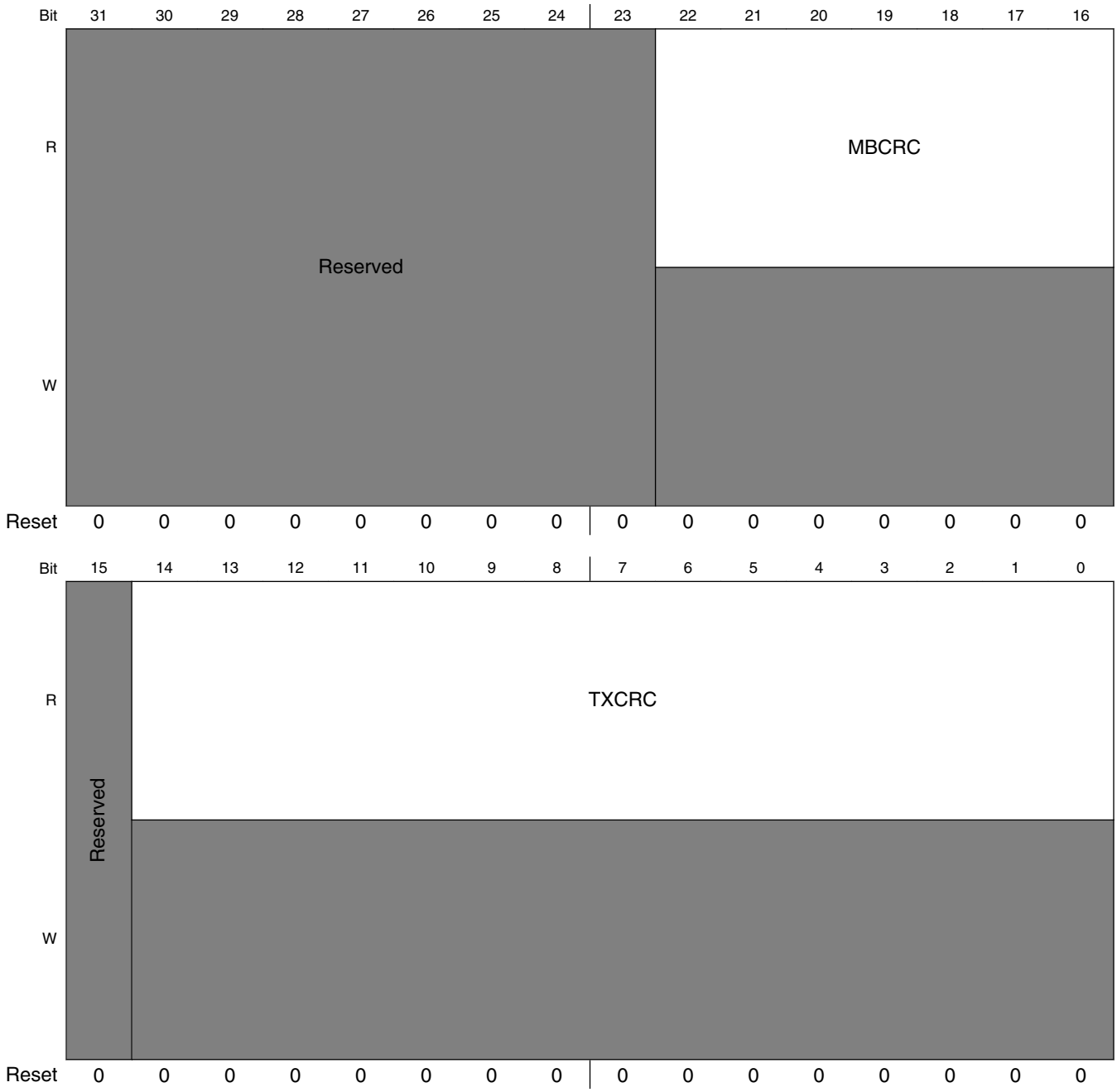
FLEXCANx_ESR2 field descriptions

Field	Description
31–23 -	This field is reserved. Reserved
22–16 LPTM	If ESR2[VPS] is asserted, his 7-bit field indicates the lowest number inactive Mailbox (refer to IMB bit description). If there is no inactive Mailbox then the Mailbox indicated depends on CTRL1[LBUF] bit value. If CTRL1[LBUF] bit is negated then the Mailbox indicated is the one which has the greatest arbitration value (see Highest Mailbox priority first). If CTRL1[LBUF] bit is asserted then the Mailbox indicated is the highest number active Tx Mailbox. If a Tx Mailbox is being transmitted it is not considered in LPTM calculation. If ESR2[IMB] is not asserted and a frame is transmitted successfully, LPTM is updated with its Mailbox number.
15 -	This field is reserved. Reserved
14 VPS	This bit indicates whether IMB and LPTM contents are currently valid or not. VPS is asserted upon every complete Tx arbitration process unless the ARM writes to Control and Status word of a Mailbox that has already been scanned (i.e. it is behind Tx Arbitration Pointer) during the Tx arbitration process. If there is no inactive Mailbox and only one Tx Mailbox which is being transmitted then VPS is not asserted. VPS is negated upon the start of every Tx arbitration process or upon a write to Control and Status word of any Mailbox.ESR2[VPS] is not affected by any ARM write into Control Status (C/S) of a MB which is blocked by abort mechanism. When MCR[AEN] is asserted, the abort code write in C/S of a MB that is been transmitted (pending abort), or any write attempt into a Tx MB with IFLAG set is blocked. 1 Contents of IMB and LPTM are valid 0 Contents of IMB and LPTM are invalid
13 IMB	If ESR2[VPS] is asserted, this bit indicates whether there is any inactive Mailbox (CODE field is either 0b1000 or 0b0000). This bit is asserted in the following cases: (1) During arbitration, if a LPTM is found and it is inactive. (2) If IMB is not asserted and a frame is transmitted successfully. (3) This bit is cleared in all start of arbitration (see Arbitration process). LPTM mechanism have the following behavior: if a MB is successfully transmitted and ESR2[IMB]=0 (no inactive Mailbox), then ESR2[VPS] and ESR2[IMB] are asserted and the index related to the MB just transmitted is loaded into ESR2[LPTM]. 1 If ESR2[VPS] is asserted, there is at least one inactive Mailbox. LPTM content is the number of the first one. 0 If ESR2[VPS] is asserted, the ESR2[LPTM] is not an inactive Mailbox.
-	This field is reserved. Reserved

26.8.15 CRC Register (FLEXCANx_CRCR)

This register provides information about the CRC of transmitted messages

Address: Base address + 44h offset



FLEXCANx_CRCR field descriptions

Field	Description
31–23 -	This field is reserved. Reserved
22–16 MBCRC	This field indicates the number of the Mailbox corresponding to the value in TXCRC field.
15 -	This field is reserved. Reserved
TXCRC	This field indicates the CRC value of the last message transmitted. This field is updated at the same time the Tx Interrupt Flag is asserted.

26.8.16 Rx FIFO Global Mask Register (FLEXCANx_RXFGMASK)

If Rx FIFO is enabled RXFGMASK is used to mask the Rx FIFO ID Filter Table elements that do not have a corresponding RXIMR according to CTRL2[RFFN] field setting.

This register can only be written in Freeze Mode as it is blocked by hardware in other modes.

Table 26-65. Rx FIFO Global Mask usage

Rx FIFO ID Filter Table Elements Format (MCR[IDAM])	Identifier Acceptance Filter fields					
	RTR	IDE	RXIDA	RXIDB	RXIDC	reserved
A	FGM[31]	FGM[30]	FGM[29:1]	-	-	FGM[0]
B	FGM[31] FGM[15]	FGM[30] FGM[14]	-	FGM[29:16] FGM[13:0] 1		-
C	-	-		-	FGM[31:24] FGM[23:16] FGM[15:8] FGM[7:0] 2	

1. If MCR[IDAM] field is equivalent to the format B only the fourteen most significant bits of the Identifier of the incoming frame are compared with the Rx FIFO filter.
2. If MCR[IDAM] field is equivalent to the format C only the eight most significant bits of the Identifier of the incoming frame are compared with the Rx FIFO filter.

Address: Base address + 48h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	FGM31_FGM0																															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

FLEXCANx_RXFGMASK field descriptions

Field	Description
FGM31_FGM0	These bits mask the ID Filter Table elements bits in a perfect alignment. Rx FIFO Global Mask Register (FLEXCAN_RXFGMASK) shows in detail which FGM bits mask each IDAF field. Clear this register has the effect of disabling the ID Filter. 1 The corresponding bit in the filter is checked 0 The corresponding bit in the filter is "don't care"

26.8.17 Rx FIFO Information Register (FLEXCANx_RXFIR)

RXFIR provides information on Rx FIFO.

This register is the port through which ARM accesses the output of the RXFIR FIFO located in RAM. The RXFIR FIFO is written by the FlexCAN whenever a new message is moved into the Rx FIFO as well as its output is updated whenever the output of the Rx FIFO is updated with the next message. Refer to Rx FIFO to find instructions on reading this register.

Address: Base address + 4Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved																							IDHIT								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_RXFIR field descriptions

Field	Description
31–9 -	This field is reserved. Reserved
IDHIT	This 9-bit field indicates which Identifier Acceptance Filter (see Rx FIFO Structure) was hit by the received message that is in the output of the Rx FIFO. (refer to Rx FIFO for details) If multiple filters match the incoming message ID then the first matching IDAF found (lowest number) by the matching process is indicated. This field is valid only while the IFLAG[BUF5I] is asserted.

26.8.18 Rx Individual Mask Registers (FLEXCANx_RXIMR0_RXIMR63)

RXIMR are used as acceptance masks for ID filtering in Rx MBs and the Rx FIFO. If the Rx FIFO is not enabled, one mask register is provided for each available Mailbox, providing ID masking capability on a per Mailbox basis.

When the Rx FIFO is enabled (MCR[RFEN] bit is asserted), up to 32 Rx Individual Mask Registers can apply to the Rx FIFO ID Filter Table elements on a one-to-one correspondence depending on CTRL2[RFFN] setting. Refer to Control 2 Register (FLEXCAN_CTRL2) for details.

RXIMR can only be written by the ARM while the module is in Freeze Mode, otherwise they are blocked by hardware.

The Individual Rx Mask Registers are not affected by reset and must be explicitly initialized prior to any reception.

Address: Base address + 880h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MI31_MI0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FLEXCANx_RXIMR0_RXIMR63 field descriptions

Field	Description
MI31_MI0	These bits mask both Mailbox filter and Rx FIFO ID Filter Table element in distinct ways. For Mailbox filter refer to Rx Mailboxes Global Mask Register (FLEXCAN_RXMGMASK). For Rx FIFO ID Filter Table element refer to Rx FIFO Global Mask Register (FLEXCAN_RXFGMASK). 1 The corresponding bit in the filter is checked 0 the corresponding bit in the filter is "don't care"

26.8.19 Glitch Filter Width Registers (FLEXCANx_GFWR)

The Glitch Filter just takes effects when FLEXCAN enters the STOP mode.

Address: Base address + 9E0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																								GFWR							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1

FLEXCANx_GFWR field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
GFWR	It determines the Glitch Filter Width. The width will be divided from Oscillator clock by GFWR values. By default, it is 5.33us when the oscillator is 24MHz.

Chapter 27
General Power Controller (GPC)

27.1 Overview

The General Power Control (GPC) block includes the sub-blocks listed here.

- DVFS - Dynamic Voltage & Frequency Scaling load tracking for multi-core CPU
- CPU Power Gating Control (PGC)
- GPU/VPU accelerators power gating controller

Each sub-block has its own IP registers.

GPC determines wake-up irq for exiting STOP mode (with or without CPU power gating).

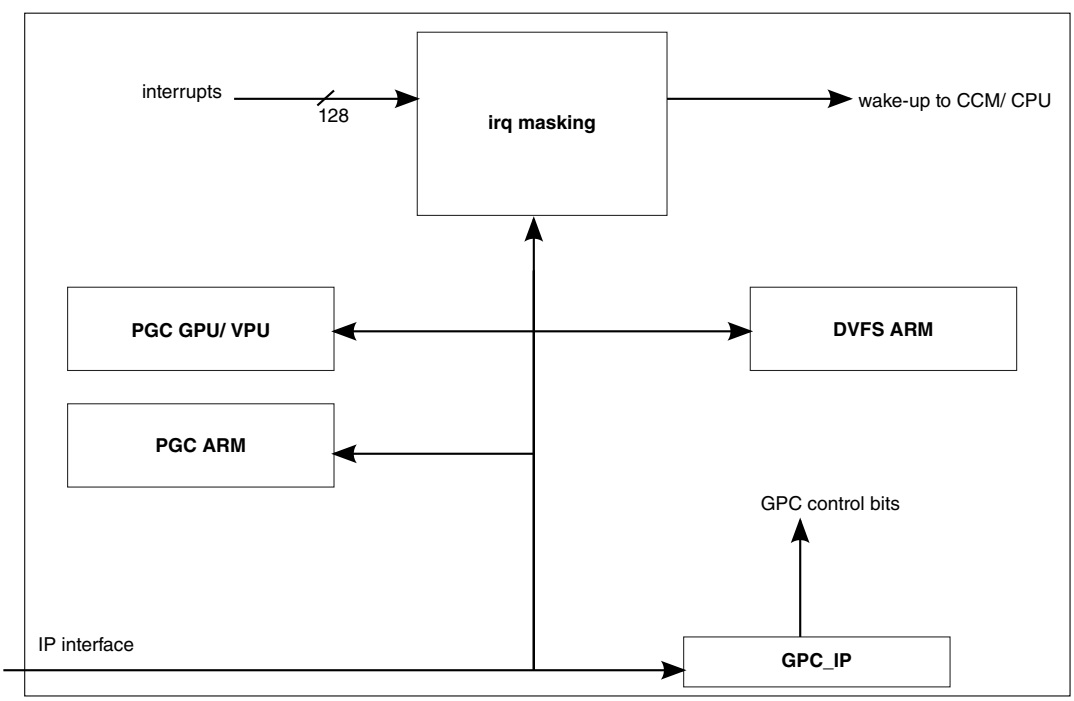


Figure 27-1. GPC Block Diagram

27.2 Clocks

The table found here describes the clock sources for GPC.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 27-1. GPC Clocks

Clock name	Clock Root	Description
ipg_clk	ipg_clk_root	Peripheral clock
ipg_clk_s	ipg_clk_root	Peripheral access clock
pgc_clk	ipg_clk_root	PGC peripheral clock
sys_clk	ipg_clk_root	Module clock

27.3 DVFS overview

The DVFS allows simple dynamic voltage frequency scaling.

The frequency of the core clock domain and the voltage of the core power domain can be changed on the fly while all modules (including the MCU) continue their normal operation. The frequency of the core clock domain can be changed by temporally switching to an alternate PLL clock, or by changing the post dividers division factors.

The DVFS load tracking block allows hardware tracking on the core load and generate an interrupt when a frequency change is requested.

NOTE

DVFS is a monitor that only provides an interrupt when CPU load exceeds the predefined value and does not send any request to make a change of voltage and frequency. This can be done by the user in the CPU interrupt routine or SDMA routine.

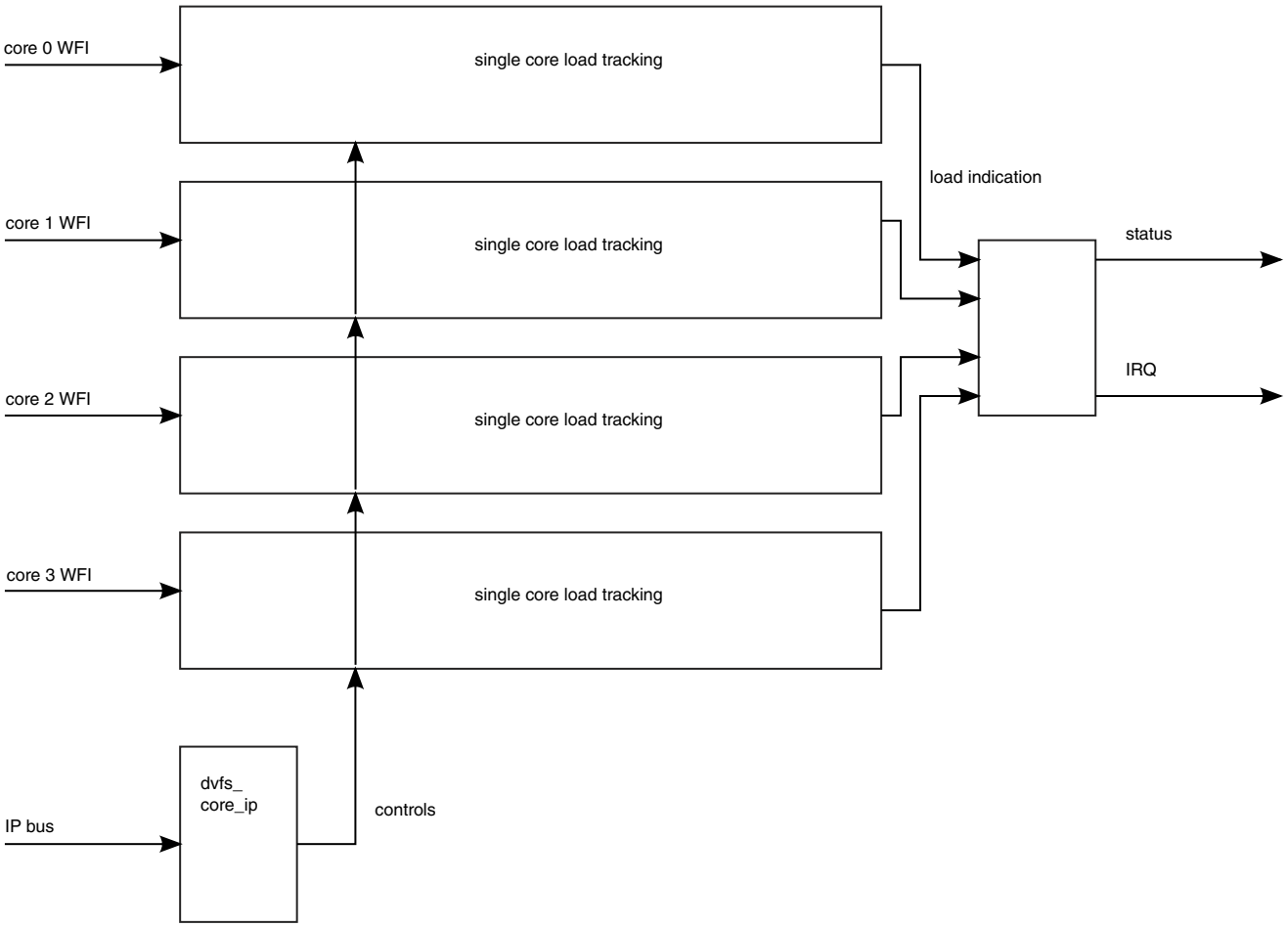


Figure 27-2. DVFS_core diagram

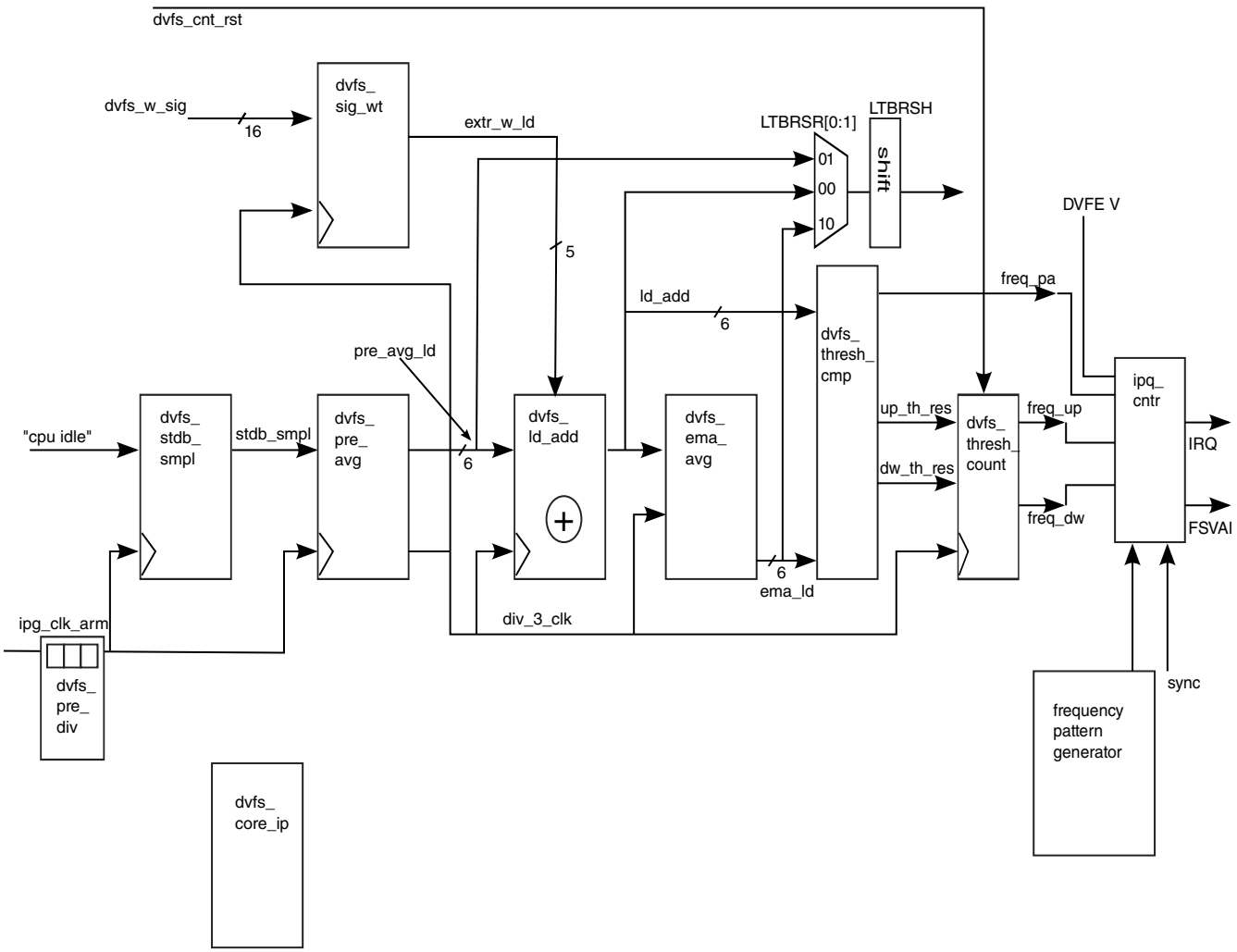


Figure 27-3. DVFS_core: single core load tracking module

27.3.1 Features

The DVFS load tracking block includes the following features:

- Configurable include/exclude of input signals:
  - per-core ARM standby signal (idle / non-idle)
  - 16 general purpose bits (common for all 4 tracking modules)
    - Configurable weight to each bit.
- Configurable generated clocks and averaging time slicing (respond time).
- Configurable panic mode respond logic (for frequency up).
- Programmable buffer for last 4, 8, 12, or 16 load tracking samples of core 0 (only).

27.4 Component Blocks description

27.4.1 dvfs_stdb_smpl block

This block samples the ccm_arm_stdb signal (ARM STANDBYWFI signal - idle state indicating) according to the edge of the ipg_clk_arm (ARM system clock) signal.

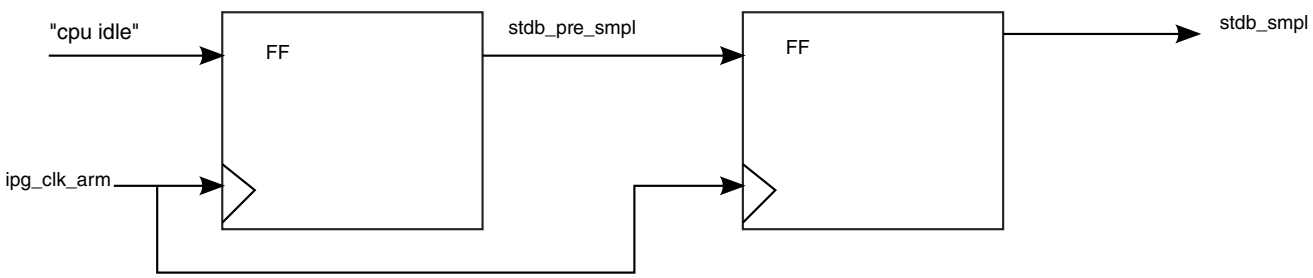


Figure 27-4. dvfs_stdb_smpl block diagram

This block synchronizes the ccm_arm_stdb signal with the ipg_clk_arm clock. The two signals enter the flip-flops twice and become synchronized. The resulting synchronized signal is stdb_smpl.

27.4.2 dvfs_sig_wt block

The purpose of this block is to sample the 16 GeP (general purpose) load signals, multiply each one of them by appropriate weight and sum products.

The sampling is done by the slow clock div_3_clk.

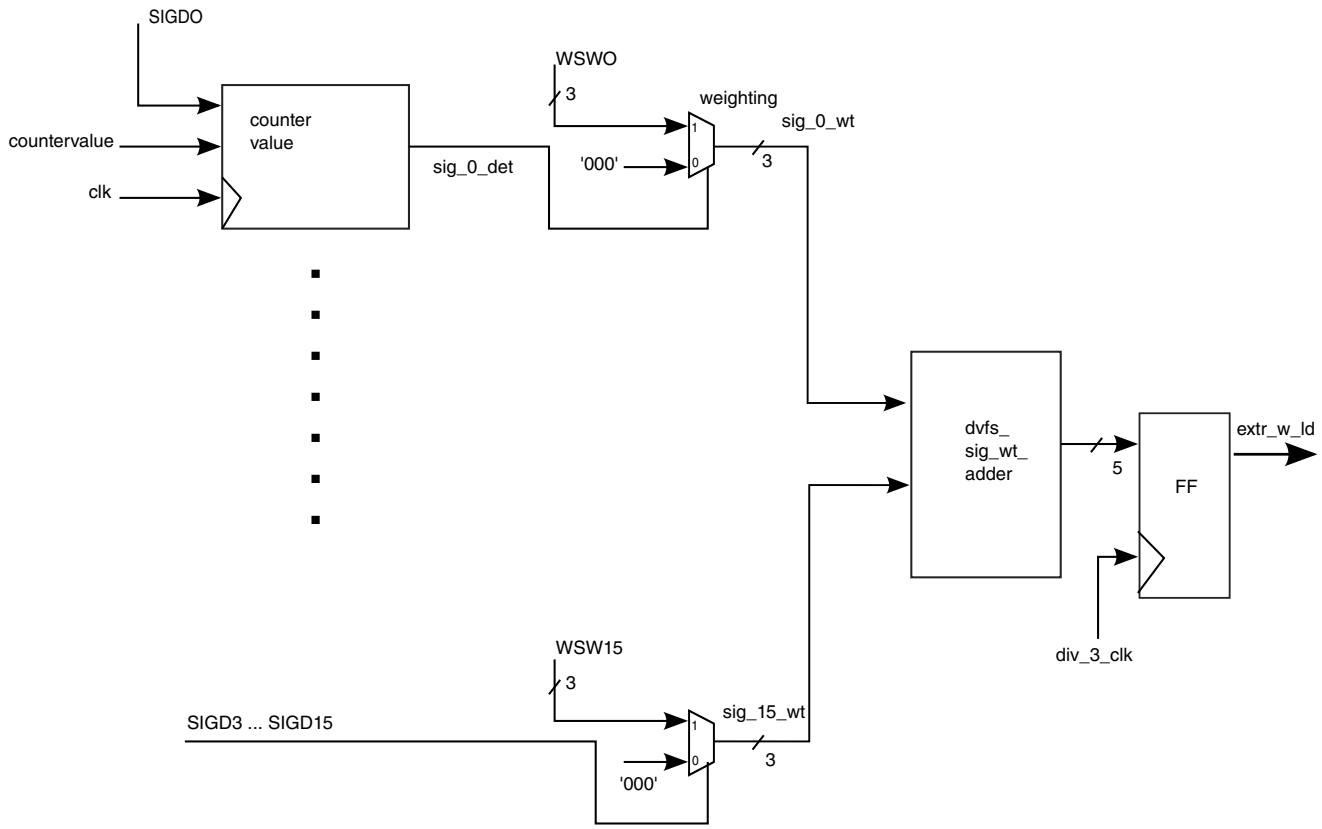


Figure 27-5. dvfs_sig_wt block diagram

27.4.3 dvfs_pre_avg block

The purpose of the dvfs_pre_avg block is to perform simple, non-overlapping averaging, reducing the sampling clock frequency and providing a level-based average index of the tracked CPU load.

External signals:

- stdb_sampl - (input) sampled ARM standby signal
- ipg_clk_arm - (input) ARM system clk
- pre_avg_ld[4..0] - (output) averaged load
- div_3_clk - (output) clock, generated (reduced) from div_2_clk

- div_2_clk - (output) clock, generated (reduced) from div_1_clk
- div_1_clk - (output) clock, generated (reduced) from ipg_clk_arm

In the current implementation, the averaging is performed in three stages:

1. The first stage !IDLE is sampled by divided sys_clock.
2. The second stage (counter 2) performs an averaging operation with constant parameters. The counter is a 9-bit adder whose output is sampled by the div_2_clk clock. The five highest bits of the sum_2 signal are passed to counter 3 (equal to shift right operation).
3. The third (last) stage (counter 3) performs an averaging operation with configurable parameters, set by the DIV3CK. The counter 3 cell is a 15-bit adder with output sampled by div_3_clk. The Pre_avg_shifter_5 cell passes configurable bits from the sum_3 signal.

The output 5 bits avg_load signal provides a result with a tolerance of ~3%. The supported values range is 0-0.97.

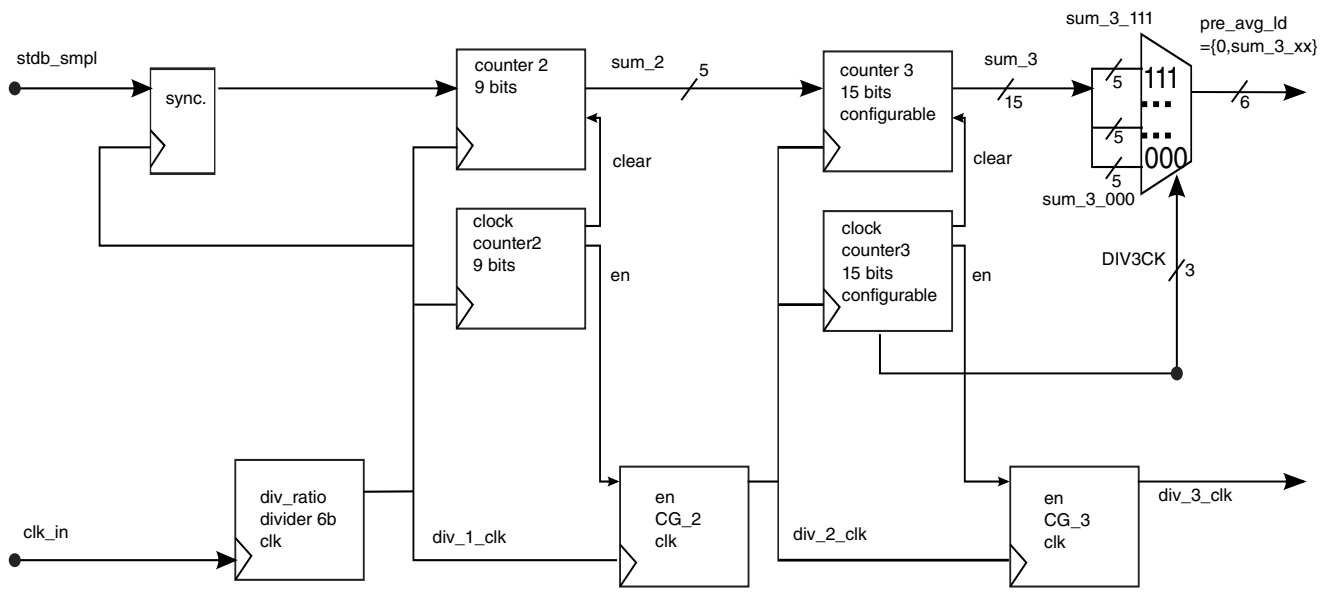


Figure 27-6. dvfs_pre_avg block diagram

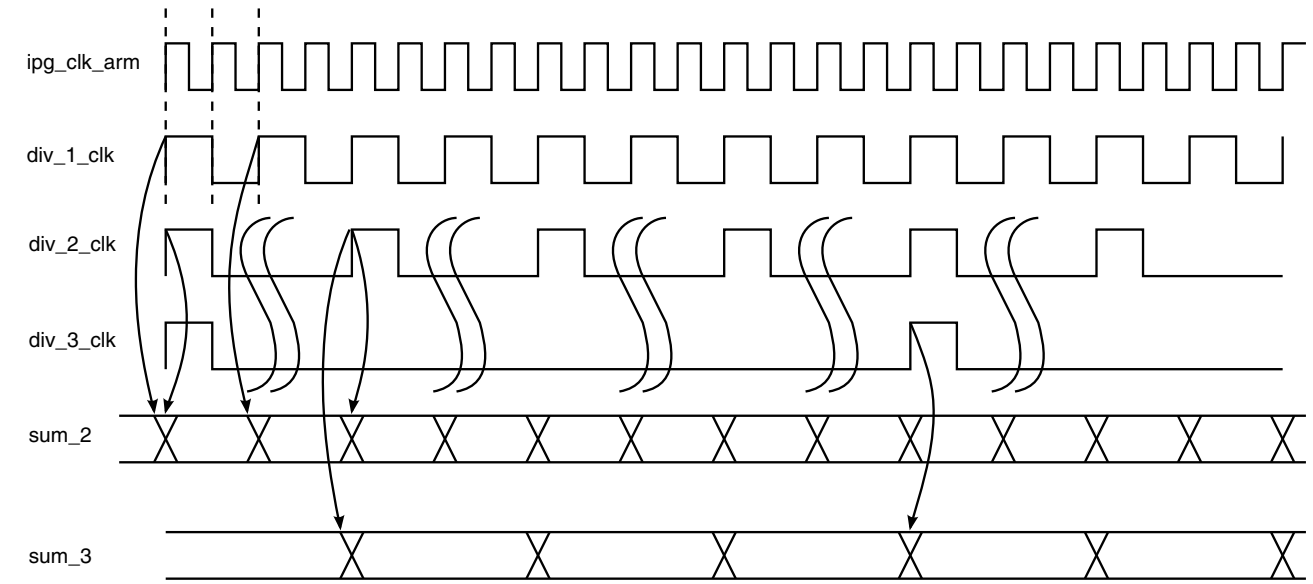


Figure 27-7. dvfs_pre_avg clocks

The internal clocks in dvfs_avg_pre block are generated from ipg_clk_arm. The details are in the tables below. For div_3_clk averaging, see DVFS Control (DVFSC_CNTR) for more information.

Table 27-2. dvfs_pre_avg signals and its values

signal	binary max value	binary min value	decimal max value	decimal min value
sum_2	11111	0000	31	0
sum_3	111'1100'0000'0000	000'0000'0000'0000	31744	0
pre_avg_load	11111	0000	31	0

Table 27-3. dvfs_pre_avg generated clocks

clock name	generated from	ratio to source	ratio to ipg_clk_arm	max clk freq.	note
ipg_clk_arm	N/A	N/A	1	66MHz	source clock
div_1_clk	ipg_clk_arm	configurable	configurable	66MHz	configurable
div_2_clk (gated)	div_1_clk	512	configurable	128KHz	none
div_3_clk (gated)	div_2_clk	configurable	configurable	128KHz	configurable

27.4.4 dvfs_ld_add block

The dvfs_ld_add block sums the CPU load tracked by idle/non-idle signal slot, and the load detected from the additional load signals weighted by dvfs_sig_wt block.

The adder should perform the following operation:

extr_w_ld[4..0] + {0,pre_avg_ld[4..0]}

The input signals of five bits produce output signal of six bits, providing 3% resolution in the range of 0-1.97 of load indication. (1 is equal to 100% load tracking with no additional signals include.)

The output ld_add[5..0] is sampled by div_3_clk signal.

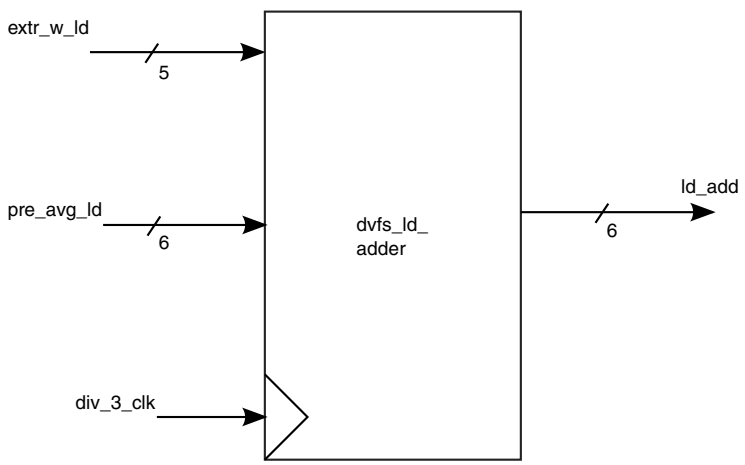


Figure 27-8. dvfs_ld_add block diagram

27.4.5 dvfs_ema_avg block

The purpose of dvfs_ema_avg (EMA - Exponential Moving Average) block is to calculate an exponential moving average of the tracked CPU load.

The parameters of EMA are defined by EMAC bits in DVFSEMAC. These parameters set how many samples of simple average will be taken into account.

The EMA formula is: EMA(i) = a*X(i) + (1-a)*EMA(i-1)

where:

a = 2 / (N+1);

N is the number of samples taken into account.

X(i) = current input sample;

EMA(i-1) = previous value of EMA.

By setting the value of "a", the behavior of EMA is defined.

In the following table, the parameter "a" is listed relatively to the amount of the X(i) samples taken into account ("N") in the equation above: (the resolution of the digital multiplier is limited, hence the lowest values of the "a" can be linked to a range of included samples in EMA instead of single number).

The following table also contains the amount of cycles in which the lower frequency will be masked from the moment that DVFS is enabled (and not between frequency switches), so that enough information about the history can be gained before the frequency is lowered.

Table 27-4. EMA settings

samples included in EMA calculation (N)	Number of cycles while lower frequency interrupt is masked	"a" value (decimal)	"a" value, adjusted by binary resolution	"a" value (binary)								
----------		----------	----------	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7	bit 8
1	6	1.000	1.000	1	0	0	0	0	0	0	0	0
2	11	0.667	0.668	0	1	0	1	0	1	0	1	1
3	11	0.500	0.500	0	1	0	0	0	0	0	0	0
4	22	0.400	0.398	0	0	1	1	0	0	1	1	0
5	22	0.333	0.332	0	0	1	0	1	0	1	0	1
6	22	0.286	0.285	0	0	1	0	0	1	0	0	1
7	22	0.250	0.250	0	0	1	0	0	0	0	0	0
8	43	0.222	0.223	0	0	0	1	1	1	0	0	1
9	43	0.200	0.199	0	0	0	1	1	0	0	1	1
10	43	0.182	0.180	0	0	0	1	0	1	1	1	0
11	43	0.167	0.168	0	0	0	1	0	1	0	1	1
12	43	0.154	0.152	0	0	0	1	0	0	1	1	1
13	43	0.143	0.141	0	0	0	1	0	0	1	0	1
14	43	0.133	0.133	0	0	0	1	0	0	0	1	0
15	43	0.125	0.125	0	0	0	1	0	0	0	0	0
16	86	0.118	0.117	0	0	0	0	1	1	1	1	0
17	86	0.111	0.109	0	0	0	0	1	1	1	0	0
18	86	0.105	0.105	0	0	0	0	1	1	0	1	1
19	86	0.100	0.102	0	0	0	0	1	1	0	1	0
20	86	0.095	0.094	0	0	0	0	1	1	0	0	0
21	86	0.091	0.090	0	0	0	0	1	0	1	1	1
22	86	0.087	0.086	0	0	0	0	1	0	1	1	0

Table continues on the next page...

Table 27-4. EMA settings (continued)

samples included in EMA calculation (N)	Number of cycles while lower frequency interrupt is masked	"a" value (decimal)	"a" value, adjusted by binary resolution	"a" value (binary)								
23	86	0.083	0.082	0	0	0	0	1	0	1	0	1
25	86	0.077	0.078	0	0	0	0	1	0	1	0	0
26	86	0.074	0.074	0	0	0	0	1	0	0	1	1
28	86	0.069	0.070	0	0	0	0	1	0	0	1	0
30	86	0.065	0.066	0	0	0	0	1	0	0	0	1
31	86	0.063	0.063	0	0	0	0	1	0	0	0	0
33	173	0.059	0.059	0	0	0	0	0	1	1	1	1
35	173	0.056	0.055	0	0	0	0	0	1	1	1	0
38	173	0.051	0.051	0	0	0	0	0	1	1	0	1
42	173	0.047	0.047	0	0	0	0	0	1	1	0	0
45	173	0.043	0.043	0	0	0	0	0	1	0	1	1
50	173	0.039	0.039	0	0	0	0	0	1	0	1	0
56	173	0.035	0.035	0	0	0	0	0	1	0	0	1
~64	173	0.031	0.031	0	0	0	0	0	1	0	0	0
~74	256	0.027	0.027	0	0	0	0	0	0	1	1	1
~86	256	0.023	0.023	0	0	0	0	0	0	1	1	0
~100	256	0.020	0.020	0	0	0	0	0	0	1	0	1
~128	256	0.016	0.016	0	0	0	0	0	0	1	0	0
~170	512	0.012	0.012	0	0	0	0	0	0	0	1	1
~260	512	0.008	0.008	0	0	0	0	0	0	0	1	0
~500	512	0.004	0.004	0	0	0	0	0	0	0	0	1
N/A		0.000	0.000	0	0	0	0	0	0	0	0	0

dvfs_ema_avg block diagram

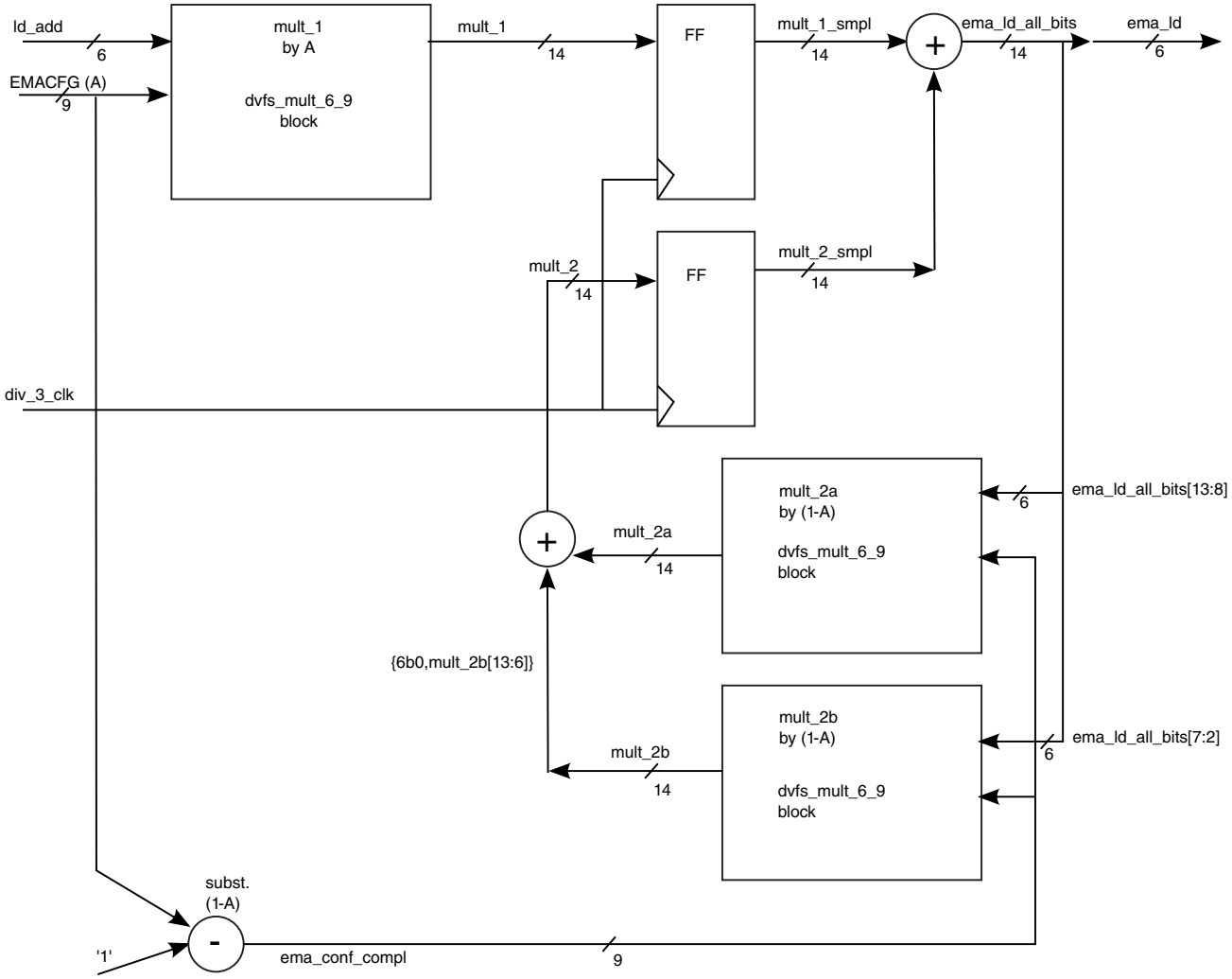


Figure 27-9. dvfs_ema_avg block diagram

The EMA block inputs are as follows:

- ld_add[5..0] - load level
- EMACFG[8..0] - "a" parameter of EMA algorithm
- div_2_clk - fast clock, not shown this in the diagram
- div_3_clk - slow clock

The EMA block outputs the following:

- ema_ld[5..0] - result of EMA algorithm (by div_3_clk)

The mult_a block multiplies between comp_load (6 bits) and EMACFG (9 bits). For the multiply operation, a faster clock (for internal sum) is required, which is provided by div_2_clk. Div_2_clk is faster than the div_3_clk by a factor of at least 16, which is enough for 6*9 or 9*6 operations. Only the highest six bits are taken from the result of mult_a.

The mult_b block operates in a similar manner.

The adder block's output is sampled with div_3_clk clock.

27.4.6 dvfs_thres_cmp block

The dvfs_thres_cmp block compares the CPU load value to programmable threshold levels.

The comparators as shown in the drawing below are used:

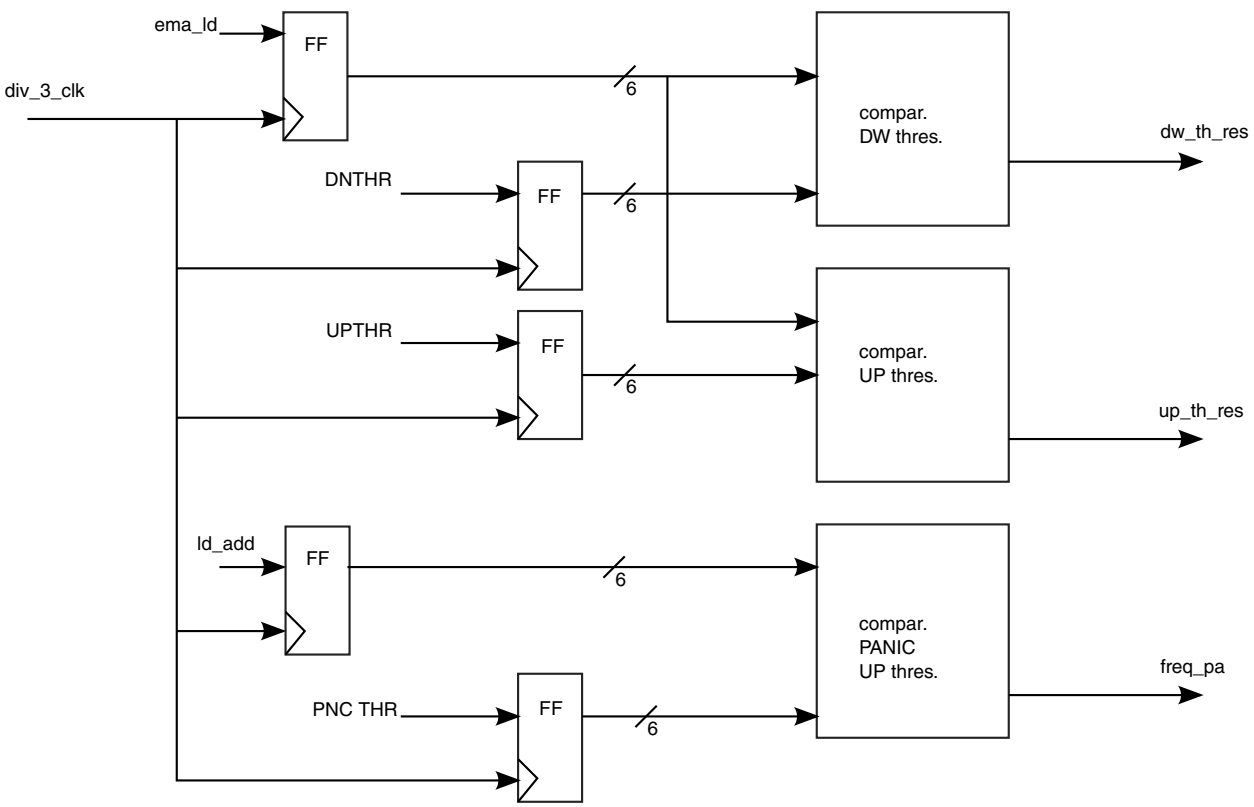


Figure 27-10. dvfs_tresh_cmp diagram

The block is composed of three (3) comparators:

1. compar_dw_thres comparator, which compares between ema_ld signal (output of EMA block) and DNTHR signal (taken from config register, bits DNTHR).

2. compar_up_thres comparator, which compares between ema_ld signal (output of EMA block) and UPTHR signal (taken from config register, bits UPTHR).
3. compar_panic_up_thres comparator, which compares between ld_add signal (output of load_adder block) and PNCTHR signal (taken from config register, bits PNCTHR).

27.4.7 dvfs_thresh_count block

The purpose of the dvfs_thres_count block is to count consecutive threshold overcomes of dw_th_res and up_th_res (outputs of threshold_comp block).

If any of the counters (see the figure below) receives a null (zero) input synchronous with the clk3 signal, the counter is reset.

If the counter reaches a user defined value, its output is set to an active level. These counters are reset each time frequency scaling occurs or if the threshold overcomes are consecutive.

This block's output signals are saved in configuration register 0, bits [3,2].

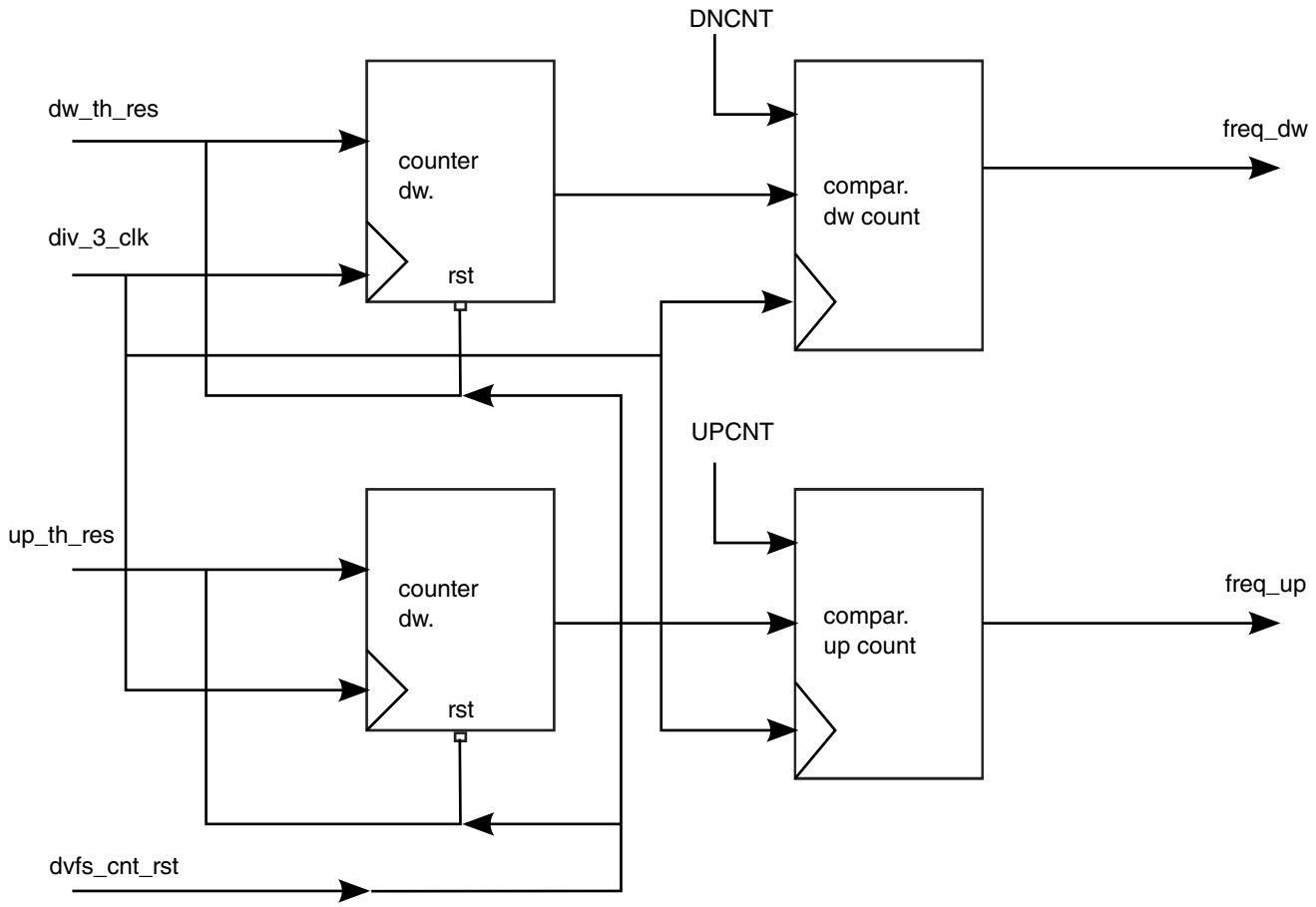


Figure 27-11. Thresh_counters block diagram

27.4.8 Load Tracking Buffer Register

The purpose of the load tracking buffer register is to save last 16 samples of the tracked load (before EMA operation).

Hence, the 4 bits of ld_add signal (depending on the LTBRSH) are saved continuously, overwriting each time the latest sample. Each save is carried upon detecting an edge of the div_3_clk signal.

27.4.9 Frequency Pattern Generator

Frequency pattern generator is able to manage the frequency update requests periodically, additionally to main frequency update requests (if bit FPUE is set).

The periodic requests are created following the DVFSPT0, DVFSPT1, DVFSPT2 and DVFSPT3 register values, as described in the figure below.

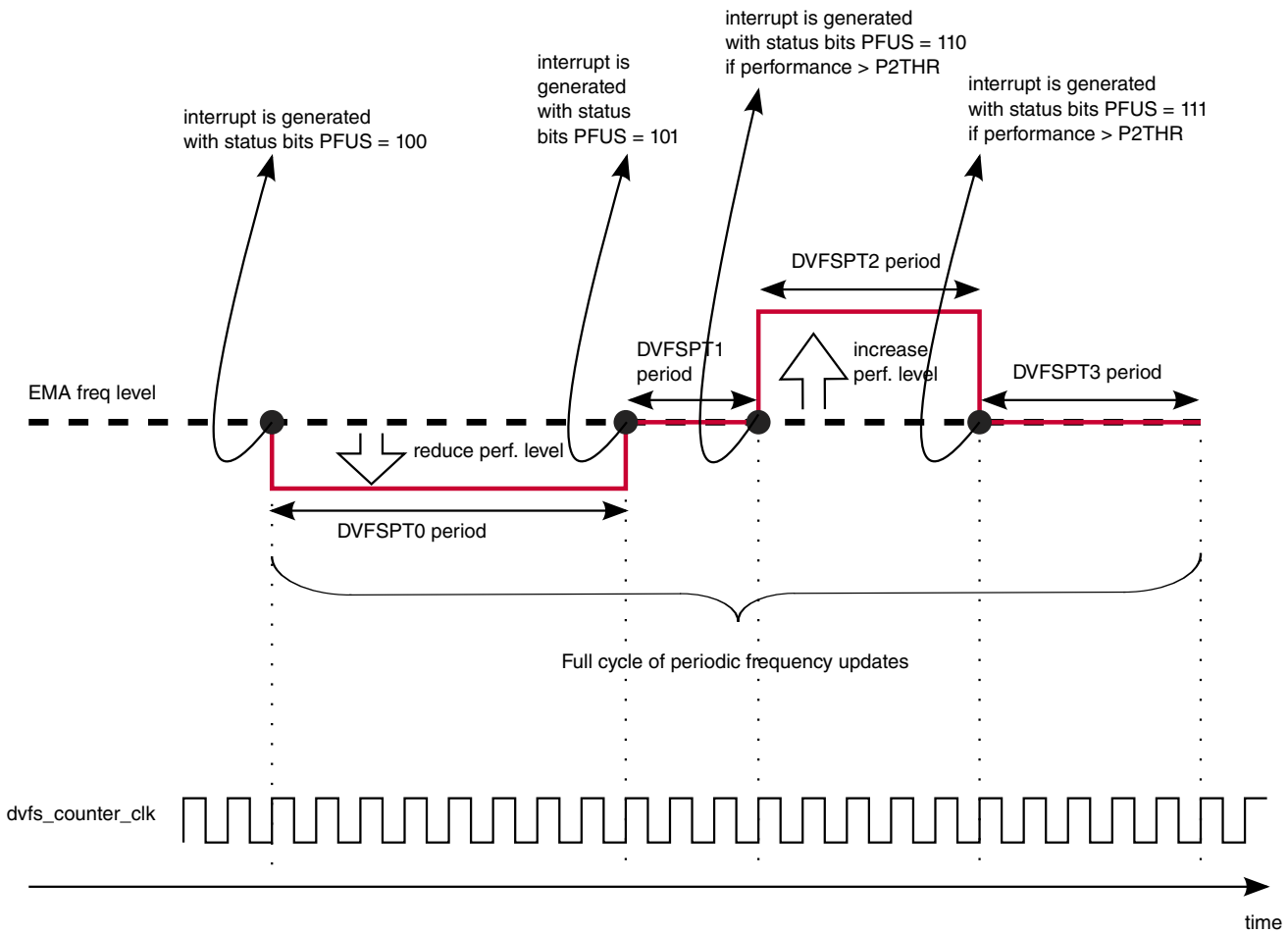


Figure 27-12. DVFS periodic frequency update requests

In case that one of the DVSPT0-DVFSPT3 period is set to "0", such frequency update will be skipped.

The periodic frequency update status is reflected in PFUS bits (reduce/increase performance request is an example - the actual steps taken upon period expiration are defined by s/w routine).

Dvfs_counter_clk is ckih divided 64: 26MHz/64=0.40625MHz. The DVFSPT0, DVFSPT1, DVFSPT2 and DVFSPT3 counter are selected for 17 bits each to provide delay of 2^{18}-1=262143 counts, that are equal to 645ms. On the other hand, clk cycle of 0.40625MHz is ~2.46us, that is fast enough to provide high resolution for frequency management for tasks.

The DVFSPT2 and DVFSPT3 interrupt generation is conditional upon current performance being greater than DVFSPT2[P2THR]. Otherwise the pattern delay will be counted, but without interrupt generation.

27.5 DVFS output event/interrupt configuration

Event/interrupt will be always high as long as LBFL is '1' and is not cleared by SW. Unless DVFEV (always event) is asserted. Then the event/interrupt will be toggled up and down at every toggle of div_3_clk.

27.5.1 Interrupts

DVFS generates an interrupt that indicates that frequency and voltage update is needed. The user has to read the FSVAIM bits in order to know which change needs to be done.

27.5.2 DVFS Change Request Sequence Diagrams

The following figures describe the sequence on DVFS interrupt.

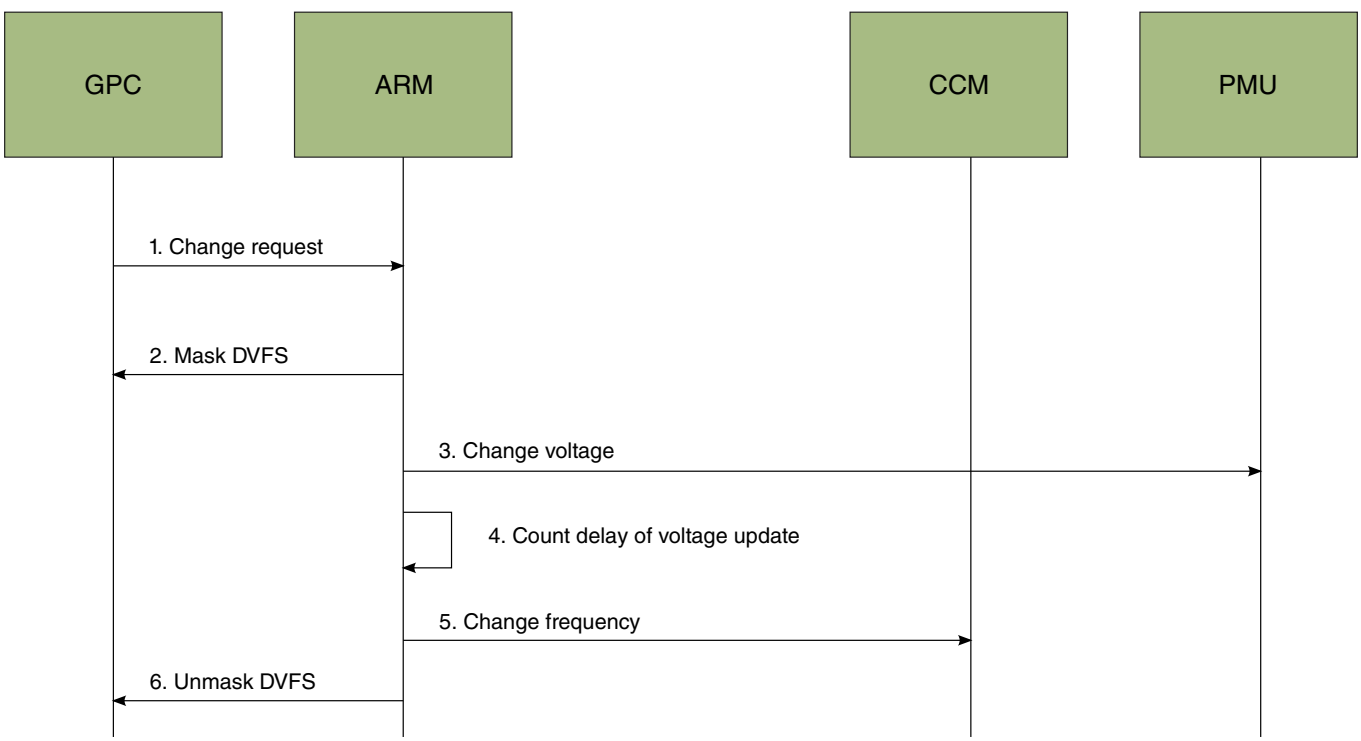


Figure 27-13. DVFS - frequency increase

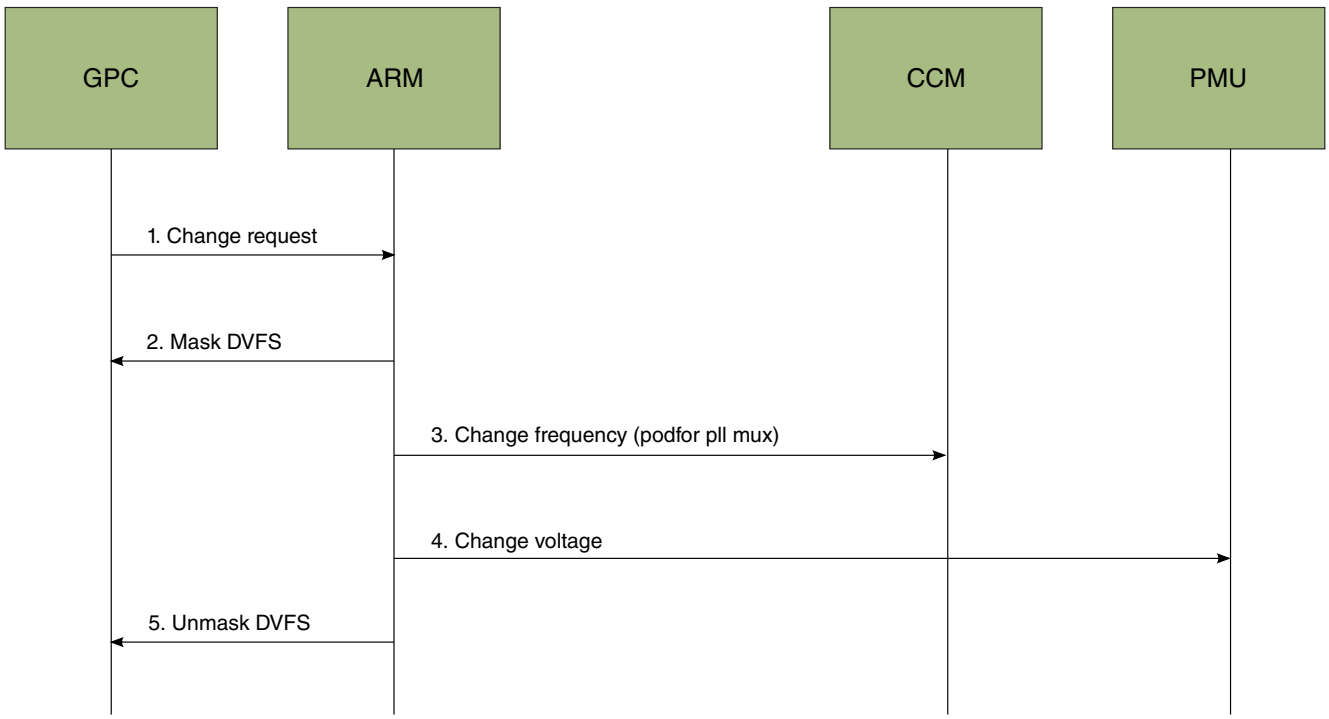


Figure 27-14. DVFS - frequency decrease

CPU DVFS frequency change can be performed in 2 ways:

- PLL inputs muxing update
- clock dividers update

27.6 Power Gating Control (PGC)

Power Gating (PGC) is applied to the ARM CPU only in STOP low power mode, after all essential CPU registers data are saved by ARM dormant procedure.

If any of the unmasked interrupts appears, CPU is powered up and clock restore request (exit from STOP mode) is sent to CCM.

PGC power down sequence:

- CCM sends power down request when the chip is about to enter stop mode. The user should define which modules will be powered down (PGCR registers of corresponding PGC module, bit 0).

PGC power up sequence:

- One of the power up irq is asserted.

- Power up request is asserted in GPC and in CCM.
- The Power Gated modules are powered up, according to PGC settings of appropriate module.

The Power Gated modules require reset after powering up. The next figure describes GPC-SRC handshake procedure for reset after power gating.

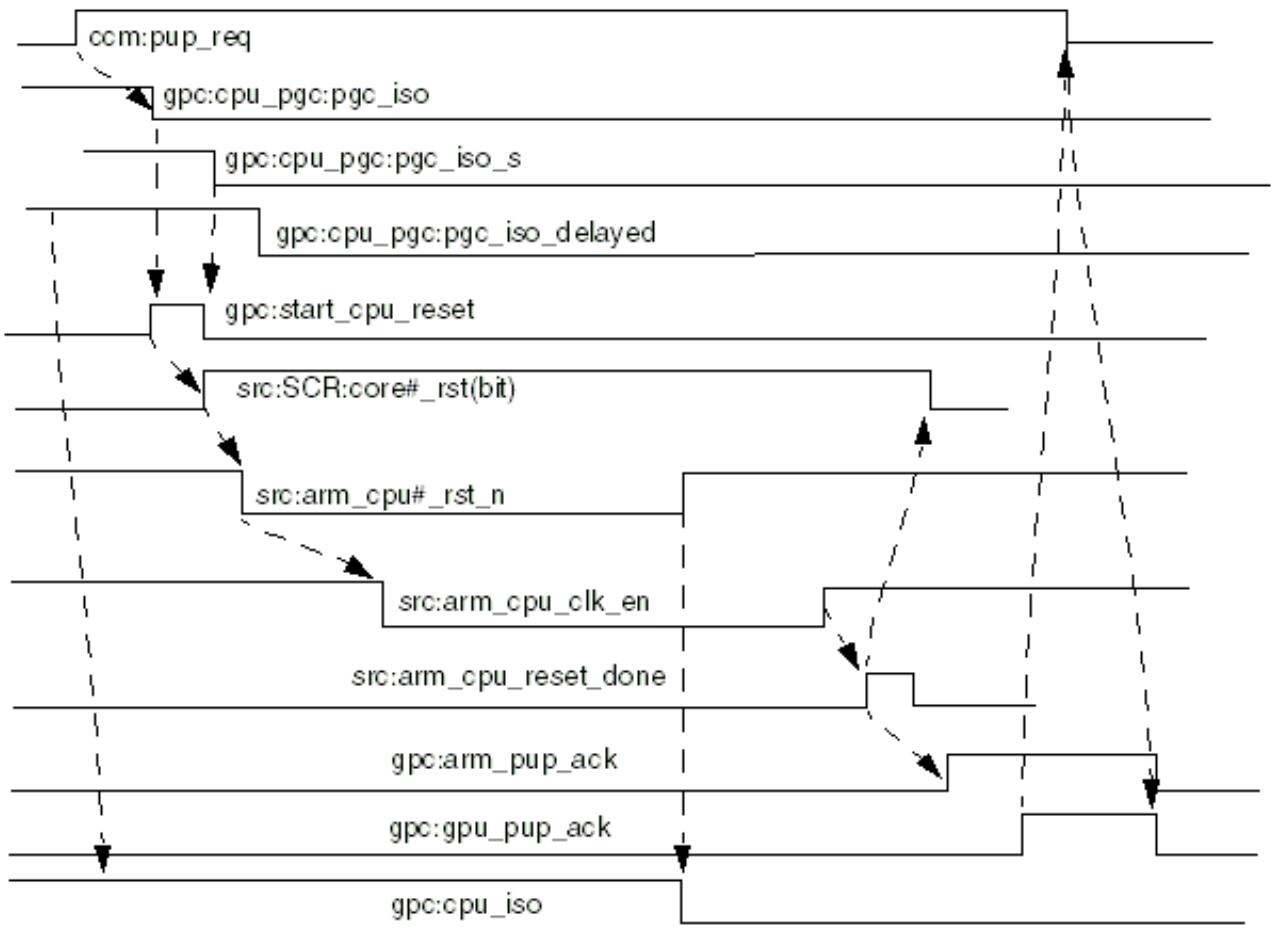


Figure 27-15. GPC-SRC handshake for reset after power gating

27.6.1 Overview

The Power Gating Controller (PGC) is a power management component that controls the power-down and power-up sequencing of individual subsystems.

The sequence timing is programmable using the PGC control registers. Figure 27-16 shows PGC as part of the SoC's overall power management scheme.

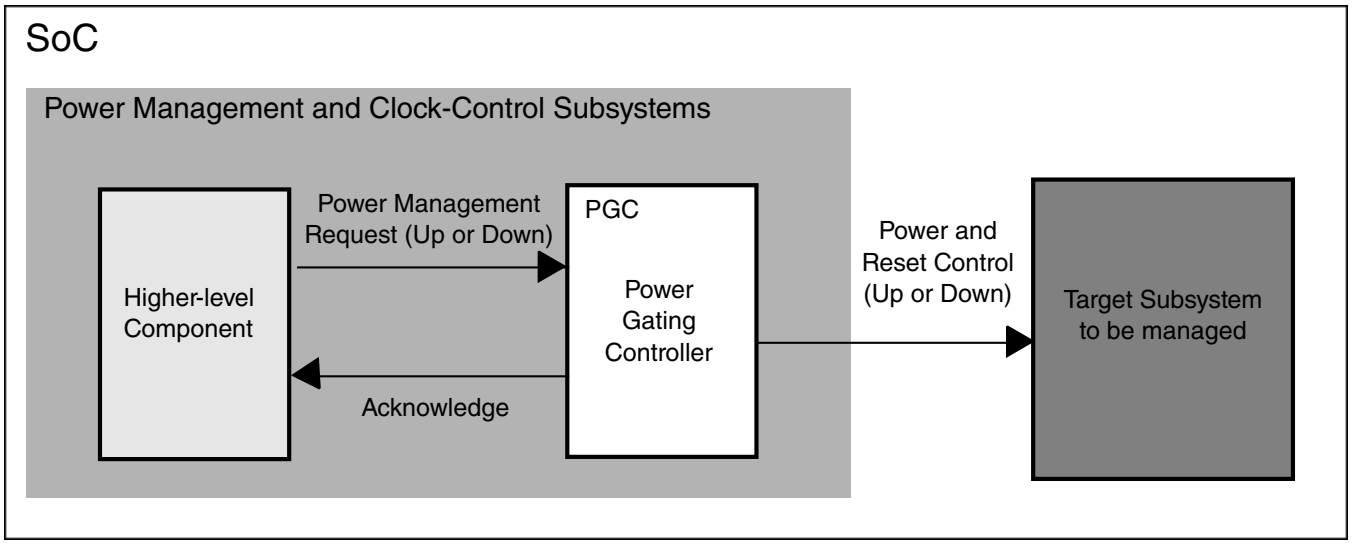


Figure 27-16. PGC Block Diagram

27.6.1.1 Features

Key features of the PGC include:

- Provides the ability to switch off power to a target subsystem.
- Generates power-up and power-down control sequences.
- Provides programmable registers to adjust the timing of the power control signals.

27.7 GPC Interrupt Controller (INTC)

The INTC (Interrupt Controller) detects an interrupt and generates the wakeup signal. It supports up to 128 interrupts.

27.7.1 Interrupt Controller features

The features of the GPC INTC are listed below.

Features:

- Supports up to 128 interrupts
- Provides an option to mask/unmask each interrupt
- Detects interrupts and generates the wake up signal
- 32-bits IP bus interface
- All registers are byte-accessible

27.8 GPC Memory Map/Register Definition

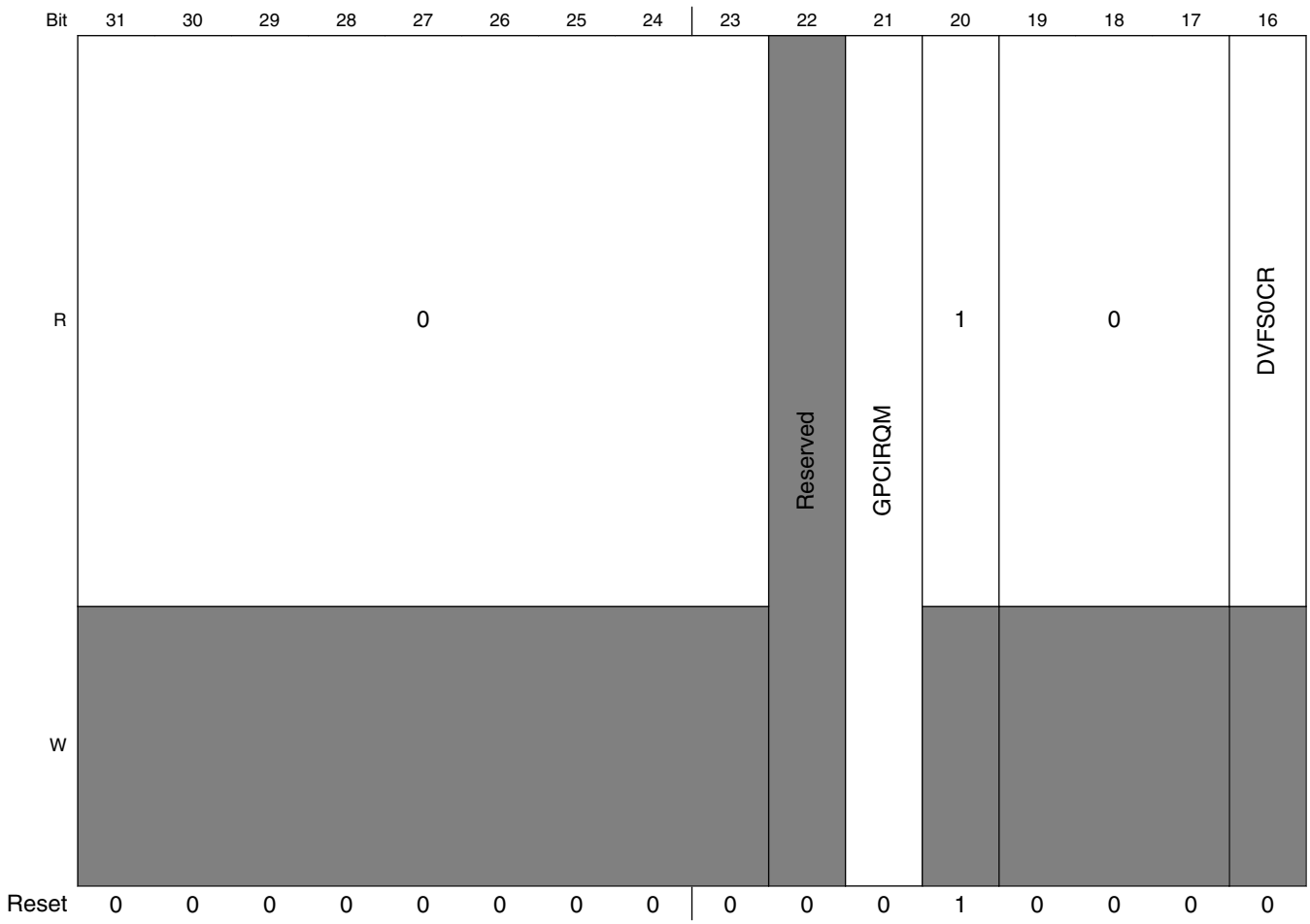
Detailed descriptions of each register can be found below.

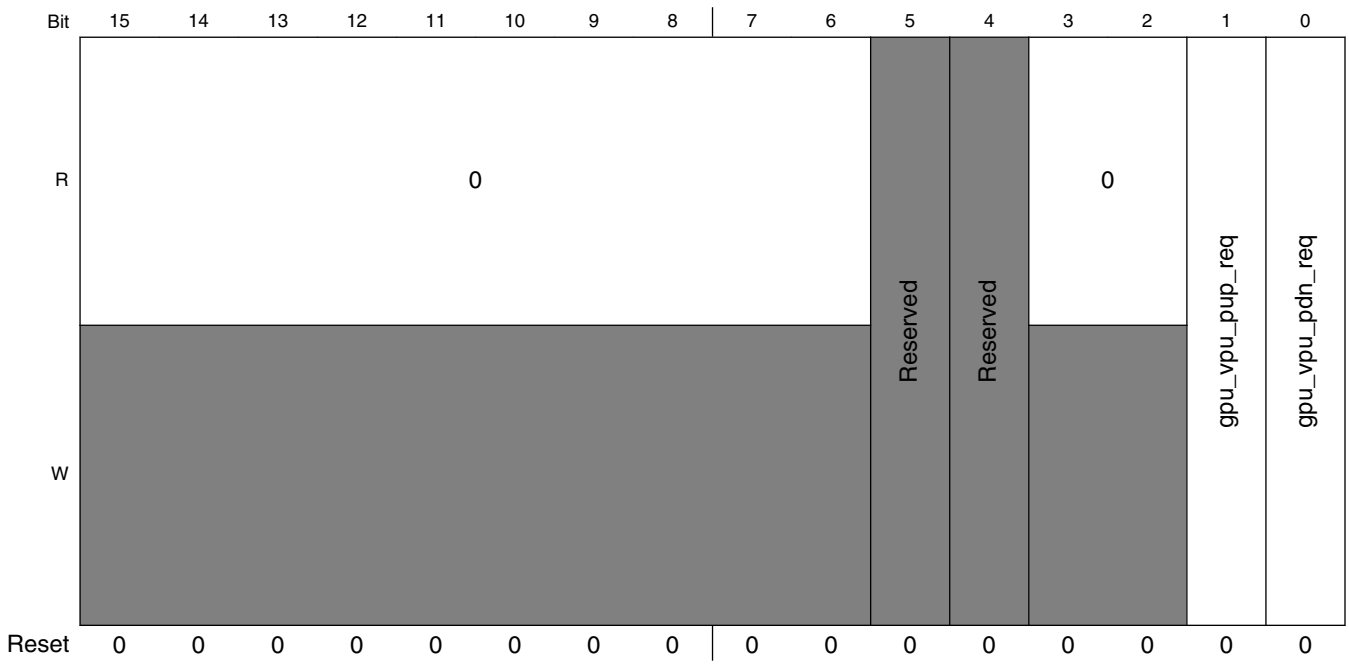
GPC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20D_C000	GPC Interface control register (GPC_CNTR)	32	R/W	0010_0000h	27.8.1/1379
20D_C004	GPC Power Gating Register (GPC_PGR)	32	R/W	0000_0000h	27.8.2/1382
20D_C008	IRQ masking register 1 (GPC_IMR1)	32	R/W	0000_0000h	27.8.3/1382
20D_C00C	IRQ masking register 2 (GPC_IMR2)	32	R/W	0000_0000h	27.8.4/1383
20D_C010	IRQ masking register 3 (GPC_IMR3)	32	R/W	0000_0000h	27.8.5/1383
20D_C014	IRQ masking register 4 (GPC_IMR4)	32	R/W	0000_0000h	27.8.6/1383
20D_C018	IRQ status resister 1 (GPC_ISR1)	32	R	0000_0000h	27.8.7/1384
20D_C01C	IRQ status resister 2 (GPC_ISR2)	32	R	0000_0000h	27.8.8/1384
20D_C020	IRQ status resister 3 (GPC_ISR3)	32	R	0000_0000h	27.8.9/1385
20D_C024	IRQ status resister 4 (GPC_ISR4)	32	R	0000_0000h	27.8.10/ 1385

27.8.1 GPC Interface control register (GPC_CNTR)

Address: 20D_C000h base + 0h offset = 20D_C000h



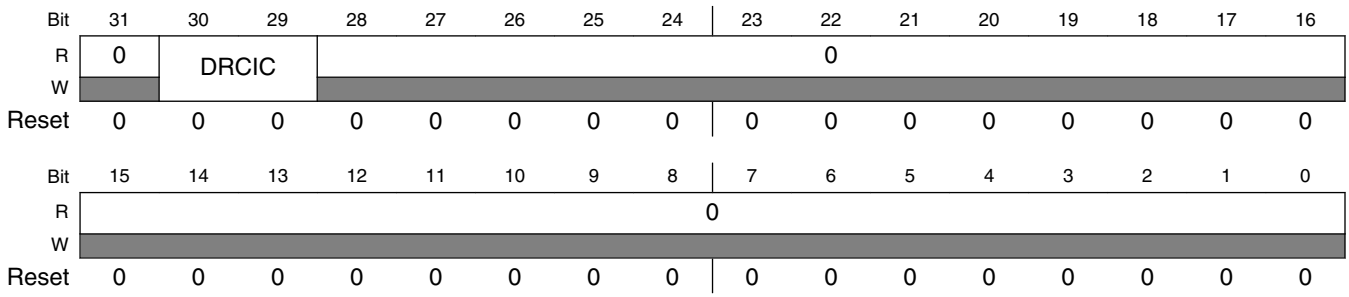


GPC_CNTR field descriptions

Field	Description
31–23 Reserved	This read-only field is reserved and always has the value 0.
22 -	This field is reserved. Reserved
21 GPCIRQM	GPC interrupt/event masking 1 interrupt/event is masked 0 not masked
20 Reserved	This read-only field is reserved and always has the value 1.
19–17 Reserved	This read-only field is reserved and always has the value 0.
16 DVFS0CR	DVFS0 (ARM) Change request (bit is read-only) 1 DVFS0 is requesting for frequency/voltage update 0 DVFS0 has no request
15–6 Reserved	This read-only field is reserved and always has the value 0.
5 -	This field is reserved. Reserved
4 -	This field is reserved. Reserved
3–2 Reserved	This read-only field is reserved and always has the value 0.
1 gpu_vpu_pup_req	GPU/VPU Power Up request. Self-cleared bit. * Note: Power switch for GPU/VPU power domain is controlled by ANALOG configuration, not GPU/VPU PGC signals 1 Request Power Up sequence to start for GPU/VPU 0 no request
0 gpu_vpu_pdn_req	GPU/VPU Power Down request. Self-cleared bit. * Note: Power switch for GPU/VPU power domain is controlled by ANALOG configuration, not GPU/VPU PGC signals 1 Request Power Down sequence to start for GPU/VPU 0 no request

27.8.2 GPC Power Gating Register (GPC_PGR)

Address: 20D_C000h base + 4h offset = 20D_C004h



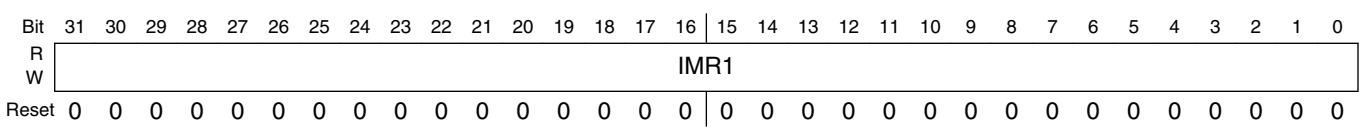
GPC_PGR field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 DRCIC	Debug ref cir in mux control 00 ccm_cosr_1_clk_in 01 ccm_cosr_2_clk_in 10 restricted 11 restricted
Reserved	This read-only field is reserved and always has the value 0.

27.8.3 IRQ masking register 1 (GPC_IMR1)

IMR1 Register - masking of irq[63:32].

Address: 20D_C000h base + 8h offset = 20D_C008h



GPC_IMR1 field descriptions

Field	Description
IMR1	IRQ[63:32] masking bits: 1-irq masked, 0-irq is not masked

27.8.4 IRQ masking register 2 (GPC_IMR2)

IMR2 Register - masking of irq[95:64].

Address: 20D_C000h base + Ch offset = 20D_C00Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IMR2																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPC_IMR2 field descriptions

Field	Description
IMR2	IRQ[95:64] masking bits: 1-irq masked, 0-irq is not masked

27.8.5 IRQ masking register 3 (GPC_IMR3)

IMR3 Register - masking of irq[127:96].

Address: 20D_C000h base + 10h offset = 20D_C010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IMR3																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPC_IMR3 field descriptions

Field	Description
IMR3	IRQ[127:96] masking bits: 1-irq masked, 0-irq is not masked

27.8.6 IRQ masking register 4 (GPC_IMR4)

IMR4 Register - masking of irq[159:128].

Address: 20D_C000h base + 14h offset = 20D_C014h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IMR4																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

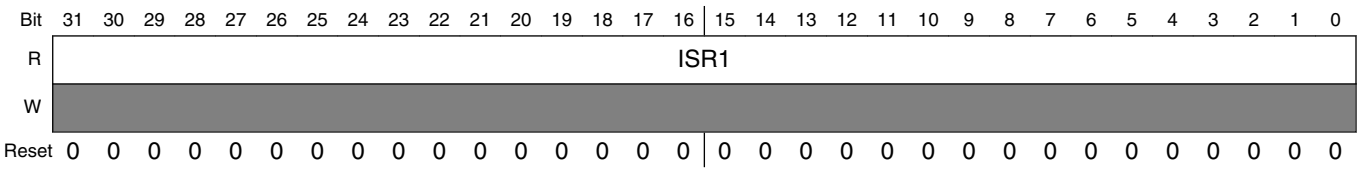
GPC_IMR4 field descriptions

Field	Description
IMR4	IRQ[159:128] masking bits: 1-irq masked, 0-irq is not masked

27.8.7 IRQ status resister 1 (GPC_ISR1)

ISR1 Register - status of irq [63:32].

Address: 20D_C000h base + 18h offset = 20D_C018h



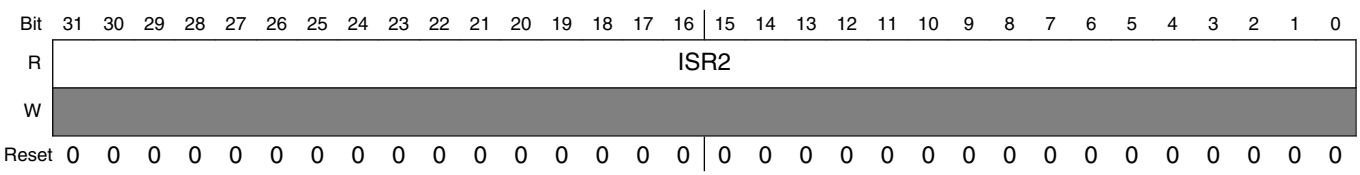
GPC_ISR1 field descriptions

Field	Description
ISR1	IRQ[63:32] status, read only

27.8.8 IRQ status resister 2 (GPC_ISR2)

ISR2 Register - status of irq [95:64].

Address: 20D_C000h base + 1Ch offset = 20D_C01Ch



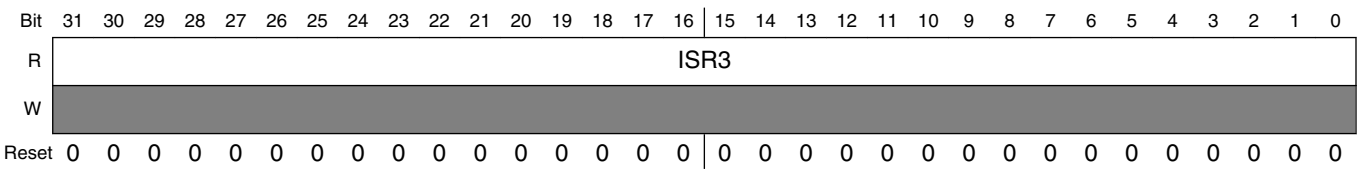
GPC_ISR2 field descriptions

Field	Description
ISR2	IRQ[95:64] status, read only

27.8.9 IRQ status resister 3 (GPC_ISR3)

ISR3 Register - status of irq [127:96].

Address: 20D_C000h base + 20h offset = 20D_C020h



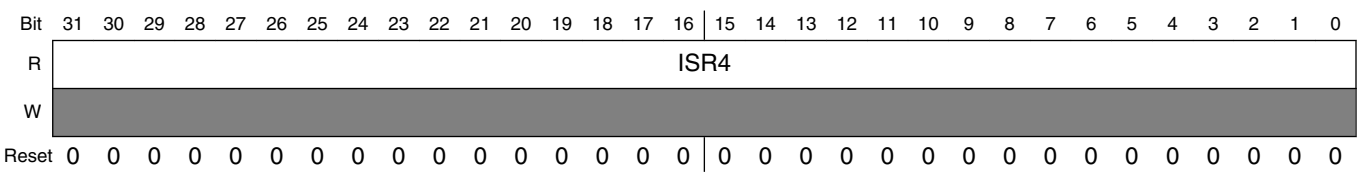
GPC_ISR3 field descriptions

Field	Description
ISR3	IRQ[127:96] status, read only

27.8.10 IRQ status resister 4 (GPC_ISR4)

ISR4 Register - status of irq [159:128].

Address: 20D_C000h base + 24h offset = 20D_C024h



GPC_ISR4 field descriptions

Field	Description
ISR4	IRQ[159:128] status, read only

27.9 PGC Memory Map/Register Definition

The PGC registers can be accessed only in supervisor mode.

Attempts to access registers when not in supervisor mode or attempts to access an unimplemented address location might trigger a bus transfer error. (The hardware asserts the signal ips_xfr_err if the PGC has been integrated with resp_sel tied low.) In this case, software should take appropriate action (such as ignore the error, log the error, or initiate a soft reset).

All PGC registers are byte-accessible.

NOTE

The base address of each PGC module instantiation is specified in the GPC module. Absolute address values will be calculated by [GPC base address] + [PGC CPU/GPU/DISPLAY Offset].

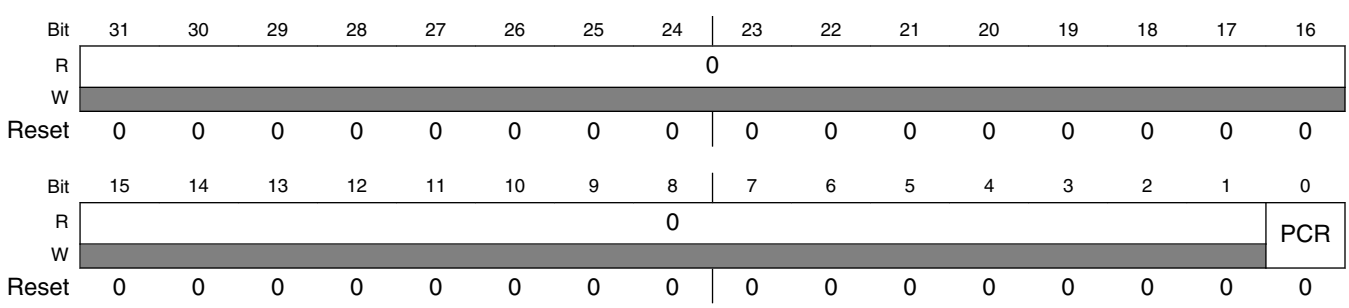
PGC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20D_C260	PGC Control Register (PGC_GPU_CTRL)	32	R/W	0000_0000h	27.9.1/1386
20D_C264	Power Up Sequence Control Register (PGC_GPU_PUPSCR)	32	R/W	0000_0F01h	27.9.2/1387
20D_C268	Pull Down Sequence Control Register (PGC_GPU_PDNSCR)	32	R/W	0000_0101h	27.9.3/1388
20D_C26C	Power Gating Controller Status Register (PGC_GPU_SR)	32	R/W	0000_0000h	27.9.4/1388
20D_C2A0	PGC Control Register (PGC_CPU_CTRL)	32	R/W	0000_0000h	27.9.5/1389
20D_C2A4	Power Up Sequence Control Register (PGC_CPU_PUPSCR)	32	R/W	0000_0F01h	27.9.6/1390
20D_C2A8	Pull Down Sequence Control Register (PGC_CPU_PDNSCR)	32	R/W	0000_0101h	27.9.7/1390
20D_C2AC	Power Gating Controller Status Register (PGC_CPU_SR)	32	R/W	0000_0000h	27.9.8/1391

27.9.1 PGC Control Register (PGC_GPU_CTRL)

The PGCR enables the response to a power-down request.

Address: 20D_C000h base + 260h offset = 20D_C260h



PGC_GPU_CTRL field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 PCR	Power Control **NOTE:** PCR must not change from power-down request (pdn_req) assertion until the target subsystem is completely powered up. 0 Do not switch off power even if pdn_req is asserted. 1 Switch off power when pdn_req is asserted.

27.9.2 Power Up Sequence Control Register (PGC_GPU_PUPSCR)

The PUPSCR contains the power-up timing parameters.

Address: 20D_C000h base + 264h offset = 20D_C264h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																SW2ISO								0		SW					
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1

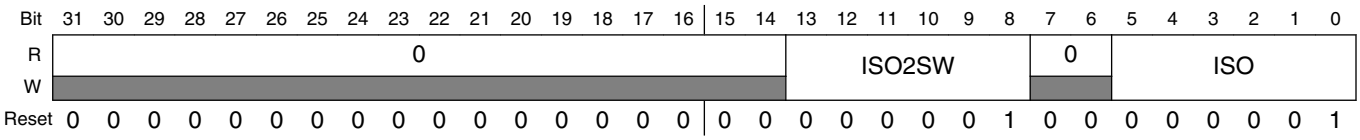
PGC_GPU_PUPSCR field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13–8 SW2ISO	After asserting power toggle on/off signal (switch_b), the PGC waits a number of IPG clocks equal to the value of SW2ISO before negating isolation. **NOTE:** SW2ISO must not be programmed to zero.
7–6 Reserved	This read-only field is reserved and always has the value 0.
SW	After a power-up request (pup_req assertion), the PGC waits a number of IPG clocks equal to the value of SW before asserting power toggle on/off signal (switch_b). **NOTE:** SW must not be programmed to zero. **NOTE:** The PGC clock is generated from the IPG_CLK_ROOT. for frequency configuration of the IPG_CLK_ROOT. See Clock Controller Module (CCM).

27.9.3 Pull Down Sequence Control Register (PGC_GPU_PDNSCR)

The PDNSCR contains the power-down timing parameters.

Address: 20D_C000h base + 268h offset = 20D_C268h



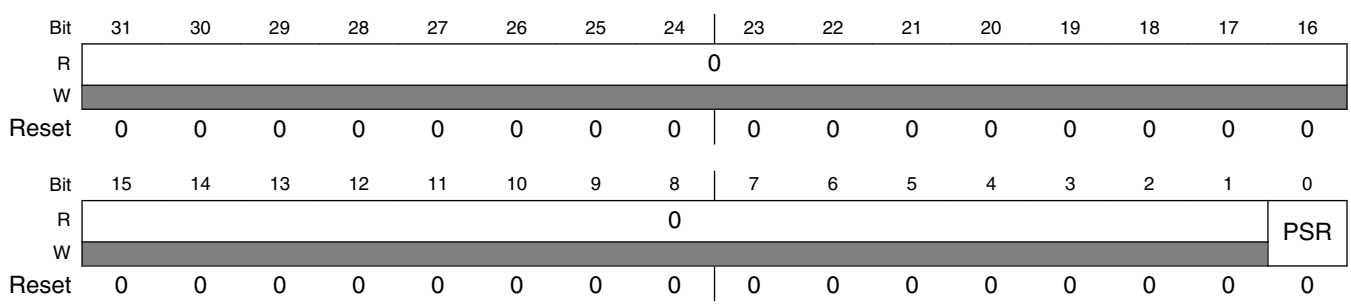
PGC_GPU_PDNSCR field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13–8 ISO2SW	After asserting isolation, the PGC waits a number of IPG clocks equal to the value of ISO2SW before negating power toggle on/off signal (switch_b). **NOTE:** ISO2SW must not be programmed to zero.
7–6 Reserved	This read-only field is reserved and always has the value 0.
ISO	After a power-down request (pdn_req assertion), the PGC waits a number of IPG clocks equal to the value of ISO before asserting isolation. **NOTE:** ISO must not be programmed to zero.

27.9.4 Power Gating Controller Status Register (PGC_GPU_SR)

The PDNSCR contains the power-down timing parameters.

Address: 20D_C000h base + 26Ch offset = 20D_C26Ch



PGC_GPU_SR field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 PSR	Power status. When in functional (or software-controlled debug) mode, PGC hardware sets PSR as soon as any of the power control output changes its state to one. Write one to clear this bit. Software should clear this bit after power up; otherwise, PSR continues to reflect the power status of the initial power down. 0 The target subsystem was not powered down for the previous power-down request. 1 The target subsystem was powered down for the previous power-down request.

27.9.5 PGC Control Register (PGC_CPU_CTRL)

The PGCR enables the response to a power-down request.

Address: 20D_C000h base + 2A0h offset = 20D_C2A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															PCR
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

PGC_CPU_CTRL field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 PCR	Power Control **NOTE:** PCR must not change from power-down request (pdn_req) assertion until the target subsystem is completely powered up. 0 Do not switch off power even if pdn_req is asserted. 1 Switch off power when pdn_req is asserted.

27.9.6 Power Up Sequence Control Register (PGC_CPU_PUPSCR)

The PUPSCR contains the power-up timing parameters.

Address: 20D_C000h base + 2A4h offset = 20D_C2A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																		SW2ISO						0		SW					
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1

PGC_CPU_PUPSCR field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13–8 SW2ISO	After asserting , the PGC waits a number of 32k clocks equal to the value of SW2ISO before negating isolation. **NOTE:** SW2ISO must not be programmed to zero. The SW2ISO value should be chosen such that the delay before negating isolation is greater than the LDO ramp-up time.
7–6 Reserved	This read-only field is reserved and always has the value 0.
SW	After a power-up request (pup_req assertion), the PGC waits a number of 32k clocks equal to the value of SW before asserting . **NOTE:** SW must not be programmed to zero.

27.9.7 Pull Down Sequence Control Register (PGC_CPU_PDNSCR)

The PDNSCR contains the power-down timing parameters.

Address: 20D_C000h base + 2A8h offset = 20D_C2A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																		ISO2SW						0		ISO					
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

PGC_CPU_PDNSCR field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13–8 ISO2SW	After asserting isolation, the PGC waits a number of 32k clocks equal to the value of ISO2SW before negating . **NOTE:** ISO2SW must not be programmed to zero.
7–6 Reserved	This read-only field is reserved and always has the value 0.
ISO	After a power-down request (pdn_req assertion), the PGC waits a number of 32k clocks equal to the value of ISO before asserting isolation. **NOTE:** ISO must not be programmed to zero.

27.9.8 Power Gating Controller Status Register (PGC_CPU_SR)

The PDNSCR contains the power-down timing parameters.

Address: 20D_C000h base + 2ACh offset = 20D_C2ACh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															PSR
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

PGC_CPU_SR field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 PSR	Power status. When in functional (or software-controlled debug) mode, PGC hardware sets PSR as soon as any of the power control output changes its state to one. Write one to clear this bit. Software should clear this bit after power up; otherwise, PSR continues to reflect the power status of the initial power down. 0 The target subsystem was not powered down for the previous power-down request. 1 The target subsystem was powered down for the previous power-down request.

27.10 DVFSC Memory Map/Register Definition

DVFSC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20D_C180	DVFS Thresholds (DVFSC_THRS)	32	R/W	0FAF_003Eh	27.10.1/ 1392
20D_C184	DVFS Counters thresholds (DVFSC_COUN)	32	R/W	0007_0020h	27.10.2/ 1393
20D_C188	DVFS general purpose bits weight (DVFSC_SIG1)	32	R/W	0000_0000h	27.10.3/ 1393
20D_C18C	DVFS general purpose bits weight (DVFSC_DVFSSIG0)	32	R/W	0000_0000h	27.10.4/ 1394
20D_C190	DVFS general purpose bit 0 weight counter (DVFSC_DVFSGPC0)	32	R/W	0000_0000h	27.10.5/ 1395
20D_C194	DVFS general purpose bit 1 weight counter (DVFSC_DVFSGPC1)	32	R/W	0000_0000h	27.10.6/ 1397
20D_C198	DVFS general purpose bits enables (DVFSC_DVFSGPBT)	32	R/W	0000_0000h	27.10.7/ 1398
20D_C19C	DVFS EMAC settings (DVFSC_DVFSEMAC)	32	R/W	0000_0004h	27.10.8/ 1400
20D_C1A0	DVFS Control (DVFSC_CNTR)	32	R/W	0900_000Eh	27.10.9/ 1402
20D_C1A4	DVFS Load Tracking Register 0, portion 0 (DVFSC_DVFSLTR0_0)	32	R	0000_0000h	27.10.10/ 1405
20D_C1A8	DVFS Load Tracking Register 0, portion 1 (DVFSC_DVFSLTR0_1)	32	R	0000_0000h	27.10.11/ 1406
20D_C1AC	DVFS Load Tracking Register 1, portion 0 (DVFSC_DVFSLTR1_0)	32	R	0000_0000h	27.10.12/ 1406
20D_C1B0	DVFS Load Tracking Register 3, portion 1 (DVFSC_DVFSLTR1_1)	32	R	0000_0000h	27.10.13/ 1407
20D_C1B4	DVFS pattern 0 length (DVFSC_DVFSPT0)	32	R/W	0000_0010h	27.10.14/ 1408
20D_C1B8	DVFS pattern 1 length (DVFSC_DVFSPT1)	32	R/W	0000_0010h	27.10.15/ 1408
20D_C1BC	DVFS pattern 2 length (DVFSC_DVFSPT2)	32	R/W	0000_0010h	27.10.16/ 1409
20D_C1C0	DVFS pattern 3 length (DVFSC_DVFSPT3)	32	R/W	0000_0010h	27.10.17/ 1410

27.10.1 DVFS Thresholds (DVFSC_THRS)

Address: 20D_C180h base + 0h offset = 20D_C180h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				UPTHR						DWTHR						0										PNCTHR					
W																																
Reset	0	0	0	0	1	1	1	1	1	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0

DVFSC_THRS field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–22 UPTHR	Upper threshold for load tracking
21–16 DWTHR	Down threshold for load tracking
15–6 Reserved	This read-only field is reserved and always has the value 0.
PNCTHR	Panic threshold for load tracking

27.10.2 DVFS Counters thresholds (DVFSC_COUN)

Address: 20D_C180h base + 4h offset = 20D_C184h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								DN_CNT								0								UPCNT							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

DVFSC_COUN field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 DN_CNT	Down counter threshold value
15–8 Reserved	This read-only field is reserved and always has the value 0.
UPCNT	UP counter threshold value

27.10.3 DVFS general purpose bits weight (DVFSC_SIG1)

Address: 20D_C180h base + 8h offset = 20D_C188h

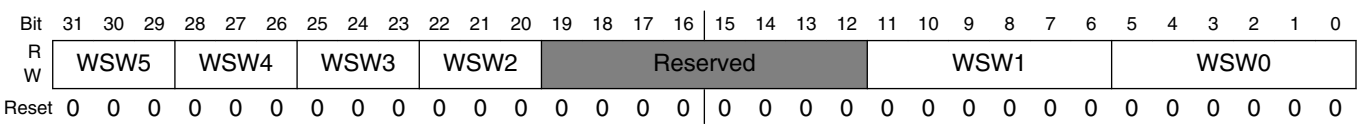
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	WSW15			WSW14			WSW13			WSW12			WSW11			WSW10			WSW9			WSW8			WSW7			WSW6			0	
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_SIG1 field descriptions

Field	Description
31–29 WSW15	General purpose load tracking signal weight dvfs_w_sig[15]
28–26 WSW14	General purpose load tracking signal weight dvfs_w_sig[14]
25–23 WSW13	General purpose load tracking signal weight dvfs_w_sig[13]
22–20 WSW12	General purpose load tracking signal weight dvfs_w_sig[12]
19–17 WSW11	General purpose load tracking signal weight dvfs_w_sig[11]
16–14 WSW10	General purpose load tracking signal weight dvfs_w_sig[10]
13–11 WSW9	General purpose load tracking signal weight dvfs_w_sig[9]
10–8 WSW8	General purpose load tracking signal weight dvfs_w_sig[8]
7–5 WSW7	General purpose load tracking signal weight dvfs_w_sig[7]
4–2 WSW6	General purpose load tracking signal weight dvfs_w_sig[6]
Reserved	This read-only field is reserved and always has the value 0.

27.10.4 DVFS general purpose bits weight (DVFSC_DVFSSIG0)

Address: 20D_C180h base + Ch offset = 20D_C18Ch



DVFSC_DVFSSIG0 field descriptions

Field	Description
31–29 WSW5	General purpose load tracking signal weight dvfs_w_sig[5]
28–26 WSW4	General purpose load tracking signal weight dvfs_w_sig[4]
25–23 WSW3	General purpose load tracking signal weight dvfs_w_sig[3]
22–20 WSW2	General purpose load tracking signal weight dvfs_w_sig[2]
19–12 -	This field is reserved. Reserved

Table continues on the next page...

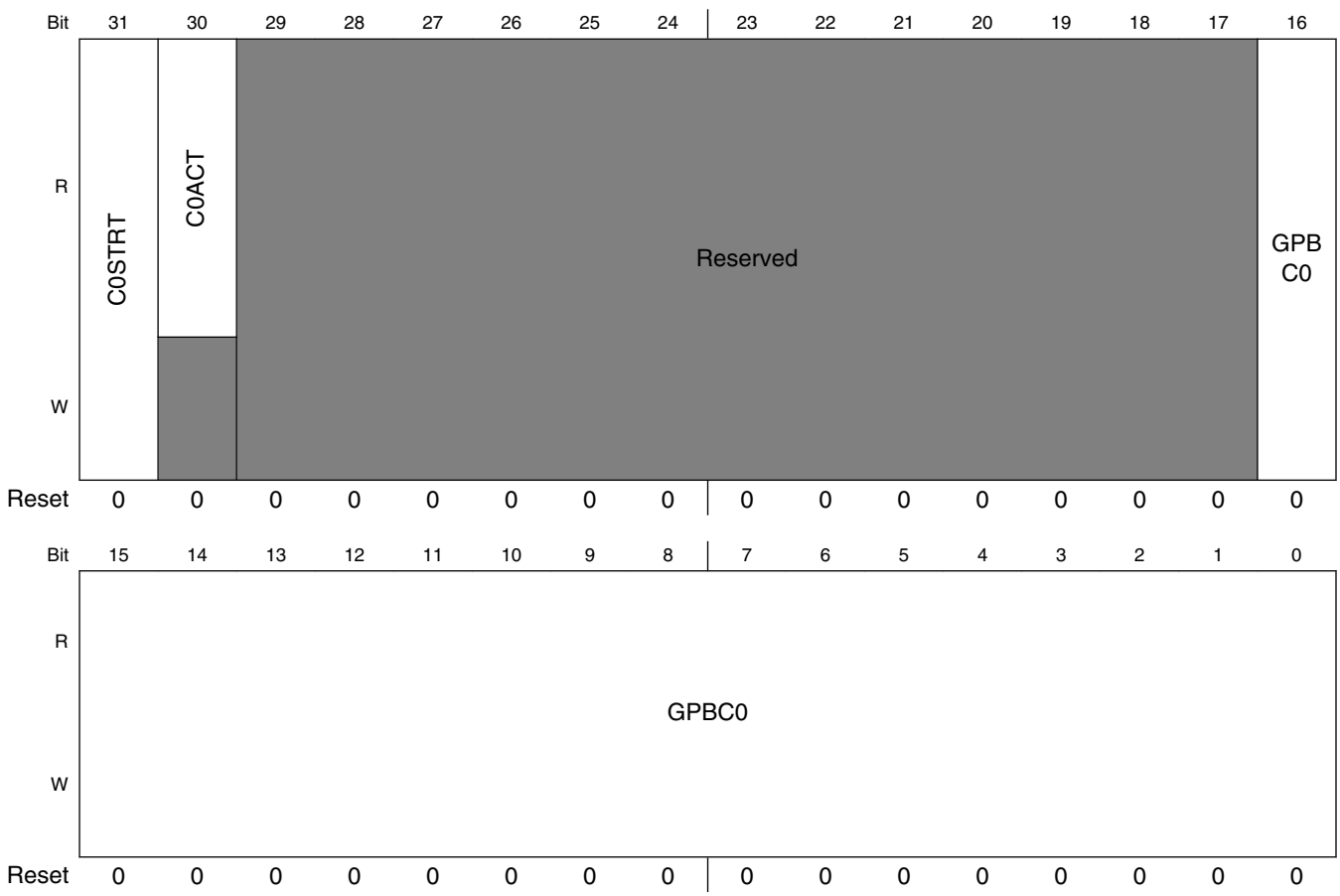
DVFSC_DVFSSIG0 field descriptions (continued)

Field	Description
11–6 WSW1	General purpose load tracking signal weight dvfs_w_sig[1]. This value is relevant during GPC1 counting period or when GPB1 is set.
WSW0	General purpose load tracking signal weight dvfs_w_sig[0]. This value is relevant during GPC0 counting period or when GPB0 is set.

27.10.5 DVFS general purpose bit 0 weight counter (DVFSC_DVFSGPC0)

DVFS general purpose bits weight counter.

Address: 20D_C180h base + 10h offset = 20D_C190h



DVFSC_DVFSGPC0 field descriptions

Field	Description
31 C0STRT	C0STRT - Counter 0 start Setting of this bit will initialize down counting of the GPC0 value.

Table continues on the next page...

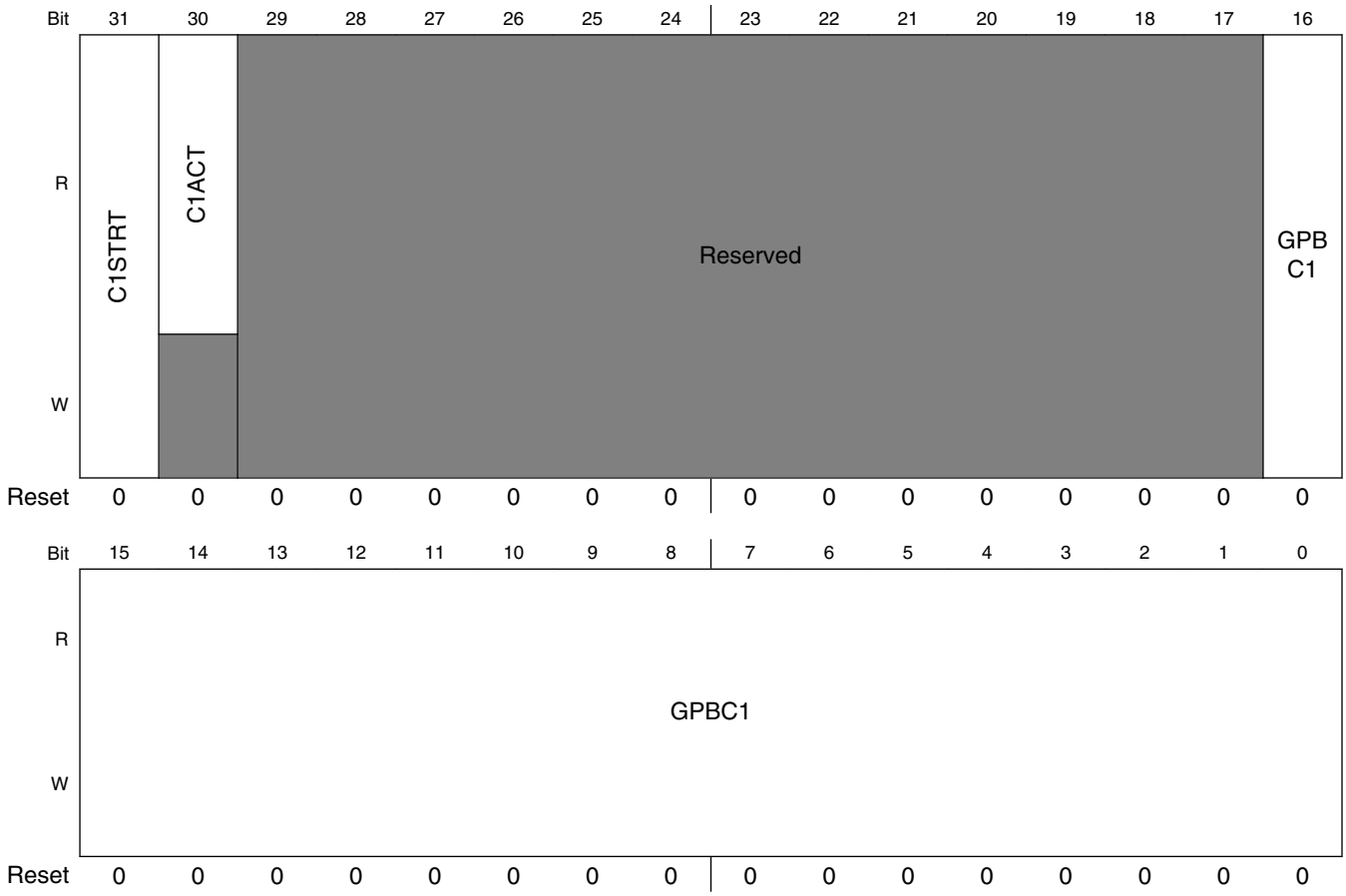
DVFSC_DVFSGPC0 field descriptions (continued)

Field	Description
	Bit is self-cleared next cycle after setting. Any setting of this bit will re-start GPC0 counter to the GPC0 value. GPB0 bit disables (overrides) GPC0 counter - WSW0 weight is applicable continuously
30 C0ACT	C0ACT - Counter 0 active indicator 1 General Purpose bit0 counter didn't reach value of "0" - the WSW0 is provided to DVFS calculation 0 General Purpose bit0 counter reached value of "0" - the instead of WSW0, "0" (zero) is provided to DVFS calculation
29–17 -	This field is reserved. reserved
GPBC0	GPBC0 - General Purpose bits Counter 0 During period of this counter the GeP bit 0 will be set and WSW0 will be added to the calculations.

27.10.6 DVFS general purpose bit 1 weight counter (DVFSC_DVFSGPC1)

DVFS general purpose bits weight counter1.

Address: 20D_C180h base + 14h offset = 20D_C194h



DVFSC_DVFSGPC1 field descriptions

Field	Description
31 C1STRT	C1STRT - Counter 1start Setting of this bit will initialize down counting of the GPC1 value. Bit is self-cleared next cycle after setting. Any setting of this bit will re-start GPC1 counter to the GPC1 value. GPB1 bit disables (overrides) GPC1 counter - WSW1 weight is applicable continuously
30 C1ACT	C1ACT - Counter 1 active indicator

Table continues on the next page...

DVFSC_DVFSGPC1 field descriptions (continued)

Field	Description
	1 General Purpose bit1 counter didn't reach value of "0" - the WSW1 is provided to DVFS calculation 0 General Purpose bit1 counter reached value of "0" - the instead of WSW1, "0" (zero) is provided to DVFS calculation
29–17 -	This field is reserved. reserved
GPBC1	GPBC1 - General Purpose bits Counter 1 During period of this counter the GeP bit 1 will be set and WSW1 will be added to the calculations.

27.10.7 DVFS general purpose bits enables (DVFSC_DVFSGPBT)

Address: 20D_C180h base + 18h offset = 20D_C198h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	GPB15	GPB14	GPB13	GPB12	GPB11	GPB10	GPB9	GPB8	GPB7	GPB6	GPB5	GPB4	GPB3	GPB2	GPB1	GPB0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_DVFSGPBT field descriptions

Field	Description
31–16 -	This field is reserved. reserved
15 GPB15	General purpose bit 15. Its weight is set by WSW15 value.
14 GPB14	General purpose bit 14. Its weight is set by WSW14 value.
13 GPB13	General purpose bit 13. Its weight is set by WSW13 value.
12 GPB12	General purpose bit 12. Its weight is set by WSW12 value.
11 GPB11	General purpose bit 11. Its weight is set by WSW11 value.
10 GPB10	General purpose bit 10. Its weight is set by WSW10 value.

Table continues on the next page...

DVFSC_DVFSGPBT field descriptions (continued)

Field	Description
9 GPB9	General purpose bit 9. Its weight is set by WSW9 value.
8 GPB8	General purpose bit 8. Its weight is set by WSW8 value.
7 GPB7	General purpose bit 7. Its weight is set by WSW7 value.
6 GPB6	General purpose bit 6. Its weight is set by WSW6 value.
5 GPB5	General purpose bit 5. Its weight is set by WSW5 value.
4 GPB4	General purpose bit 4. Its weight is set by WSW4 value.
3 GPB3	General purpose bit 3. Its weight is set by WSW3 value.
2 GPB2	General purpose bit 2. Its weight is set by WSW2 value.
1 GPB1	General purpose bit 1. Its weight is set by WSW1 value. IF set (1), the GPBC1 operation is disregarded, WSW1 value is applied continuously.
0 GPB0	General purpose bit 0. Its weight is set by WSW0 value. IF set (1), the GPBC0 operation is disregarded, WSW0 value is applied continuously.

27.10.8 DVFS EMAC settings (DVFSC_DVFSEMAC)

Address: 20D_C180h base + 1Ch offset = 20D_C19Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved				WFIM3	WFIM2	WFIM1	WFIM0	FSVAI3[1:0]		FSVAI2[1:0]		FSVAI1[1:0]		FSVAI0[1:0]	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved			DVFEN3	DVFEN2	DVFEN1	DVFEN0	EMAC								
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

DVFSC_DVFSEMAC field descriptions

Field	Description
31–28 -	This field is reserved. Reserved
27 WFIM3	DVFS Wait for Interrupt of core 3 mask bit 0 Wait for interrupt of core 3 isn't masked 1 Wait for interrupt of core 3 is masked.
26 WFIM2	DVFS Wait for Interrupt of core 2mask bit 0 Wait for interrupt of core 2 isn't masked 1 Wait for interrupt of core 2 is masked.
25 WFIM1	DVFS Wait for Interrupt of core 1 mask bit 0 Wait for interrupt of core 1 isn't masked 1 Wait for interrupt of core 1 is masked.
24 WFIM0	DVFS Wait for Interrupt of core 0 mask bit 0 Wait for interrupt of core 0 isn't masked 1 Wait for interrupt of core 0 is masked.

Table continues on the next page...

DVFSC_DVFSEMAC field descriptions (continued)

Field	Description
23–22 FSVAI3[1:0]	DVFS Frequency adjustment status of core 3. These status bits indicate that frequency should be changed, following load of core 3. 00 no change 01 frequency should be increased. Low priority source for interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency). 10 frequency should be decreased. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MINF= 1 (lowest frequency). 11 frequency should be increased immediately. High priority source of interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency).
21–20 FSVAI2[1:0]	DVFS Frequency adjustment status of core 2. These status bits indicate that frequency should be changed, following load of core 2. 00 no change 01 frequency should be increased. Low priority source for interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency). 10 frequency should be decreased. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MINF= 1 (lowest frequency). 11 frequency should be increased immediately. High priority source of interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency).
19–18 FSVAI1[1:0]	DVFS Frequency adjustment status of core 1. These status bits indicate that frequency should be changed, following load of core 1. 00 no change 01 frequency should be increased. Low priority source for interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency). 10 frequency should be decreased. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MINF= 1 (lowest frequency). 11 frequency should be increased immediately. High priority source of interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency).
17–16 FSVAI0[1:0]	DVFS Frequency adjustment status of core 0. These status bits indicate that frequency should be changed, following load of core 0. 00 no change 01 frequency should be increased. Low priority source for interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency). 10 frequency should be decreased. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MINF= 1 (lowest frequency). 11 frequency should be increased immediately. High priority source of interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency).
15–13 -	This field is reserved. Reserved
12 DVFEN3	DVFS tracking for core3 enable. 1 DVFS enabled. 0 DVFS disabled.
11 DVFEN2	DVFS tracking for core2 enable. 1 DVFS enabled. 0 DVFS disabled.

Table continues on the next page...

DVFSC_DVFSEMAC field descriptions (continued)

Field	Description
10 DVFEN1	DVFS tracking for core1 enable. 1 DVFS enabled. 0 DVFS disabled.
9 DVFEN0	DVFS tracking for core0 enable. 1 DVFS enabled. 0 DVFS disabled.
EMAC	EMAC - EMA control value

27.10.9 DVFS Control (DVFSC_CNTR)

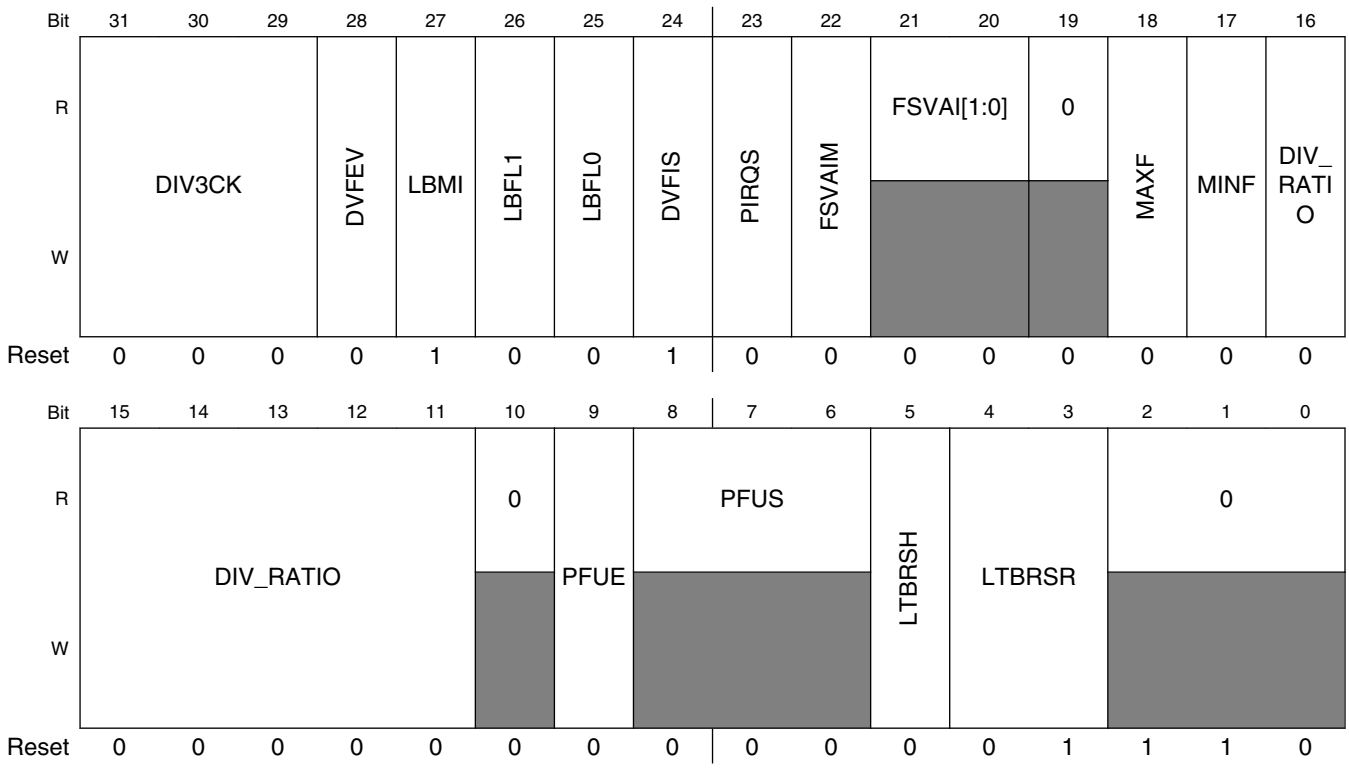
Table 27-35. DIV3CK division

DIV3CK setting	dividing ratio	sum_3 passing bits	div_1_clk cumulative divider
00	1	4-0	1*512=512
001	4	6-2	4*512=2048
010	16	8-4	16*512=8192
011	64	10-6	64*512=32768
100	256	12-8	256*512=131072
101	1024	16-10	1024*512=524288

Table 27-36. Preliminary Divider definition

DIV_RATIO value	ARM clk division ratio
000000	1
000001	2
000010	3
...	...

Address: 20D_C180h base + 20h offset = 20D_C1A0h



DVFSC_CNTR field descriptions

Field	Description
31–29 DIV3CK	DIV3CK - div_3_clk division ratio inside the DVFS module. According to the Table 27-35
28 DVFEV	Always give a DVFS event. 0 Do not give an event always. 1 Always give event.
27 LBMI	Load buffer full mask interrupt. This bit masks the generation of this interrupt. Load buffer full interrupt is masked (LBFL0 and LBFL1 bits still will be updated, but interrupt won't be generated) Load buffer full interrupt is enabled.
26 LBFL1	Load buffer 1 - full status bit. This bit indicates that log buffer registers are full. The bit is set to 1 automatically. An interrupt will be generated if LBMI bit is set to "0" Write '1' to clear. (write '0' leaves bit unchanged) 1 Load buffer0 is full. 0 Load buffer0 is not full.
25 LBFL0	Load buffer 0 - full status bit. This bit indicates that log buffer registers are full. The bit is set to 1 automatically. An interrupt will be generated if LBMI bit is set to "0" Write '1' to clear. (write '0' leaves bit unchanged)

Table continues on the next page...

DVFSC_CNTR field descriptions (continued)

Field	Description
	1 Load buffer1 is full. 0 Load buffer1 is not full.
24 DVFIS	DVFS Interrupt select. These bits define destination of DVFS interrupts. 1 MCU interrupt will be generated for DVFS events. 0 SDMA interrupt will be generated for DVFS events.
23 PIRQS	PIRQS - Pattern IRQ Source * write '1' to clear. Writing '1' will clear interrupt if interrupt was from pattern 1 DVFS IRQ source was from pattern 0 DVFS IRQ source was not from pattern
22 FSVAIM	DVFS Frequency adjustment interrupt mask. This bit masks the DVFS frequency adjustment interrupt. FSVAI status bits will be still asserted in relevant cases. 1 interrupt is masked. 0 interrupt is enabled.
21–20 FSVAI[1:0]	FSVAI DVFS Frequency adjustment interrupt. These status bits indicate that the system frequency should be changed. 00 no interrupt 01 frequency should be increased. Low priority interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency). 10 frequency should be decreased. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MINF= 1 (lowest frequency). 11 frequency should be increased immediately. High priority interrupt. Interrupt is asserted, if FSVAIM=0. Interrupt is masked if MAXF = 1 (highest frequency).
19 Reserved	This read-only field is reserved and always has the value 0.
18 MAXF	Maximum frequency reached. Interrupt will not be created in maximum frequency reached and frequency increase required. 1 max frequency reached 0 max frequency not reached
17 MINF	Minimum frequency reached. Interrupt will not be created in minimum frequency reached and frequency decrease required. 1 min frequency reached 0 min frequency not reached
16–11 DIV_RATIO	DIV_RATIO - Divider value. Divider divides the input ARM clock, following the table Table 27-36
10 Reserved	This read-only field is reserved and always has the value 0.
9 PFUE	PFUE - Period Frequency Update Enable 1 enabled 0 disabled
8–6 PFUS	PFUS - Periodic Frequency Update Status

Table continues on the next page...

DVFSC_CNTR field descriptions (continued)

Field	Description
	000 no update 100 DVFSPT0 period, previous finished(can be performance level decrease) 101 DVFSPT1 period, previous finished(can be EMA-detected performance level) 110 DVFSPT2 period, previous finished(can be performance level increase) 111 DVFSPT3 period, previous finished (can be EMA-detected performance level)
5 LTBRSH	LTBRSH - Load Tracking Buffer Register Shift: 0 values of [5:2] of the selected input are saving in Load Tracking Buffer 1 values of [4:1] of the selected input are saving in Load Tracking Buffer
4–3 LTBRSR	LTBRSR - Load Tracking Buffer Register Source: 00 pre_ld_add 01 ld_add 10 ema_ld 11 reserved
Reserved	This read-only field is reserved and always has the value 0.

27.10.10 DVFS Load Tracking Register 0, portion 0 (DVFSC_DVFSLTR0_0)

Address: 20D_C180h base + 24h offset = 20D_C1A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LTS0_7				LTS0_6				LTS0_5				LTS0_4				LTS0_3				LTS0_2				LTS0_1				LTS0_0			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_DVFSLTR0_0 field descriptions

Field	Description
31–28 LTS0_7	core 0 Load Tracking Sample 7
27–24 LTS0_6	core 0 Load Tracking Sample 6
23–20 LTS0_5	core 0 Load Tracking Sample 5
19–16 LTS0_4	core 0 Load Tracking Sample 4
15–12 LTS0_3	core 0 Load Tracking Sample 3
11–8 LTS0_2	core 0 Load Tracking Sample 2
7–4 LTS0_1	core 0 Load Tracking Sample 1
LTS0_0	core 0 Load Tracking Sample 0

27.10.11 DVFS Load Tracking Register 0, portion 1 (DVFSC_DVFSLTR0_1)

Address: 20D_C180h base + 28h offset = 20D_C1A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LTS0_15				LTS0_14				LTS0_13				LTS0_12				LTS0_11				LTS0_10				LTS0_9				LTS0_8			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_DVFSLTR0_1 field descriptions

Field	Description
31–28 LTS0_15	core 0 Load Tracking Sample 15
27–24 LTS0_14	core 0 Load Tracking Sample 14
23–20 LTS0_13	core 0 Load Tracking Sample 13
19–16 LTS0_12	core 0 Load Tracking Sample 12
15–12 LTS0_11	core 0 Load Tracking Sample 11
11–8 LTS0_10	core 0 Load Tracking Sample 10
7–4 LTS0_9	core 0 Load Tracking Sample 9
LTS0_8	core 0 Load Tracking Sample 8

27.10.12 DVFS Load Tracking Register 1, portion 0 (DVFSC_DVFSLTR1_0)

Address: 20D_C180h base + 2Ch offset = 20D_C1ACh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LTS1_7				LTS1_6				LTS1_5				LTS1_4				LTS1_3				LTS1_2				LTS1_1				LTS1_0			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_DVFSLTR1_0 field descriptions

Field	Description
31–28 LTS1_7	core 0 Load Tracking Sample 7

Table continues on the next page...

DVFSC_DVFSLTR1_0 field descriptions (continued)

Field	Description
27–24 LTS1_6	core 0 Load Tracking Sample 6
23–20 LTS1_5	core 0 Load Tracking Sample 5
19–16 LTS1_4	core 0 Load Tracking Sample 4
15–12 LTS1_3	core 0 Load Tracking Sample 3
11–8 LTS1_2	core 0 Load Tracking Sample 2
7–4 LTS1_1	core 0 Load Tracking Sample 1
LTS1_0	core 0 Load Tracking Sample 0

27.10.13 DVFS Load Tracking Register 3, portion 1 (DVFSC_DVFSLTR1_1)

Address: 20D_C180h base + 30h offset = 20D_C1B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LTS1_15				LTS1_14				LTS1_13				LTS1_12				LTS1_11				LTS1_10				LTS1_9				LTS1_8			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DVFSC_DVFSLTR1_1 field descriptions

Field	Description
31–28 LTS1_15	core 0 Load Tracking Sample 15
27–24 LTS1_14	core 0 Load Tracking Sample 14
23–20 LTS1_13	core 0 Load Tracking Sample 13
19–16 LTS1_12	core 0 Load Tracking Sample 12
15–12 LTS1_11	core 0 Load Tracking Sample 11
11–8 LTS1_10	core 0 Load Tracking Sample 10
7–4 LTS1_9	core 0 Load Tracking Sample 9
LTS1_8	core 0 Load Tracking Sample 8

27.10.14 DVFS pattern 0 length (DVFSC_DVFSPT0)

Address: 20D_C180h base + 34h offset = 20D_C1B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved														PT0A	FPTN0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	FPTN0															
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

DVFSC_DVFSPT0 field descriptions

Field	Description
31–18 -	This field is reserved. reserved
17 PT0A	PT0A - Pattern 0 currently active (read-only) 1 active 0 non-active
FPTN0	FPTN0 - Frequency pattern 0 counter During period of this counter the frequency will be reduced from the EMA-detected level.

27.10.15 DVFS pattern 1 length (DVFSC_DVFSPT1)

Address: 20D_C180h base + 38h offset = 20D_C1B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved														PT1A	FPTN1
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

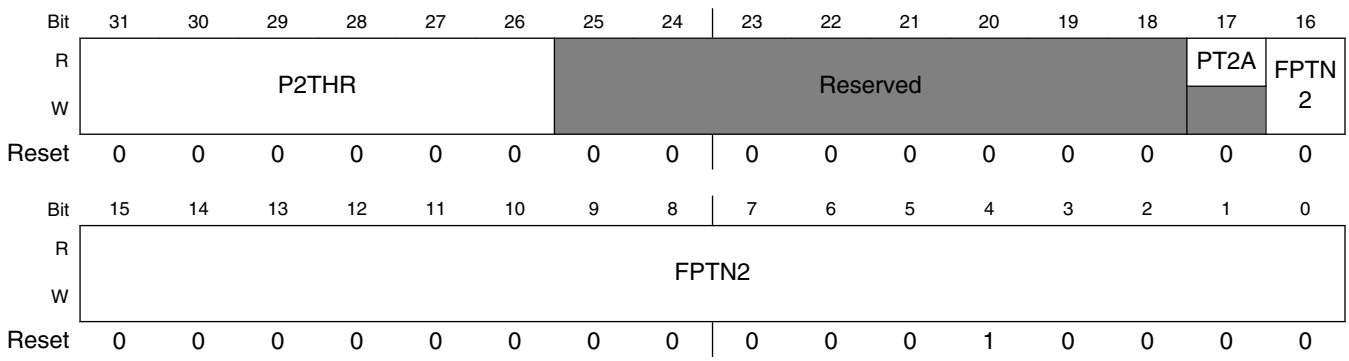
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	FPTN1															
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

DVFSC_DVFSPT1 field descriptions

Field	Description
31–18 -	This field is reserved. reserved
17 PT1A	PT1A - Pattern 1 currently active (read-only) 1 active 0 non-active
FPTN1	FPTN1 - Frequency pattern 1 counter During period of this counter the frequency will be set to the EMA-detected level.

27.10.16 DVFS pattern 2 length (DVFSC_DVFSPT2)

Address: 20D_C180h base + 3Ch offset = 20D_C1BCh



DVFSC_DVFSPT2 field descriptions

Field	Description
31–26 P2THR	P2THR - Pattern 2 Threshold Threshold of current DVFS load (after EMA), for generating interrupts with PFUS indicators 110, 111. If the current performance is greater than the P2THR value, the interrupts will be generated. Otherwise, pattern delay will be counted, but without interrupt generation.
25–18 -	This field is reserved. reserved
17 PT2A	PT2A - Pattern 2 currently active (read-only) 1 active 0 non-active
FPTN2	FPTN2 - Frequency pattern 2 counter During period of this counter the frequency will be increased to higher, than detected by the EMA-detected level.

27.10.17 DVFS pattern 3 length (DVFSC_DVFSPT3)

Address: 20D_C180h base + 40h offset = 20D_C1C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved														PT3A	FPTN 3
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	FPTN3															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

DVFSC_DVFSPT3 field descriptions

Field	Description
31–18 -	This field is reserved. reserved
17 PT3A	PT3A - Pattern 3 currently active (read-only) 1 active 0 non-active
FPTN3	FPTN3 - Frequency pattern 3 counter During period of this counter the frequency will be set to the EMA-detected level.

Chapter 28
General Purpose Input/Output (GPIO)

28.1 Overview

The GPIO general-purpose input/output peripheral provides dedicated general-purpose pins that can be configured as either inputs or outputs.

When configured as an output, it is possible to write to an internal register to control the state driven on the output pin. When configured as an input, it is possible to detect the state of the input by reading the state of an internal register. In addition, the GPIO peripheral can produce CORE interrupts.

The GPIO is one of the blocks controlling the IOMUX of the chip.

Figure 28-1 shows the chip multiplexing scheme.

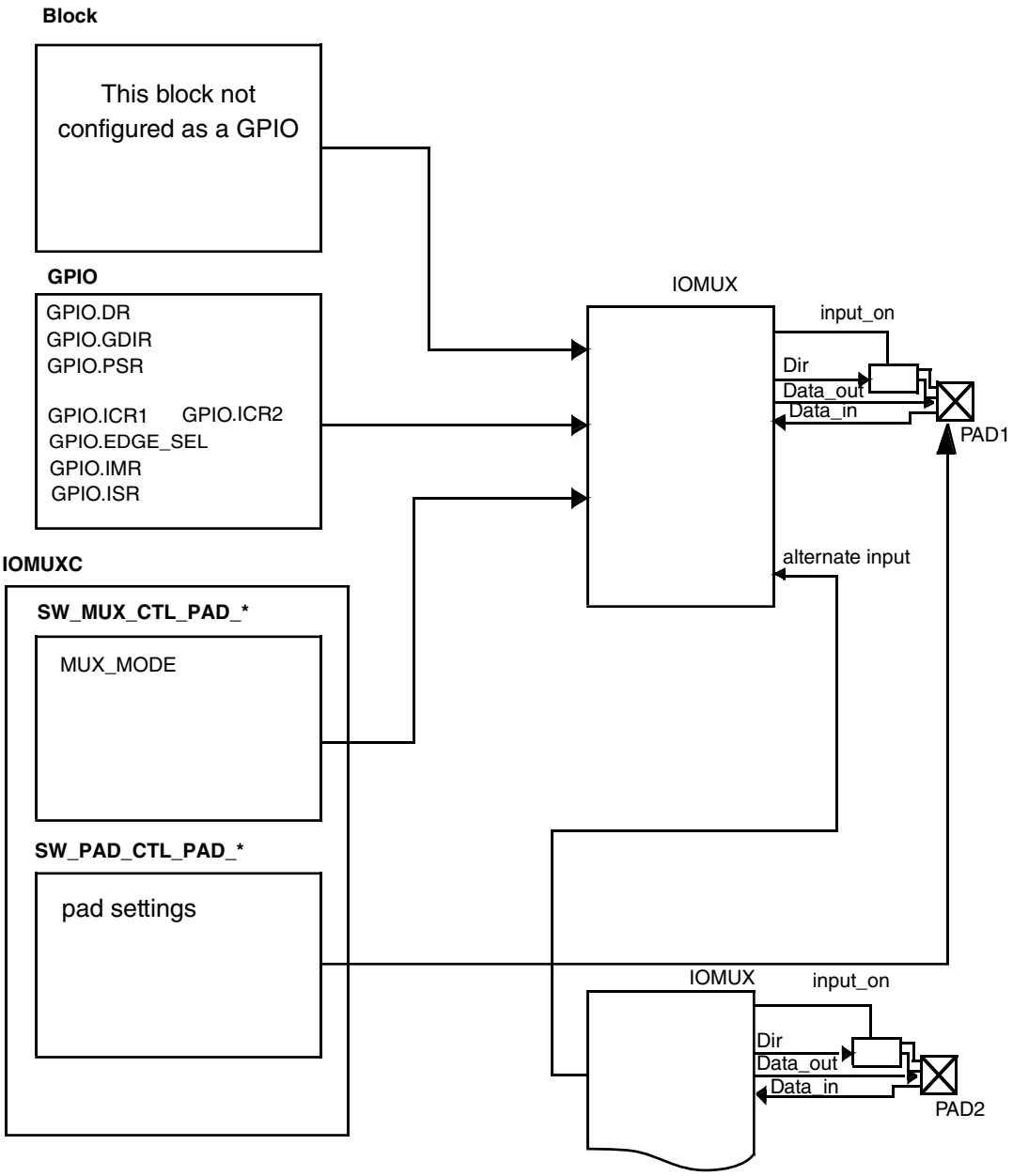


Figure 28-1. Chip IOMUX Scheme

The GPIO functionality is provided through eight registers, an edge-detect circuit, and interrupt generation logic.

The eight registers are:

- Data register (GPIO_DR)
- GPIO direction register (GPIO_GDIR)
- Pad sample register (GPIO_PSR)
- Interrupt control registers (GPIO_ICR1, GPIO_ICR2)

- Edge select register (GPIO_EDGE_SEL)
- Interrupt mask register (GPIO_IMR)
- Interrupt status register (GPIO_ISR)

These registers are described in detail in GPIO Memory Map/Register Definition.

Each GPIO input has a dedicated edge-detect circuit which can be configured through software to detect rising edges, falling edges, logic low-levels or logic high-levels on the input signals. The outputs of the edge detect circuits are optionally masked by setting the corresponding bit in the interrupt mask register (GPIO_IMR). These qualified outputs are OR'ed together to generate two one-bit interrupt lines:

- Combined interrupt indication for GPIOx signals 0 - 15
- Combined interrupt indication for GPIOx signals 16 - 31

In addition, GPIO1 provides visibility to each of its 8 low order interrupt sources (i.e. GPIO1 interrupt n, for n = 0 – 7). However, individual interrupt indications from other GPIOx are not available.

The GPIO edge detection is described further in Interrupt Control Unit.

The GPIO's overall functionality is described further in GPIO Functional Description.

28.1.1 Block Diagram

The GPIO subsystem contains 8 GPIO blocks which can generate and control up to 32 signals for general purpose.

A block diagram of the GPIO is shown in Figure 28-2.

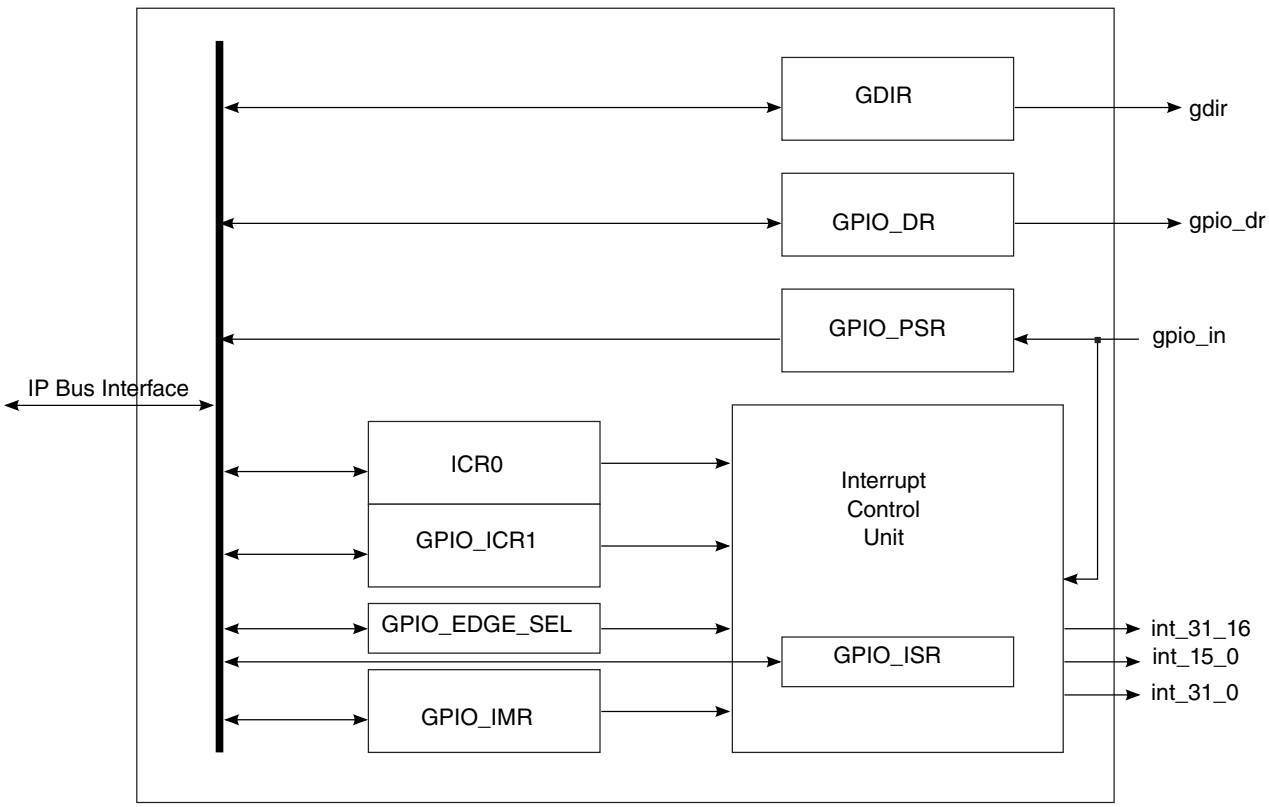


Figure 28-2. GPIO Block Diagram

28.1.2 Features

The GPIO includes the following features:

- General purpose input/output logic capabilities:
  - Drives specific data to output using the data register (GPIO_DR)
  - Controls the direction of the signal using the GPIO direction register (GPIO_GDIR)
  - Enables the core to sample the status of the corresponding inputs by reading the pad sample register (GPIO_PSR).
- GPIO interrupt capabilities:
  - Supports up to 32 interrupts

- Identifies interrupt edges
- Generates three active-high interrupts to the SoC interrupt controller

28.2 External Signals

The tables found here describe the external signals of GPIO.

Table 28-1. GPIO1 External Signals

Signal	Description	Pad	Mode	Direction
GPIO1_IO00	-	GPIO_0	ALT5	IO
GPIO1_IO01	-	GPIO_1	ALT5	IO
GPIO1_IO02	-	GPIO_2	ALT5	IO
GPIO1_IO03	-	GPIO_3	ALT5	IO
GPIO1_IO04	-	GPIO_4	ALT5	IO
GPIO1_IO05	-	GPIO_5	ALT5	IO
GPIO1_IO06	-	GPIO_6	ALT5	IO
GPIO1_IO07	-	GPIO_7	ALT5	IO
GPIO1_IO08	-	GPIO_8	ALT5	IO
GPIO1_IO09	-	GPIO_9	ALT5	IO
GPIO1_IO10	-	SD2_CLK	ALT5	IO
GPIO1_IO11	-	SD2_CMD	ALT5	IO
GPIO1_IO12	-	SD2_DAT3	ALT5	IO
GPIO1_IO13	-	SD2_DAT2	ALT5	IO
GPIO1_IO14	-	SD2_DAT1	ALT5	IO
GPIO1_IO15	-	SD2_DAT0	ALT5	IO
GPIO1_IO16	-	SD1_DAT0	ALT5	IO
GPIO1_IO17	-	SD1_DAT1	ALT5	IO
GPIO1_IO18	-	SD1_CMD	ALT5	IO
GPIO1_IO19	-	SD1_DAT2	ALT5	IO
GPIO1_IO20	-	SD1_CLK	ALT5	IO
GPIO1_IO21	-	SD1_DAT3	ALT5	IO
GPIO1_IO22	-	ENET_MDIO	ALT5	IO
GPIO1_IO23	-	ENET_REF_CLK	ALT5	IO
GPIO1_IO24	-	ENET_RX_ER	ALT5	IO
GPIO1_IO25	-	ENET_CRS_DV	ALT5	IO
GPIO1_IO26	-	ENET_RXD1	ALT5	IO
GPIO1_IO27	-	ENET_RXD0	ALT5	IO
GPIO1_IO28	-	ENET_TX_EN	ALT5	IO
GPIO1_IO29	-	ENET_TXD1	ALT5	IO
GPIO1_IO30	-	ENET_TXD0	ALT5	IO
GPIO1_IO31	-	ENET_MDC	ALT5	IO

Table 28-2. GPIO2 External Signals

Signal	Description	Pad	Mode	Direction
GPIO2_IO00	-	NANDF_D0	ALT5	IO
GPIO2_IO01	-	NANDF_D1	ALT5	IO
GPIO2_IO02	-	NANDF_D2	ALT5	IO
GPIO2_IO03	-	NANDF_D3	ALT5	IO
GPIO2_IO04	-	NANDF_D4	ALT5	IO
GPIO2_IO05	-	NANDF_D5	ALT5	IO
GPIO2_IO06	-	NANDF_D6	ALT5	IO
GPIO2_IO07	-	NANDF_D7	ALT5	IO
GPIO2_IO08	-	SD4_DAT0	ALT5	IO
GPIO2_IO09	-	SD4_DAT1	ALT5	IO
GPIO2_IO10	-	SD4_DAT2	ALT5	IO
GPIO2_IO11	-	SD4_DAT3	ALT5	IO
GPIO2_IO12	-	SD4_DAT4	ALT5	IO
GPIO2_IO13	-	SD4_DAT5	ALT5	IO
GPIO2_IO14	-	SD4_DAT6	ALT5	IO
GPIO2_IO15	-	SD4_DAT7	ALT5	IO
GPIO2_IO16	-	EIM_A22	ALT5	IO
GPIO2_IO17	-	EIM_A21	ALT5	IO
GPIO2_IO18	-	EIM_A20	ALT5	IO
GPIO2_IO19	-	EIM_A19	ALT5	IO
GPIO2_IO20	-	EIM_A18	ALT5	IO
GPIO2_IO21	-	EIM_A17	ALT5	IO
GPIO2_IO22	-	EIM_A16	ALT5	IO
GPIO2_IO23	-	EIM_CS0	ALT5	IO
GPIO2_IO24	-	EIM_CS1	ALT5	IO
GPIO2_IO25	-	EIM_OE	ALT5	IO
GPIO2_IO26	-	EIM_RW	ALT5	IO
GPIO2_IO27	-	EIM_LBA	ALT5	IO
GPIO2_IO28	-	EIM_EB0	ALT5	IO
GPIO2_IO29	-	EIM_EB1	ALT5	IO
GPIO2_IO30	-	EIM_EB2	ALT5	IO
GPIO2_IO31	-	EIM_EB3	ALT5	IO

Table 28-3. GPIO3 External Signals

Signal	Description	Pad	Mode	Direction
GPIO3_IO00	-	EIM_DA0	ALT5	IO
GPIO3_IO01	-	EIM_DA1	ALT5	IO
GPIO3_IO02	-	EIM_DA2	ALT5	IO

Table continues on the next page...

Table 28-3. GPIO3 External Signals (continued)

Signal	Description	Pad	Mode	Direction
GPIO3_IO03	-	EIM_DA3	ALT5	IO
GPIO3_IO04	-	EIM_DA4	ALT5	IO
GPIO3_IO05	-	EIM_DA5	ALT5	IO
GPIO3_IO06	-	EIM_DA6	ALT5	IO
GPIO3_IO07	-	EIM_DA7	ALT5	IO
GPIO3_IO08	-	EIM_DA8	ALT5	IO
GPIO3_IO09	-	EIM_DA9	ALT5	IO
GPIO3_IO10	-	EIM_DA10	ALT5	IO
GPIO3_IO11	-	EIM_DA11	ALT5	IO
GPIO3_IO12	-	EIM_DA12	ALT5	IO
GPIO3_IO13	-	EIM_DA13	ALT5	IO
GPIO3_IO14	-	EIM_DA14	ALT5	IO
GPIO3_IO15	-	EIM_DA15	ALT5	IO
GPIO3_IO16	-	EIM_D16	ALT5	IO
GPIO3_IO17	-	EIM_D17	ALT5	IO
GPIO3_IO18	-	EIM_D18	ALT5	IO
GPIO3_IO19	-	EIM_D19	ALT5	IO
GPIO3_IO20	-	EIM_D20	ALT5	IO
GPIO3_IO21	-	EIM_D21	ALT5	IO
GPIO3_IO22	-	EIM_D22	ALT5	IO
GPIO3_IO23	-	EIM_D23	ALT5	IO
GPIO3_IO24	-	EIM_D24	ALT5	IO
GPIO3_IO25	-	EIM_D25	ALT5	IO
GPIO3_IO26	-	EIM_D26	ALT5	IO
GPIO3_IO27	-	EIM_D27	ALT5	IO
GPIO3_IO28	-	EIM_D28	ALT5	IO
GPIO3_IO29	-	EIM_D29	ALT5	IO
GPIO3_IO30	-	EIM_D30	ALT5	IO
GPIO3_IO31	-	EIM_D31	ALT5	IO

Table 28-4. GPIO4 External Signals

Signal	Description	Pad	Mode	Direction
GPIO4_IO05	-	GPIO_19	ALT5	IO
GPIO4_IO06	-	KEY_COL0	ALT5	IO
GPIO4_IO07	-	KEY_ROW0	ALT5	IO
GPIO4_IO08	-	KEY_COL1	ALT5	IO
GPIO4_IO09	-	KEY_ROW1	ALT5	IO
GPIO4_IO10	-	KEY_COL2	ALT5	IO

Table continues on the next page...

Table 28-4. GPIO4 External Signals (continued)

Signal	Description	Pad	Mode	Direction
GPIO4_IO11	-	KEY_ROW2	ALT5	IO
GPIO4_IO12	-	KEY_COL3	ALT5	IO
GPIO4_IO13	-	KEY_ROW3	ALT5	IO
GPIO4_IO14	-	KEY_COL4	ALT5	IO
GPIO4_IO15	-	KEY_ROW4	ALT5	IO
GPIO4_IO16	-	DI0_DISP_CLK	ALT5	IO
GPIO4_IO17	-	DI0_PIN15	ALT5	IO
GPIO4_IO18	-	DI0_PIN2	ALT5	IO
GPIO4_IO19	-	DI0_PIN3	ALT5	IO
GPIO4_IO20	-	DI0_PIN4	ALT5	IO
GPIO4_IO21	-	DISP0_DAT0	ALT5	IO
GPIO4_IO22	-	DISP0_DAT1	ALT5	IO
GPIO4_IO23	-	DISP0_DAT2	ALT5	IO
GPIO4_IO24	-	DISP0_DAT3	ALT5	IO
GPIO4_IO25	-	DISP0_DAT4	ALT5	IO
GPIO4_IO26	-	DISP0_DAT5	ALT5	IO
GPIO4_IO27	-	DISP0_DAT6	ALT5	IO
GPIO4_IO28	-	DISP0_DAT7	ALT5	IO
GPIO4_IO29	-	DISP0_DAT8	ALT5	IO
GPIO4_IO30	-	DISP0_DAT9	ALT5	IO
GPIO4_IO31	-	DISP0_DAT10	ALT5	IO

Table 28-5. GPIO5 External Signals

Signal	Description	Pad	Mode	Direction
GPIO5_IO00	-	EIM_WAIT	ALT5	IO
GPIO5_IO02	-	EIM_A25	ALT5	IO
GPIO5_IO04	-	EIM_A24	ALT5	IO
GPIO5_IO05	-	DISP0_DAT11	ALT5	IO
GPIO5_IO06	-	DISP0_DAT12	ALT5	IO
GPIO5_IO07	-	DISP0_DAT13	ALT5	IO
GPIO5_IO08	-	DISP0_DAT14	ALT5	IO
GPIO5_IO09	-	DISP0_DAT15	ALT5	IO
GPIO5_IO10	-	DISP0_DAT16	ALT5	IO
GPIO5_IO11	-	DISP0_DAT17	ALT5	IO
GPIO5_IO12	-	DISP0_DAT18	ALT5	IO
GPIO5_IO13	-	DISP0_DAT19	ALT5	IO
GPIO5_IO14	-	DISP0_DAT20	ALT5	IO
GPIO5_IO15	-	DISP0_DAT21	ALT5	IO

Table continues on the next page...

Table 28-5. GPIO5 External Signals (continued)

Signal	Description	Pad	Mode	Direction
GPIO5_IO16	-	DISP0_DAT22	ALT5	IO
GPIO5_IO17	-	DISP0_DAT23	ALT5	IO
GPIO5_IO18	-	CSI0_PIXCLK	ALT5	IO
GPIO5_IO19	-	CSI0_MCLK	ALT5	IO
GPIO5_IO20	-	CSI0_DATA_EN	ALT5	IO
GPIO5_IO21	-	CSI0_VSYNC	ALT5	IO
GPIO5_IO22	-	CSI0_DAT4	ALT5	IO
GPIO5_IO23	-	CSI0_DAT5	ALT5	IO
GPIO5_IO24	-	CSI0_DAT6	ALT5	IO
GPIO5_IO25	-	CSI0_DAT7	ALT5	IO
GPIO5_IO26	-	CSI0_DAT8	ALT5	IO
GPIO5_IO27	-	CSI0_DAT9	ALT5	IO
GPIO5_IO28	-	CSI0_DAT10	ALT5	IO
GPIO5_IO29	-	CSI0_DAT11	ALT5	IO
GPIO5_IO30	-	CSI0_DAT12	ALT5	IO
GPIO5_IO31	-	CSI0_DAT13	ALT5	IO

Table 28-6. GPIO6 External Signals

Signal	Description	Pad	Mode	Direction
GPIO6_IO00	-	CSI0_DAT14	ALT5	IO
GPIO6_IO01	-	CSI0_DAT15	ALT5	IO
GPIO6_IO02	-	CSI0_DAT16	ALT5	IO
GPIO6_IO03	-	CSI0_DAT17	ALT5	IO
GPIO6_IO04	-	CSI0_DAT18	ALT5	IO
GPIO6_IO05	-	CSI0_DAT19	ALT5	IO
GPIO6_IO06	-	EIM_A23	ALT5	IO
GPIO6_IO07	-	NANDF_CLE	ALT5	IO
GPIO6_IO08	-	NANDF_ALE	ALT5	IO
GPIO6_IO09	-	NANDF_WP_B	ALT5	IO
GPIO6_IO10	-	NANDF_RB0	ALT5	IO
GPIO6_IO11	-	NANDF_CS0	ALT5	IO
GPIO6_IO14	-	NANDF_CS1	ALT5	IO
GPIO6_IO15	-	NANDF_CS2	ALT5	IO
GPIO6_IO16	-	NANDF_CS3	ALT5	IO
GPIO6_IO17	-	SD3_DAT7	ALT5	IO
GPIO6_IO18	-	SD3_DAT6	ALT5	IO
GPIO6_IO19	-	RGMII_TXC	ALT5	IO
GPIO6_IO20	-	RGMII_TD0	ALT5	IO

Table continues on the next page...

Table 28-6. GPIO6 External Signals (continued)

Signal	Description	Pad	Mode	Direction
GPIO6_IO21	-	RGMII_TD1	ALT5	IO
GPIO6_IO22	-	RGMII_TD2	ALT5	IO
GPIO6_IO23	-	RGMII_TD3	ALT5	IO
GPIO6_IO24	-	RGMII_RX_CTL	ALT5	IO
GPIO6_IO25	-	RGMII_RD0	ALT5	IO
GPIO6_IO26	-	RGMII_TX_CTL	ALT5	IO
GPIO6_IO27	-	RGMII_RD1	ALT5	IO
GPIO6_IO28	-	RGMII_RD2	ALT5	IO
GPIO6_IO29	-	RGMII_RD3	ALT5	IO
GPIO6_IO30	-	RGMII_RXC	ALT5	IO
GPIO6_IO31	-	EIM_BCLK	ALT5	IO

Table 28-7. GPIO7 External Signals

Signal	Description	Pad	Mode	Direction
GPIO7_IO00	-	SD3_DAT5	ALT5	IO
GPIO7_IO01	-	SD3_DAT4	ALT5	IO
GPIO7_IO02	-	SD3_CMD	ALT5	IO
GPIO7_IO03	-	SD3_CLK	ALT5	IO
GPIO7_IO04	-	SD3_DAT0	ALT5	IO
GPIO7_IO05	-	SD3_DAT1	ALT5	IO
GPIO7_IO06	-	SD3_DAT2	ALT5	IO
GPIO7_IO07	-	SD3_DAT3	ALT5	IO
GPIO7_IO08	-	SD3_RST	ALT5	IO
GPIO7_IO09	-	SD4_CMD	ALT5	IO
GPIO7_IO10	-	SD4_CLK	ALT5	IO
GPIO7_IO11	-	GPIO_16	ALT5	IO
GPIO7_IO12	-	GPIO_17	ALT5	IO
GPIO7_IO13	-	GPIO_18	ALT5	IO

28.3 Clocks

The table found here describes the clock sources for GPIO.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 28-8. GPIO Clocks

Clock name	Clock Root	Description
ipg_clk_s	ipg_clk_root	Peripheral access clock

28.4 GPIO Functional Description

This section provides a complete functional description of the block.

28.4.1 GPIO Function

A GPIO signal can operate as a general-purpose input/output when the IOMUX is set to GPIO mode. Each GPIO signal may be independently configured as either an input or an output using the GPIO direction register (GPIO_GDIR).

When configured as an output (GPIO_GDIR bit = 1), the value in the data bit in the GPIO data register (GPIO_DR) is driven on the corresponding GPIO line. When a signal is configured as an input (GPIO_GDIR bit = 0), the state of the input can be read from the corresponding GPIO_PSR bit.

28.4.2 GPIO pad structure

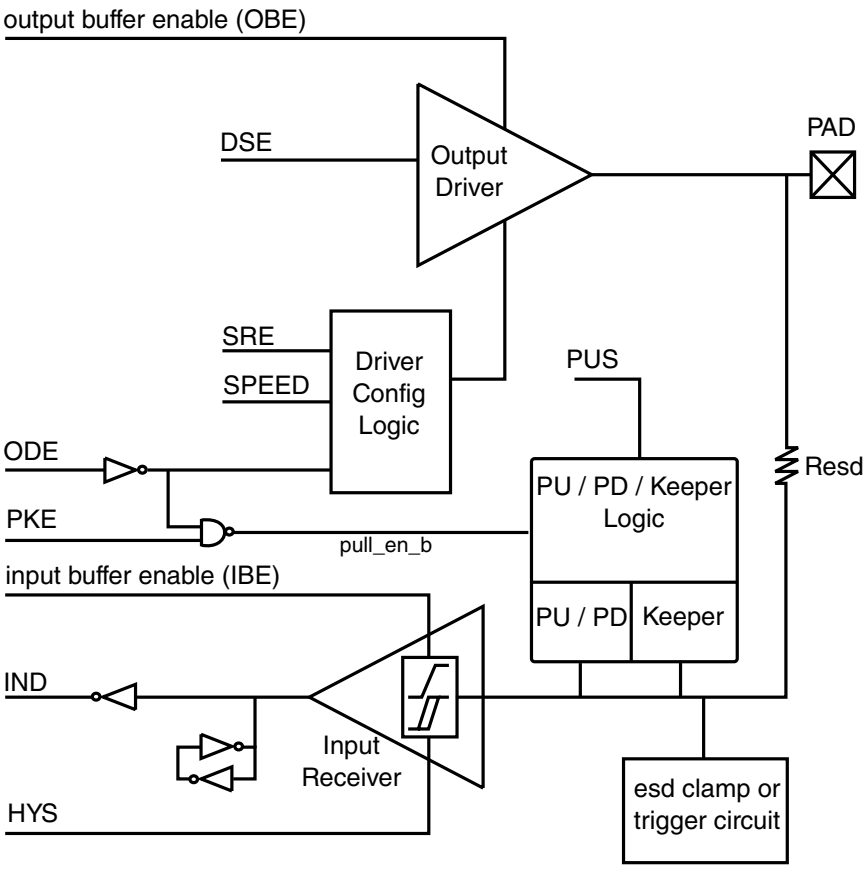


Figure 28-3. GPIO pad functional diagram

28.4.2.1 Input Driver

Input driver characteristics

- Selectable Schmitt trigger or CMOS input mode
- Keeper structure with buffer at the input receiver output to Core
- Receiver is tri-stated when I/O supply (OVDD) is powered down. (Keeper at receiver output keeps its previous state).

28.4.2.1.1 Schmitt trigger

The anti-jamming functionality of the Schmitt trigger is illustrated below.

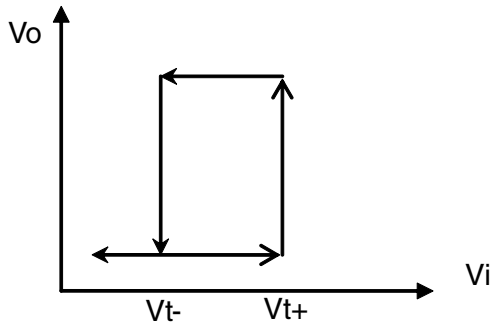


Figure 28-4. Schmitt trigger transfer characteristic

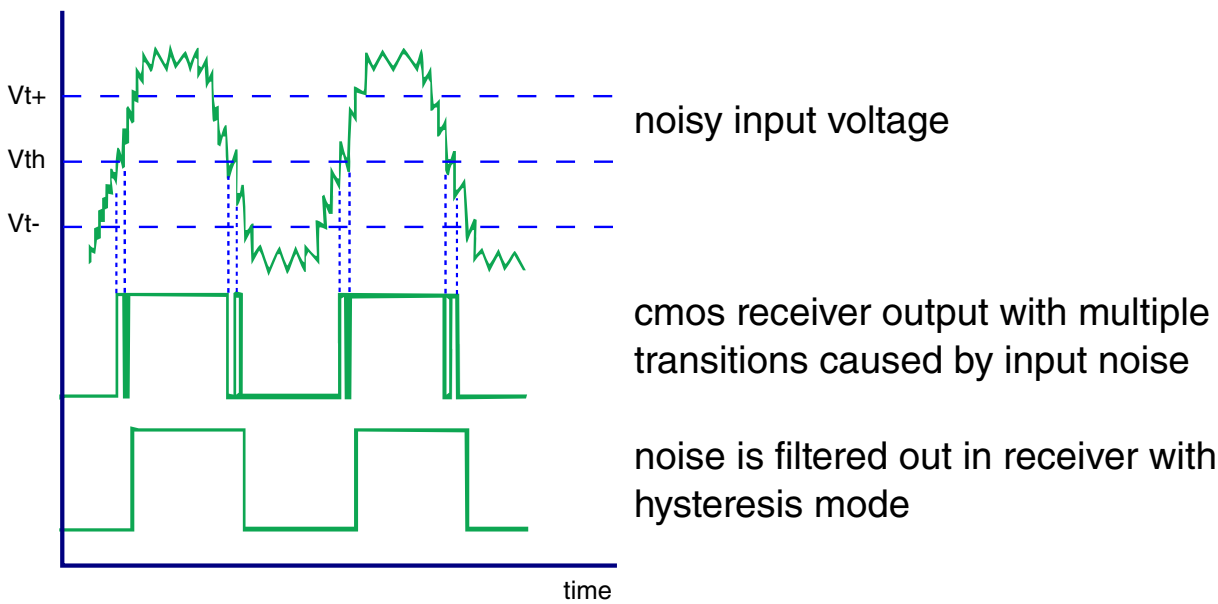


Figure 28-5. Receiver output in CMOS and hysteresis

28.4.2.1.2 Input keeper

A simple latch to hold the input value when OVDD is powered down, or the first inverter is tri-stated. Input buffer's keeper is always enabled for all the pads.

28.4.2.2 Output Driver

Output driver characteristics

- Selectable CMOS or open-drain output type

- Selectable pull-keeper enable signal to enable/disable the pull-up/down and output keeper
- Selectable pull-up resistors of 22K, 47K, 100K and a pull-down resistor of 100KOhm. Unsilicided P+ poly resistor is used to limit resistance variation to within +/- 20%.
- Pull-up, pull-down, and pad keeper are disabled in output mode.
- Seven drive strengths in each operating mode
- Additional 2-bit slew rate control to select between 50, 100, and 200 MHz IO cell operation range with reduced switching noise

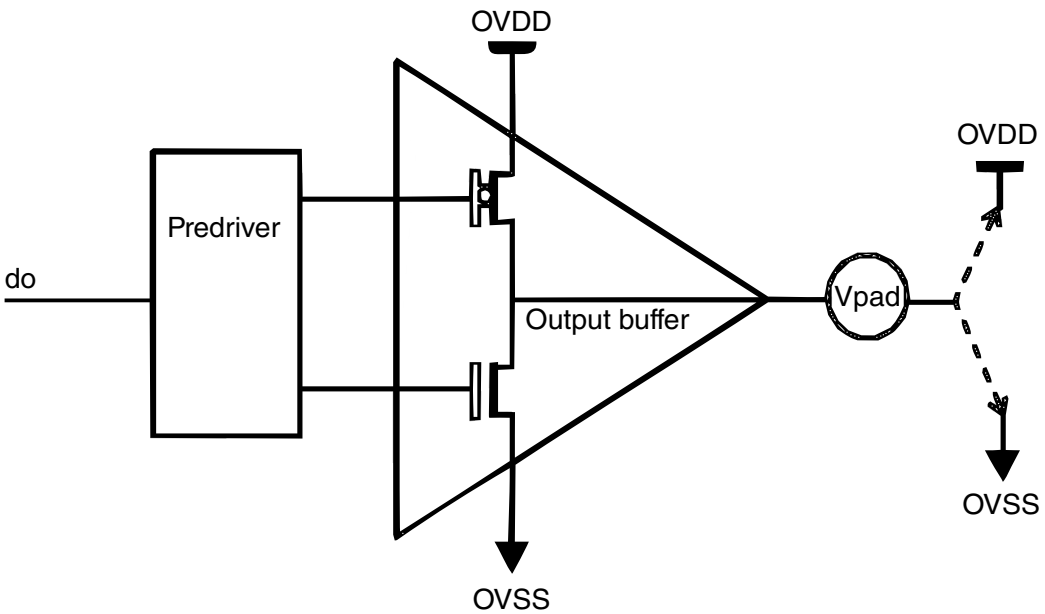


Figure 28-6. Output Driver Functional Diagram

28.4.2.2.1 Drive strength

Drive strength selection can be use to make the impedance matched and get better signal integrity.

28.4.2.2.2 Output keeper

A simple latch to hold the input value.

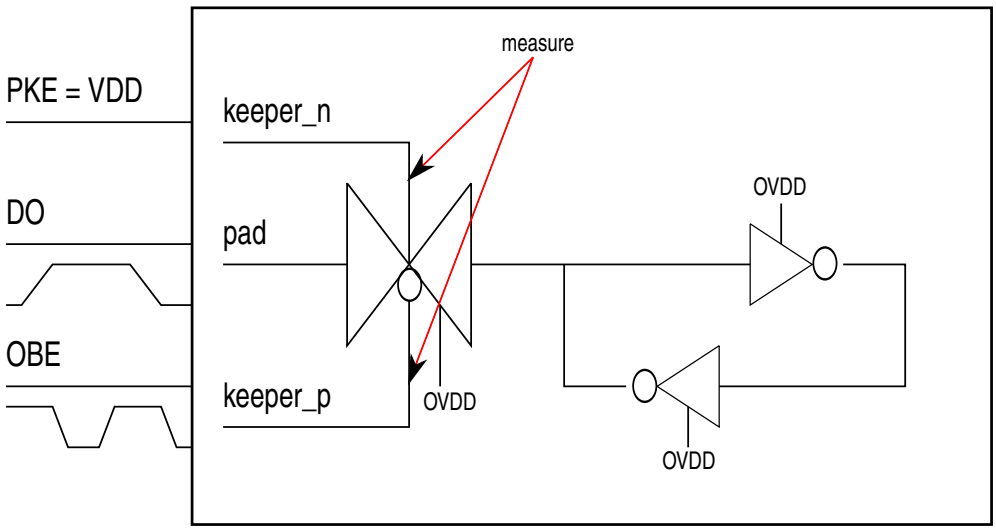


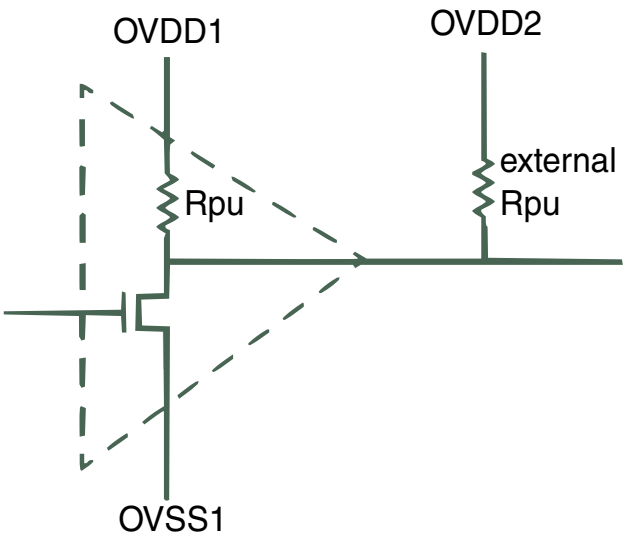
Figure 28-7. Keeper functional diagram

28.4.2.2.3 PU / PD / Keeper Logic

When Keeper is enabled, the pull-up and pull-down are disabled, and the output value of the pad depends on the Keeper. The output keeper is powered by OVDD. When the core VDD is powered down or the first inverter is tri-stated, the pad's state can be kept. Keeper and Pull can't be enabled together.

28.4.2.2.4 Open drain

Open drain is a circuit technique which allows multiple devices to communication over a single wire bi-directionally. Open drain drivers usually operate with an external or internal pull-up resistor that holds the signal line high until a device sinks enough current to pull the line low, usually used for a bus with multiple devices.



If internal pull-up resistor (Rpu) is used, output level will depend on OVDD1

If external Rpu is used, output level will depend on OVDD2

Figure 28-8. Output buffer in open drain mode

28.4.2.3 Operating Frequency

Table 28-9. IO Operating Frequency

DSE Setting	SRE / SPEED Setting	R_fixture (Ohm) OVDD = 1.8V nominal	R_fixture (Ohm) OVDD = 3.3V nominal	Operating Frequency (MHz)
1	0	400	220	50
	1			100
	10			100
	11			100
	100			50
	101			100
	110			100
	111			150
10	0	200	110	50
	1			100
	10			100
	11			100
	100			50
	101			100
	110			100
	111			150
11	0	132	73	50
	1			100

Table continues on the next page...

Table 28-9. IO Operating Frequency (continued)

DSE Setting	SRE / SPEED Setting	R_fixture (Ohm) OVDD = 1.8V nominal	R_fixture (Ohm) OVDD = 3.3V nominal	Operating Frequency (MHz)
	10			100
	11			100
	100			50
	101			100
	110			100
	111			150
100	0	104	58	50
	1			100
	10			100
	11			100
	100			50
	101			100
	110			100
	111			150
101	0	83	46	50
	1			100
	10			100
	11			100
	100			50
	101			100 (OVDD=1.8V nom) 150 (OVDD=3.3V nom)
	110			100 (OVDD=1.8V nom) 150 (OVDD=3.3V nom)
	111			150 (OVDD=1.8V nom) 200 (OVDD=3.3V nom)
110	0	70	38	50
	1			100
	10			100
	11			100
	100			50
	101			150
	110			150
	111			200
111	0	55	32	50
	1			100
	10			100
	11			100
	100			50

Table continues on the next page...

Table 28-9. IO Operating Frequency (continued)

DSE Setting	SRE / SPEED Setting	R_fixture (Ohm) OVDD = 1.8V nominal	R_fixture (Ohm) OVDD = 3.3V nominal	Operating Frequency (MHz)
	101			150
	110			150
	111			200

28.4.3 GPIO Programming

28.4.3.1 GPIO Read Mode

The programming sequence for reading input signals should be as follows:

1. Configure IOMUX to select GPIO mode (Via IOMUX Controller (IOMUXC)).
2. Configure GPIO direction register to input (GPIO_GDIR[GDIR] set to 0b).
3. Read value from data reg*ister*/pad status register.

A pseudocode description to read [input3:input0] values is as follows:

```
// SET INPUTS TO GPIO MODE.
write sw_mux_ctl_<input0>_<input1>_<input2>_<input3>, 32'h00000000
// SET GDIR TO INPUT.
write GDIR[31:4,input3_bit, input2_bit, input1_bit, input0_bit,] 32'hxxxxxxx0
// READ INPUT VALUE FROM DR.
read DR
// READ INPUT VALUE FROM PSR.
read PSR
```

NOTE

While the GPIO direction is set to input (GPIO_GDIR = 0), a read access to GPIO_DR does not return GPIO_DR data. Instead, it returns the GPIO_PSR data, which is the corresponding input signal value.

28.4.3.2 GPIO Write Mode

The programming sequence for driving output signals should be as follows:

1. Configure IOMUX to select GPIO mode (Via IOMUXC), also enable SION if need to read loopback pad value through PSR
2. Configure GPIO direction register to output (GPIO_GDIR[GDIR] set to 1b).
3. Write value to data register (GPIO_DR).

A pseudocode description to drive 4'b0101 on [output3:output0] is as follows:

```
// SET PADS TO GPIO MODE VIA IOMUX.
write sw_mux_ctl_pad_<output[0-3]>.mux_mode, <GPIO_MUX_MODE>
// Enable loopback so we can capture pad value into PSR in output mode
write sw_mux_ctl_pad_<output[0-3]>.sion, 1
// SET GDIR=1 TO OUTPUT BITS.
write GDIR[31:4,output3_bit,output2_bit, output1_bit, output0_bit,] 32'hxxxxxxxF
// WRITE OUTPUT VALUE=4'b0101 TO DR.
write DR, 32'hxxxxxxx5
// READ OUTPUT VALUE FROM PSR ONLY.
read_cmp PSR, 32'hxxxxxxx5
```

28.4.4 Interrupt Control Unit

In addition to the general-purpose input/output function, the edge-detect logic in the GPIO peripheral reflects whether a transition has occurred on a given GPIO signal that is configured as an input (GDIR bit = 0). The interrupt control registers (GPIO_ICR1 and GPIO_ICR2) may be used to independently configure the interrupt condition of each input signal (low-to-high transition, high-to-low transition, low, or high). For information about GPIO_ICR1 and GPIO_ICR2 settings, see GPIO Memory Map/Register Definition.

The interrupt control unit is built of 32 interrupt control subunits, where each subunit handles a single interrupt line.

28.5 GPIO Memory Map/Register Definition

There are eight 32-bit GPIO registers. All registers are accessible from the IP interface. Only 32-bit access is supported.

The GPIO memory map is shown in the following table.

GPIO memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_C000	GPIO data register (GPIO1_DR)	32	R/W	0000_0000h	28.5.1/1431
209_C004	GPIO direction register (GPIO1_GDIR)	32	R/W	0000_0000h	28.5.2/1432
209_C008	GPIO pad status register (GPIO1_PSR)	32	R	0000_0000h	28.5.3/1433
209_C00C	GPIO interrupt configuration register1 (GPIO1_ICR1)	32	R/W	0000_0000h	28.5.4/1433
209_C010	GPIO interrupt configuration register2 (GPIO1_ICR2)	32	R/W	0000_0000h	28.5.5/1437

Table continues on the next page...

GPIO memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_C014	GPIO interrupt mask register (GPIO1_IMR)	32	R/W	0000_0000h	28.5.6/1440
209_C018	GPIO interrupt status register (GPIO1_ISR)	32	w1c	0000_0000h	28.5.7/1441
209_C01C	GPIO edge select register (GPIO1_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20A_0000	GPIO data register (GPIO2_DR)	32	R/W	0000_0000h	28.5.1/1431
20A_0004	GPIO direction register (GPIO2_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20A_0008	GPIO pad status register (GPIO2_PSR)	32	R	0000_0000h	28.5.3/1433
20A_000C	GPIO interrupt configuration register1 (GPIO2_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20A_0010	GPIO interrupt configuration register2 (GPIO2_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20A_0014	GPIO interrupt mask register (GPIO2_IMR)	32	R/W	0000_0000h	28.5.6/1440
20A_0018	GPIO interrupt status register (GPIO2_ISR)	32	w1c	0000_0000h	28.5.7/1441
20A_001C	GPIO edge select register (GPIO2_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20A_4000	GPIO data register (GPIO3_DR)	32	R/W	0000_0000h	28.5.1/1431
20A_4004	GPIO direction register (GPIO3_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20A_4008	GPIO pad status register (GPIO3_PSR)	32	R	0000_0000h	28.5.3/1433
20A_400C	GPIO interrupt configuration register1 (GPIO3_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20A_4010	GPIO interrupt configuration register2 (GPIO3_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20A_4014	GPIO interrupt mask register (GPIO3_IMR)	32	R/W	0000_0000h	28.5.6/1440
20A_4018	GPIO interrupt status register (GPIO3_ISR)	32	w1c	0000_0000h	28.5.7/1441
20A_401C	GPIO edge select register (GPIO3_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20A_8000	GPIO data register (GPIO4_DR)	32	R/W	0000_0000h	28.5.1/1431
20A_8004	GPIO direction register (GPIO4_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20A_8008	GPIO pad status register (GPIO4_PSR)	32	R	0000_0000h	28.5.3/1433
20A_800C	GPIO interrupt configuration register1 (GPIO4_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20A_8010	GPIO interrupt configuration register2 (GPIO4_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20A_8014	GPIO interrupt mask register (GPIO4_IMR)	32	R/W	0000_0000h	28.5.6/1440
20A_8018	GPIO interrupt status register (GPIO4_ISR)	32	w1c	0000_0000h	28.5.7/1441
20A_801C	GPIO edge select register (GPIO4_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20A_C000	GPIO data register (GPIO5_DR)	32	R/W	0000_0000h	28.5.1/1431
20A_C004	GPIO direction register (GPIO5_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20A_C008	GPIO pad status register (GPIO5_PSR)	32	R	0000_0000h	28.5.3/1433
20A_C00C	GPIO interrupt configuration register1 (GPIO5_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20A_C010	GPIO interrupt configuration register2 (GPIO5_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20A_C014	GPIO interrupt mask register (GPIO5_IMR)	32	R/W	0000_0000h	28.5.6/1440
20A_C018	GPIO interrupt status register (GPIO5_ISR)	32	w1c	0000_0000h	28.5.7/1441
20A_C01C	GPIO edge select register (GPIO5_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20B_0000	GPIO data register (GPIO6_DR)	32	R/W	0000_0000h	28.5.1/1431
20B_0004	GPIO direction register (GPIO6_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20B_0008	GPIO pad status register (GPIO6_PSR)	32	R	0000_0000h	28.5.3/1433

Table continues on the next page...

GPIO memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20B_000C	GPIO interrupt configuration register1 (GPIO6_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20B_0010	GPIO interrupt configuration register2 (GPIO6_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20B_0014	GPIO interrupt mask register (GPIO6_IMR)	32	R/W	0000_0000h	28.5.6/1440
20B_0018	GPIO interrupt status register (GPIO6_ISR)	32	w1c	0000_0000h	28.5.7/1441
20B_001C	GPIO edge select register (GPIO6_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442
20B_4000	GPIO data register (GPIO7_DR)	32	R/W	0000_0000h	28.5.1/1431
20B_4004	GPIO direction register (GPIO7_GDIR)	32	R/W	0000_0000h	28.5.2/1432
20B_4008	GPIO pad status register (GPIO7_PSR)	32	R	0000_0000h	28.5.3/1433
20B_400C	GPIO interrupt configuration register1 (GPIO7_ICR1)	32	R/W	0000_0000h	28.5.4/1433
20B_4010	GPIO interrupt configuration register2 (GPIO7_ICR2)	32	R/W	0000_0000h	28.5.5/1437
20B_4014	GPIO interrupt mask register (GPIO7_IMR)	32	R/W	0000_0000h	28.5.6/1440
20B_4018	GPIO interrupt status register (GPIO7_ISR)	32	w1c	0000_0000h	28.5.7/1441
20B_401C	GPIO edge select register (GPIO7_EDGE_SEL)	32	R/W	0000_0000h	28.5.8/1442

28.5.1 GPIO data register (GPIOx_DR)

The 32-bit GPIO_DR register stores data that is ready to be driven to the output lines. If the IOMUXC is in GPIO mode and a given GPIO direction bit is set, then the corresponding DR bit is driven to the output. If a given GPIO direction bit is cleared, then a read of GPIO_DR reflects the value of the corresponding signal.Two wait states are required in read access for synchronization.

The results of a read of a DR bit depends on the IOMUXC input mode settings and the corresponding GDIR bit as follows:

- If GDIR[*n*] is set and IOMUXC input mode is GPIO, then reading DR[*n*] returns the contents of DR[*n*].
- If GDIR[*n*] is cleared and IOMUXC input mode is GPIO, then reading DR[*n*] returns the corresponding input signal's value.
- If GDIR[*n*] is set and IOMUXC input mode is not GPIO, then reading DR[*n*] returns the contents of DR[*n*].
- If GDIR[*n*] is cleared and IOMUXC input mode is not GPIO, then reading DR[*n*] always returns zero.

Address: Base address + 0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_DR field descriptions

Field	Description
DR	Data bits. This register defines the value of the GPIO output when the signal is configured as an output (GDIR[n]=1). Writes to this register are stored in a register. Reading GPIO_DR returns the value stored in the register if the signal is configured as an output (GDIR[n]=1), or the input signal's value if configured as an input (GDIR[n]=0). **NOTE:** The I/O multiplexer must be configured to GPIO mode for the GPIO_DR value to connect with the signal. Reading the data register with the input path disabled always returns a zero value.

28.5.2 GPIO direction register (GPIOx_GDIR)

GPIO_GDIR functions as direction control when the IOMUXC is in GPIO mode. Each bit specifies the direction of a one-bit signal. The mapping of each DIR bit to a corresponding SoC signal is determined by the SoC's pin assignment and the IOMUX table. For more details consult the IOMUXC chapter.

Address: Base address + 4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	GDIR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_GDIR field descriptions

Field	Description
GDIR	GPIO direction bits. Bit n of this register defines the direction of the GPIO[n] signal. **NOTE:** GPIO_GDIR affects only the direction of the I/O signal when the corresponding bit in the I/O MUX is configured for GPIO. 0 **INPUT** — GPIO is configured as input. 1 **OUTPUT** — GPIO is configured as output.

28.5.3 GPIO pad status register (GPIOx_PSR)

GPIO_PSR is a read-only register. Each bit stores the value of the corresponding input signal (as configured in the IOMUX). This register is clocked with the ipg_clk_s clock, meaning that the input signal is sampled only when accessing this location. Two wait states are required any time this register is accessed for synchronization.

Address: Base address + 8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PSR																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_PSR field descriptions

Field	Description
PSR	GPIO pad status bits (status bits). Reading GPIO_PSR returns the state of the corresponding input signal. Settings: **NOTE:** The IOMUXC must be configured to GPIO mode for GPIO_PSR to reflect the state of the corresponding signal.

28.5.4 GPIO interrupt configuration register1 (GPIOx_ICR1)

GPIO_ICR1 contains 16 two-bit fields, where each field specifies the interrupt configuration for a different input signal.

Address: Base address + Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	ICR15		ICR14		ICR13		ICR12		ICR11		ICR10		ICR9		ICR8	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ICR7		ICR6		ICR5		ICR4		ICR3		ICR2		ICR1		ICR0	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_ICR1 field descriptions

Field	Description
31–30 ICR15	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 15.

Table continues on the next page...

GPIOx_ICR1 field descriptions (continued)

Field	Description
	Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
29–28 ICR14	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 14. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
27–26 ICR13	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 13. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
25–24 ICR12	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 12. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
23–22 ICR11	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 11. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
21–20 ICR10	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 10. Settings:

Table continues on the next page...

GPIOx_ICR1 field descriptions (continued)

Field	Description
	Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
19–18 ICR9	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 9. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
17–16 ICR8	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 8. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
15–14 ICR7	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 7. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
13–12 ICR6	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 6. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
11–10 ICR5	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 5. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows:

Table continues on the next page...

GPIOx_ICR1 field descriptions (continued)

Field	Description
	00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
9–8 ICR4	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 4. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
7–6 ICR3	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 3. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
5–4 ICR2	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 2. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
3–2 ICR1	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 1. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
ICR0	Interrupt configuration 1 fields. This register controls the active condition of the interrupt function for GPIO interrupt 0. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive.

Table continues on the next page...

GPIOx_ICR1 field descriptions (continued)

Field	Description
	01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.

28.5.5 GPIO interrupt configuration register2 (GPIOx_ICR2)

GPIO_ICR2 contains 16 two-bit fields, where each field specifies the interrupt configuration for a different input signal.

Address: Base address + 10h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	ICR31		ICR30		ICR29		ICR28		ICR27		ICR26		ICR25		ICR24	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ICR23		ICR22		ICR21		ICR20		ICR19		ICR18		ICR17		ICR16	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_ICR2 field descriptions

Field	Description
31–30 ICR31	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 31. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
29–28 ICR30	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 30. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
27–26 ICR29	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 29.

Table continues on the next page...

GPIOx_ICR2 field descriptions (continued)

Field	Description
	Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
25–24 ICR28	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 28. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
23–22 ICR27	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 27. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
21–20 ICR26	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 26. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
19–18 ICR25	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 25. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
17–16 ICR24	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 24. Settings:

Table continues on the next page...

GPIOx_ICR2 field descriptions (continued)

Field	Description
	Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
15–14 ICR23	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 23. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
13–12 ICR22	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 22. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
11–10 ICR21	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 21. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
9–8 ICR20	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 20. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
7–6 ICR19	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 19. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows:

Table continues on the next page...

GPIOx_ICR2 field descriptions (continued)

Field	Description
	00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
5–4 ICR18	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 18. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
3–2 ICR17	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 17. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.
ICR16	Interrupt configuration 2 fields. This register controls the active condition of the interrupt function for GPIO interrupt 16. Settings: Bits ICRn[1:0] determine the interrupt condition for signal n as follows: 00 **LOW_LEVEL** — Interrupt n is low-level sensitive. 01 **HIGH_LEVEL** — Interrupt n is high-level sensitive. 10 **RISING_EDGE** — Interrupt n is rising-edge sensitive. 11 **FALLING_EDGE** — Interrupt n is falling-edge sensitive.

28.5.6 GPIO interrupt mask register (GPIOx_IMR)

GPIO_IMR contains masking bits for each interrupt line.

Address: Base address + 14h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IMR																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_IMR field descriptions

Field	Description
IMR	Interrupt Mask bits. This register is used to enable or disable the interrupt function on each of the 32 GPIO signals. Settings: Bit IMR[n] (n=0...31) controls interrupt n as follows: 0 **UNMASKED** — Interrupt n is disabled. 1 **MASKED** — Interrupt n is enabled.

28.5.7 GPIO interrupt status register (GPIOx_ISR)

The GPIO_ISR functions as an interrupt status indicator. Each bit indicates whether an interrupt condition has been met for the corresponding input signal. When an interrupt condition is met (as determined by the corresponding interrupt condition register field), the corresponding bit in this register is set. Two wait states are required in read access for synchronization. One wait state is required for reset.

Address: Base address + 18h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	ISR																															
W	w1c																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_ISR field descriptions

Field	Description
ISR	Interrupt status bits - Bit n of this register is asserted (active high) when the active condition (as determined by the corresponding ICR bit) is detected on the GPIO input and is waiting for service. The value of this register is independent of the value in GPIO_IMR. When the active condition has been detected, the corresponding bit remains set until cleared by software. Status flags are cleared by writing a 1 to the corresponding bit position.

28.5.8 GPIO edge select register (GPIOx_EDGE_SEL)

GPIO_EDGE_SEL may be used to override the ICR registers' configuration. If the GPIO_EDGE_SEL bit is set, then a rising edge or falling edge in the corresponding signal generates an interrupt. This register provides backward compatibility. On reset all bits are cleared (ICR is not overridden).

Address: Base address + 1Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	GPIO_EDGE_SEL																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPIOx_EDGE_SEL field descriptions

Field	Description
GPIO_EDGE_SEL	Edge select. When GPIO_EDGE_SEL[*n*] is set, the GPIO disregards the ICR[*n*] setting, and detects any edge on the corresponding input signal.

Chapter 29
General Purpose Media Interface (GPMI)

29.1 Overview

The GPMI controller is a flexible interface to supporting up to four NAND flash chip selects.

- ONFI 2.2, DDR Mode, Samsung/Toshiba Toggle NAND protocol compatible
- Fully configurable address and command behavior, providing support for future devices not yet specified.

The GPMI resides on the APBH. The GPMI also provides an interface to the BCH module to allow direct parity processing.

Registers are clocked on the HCLK domain. The I/O and pin timing are clocked on a dedicated GPMICLK domain. GPMICLK can be set to maximize I/O performance.

The figure below shows a block diagram of the GPMI controller.

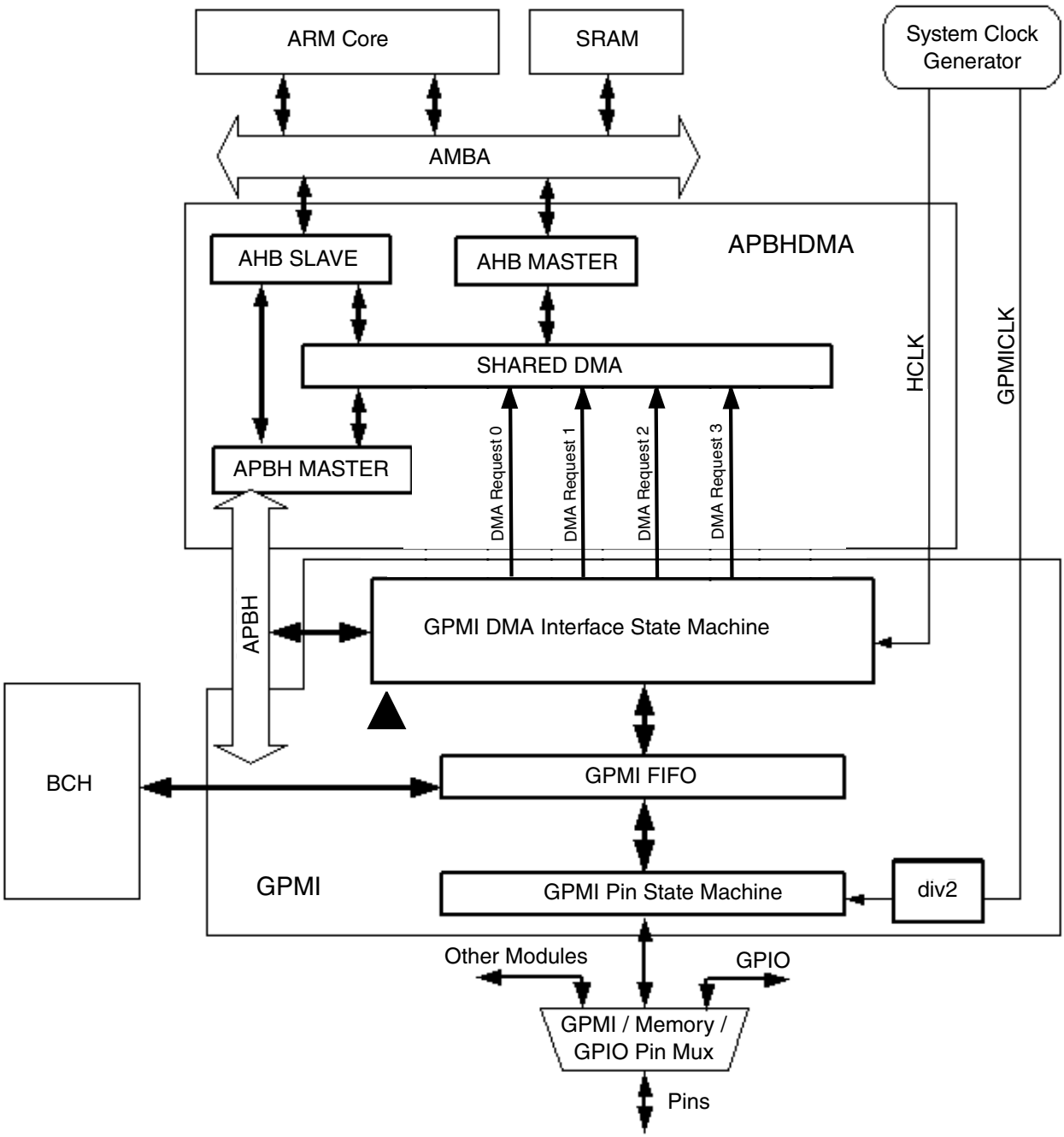


Figure 29-1. General-Purpose Media Interface Controller Block Diagram

29.2 External Signals

The table found here describes the external signals of GPMI.

Table 29-1. GPMI External Signals

Signal	Description	Pad	Mode	Direction
NAND_ALE	Address latch enable signal	NANDF_ALE	ALT0	O
NAND_CE0_B	Chip enable signal	NANDF_CS0	ALT0	O
NAND_CE1_B	Chip enable signal	NANDF_CS1	ALT0	O
NAND_CE2_B	Chip enable signal	NANDF_CS2	ALT0	O
NAND_CE3_B	Chip enable signal	NANDF_CS3	ALT0	O
NAND_CLE	Command latch enable signal	NANDF_CLE	ALT0	O
NAND_DATA00	Data signal	NANDF_D0	ALT0	I/O
NAND_DATA01	Data signal	NANDF_D1	ALT0	I/O
NAND_DATA02	Data signal	NANDF_D2	ALT0	I/O
NAND_DATA03	Data signal	NANDF_D3	ALT0	I/O
NAND_DATA04	Data signal	NANDF_D4	ALT0	I/O
NAND_DATA05	Data signal	NANDF_D5	ALT0	I/O
NAND_DATA06	Data signal	NANDF_D6	ALT0	I/O
NAND_DATA07	Data signal	NANDF_D7	ALT0	I/O
NAND_DQS	DQS signal	SD4_DAT0	ALT2	I/O
NAND_READY	Ready signal	NANDF_RB0	ALT0	I/O
NAND_RE_B	Read enable signal	SD4_CMD	ALT1	O
NAND_WE_B	Write enable signal	SD4_CLK	ALT1	O
NAND_WP_B	Wait polarity signal	NANDF_WP_B	ALT0	O

29.3 Clocks

The table found here describes the clock sources for GPMI.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 29-2. GPMI Clocks

Clock name	Clock Root	Description
bch_input_apb_clk	usdhc3_clk_root	BCH to APBH input clock
gpmi_bch_input_bch_clk	usdhc4_clk_root	BCH input clock
gpmi_bch_input_gpmi_io_clk	enfc_clk_root	GPMI IO input clock
gpmi_input_apb_clk	usdhc3_clk_root	GPMI to APBH clock

29.4 GPMI NAND Mode

The general-purpose media interface has several features to efficiently support NAND:

- Individual chip select pins and ganged ready/busy pin for up to four NANDs.
- Individual state machine and DMA channel for each chip select.
- Special command modes work with DMA controller to perform all normal NAND functions without CPU intervention.
- Configurable timing based on a dedicated clock allows optimal balance of high NAND performance and low system power.

GPMI and DMA have been designed to handle complex multi-page operations without CPU intervention. The DMA uses a linked descriptor function with branching capability to automatically handle all of the operations needed to read/write multiple pages:

- **Data/Register Read/Write**-The GPMI can be programmed to read or write multiple cycles to the NAND address, command or data registers.
- **Wait for NAND Ready**-The GPMI's Wait-for-Ready mode can monitor the ready/busy signal of a single NAND flash and signal the DMA when the device has become ready. It also has a time-out counter and can indicate to the DMA that a time-out error has occurred. The DMAs can conditionally branch to a different descriptor in the case of an error.
- **Check Status**-The Read-and-Compare mode allows the GPMI to check NAND status against a reference. If an error is found, the GPMI can instruct the DMA to branch to an alternate descriptor, which attempts to fix the problem or asserts a CPU IRQ.

29.4.1 Multiple NAND Support

The GPMI supports up to four NAND chip selects, with ganged ready/busy pins.

Since they share a data bus and control lines, the GPMI can only actively communicate with a single NAND at a time. However, all NANDs can concurrently perform internal read, write, or erase operations. With fast NAND flash and software support for concurrent NAND operations, this architecture allows the total throughput to approach the data bus speed, which can be as high as 50 MB/s (8-bit bus running at 50 MHz single clock edge) in asynchronous mode and 200MB/s (8-bit bus running at 100MHz both clock edges) in Source Synchronous mode.

There are two options for controlling the four NAND chip selects via the DMA interface. The first option is the one to one mapping, where the each DMA channel is tied to its own NAND chip select. For example DMA channel 'n' accesses only NAND attached to chip select 'n'. The second option is the decoupled mode where a DMA channel can access any or all NAND chip selects connected to the GPMI. A DMA channel will signify the NAND chip select it wants to access by writing its chip select value in the GPMI_CTRL0[CS] field and setting the GPMI_CTRL1[DECOUPLE_CS] to 1. This option is useful if software chooses to use only one DMA channel to access all the attached NAND devices.

29.4.2 GPMI NAND Timing and Clocking

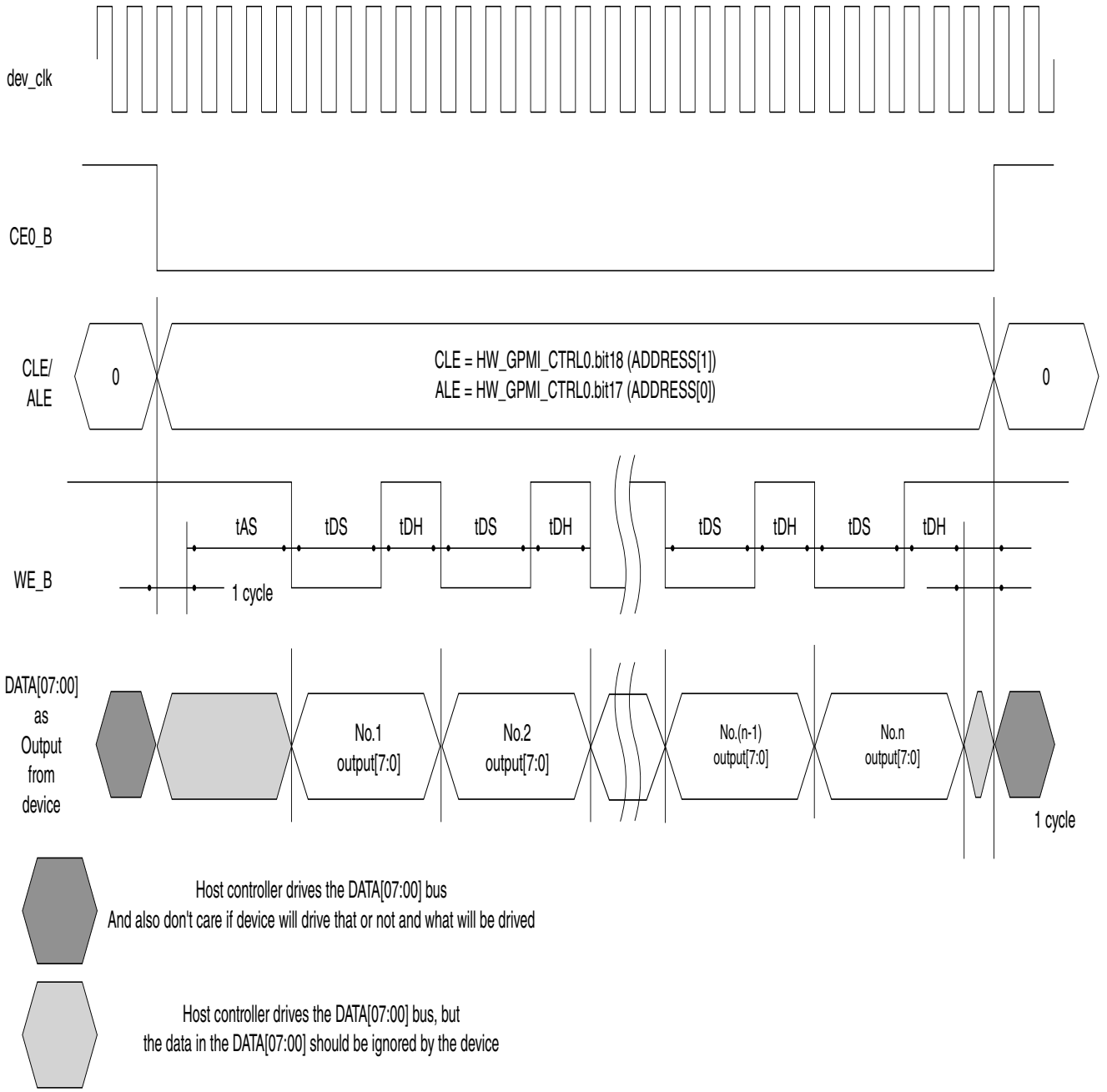
The dedicated clock, GPMICLK, is used as a timing reference for NAND flash I/O. Since various NANDs have different timing requirements, GPMICLK may need to be adjusted for each application.

While the actual pin timings are limited by the NAND chips used and the I/O pad configuration, the GPMI can support data bus speeds of up to 200 MHz x 8 bits. The actual read/write strobe timing parameters are adjusted as indicated in the register descriptions in Memory Map.

29.4.3 Basic NAND Timing

29.4.3.1 NAND Asynchronous Timing

Figure 29-3 and illustrates the operation of the output (from host to device) timing parameters in NAND ONFI asynchronous mode.



- tAS is configurable by programming HW_GPMI_TIMING0 Address_Setup: in this example, Address_Setup = 4, tAS is equal to 4 dev_clk cycles.
- tDS is configurable by programming HW_GPMI_TIMING0 Data_Setup; in this example, Data_Setup = 3, tDS is equal to 3 dev_clk cycles
- tDH is configuarble by programming HW_GPMI_TIMING0 Data_Hold: in this example, Data_Hold = 2, tDH is equal to 2 dev_clk cycles
- tAS/tDS/tDH will extend, if the output data is not ready in device fifo.

Figure 29-2. Asynchronous Mode Basic Write Timing Diagram (command write, address write, or data write

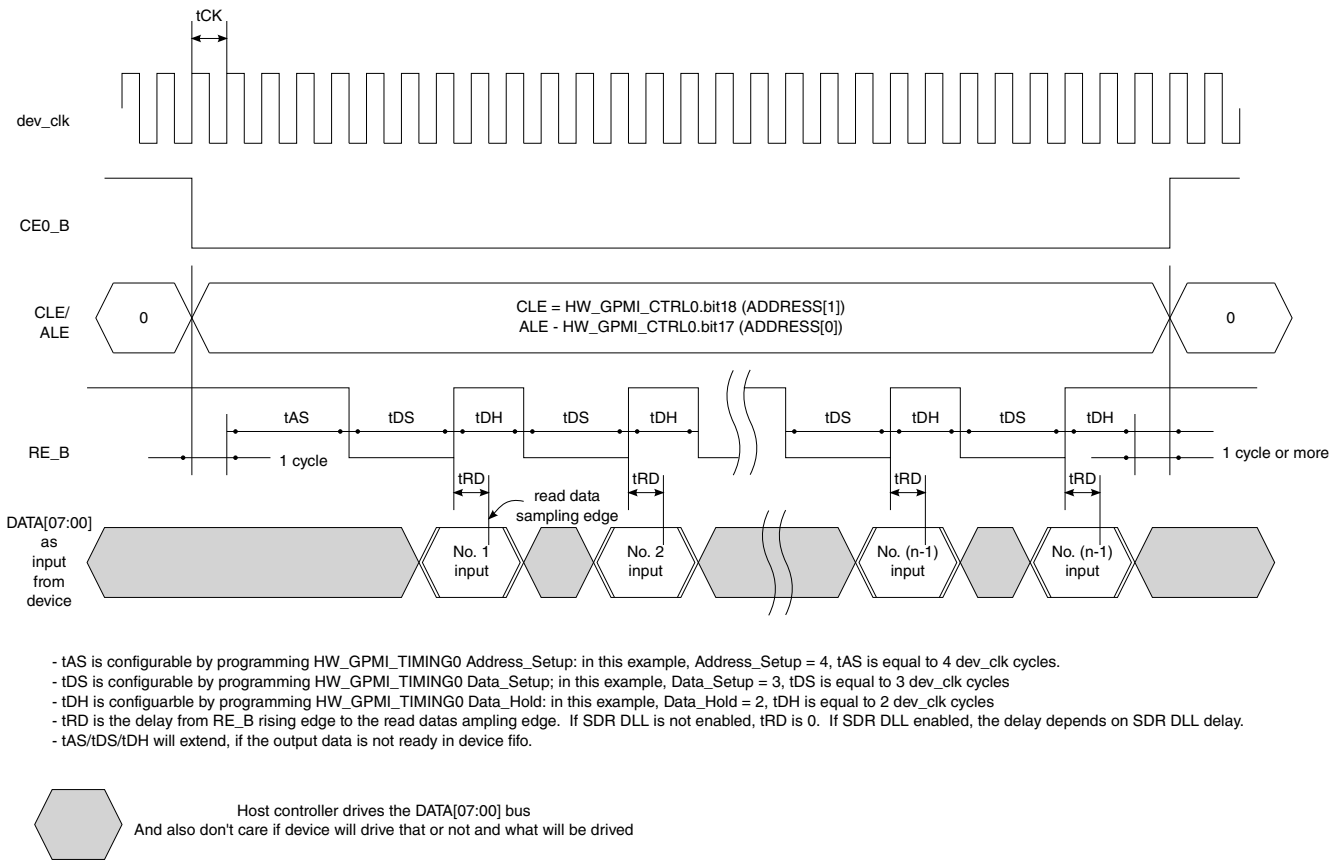


Figure 29-3. ONFI Asynchronous Mode Basic Read Timing Diagram (data read)

29.4.3.2 NAND Asynchronous EDO Mode Timing

In high-speed NANDS, the read data may not be valid until after the read strobe (NAND_RE_B) deasserts. This is the case when the minimum tDS is programmed to achieve higher bandwidth.

The GPMI implements a feedback read strobe to sample the read data. The feedback read strobe can be delayed to support fast nand EDO (Extended Data Out) timing where the read strobe may deassert before the read data is valid, and read data is valid for some time after read strobe.

Nand EDO timings is applied typically for read cycle frequency above 33 MHz. See Figure 29-4.

The GPMI provides control over the amount of delay applied to the feedback read strobe. This delay depends on the maximum read access time (tREA) of the nand and the read pulse width (tRP) used to access the nand. tRP is specified by

GPMI_TIMING0[DATA_SETUP] register. When (tREA + 4ns) is less than tRP, no delay is required to sample to nand read data. (The 4ns provides adequate data setup time for the GPMI.) In this case set GPMI_CTRL1[HALF_PERIOD] = 0; GPMI_CTRL1[RDN_DELAY] = 0; GPMI_CTRL1[DLL_ENABLE] = 0.

When (tREA + 4ns) is greater than or equal to tRP, a delay of the feedback read strobe is required to sample to nand read data. This delay is equal to the difference between these two timings:

DELAY = tREA + 4ns - tRP.

Since the GPMI delay chain is limited to 12ns maximum, if DELAY > 12ns then increase tRP by increasing the value of GPMI_TIMING0[DATA_SETUP] until DELAY is less than or equal to 12ns.

The GPMI programming for this DELAY depends on the GPMICLK period. The GPMI DLL will not function properly if the GPMICLK period is greater than 24ns: disable the DLL if this is the case. If the GPMICLK period is greater than 12ns (and not greater than 24ns), set the GPMI_CTRL1[HALF_PERIOD]=1; This will cause the DLL reference period (RP) to be one-half of the GPMICLK period. If the GPMICLK period is 12ns or less then set the GPMI_CTRL1[HALF_PERIOD]=0; This will cause the DLL reference period (RP) to be equal to the GPMICLK period. DELAY is a multiple (0 to 1.875) of RP.

The GPMI_CTRL1[RDN_DELAY] is encoded as a 1-bit integer and 3-bit fraction delay factor. DELAY is a multiple of the delay factor and the reference period. See table below for details.

Table 29-3. RDN DELAY

HW_GPMI_CTRL1[RDN_DELAY]	Delay Factor
0	0.000
1	0.125
2	0.250
3	0.375
4	0.500
5	0.625
6	0.750
7	0.875
8	1.000
9	1.125
10	1.250
11	1.375
12	1.500

Table continues on the next page...

Table 29-3. RDN DELAY (continued)

HW_GPMI_CTRL1[RDN_DELAY]	Delay Factor
13	1.625
14	1.750
15	1.875

DELAY = DelayFactor x RP or DELAY = GPMI_CTRL1[RDN_DELAY] x 0.125 x RP.

Use this equation to calculate the value for GPMI_CTRL1[RDN_DELAY]. Then set GPMI_CTRL1[DLL_ENABLE]=1.

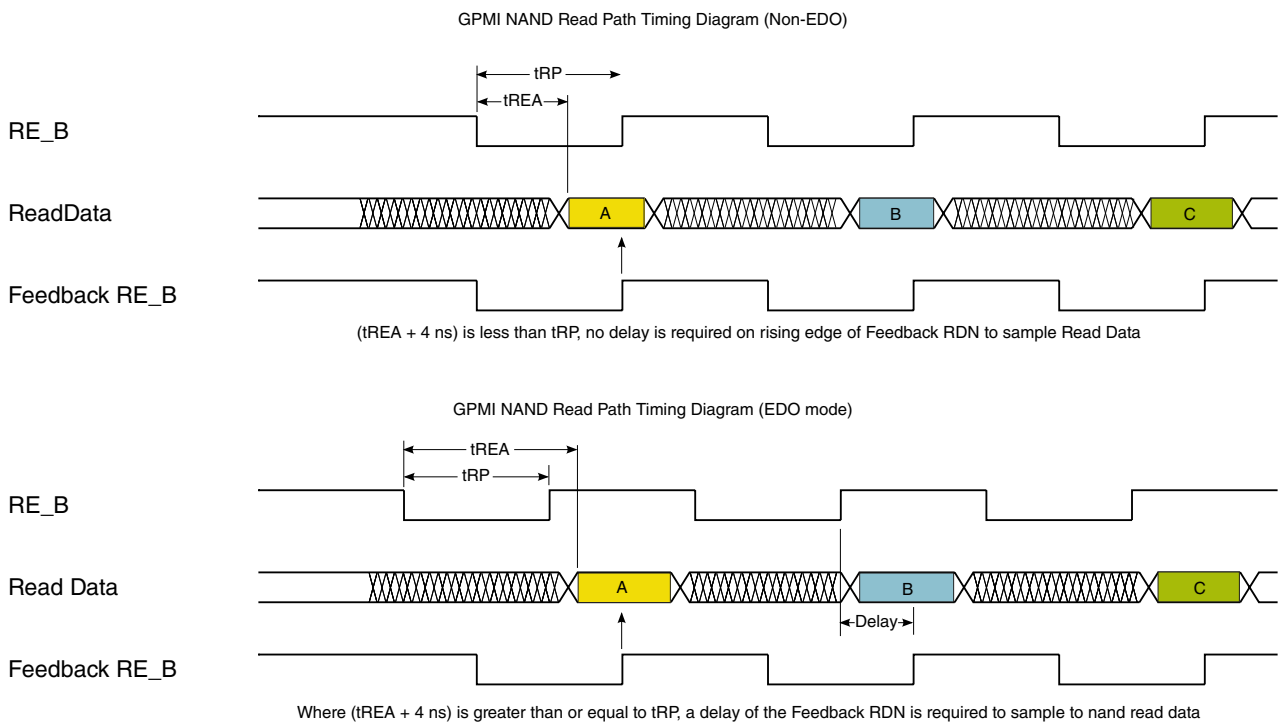


Figure 29-4. NAND Read Path Timing

For example, a NAND with tREAmax = 20 ns, tRPmin = 12 ns, and tRCmin = 25 ns (read cycle time) may be programmed as follows:

- GPMICLK clock frequency: Consider 480/6 = 80 MHz which is 12.5 ns clock period. This is too close to the minimum NAND spec if we program the data setup and hold to 1 GPMICLK cycle. Consider 480/7=68.57 MHz which is 14.58 ns clock period. With data setup and hold set to 1, we have a tRP of 14.58ns and a tRC of 29.16 ns (good margins).

- Since (tREA +4ns) is greater than tRP, required DELAY = tREA + 4ns - tRP = 20 + 4 - 14.58ns = 9.42 ns.
- GPMI_CTRL1[HALF_PERIOD] =1, since GPMICLK period is large than 12ns. So RP=GPMICLK period = 7.29ns.
- DELAY = GPMI_CTRL1[RDN_DELAY] x 0.125 x RP. 9.42 ns = GPMI_CTRL1[RDN_DELAY] x 0.125 x 7.29ns. GPMI_CTRL1[RDN_DELAY] = 10 (round of 10.33)

For ONFI spec EDO timing mode 4/5, It's recommended to have below settings.

EDO timing mode 5 :
GPMICLK=100MHz;GPMI_CTRL1[RDN_DELAY]=8;GPMI_CTRL1[HALF_PERIOD] =0;

EDO timing mode 4 :
GPMICLK=80MHz;GPMI_CTRL1[RDN_DELAY]=15;GPMI_CTRL1[HALF_PERIOD] =1;

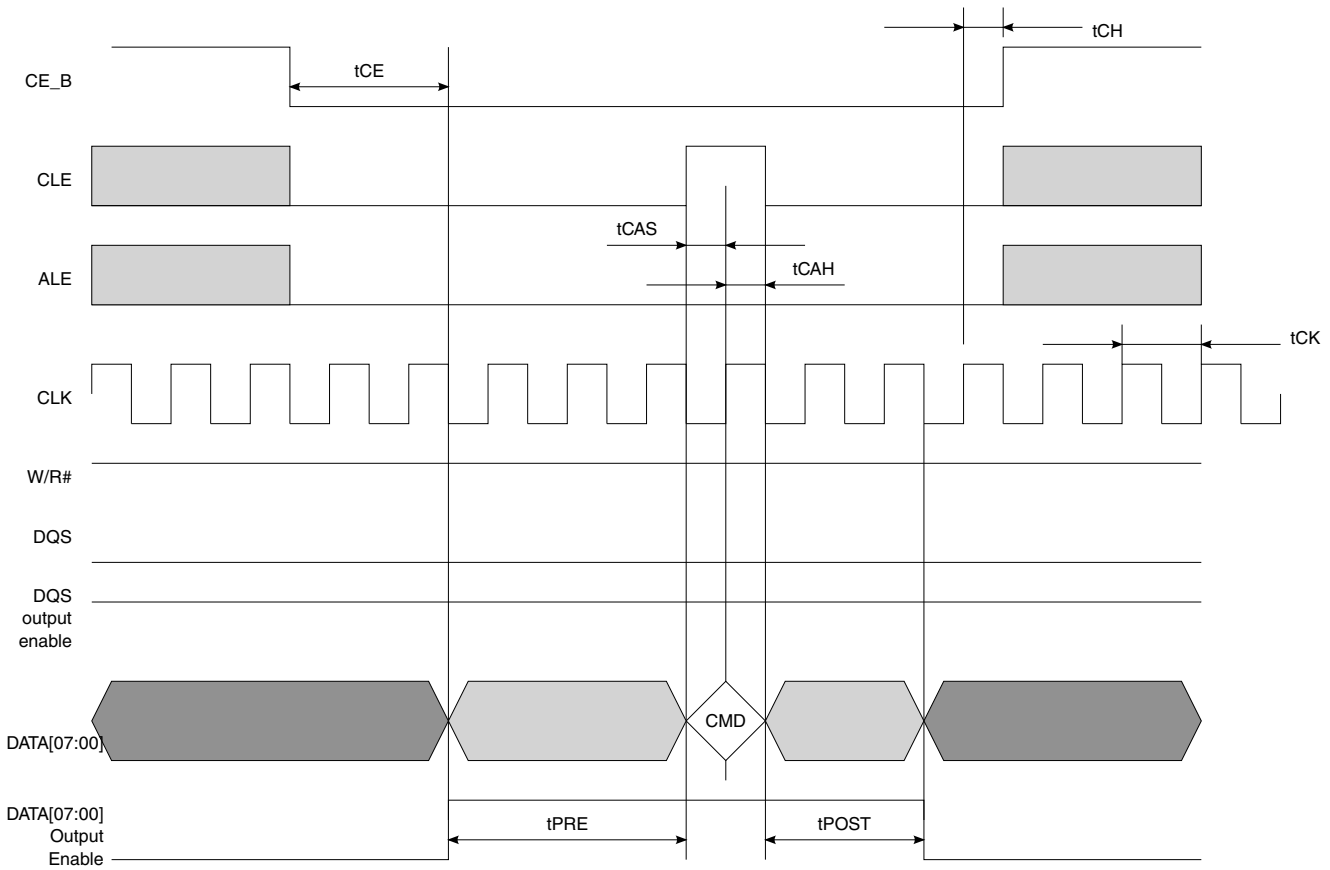
NOTE

It is recommended that the drive strength of NAND_RE_B and NAND_WE_B output pins be set to 8 mA. This will reduce the transition time under heavy loads. Low transition times will be important when NAND interface read and write cycle times are below 30 ns. The other GPMI pins may remain at 4 mA, since their frequency is only up to half that of NAND_RE_B and NAND_WE_B.

29.4.3.3 NAND ONFI Source Synchronous Mode Timing

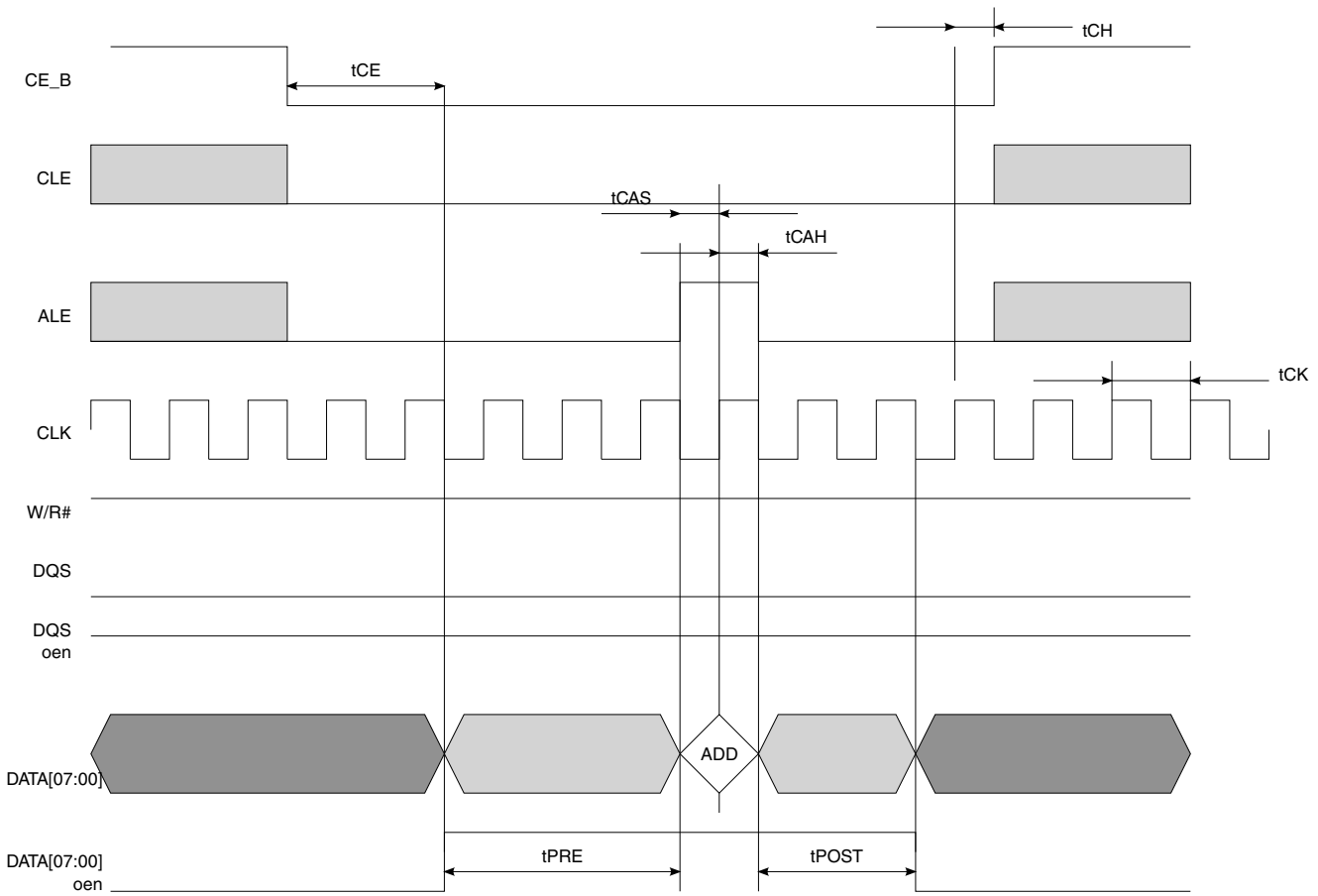
NOTE

In the following figures, CLK shares the same pin as WE_B in Async Mode. And W/R# shares the same pin as RE_B in Async Mode.



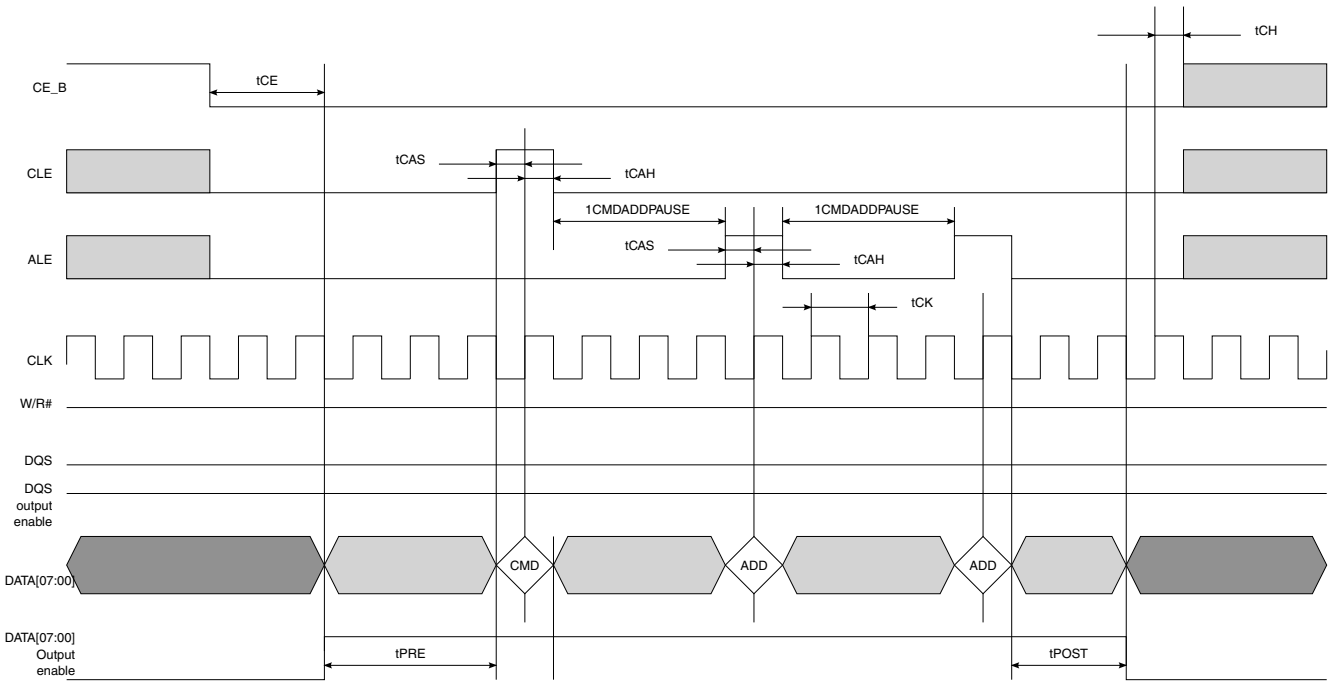
tCK is equal to device clock period
tCAS = 0.5 * tCK
tCAH = 0.5 * tCK
tCE = HW_GPMI_TIMING2.CE_DELAY * tCK
tCH = 0.5 * tCK
tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY * tCK
tPOST = HW_GPMI_TIMING2.POST_DELAY * tCK

Figure 29-5. ONFI Source Synchronous Mode Basic Command Write Timing Diagram



tCK is equal to device clock period
tCAS = 0.5 * tCK
tCAH = 0.5 * tCK
tCE = HW_GPMI_TIMING2.CE_DELAY * tCK
tCH = 0.5 * tCK
tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY * tCK
tPOST = HW_GPMI_TIMING2.POST_DELAY * tCK

Figure 29-6. ONFI Source Synchronous Mode Basic Address Write Timing Diagram

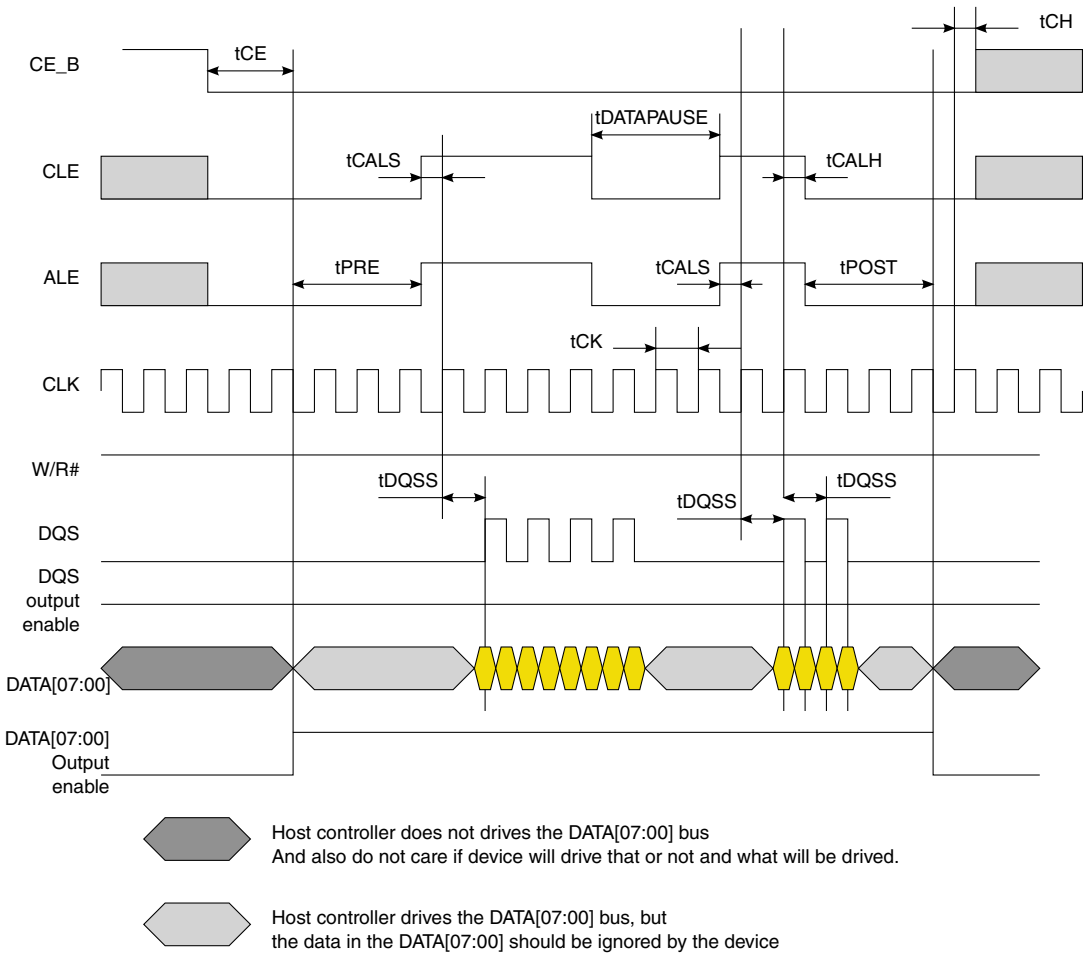


Host controller does not drives the DATA[07:00] bus
And also do not care if device will drive that or not and what will be drived.

Host controller drives the DATA[07:00] bus, but
the data in the DATA[07:00] should be ignored by the device

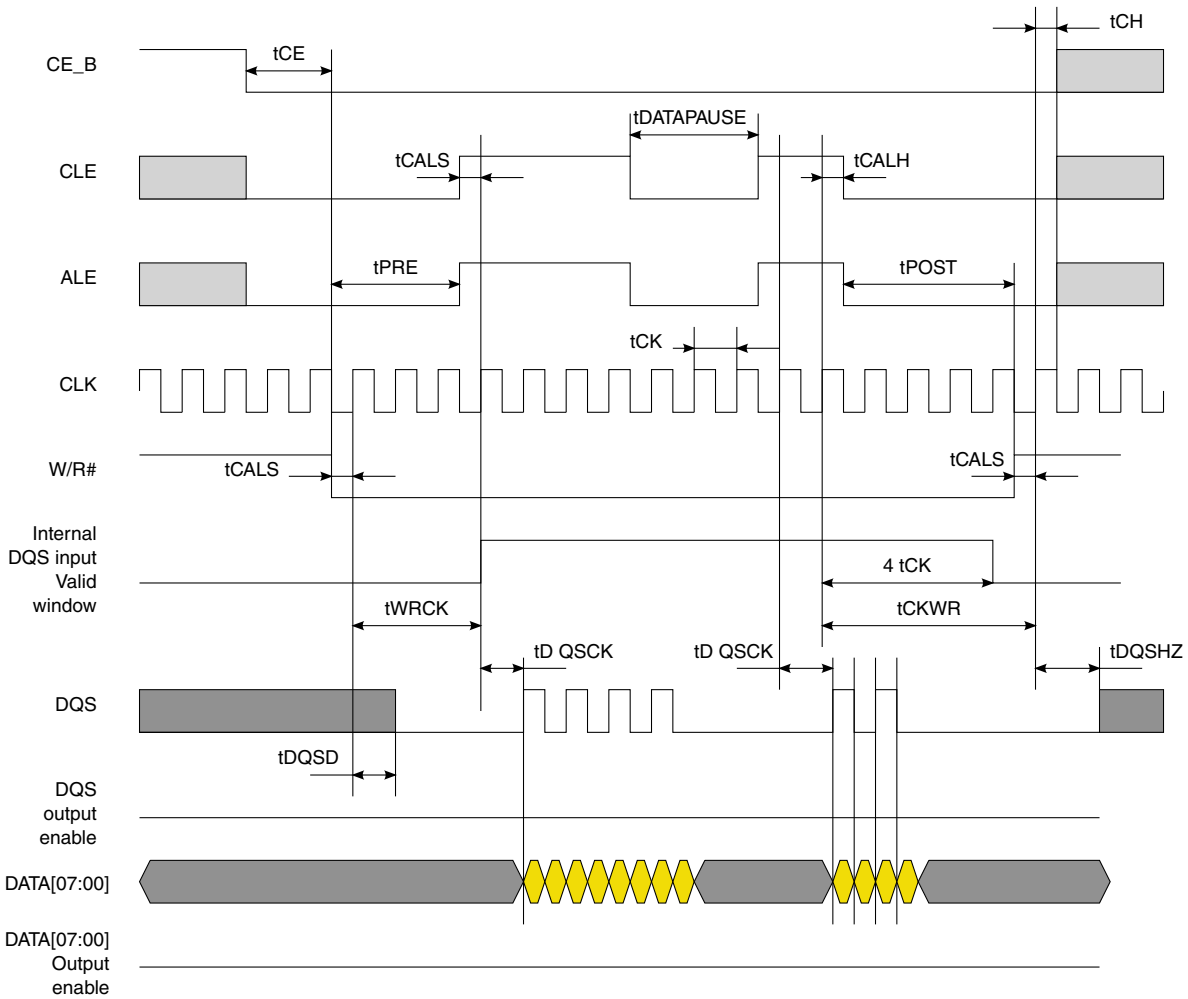
tCK is equal to device clock period
tCAS = 0.5 * tCK
tCAH = 0.5 *tCK
tCE = HW_GPMI_TIMING2.CE_DELAY * tCK
tCH = 0.5 * tCK
tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY * tCK
tPOST = HW_GPMI_TIMING2.POST_DELAY * tCK
tCMDADDPAUSE = HW_GPMI_TIMING2.CMDADD_PAUSE * tCK

Figure 29-7. ONFI Source Synchronous Mode Command + Address Write Timing Diagram



tCK is equal to device clock period
tCE = HW_GPI_TIMING2.CE_DELAY * tCK
tCH = 0.5 * tCK
tCALS = 0.5 * tCK
tCALH = 0.5 * tCK
tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY *tCK
tPOST = HW_GPMI_TIMING2.POST_DELAY * tCK
tDATAPAUSE = HW_GPMI_TIMING2.DATA_PAUSE * tCK
tDQSS = tCK

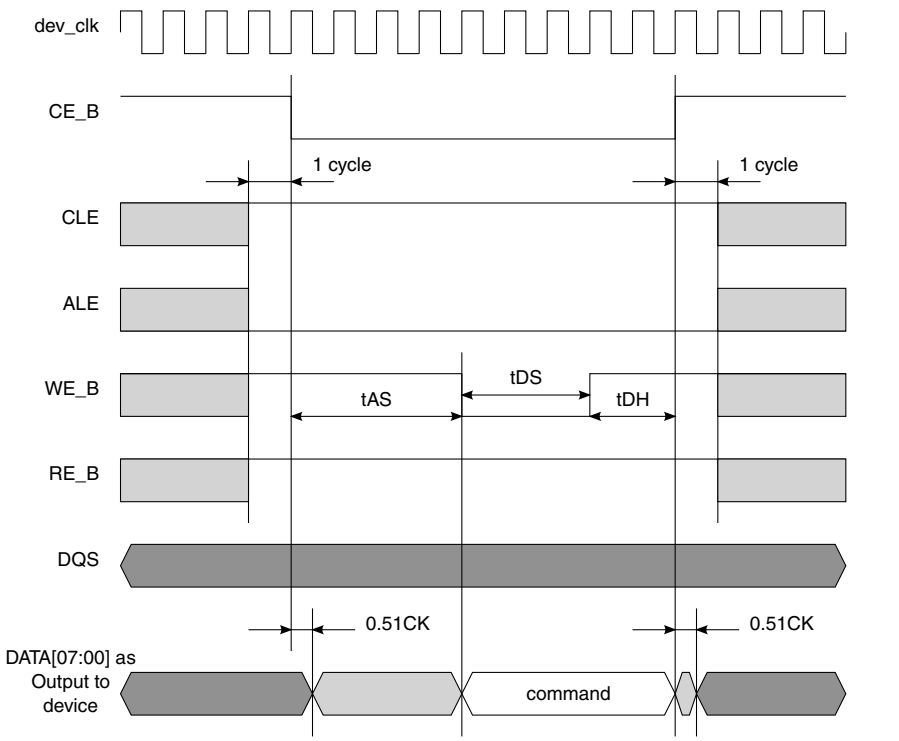
Figure 29-8. ONFI Source Synchronous Mode Data Write Timing Diagram



tCK is equal to device clock period
tCE = HW_GPI_TIMING2.CE_DELAY * tCK
tCH = 0.5 * tCK
tCALS = 0.5 * tCK
tCALH = 0.5 * tCK
tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY *tCK
tPOST = HW_GPMI_TIMING2.POST_DELAY * tCK
tDATAPAUSE = HW_GPMI_TIMING2.DATA_PAUSE * tCK
tWRCK/tCKWR/tDQSD/tDASCK/tDASHZ are device parameters

Figure 29-9. ONFI Source Synchronous Mode Data Read Timing Diagram

29.4.3.4 NAND Toggle Mode Timing



Host controller does not drive the DATA[07:00] bus
And also does not care if device will drive that or not and what will be drived.

Host controller drives the DATA[07:00] bus, but
the data in the DATA[07:00] should be ignored by the device

- tAS is configurable by programming HW_GPMI_TIMING0 Address_Setup;
- tDS is configurable by programming HW_GPMI_TIMING0 Data_Setup;
- tDH is configurable br programming HW_GPMI_TIMING Data_Hold;
- tAS/tDS/tDH will extend, if the output data is not ready in device fifo.

Figure 29-10. Samsung Toggle Mode Basic Command Write Timing Diagram

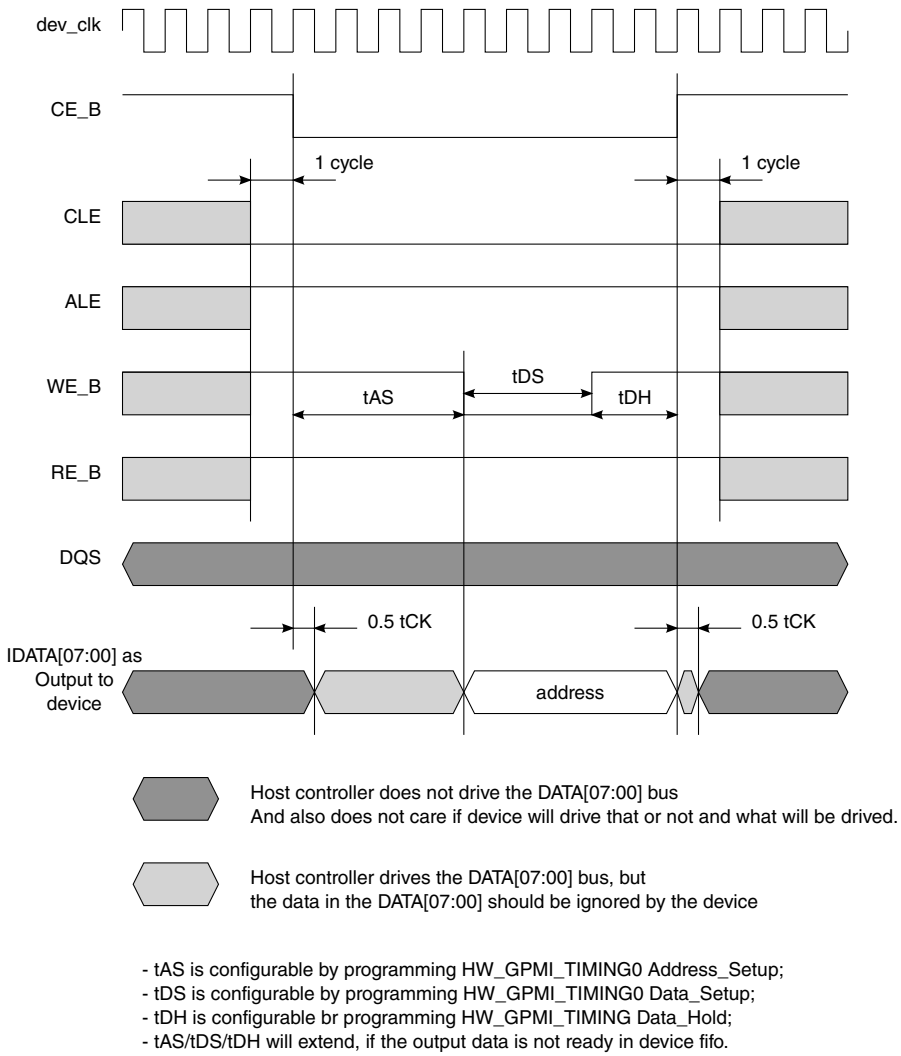


Figure 29-11. Samsung Toggle Mode Basic Address Write Timing Diagram

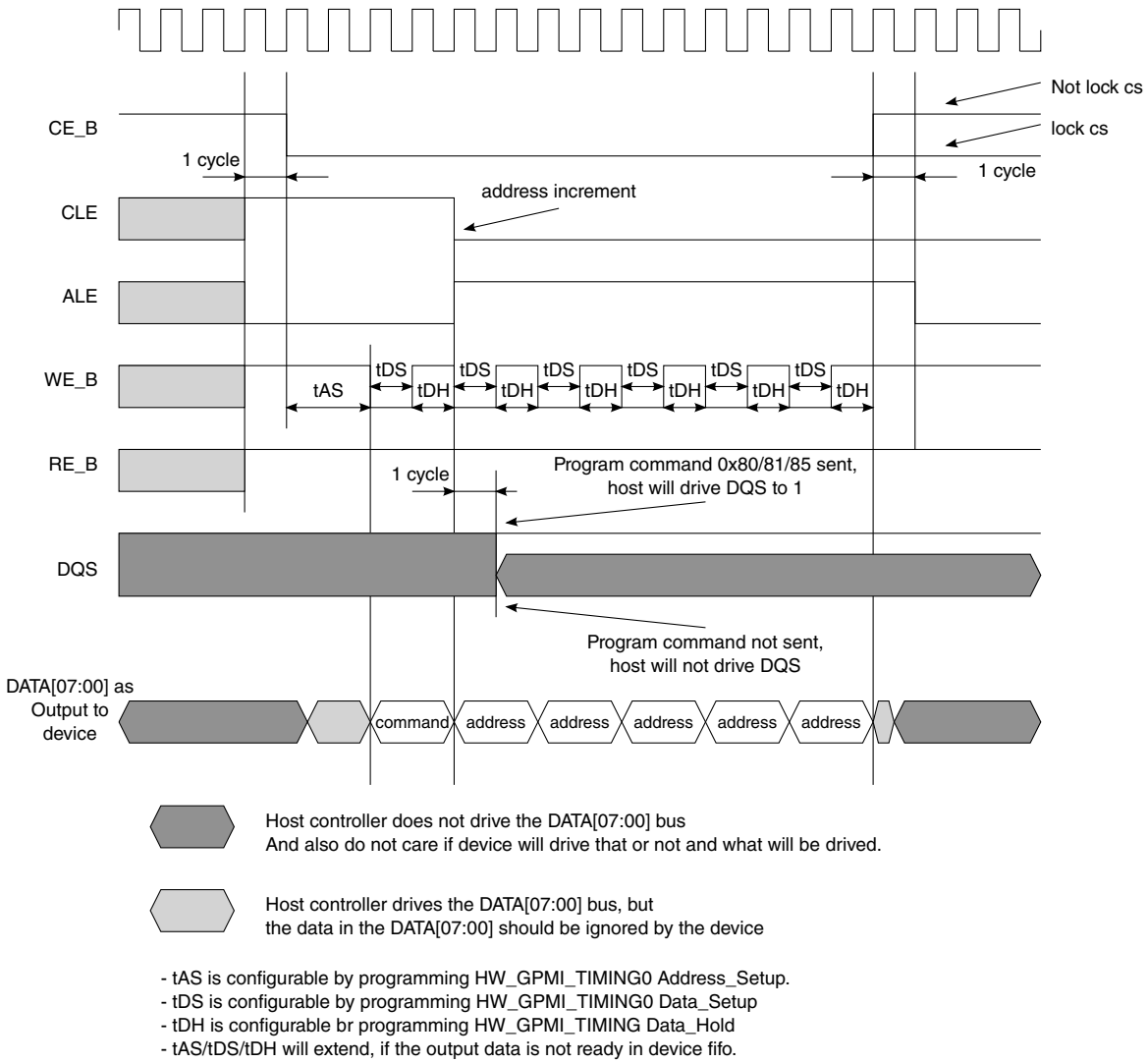


Figure 29-12. Samsung Toggle Mode Basic Command + Address Timing Diagram

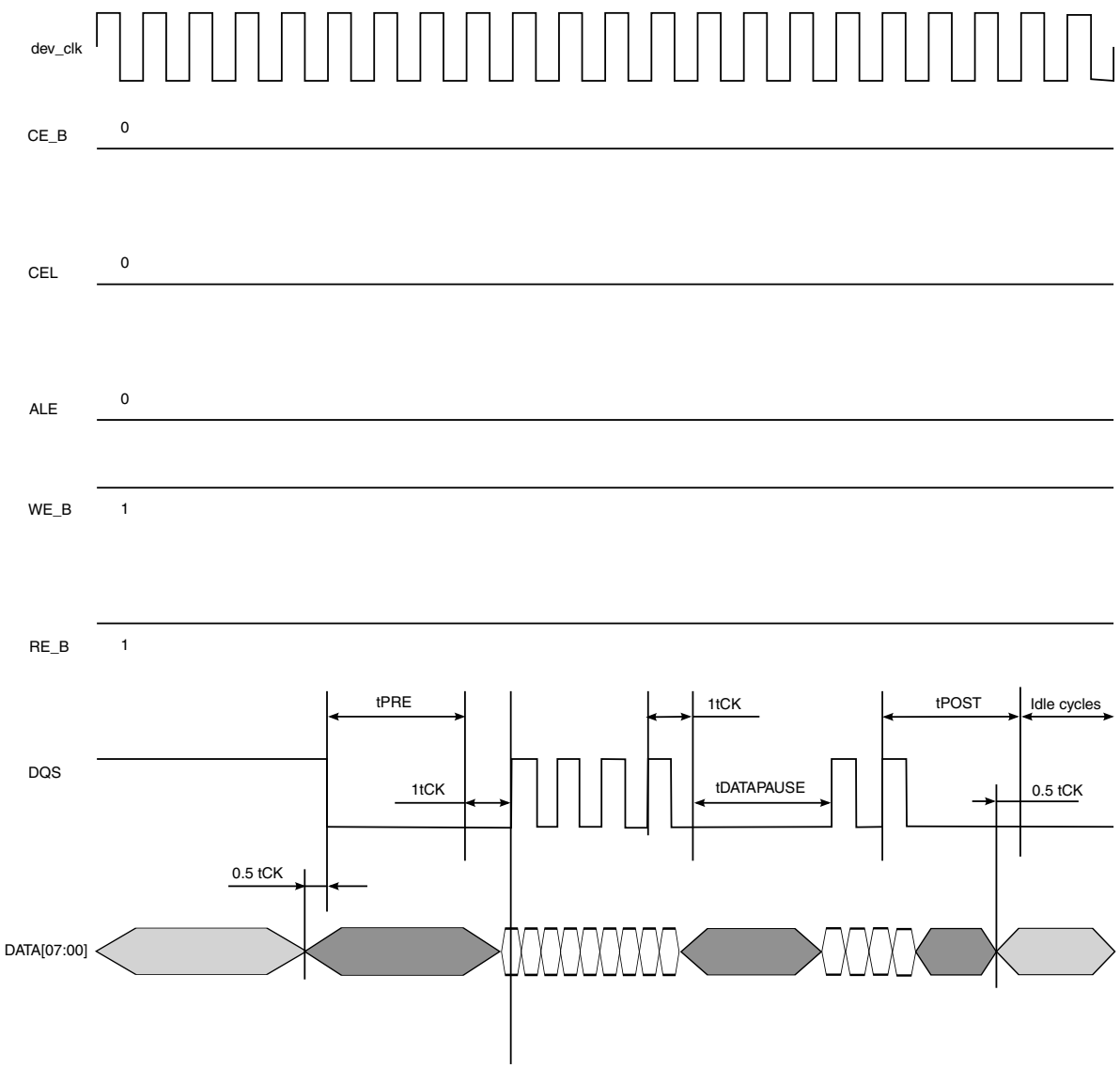
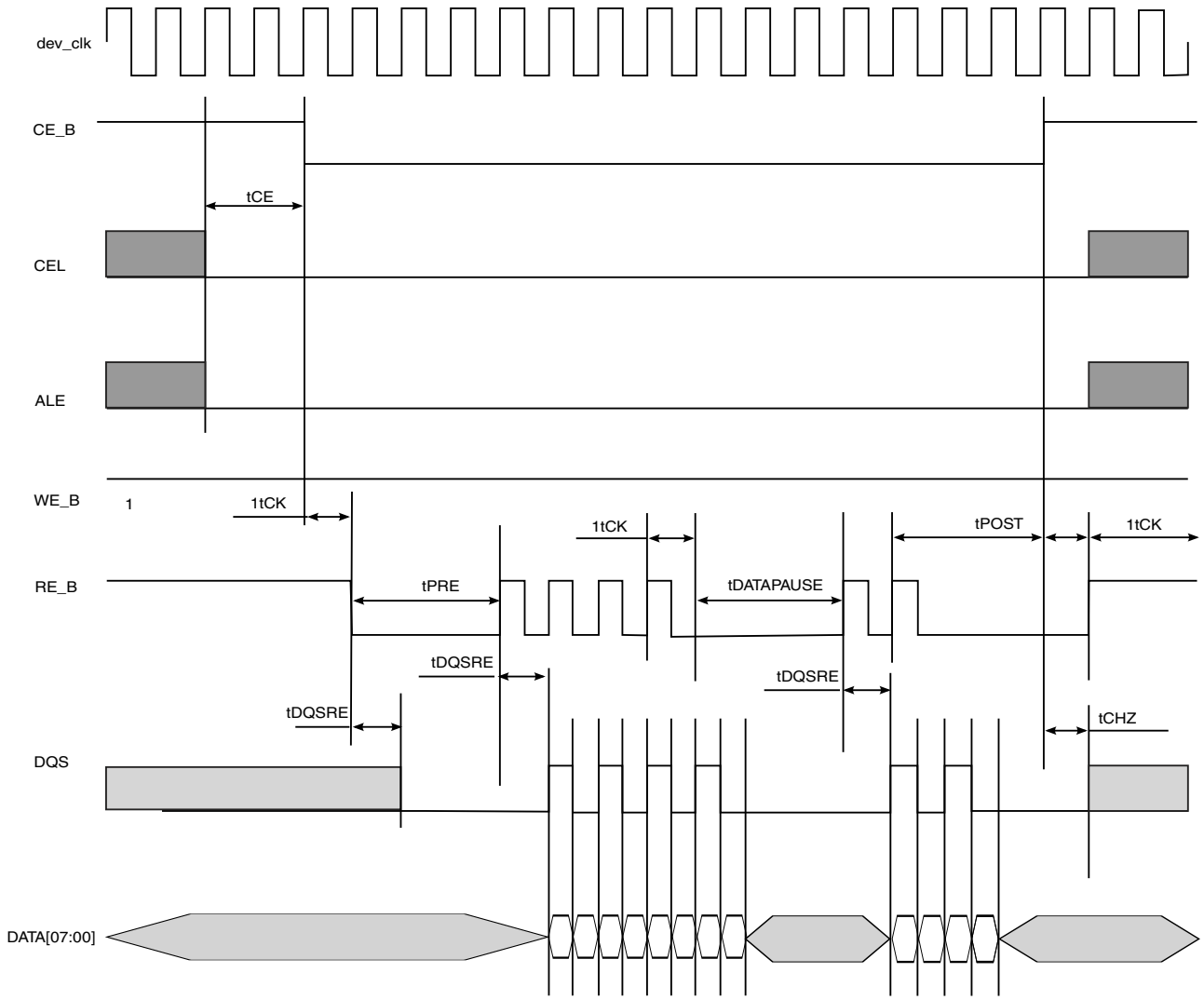


Figure 29-13. Toggle Mode Data Write Timing Diagram



Host controller does not drive the DATA[07:00] bus.
It also does not care if the device will drive or what will be driven.

Host controller drives the DATA[07:00] bus, but
the data in the DATA[07:00] should be ignored by device.

tCK is equal to device clock period

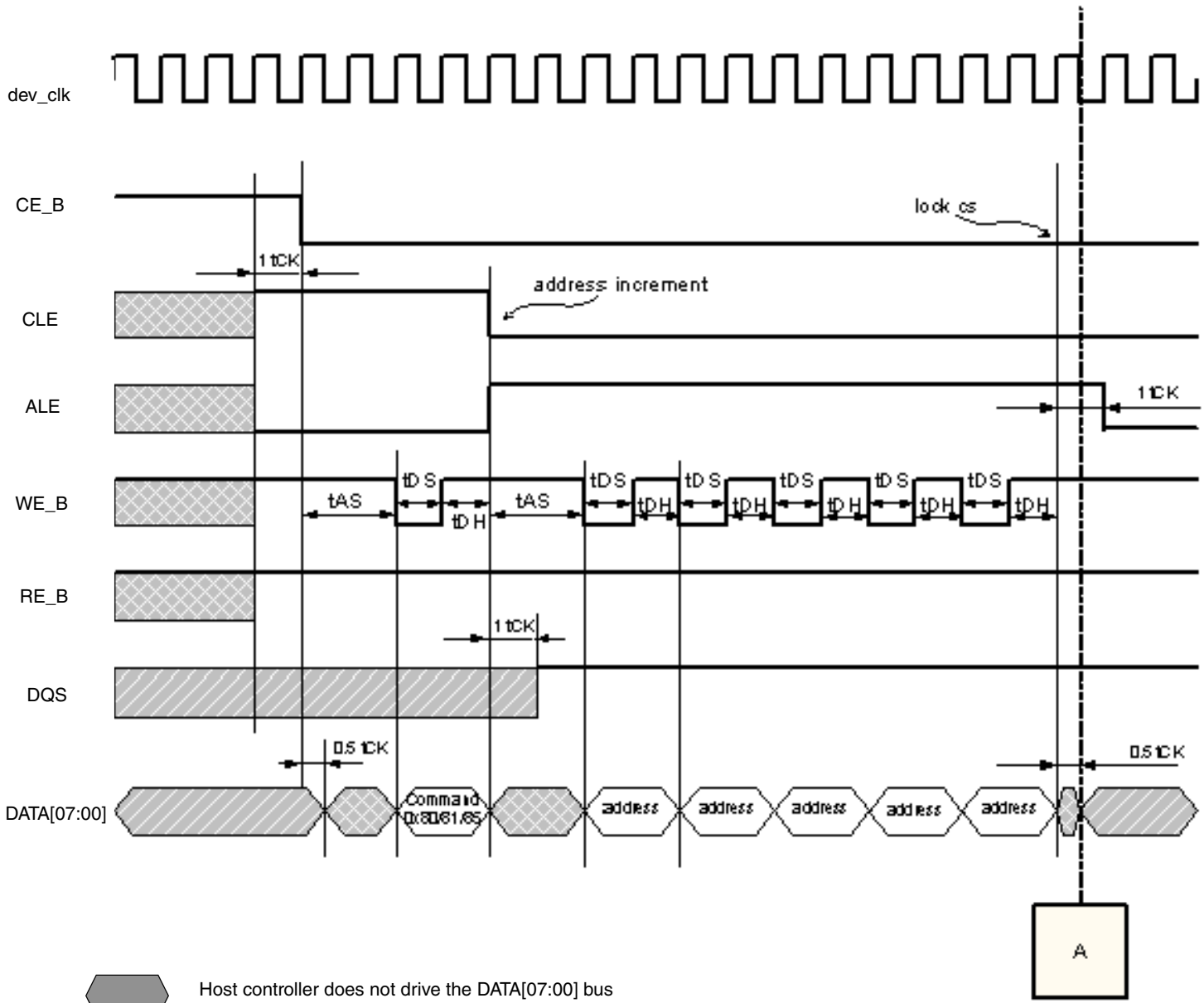
tCE = HW_GPMI_TIMIN G2.CE_DELAY* tCK

tPRE = HW_GPMI_TIMIN G2.PREAMBLE_DELAY* tCK

tPOST = HW_GPMI_TIMIN G2.POSTAMBLE_DELAY* tCK

tDATAPAUSE = HW_GPMI_TIMIN G2.DATA_PAUSE* tCK

Figure 29-14. Toggle Mode Data Read Timing Diagram

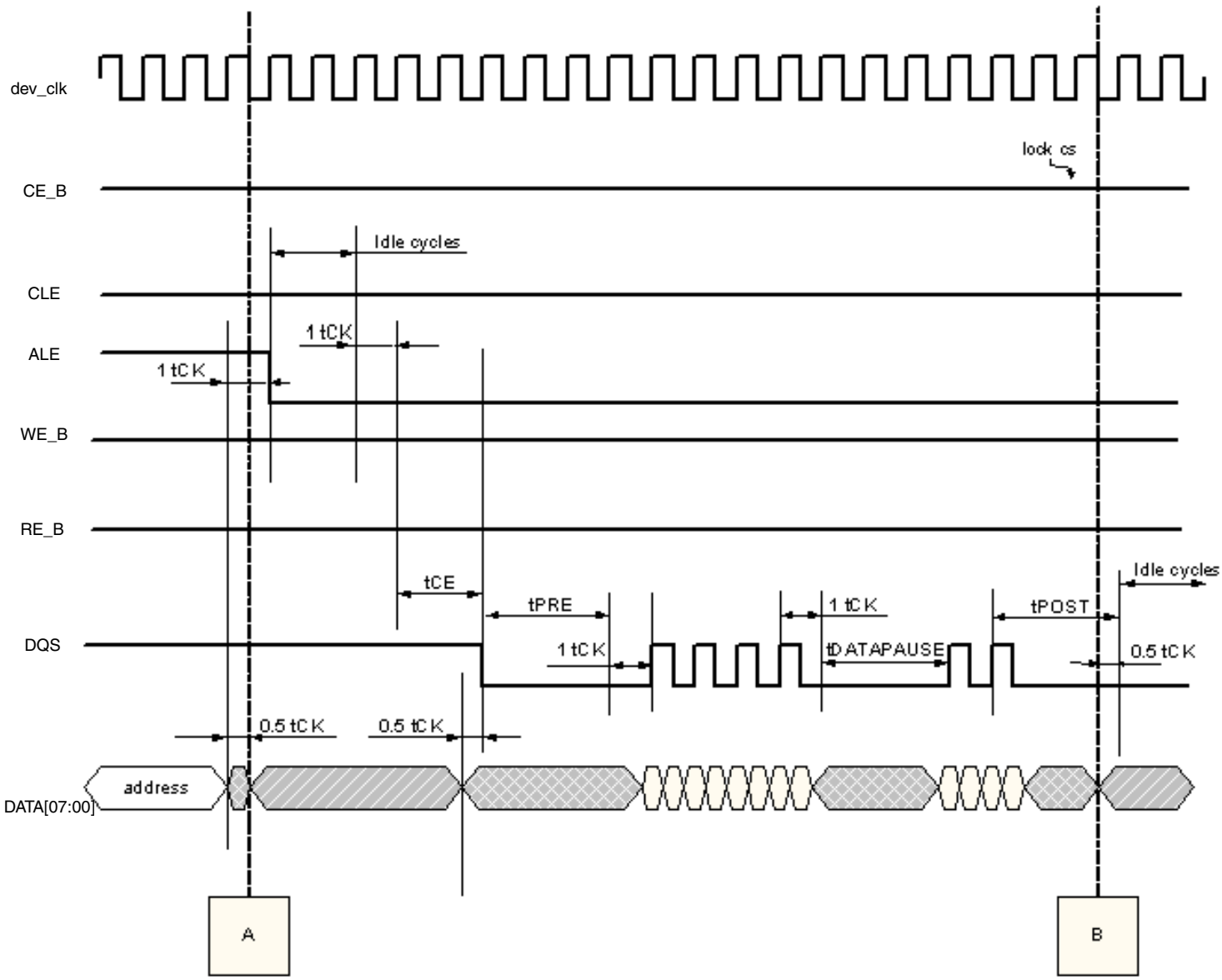


Host controller does not drive the DATA[07:00] bus
And also does not care if device will drive that or not and what will be drived.

Host controller drives the DATA[07:00] bus, but
the data in the DATA[07:00] should be ignored by the device

- tAS is configurable by programming HW_GPMI_TIMING0 Address_Setup:;
- tDS is configurable by programming HW_GPMI_TIMING0 Data_Setup;
- tDH is configuarble by programming HW_GPMI_TIMING0 Data_Hold;
- tAS/tDS/tDH will extend, if the output data is not ready in device fifo.

Figure 29-15. Toggle Mode Program Timing Diagram (A)



Host controller does not drive the DATA[07:00] bus
And also does not care if device will drive that or not and what will be drived.

Host controller drives the DATA[07:00] bus, but
the data in the DATA[07:00] should be ignored by the device

- tCK is equal to device clock period
- tCE = HW_GPMI_TIMING2.CE_DELAY * tCK
- tPRE = HW_GPMI_TIMING2.PREAMBLE_DELAY * tCK
- tPOST = HW_GPMI_TIMING2.POSTAMBLE_DELAY * tCK
- tDATAPULSE = HW_GPMI_TIMING2.DATA_PULSE * tCK

Figure 29-16. Toggle Mode Program Timing Diagram (B)

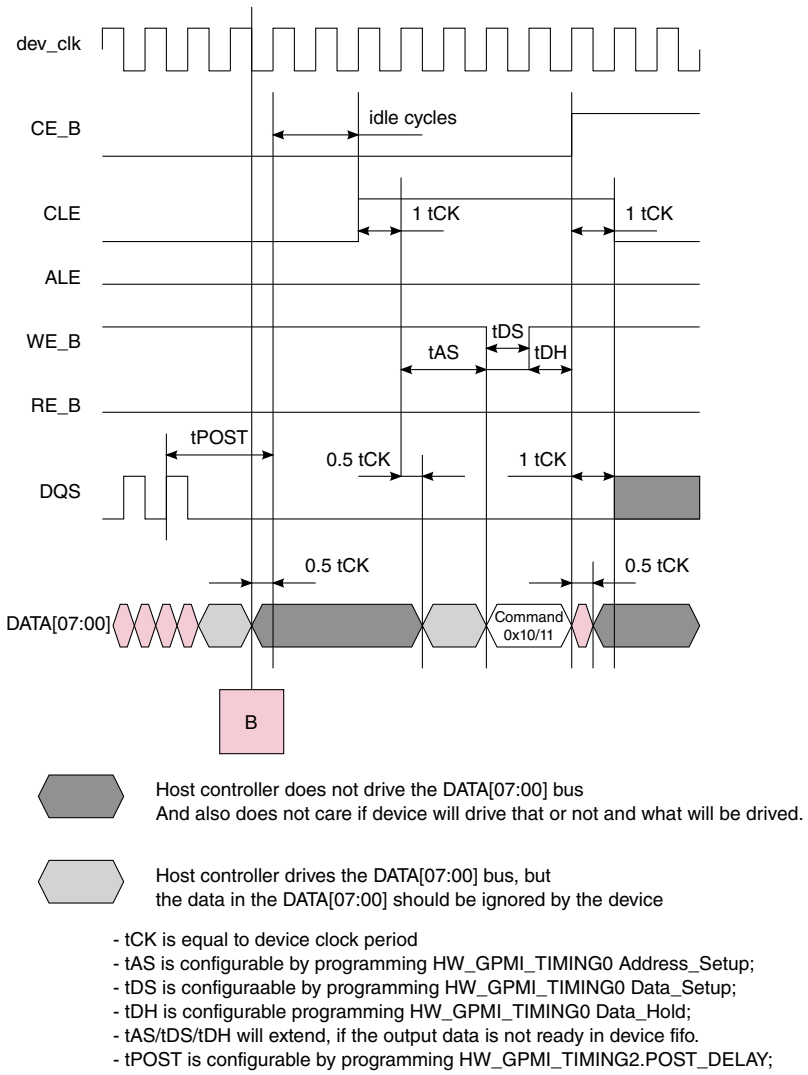


Figure 29-17. Toggle Mode Program Timing Diagram (C)

29.4.4 Hardware BCH Interface

The GPMI provides an interface to the BCH module. This reduces the SOC bus traffic and the software involvement.

When BCH ECC is enable, parity information is inserted on-the-fly during writes to 8-bit NAND devices. The BCH will supply payload and parity to the GPMI to write to the NAND. During NAND reads, parity is checked and ECC processing is performed after each read block. In this case the GPMI reads the NAND device and redirects the data and parity to the BCH module for ECC processing.

To program the BCH for NAND writes, remove the soft reset and clock gates from BCH_CTRL[SFTRST] and BCH_CTRL[CLKGATE]. The bulk of BCH programming is actually applied to the GPMI via PIO operations embedded in its DMA command structures. This has a subtle implication when writing to the GPMI ECC registers: access to the these registers must be written in progressive register order. Thus, to write to the GPMI_ECCCOUNT register, write first (in order) to registers GPMI_CTRL0, GPMI_COMPARE, and GPMI_ECCCTRL before writing to GPMI_ECCCOUNT. These additional register writes need to be accounted for in the CMDWORDS field of the respective DMA channel command register.

Note that the GPMI_PAYLOAD and GPMI_AUXILIARY pointers need to be word-aligned for proper ECC operation. If those pointers are non-word-aligned, then the BCH engine will not operate properly and could possibly corrupt system memory in the adjoining memory regions.

29.5 Behavior During Reset

A soft reset (SFTRST) can take multiple clock periods to complete, so do NOT set CLKGATE when setting SFTRST. The reset process gates the clocks automatically.

29.6 GPMI Memory Map/Register Definition

The following registers provide control for programmable elements of the GPMI module.

NOTE

All ATA or UDMA features are not supported for the chip.

GPMI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_2000	GPMI Control Register 0 Description (GPMI_CTRL0)	32	R/W	C000_0000h	29.6.1/1468
11_2004	GPMI Control Register 0 Description (GPMI_CTRL0_SET)	32	R/W	C000_0000h	29.6.1/1468
11_2008	GPMI Control Register 0 Description (GPMI_CTRL0_CLR)	32	R/W	C000_0000h	29.6.1/1468
11_200C	GPMI Control Register 0 Description (GPMI_CTRL0_TOG)	32	R/W	C000_0000h	29.6.1/1468
11_2010	GPMI Compare Register Description (GPMI_COMPARE)	32	R/W	0000_0000h	29.6.2/1470
11_2020	GPMI Integrated ECC Control Register Description (GPMI_ECCCTRL)	32	R/W	0000_0000h	29.6.3/1471

Table continues on the next page...

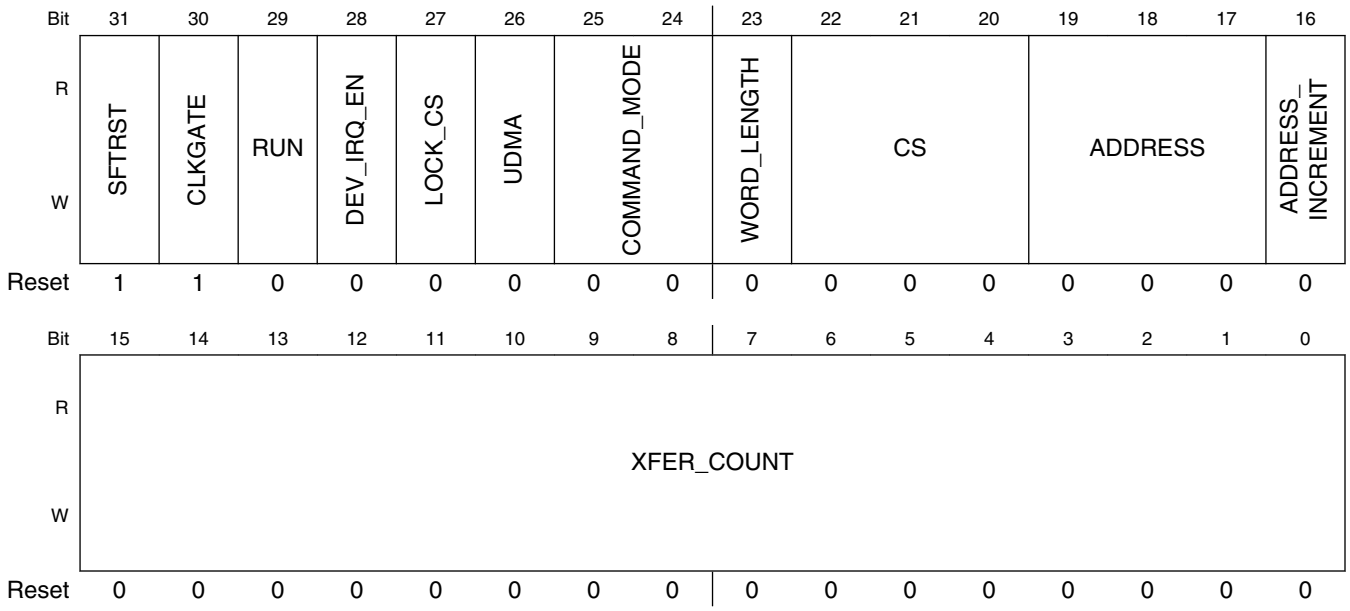
GPMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
11_2024	GPMI Integrated ECC Control Register Description (GPMI_ECCCTRL_SET)	32	R/W	0000_0000h	29.6.3/1471
11_2028	GPMI Integrated ECC Control Register Description (GPMI_ECCCTRL_CLR)	32	R/W	0000_0000h	29.6.3/1471
11_202C	GPMI Integrated ECC Control Register Description (GPMI_ECCCTRL_TOG)	32	R/W	0000_0000h	29.6.3/1471
11_2030	GPMI Integrated ECC Transfer Count Register Description (GPMI_ECCCOUNT)	32	R/W	0000_0000h	29.6.4/1472
11_2040	GPMI Payload Address Register Description (GPMI_PAYLOAD)	32	R/W	0000_0000h	29.6.5/1472
11_2050	GPMI Auxiliary Address Register Description (GPMI_AUXILIARY)	32	R/W	0000_0000h	29.6.6/1473
11_2060	GPMI Control Register 1 Description (GPMI_CTRL1)	32	R/W	0004_0004h	29.6.7/1474
11_2064	GPMI Control Register 1 Description (GPMI_CTRL1_SET)	32	R/W	0004_0004h	29.6.7/1474
11_2068	GPMI Control Register 1 Description (GPMI_CTRL1_CLR)	32	R/W	0004_0004h	29.6.7/1474
11_206C	GPMI Control Register 1 Description (GPMI_CTRL1_TOG)	32	R/W	0004_0004h	29.6.7/1474
11_2070	GPMI Timing Register 0 Description (GPMI_TIMING0)	32	R/W	0001_0203h	29.6.8/1477
11_2080	GPMI Timing Register 1 Description (GPMI_TIMING1)	32	R/W	0000_0000h	29.6.9/1477
11_2090	GPMI Timing Register 2 Description (GPMI_TIMING2)	32	R/W	0302_3336h	29.6.10/ 1478
11_20A0	GPMI DMA Data Transfer Register Description (GPMI_DATA)	32	R/W	0000_0000h	29.6.11/ 1479
11_20B0	GPMI Status Register Description (GPMI_STAT)	32	R	0000_0005h	29.6.12/ 1479
11_20C0	GPMI Debug Information Register Description (GPMI_DEBUG)	32	R	0000_0000h	29.6.13/ 1482
11_20D0	GPMI Version Register Description (GPMI_VERSION)	32	R	0501_0000h	29.6.14/ 1483
11_20E0	GPMI Debug2 Information Register Description (GPMI_DEBUG2)	32	R/W	0000_F100h	29.6.15/ 1483
11_20F0	GPMI Debug3 Information Register Description (GPMI_DEBUG3)	32	R	0000_0000h	29.6.16/ 1486
11_2100	GPMI Double Rate Read DLL Control Register Description (GPMI_READ_DDR_DLL_CTRL)	32	R/W	0000_0038h	29.6.17/ 1487
11_2110	GPMI Double Rate Write DLL Control Register Description (GPMI_WRITE_DDR_DLL_CTRL)	32	R/W	0000_0038h	29.6.18/ 1488
11_2120	GPMI Double Rate Read DLL Status Register Description (GPMI_READ_DDR_DLL_STS)	32	R	0000_0000h	29.6.19/ 1490
11_2130	GPMI Double Rate Write DLL Status Register Description (GPMI_WRITE_DDR_DLL_STS)	32	R	0000_0000h	29.6.20/ 1491

29.6.1 GPMI Control Register 0 Description (GPMI_CTRL0*n*)

The GPMI control register 0 specifies the GPMI transaction to perform for the current command chain item.

Address: 11_2000h base + 0h offset + (4d × i), where i=0d to 3d



GPMI_CTRL0*n* field descriptions

Field	Description
31 SFTRST	Set to zero for normal operation. When this bit is set to one (default), then the entire block is held in its reset state. This will not work if the CLKGATE bit is already set to '1'. CLKGATE must be cleared to '0' before issuing a soft reset. Also the GPMICLK must be running for this to work properly. RUN = 0x0 Allow GPMI to operate normally. RESET = 0x1 Hold GPMI in reset.
30 CLKGATE	Set this bit zero for normal operation. Setting this bit to one (default), gates all of the block level clocks off for miniminizing AC energy consumption. RUN = 0x0 Allow GPMI to operate normally. NO_CLKS = 0x1 Do not clock GPMI gates in order to minimize power consumption.
29 RUN	The GPMI is busy running a command whenever this bit is set to '1'. The GPMI is idle whenever this bit set to zero. This can be set to one by a CPU write. In addition, the DMA sets this bit each time a DMA command has finished its PIO transfer phase. IDLE = 0x0 The GPMI is idle. BUSY = 0x1 The GPMI is busy running a command.

Table continues on the next page...

GPMI_CTRL0*n* field descriptions (continued)

Field	Description
28 DEV_IRQ_EN	When set to '1' and ATA_IRQ pin is asserted, the GPMI_IRQ output will assert.
27 LOCK_CS	For ATA/NAND mode: 0= Deassert chip select (CS) after RUN is complete. 1= Continue to assert chip select (CS) after RUN is complete. For Camera Mode: 0= Dont wait for VSYNC rising edge before capturing data. 1= Wait for VSYNC rising edge before capturing data (Camera mode only). DISABLED = 0x0 Deassert chip select (CS) after RUN is complete. ENABLED = 0x1 Continue to assert chip select (CS) after RUN is complete.
26 UDMA	DISABLED = 0x0 Use ATA-PIO mode on the external bus. ENABLED = 0x1 Use ATA-Ultra DMA mode on the external bus. 0 Use ATA-PIO mode on the external bus. 1 Use ATA-Ultra DMA mode on the external bus.
25–24 COMMAND_ MODE	WRITE = 0x0 Write mode. READ = 0x1 Read mode. READ_AND_COMPARE = 0x2 Read and Compare mode (setting sense flop). WAIT_FOR_READY = 0x3 Wait for Ready mode. For ATA WAIT_FOR_READY command set CS=01. 00 Write mode. 01 Read Mode. 10 Read and Compare Mode (setting sense flop). 11 Wait for Ready.
23 WORD_LENGTH	This bit should only be changed when RUN==0. Reserve = 0x0 Reserved. 8_BIT = 0x1 8-bit Data Bus mode. 0 Reserved. 1 8-bit Data Bus mode.
22–20 CS	Selects which chip select is active for this command. For ATA WAIT_FOR_READY command, this must be set to b01.
19–17 ADDRESS	Specifies the three address lines for ATA mode. In NAND mode, use A0 for CLE and A1 for ALE. NAND_DATA = 0x0 In NAND mode, this address is used to read and write data bytes. NAND_CLE = 0x1 In NAND mode, this address is used to write command bytes. NAND_ALE = 0x2 In NAND mode, this address is used to write address bytes.
16 ADDRESS_ INCREMENT	In ATA mode, the address will increment with each cycle. In NAND mode, the address will increment once, after the first cycle (going from CLE to ALE). DISABLED = 0x0 Address does not increment. ENABLED = 0x1 Increment address. 0 Address does not increment. 1 Increment address.
XFER_COUNT	Number of bytes to transfer for this command. A value of zero will transfer 64K bytes.

29.6.2 GPMI Compare Register Description (GPMI_COMPARE)

The GPMI compare register specifies the expect data and the xor mask for comparing to the status values read from the device. This register is used by the Read and Compare command.

GPMI_COMPARE 0x010

Address: 11_2000h base + 10h offset = 11_2010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MASK																REFERENCE															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

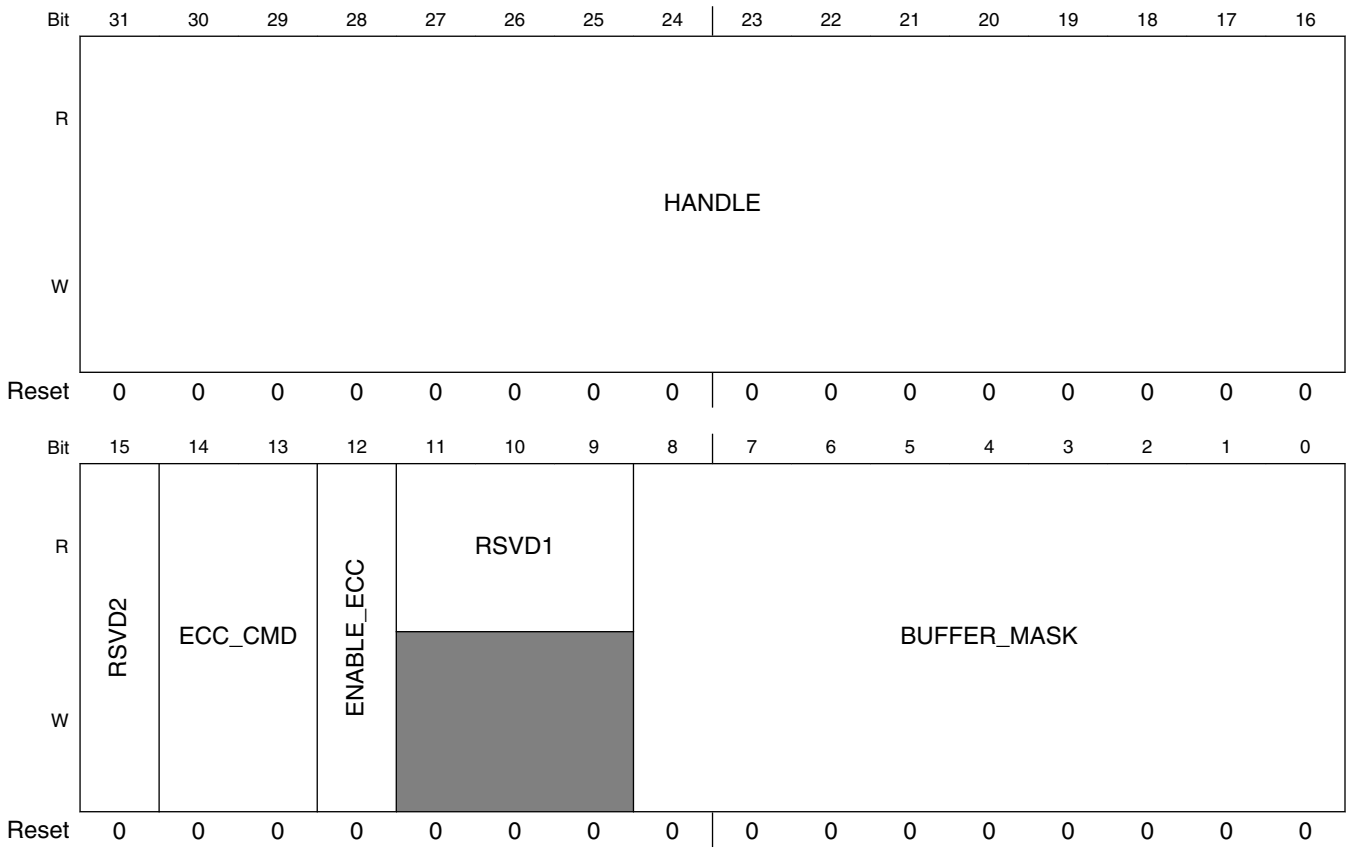
GPMI_COMPARE field descriptions

Field	Description
31–16 MASK	16-bit mask which is applied after the read data is XORed with the REFERENCE bit field.
REFERENCE	16-bit value which is XORed with data read from the NAND device.

29.6.3 GPMI Integrated ECC Control Register Description (GPMI_ECCCTRL*n*)

The GPMI ECC control register handles configuration of the integrated ECC accelerator.

Address: 11_2000h base + 20h offset + (4d × i), where i=0d to 3d



GPMI_ECCCTRL*n* field descriptions

Field	Description
31–16 HANDLE	This is a register available to software to attach an identifier to a transaction in progress. This handle will be available from the ECC register space when the completion interrupt occurs.
15 RSVD2	Always write zeroes to this bit field.
14–13 ECC_CMD	ECC Command information. DECODE = 0x0 Decode. ENCODE = 0x1 Encode. RESERVE2 = 0x2 Reserved. RESERVE3 = 0x3 Reserved.

Table continues on the next page...

GPMI_ECCCTRL*n* field descriptions (continued)

Field	Description
12 ENABLE_ECC	Enable ECC processing of GPMI transfers. ENABLE = 0x1 Use integrated ECC for read and write transfers. DISABLE = 0x0 Integrated ECC remains in idle.
11–9 RSVD1	Always write zeroes to this bit field.
BUFFER_MASK	ECC buffer information. The BCH error correction only allows two configurations of the buffer mask - software may either read just the first block on the flash page or the entire flash page. Write operations must be for the entire flash page. Invalid buffer mask values will cause the DMA descriptor command to be terminated. BCH_AUXONLY = 0x100 Set to request transfer from only the auxiliary buffer (block 0 on flash). BCH_PAGE = 0x1FF Set to request transfer to/from the entire page.

29.6.4 GPMI Integrated ECC Transfer Count Register Description (GPMI_ECCCOUNT)

The GPMI ECC Transfer Count Register contains the count of bytes that flow through the ECC subsystem.

GPMI_ECCCOUNT 0x030

Address: 11_2000h base + 30h offset = 11_2030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RSVD2																COUNT															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_ECCCOUNT field descriptions

Field	Description
31–16 RSVD2	Always write zeroes to this bit field.
COUNT	Number of bytes to pass through ECC. This is the GPMI transfer count plus the syndrome count that will be inserted into the stream by the ECC. In DMA2ECC_MODE this count must match the GPMI_CTRL0_XFER_COUNT. A value of zero will transfer 64K words.

29.6.5 GPMI Payload Address Register Description (GPMI_PAYLOAD)

The GPMI payload address register specifies the location of the data buffers in system memory. This value must be word aligned.

GPMI_PAYLOAD 0x040

Address: 11_2000h base + 40h offset = 11_2040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	ADDRESS															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ADDRESS														RSVD0	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_PAYLOAD field descriptions

Field	Description
31–2 ADDRESS	Pointer to an array of one or more 512 byte payload buffers.
RSVD0	Always write zeroes to this bit field.

29.6.6 GPMI Auxiliary Address Register Description (GPMI_AUXILIARY)

The GPMI auxiliary address register specifies the location of the auxiliary buffers in system memory. This value must be word aligned.

GPMI_AUXILIARY 0x050

Address: 11_2000h base + 50h offset = 11_2050h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	ADDRESS															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ADDRESS														RSVD0	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

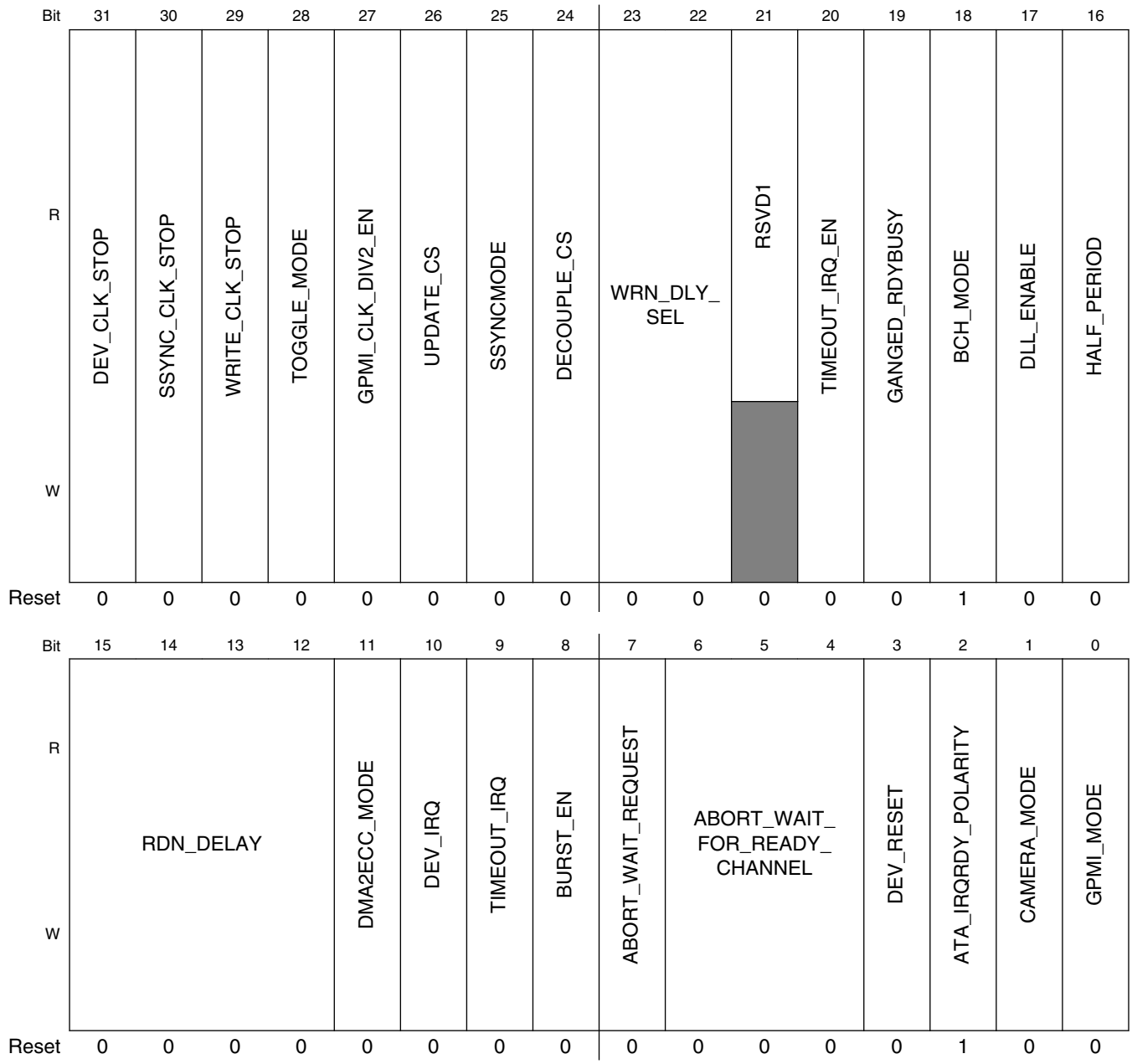
GPMI_AUXILIARY field descriptions

Field	Description
31–2 ADDRESS	Pointer to ECC control structure and meta-data storage.
RSVD0	Always write zeroes to this bit field.

29.6.7 GPMI Control Register 1 Description (GPMI_CTRL1*n*)

The GPMI control register 1 specifies additional control fields that are not used on a per-transaction basis.

Address: 11_2000h base + 60h offset + (4d × i), where i=0d to 3d



GPMI_CTRL1*n* field descriptions

Field	Description
31 DEV_CLK_STOP	set this bit to 1 will stop gpmi io working clk.
30 SSYNC_CLK_ STOP	set this bit to 1 will stop the source synchronous mode clk.
29 WRITE_CLK_ STOP	In onfi source synchronous mode, host may save power during the data write cycles by holding the CLK signal high (i.e. stopping the CLK). The host may only stop the CLK during data write, by setting this bit to 1, if the device supports this feature as indicated in the parameter page.
28 TOGGLE_MODE	enable samsung toggle mode.
27 GPMI_CLK_ DIV2_EN	This bit should be reset to 0 in asynchronous mode. The frequence ratio of (device clock : ccm gpmi clock) will be (1 : 1). This bit should be set to 1, in source synchronous mode or toggle mode. The frequence raito of (device clock : ccm gpmi clock) will be (1 : 2). enable the gpmi clk divider. 0x0 internal factor-2 clock divider is disabled 0x1 internal factor-2 clock divider is enabled.
26 UPDATE_CS	force the CS value is be updated to external chip select pin, even GPMI is idle.
25 SSYNCMODE	source synchronouse mode 1 or asynchrous mode 0. ASYNC = 0x0 Asynchronous mode. SSYNC = 0x1 Source Synchronous mode.
24 DECOUPLE_CS	Decouple Chip Select from DMA Channel. Setting this bit to 1 will allow a DMA channel to specify any value in the CTRL0_CS register field. Software can use one DMA channel to access all 8 Nand devices.
23–22 WRN_DLY_SEL	Since the GPMI write strobe (WRN) is a fast clock pin, the delay on this signal can be programmed to match the load on this pin. 0 = 4ns to 8ns; 1 = 6ns to 10ns; 2 = 7ns to 12ns; 3 = no delay.
21 RSVD1	Always write zeroes to this bit field.
20 TIMEOUT_IRQ_ EN	Setting this bit to '1' will enable timeout IRQ for transfers in ATA mode only, and for WAIT_FOR_READY commands in both ATA and Nand mode. The Device_Busy_Timeout value is used for this timeout.
19 GANGED_ RDYBUSY	Set this bit to 1 will force all Nand RDY_BUSY inputs to be sourced from (tied to) RDY_BUSY0. This will free up all, except one, RDY_BUSY input pins.
18 BCH_MODE	This bit selects which error correction unit will access GPMI. This bit must always be set to '1', since only the BCH unit is available in this design.
17 DLL_ENABLE	Set this bit to 1 to enable the GPMI DLL. This is required for fast NAND reads (above 30 MHz read strobe). After setting this bit, wait 64 GPMI clock cycles for the DLL to lock before performing a NAND read.
16 HALF_PERIOD	Set this bit to 1 if the GPMI clock period is greater than 16ns for proper DLL operation. DLL_ENABLE must be zero while changing this field.
15–12 RDN_DELAY	This variable is a factor in the calculated delay to apply to the internal read strobe for correct read data sampling.

Table continues on the next page...

GPMI_CTRL1*n* field descriptions (continued)

Field	Description
	The applied delay (AD) is between 0 and 1.875 times the reference period (RP). RP is one half of the GPMI clock period if HALF_PERIOD=1 otherwise it is the full GPMI clock period. The equation is: AD = RDN_DELAY x 0.125 x RP. This value must not exceed 16ns. This variable is used to achieve faster NAND access. For example if the Read Strobe is asserted from time 0 to 13ns but the read access time is 20ns, then choose AD=12ns will cause the data to be sampled at time 25ns (13+12) giving a 5ns data setup time. If RP=13ns then RDN_DELAY = 12/(0.125 x 13ns) = 7.38 (0111b). DLL_ENABLE must be zero while changing this field.
11 DMA2ECC_MODE	This is mainly for testing HWECC without involving the Nand device. Setting this bit will cause DMA write data to redirected to HWECC module (instead of Nand Device) for encoding or decoding.
10 DEV_IRQ	This bit is set when an Interrupt is received from the ATA device. Write 0 to clear.
9 TIMEOUT_IRQ	This bit is set when a timeout occurs using the Device_Busy_Timeout value. Write 0 to clear.
8 BURST_EN	When set to 1 each DMA request will generate a 4-transfer burst on the APB bus.
7 ABORT_WAIT_REQUEST	Request to abort "wait for ready" command on channel indicated by ABORT_WAIT_FOR_READY_CHANNEL. Hardware will clear this bit when abort is done.
6–4 ABORT_WAIT_FOR_READY_CHANNEL	Abort a wait for ready command on selected channel. Set the ABORT_WAIT_REQUEST to kick of operation.
3 DEV_RESET	ENABLED = 0x0 NANDF_WP_B(WPN) pin is held low (asserted). DISABLED = 0x1 NANDF_WP_B(WPN) pin is held high (de-asserted). 0 NANDF_WP_B pin is held low (asserted). 1 NANDF_WP_B pin is held high (de-asserted).
2 ATA_IRQRDY_POLARITY	For ATA MODE: Note NAND_RDY_BUSY[3:2] are not affected by this bit. ACTIVELOW = 0x0 ATA IORDY and IRQ are active low, or NAND_RDY_BUSY[1:0] are active low ready. ACTIVEHIGH = 0x1 ATA IORDY and IRQ are active high, or NAND_RDY_BUSY[1:0] are active high ready. 0 External ATA IORDY and IRQ are active low. 1 External ATA IORDY and IRQ are active high. For NAND MODE: 0 External RDY_BUSY[1] and RDY_BUSY[0] pins are ready when low and busy when high. 1 External RDY_BUSY[1] and RDY_BUSY[0] pins are ready when high and busy when low.
1 CAMERA_MODE	When set to 1 and ATA UDMA is enabled the UDMA interface becomes a camera interface.
0 GPMI_MODE	ATA mode is only supported on channel zero. If ATA mode is selected, then only channel three is available for NAND use.

Table continues on the next page...

GPMI_CTRL1*n* field descriptions (continued)

Field	Description
	NAND = 0x0 NAND mode. ATA = 0x1 ATA mode. 0 NAND mode. 1 ATA mode.

29.6.8 GPMI Timing Register 0 Description (GPMI_TIMING0)

The GPMI timing register 0 specifies the timing parameters that are used by the cycle state machine to guarantee the various setup, hold and cycle times for the external media type.

GPMI_TIMING0 0x070

Address: 11_2000h base + 70h offset = 11_2070h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R									ADDRESS_SETUP								DATA_HOLD								DATA_SETUP							
W	0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1

GPMI_TIMING0 field descriptions

Field	Description
31–24 RSVD1	Always write zeroes to this bit field.
23–16 ADDRESS_ SETUP	Number of GPMICLK cycles that the CE/ADDR signals are active before a strobe is asserted. A value of zero is interpreted as 0. For ATA PIO modes this is known in the ATA7 specification as "Address valid to DIOR-/DIOW- setup"
15–8 DATA_HOLD	Data bus hold time in GPMICLK cycles. Also the time that the data strobe is de-asserted in a cycle. A value of zero is interpreted as 256. For ATA PIO modes this is known in the ATA7 specification as "DIOR-/DIOW- recovery time"
DATA_SETUP	Data bus setup time in GPMICLK cycles. Also the time that the data strobe is asserted in a cycle. This value must be greater than 2 for ATA devices that use IORDY to extend transfer cycles. A value of zero is interpreted as 256. For ATA PIO modes this is known in the ATA7 specification as ""DIOR-/DIOW-"

29.6.9 GPMI Timing Register 1 Description (GPMI_TIMING1)

The GPMI timing register 1 specifies the timeouts used when monitoring the NAND READY pin or the ATA IRQ and IOWAIT signals.

GPMI_TIMING1 0x080

Address: 11_2000h base + 80h offset = 11_2080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DEVICE_BUSY_TIMEOUT																RSVD1																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_TIMING1 field descriptions

Field	Description
31–16 DEVICE_BUSY_TIMEOUT	Timeout waiting for NAND Ready/Busy or ATA IRQ. Used in WAIT_FOR_READY mode. This value is the number of GPMI_CLK cycles multiplied by 4096.
RSVD1	Always write zeroes to this bit field.

29.6.10 GPMI Timing Register 2 Description (GPMI_TIMING2)

The GPMI timing register 2 specifies the double data rate timing parameters that are used by the cycle state machine to guarantee the various cs delay, pre-amble delay, post-amble delay, command/address delay, data delay, and read latency cycle times for the external media type.

GPMI_TIMING2 0x090

Address: 11_2000h base + 90h offset = 11_2090h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RSVD1					READ_LATENCY			RSVD0			CE_DELAY					PREAMBLE_DELAY				POSTAMBLE_DELAY				CMDADD_PAUSE				DATA_PAUSE				
W																																
Reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	1	1	0

GPMI_TIMING2 field descriptions

Field	Description
31–27 RSVD1	Always write zeroes to this bit field.
26–24 READ_LATENCY	This field is for double data rate read latency configuration. others READ LATENCY is 3 000 READ LATENCY is 0 001 READ LATENCY is 1 010 READ LATENCY is 2 011 READ LATENCY is 3 100 READ LATENCY is 4 101 READ LATENCY is 5
23–21 RSVD0	Always write zeroes to this bit field.

Table continues on the next page...

GPMI_TIMING2 field descriptions (continued)

Field	Description
20–16 CE_DELAY	GPMI dealy from CEn assert to W/Rn changing edge. value of zero is interpreted as 32.
15–12 PREAMBLE_ DELAY	GPMI pre-amble delay in GPMICLK cycles. A value of zero is interpreted as 16.
11–8 POSTAMBLE_ DELAY	GPMI post-amble delay in GPMICLK cycles. A value of zero is interpreted as 16.
7–4 CMDADD_ PAUSE	GPMI delay time from command or addres pause to command or address resume in GPMICLK cycles. A value of zero is interpreted as 16.
DATA_PAUSE	GPMI delay time from data pause to data resume in GPMICLK cycles. A value of zero is interpreted as 16.

29.6.11 GPMI DMA Data Transfer Register Description (GPMI_DATA)

The GPMI DMA data transfer register is used by the DMA to read or write data to or from the ATA/NAND control state machine.

GPMI_DATA 0x0A0

Address: 11_2000h base + A0h offset = 11_20A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_DATA field descriptions

Field	Description
DATA	In 8-bit mode, one, two, three or four bytes can can be accessed to send the same number of bus cycles.

29.6.12 GPMI Status Register Description (GPMI_STAT)

The GPMI control and status register provides a read back path for various operational states of the GPMI controller.

GPMI_STAT 0x0B0

Address: 11_2000h base + B0h offset = 11_20B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	READY_BUSY								RDY_TIMEOUT							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DEV7_ERROR	DEV6_ERROR	DEV5_ERROR	DEV4_ERROR	DEV3_ERROR	DEV2_ERROR	DEV1_ERROR	DEV0_ERROR	RSVD1			ATA_IRQ	INVALID_BUFFER_MASK	FIFO_EMPTY	FIFO_FULL	PRESENT
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

GPMI_STAT field descriptions

Field	Description
31–24 READY_BUSY	Read-only view of NAND Ready_Busy Input pins.
23–16 RDY_TIMEOUT	State of the RDY/BUSY Timeout Flags. When any bit is set to '1' in this field, it indicates that a time out has occurred while waiting for the ready state of the requested NAND device. Multiple bits may be set simultaneously. When GPMI_CTRL1_DECOUPLE_CS = 0, RDY_TIMEOUT[n] is associated with the NAND device on chip_select[n]. When GPMI_CTRL1_DECOUPLE_CS = 1, these flags become associated to a DMA channel instead of a NAND device. For example if DMA channel 6 sends a WAIT_FOR_READY command for NAND Device 2, and a timeout occured on READY_BUSY2, then READY_TIMEOUT[6] will be set instead of READY_TIMEOUT[2].
15 DEV7_ERROR	DMA channel 7 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 7. 1 An Error has occurred on ATA/NAND Device accessed by
14 DEV6_ERROR	DMA channel 6 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 6. 1 An Error has occurred on ATA/NAND Device accessed by
13 DEV5_ERROR	DMA channel 5 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 5. 1 An Error has occurred on ATA/NAND Device accessed by
12 DEV4_ERROR	DMA channel 4 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 4. 1 An Error has occurred on ATA/NAND Device accessed by
11 DEV3_ERROR	DMA channel 3 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 3. 1 An Error has occurred on ATA/NAND Device accessed by
10 DEV2_ERROR	DMA channel 2 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 2. 1 An Error has occurred on ATA/NAND Device accessed by
9 DEV1_ERROR	DMA channel 1 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 1. 1 An Error has occurred on ATA/NAND Device accessed by
8 DEV0_ERROR	DMA channel 0 (Timeout or compare failure, depending on COMMAND_MODE). 0 No error condition present on ATA/NAND Device accessed by DMA channel 0. 1 An Error has occurred on ATA/NAND Device accessed by
7–5 RSVD1	Always write zeroes to this bit field.

Table continues on the next page...

GPMI_STAT field descriptions (continued)

Field	Description
4 ATA_IRQ	Status of the ATA_IRQ input pin.
3 INVALID_ BUFFER_MASK	Buffer Mask Validity bit. 0 ECC Buffer Mask is not invalid. 1 ECC Buffer Mask is invalid.
2 FIFO_EMPTY	NOT_EMPTY = 0x0 FIFO is not empty. EMPTY = 0x1 FIFO is empty. 0 FIFO is not empty. 1 FIFO is empty.
1 FIFO_FULL	NOT_FULL = 0x0 FIFO is not full. FULL = 0x1 FIFO is full. 0 FIFO is not full. 1 FIFO is full.
0 PRESENT	UNAVAILABLE = 0x0 GPMI is not present in this product. AVAILABLE = 0x1 GPMI is present in this product. 0 GPMI is not present in this product. 1 GPMI is present is in this product.

29.6.13 GPMI Debug Information Register Description (GPMI_DEBUG)

The GPMI debug information register provides a read back path for diagnostics to determine the current operating state of the GPMI controller.

GPMI_DEBUG 0x0C0

Address: 11_2000h base + C0h offset = 11_20C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	WAIT_FOR_READY_END								DMA_SENSE								DMAREQ								CMD_END							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_DEBUG field descriptions

Field	Description
31–24 WAIT_FOR_ READY_END	Read Only view of the Wait_For_Ready End toggle signals to DMA. One per channel

Table continues on the next page...

GPMI_DEBUG field descriptions (continued)

Field	Description
23–16 DMA_SENSE	Read-only view of sense state of the 8 DMA channels. A value of "1" in any bit position indicates that a read and compare command failed or a timeout occured for the corresponding channel.
15–8 DMAREQ	Read-only view of DMA request line for 8 DMA channels. A toggle on any bit position indicates a DMA request for the corresponding channel.
CMD_END	Read Only view of the Command End toggle signals to DMA. One per channel

29.6.14 GPMI Version Register Description (GPMI_VERSION)

This register reflects the version number for the GPMI.

GPMI_VERSION 0x0D0

Address: 11_2000h base + D0h offset = 11_20D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	MAJOR								MINOR								STEP															
W																																
Reset	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_VERSION field descriptions

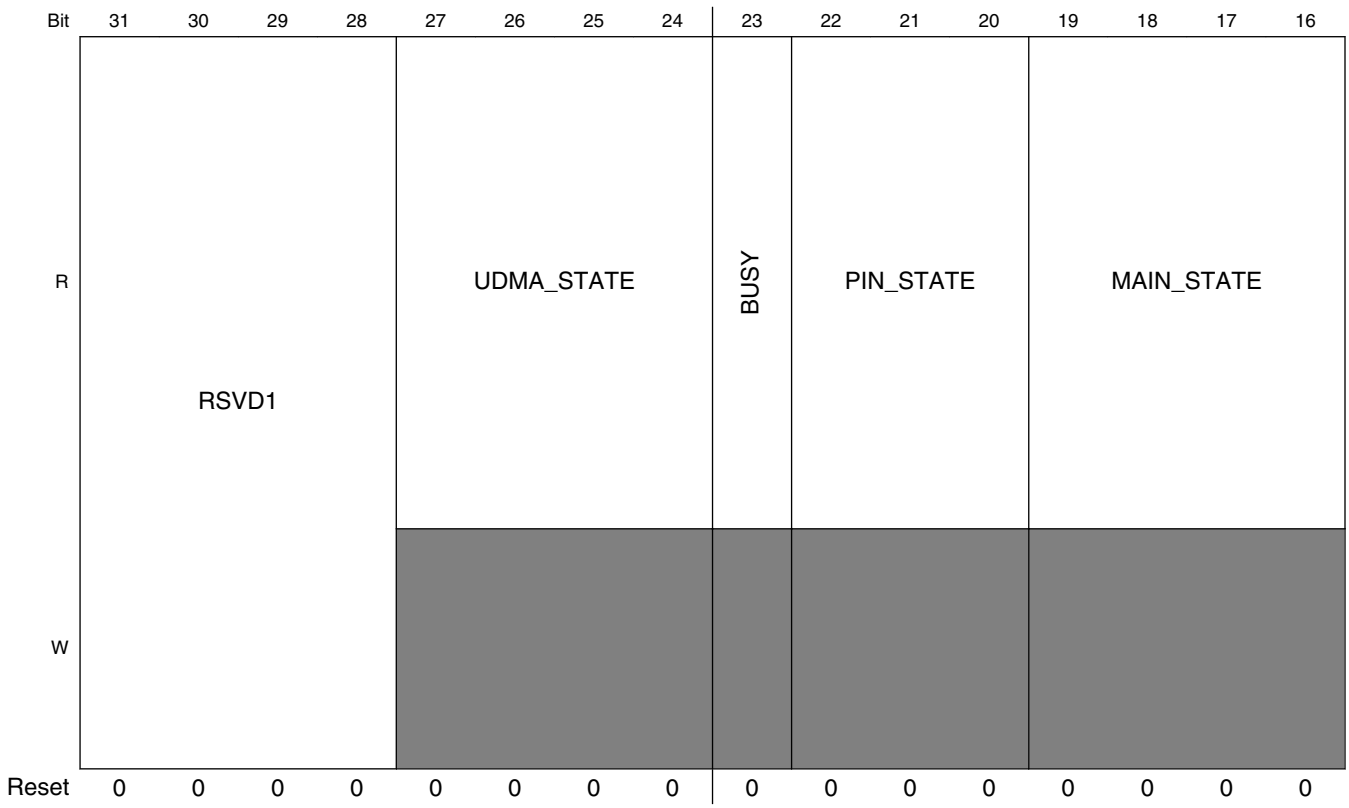
Field	Description
31–24 MAJOR	Fixed read-only value reflecting the MAJOR field of the RTL version.
23–16 MINOR	Fixed read-only value reflecting the MINOR field of the RTL version.
STEP	Fixed read-only value reflecting the stepping of the RTL version.

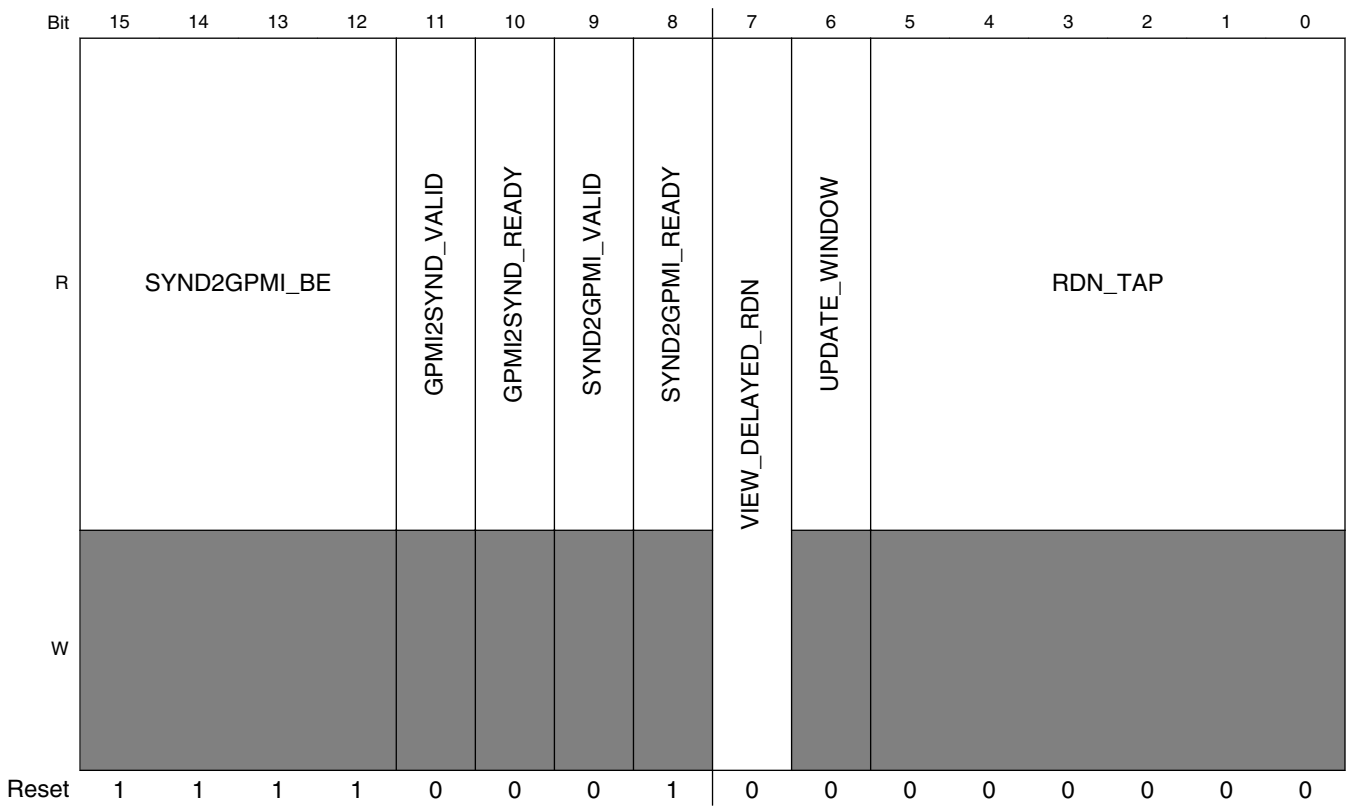
29.6.15 GPMI Debug2 Information Register Description (GPMI_DEBUG2)

The GPMI Debug2 information register provides a read back path for diagnostics to determine the current operating state of the GPMI controller.

GPMI_DEBUG2 0x0E0

Address: 11_2000h base + E0h offset = 11_20E0h





GPMI_DEBUG2 field descriptions

Field	Description
31–28 RSVD1	Always write zeroes to this bit field.
27–24 UDMA_STATE	USM_IDLE = 4'h0, idle USM_DMARQ = 4'h1, DMA req USM_ACK = 4'h2, DMA ACK USM_FIFO_E = 4'h3, Fifo empty USM_WPAUSE = 4'h4, WR DMA Paused by device USM_TSTRB = 4'h5, Toggle HSTROBE USM_CAPTUR = 4'h6, Capture Stage, (data sampled with DSTROBE is valid) USM_DATOUT = 4'h7, Change Burst DATAOUT USM_CRC = 4'h8, Source CRC to Device USM_WAIT_R = 4'h9, Waiting for DDMARDY- USM_END = 4'ha; Negate DMAACK (end of DMA) USM_WAIT_S = 4'hb, Waiting for DSTROBE USM_RPAUSE = 4'hc, Rd DMA Paused by Host USM_RSTOP = 4'hd, Rd DMA Stopped by Host USM_WTERM = 4'he, Wr DMA Termination State USM_RTERM = 4'hf, Rd DMA Termination state
23 BUSY	When asserted the GPMI is busy. Undefined results may occur if any registers are written when BUSY is asserted. DISABLED = 0x0 The GPMI is not busy. ENABLED = 0x1 The GPMI is busy.
22–20 PIN_STATE	parameter PSM_IDLE = 3'h0, PSM_BYTCNT = 3'h1, PSM_ADDR = 3'h2, PSM_STALL = 3'h3, PSM_STROBE = 3'h4, PSM_ATARDY = 3'h5, PSM_DHOLD = 3'h6, PSM_DONE = 3'h7. PSM_IDLE = 0x0 PSM_BYTCNT = 0x1 PSM_ADDR = 0x2 PSM_STALL = 0x3 PSM_STROBE = 0x4 PSM_ATARDY = 0x5 PSM_DHOLD = 0x6 PSM_DONE = 0x7
19–16 MAIN_STATE	parameter MSM_IDLE = 4'h0, MSM_BYTCNT = 4'h1, MSM_WAITFE = 4'h2, MSM_WAITFR = 4'h3, MSM_DMAREQ = 4'h4, MSM_DMAACK = 4'h5, MSM_WAITFF = 4'h6, MSM_LDFIFO = 4'h7, MSM_LDDMAR = 4'h8, MSM_RDCMP = 4'h9, MSM_DONE = 4'hA. MSM_IDLE = 0x0 MSM_BYTCNT = 0x1 MSM_WAITFE = 0x2 MSM_WAITFR = 0x3

Table continues on the next page...

GPMI_DEBUG2 field descriptions (continued)

Field	Description
	MSM_DMAREQ = 0x4 MSM_DMAACK = 0x5 MSM_WAITFF = 0x6 MSM_LDFIFO = 0x7 MSM_LDDMAR = 0x8 MSM_RDCMP = 0x9 MSM_DONE = 0xA
15–12 SYND2GPMI_BE	Data byte enable Input from BCH.
11 GPMI2SYND_ VALID	Data handshake output to BCH.
10 GPMI2SYND_ READY	Data handshake output to BCH.
9 SYND2GPMI_ VALID	Data handshake Input from BCH.
8 SYND2GPMI_ READY	Data handshake Input from BCH.
7 VIEW_ DELAYED_RDN	Set to a 1 to select the delayed feedback RE_B to drive the GPMI_ADDR[0] (Nand CLE) pin. For debug purposes, this will allow you see if DLL is functioning properly.
6 UPDATE_ WINDOW	A 1 indicates that the DLL is busy generating the required delay.
RDN_TAP	This is the DLL tap calculated by the DLL controller. The selects the amount of delay form the DLL chain.

29.6.16 GPMI Debug3 Information Register Description (GPMI_DEBUG3)

The GPMI Debug3 information register provides a read back path for diagnostics to determine the current operating state of the GPMI controller.

GPMI_DEBUG3 0x0F0

Address: 11_2000h base + F0h offset = 11_20F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	APB_WORD_CNTR																DEV_WORD_CNTR															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GPMI_DEBUG3 field descriptions

Field	Description
31–16 APB_WORD_ CNTR	Reflects the number of bytes remains to be transferred on the APB bus.
DEV_WORD_ CNTR	Reflects the number of bytes remains to be transferred on the ATA/Nand bus.

29.6.17 GPMI Double Rate Read DLL Control Register Description (GPMI_READ_DDR_DLL_CTRL)

GPMI DDR Read Delay Loop Lock Control Register. This register provides programmability in DDR mode for data input timing and data formats.

GPMI_READ_DDR_DLL_CTRL 0x100

Address: 11_2000h base + 100h offset = 11_2100h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	REF_UPDATE_INT				SLV_UPDATE_INT								RSVD1		SLV_ OVERRIDE_ VAL	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	SLV_OVERRIDE_VAL						SLV_OVERRIDE	REFCLK_ON	GATE_UPDATE	SLV_DLY_TARGET				SLV_FORCE_UPD	RESET	ENABLE
W																
Reset	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0

GPMI_READ_DDR_DLL_CTRL field descriptions

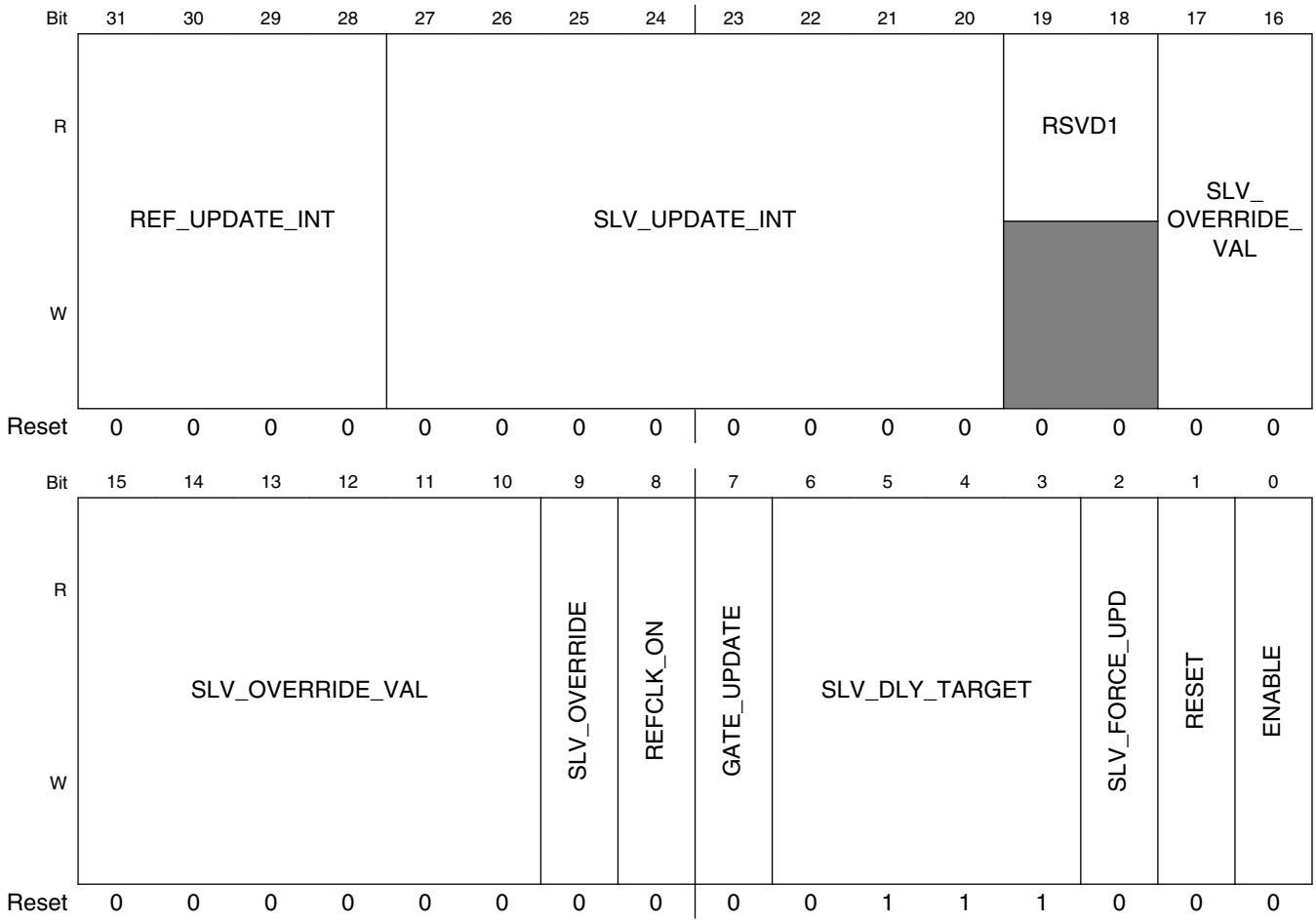
Field	Description
31–28 REF_UPDATE_INT	This field allows the user to add additional delay cycles to the DLL control loop (reference delay line control). By default, the DLL control loop shall update every two GPMICLK cycles. Programming this field results in a DLL control loop update interval of (2 + REF_UPDATE_INT) * GPMICLK. It should be noted that increasing the reference delay-line update interval reduces the ability of the DLL to adjust to fast changes in conditions that may effect the delay (such as voltage and temperature)
27–20 SLV_UPDATE_INT	Setting a value greater than 0 in this field, shall over-ride the default slave delay-line update interval of 256 GPMICLK cycles. A value of 0 results in an update interval of 256 GPMICLK cycles (default setting). A value of 0x0f results in 15 cycles and so on. Note that software can always cause an update of the slave-delay line using the SLV_FORCE_UPDATE register. Note that the slave delay line will also update automatically when the reference DLL transitions to a locked state (from an un-locked state).
19–18 RSVD1	Reserved
17–10 SLV_OVERRIDE_VAL	When SLV_OVERRIDE=1 This field is used to select 1 of 256 physical taps manually. A value of 0 selects tap 1, and a value of 0x7f selects tap 256.
9 SLV_OVERRIDE	Set this bit to 1 to Enable manual override for slave delay chain using SLV_OVERRIDE_VAL; to set 0 to disable manual override. This feature does not require the DLL to tbe enabled using the ENABLE bit. In fact to reduce power, if SLV_OVERRIDE is used, it is recommended to disable the DLL with ENABLE=0
8 REFCLK_ON	set this bit to 1 will turn on the reference clock
7 GATE_UPDATE	Setting this bit to 1, forces the slave delay line not update
6–3 SLV_DLY_TARGET	The delay target for the read clock is can be programmed in 1/16th increments of an GPMICLK half-period. So the input read-clock can be delayed relative input data from (GPMICLK/2)/16 to GPMICLK/2.
2 SLV_FORCE_UPD	Setting this bit to 1, forces the slave delay line to update to the DLL calibrated value immediately. The slave delay line shall update automatically based on the SLV_UPDATE_INT interval or when a DLL lock condition is sensed. Subsequent forcing of the slave-line update can only occur if SLV_FORCE_UP is set back to 0 and then asserted again (edge triggered).
1 RESET	Setting this bit to 1 force a reset on DLL. This will cause the DLL to lose lock and re-calibrate to detect an GPMICLK half period phase shift. This signal is used by the DLL as edge-sensitive, so in order to create a subsequent reset, RESET must be taken low and then asserted again.
0 ENABLE	Set this bit to 1 to enable the DLL and delay chain; otherwise; set to 0 to bypasses DLL. Note that using the slave delay line override feature with SLV_OVERRIDE and SLV_OVERRIDE VAL, the DLL does not need to be enabled.

29.6.18 GPMI Double Rate Write DLL Control Register Description (GPMI_WRITE_DDR_DLL_CTRL)

GPMI DDR Write Delay Loop Lock Control Register. This register provides programmability in DDR mode for data output timing and data formats.

GPMI_WRITE_DDR_DLL_CTRL 0x110

Address: 11_2000h base + 110h offset = 11_2110h



GPMI_WRITE_DDR_DLL_CTRL field descriptions

Field	Description
31–28 REF_UPDATE_ INT	This field allows the user to add additional delay cycles to the DLL control loop (reference delay line control). By default, the DLL control loop shall update every two GPMICLK cycles. Programming this field results in a DLL control loop update interval of (2 + REF_UPDATE_INT) * GPMICLK. It should be noted that increasing the reference delay-line update interval reduces the ability of the DLL to adjust to fast changes in conditions that may effect the delay (such as voltage and temperature)
27–20 SLV_UPDATE_ INT	Setting a value greater than 0 in this field, shall over-ride the default slave delay-line update interval of 256 GPMICLK cycles. A value of 0 results in an update interval of 256 GPMICLK cycles (default setting). A value of 0x0f results in 15 cycles and so on. Note that software can always cause an update of the slave-delay line using the SLV_FORCE_UPDATE register. Note that the slave delay line will also update automatically when the reference DLL transitions to a locked state (from an un-locked state).
19–18 RSVD1	Reserved
17–10 SLV_ OVERRIDE_VAL	When SLV_OVERRIDE=1 This field is used to select 1 of 256 physical taps manually. A value of 0 selects tap 1, and a value of 0x7f selects tap 256.
9 SLV_OVERRIDE	Set this bit to 1 to Enable manual override for slave delay chain using SLV_OVERRIDE_VAL; to set 0 to disable manual override. This feature does not require the DLL to tbe enabled using the ENABLE bit. In fact to reduce power, if SLV_OVERRIDE is used, it is recommended to disable the DLL with ENABLE=0

Table continues on the next page...

GPMI_WRITE_DDR_DLL_CTRL field descriptions (continued)

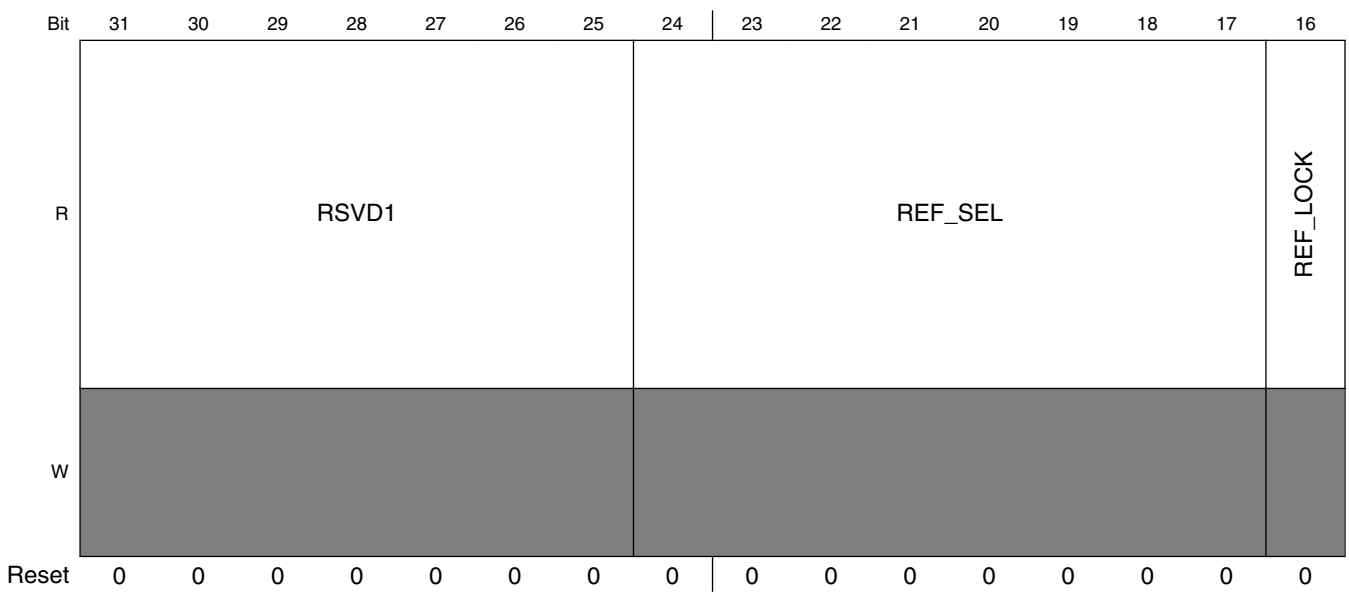
Field	Description
8 REFCLK_ON	set this bit to 1 will turn on the reference clock
7 GATE_UPDATE	Setting this bit to 1, forces the slave delay line not update
6–3 SLV_DLY_ TARGET	The delay target for the read clock can be programmed in 1/16th increments of an GPMICLK half-period. So the input read-clock can be delayed relative input data from (GPMICLK/2)/16 to GPMICLK/2.
2 SLV_FORCE_ UPD	Setting this bit to 1, forces the slave delay line to update to the DLL calibrated value immediately. The slave delay line shall update automatically based on the SLV_UPDATE_INT interval or when a DLL lock condition is sensed. Subsequent forcing of the slave-line update can only occur if SLV_FORCE_UP is set back to 0 and then asserted again (edge triggered).
1 RESET	Setting this bit to 1 force a reset on DLL. This will cause the DLL to lose lock and re-calibrate to detect an GPMICLK half period phase shift. This signal is used by the DLL as edge-sensitive, so in order to create a subsequent reset, RESET must be taken low and then asserted again.
0 ENABLE	Set this bit to 1 to enable the DLL and delay chain; otherwise; set to 0 to bypasses DLL. Note that using the slave delay line override feature with SLV_OVERRIDE and SLV_OVERRIDE VAL, the DLL does not need to be enabled.

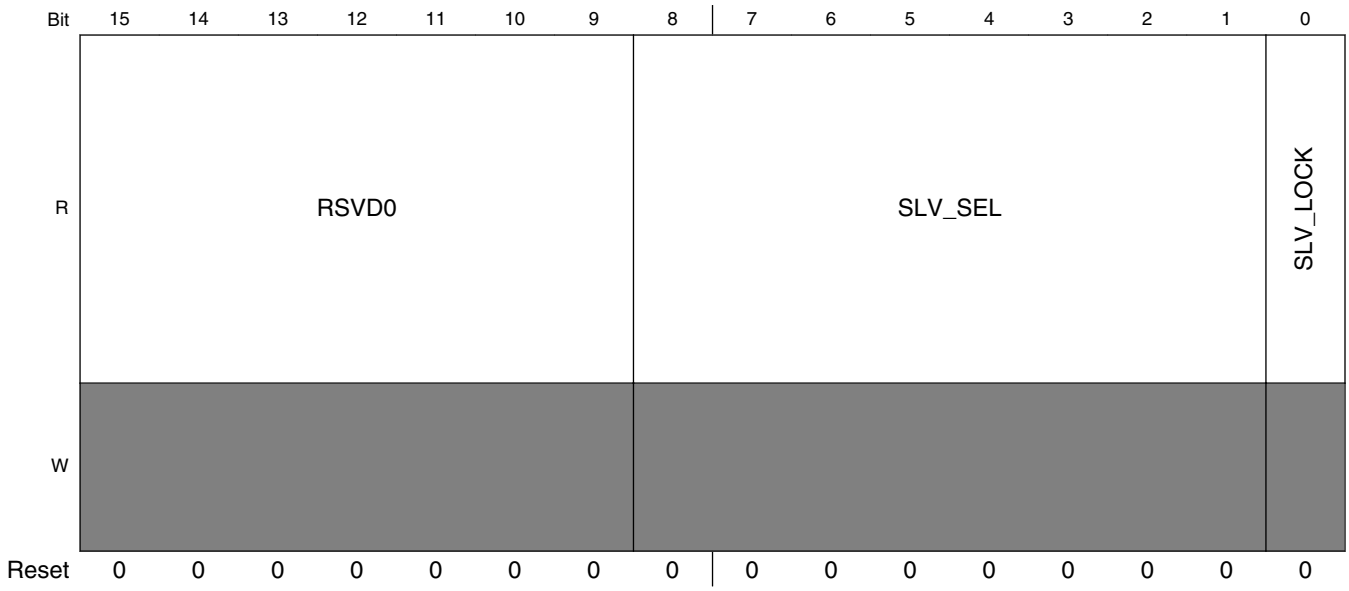
29.6.19 GPMI Double Rate Read DLL Status Register Description (GPMI_READ_DDR_DLL_STS)

GPMI Double Rate Read DLL Status Register, Read Only. GPMI DLL status fields are provided in this register.

GPMI_READ_DDR_DLL_STS 0x120

Address: 11_2000h base + 120h offset = 11_2120h





GPMI_READ_DDR_DLL_STS field descriptions

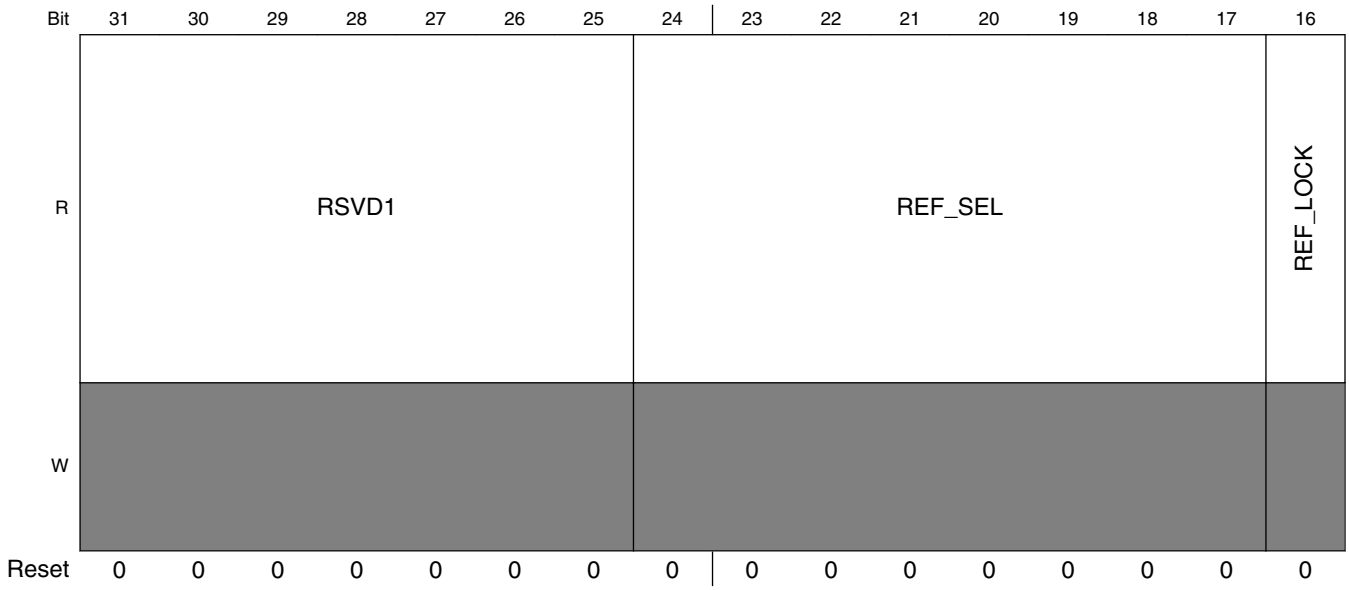
Field	Description
31–25 RSVD1	Reserved
24–17 REF_SEL	Reference delay line select status.
16 REF_LOCK	Reference DLL lock status. This signifies that the DLL has detected and locked to a half-phase GPMICLK shift, allowing the slave delay-line to perform programmed clock delays.
15–9 RSVD0	Reserved
8–1 SLV_SEL	Slave delay line select status
0 SLV_LOCK	Slave delay-line lock status. This signifies that a valid calibration has been set to the slave-delay line and that the slave-delay line is implementing the programmed delay value.

29.6.20 GPMI Double Rate Write DLL Status Register Description (GPMI_WRITE_DDR_DLL_STS)

GPMI Double Rate Write DLL Status Register, Read Only. GPMI DLL status fields are provided in this register.

GPMI_WRITE_DDR_DLL_STS 0x130

Address: 11_2000h base + 130h offset = 11_2130h



GPMI_WRITE_DDR_DLL_STS field descriptions

Field	Description
31–25 RSVD1	Reserved
24–17 REF_SEL	Reference delay line select status.
16 REF_LOCK	Reference DLL lock status. This signifies that the DLL has detected and locked to a half-phase GPMICLK shift, allowing the slave delay-line to perform programmed clock delays.
15–9 RSVD0	Reserved
8–1 SLV_SEL	Slave delay line select status

Table continues on the next page...

GPMI_WRITE_DDR_DLL_STS field descriptions (continued)

Field	Description
0 SLV_LOCK	Slave delay-line lock status. This signifies that a valid calibration has been set to the slave-delay line and that the slave-delay line is implementing the programmed delay value.

Chapter 30
General Purpose Timer (GPT)

30.1 Overview

This chapter describes the General Purpose Timer (GPT) module interface. It is also a reference for software driver programming.

The GPT has a 32-bit up-counter. The timer counter value can be captured in a register using an event on an external pin. The capture trigger can be programmed to be a rising or/and falling edge. The GPT can also generate an event on the DO_CMPOUT*n* pins and an interrupt when the timer reaches a programmed value. The GPT has a 12-bit prescaler, which provides a programmable clock frequency derived from multiple clock sources.

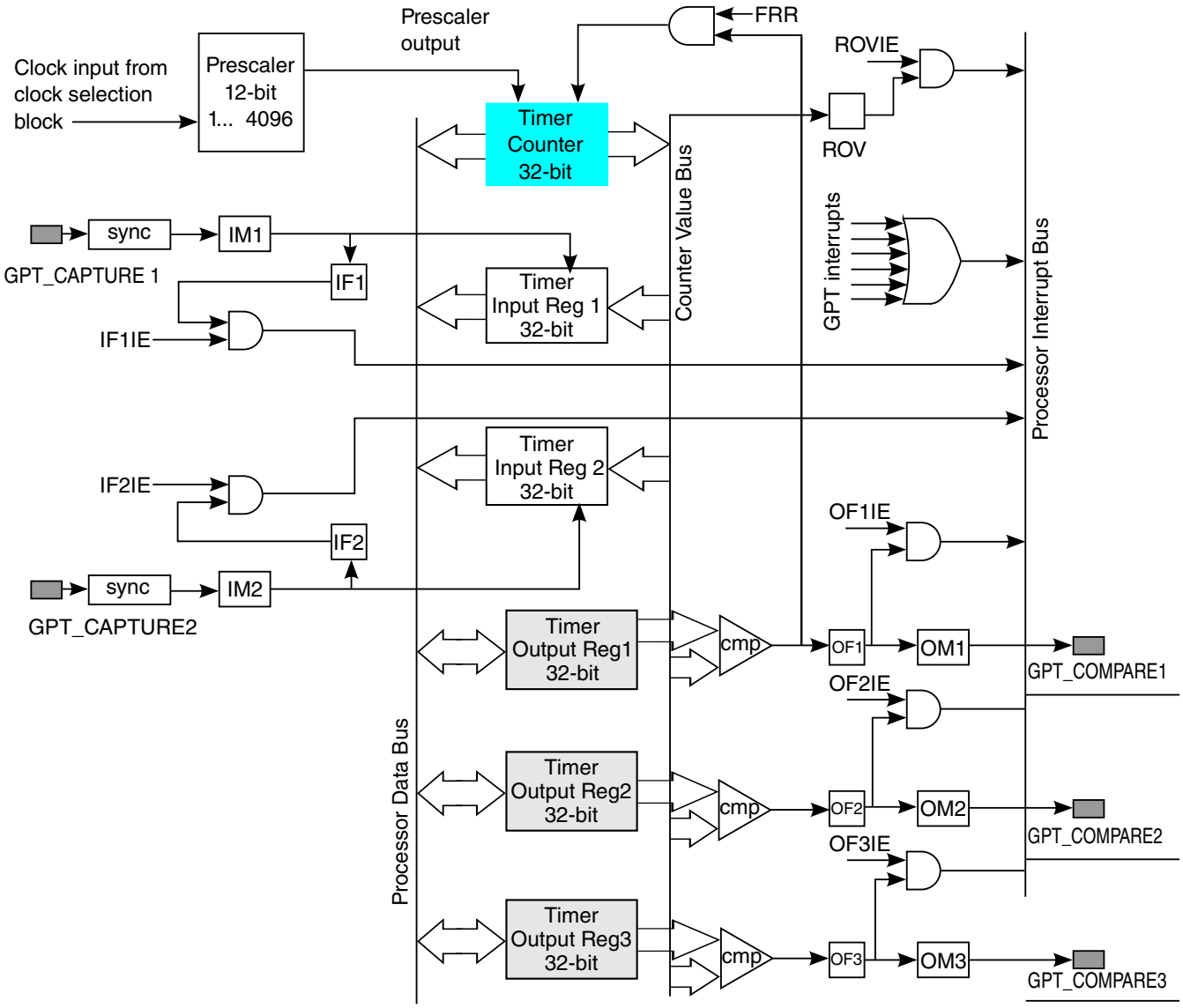


Figure 30-1. GPT Block Diagram

The following figure shows the GPT functional clocking scheme.

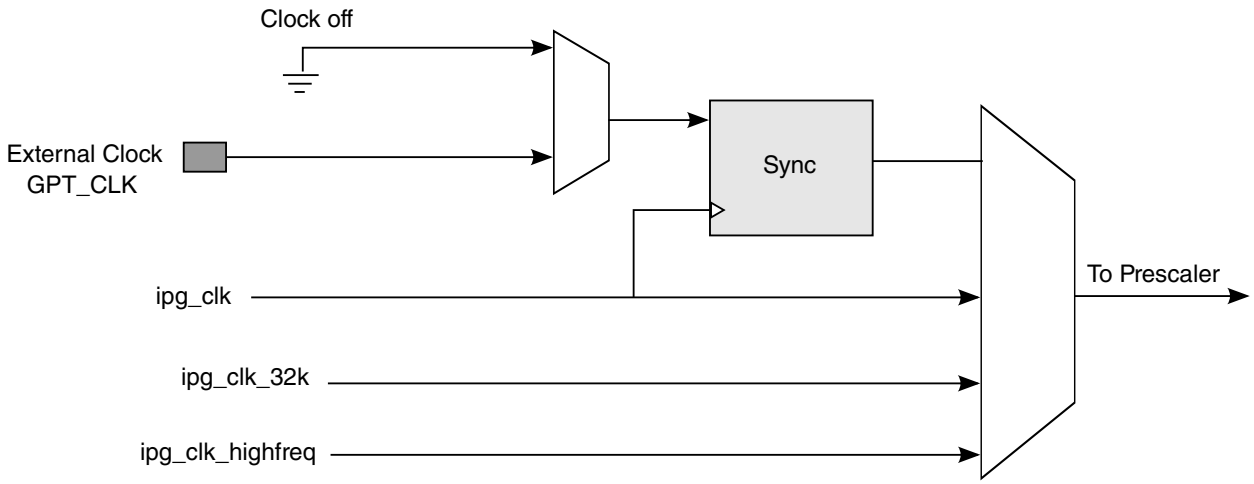


Figure 30-2. GPT Counter Clocks Diagram

30.1.1 Features

- One 32-bit up-counter with clock source selection, including external clock.
- Two input capture channels with a programmable trigger edge.
- Three output compare channels with a programmable output mode. A "forced compare" feature is also available.
- Can be programmed to be *active* in low power and debug modes.
- Interrupt generation at capture, compare, and rollover events.
- Restart or free-run modes for counter operations.

30.1.2 Modes and Operation

The GPT supports the modes described in the indicated sections:

- Operating Modes
  - Restart Mode
  - Free-Run Mode

30.2 External Signals

The GPT follows the IP Bus protocol for interfacing with the processor core. The GPT does not have *any interface signals with any other module inside the chip,* except for the clock and reset inputs (from the clock and reset controller module) and for the interrupt signals *to* the processor interrupt handler. There are functional and clock inputs, and functional output signals going outside the chip boundary.

The following table describes all block signals that connect off-chip.

Table 30-1. GPT External Signals

Signal	Description	Pad	Mode	Direction
GPT_CLK	Input pin for an external clock that the counter can be operated at.	SD1_CLK	ALT3	I
GPT_CAPTURE1	Input pin for a capture event for Input Capture Channel 1.	SD1_DAT0	ALT3	I
GPT_CAPTURE2	Input pin for a capture event for Input Capture Channel 2.	SD1_DAT1	ALT3	I
GPT_COMPARE1	Output pin that indicates a "compare event" occurrence in Output Compare Channel 1.	SD1_CMD	ALT3	O
GPT_COMPARE2	Output pin that indicates a "compare event" occurrence in Output Compare Channel 2.	SD1_DAT2	ALT2	O
GPT_COMPARE3	Output pin that indicates a "compare event" occurrence in Output Compare Channel 3.	SD1_DAT3	ALT2	O

There are six signals (three input, three output) in the GPT module that *can be* connected to the chip pads.

30.2.1 External Clock Input

The GPT counter can be operated using an external clock from outside the device, and this is the input pin used for that purpose.

The external clock input GPT_CLK is treated as asynchronous to the peripheral clock. To ensure proper operations of GPT, the external clock input frequency should be less than 1/4 of frequency of the peripheral clock. Hysteresis characteristics on this pad will be required because this is a clock input.

30.2.2 Input Capture Trigger Signals

The GPT counter value can be stored in a register, triggered by an event from *outside the device*.

A positive or/and negative edge on these signals GPT_CAPTURE1 , GPT_CAPTURE2 can trigger this capture event. These signals are treated as asynchronous to the peripheral clock. Only those transitions which occur *at least a single clock cycle* (the clock selected to run the counter) *after the previous recorded transition* are guaranteed to trigger a capture event.

30.2.3 Output Compare Signals

The output compare signals: GPT_COMPARE1, GPT_COMPARE2, GPT_COMPARE3, indicate that output compare events have gone through a specified transition.

30.3 Clocks

The clock that is input to the prescaler can be selected from 4 clock sources:

The following table describes the clock sources for GPT. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 30-2. GPT Clocks

Clock name	Clock Root	Description
ipg_clk	ipg_clk_root	Peripheral clock
ipg_clk_32k	ckil_sync_clk_root	Low-frequency reference clock (32kHz)
ipg_clk_highfreq	perclk_clk_root	High-frequency reference clock
ipg_clk_s	ipg_clk_root	Peripheral access clock

- High-Frequency Clock (ipg_clk_highfreq)

  Provided by the Clock Controller Module (CCM), the High Frequency Clock is intended to be ON in Normal Power mode when the Peripheral Clock (ipg_clk) is turned OFF, thereby enabling the GPT to be operated using the High Frequency Clock *in Normal Power mode*. The CCM is expected to provide this clock *after* synchronizing it to the System Bus Clock in Normal functional mode; the CCM is also expected to switch to the *unsynchronized* version of the High Frequency Clock in a Low Power mode.

- Low-Reference Clock (ipg_clk_32k)

  This 32 kHz Low Reference Clock (provided by the CCM) is intended to be ON in Low Power mode when the Peripheral Clock (ipg_clk) is turned OFF, thereby enabling the GPT to be operated using the Low Reference Clock in Low Power mode. The CCM is expected to provide the Low Reference Clock *after* synchronizing it to the System Bus Clock in Normal functional mode; the CCM is also expected to switch to the *unsynchronized* version of the Low Reference Clock in a Low Power mode.

- Peripheral Clock (ipg_clk)

  If the Peripheral Clock or the External Clock is selected(CLKSRC=001 or 011) as Clock Source, then the Peripheral Clock will be ON in normal GPT operations. In Low Power modes, if the GPT is programmed to be disabled (STOPEN or WAITEN or DOZEN=0), then the Peripheral Clock can be switched OFF.

- External Clock

  The External Clock comes from *outside the device* and can be selected to run the GPT counter. The External Clock is treated as *asynchronous to the Peripheral Clock,* and is synchronized to the Peripheral Clock, *inside* the module. Therefore, the External Clock frequency is limited to < 1/4 frequency of the Peripheral Clock, for proper GPT operations. Note that in Low Power modes, *if* the Peripheral Clock is not available, then the External Clock *cannot be used* to run the counter.

- Crystal Oscillator Clock

  This 24MHz Crystal Oscillator Clock (provided by the CCM) is intended to be used against frequency change of Peripheral Clock changes to provide a more accurate timer clock for operation system. The CCM is expected to provide the 24MHz Crystal Oscillator Clock *without* synchronizing it to the System Bus Clock in Normal functional mode. Synchronization is done in GPT module. Before synchronization, the 24MHz Crystal Oscillator Clock is divided by a 24MHz clock prescaler, to make sure the clock frequency less than half of System Bus Clock .

The clock input source is configured using the clock source field (CLKSRC, in the GPT_CR control register). The clock input to the prescaler can be disabled by programming the CLKSRC bits (of the GPT_CR control register) to 000. **The CLKSRC field value should be changed** *only after disabling the GPT* (by setting the EN bit in the GPT_CR to 0).

The PRESCALER field selects the divide ratio of the input clock that drives the main counter. The prescaler can divide the input clock by a value (from 1 to 4096) and can be changed *at any time*. A change in the value of the PRESCALER field *immediately affects* the output clock frequency.

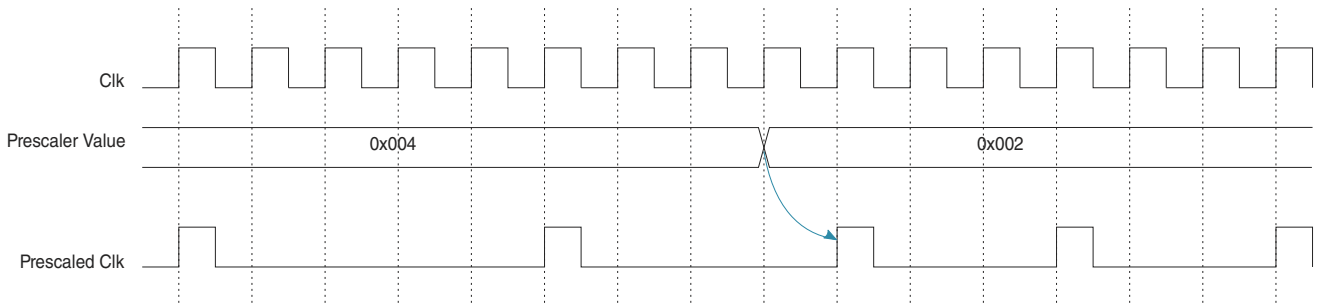


Figure 30-3. Prescaler Value Change Timing Diagram

30.4 Functional Description

This section provides a complete functional description of the GPT.

30.4.1 Operating Modes

The GPT counter can be programmed to work in either of two modes: Restart mode or Free-Run mode.

30.4.1.1 Restart Mode

In Restart mode (selectable through the GPT Control Register GPT_CR), when the counter reaches the compared value, the counter resets and starts again from 0x00000000. The Restart feature is associated only with Compare Channel 1.

Any write access to the Compare register of Channel 1 will reset the GPT counter. This is done to avoid possibly missing a compare event when compare value is changed from a higher value to lower value while counting is proceeding.

For the other two compare channels, when the compare event occurs the counter is *not reset*.

30.4.1.2 Free-Run Mode

In Free-Run mode, when compare events occur for all 3 channels, the counter is *not reset*; instead the counter continues to count until 0xffffffff, and then rolls over (to 0x00000000).

30.4.2 Operation

The General Purpose Timer (GPT) has a single counter (GPT_CNT) that is a 32-bit free-running *up-counter,* which starts counting *after it is enabled by software* (EN=1).

The counter's clock source is the output of the prescaler labelled "Prescaler output" in Figure 30-1.

- If the GPT timer is disabled (EN=0), then the Main Counter *and* Prescaler Counter freeze their current count values. The ENMOD bit determines the value of the GPT counter when the EN bit is set and the Counter is enabled again.
  - If the ENMOD bit is set (=1), then the Main Counter and Prescaler Counter values are reset to 0, when GPT is enabled (EN=1).
  - If ENMOD bit is programmed to 0, then the Main Counter and Prescaler Counter restart counting from their frozen values, when GPT is enabled again (EN=1).
- If GPT is programmed to be disabled in a low power mode (STOP/WAIT), then the Main Counter and Prescaler Counter freeze at their current count values *when* GPT enters low power mode. When GPT exits a low power mode, the Main Counter and Prescaler Counter start counting from their frozen values *regardless* of the ENMOD bit value. Note that the GPT_CNT can be read *at any time* by the processor, and that *both* Input Capture Channels use the *same* counter (GPT_CNT).
- A hardware reset resets all the GPT registers to their respective reset values. All registers except the Output Compare Registers (OCR1, OCR2, OCR3) obtain a value of 0x0. The Compare registers are reset to 0xffffffff.
- The software reset (SWR bit in the GPT_CR control register) resets *all* of the register bits *except* the EN, ENMOD, STOPEN, WAITEN, and DBGEN bits. The state of these bits is not affected by a software reset. Note that a software reset can be given *while the GPT is disabled*.

30.4.2.1 Input Capture

There are two Input Capture Channels, and each Input Capture Channel has a dedicated capture pin, capture register and input edge detection/selection logic. Each input capture function has an associated status flag, and can cause the processor to make an interrupt service request.

When a selected edge transition occurs on an Input Capture pin, the contents of the GPT_CNT is captured on the corresponding capture register and the appropriate interrupt status flag is set. An interrupt request can be generated when the transition is detected *if* its corresponding enable bit is set (in the Interrupt Register). The capture can be programmed to occur on the input pin's rising edge, falling edge, on both rising and falling edges, or the capture can be disabled. The events are synchronized with the clock that was selected to run the counter. Only those transitions that occur at least one clock cycle (clock selected to run the counter) *after* the previous recorded transition will be guaranteed to trigger a capture event. There can be up to one clock cycle of uncertainty in the latching of the input transition. The Input Capture registers can be read *at any time* without affecting their values.

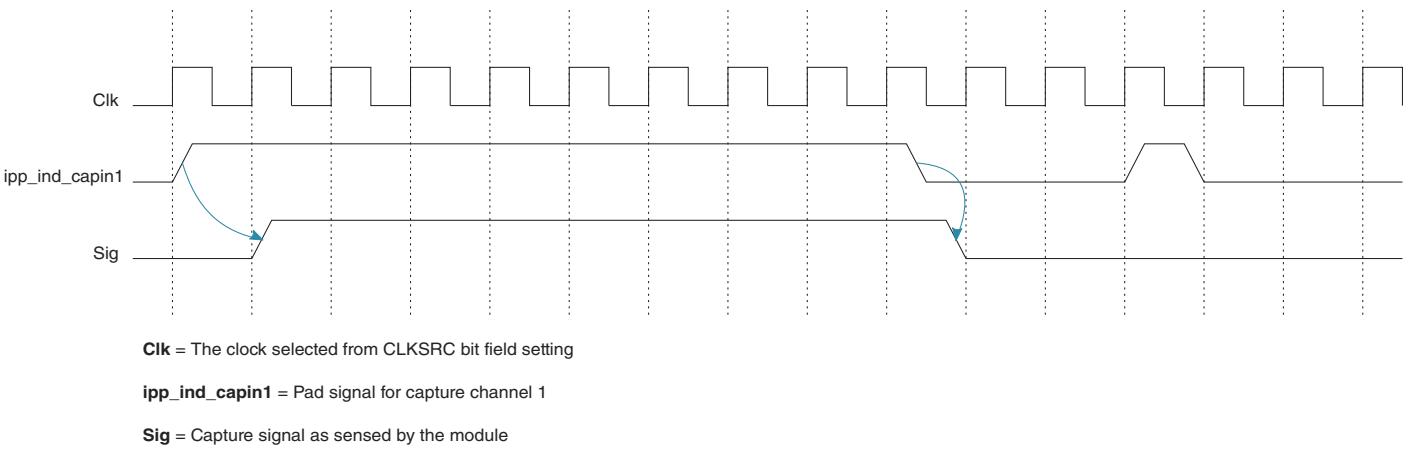


Figure 30-4. Input Capture Event Timing

30.4.2.2 Output Compare

The three Output Compare Channels *use the same counter* (GPT_CNT) as the Input Capture Channels. When the programmed content of an Output Compare register matches the value in GPT_CNT, an output compare status flag is set and an interrupt is generated (if the corresponding bit is set in the interrupt register). Consequently, the Output Compare timer pin will be set, cleared, toggled, not affected at all or provide an active-low pulse for one input clock period (subject to the restriction on the maximum frequency allowed on the pad) according to the mode bits (that were programmed).

There is also a "forced-compare" feature that allows the software to generate a compare event when required, *without the condition of the counter value that is equal to the compare value*. The action taken as a result of a forced compare is the same as when an output compare match occurs, *except that the status flags are not set and no interrupt can be generated*. Forced channels take programmed action immediately after the write to the force-compare bits. These bits are self-negating and always read as zeros.

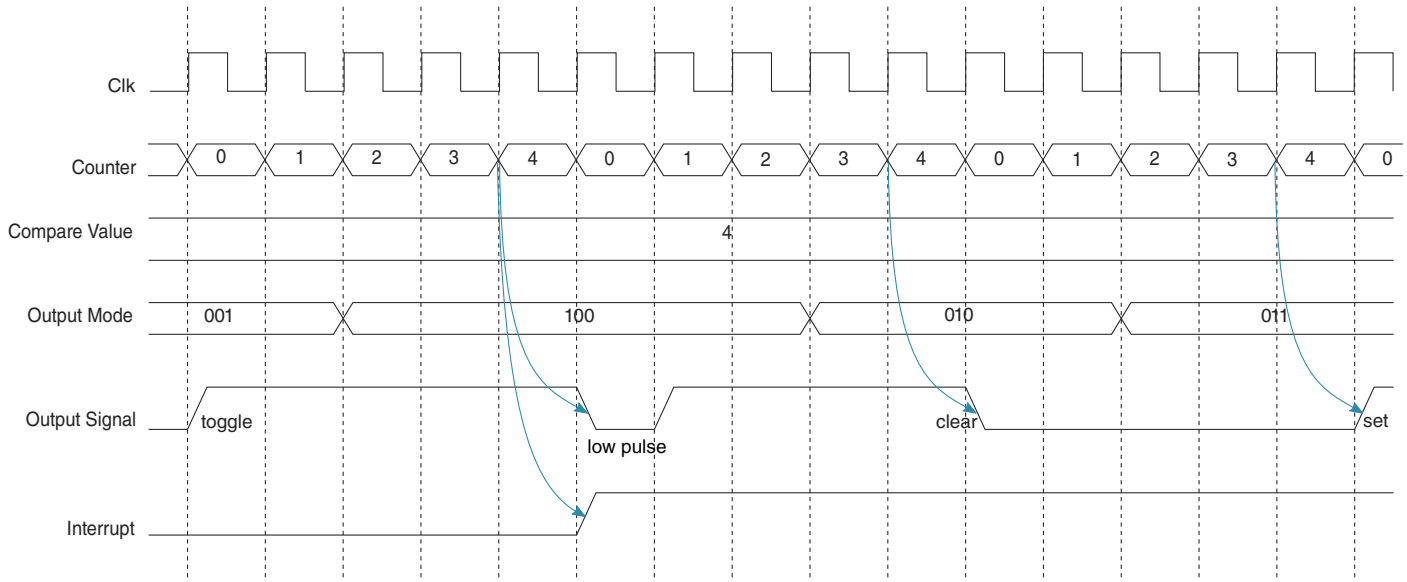


Figure 30-5. Output Compare and Interrupt Timing

30.4.2.3 Interrupts

There are 6 different interrupts that are generated by the GPT. If the selected clock for running the counter is available, then *all interrupts can be generated in Low Power and Debug modes*.

- Rollover Interrupt

  The Rollover Interrupt is generated when the GPT counter reaches 0xffffffff, then resets to 0x00000000 and continues counting. The Rollover Interrupt is enabled by the ROVIE bit in the GPT_IR register; the associated status bit is the ROV bit in the GPT_SR register.

- Input Capture Interrupt 1, 2

  After a capture event occurs, the associated Input Capture Channel generates an interrupt. The "capture event" interrupts are enabled by the IF2IE and IF1IE bits (in the GPT_IR register); the associated status bits are IF2 and IF1 (in the GPT_SR register). The capture of the counter value because of a capture event is *not affected by a pending capture interrupt*. The Capture register is updated with a new counter value when a capture event occurs, regardless of whether that Capture Channels' interrupt has been serviced or not.

- Output Compare Interrupt 1, 2, 3

After a compare event occurs, the associated Output Compare Channel generates an interrupt. The "compare event" interrupts are enabled by the OF3IE, OF2IE, and OF1IE bits (in the GPT_IR register); the associated status bits are OF3, OF2, and OF1 (in the GPT_SR register). A "forced compare" does not generate an interrupt.

A *cumulative* interrupt line is also present, which is asserted whenever any of the above interrupts are posted. The cumulative interrupt line has *no* associated enables or status bits.

30.4.2.4 Low Power Mode Behavior

In Low Power modes, if the clock from the selected clock source is available (except for the External Clock, which can be used *only if* the Peripheral Clock is available), the counter will continue to run depending on whether the control bit for that mode is set. If the clock is not present or if the corresponding low power bit in the GPT_CR control register is 0, the Main Counter and the Prescaler Counter freeze at their current values and resume counting (from their frozen values) when the Low Power mode is exited.

30.4.2.5 Debug Mode Behavior

In Debug mode, the modules in the device have the option of continuing to run or be halted.

- If the DBGEN bit is set, then the GPT timer will continue to run in Debug mode.
- If the DBGEN bit is not set (in the GPT_CR control register), then the GPT timer is halted.

30.5 Initialization/ Application Information

30.5.1 Selecting the Clock Source

The CLKSRC field in the GPT_CR register selects the clock source. The CLKSRC field value should be changed only after disabling the GPT (EN=0).

The software sequence to be followed while changing clock source is:

1. Disable GPT by setting EN=0 in GPT_CR register.
2. Disable GPT interrupt register (GPT_IR).

3. Configure Output Mode to unconnected/ disconnected—Write zeros in OM3, OM2, and OM1 in GPT_CR
4. Disable Input Capture Modes—Write zeros in IM1 and IM2 in GPT_CR
5. Change clock source CLKSRC to the desired value in GPT_CR register.
6. Assert the SWR bit in GPT_CR register.
7. Clear GPT status register (GPT_SR) (i.e., w1c).
8. Set ENMOD=1 in GPT_CR register, to bring GPT counter to 0x00000000.
9. Enable GPT (EN=1) in GPT_CR register.
10. Enable GPT interrupt register (GPT_IR).

30.6 GPT Memory Map/Register Definition

The GPT has 10 user-accessible 32-bit registers, which are used to configure, operate, and monitor the state of the GPT.

An IP bus write access to the GPT Control Register (GPT_CR) and the GPT Output Compare Register1 (GPT_OCR1) results in *one cycle of wait state*, while other valid IP bus accesses incur 0 wait states.

Irrespective of the Response Select signal value, a Write access to the GPT Status Registers (Read-only registers GPT_ICR1, GPT_ICR2, GPT_CNT) will generate a bus exception.

- If the Response Select signal is driven Low, then the Read/Write access to the *unimplemented* address space of GPT (*ips_addr* is greater than or equal to $BASE + $028) will generate a bus exception.
- If the Response Select is driven High, then the Read/Write access to the unimplemented address space of GPT will *not* generate any error response (like a bus exception).

GPT memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_8000	GPT Control Register (GPT_CR)	32	R/W	0000_0000h	30.6.1/1507
209_8004	GPT Prescaler Register (GPT_PR)	32	R/W	0000_0000h	30.6.2/1511
209_8008	GPT Status Register (GPT_SR)	32	R/W	0000_0000h	30.6.3/1512
209_800C	GPT Interrupt Register (GPT_IR)	32	R/W	0000_0000h	30.6.4/1513
209_8010	GPT Output Compare Register 1 (GPT_OCR1)	32	R/W	FFFF_FFFFh	30.6.5/1514
209_8014	GPT Output Compare Register 2 (GPT_OCR2)	32	R/W	FFFF_FFFFh	30.6.6/1515

Table continues on the next page...

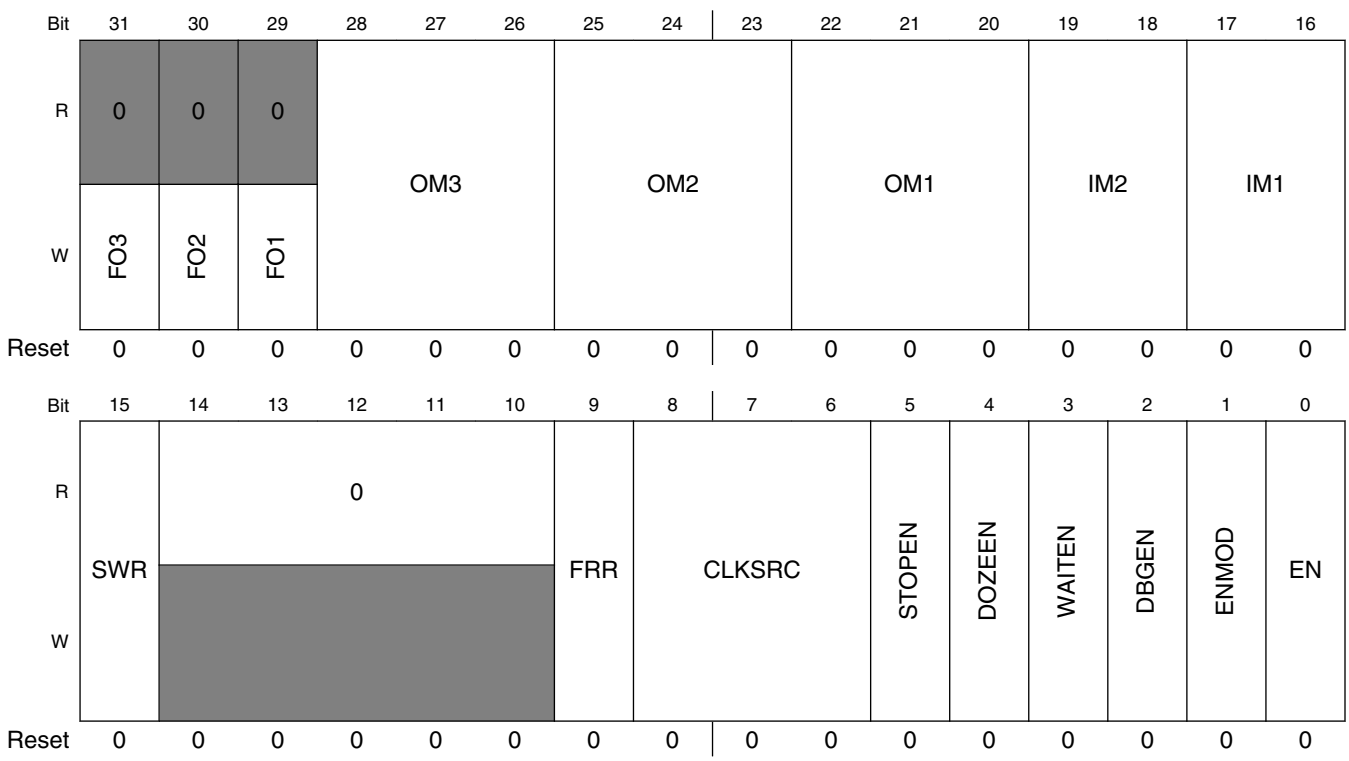
GPT memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
209_8018	GPT Output Compare Register 3 (GPT_OCR3)	32	R/W	FFFF_FFFFh	30.6.7/1515
209_801C	GPT Input Capture Register 1 (GPT_ICR1)	32	R	0000_0000h	30.6.8/1516
209_8020	GPT Input Capture Register 2 (GPT_ICR2)	32	R	0000_0000h	30.6.9/1516
209_8024	GPT Counter Register (GPT_CNT)	32	R	0000_0000h	30.6.10/ 1517

30.6.1 GPT Control Register (GPT_CR)

The GPT Control Register (GPT_CR) is used to program and configure GPT operations. An IP Bus Write to the GPT Control Register occurs after one cycle of wait state, while an IP Bus Read occurs after 0 wait states.

Address: 209_8000h base + 0h offset = 209_8000h



GPT_CR field descriptions

Field	Description
31 FO3	FO3 Force Output Compare Channel 3 FO2 Force Output Compare Channel 2

Table continues on the next page...

GPT_CR field descriptions (continued)

Field	Description
	FO1 Force Output Compare Channel 1 The FO*n* bit causes the pin action *programmed* for the timer Output Compare *n* pin (according to the OM*n* bits in this register). • The OF*n* flag (OF3, OF2, OF1) in the status register is **not affected**. • This bit is self-negating and always read as zero. 0 Writing a 0 has no effect. 1 Causes the programmed pin action on the timer Output Compare *n* pin; the OF*n* flag is not set.
30 FO2	See F03
29 FO1	See F03
28–26 OM3	OM3 (bits 28-26) controls the Output Compare Channel 3 operating mode. OM2 (bits 25-23) controls the Output Compare Channel 2 operating mode. OM1 (bits 22-20) controls the Output Compare Channel 1 operating mode. The OM*n* bits specify the response that a compare event will generate on the output pin of Output Compare Channel *n*. • The toggle, clear, and set options cause a change on the output pin *only* if a compare event occurs. • When OM*n* is programmed as 1xx (active low pulse), the output pin is set to one immediately on the next input clock; a low pulse (that is an input clock in width) occurs when there is a compare event. Note that here, "input clock" refers to the clock selected by the CLKSRC bits of the GPT Control Register. 000 Output disconnected. No response on pin. 001 Toggle output pin 010 Clear output pin 011 Set output pin 1xx Generate an active low pulse (that is one input clock wide) on the output pin.
25–23 OM2	See OM3
22–20 OM1	See OM3
19–18 IM2	IM2 (bits 19-18, Input Capture Channel 2 operating mode) IM1 (bits 17-16, Input Capture Channel 1 operating mode) The IM*n* bit field determines the transition on the input pin (for Input capture channel *n*), which will trigger a capture event. 00 capture disabled 01 capture on rising edge only 10 capture on falling edge only 11 capture on both edges
17–16 IM1	See IM2
15 SWR	Software reset. This is the software reset of the GPT module. It is a self-clearing bit. • The SWR bit is set when the module is in reset state.

Table continues on the next page...

GPT_CR field descriptions (continued)

Field	Description
	• The SWR bit is cleared when the reset procedure finishes. • Setting the SWR bit resets **all of the registers** to their default reset values, except for the CLKSRC, EN, ENMOD, STOPEN, WAITEN, and DBGEN bits in the GPT Control Register (this control register). 0 GPT is not in reset state 1 GPT is in reset state
14–10 Reserved	This read-only field is reserved and always has the value 0.
9 FRR	Free-Run or Restart mode. The FFR bit determines the behavior of the GPT when a compare event in channel 1 occurs. • In Restart mode, after a compare event, the counter resets to 0x00000000 and resumes counting (after the occurrence of a compare event). • In Free-Run mode, after a compare event, the counter continues counting until 0xFFFFFFFF and then rolls over to 0. 0 Restart mode 1 Free-Run mode
8–6 CLKSRC	Clock Source select. The CLKSRC bits select which clock will go to the prescaler (and subsequently be used to run the GPT counter). • The CLKSRC bit field value should only be changed after disabling the GPT by clearing the EN bit in this register (GPT_CR). • A software reset does not affect the CLKSRC bit. 000 No clock 001 Peripheral Clock 010 High Frequency Reference Clock 011 External Clock (CLKIN) 100 Low Frequency Reference Clock 101 Crystal oscillator divided by 8 as Reference Clock 111 Crystal oscillator as Reference Clock others Reserved
5 STOPEN	GPT Stop Mode enable. The STOPEN read/write control bit enables GPT operation *during Stop mode*. • A hardware reset resets the STOPEN bit. • A software reset *does not affect* the STOPEN bit. 0 GPT is disabled in Stop mode. 1 GPT is enabled in Stop mode.
4 DOZEEN	GPT Doze Mode Enable. • A hardware reset resets the DOZEEN bit. • A software reset *does not affect* the DOZEEN bit. 0 GPT is disabled in doze mode. 1 GPT is enabled in doze mode.
3 WAITEN	GPT Wait Mode enable. The WAITEN read/write control bit enables GPT operation *during Wait mode*.

Table continues on the next page...

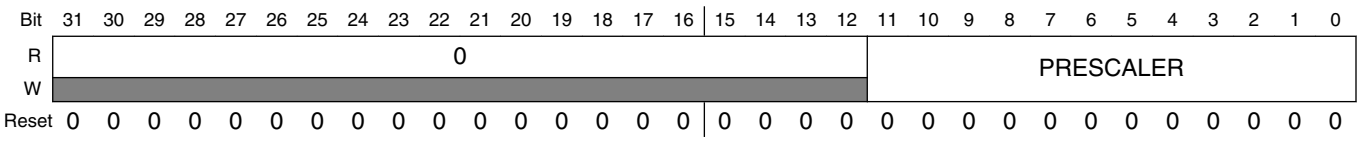
GPT_CR field descriptions (continued)

Field	Description
	• A hardware reset resets the WAITEN bit. • A software reset *does not affect* the WAITEN bit. 0 GPT is disabled in wait mode. 1 GPT is enabled in wait mode.
2 DBGEN	GPT debug mode enable. The DBGEN read/write control bit enables GPT operation *during Debug mode*. • A hardware reset resets the DBGEN bit. • A software reset *does not affect* the DBGEN bit. 0 GPT is disabled in debug mode. 1 GPT is enabled in debug mode.
1 ENMOD	GPT Enable mode. When the GPT is disabled (EN=0), then both the Main Counter and Prescaler Counter *freeze their current count values*. The ENMOD bit determines the value of the GPT counter when Counter is enabled again (if the EN bit is set). • If the ENMOD bit is 1, then the Main Counter and Prescaler Counter values are reset to 0 after GPT is enabled (EN=1). • If the ENMOD bit is 0, then the Main Counter and Prescaler Counter restart counting *from their frozen values* after GPT is enabled (EN=1). • If GPT is programmed to be disabled in a low power mode (STOP/WAIT), then the Main Counter and Prescaler Counter *freeze at their current count values* when the GPT enters low power mode. • When GPT exits low power mode, the Main Counter and Prescaler Counter start counting from their frozen values, regardless of the ENMOD bit value. • Setting the SWR bit will clear the Main Counter and Prescaler Counter values, regardless of the value of EN or ENMOD bits. • A hardware reset resets the ENMOD bit. • A software reset *does not affect* the ENMOD bit. 0 GPT counter will retain its value when it is disabled. 1 GPT counter value is reset to 0 when it is disabled.
0 EN	GPT Enable. The EN bit is the GPT module enable bit. **Before setting the EN bit**, we recommend that *all registers be properly programmed*. • A hardware reset resets the EN bit. • A software reset *does not affect* the EN bit. 0 GPT is disabled. 1 GPT is enabled.

30.6.2 GPT Prescaler Register (GPT_PR)

The GPT Prescaler Register (GPT_PR) contains bits that determine the *divide value* of the clock that runs the counter.

Address: 209_8000h base + 4h offset = 209_8004h



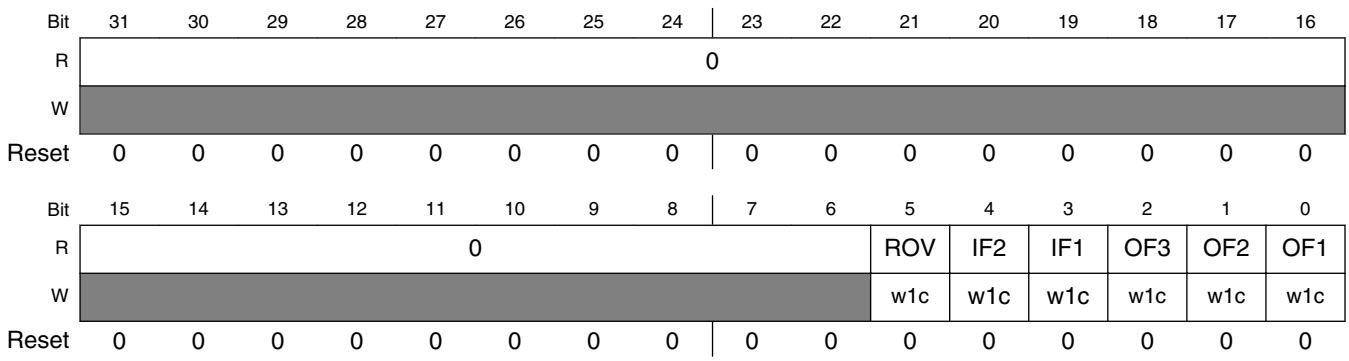
GPT_PR field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
PRESCALER	Prescaler bits. The clock selected by the CLKSRC field is divided by [PRESCALER + 1], and then used to run the counter. • A change in the value of the PRESCALER bits cause the Prescaler counter to reset and a new count period to start immediately. • See Figure 30-3 for the timing diagram. 0x000 Divide by 1 0x001 Divide by 2 ... ... 0xFFF Divide by 4096

30.6.3 GPT Status Register (GPT_SR)

The GPT Status Register (GPT_SR) contains bits that indicate that a counter has rolled over, and if any event has occurred on the Input Capture and Output Compare channels. The bits are cleared by writing a 1 to them.

Address: 209_8000h base + 8h offset = 209_8008h



GPT_SR field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5 ROV	Rollover Flag. The ROV bit indicates that the counter has reached its *maximum possible value* and *rolled over* to 0 (from which the counter continues counting). The ROV bit is only set if the counter has reached 0xFFFFFFFF in both Restart and Free-Run modes. 0 Rollover has not occurred. 1 Rollover has occurred.
4 IF2	IF2 Input capture 2 Flag IF1 Input capture 1 Flag The IF*n* bit indicates that a capture event has occurred on Input Capture channel *n*. 0 Capture event has not occurred. 1 Capture event has occurred.
3 IF1	See IF2
2 OF3	OF3 Output Compare 3 Flag OF2 Output Compare 2 Flag OF1 Output Compare 1 Flag The OF*n* bit indicates that a compare event has occurred on Output Compare channel *n*.

Table continues on the next page...

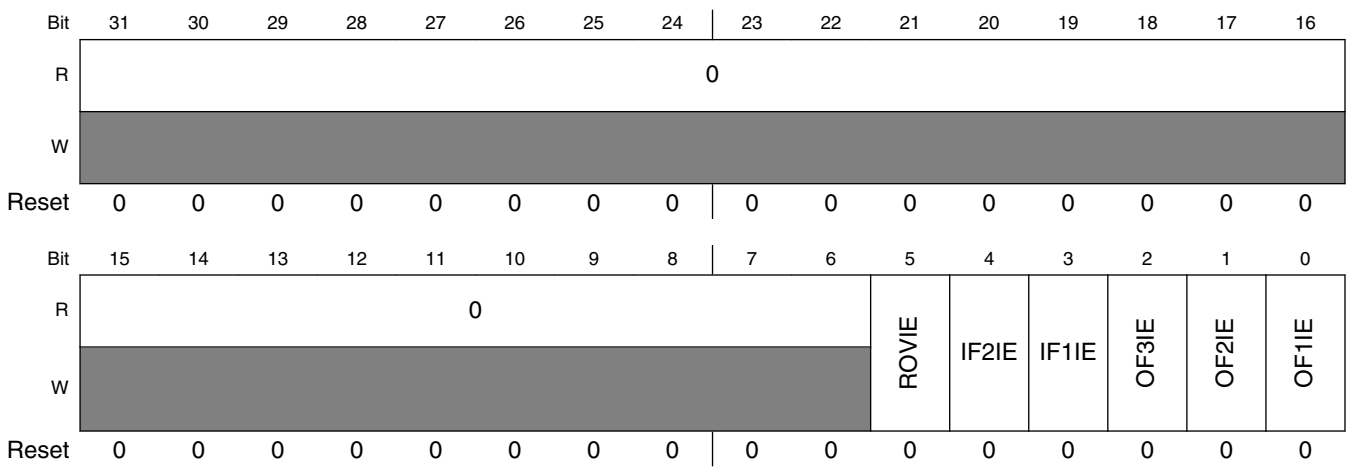
GPT_SR field descriptions (continued)

Field	Description
	0 Compare event has not occurred. 1 Compare event has occurred.
1 OF2	See OF3
0 OF1	See OF3

30.6.4 GPT Interrupt Register (GPT_IR)

The GPT Interrupt Register (GPT_IR) contains bits that control whether interrupts are generated after rollover, input capture and output compare events.

Address: 209_8000h base + Ch offset = 209_800Ch



GPT_IR field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5 ROVIE	Rollover Interrupt Enable. The ROVIE bit controls the Rollover interrupt. 0 Rollover interrupt is disabled. 1 Rollover interrupt enabled.
4 IF2IE	IF2IE Input capture 2 Interrupt Enable IF1IE Input capture 1 Interrupt Enable The IF*n*IE bit controls the IF*n*IE Input Capture *n* Interrupt Enable.

Table continues on the next page...

GPT_IR field descriptions (continued)

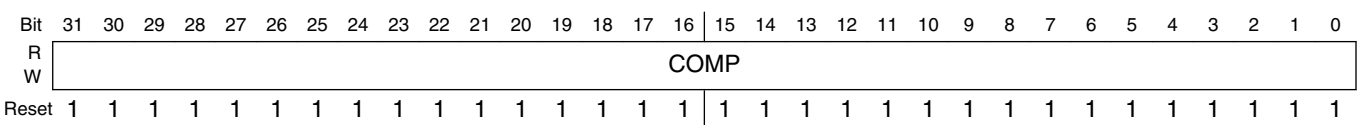
Field	Description
	0 IF2IE Input Capture *n* Interrupt Enable is disabled. 1 IF2IE Input Capture *n* Interrupt Enable is enabled.
3 IF1IE	See IF2IE
2 OF3IE	OF3IE Output Compare 3 Interrupt Enable OF2IE Output Compare 2 Interrupt Enable OF1IE Output Compare 1 Interrupt Enable The OF*n*IE bit controls the Output Compare Channel *n* interrupt. 0 Output Compare Channel *n* interrupt is disabled. 1 Output Compare Channel *n* interrupt is enabled.
1 OF2IE	See OF3IE
0 OF1IE	See OF3IE

30.6.5 GPT Output Compare Register 1 (GPT_OCR1)

The GPT Compare Register 1 (GPT_OCR1) holds the value that determines when a compare event will be generated on Output Compare Channel 1. Any write access to the Compare register of Channel 1 while in Restart mode (FRR=0) will reset the GPT counter.

An IP Bus Write access to the GPT Output Compare Register1 (GPT_OCR1) occurs *after* one cycle of wait state; an IP Bus Read access occurs *immediately* (0 wait states).

Address: 209_8000h base + 10h offset = 209_8010h



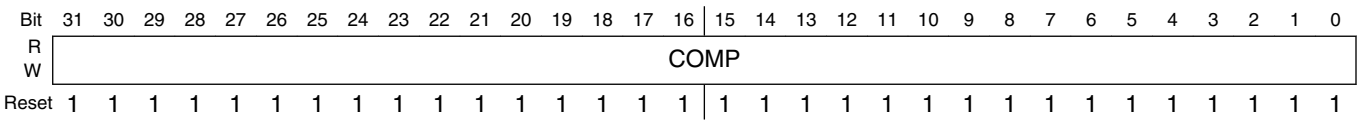
GPT_OCR1 field descriptions

Field	Description
COMP	Compare Value. When the counter value equals the COMP bit field value, a compare event is generated on Output Compare Channel 1.

30.6.6 GPT Output Compare Register 2 (GPT_OCR2)

The GPT Compare Register 2 (GPT_OCR2) holds the value that determines when a compare event will be generated on Output Compare Channel 2.

Address: 209_8000h base + 14h offset = 209_8014h



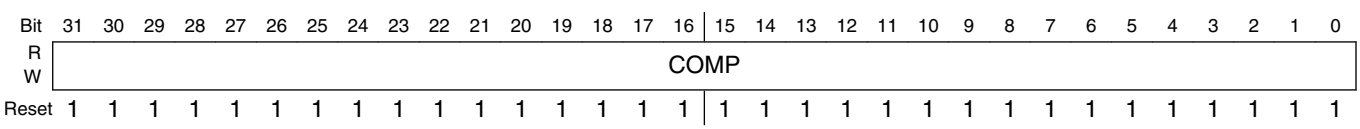
GPT_OCR2 field descriptions

Field	Description
COMP	Compare Value. When the counter value equals the COMP bit field value, a compare event is generated on Output Compare Channel 2.

30.6.7 GPT Output Compare Register 3 (GPT_OCR3)

The GPT Compare Register 3 (GPT_OCR3) holds the value that determines when a compare event will be generated on Output Compare Channel 3.

Address: 209_8000h base + 18h offset = 209_8018h



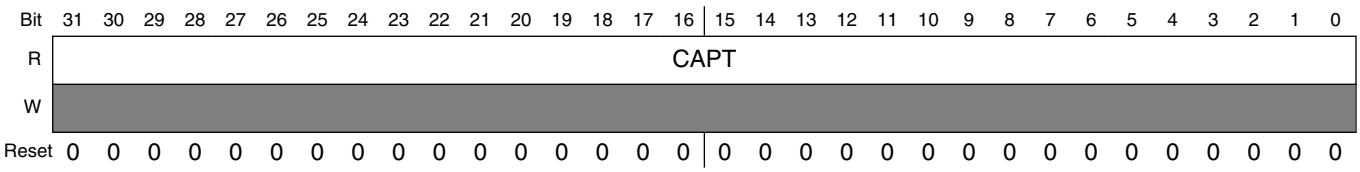
GPT_OCR3 field descriptions

Field	Description
COMP	Compare Value. When the counter value equals the COMP bit field value, a compare event is generated on Output Compare Channel 3.

30.6.8 GPT Input Capture Register 1 (GPT_ICR1)

The GPT Input Capture Register 1 (GPT_ICR1) is a read-only register that holds the value *that was in the counter during the last capture event* on Input Capture Channel 1.

Address: 209_8000h base + 1Ch offset = 209_801Ch



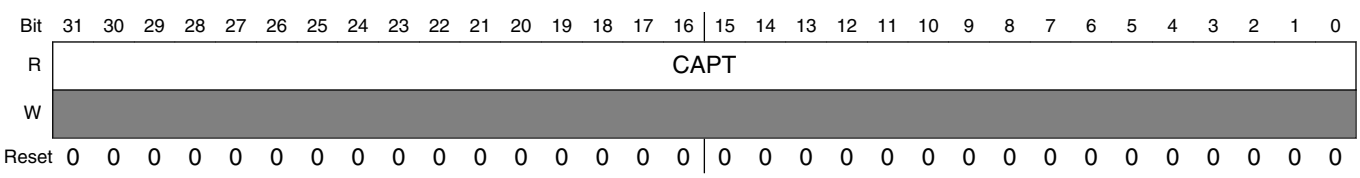
GPT_ICR1 field descriptions

Field	Description
CAPT	Capture Value. After a capture event on Input Capture Channel 1 occurs, the current value of the counter is loaded into GPT Input Capture Register 1.

30.6.9 GPT Input Capture Register 2 (GPT_ICR2)

The GPT Input capture Register 2 (GPT_ICR2) is a read-only register which holds the value that was in the counter during the last capture event on input capture channel 2.

Address: 209_8000h base + 20h offset = 209_8020h



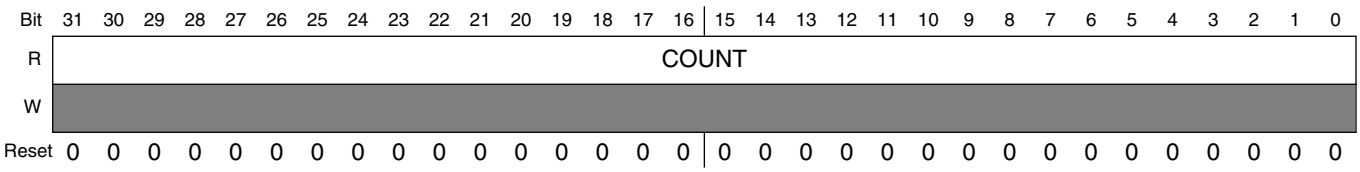
GPT_ICR2 field descriptions

Field	Description
CAPT	Capture Value. After a capture event on Input Capture Channel 2 occurs, the current value of the counter is loaded into GPT Input Capture Register 2.

30.6.10 GPT Counter Register (GPT_CNT)

The GPT Counter Register (GPT_CNT) is the main counter's register. GPT_CNT is a read-only register and can be read *without affecting the counting process* of the GPT.

Address: 209_8000h base + 24h offset = 209_8024h



GPT_CNT field descriptions

Field	Description
COUNT	Counter Value. The COUNT bits show the current count value of the GPT counter.

Chapter 31
2D Graphics Processing Unit (GPU2D)

31.1 Overview

The R2D GPU2D module is designed to display on a variety of consumer devices. Addressable screen sizes range from small displays featured on cell phones to large 1080p high definition displays.

The V2D GPU2D module is designed to display on a variety of consumer devices. Addressable screen sizes range from small displays featured on cell phones to large 1080p high definition displays.

The GPU2D cores provide powerful graphics at low power consumption, utilizing the smallest silicon footprints. Dynamic power consumption is minimized by extensive use of localized clock gating.

R2D GPU Hardware acceleration is brought to numerous 2D including graphical user interfaces (GUI), menu displays, flash animation, and gaming. The GPU R2D block diagrams is presented in the Figure 31-1.

V2D GPU Hardware acceleration is brought to numerous VG applications including graphical user interfaces (GUI), menu displays, flash animation, and gaming. The GPU V2D block diagrams is presented in the Figure 31-2.

31.2 GPU2D Block Diagram

31.2.1 R2D GPU

The R2D graphics processing unit (GPU) defines a high-performance 2D raster graphics core that accelerates the 2D graphics display.

R2D GPU supports acceleration of the following graphics APIs:

- DirectFB (on Linux)
- GDI/DirectDraw (on Windows CE)

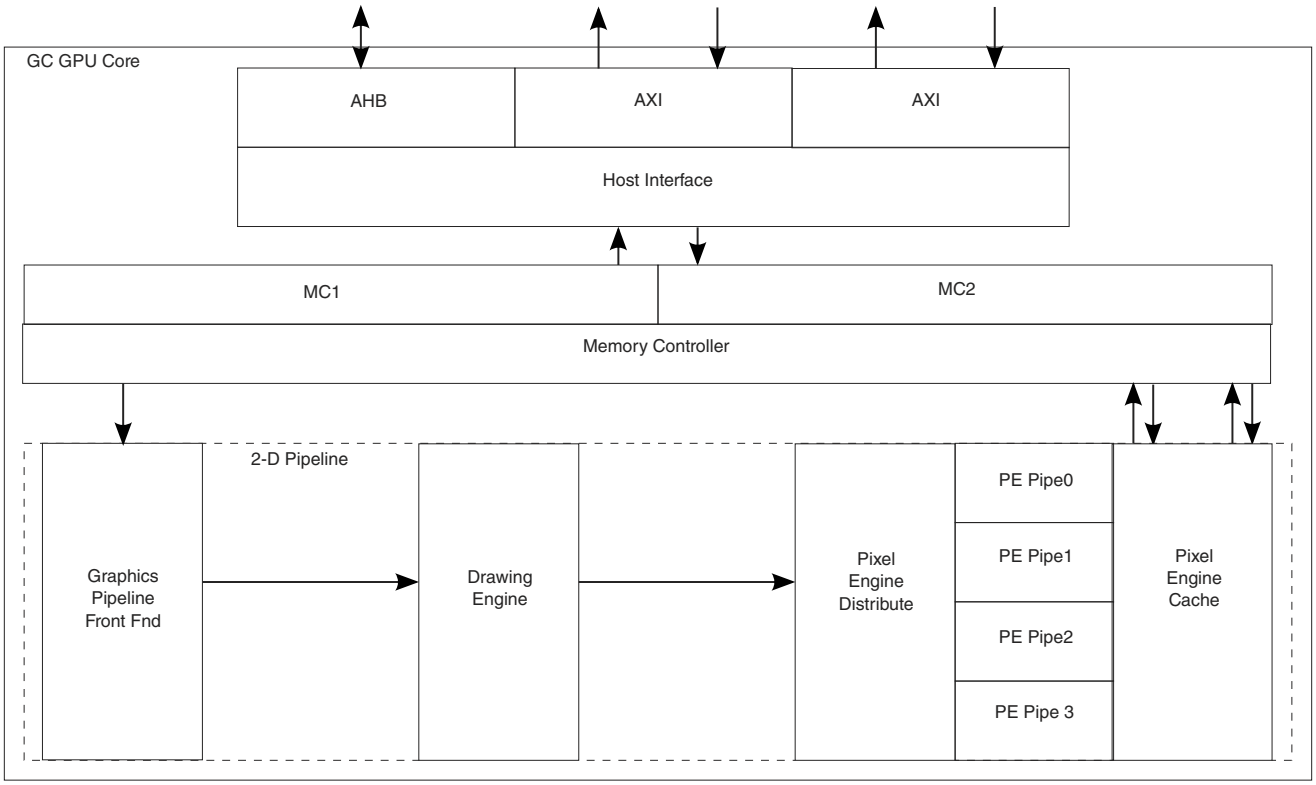


Figure 31-1. R2D GPU Block Diagram

31.2.2 V2D GPU

V2D GPU defines a high-performance graphics core designed for hardware acceleration of OpenVG vector graphics display. V2D GPU is designed for easy integration onto the SoC.

V2D GPU supports the following graphics APIs:

- OpenVG 1.1

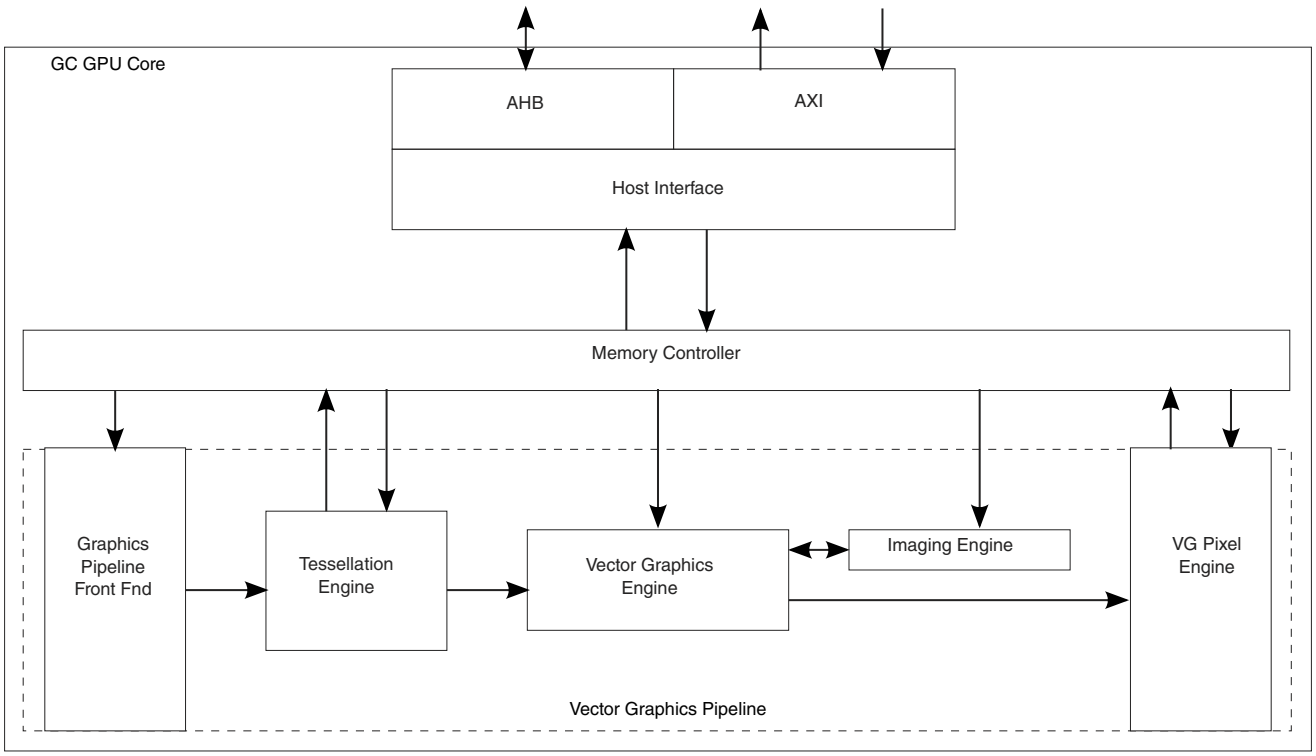


Figure 31-2. V2D GPU Block Diagram

31.3 GPU2D Features

The following sections describe the functional features of the R2D and V2D GPU.

31.3.1 Full Featured R2D GPU Pipeline

- Bit BLT
- Stretch BLT
- Rectangle fill and clear
- Line drawing
- Filter BLT
- Mono expansion for text rendering
- ROP2, ROP3, and ROP4
- Alpha blending, including Java 2 Porter-Duff compositing blending rules
- 32K x 32K coordinate system
- 90 / 180 / 270 degree rotation
- Transparency by monochrome mask, chroma key, or pattern mask

31.3.2 Full Featured V2D GPU Pipeline

- Coordinate Systems and Transformations (Image drawing uses a 3x3 perspective transformation matrix)
- Viewport Clipping, Scissoring and Alpha Masking
- Paths
- Images
- Image Filters
- Paint(gradient and pattern)
- Blending
- Higher-level Geometric Primitives
- Image Warping

31.4 GPU2D OPERATIONS

31.4.1 R2D GPU Operations

Information detailing the R2D GPU operations can be found here.

31.4.1.1 Line

The LINE operation draws a line. Coordinates for two points are given: start point and end point. The end point is not drawn.

Lines are rendered using the Bresenham algorithm. The Bresenham algorithm has the advantage of using integer arithmetic and has no accumulation of rounding errors. In the case of line, only ROP2 and ROP4 are supported. It operates on pattern and destination. The pattern should have a transparency mask in order to use ROP4.

Clipping is supported for lines on a per pixel basis.

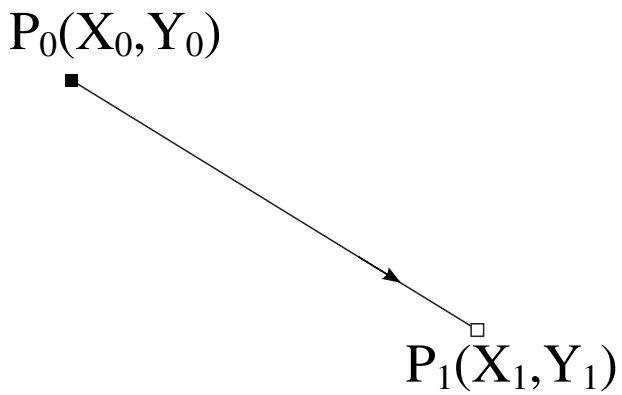


Figure 31-3. Line

31.4.1.2 Rectangle Fill and Clear

Rectangle fill creates a rectangle area with a given color. Essentially rectangle fill is a pattern fill, where an 8x8 pattern is initialized with the specified color. It supports ROP2 and ROP4 with the pattern and destination as its inputs. If ROP4 is used, the pattern should have a transparency mask.

Clear is similar to rectangle fill except that it does not does not use a pattern. A 32-bit clear value with 4-bit byte mask is used to fill the entire rectangle area.

Both rectangle fill and clear support clipping, which is performed on a per primitive basis.

31.4.1.3 BitBLT

Bit blit transfers data from one area of a memory (source) to another area of the memory (destination).

The source and destination can be from the same or different memory locations. Both source and destination must be described by a rectangular area. The source and destination rectangles can be the same size (most bit blits are of this nature) or they can be different sizes, in which case the operation becomes a stretch or shrink blit.

Bit blit supports ROP2, ROP3, and ROP4 which includes source, destination and pattern, and an optional transparency color.

Clipping can be performed on a primitive basis.

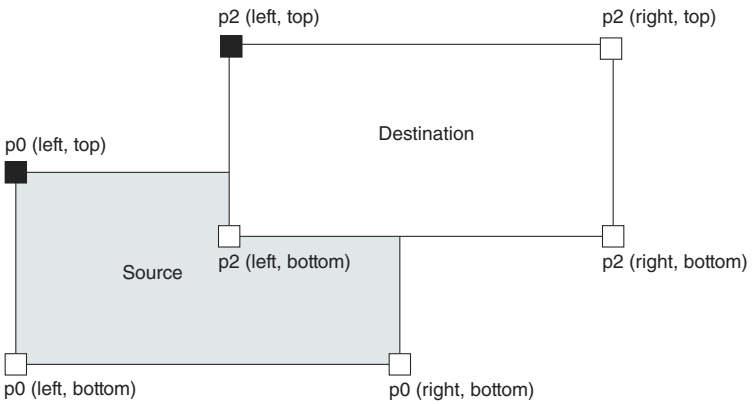


Figure 31-4. BitBLT

The BIT BLT primitive supports the following 10 source and 8 destination image formats:

Table 31-1. Bit BLT Formats

Formats	Source Image	Destination Image
A1R5G5B5	Yes	Yes
A4R4G4B4	Yes	Yes
X1R5G5B5	Yes	Yes
X4R4G4B4	Yes	Yes
R5G6B5	Yes	Yes
A8R8G8B8	Yes	Yes
X8R8G8B8	Yes	Yes
A8	Yes	Yes
1-bit monochrome	Yes	No
8-bit color index	Yes	No

31.4.1.4 Stretch BLT

The STRETCH BLT primitive performs a BitBlt operation with stretch or shrink. The modified Bresenham algorithm is used to generate corresponding coordinates for fast stretching.

The stretch factor is specified in a 15.0 fixed-point format. Stretch blit is not allowed to overlap therefore no part of source and destination can share any piece of memory. Non-stretch blits can overlap. For stretch blit, clipping is performed on a per pixel basis.

31.4.1.5 Monochrome Expansion and Mask BLT

Monochrome expansion and mask blit are different operations, although both use the bit stream from command buffer and both can be the source for ROP4 source selection. This means that each output pixel can be a combination of source, pattern, monochrome mask (for masked blits) and destination.

31.4.1.5.1 Monochrome expansion

For monochrome expansion, the bit from the stream is used to switch on/off a solid color that is defined in a register. This mechanism enables the use of just one bit per pixel to represent colors. In effect, the MONO EXPANSION primitive increases color representation from one bit per pixel to multiple bits per pixel. A typical application for mono color is font drawing.

Monochrome expansion does not support overlapping of the source and destination. It is the responsibility of the driver to make sure that the command will never be executed with overlapping source and destination.

31.4.1.5.2 Mask BLT

For Mask BLT, the bit from the stream is used to toggle on/off a color in the source frame buffer. Mask BLT takes its color source from memory and its monochrome mask from the command stream. Clipping is supported and is performed on a per pixel basis.

31.4.1.6 Filter BLT

Filter blit performs high quality scaling, up or down, using an FIR re-sampling filter with up to 9 taps. Sub-pixel coordinates (locations between the pixel grids) are generated by the drawing engine. The filter block in the drawing engine uses the sub-pixel information to select the appropriate filter kernel. R2D GPU processes 1 pixel every cycle when performing filter blit.

A stretch- or shrink-factor of 15.16 fixed-point format is supported. To generate a single destination pixel requires 9 source pixels. An image is scaled in two passes, one for X-dimension (HOR_FILTER_BLT) and the other for Y-dimension (VER_FILTER_BLT). Software sets up the filter kernel/coefficient table and the kernel size, as well as a temporary buffer for storing intermediate results. After the first pass is completed, intermediate results are sent back to memory, and then the second pass starts to scale the first-pass image. Because of this two-step procedure, the throughput of FILTER BLT is lower than that of STRETCH BLT. Also the Filter Kernel Table may need to be reloaded, and some cycles are consumed in calculating the stepping parameters.

When the stretch or shrink factor is 1, the filterBlit works as a bitBlit copy. It can be used as format converter in that case, for instance, YUV to RGB converter. To use as a format converter, only one pass (HOR_FILTER_BLT or VER_FILTER_BLT) is needed. To optimize the memory bandwidth, when using filterBlit to do YUV to RGB filtering, the temporary target buffer format can be specified as YUY2 to process Y-dimension filtering (VER_FILTER_BLT). This is to avoid converting YUV to A8R8G8B8 in the 1st vertical pass to reduce the memory bandwidth and increase the pixel processing rate. This is the only special case that GPU may use YUY2 as target format.

The FILTER BLT primitive supports the following 13 source and 7 destination image formats:

Filter BLT Formats

Formats		Source Image	Destination Image
A1R5G5B5		Yes	Yes
A4R4G4B4		Yes	Yes
A8R8G8B8		Yes	Yes
R5G6B5		Yes	Yes
X1R5G5B5		Yes	Yes
X4R4G4B4		Yes	Yes
X8R8G8B8		Yes	Yes
YUV	NV12 (4:2:0, 2 planes)	Yes	No
	NV16 (4:2:2, 2 planes)	Yes	No
	UYVY (4:2:2, interleave)	Yes	No
	YUY2 (4:2:2, interleave)	Yes	No
	YV12 (4:2:0, 3 planes)	Yes	No
	8-bit color index	Yes	No

31.4.1.7 R2D Performance of different operations

Table 31-2. Performance of different operations without rotation

Primitive	Peak Performance	Source/destination overlap	Clipping
Line	1 pixel / cycle	N/A	Done on pixel basis
Rectangle	2.4 pixel / cycle	N/A	Done on primitive basis
Clear	2.4 pixel / cycle	N/A	Done on primitive basis
Blit	2.4 pixel / cycle	Overlap is allowed	Done on primitive basis
Stretch blit	1 pixel / cycle	No overlap allowed	Done on pixel basis
Monochrome expansion	1.6 pixel / cycle	No overlap allowed	Done on pixel basis
Filter blit	1 pixel / cycle	N/A	N/A

Table 31-3. Performance of different operations with rotation

Primitive	Performance	Source/destination overlap	Clipping
Line	1 pixel / cycle	N/A	Done on pixel basis
Rectangle	2 pixel / cycle	N/A	Done on primitive basis
Clear	2 pixel / cycle	N/A	Done on primitive basis
Blit	2 pixel / cycle	Overlap is allowed	Done on primitive basis
Stretch blit	1 pixel / cycle	No overlap allowed	Done on pixel basis
Monochrome expansion	1 pixel / cycle	No overlap allowed	Done on pixel basis
Filter blit	1 pixel / cycle	N/A	N/A

31.4.1.8 Rotation

90° / 180° / 270° / X-Flip / Y-Flip / Mirror rotation is supported for all primitives.

31.4.1.9 Transparency Mode

For monochrome expansion:

- Opaque
- Conditional transparency. Transparent if the current pixel matches the specified value.

For blits:

- Opaque
- Masked transparency. Transparent if the mask for the current pixel or pattern is zero.
- Source Conditional transparency. Transparent if the source pixel is within the specified value range.
- Destination Conditional transparency. Transparent if the destination pixel is not within the specified value range.

31.4.1.10 Clipping

One clipping rectangle is supported for all bitBlit primitives.

31.4.1.11 R2D GPU Data Formats

The graphics engine supports 14 source data formats. In addition to these 14 source formats, for RGB source formats, GPU also supports their swizzle formats (ARGB, RGBA, ABGR, BGRA) for RGB formats. For YUV formats, GPU supports their U/V swap formats.

- A1R5G5B5
- A4R4G4B4
- A8R8G8B8
- R5G6B5
- X1R5G5B5
- X4R4G4B4
- X8R8G8B8
- A8
- NV12
- NV16
- UYVY (4:2:2)
- YUY2 (4:2:2)
- YV12 (4:2:0)
- 8-bit color index

There are 8 destination data formats supported by the graphics engine. In addition to these destination RGB formats, their swizzle formats (ARGB, RGBA, ABGR, BGRA) are also supported.

- A1R5G5B5
- A4R4G4B4
- A8R8G8B8
- R5G6B5
- X1R5G5B5
- X4R4G4B4
- X8R8G8B8
- A8

31.4.1.12 ARGB Data Conversion of R2D GPU

The pixels read from source or destination will be expanded into A8R8G8B8 format internally to maintain lossless pixel operations. The resulting pixels will be converted into the destination format.

31.4.1.13 YUV to RGB Conversion of R2D GPU

YUV data can be converted into 8-bit per component RGB format at the output of the cache only. Once converted, there is no way back to YUV format. GPU supports BT.601 and BT.709 YUV to RGB color conversion standards.

In BT.601, the YUV to RGB conversion is done using the following approximation:

16 <= Y <= 235

16 <= U <= 240

16 <= V <= 240

A = Y - 16

B = U - 128

C = V - 128

R = clip((298*A + 410*C + 128) >> 8)

G = clip((298*A - 101*B - 209*C + 128) >> 8)

B = clip((298*A + 519*B + 128) >> 8)

The Y, U and V components are clamped prior to the conversion.

Y is clamped between 16 and 235, inclusively.

U and V are clamped between 16 and 240, inclusively.

In BT.709, the R, G, B equations are slightly changed to

R = clip((298*A + 461*C + 128) >> 8)

G = clip((298*A - 55*B - 137*C + 128) >> 8)

B = clip((298*A + 543*B + 128) >> 8)

31.4.1.14 Color Index Input Conversion Support of R2D GPU

Color index is supported for source data only. A look-up table with 256 entries is provided for indexing the data. The table is fully programmable. The conversion is done when pixels are read out of the cache.

31.4.1.15 Source/Destination Pre-multiply and De-Multiply Support

GPU supports source pre-multiply source alpha or global alpha, or source global color for global colorizing. On destination, the GPU supports destination pre-multiply destination alpha, destination de-multiply alpha.

31.4.1.16 Alpha Blending

The GPU supports alpha blending together with ROP. The alpha blending function is performed on ROP function result source.

The general alpha blending equations are:

Cd = Fs * Cs' + Fd * Cd'

Ad = Fs * As'' + Fd * Ad''

Where

- Cs' is the source color component (adjusted for NPM if necessary)
- Cd' is the destination color component (adjusted for NPM if necessary)
- As'' is the modified source alpha component
- Ad'' is the modified destination alpha component
- Fs is fraction of the source that contributes to the final value
- Fd is fraction of the destination that contributes to the final value

The blending is done in 5 logical stages (not real implementation stages):

1. Transparent/opaque conversion
   - In this stage, the incoming alpha (source or destination independently) can be inverted if needed to match the internal alpha rule. Internally, an alpha of 0 means transparent, while an alpha of "0xFF" means opaque. External content might follow the opposite rule. The output of the block is either As (Ad for destination) or 1-As (1-Ad for destination).
2. Global value substitution
   - A global alpha value from a register can be used to substitute or scale the incoming alpha. An incoming alpha As can pass-through, be directly substituted by Ags (global alpha) or scaled by the global alpha value (As * Ags). The source and destination have distinct global alpha values.
3. Blending factor generation
   - At this stages, the blending factors are generated (refer to table below). Each alpha can take the values 0, 1, A or 1-A depending on the blending mode.
4. Final blending
   - This is the final stage which implements blending equations.

The fractions take the values described in the following table, depending on the blending mode.

Table 31-4. Blending Modes Fractions Description

Blending Mode	Fs	Fd
Clear	0	0
SRC	1	0
DST	0	1
SRC_OVER	1	1 - As"
DST_OVER	1 - Ad"	1
SRC_IN	Ad"	0
DST_IN	0	As"
SRC_OUT	1 - Ad"	0
DST_OUT	0	1 - As"
SRC_ATOP	Ad"	1 - As"
DST_ATOP	1 - Ad"	As"
XOR	1 - Ad"	1 - As"

To control the blending modes, the following register fields are used:

- 1 bit for transparent/opaque conversion for source alpha
- 1 bit for transparent/opaque conversion for destination alpha
- 2 bits for source alpha modifications, to specify the 3 cases (As, Ags, As*Ags)
- 2 bits for destination alpha modifications, to specify the 3 cases (Ad, Agd, Ad*Agd)
- 4 bits to select between the 12 blending modes
- 8-bits for global source alpha
- 8-bits for global destination alpha

Alpha blending is supported on bit blit and filter blit primitives.

31.4.1.17 GPU Cache Management

SW cache flush is supported to flush the GPU cache to memory. Auto-flush of GPU cache is also supported. SW sets up the auto-flush interval, and HW will do the cache flush automatically at programmable intervals.

31.4.2 V2D GPU Operations

Information detailing the V2D GPU operations can be found here.

31.4.2.1 OPENVG 1.1-API STANDARD for VECTOR GRAPHICS ACCELERATION

OpenVG is a royalty-free, cross-platform API managed by the member-funded consortium known as Khronos Group. It provides a low-level hardware acceleration interface for vector graphics libraries such as Flash and SVG. OpenVG is used for acceleration of high-quality vector graphics for user interfaces and text on small screen devices.

31.4.2.2 Advantages of Using OpenVG

- Hardware accelerators can reduce power consumption by up to 90% compared to a software engine.
- Scalability with high-quality rendering, including anti-aliasing, to different screen sizes without multiple bitmaps.

31.4.2.3 OpenVG Target Applications

- SVG Viewers
- Portable Mapping Applications
- E-book Readers
- Games
- Scalable User Interface

31.4.2.4 OpenVG Features

31.4.2.4.1 Core API

- Coordinate Systems and Transformations (Image drawing uses a 3x3 perspective transformation matrix)
- Viewport Clipping, Scissoring and Alpha Masking
- Paths
- Images
- Image Filters
- Paint (gradient and pattern)
- Blending
- Dithering

31.4.2.4.2 The VGU Utility Library

- Higher-level Geometric Primitives
- Image Warping

31.4.2.4.3 OpenVG Rendering Pipeline

- Stage 1: Path, Transformation, Stroke, and Paint
- Stage 2: Stroked Path Generation
- Stage 3: Transformation
- Stage 4: Rasterization
- Stage 5: Clipping and Masking
- Stage 6: Paint Generation
- Stage 7: Image Interpolation
- Stage 8: Blending and Anti-aliasing

Chapter 32
3D Graphics Processing Unit (GPU3D)

32.1 Overview

The GPU3D is a high-performance core that delivers hardware acceleration for 3D graphics display.

Addressable screen sizes range from the smallest cell phones to HD 1080p displays. It provides high performance, high quality graphics, low power consumption, and the smallest silicon footprint.

GPU3D accelerates numerous 3D graphics applications, including graphical user interfaces (GUI), menu displays, flash animation, and gaming. This module supports the following graphics APIs:

- OpenGL ES 2.0
- OpenGL ES 1.1
- OpenVG 1.1
- EGL 1.4
- DirectX 11_9_3
- OpenGL 2.1 and 3.0
- OpenCL 1.1 E

32.2 GPU3D Block Diagram

The main functional units of the GPU3D are shown in the figure below and their description is as follows:

- Host Interface: Allows GPU3D to communicate with external memory and the CPU through AXI or AHB bus. In this block data crosses clock domain boundaries.
- Memory Controller: Internal memory management unit that is the block-to-host memory request interface.

- Graphic Pipeline Front End: Inserts high level primitives and commands into the graphics pipeline.
- Ultra-threaded Unified Shader: SIMD processor that performs as both vertex shader and fragment shader. When used as a vertex shader it performs geometry transformations and lighting computations. When used as a fragment shader it applies texture data and computes color values for each pixel. GPU3D has four (4) such shaders.
- 3D Rendering Engine: Converts triangles and lines into pixels. Computes slopes of color attributes and texture coordinates. Performs clipping.
- Texture Engine: Retrieves texture information from memory upon request by the fragment shader. Performs interpolation and filtering, and transfers the computed value to the fragment shader or the vertex shader. GPU3D has 2 texture units.
- Pixel Engine/Resolve: Pixel engine does alpha blending and visible surface determination. Resolve does tiling and de-tiling as well as FSAA filtering. GPU3D has two 2 pixel units.

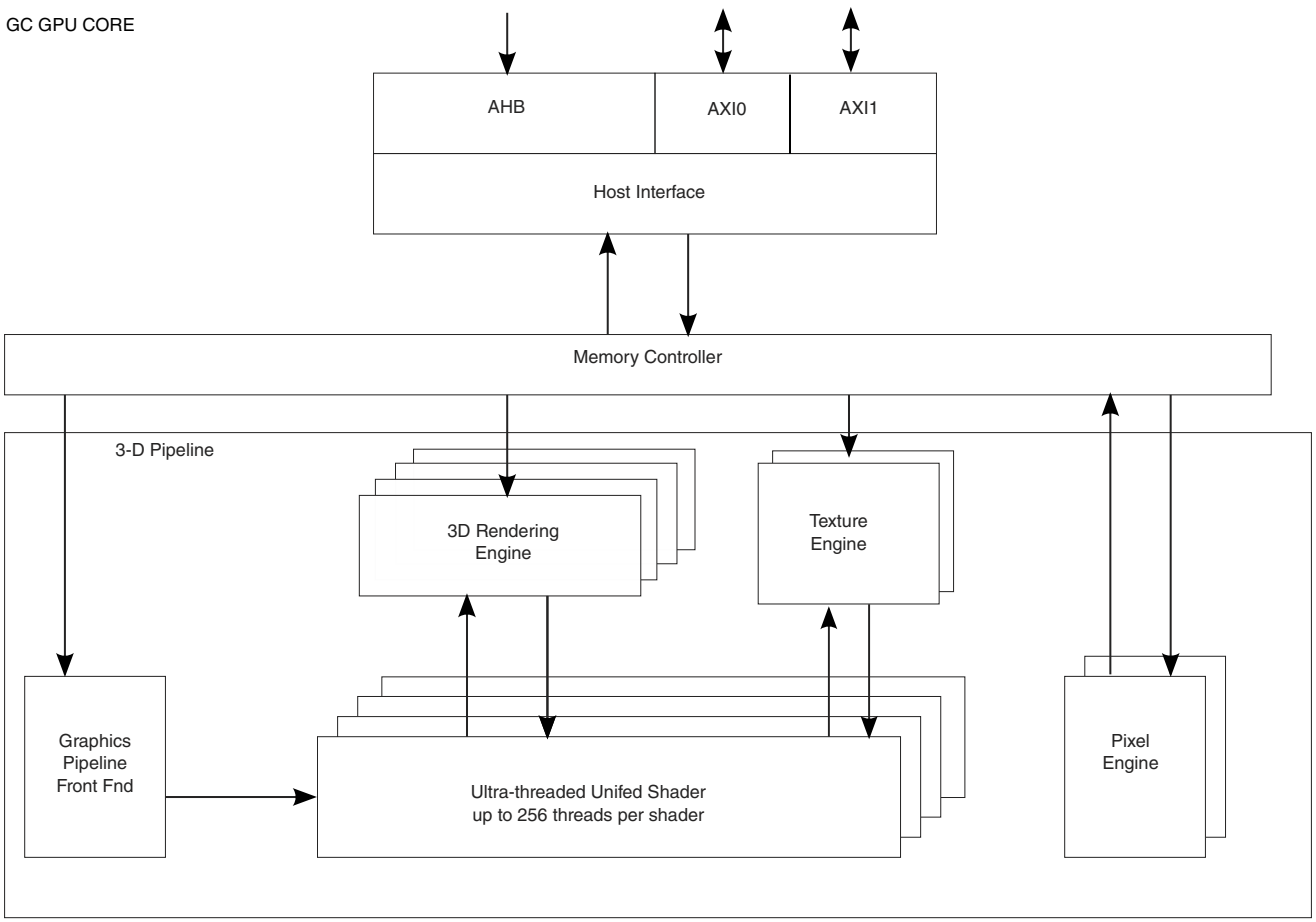


Figure 32-1. GPU3D Top Level Block Diagram

There are two main clocks for the processing control: shader clk for the shader part, and core clk for the remaining part. The core clk and shader clk frequency are programmable in CCM (as the clk frequency for GPU3D_CORE_CLK_ROOT, GPU3D_SHADER_CLK_ROOT).

The GPU3D provides architectural support for the following features:

Table 32-1. GPU3D Architectural Support

Feature	GPU Support
Primary API	OpenGLES1.1 and 2.0, Halti, OpenCL1.1
Additional API's	OpenVG1.1
	OpenGL
	OpenCL
Other Graphics Support	EGL1.4
Drivers	OpenGLES1.1 and Halti
	OpenVG1.1
	EGL1.4
	OpenGL2.1 and 3.0
	OpenCL1.1 EP
Operating Systems	Windows CE
	Linux Embedded
	Android
Z (depth)	Early Z support included
Stencil	Early stensil support included
Shader Languages	GLSL ES 1.0
Shader model compatability	Shader model 3.0
Shader types and execution units	Four programmable Scalable Ultra-threaded Unified Shaders (SIMD4:transcendental,ctl-flow,tx-load) one instruction issue per shader per clock; IEEE 32-bit floating-point pipeline supports long shader instructions
FSAA anti-aliasing mechanisms	High quality MSAA 4x; MSAA 16x for OpenVG
Code and data memory location restrictions	Unrestricted; arbitrary memory reads and writes
Physical address	31 bits
MMU description	32-bit virtual address, 4 kB pages, error reporting outside of address space
TLB	4 cache lines per requestor
Resource locks with CPU	Semaphore lock
Max memory latency without a performance hit	256 GPU cycles

32.3 GPU3D Hardware Features

The GPU3D block has the following hardware features:

- OpenGL ES 2.0 compliance, including extensions; OpenGL ES 1.1; OpenVG 1.1
- IEEE 32-bit floating-point pipeline
- Ultra-threaded, unified vertex and fragment shaders
- Low bandwidth at both high and low data rates
- Low CPU loading
- Up to 16 programmable elements per vertex
- Dependent texture operation with high-performance
- Alpha blending
- Depth and stencil compare
- Support for 16 fragment shader simultaneous textures
- Support for 16 vertex shader simultaneous textures
- Point sampling, bi-linear sampling, tri-linear filtering, and cubic textures
- Resolve and fast clear
- 8k x 8k texture size and 8k x 8k rendering target
- 8 Vertex DMA streams
- 2 texture units and 2 pixel units for higher pixel processing rate

Table 32-2. Unified vertex-fragment shader features

Feature	GPU Support
Shader type and execution units	Four (4) unified shaders, SIMD4, SFP32 Trans
Swizzle capabilities	Full 32-bit word level swizzle in a 128-bit vector
GPR's per shader	Up to 512 general purpose registers, 128 bits each
Uniform registers	Vertex Shader: 168 registers, 128 bits each Fragment Shader: 64 registers, 128 bits each
FP denorm and rounding options	Denorms are set to zero. Supports rounding to zero.
Maximum number of data input attributes	Maximum of 16 vertex shader input elements; maximumof 12 fragment shader input elements;
Maximum number of instructions	512 unified for both vertex and fragment shaders
Maximum number of vertex streams	8
Maximum number of threads in flight per shader	256
Subroutines	4 levels
Conditional branch support	GT, LT, EQ, GE, LE, NE, ISNAN, ISFINITY, ISINFINITY, ISNORMAL, AND, OR, XOR, NOT, ANYMSB, ALLMSB, SELMSB
Shader instruction rate	1-cycle throughput for all shader instructions
Floating-point instruction precision	Transcendental: 22 bits SIMD4 (vector): 23.5 bits
Fragment shader video	Supports video texture

Table continues on the next page...

Table 32-2. Unified vertex-fragment shader features (continued)

Feature	GPU Support
Control flow instructions	Yes
Standard derivatives	Yes
Integer pipeline	Support 8, 16, 32-bit integer operations
Local shared memory	1 kB
L1 Cache	4 kB

Table 32-3. Vertex Processing Features

Feature	GPU Support
Vx D3D, OGL ES formats supported	BYTE, UBYTE, SHORT, USHORT, INT, UINT, DEC, UDEC, FLOAT, FLOAT16, D3DCOLOR, FIXED16DOT16,R10B10G10A02
Vertex data size limits	256 bytes
Pre shader cache	2 kB
Post shader cache	8/16 vertices

Table 32-4. Primitive Processing Features

Feature	GPU Support
Primitives supported	triangle strip, fan, and list; line strip and list; point list
Vertex/primitive geometry input index sizes	8-bit ,16-bit and 32-bit indices
Setup parameters available to fragment shader	12 vec4 parameters; all available to fragment shader
Primitive restart	Yes

Table 32-5. Texture Processing Features

<table>
<tr><th>Feature</th><th>GPU Support</th></tr>
<tr><td>Fixed-point input texture formats</td><td>A8, L8, I8, A8L8, ARGB4, XRGB4, ARGB8, XRGB8, ABGR8, XBGR8, R5G6B5, A1RGB5, X1RGB5, YV12, YUY2, UYVY, D16, D24S8, A8_OES, DXT1, DXT2, DXT3, DXT4, DXT5, ETC1,A8G8,R10G10B10A02,ETC2,EAC11; all fixed-point formats are filtered. sRGB conversion and Swizzling supported for color channels.
<table>
<tr><th>Bits</th><th>Format</th><th>Alpha</th><th>R</th><th>B</th><th>G</th></tr>
<tr><td>16</td><td>ARGB4444</td><td>4</td><td>4</td><td>4</td><td>4</td></tr>
<tr><td>16</td><td>XRGB4444</td><td>4 don't care</td><td>4</td><td>4</td><td>4</td></tr>
<tr><td>16</td><td>ARGB1555</td><td>1</td><td>5</td><td>5</td><td>5</td></tr>
<tr><td>16</td><td>XRGB1555</td><td>1 don't care</td><td>5</td><td>5</td><td>5</td></tr>
<tr><td>16</td><td>RGB565</td><td>0</td><td>5</td><td>6</td><td>5</td></tr>
<tr><td>32</td><td>ARGB8888</td><td>8</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>XRGB8888</td><td>8 don't care</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>ABGR8888</td><td>8</td><td>8</td><td>8</td><td>8</td></tr>
</table>
</td></tr>
</table>

Table continues on the next page...

Table 32-5. Texture Processing Features (continued)

Feature	GPU Support
	32 | XBGR8888 | 8 don't care | 8 | 8 | 8 Planes | Format | Mode | Y | U | V | UV | YUYV | UYVY 3 | YV12 | 4:2:0 | 1 | 1 | 1 | | | 2 | NV12 | 4:2:0 | 1 | | | 1 | | 1 | YUY2 | 4:2:2 | | | | | 1 | 1 | UYVY | 4:2:2 | | | | | | 1
Texture compression	4 bits and 8 bits per texel
Compressed texture formats	DXT1, DXT2, DXT3, DXT4, DXT5, ETC1, ETC2, EAC11 All compressed formats are filtered.
Texture size maximum	8k x 8k
Addressing modes	Wrap, mirror, clamp
Mipmap support	14 mipmap levels; programmable LOD biasing & replacement
Shadow texture	Depth texture PDF filtering
Texture cache organization	Tiled, 4x4 texels; Linear, 1x16 texels
Texture cache size	32 cache lines, with 64 bytes per cache line; 2 KB texture cache total
Texture coordinate fraction bits	5 bits
Texture sampler units	12 samplers, indexable
Textures per fragment maximum	8 texture samplers
Dependent texture operation	High performance; unlimited dependent texture reads
Dependent tx per fragment max, relative sampling	No limit
Texture repeat max	256
Texture types	2D, cube map, 1D, projected, depth, bump map, displacement map, PCF, 3D, texture array
Texture filters	Point sample, bi-linear, tri-linear, quad-linear, Anistropic
Texture component mapping: D3D, OGL, ES options	Supports both D3D and OES options
Texture size types	Power-of-2, Non-square, Non-power-of-2
Texture swizzle	Yes

32.3.1 Rasterization

Table 32-6. Rasterization features

Feature	GPU Support
Interpolant attributes limit	12

Table continues on the next page...

Table 32-6. Rasterization features (continued)

Hierarchical Z	Yes
Render target size	8K x 8K
Clipping window support	Yes
Centroid interpolation	Yes

32.3.2 Fragment Processing

Table 32-7. Fragment Processing Features

Feature	GPU Support
FSAA mechanisms	MSAA 4x, SSAA 4x;
FSAA mechanisms using fragment centroid	MSAA 4x
Fragment color, alpha, Z, stencil precision	

Table continues on the next page...

Table 32-7. Fragment Processing Features (continued)

<table>
<tr><td></td><td>
<table>
<tr><th>Bits</th><th>Format</th><th>Alpha</th><th>R</th><th>B</th><th>G</th></tr>
<tr><td>16</td><td>ARGB4444</td><td>4</td><td>4</td><td>4</td><td>4</td></tr>
<tr><td>16</td><td>XRGB4444</td><td>4
don't care</td><td>4</td><td>4</td><td>4</td></tr>
<tr><td>16</td><td>ARGB1555</td><td>1</td><td>5</td><td>5</td><td>5</td></tr>
<tr><td>16</td><td>XRGB1555</td><td>1
don't care</td><td>5</td><td>5</td><td>5</td></tr>
<tr><td>16</td><td>RGB565</td><td>0</td><td>5</td><td>6</td><td>5</td></tr>
<tr><td>32</td><td>ARGB8888</td><td>8</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>XRGB8888</td><td>8
don't care</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>ABGR8888</td><td>8</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>XBGR8888</td><td>8
don't care</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>RGB10_a2ui</td><td>2</td><td>10</td><td>10</td><td>10</td></tr>
</table>
<table>
<tr><th>Bits</th><th>Format</th><th>Depth</th><th>Stencil</th></tr>
<tr><td>16</td><td>D16</td><td>16</td><td>0</td></tr>
<tr><td>32</td><td>D2488</td><td>24</td><td>8</td></tr>
</table>
</td></tr>
<tr><td>Fragment storage</td><td>16-bit color and z, 32-bit color and z for each fragment. Lossless compression, no storage reduction.</td></tr>
<tr><td>Individual fragment alpha masking</td><td>Yes</td></tr>
<tr><td>Two sided stencil support</td><td>Yes</td></tr>
<tr><td>Fragment cache</td><td>32 cache lines for color 32 cache lines for Z 64 bytes per cache line</td></tr>
<tr><td>Multiple render target</td><td>Yes</td></tr>
</table>

32.3.3 Dest/Alpha Blending

Table 32-8. Dest/Alpha Blending Features

<table>
<tr><th>Feature</th><th>GPU Support</th></tr>
<tr><td>Destination color formats</td><td>
<table>
<tr><th>Bits</th><th>Format</th><th>Alpha</th><th>R</th><th>B</th><th>G</th></tr>
<tr><td>16</td><td>ARGB4444</td><td>4</td><td>4</td><td>4</td><td>4</td></tr>
<tr><td>16</td><td>ARGB1555</td><td>1</td><td>5</td><td>5</td><td>5</td></tr>
<tr><td>16</td><td>RGB565</td><td>0</td><td>5</td><td>6</td><td>5</td></tr>
<tr><td>32</td><td>ARGB8888</td><td></td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>ABGR8888</td><td></td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>32</td><td>RGB10_a2ui</td><td></td><td>10</td><td>10</td><td>10</td></tr>
</table>
</td></tr>
<tr><td>Blend modes</td><td>Porter-Duff blending modes</td></tr>
<tr><td>Render target dithering support</td><td>Yes</td></tr>
</table>

32.3.4 Z/Stencil Buffer

Table 32-9. Z/Stencil Buffer Features

Feature	GPU Support
Z/stencil formats	16-bit Z; 24-bit Z plus 8-bit stencil, with lossless compression support
Z/stencil buffer	32 cache lines; 64 bytes per line;
Z compression	Yes, 30% - 70%
Stencil compression	Yes
Fast clear and initialization options	Yes

32.3.5 Render Target

Feature	GPU Support
Formats	16-bit and 32-bit, with lossless compression support
RT buffer cache	32 cache lines; 64 bytes per line; RT caches are fully set associative.
Compressed formats	Yes, 30% - 70%
Fast clear and initialization options	Yes
Multi-Render Target support	4 Targets

32.4 Usage Mode

The GPU3D should be programmed through the Freescale provided driver. Freescale does not provide support for software that directly programs the GPU3D registers. APIs for programming the GPU3D through the software driver are described in separate driver documentation.

Chapter 33
HDMI Transmitter (HDMI)

33.1 Overview

33.1.1 HDMI Operational Model Overview

The High Definition Multimedia Interface (HDMI) is a wired digital interconnect that replaces the analog TV out or VGA out.

HDMI is capable of transferring uncompressed video, audio, and data using a single cable. The video pixel rates are typically from 25 MHz up to 266 MHz (and 3D video modes), but HDMI can support higher rates up to 266 MHz. It can support S/PDIF (IEC60958 L-PCM and IEC61937 compressed non-linear PCM: AC-3, MPEG-1/-2 Audio, DTS®, MPEG-2/-4 AAC, ATRAC, WMA, MAT) and Parallel HBR (high bit rate) audio interface, enabling the support of Dolby® True-HD and DTS-HD Master Audio. HDMI has the capability of automatically setting the display format configuration (intelligent link).

HDMI include a content protection system called HDCP (High-bandwidth Data Content Protection). The HDMI connections can be used to connect DVD recorders, set-top boxes, and game consoles to flat panel televisions and an AV amplifier that can act as repeater/router.

HDMI system architecture consists of sources (transmitter) and sinks (receiver). As shown in the figure below, the HDMI cable and connectors carry four differential pairs that make up the TMDS data and clock channels. These channels are used to carry video, audio, and auxiliary data. In addition, HDMI carries a VESA Data Display Channel (DDC). The DDC is used for configuration and status exchange between a single source and a single sink. The optional CEC protocol provides high-level control functions between all of the various audiovisual products in a user's environment.

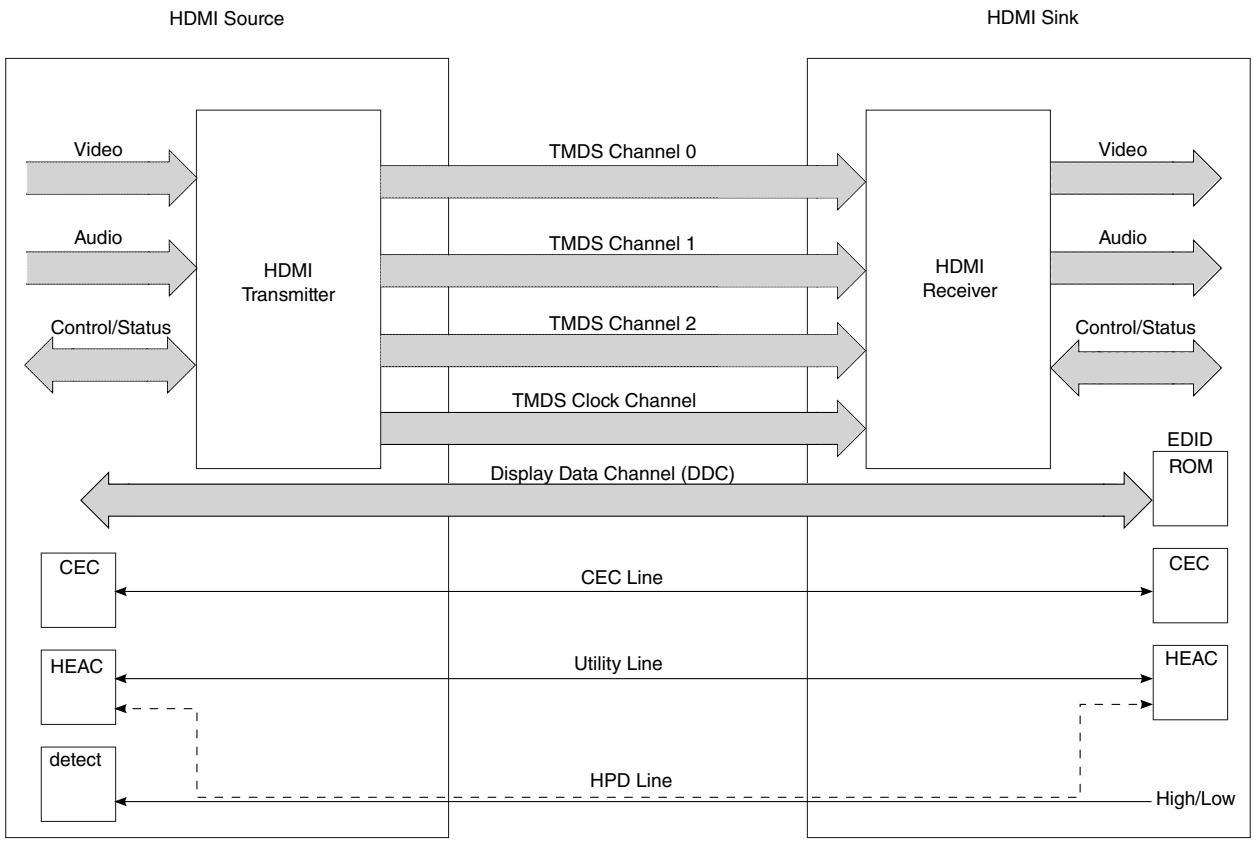


Figure 33-1. HDMI Block Diagram

Audio, video, and auxiliary data is transmitted across the three TMDS data channels. A TMDS clock running at 1x (24-bit true color mode), the video pixel rate is transmitted on the TMDS clock channel and used by the receiver as a frequency reference for data recovery on the three TMDS data channels. Video data can have a pixel size of 24bits . Video at the default 24-bit color depth is carried at a TMDS clock rate equal to the pixel clock rate. Higher color depths are carried using a correspondingly higher TMDS clock rate. Video formats with TMDS rates below 25MHz (such as, 13.5MHz for 480i/NTSC) can be transmitted using a pixel-repetition scheme. The video pixels can be encoded in either RGB, YCBCR 4:4:4, or YCBCR 4:2:2 formats.

HDMI uses a packet structure to transmit audio and auxiliary data across the TMDS channels. To attain the highest reliability required of audio and control data, this data is protected with a BCH error correction code and is encoded using a special error reduction code to produce the transmitted 10-bit word.

Basic audio functionality consists of a single IEC 60958 L-PCM audio stream (two audio channels) at sample rates of 32 KHz, 44.1 KHz, or 48 KHz, which can accommodate any normal stereo stream. Optionally, HDMI can carry audio at sample rates up to 192KHz

and with three to eight audio channels. HDMI can also carry an IEC 61937 compressed (such as, surround sound) audio stream at bit rates up to 24.576 Mbps. For bit rates above 6.144 Mbps, compressed audio streams conforming to IEC 61937 are carried using HBR Audio Stream Packets. Each packet carries four IEC 60958 frames, which corresponds to (4x2x16 =) 128 contiguous bits of an IEC 61937 stream.

The source uses the DDC to read the sink's Enhanced Extended Display Identification Data (E-EDID) to obtain the sink's configuration and/or capabilities.

The HDMI TX Controller schedules the three periods: Video Data Period, Data Island period, and Control period. During the Video Data Period, the active pixels of an active video line are transmitted. During the Data Island period, audio and auxiliary data are transmitted using a series of packets. The Control period is used when no video, audio, or auxiliary data needs to be transmitted. A Control Period is required between any two periods that are not Control Periods.

An example of each period placement is shown in the figure below.

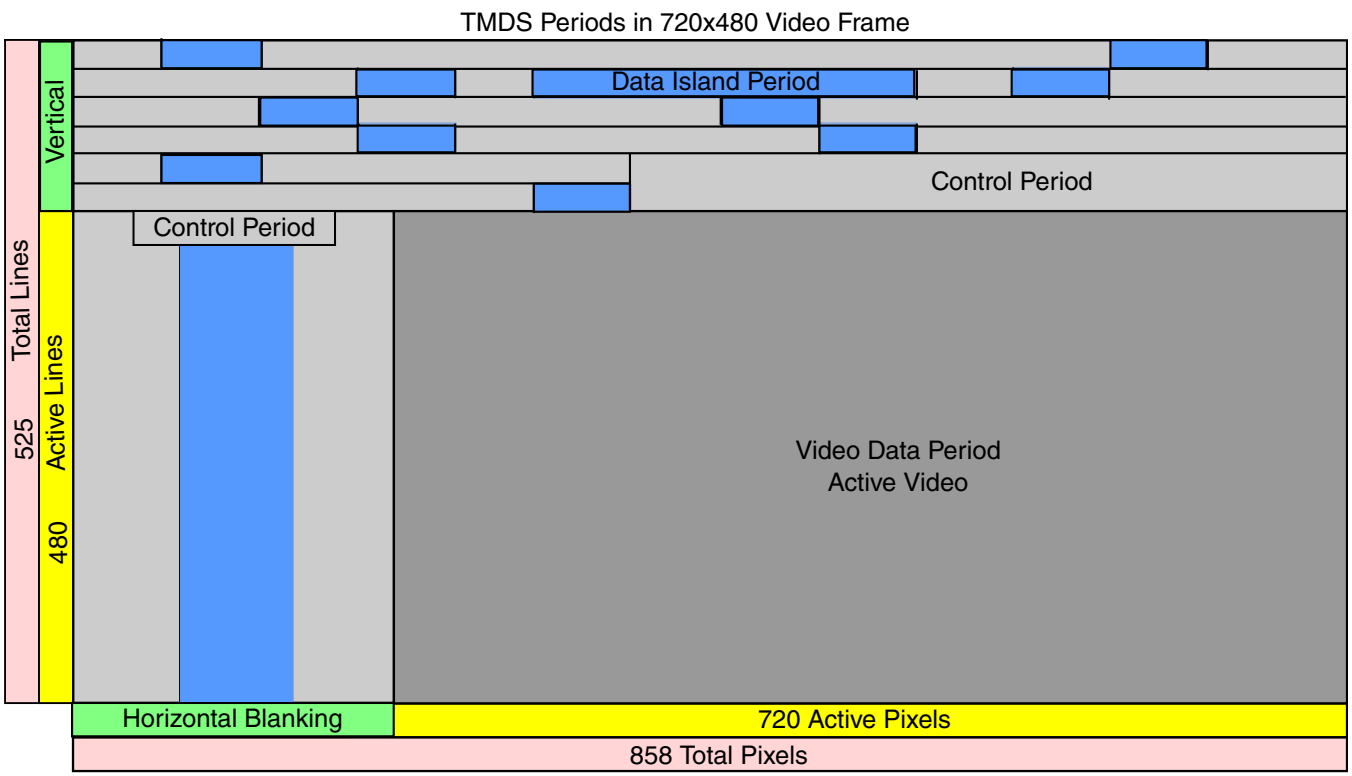


Figure 33-2. TMDS Periods in 720x480p Video Frame

33.1.1.1 Interfaces

HDMI TX has the following interfaces:

- HDCP interface
- External ROM interface for key storage
- External RAM interface for revocation
- Random number generator interface
- Video input interface
- RGB 4:4:4
- YCbCr4:2:2
- YCbCr4:4:4
- Digital audio input interface
- AHB audio DMA
- System interface
- AMBA AHB
- Scan test interface
- HDMI TX PHY interface
- CEC interface

33.1.1.2 Features

HDMI TX includes the following features:

- Supported video formats:
- All CEA-861-E video formats up to 1080p at 60Hz and 720p/1080i at 120Hz
- Supported colorimetry:
- 24bit RGB 4:4:4
- 24bit YCbCr 4:4:4
- 16bit YCbCr 4:2:2
- xvYCC601
- xvYCC709
- Integrated color space converter:
- RGB(4:4:4) to/from YCbCr(4:4:4 or 4:2:2)
- Optional HDMI 1.4a supported video formats:
- HDMI 1.4a 3D video modes with up to 266MHz (TMDS clock)
- Optional HDMI 1.4a supported colorimetry:
- sYCC601
- Adobe RGB
- Adobe YCC601
- Optional HDMI 1.4a supported Infoframes:
- Audio InfoFrame packet extension to support LFE playback level information
- AVI infoFrame packet extension to support YCC Quantization range (Limited Range, Full
- Range)

- AVI infoFrame packet extension to support Content type (Graphics, Photo, Cinema, Game)
- Supported Audio formats:
- Up to four I^2S interface for eight-channel Linear-PCM audio
- S/PDIF interface for linear and non-linear PCM formats:
  - AC-3
  - MPEG-1/-2 Audio
  - DTS
  - MPEG- 2/-4 AAC
  - ATRAC
  - WMA
  - MAT
- Parallel audio interface for High-Bit Rate (HBR) Audio:
  - Dolby® True-HD
  - DTS®-HD Master Audio
  - Generic Parallel Audio interface
- AHB DMA Audio interface
- Up to 192 KHz IEC60958 audio sampling rate
- Pixel clock from 13.5MHz up to 266 MHz
- Option to remove pixel repetition clock (prepclk) from HDMI TX interface for an easy integration with third-party HDMI TX PHYs
- Flexible synchronous enable per clock domain to set functional power down modes
- Register access:
- AMBA AHB
- I^2C DDC, EDID block read mode
- Advanced PHY testability
- Integrated CEC hardware engine

33.2 External Signals

See HDMI PHY for external signal information.

33.3 Clocks

The table found here describes the clock sources for HDMI.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 33-1. HDMI Clocks

Clock name	Clock Root	Description
iahbclk	ahb_clk_root	Bus clock
icecclk	ckil_sync_clk_root	CEC low-frequency clock (32kHZ)
ihclk	ahb_clk_root	Module clock
isfrclk	video_27m_clk_root	Internal SFR clock (video clock 27MHz)

33.4 Functional Description

This section describes the functional architecture of the HDMI TX controller.

33.4.1 HDMI TX Functional Overview

The HDMI TX provides a variety of standard audio, video, and system interfaces.

It includes an high-bandwidth data content protection (HDCP) encryption engine for HDMI receiver authentication, revocation, and data encryption.

Figure below illustrates the top level diagram of the HDMI TX solution.

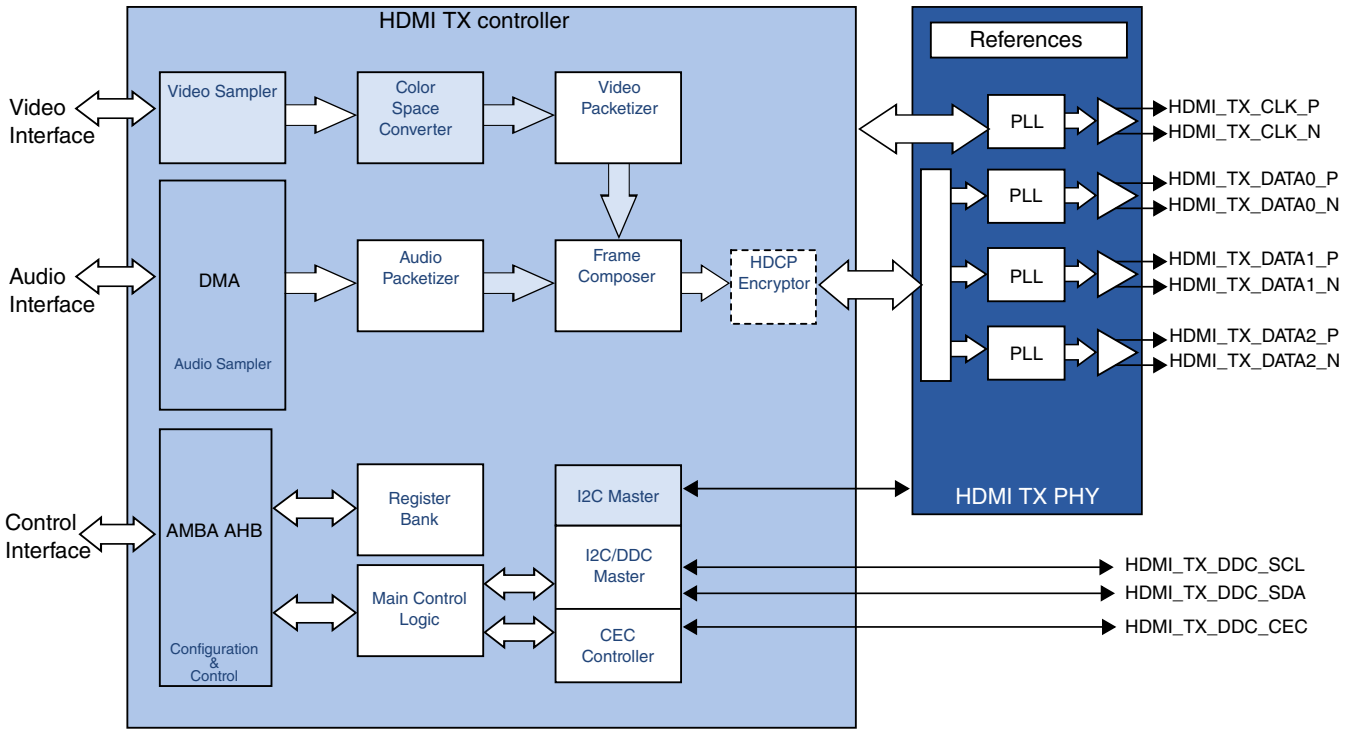


Figure 33-3. HDMI TX Top Level Block Diagram

The HDCP encryption engine is responsible for HDMI receiver authentication, revocation, and data encryption.

The input video stream can be either RGB 4:4:4, YcbCr 4:2:2, or YcbCr 4:44 in single data rate (SDR) bus formats as described in Table 33-2. The video mode's timing format must follow the CEA-861-E specification. An embedded color space conversion allows the pixel color format to be converted on the HDMI source side to match the best with the HDMI sink capabilities. 24

The input audio stream can be provided through audio AHB DMA interface;

Finally, HDMI_TX can output video in full HD with up to 48-bit color mode and inserts high fidelity audio up to eight-channels over low resolution video formats by performing automatic pixel repetition over the input video stream.

33.4.2 Video Pixel Sampler

The Video pixel sampler block is responsible for the video data synchronization, according to the video data input mapping defined by the Color Depth (Deep Color) and format configuration.

Optionally, for YCbCr 4:2:2 format, data mapping can be performed to conform to ITU. 601 and ITU.656 standards but without the support of embedded synchronizers.

The video pixel sampler registers base address is 0x0200.

33.4.2.1 HDMI Transmitter Controller Databook Functional Description

Table 33-2. Input Data Mapping

Video Input Format		ivdata[47:0] mapping																								
Color Space	Color Depth	47-46	45-44	43-42	41-40	39-38	37-35	35-34	33-32	31-30	29-28	27-26	25-24	23-22	21-20	19-18	17-16	15-14	13-12	11-10	9-8	7-6	5-4	3-2	1-0
RGB 4:4:4	8-bit	R[7:0]								G[7:0]								B[7:0]							
YCbCr 4:4:4	8-bit	Cb[7:0]								Y[7:0]								Cr[7:0]							
YCbCr 4:2:2	8-bit	Cb[7:0]								Y[7:0]															
		Cr[7:0]								Y[7:0]															

For each video timing format, there is a specific timing parameters defined in the CEA-861-E specification.

The following timing diagram is an example for the video mode format 1 (640x480p @ 59.94/60 Hz): Data Enable = idataen, HSYNC = ihsync, VSYNC = ivsync.

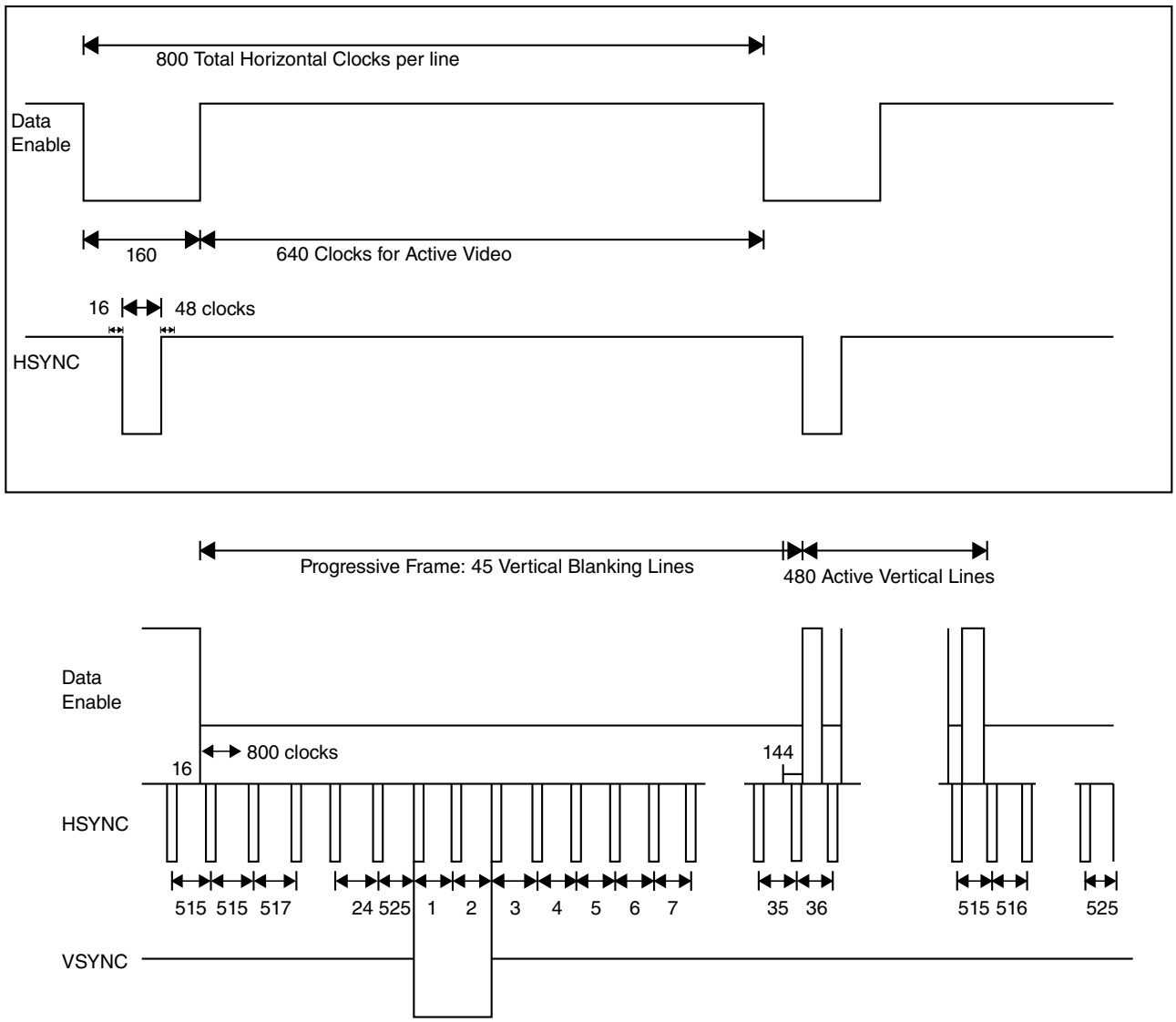


Figure 33-4. Timing Parameters for 640x480p @ 59.94/60 Hz

For a complete list of timing parameters and diagrams, refer to the CEA-861-E specification. The SDR video sample input format is illustrated in the figure below.

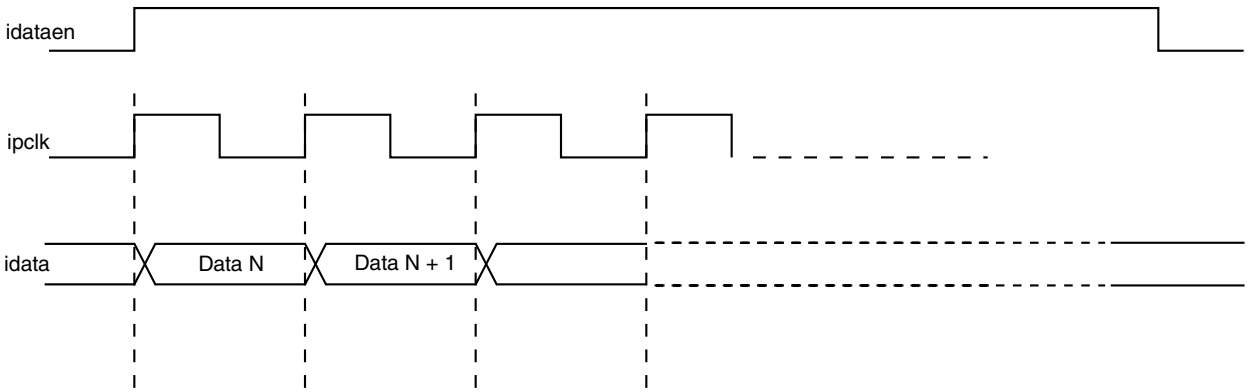


Figure 33-5. Video Sample Timing Interface for RGB and YCbCr SDR Format

33.4.3 Supported Video Mode

The table below shows examples of the supported video modes.

Table 33-3. Video Modes

				2D	3D Structure							
VideoMode	Mode	H x V Active Resolution (pixel)	Refresh Rate (Hz)	2D Pixel Rate (Mp/s)	Frame Packing Pixel Rate (Mp/s)	Field Alt. Pixel Rate (Mp/s)	Line Alt. Pixel Rate (Mp/s)	Side-by-Side (full) Pixel Rate (Mp/s)	L+depth Pixel Rate (Mp/s)	L +depth + graphics+ graphics-depth Pixel Rate (Mp/s)	Side-by-Side (Half) Pixel Rate (Mp/s)	Top-and-Bottom Pixel Rate (Mp/s)
					ALL CEA	Interlaced Only	Progr. Only	ALL CEA	Progr. Only	Progr. Only	ALL CEA	ALL CEA
Primary HDMI Video Format Timings (CEA-861-E)												

Table continues on the next page...

Table 33-3. Video Modes (continued)

1	640x480p (EDTV)	640 x 480	59.94	25.18	50.35		50.35	50.35	50.35	100.70	25.18	25.18
1			60.00	25.2	50.40		50.40	50.40	50.40	100.80	25.2	25.2
19	1280x720p (HDTV)	1280 x 720	50.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
4			59.94	74.18	148.35		148.35	148.35	148.35	296.70	74.18	74.18
4			60.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
20	1920x1080i (HDTV)	1920 x 1080	50.00	74.25	148.50	148.50		148.50			74.25	74.25
5			59.94	74.18	148.35	148.35		148.35			74.18	74.18
5			60.00	74.25	148.50	148.50		148.50			74.25	74.25
2.3	720x480p (EDTV)	720 x 480	59.94	27.00	54.00		54.00	54.00	54.00	108.00	27.00	27.00
2.3			60.00	27.03	54.05		54.05	54.05	54.05	108.11	27.03	27.03
6.7	720(1440)x480i (SDTV)	1440 x 480	59.94	27.00	54.00	54.00		54.00			27.00	27.00
6.7			60.00	27.03	54.05	54.05		54.05			27.03	27.03
17.18	720x576p (EDTV)	720 x 576	50.00	27.00	54.00			54.00			27.00	27.00
21.22	720(1440)x576i (SDTV)	1440 x 576	50.00	27.00	54.00	54.00		54.00			27.00	27.00
Secondary HDMI video format timings (CEA-861-E)												
8.9	720(1440)x240p	1440 x 240	59.94	27.00	54.00		54.00	54.00	54.00	108.00	27.00	27.00
8.9			60.00	27.03	54.05		54.05	54.05	54.05	108.11	27.03	27.03
10.11	1440(2880)x480i	2880 x 480	59.94	54.00	108.00	108.00		108.00			54.00	54.00
10.11			60.00	54.05	108.11	108.11		108.11			54.05	54.05
12.13	1440(2880)x240p	2880 x 240	59.94	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
12.13			60.00	54.05	108.11		108.11	108.11	108.11	216.22	54.05	54.05
32	1920x1080p (HDTV)	1920 x 1080	23.98	74.18	148.35		148.35	148.35	148.35	296.70	74.18	74.18
32			24.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
33			25.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
34			29.97	74.18	148.35		148.35	148.35	148.35	296.70	74.18	74.18
34			30.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
31			50.00	148.5	297.00		297.00	297.00	297.00		148.5	148.5
16			59.94	148.35	296.70		296.70	296.70	296.70		148.35	148.35
16			60.00	148.5	297.00		297.00	297.00	297.00		148.5	148.5
64			100.00	297.00							297.00	297.00

Table continues on the next page...

Table 33-3. Video Modes (continued)

63			120.00	297.00							297.00	297.00
40	1920x1080i (HDTV)	1920 x 1080	100.00	148.5	297.00	297.00		297.00			148.5	148.5
46			119.88	148.35	296.70	296.70		296.70			148.35	148.35
46			120.00	148.5	297.00	297.00		297.00			148.5	148.5
60	1280x720p (HDTV)	1280 x 720	24.00	59.40	118.80		118.80	118.80	118.80	237.60	59.40	59.40
61			25.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
62			30.00	74.25	148.50		148.50	148.50	148.50	297.00	74.25	74.25
41			100.00	148.5	297.00		297.00	297.00	297.00		148.5	148.5
47			119.88	148.35	296.70		296.70	296.70	296.70		148.35	148.35
47			120.00	148.5	297.00		297.00	297.00	297.00		148.5	148.5
29, 30	1440x576p (EDTV)	1440 x 576	50.00	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
37.38	2880x576p (EDTV)	2880 x 576	50.00	108.00	216.00		216.00	216.00	216.00		108.00	108.00
14.15	1440x480p (EDTV)	1440 x 480	59.94	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
14.15			60.00	54.05	108.11		108.11	108.11	108.11	216.22	54.05	54.05
35.36	2880x480p (EDTV)	2880 x 480	59.94	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
35.36			60.00	54.05	108.11		108.11	108.11	108.11	216.22	54.05	54.05
48.49	720x480p (EDTV)	720 x 480	119.88	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
48.49			120.00	54.05	108.11		108.11	108.11	108.11	216.22	54.05	54.05
56.57			239.76	108.00	216.00		216.00	216.00	216.00		108.00	108.00
56.57			240.00	108.11	216.22		216.22	216.22	216.22		108.11	108.11
50.51	720(1440)x480i (SDTV)	1440 x 480	119.88	54.00	108.00	108.00		108.00			54.00	54.00
50.51			120.00	54.05	108.11	108.11		108.11			54.05	54.05
58.59			239.76	108.00	216.00	216.00		216.00			108.00	108.00
58.59			240.00	108.11	216.22	216.22		216.22			108.11	108.11
42.43	720x576p (EDTV)	720 x 576	100.00	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00

Table continues on the next page...

Table 33-3. Video Modes (continued)

52.53			200.00	108.00	216.00		216.00	216.00	216.00		108.00	108.00
44.45	720(1440)x576i (SDTV)	1440 x 576	100.00	54.00	108.00	108.00		108.00			54.00	54.00
54.55			200.00	108.00	216.00	216.00		216.00			108.00	108.00
23.24	720(1440)x288p	1440 x 288	50.00	27.00	54.00		54.00	54.00	54.00	108.00	27.00	27.00
25.26	720(1440)x576i	1440 x 576	50.00	54.00	108.00	108.00		108.00			54.00	54.00
27.28	1440(2880)x288p	2880 x 288	50.00	54.00	108.00		108.00	108.00	108.00	216.00	54.00	54.00
39	1920x1080i	1920 x 1080	50.00	72.00	144.00	144.00		144.00			72.00	72.00

33.4.4 Video Packetizer

This block is responsible for:

- Pixel repetition (if not already performed in the input video stream and needed by the user)
- 10-bit, 12-bit, and 16-bit packing when in deep color modes
- YCC 422 remapping according to the HDMI 1.4a specification
- Clock rate transformation from pixel or repetition clock to the final TMDS clock domain (by means of FIFOs)

The figure below depicts a functional diagram of the Video Packetizer block.

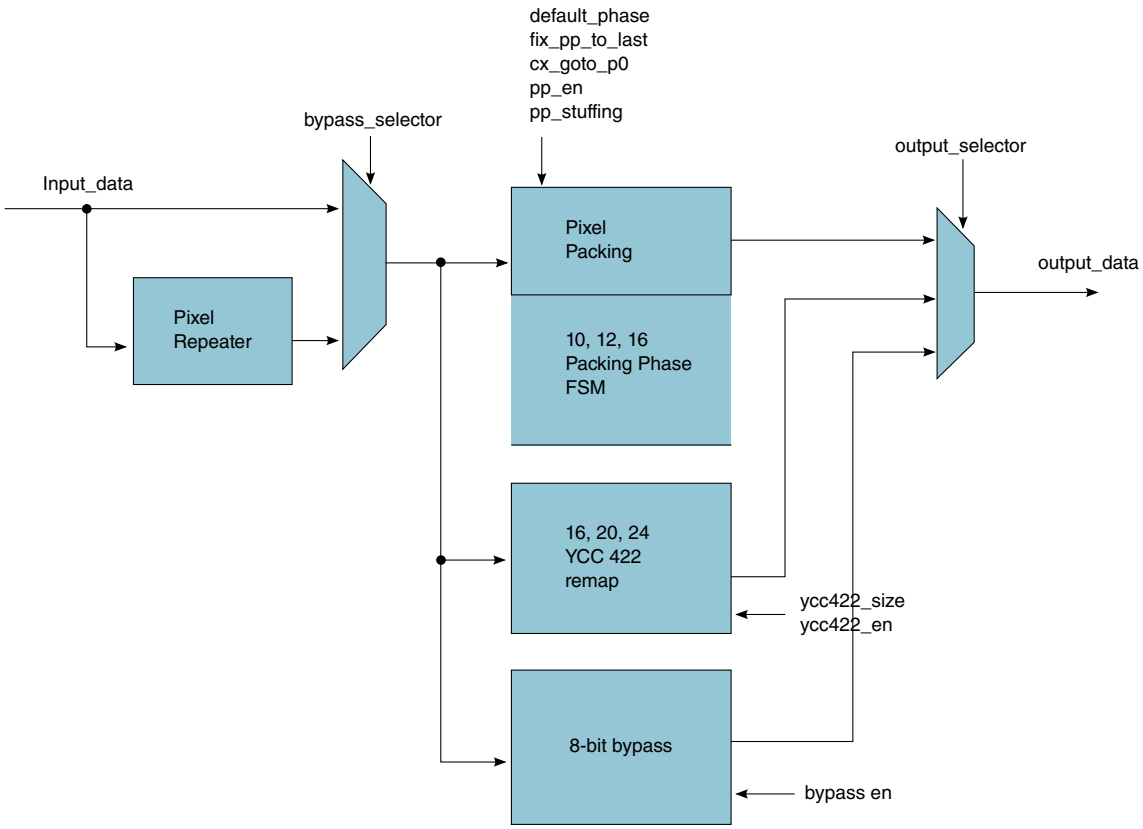


Figure 33-6. Video Packetizer Functional Diagram

33.4.5 Color Space Conversion

This block is responsible for carrying out the following video color space conversion functions:

- RGB to/from YCbCr
- 4:2:2 to/from 4:4:4 up (pixel repetition or linear interpolation)/down-converter
- Limited to/from full quantization range conversion

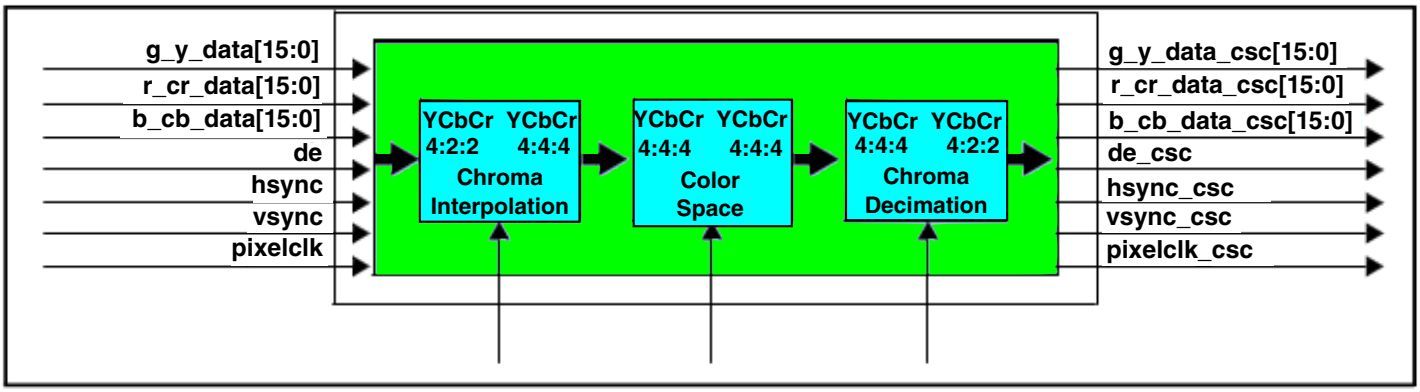


Figure 33-7. Color Space Converter Simplified Block Diagram

The Color Space Converter (CSC) supports all the timings reported in the CEA-861-D specification and the following pixel modes:

- RGB444 and YCbCr444: 24, 30, 36, and 48 bits
- YCbCr422: 16, 20, and 24 bits

The color space conversion matrix is ruled by the following equations listed below. The color space conversion registers base address is 0x4100.

$$out_1 = (X_1 \times in_1 /4096 + X_2 \times in_2 /4096 + X_3 \times in_3 /4096 + X_4) \times 2^{scale}$$

$$out_2 = (Y_1 \times in_1 /4096 + Y_2 \times in_2 /4096 + Y_3 \times in_3 /4096 + Y_4) \times 2^{scale}$$

$$out_3 = (Z_1 \times in_1 /4096 + Z_2 \times in_2 /4096 + Z_3 \times in_3 /4096 + Z_4) \times 2^{scale}$$

33.4.6 Audio Interfaces

The supported audio input interfaces are:

- AHB Direct Memory Access (DMA)

No lipsync support is available inside the HDMI TX. If necessary, this feature can be performed at the system audio processor side. From the HDMI TX, no audio/video delay or skew is added.

The audio sampler registers base address is 0x3100.

33.4.6.1 CTS Calculation

Because there is no audio clock carried through the HDMI link, only the pixel clock is used.

The CTS/N has to be set by software with value taken in the following table. Table below shows the CTS and N value for the supported standard. All other TMDS clocks are not supported; the TMDS clocks divided or multiplied by 1,001 coefficients are not supported.

Table 33-4. N and CTS for 8-Bit Color Depth

	TMDS Clock (MHz)											
	25.2		27		54		74.25		148.5		297	
Fs (kHZ)	N	CTS	N	CTS	N	CTS	N	CTS	N	CTS	N	CTS
32	4096	25200	4096	27000	4096	54000	4096	74250	4096	148500	3072	222750
44.1	6272	28000	6272	30000	6272	60000	6272	82500	6272	165000	4704	247500
48	6144	25200	6144	27000	6144	54000	6144	74250	6144	148500	5120	247500
88.2	12544	28000	12544	30000	12544	60000	12544	82500	12544	165000	9408	247500
96	12288	25200	12288	27000	12288	54000	12288	74250	12288	148500	10240	247500
176.4	25088	28000	25088	30000	25088	60000	25088	82500	25088	165000	18816	247500
192	24576	25200	24576	27000	24576	54000	24576	74250	24576	148500	20480	247500
768	Used for HBR audio only. N and CTS configured for Fs=192kHZ (1/4th ACR value per spec)											

To support the deep color mode and/or 3D video modes, the TMDS clock is multiplied by 4, 2, 1.5, or 1.25, depending on the mode. In this case, the CTS value must also follow the same ratio.

33.4.6.2 Audio DMA Interface

This audio direct memory access (DMA) interface is intended for advanced systems running 32-bit CPU SoC solutions.

This module provides a direct audio DMA interface, which is useful in systems where a DSP handles audio processing. In these systems, sending the incoming audio samples directly to the memory provides a cleaner architecture to the SoC, without the overhead of converting several audio standards.

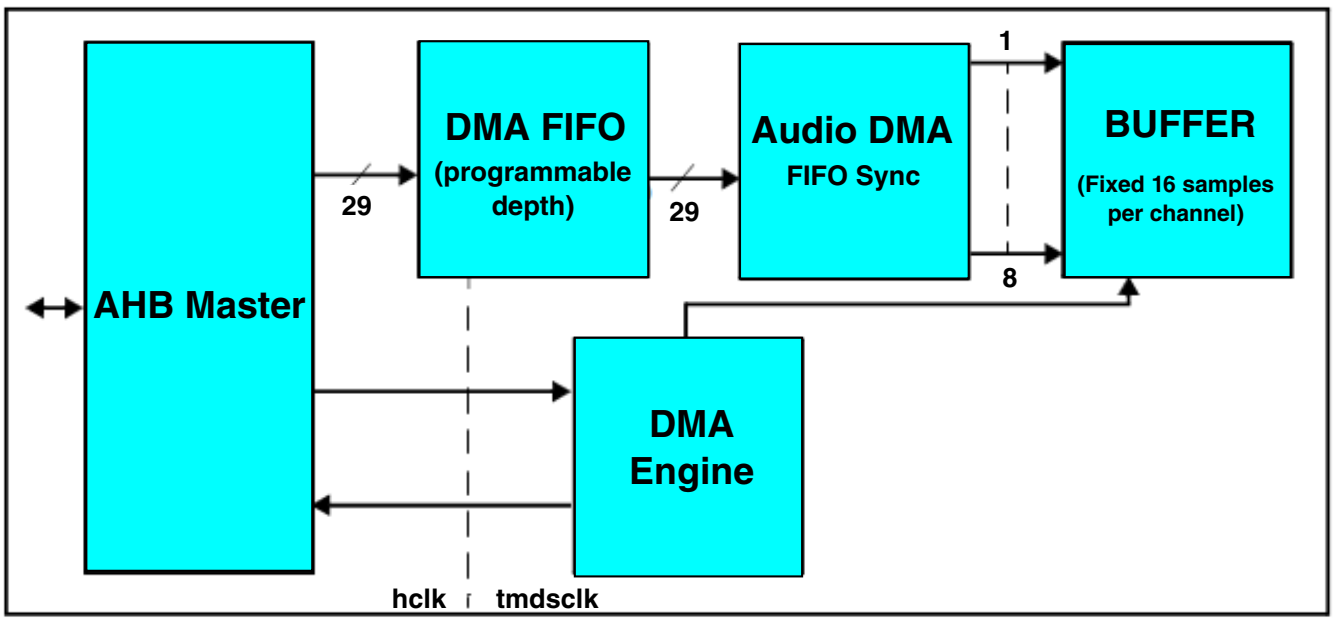


Figure 33-8. Audio DMA Block Diagram

The audio DMA block combines an AHB master interface with a FIFO to perform direct memory access to audio samples stored in a system memory.

The DMA engine is configurable through programmable software registers to perform autonomous burst reading on a configured memory range.

33.4.6.2.1 AHB Master

The AHB master is compliant with the AMBA AHB Specification, Revision 2.0 from ARM.

It has the following features:

- Multi-master capable operation
- 32-bit data transfer
- OKAY, ERROR, RETRY, and SPLIT slave responses
- Rescheduling of burst requirements
- IDLE, NONSEQ, and SEQ transfer types
- Incremental burst modes: unspecified lengths (upper limit is 1 kB boundary) and INCR, INCR4, INCR8, and INCR16 fixed-beat bursts
- Master burst lock mechanism

- Bus access granting
- The following features are not supported:
- Write transaction
- Protection control
- BUSY transfer type
- Wrapping burst

Data Organization in System Memory

The AHB master block fetches the samples from system memory. The Audio Samples are organized according to the channel allocation.

For example, channel 0, 1, 3, 5 are enabled (0 and 1 are always enabled). The Audio Samples must be organized in the system memory like the following:

Table 33-5. Audio Sample Arrangement in System Memory

Position	Sample	Channel
0	n-1	0
1	n-1	1
2	n-1	3
3	n-1	5
4	n-1	0
5	n-1	1
6	n-1	3
7	n-1	5
...	...	...

Table 33-6. Data Arrangement in System Memory for L-PCM (24 bits)

Bit	Description
28	B - IEC B bit
27	P - Parity bit
26	C - Channel Status bit
25	U - User Data bit
24	V - Validity Bit
[23:0]	Audio Sample Data

Table 33-7. Data Arrangement in System Memory for L-PCM (16 bits) and NL-PCM (16 bits)

Bit	Description
28	B - IEC B bit
27	P - Parity bit
26	C - Channel Status bit

Table continues on the next page...

Table 33-7. Data Arrangement in System Memory for L-PCM (16 bits) and NL-PCM (16 bits) (continued)

25	U - User Data bit
24	V - Validity Bit
[23:8]	Audio Sample Data
[7:0]	0x00

33.4.6.2.2 DMA Engine

The DMA engine is responsible for requesting burst transfers to the AHB master, taking into account the FIFO threshold and register settings.

33.4.6.2.2.1 Functional Behavior

The engine:

- Arbitrates read requests to start the burst in the initial address with the size sufficient to fill the FIFO (the size of the FIFO is a parameter in the audio DMA core).
- After this first request, the DMA engine performs subsequent burst requests (incrementing accordingly ohaddr[31:0] and determining correct ohburst[2:0]) towards final_addr[31:0] configured at the register bank and taking into account the AUDIO_FIFO_DEPTH parameter and fifo_threshold[7:0] configuration.
- In the burst mode (INCR4, INCR8, INCR16), the operation stops at the end of the burst.
- Stops operation upon ERROR slave response, signaling ointerror interrupt and staterror signal
- Continues burst transaction:
- Upon RETRY/SPLIT slave response, signaling ointretrysplit interrupt and statretrysplit signal
- Upon losing ownership (no ihgrant) as consequence of arbiter action, signaling ointlostownership interrupt and statlostownership signal
- Decides through register configuration which burst method (unspecified length incrementing or fixed beat incrementing) to use in the read transfers
- Issues ointdone interrupt when it reaches final address reading or is stopped upon user request
- Automatically starts new burst requests until the final_addr[31:0] is reached
- The DMA engine is either stopped by the user or an error/fail condition appears at the slave response.
- Takes into account that an incrementing burst can be of any length (if unspecified INCR type), but upper limit is set because the address must not cross a 1kB boundary

- A maximum theoretical length of a burst is 1024. The burst size must be declared on the mburstlength_addr[10:0].
- Has INCR with an unspecified burst as the default operation burst mode

33.4.6.2.2.2 DMA Operation

Normal operation of the DMA engine is as follows:

1. The enable_hlock, incr_type[1:0], burst_mode, fifo_threshold[7:0], initial_addr[31:0], and final_addr[31:0] are configured according to desired DMA operation.

   NOTE

   Configured values have to follow these rules:

   The number of memory positions (between initial_addr and final_addr) has to be a multiple of theactive audio channels.

   The final_addr[31:0] signal is always bigger than the initial_addr[31:0].

2. To start the audio DMA operation, a '1' is written to data_buffer_ready.
3. The DMA engine starts the operation. The first burst transfer is:

```
        ohaddr[31:0] = initial_addr[31:0];
        ohburst[2:0] = INCR;
        mburstlength[10:0] = ((initial_addr[31:0] + AUDIO_FIFO_DEPTH) <=
final_addr[31:0]) ?
        ((AUDIO_FIFO_DEPTH < 1024) ? AUDIO_FIFO_DEPTH : 1024) : (final_addr[31:0] -
initial_addr[31:0]);
```

4. While DMA is reading samples from the AHB master and writing samples to the Audio FIFO, a datafetch request from the internal frame composer block might happen at the Audio FIFO interface, diminishing the number of samples in the FIFO.
5. When the number of samples in the Audio FIFO is lower than the configured fifo_threshold[7:0], the DMA engine requests a new burst request to the AHB master interface with:

```
ohaddr[15:0] = last address in step 5);

ohburst[2:0] = INCR;

mburstlength[10:0] = AUDIO_FIFO_DEPTH - fifo_threshold[7:0];
```

6. Steps 4) and 5) continue until the final_addr[31:0] is reached.

NOTE

In the last burst request, the DMA engine calculates the mburstlength[10:0] such that the last requested read position is the final_addr[31:0].

7. After completion of the DMA operation, the DMA engine issues the ointdone interrupt signaling end of operation.

Variations of the DMA engine's behavior occur when fixed-beat, incremental bursts are used by INCR4/INCR8/INCR16 burst selects. When this forcing mode is used, you must correctly configure the FIFO's threshold such that the last of the consecutive INCRx (with x = 4, 8, or 16) bursts correctly fill the FIFO at last burst received. Note there is a re-alignment at the 1k boundary.

The following are exceptions to the described DMA behavior:

1. When a user requests end stop_dma_transaction, the DMA engine stops at the end of the current burst operation and signals its completion with an ointdone interrupt.
2. When the AHB slave sends an error response, the DMA engine stops the current operation and signals ointerror interrupt.

Rules for Configuration of Address Registers:

1. Configure the last 2 bits of initial_addr with 0.

```
For example, 32'h0000_0000.
```

2. Configure the last 2 bits of final_addr with non-zero values.

```
For example
32'hxxx_xxx3 or
32'hxxx_xxx7 or
32'hxxx_xxxB or
32'hxxx_xxxF
Where x= any value
```

3. The number of samples is calculated by using the following formula:

```
Number of samples = (final_addr - initial_addr + 1) / 4
Therefore, final_addr = (Number of samples x 4) + initial_addr -1
If a defined length burst is used, align initial_addr, final_addr and
fifo_threshold with the value. If
the burst is not aligned, DMA uses AHB INCR transfers when required.
```

```
Then the number of samples = 100 (a multiple of 5)
```

4. The threshold must be

```
Greater than the selected FIFO DEPTH;

Greater than the number of channels enabled in channel allocation.
```

Due to the limit of audio DMA design, some registers configuration are updated by SDMA (Smart Direct Memory Access). SDMA need to do these items below:

1. clear the audio DMA done request(actually it's an interrupt); set offset 0x00120109 -- bit2.
2. configure next audio DMA start address(offset 0x00123604~0x00123607) and next stop address(offset 0x00123608~0x0012360b); Start address and stop address are provided by S/W and it's variable.
3. Set offset 0x00123601 – bit0 "1" to start DMA;

33.4.6.2.2.3 Transfer Data, Package, and Word

One transfer data can be composed of several transfer packages, and one transfer package can be composed of one or several transfer bursts. One transfer burst can be composed of several transfer words.

The figure below shows the transfer data structure for a fixed-beat, incremental burst.

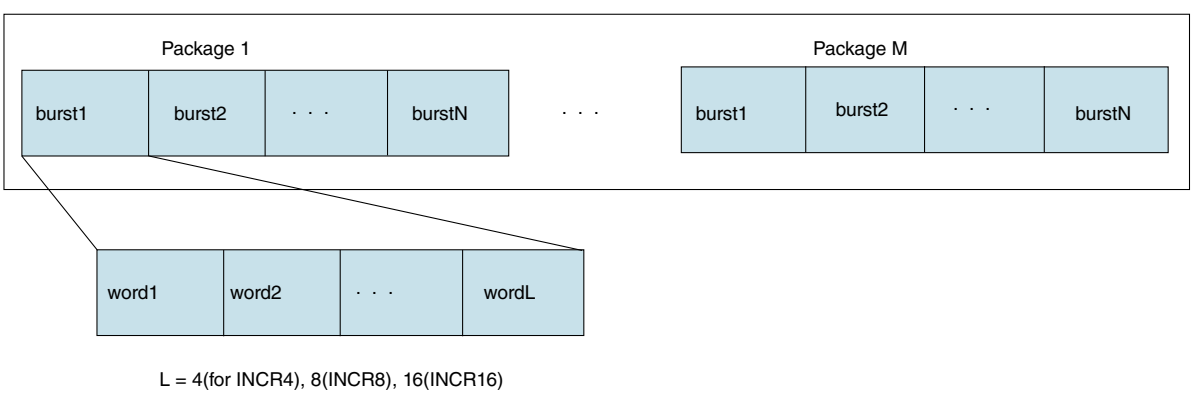


Figure 33-9. Transfer Data Constitution for Fixed-Beat, Incremental Burst

The figure below depicts the transfer data structure for an unspecified burst length.

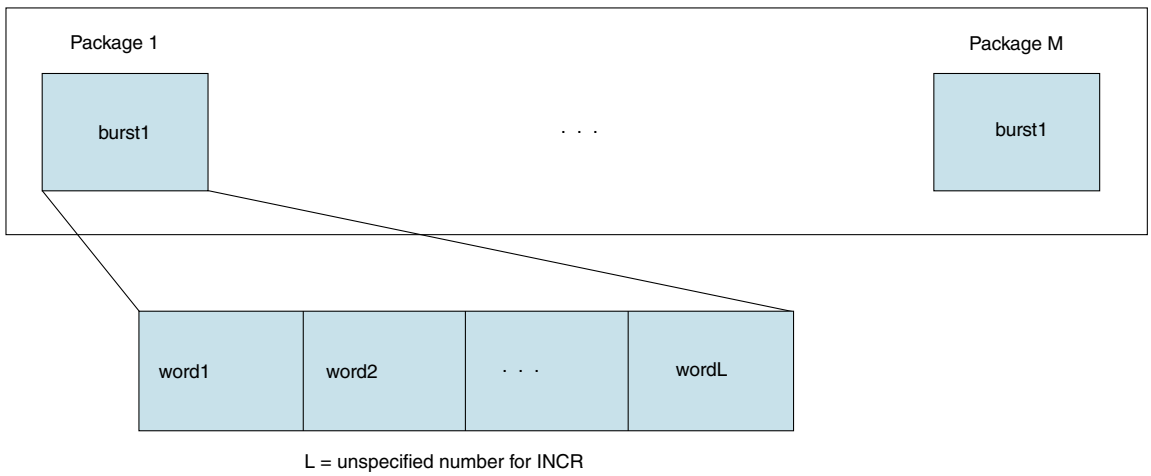


Figure 33-10. Transfer Data Constitution for Unspecified Burst Length

The figure below illustrates the DMA state machine.

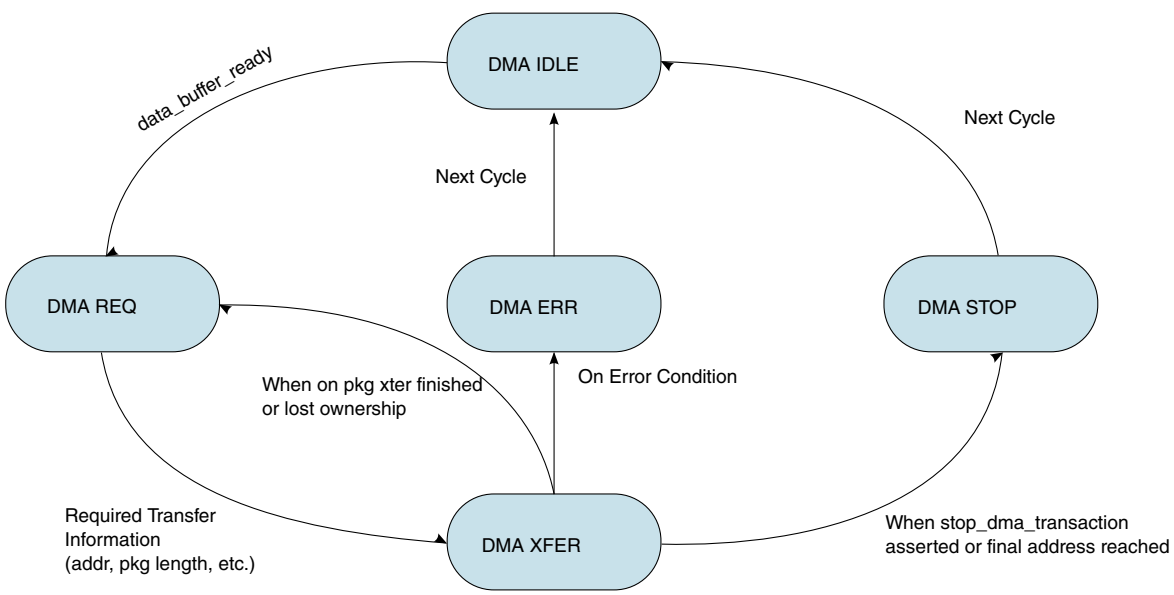


Figure 33-11. DMA FSM Diagram

1. When the operation request is written into the data_buffer_ready, the state switches from IDLE to DMAREQ.
2. When enough transfer information is ready, the state changes to DMAXFER.

3. When one package is completed-indicated by "OnePackOver"-or a lost ownership occurs, the state jumps back to DMAREQ, so the new transaction information can be calculated.
4. When the DMA is ready again, it jumps to DMAXFER.

The previous steps continue until you request an end operation or until the whole operation is completed.

33.4.6.2.3 Audio FIFO

This block contains a FIFO with a configurable depth.

The statthrfiffoempty flag is a version of the FIFO empty, that is active whenever the amount of samples in the FIFO is smaller than four samples. The AHB_DMA_THRSLD and the statthrfifoempty indicators are different. The AHB_DMA_THRSLD defines the occupation of the FIFO, while statthrfifoempty helps the HDMI TX's Frame Composer in the audio packet composition (required when non-linear audio is being packed, in which case four pair of samples are needed to compose one packet).

33.4.6.2.4 FIFO Occupancy/FIFO Almost Empty Flags

The FIFO depth is configured by setting the AUDIO_FIFO_DEPTH parameter in coreConsultant. The FIFO occupancy is calculated from the pointer values.

Also, the most significant pointer bits are used to perform the following calculation:

```
        occupancy = wrptr[n-1:0] - rdptr_previous[n-1:0], if wrptr > rdptr_previous

        occupancy = AUDIO_FIFO_DEPTH + wrptr[n-1:0] - rdptr_previous[n-1:0], if wrptr <
rdptr_previous
```

If occupancy is less than 4, then statthrfifoempty is active.

An example of full/empty flags and FIFO occupancy is shown in the figure below.

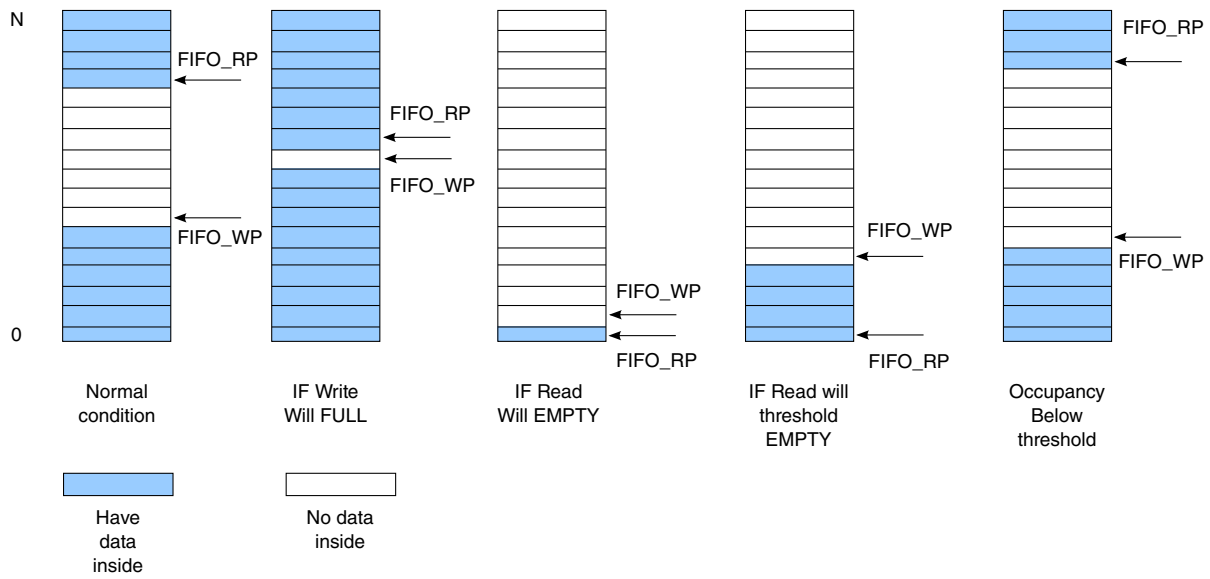


Figure 33-12. Audio FIFO Status Indication

33.4.7 Supported Audio Formats

The HDMI TX has several audio interfaces and each of them has different audio format support capabilities.

Table below represents the audio interface and format dependencies.

Table 33-8. Supported Audio Formats

Audio Input Interface	Audio Format Supported
Audio DMA	Up to eight channels L-PCM/NL-PCM and HBR audio, allowing all audio formats listed to support one single audio DMA interface.

33.4.8 Frame Composer

This block is responsible for assembling video, audio, and data packets in a consistent frame that are streamed to the HDCP cipher and then finally to the HDMI TX PHY.

The HDMI 1.3a standard precisely describes the packet insertion timing and distribution that must be followed to correctly compose an HDMI TMDS (transition minimized differential signaling) stream. In this context, there are data island packets that-when available (ready for insertion in output stream)-have higher priority over others. Two packet descriptor queues are responsible for prioritizing packet insertion.

The higher priority packets are described in Table below. These packets are inserted in the output stream as soon as data to compose them is available (see the HDMI 1.3a standard).

Table 33-9. High Priority Data Island Packets

Packet	Description
Audio Clock Regeneration (ACR)	Indicates to sink device the N/CTS values that should be used in the ACR process
Audio Sample (AUDS)	Transports L-PCM and IEC 61937 compressed audio
General Control (GCP)	Indicates Color Depth, Pixel Packing phase, and AV mute information to sink device

The packets described in tables below can be considered as low priority packets-even though they have precise timing insertion-because their insertion timing is large (for example, one per frame or one per two frames without specific location for some of the packet types and on user request transmit for others).

Table 33-10. Low Priority Data Island Packets

Packet	Description
Audio Content Protection (ACP)	Used to convey content-related information about the active audio stream transmitted
Audio InfoFrame (AUDI)	Indicates characteristics of the active audio stream by using IEC 60958 channel status bits, IEC 61937 burst info, and/or stream data (if present).
Null (NULL)	Ignored by sink devices.
International Standard Recording Code (ISRC1/ISRC2)	See HDMI 1.3a section 5.3.8.
Vendor Specific (VSD) InfoFrame	According to CEA-861-E standard.
AVI infoFrame (AVI)	Video information from source to sink.
Source Data Product Descriptor (SPD) infoFrame	Name and product type of the source device. MPEG (MPEG) Source InfoFrame packets (optional, implementation discouraged by CEA-861-E Section 6.7). Describes several aspects of the compressed video stream that were used to produce the uncompressed video.

The Frame Composer distributes and assembles the data island packets according to the module register bank configuration. The block allows extended control periods to appear with a certain programmed spacing. The Frame Composer uses two packet buffers that allow a packet to be composed while another is being sent to the output HDMI stream.

Packet requests are inserted into the packet queues by a data island flexible scheduler. The HDMI 1.3a specification requires that packet distribution and insertion timing correctly compose an output HDMI TMDS stream. In this context, there are data island packets that are sent on data availability, while others are sent once per frame or once per two frames. and finally others that are sent on user request. Classification of the packets according to this insertion timing is described in the table below.

Table 33-11. Packet Classification

Packet	Classification
Audio Clock Regeneration (ACR)	Sent on data availability.
Audio Sample (AUDS)	Sent on data availability (precede ACR if present).
Audio Content Protection (ACP)	On user request or automatic insertion.
Audio InfoFrame (AUDI)	Once per two frames.
Null (NULL)	On user request or automatic insertion to fill Data Island period.
General Control (GCP)	Once per frame.
International Standard Recording Code (ISRC1/ ISRC2)	On user request.
Vendor Specific (VSD) InfoFrame	On user request or automatic insertion.
AVI infoFrame (AVI)	Once per frame.
Source Data Product Descriptor (SPD) infoFrame	On user request or automatic insertion.

The Data Island Scheduler (DIS) handles packet distribution in the Frame Composer. The DIS is a round- robin (RDRB) state machine that is able to schedule packet insertion on an input video frame or line basis. The DIS is fully configurable and can schedule any packet type to be inserted at a given input video frame rate or input video line rate.

While determining packet distribution on an input video frame or line basis, the DIS schedules the packets to be inserted in the output HDMI stream by inserting the packet descriptor in the corresponding packet priority queue, according to packet priority classification.

After the packet descriptor has been inserted in the packet priority queues, the Data Island Packer (DIP) is responsible for assembling and sequencing the packets for output HDMI stream insertion.

Dedicated ECC generators and checksum byte-wide sum hardware generate the BCH ECC parity codes and infoFrames checksums for all the data islands packets.

The content of GCP, ISRC1/2, VSD, AVI, SPD, and MPEG packets are configured through the registers bank starting at address 0x1000.

33.4.9 HDCP Encryption Engine

HDCP is designed to protect the transmission of audio-visual content between an HDCP Transmitter and an HDCP Receiver.

The system also allows for HDCP Repeaters that support downstream HDCP-protected interface ports. The HDCP system allows up to seven levels of HDCP Repeaters and as many as 128 total HDCP devices, including HDCP Repeaters, to be attached to an HDCP-protected interface port.

NOTE

This feature must be configured and requires a separate license. Contact your Freescale representative for more information on HDCP.

The authentication protocol enables the HDCP Transmitter to verify that a given HDCP Receiver is licensed. With the legitimacy of the Receiver determined, encrypted HDCP content is transmitted between the two based on shared secrets established during authentication. In the event that legitimate devices are compromised to permit unauthorized use of the content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of the content.

The implemented HDCP functionality is compliant with HDCP revision 1.4. The HDCP transmitter implements the three layers of the HDCP cipher, including LFSR and other functions required to generate the encryption key bytes that are then XORed with the data.

33.4.10 EDID/HDCP I^2C E-DDC Interface

The E-DDC channel is a dedicated I2C master interface that allows the read of sink E-EDID based on system needs.

Data read from sink E-EDID can be then transferred to the I2C Master register bank, starting at address 0x7E00.

This block reads the E-EDID (and all its segments according to user configuration) and, after completion, it warns the CPU of data availability. This block also arbitrates the I2C master interface to allow the HDMI TX's HDCP authentication protocol to be performed through this interface. Sink HDCP links (with addresses 0x74 and/or 0x76) should be present at these lines to enable HDCP-compliant behavior.

The interface is shared with the DDC channel of the HDMI controller through multiplexers and is I2C compliant.

33.4.10.1 I²C Master Interface Normal Mode

This operation implements a single read or write operation using the Special Function Register configuration.

The I²C data transfer protocol used is the 7-bit addressed, as defined in Section 9 of the I²C-bus Specification, version 2.1.

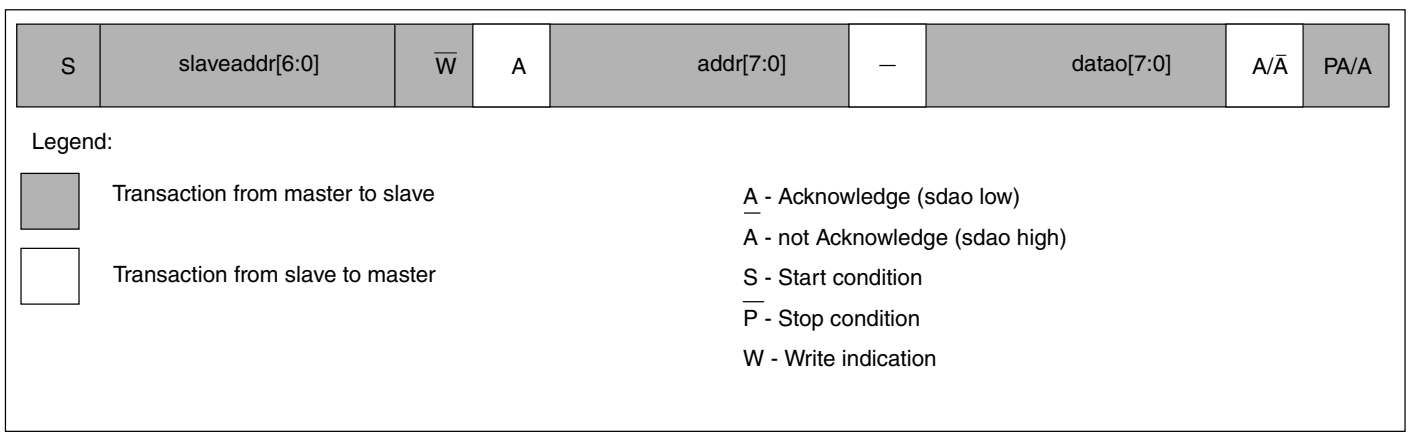


Figure 33-13. Data Write Transaction

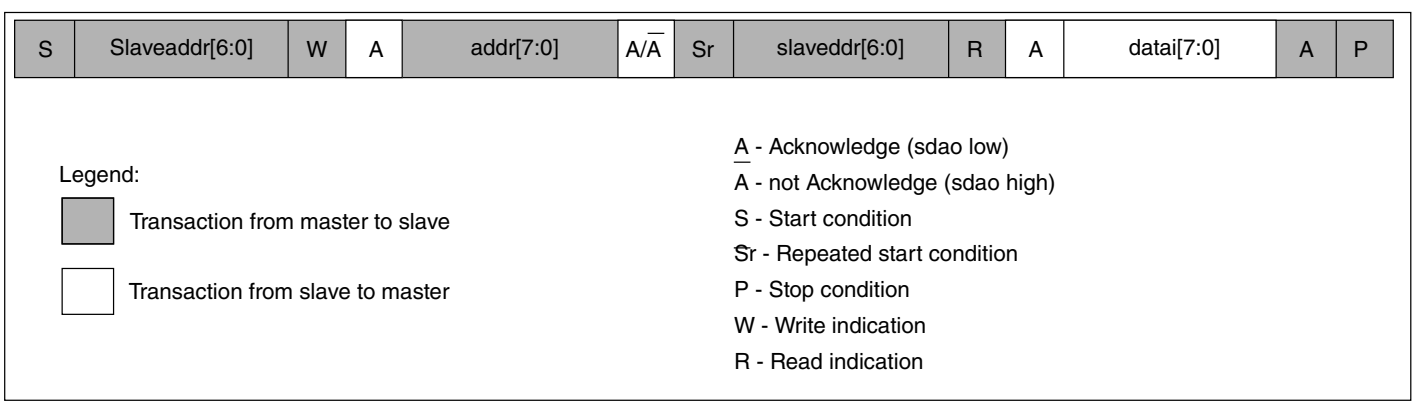


Figure 33-14. Data Read Transaction

33.4.10.2 I²C Master Interface Extended Read Mode

This I2C extended read mode operation implements a segment pointer-based read operation using the Special Register configuration.

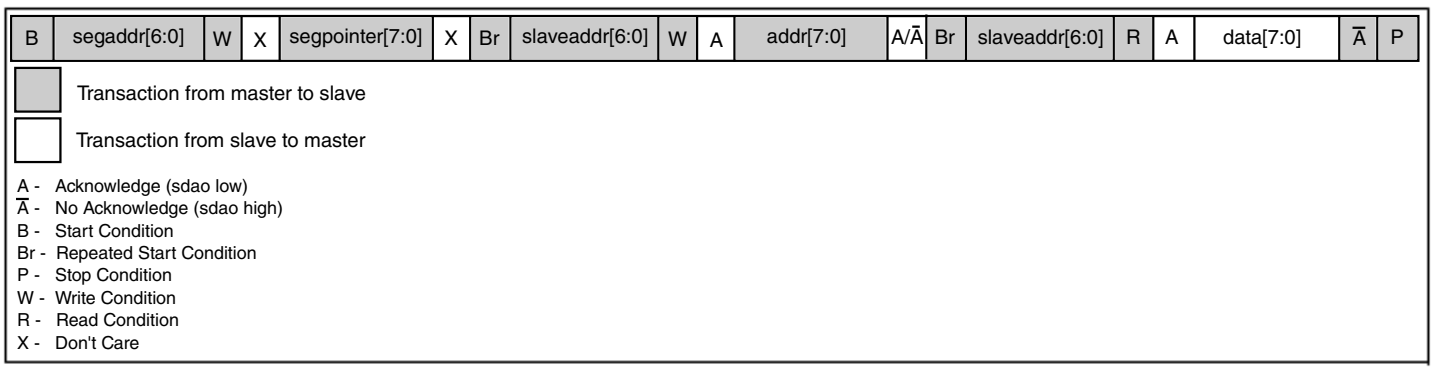


Figure 33-15. Extended Data Read Operation

33.4.11 System Configuration Interfaces

The internal register set is distributed with the HDMI TX. The system interface (the interface that connects to the processor bus) bridges these registers using a simple standard interface.

The system interface is AMBA AHB,.

The AHB bridges the bus to the internal SFR bus.

33.4.11.1 AMBA AHB Slave Interface

The AMBA AHB slave interface is compatible with the AMBA Specification, revision 2.0.

The AHB slave interface is used for register configuration implementing only a simple transaction mode and single master slave operation.

The HDMI TX does not support protection control, burst transfers, or split transactions.

33.4.12 CEC Hardware Engine

Consumer Electronics Control (CEC) is a protocol that provides high-level control functions between all of the various audiovisual products in a user's environment.

It is an optional feature in the HDMI 1.3a Specification. It uses only one bidirectional line for transmission and reception.

All transactions on the CEC line consist of an initiator and one or more followers. The initiator is responsible for sending the message structure and the data. The follower is the recipient of any data and is responsible for setting any acknowledgement bits.

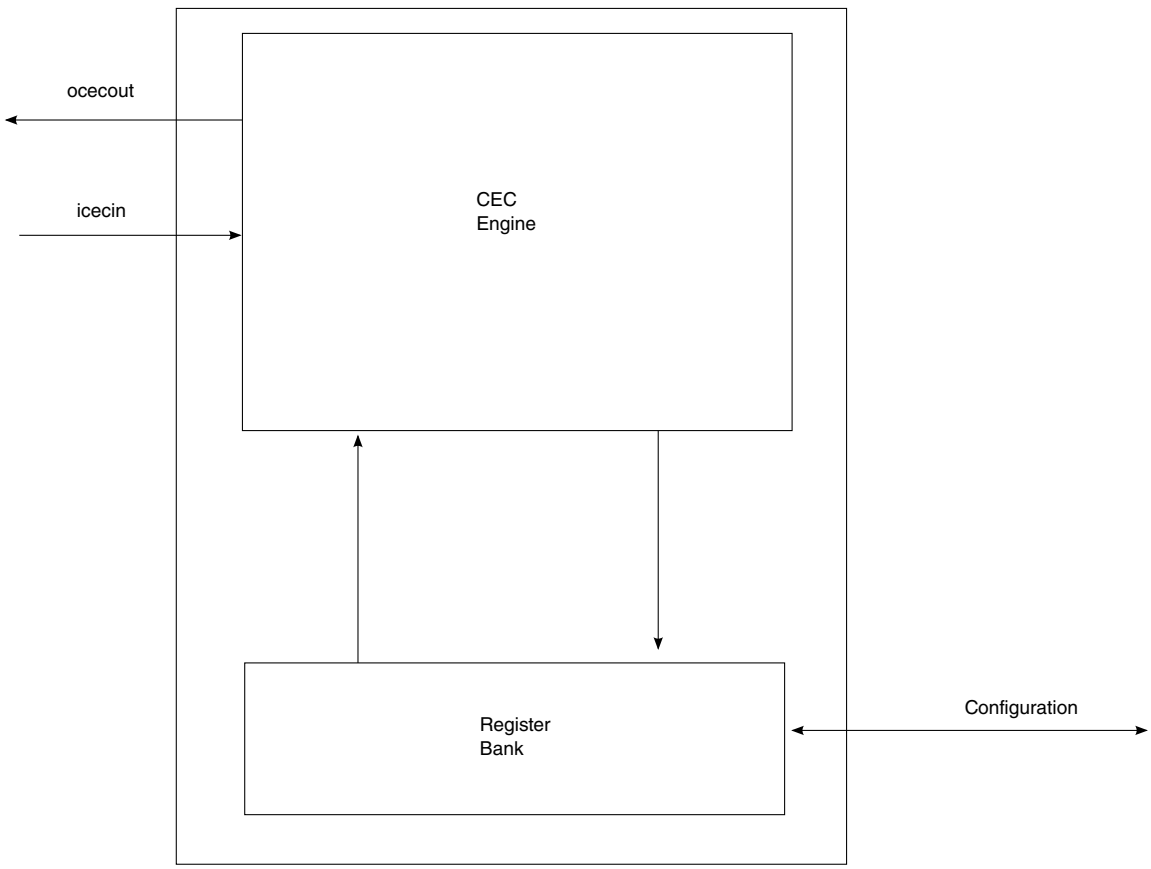


Figure 33-16. CEC Engine Simplified Block Diagram

There are two operation modes for a CEC controller.

- Initiator Mode
- In this mode, the CEC controller sends messages out and waits for a follower to feedback. The CEC controller works in this mode when it starts to send a frame. After the transmission is done, it automatically returns to the follower mode (no software control involved).

- Follower Mode
- In this mode, the CEC controller receives messages and feeds back the initiator with appropriate signals. The CEC controller always works in the follower mode whenever it is not transmitting any data.

For correct CEC controller interface operation, initial reset is required in order to set internal registers to a known state. After this reset, the interface is in an IDLE state, waiting for a read or write request coming from the register configuration.

A specific CEC API is provided that implements all necessary low-level register configuration to send and receive CEC messages. For more information, see the CEC API documentation.

The CEC engine registers base address is 0x7D00. For more information about these registers, see HDMI Memory Map/Register Definition.

For more information about CEC, see *Consumer Electronics Control (CEC) Application Note*.

33.5 HDMI Memory Map/Register Definition

All registers are addressable on 32-bit boundaries; each unused bit or address location is reserved for future use and read back as 0.

.

HDMI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_0000	Design Identification Register (HDMI_DESIGN_ID)	8	R	00h	33.5.1/1593
12_0001	Revision Identification Register (HDMI_REVISION_ID)	8	R	00h	33.5.2/1594
12_0002	Product Identification Register 0 (HDMI_PRODUCT_ID0)	8	R	00h	33.5.3/1594
12_0003	Product Identification Register 1 (HDMI_PRODUCT_ID1)	8	R	00h	33.5.4/1595
12_0004	Configuration Identification Register 0 (HDMI_CONFIG0_ID)	8	R	00h	33.5.5/1595
12_0005	Configuration Identification Register 1 (HDMI_CONFIG1_ID)	8	R	00h	33.5.6/1596
12_0006	Configuration Identification Register 2 (HDMI_CONFIG2_ID)	8	R	00h	33.5.7/1597
12_0007	Configuration Identification Register 3 (HDMI_CONFIG3_ID)	8	R	00h	33.5.8/1598
12_0100	Frame Composer Interrupt Status Register 0 (HDMI_IH_FC_STAT0)	8	w1c	00h	33.5.9/1598

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_0101	Frame Composer Interrupt Status Register 1 (HDMI_IH_FC_STAT1)	8	w1c	00h	33.5.10/ 1599
12_0102	Frame Composer Interrupt Status Register 2 (HDMI_IH_FC_STAT2)	8	w1c	00h	33.5.11/ 1600
12_0103	Audio Sampler Interrupt Status Register (HDMI_IH_AS_STAT0)	8	w1c	00h	33.5.12/ 1601
12_0104	PHY Interface Interrupt Status Register (HDMI_IH_PHY_STAT0)	8	w1c	00h	33.5.13/ 1602
12_0105	E-DDC I2C Master Interrupt Status Register (HDMI_IH_I2CM_STAT0)	8	w1c	00h	33.5.14/ 1603
12_0106	CEC Interrupt Status Register (HDMI_IH_CEC_STAT0)	8	w1c	00h	33.5.15/ 1604
12_0107	Video Packetizer Interrupt Status Register (HDMI_IH_VP_STAT0)	8	w1c	00h	33.5.16/ 1605
12_0108	PHY GEN2 I2C Master Interrupt Status Register (HDMI_IH_I2CMPHY_STAT0)	8	w1c	00h	33.5.17/ 1606
12_0109	AHB Audio DMA Interrupt Status Register (HDMI_IH_AHBDMAAUD_STAT0)	8	w1c	00h	33.5.18/ 1606
12_0180	Frame Composer Interrupt Mute Control Register 0 (HDMI_IH_MUTE_FC_STAT0)	8	R/W	00h	33.5.19/ 1608
12_0181	Frame Composer Interrupt Mute Control Register 1 (HDMI_IH_MUTE_FC_STAT1)	8	R/W	00h	33.5.20/ 1609
12_0182	Frame Composer Interrupt Mute Control Register 2 (HDMI_IH_MUTE_FC_STAT2)	8	R/W	00h	33.5.21/ 1610
12_0183	Audio Sampler Interrupt Mute Control Register 0 (HDMI_IH_MUTE_AS_STAT0)	8	R/W	00h	33.5.22/ 1610
12_0184	PHY Interface Interrupt Mute Control Register (HDMI_IH_MUTE_PHY_STAT0)	8	R/W	00h	33.5.23/ 1611
12_0185	E-DDC I2C Master Interrupt Mute Control Register (HDMI_IH_MUTE_I2CM_STAT0)	8	R/W	00h	33.5.24/ 1612
12_0186	CEC Interrupt Mute Control Register (HDMI_IH_MUTE_CEC_STAT0)	8	R/W	00h	33.5.25/ 1612
12_0187	Video Packetizer Interrupt Mute Control Register (HDMI_IH_MUTE_VP_STAT0)	8	R/W	00h	33.5.26/ 1613
12_0188	PHY GEN 2 I2C Master Interrupt Mute Control Register (HDMI_IH_MUTE_I2CMPHY_STAT0)	8	R/W	00h	33.5.27/ 1614
12_0189	AHB Audio DMA Interrupt Mute Control Register (HDMI_IH_MUTE_AHBDMAAUD_STAT0)	8	R/W	00h	33.5.28/ 1615
12_01FF	Global Interrupt Mute Control Register (HDMI_IH_MUTE)	8	R/W	03h	33.5.29/ 1616
12_0200	Video Input Mapping and Internal Data Enable Configuration Register (HDMI_TX_INVID0)	8	R/W	01h	33.5.30/ 1616
12_0201	Video Input Stuffing Enable Register (HDMI_TX_INSTUFFING)	8	R/W	00h	33.5.31/ 1617

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_0202	Video Input GY Data Channel Stuffing Register 0 (HDMI_TX_GYDATA0)	8	R/W	00h	33.5.32/ 1618
12_0203	Video Input GY Data Channel Stuffing Register 1 (HDMI_TX_GYDATA1)	8	R/W	00h	33.5.33/ 1619
12_0204	Video Input RCR Data Channel Stuffing Register 0 (HDMI_TX_RCRDATA0)	8	R/W	00h	33.5.34/ 1619
12_0205	Video Input RCR Data Channel Stuffing Register 1 (HDMI_TX_RCRDATA1)	8	R/W	00h	33.5.35/ 1620
12_0206	Video Input RCB Data Channel Stuffing Register 0 (HDMI_TX_BCBDATA0)	8	R/W	00h	33.5.36/ 1620
12_0207	Video Input RCB Data Channel Stuffing Register 1 (HDMI_TX_BCBDATA1)	8	R/W	00h	33.5.37/ 1621
12_0800	Video Packetizer Packing Phase Status Register (HDMI_VP_STATUS)	8	R	00h	33.5.38/ 1621
12_0801	Video Packetizer Pixel Repetition and Color Depth Register (HDMI_VP_PR_CD)	8	R/W	00h	33.5.39/ 1622
12_0802	Video Packetizer Stuffing and Default Packing Phase Register (HDMI_VP_STUFF)	8	R/W	00h	33.5.40/ 1623
12_0803	Video Packetizer YCC422 Remapping Register (HDMI_VP_REMAP)	8	R/W	00h	33.5.41/ 1624
12_0804	Video Packetizer Output, Bypass, and Enable Configuration Register (HDMI_VP_CONF)	8	R/W	46h	33.5.42/ 1625
12_0805	VP_STAT (HDMI_VP_STAT)	8	R	00h	33.5.43/ 1625
12_0806	VP_INT (HDMI_VP_INT)	8	R	00h	33.5.44/ 1626
12_0807	Video Packetizer Interrupt Mask Register (HDMI_VP_MASK)	8	R/W	00h	33.5.45/ 1627
12_0808	VP_POL (HDMI_VP_POL)	8	R/W	FFh	33.5.46/ 1628
12_1000	Frame Composer Input Video Configuration and HDCP Keepout Register (HDMI_FC_INVIDCONF)	8	R/W	70h	33.5.47/ 1629
12_1001	Frame Composer Input Video HActive Pixels Register 0 (HDMI_FC_INHACTIV0)	8	R/W	00h	33.5.48/ 1630
12_1002	Frame Composer Input Video HActive Pixels Register 1 (HDMI_FC_INHACTIV1)	8	R/W	00h	33.5.49/ 1631
12_1003	Frame Composer Input Video HBlank Pixels Register 0 (HDMI_FC_INHBLANK0)	8	R/W	00h	33.5.50/ 1631
12_1004	Frame Composer Input Video HBlank Pixels Register 1 (HDMI_FC_INHBLANK1)	8	R/W	00h	33.5.51/ 1632
12_1005	Frame Composer Input Video VActive Pixels Register 0 (HDMI_FC_INVACTIV0)	8	R/W	00h	33.5.52/ 1633
12_1006	Frame Composer Input Video VActive Pixels Register 1 (HDMI_FC_INVACTIV1)	8	R/W	00h	33.5.53/ 1633

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_1007	Frame Composer Input Video VBlank Pixels Register (HDMI_FC_INVBLANK)	8	R/W	00h	33.5.54/ 1634
12_1008	Frame Composer Input Video HSync Front Porch Register 0 (HDMI_FC_HSYNCINDELAY0)	8	R/W	00h	33.5.55/ 1634
12_1009	Frame Composer Input Video HSync Front Porch Register 1 (HDMI_FC_HSYNCINDELAY1)	8	R/W	00h	33.5.56/ 1635
12_100A	Frame Composer Input Video HSync Width Register 0 (HDMI_FC_HSYNCINWIDTH0)	8	R/W	00h	33.5.57/ 1636
12_100B	Frame Composer Input Video HSync Width Register 1 (HDMI_FC_HSYNCINWIDTH1)	8	R/W	00h	33.5.58/ 1636
12_100C	Frame Composer Input Video VSync Front Porch Register (HDMI_FC_VSYNCINDELAY)	8	R/W	00h	33.5.59/ 1637
12_100D	Frame Composer Input Video VSync Width Register (HDMI_FC_VSYNCINWIDTH)	8	R/W	00h	33.5.60/ 1637
12_100E	Frame Composer Input Video Refresh Rate Register 0 (HDMI_FC_INFREQ0)	8	R/W	00h	33.5.61/ 1638
12_100F	Frame Composer Input Video Refresh Rate Register 1 (HDMI_FC_INFREQ1)	8	R/W	00h	33.5.62/ 1638
12_1010	Frame Composer Input Video Refresh Rate Register 2 (HDMI_FC_INFREQ2)	8	R/W	00h	33.5.63/ 1639
12_1011	Frame Composer Control Period Duration Register (HDMI_FC_CTRLDUR)	8	R/W	00h	33.5.64/ 1640
12_1012	Frame Composer Extended Control Period Duration Register (HDMI_FC_EXCTRLDUR)	8	R/W	00h	33.5.65/ 1640
12_1013	Frame Composer Extended Control Period Maximum Spacing Register (HDMI_FC_EXCTRLSPAC)	8	R/W	00h	33.5.66/ 1641
12_1014	Frame Composer Channel 0 Non-Preamble Data Register (HDMI_FC_CH0PREAM)	8	R/W	00h	33.5.67/ 1641
12_1015	Frame Composer Channel 1 Non-Preamble Data Register (HDMI_FC_CH1PREAM)	8	R/W	00h	33.5.68/ 1642
12_1016	Frame Composer Channel 2 Non-Preamble Data Register (HDMI_FC_CH2PREAM)	8	R/W	00h	33.5.69/ 1642
12_1017	Frame Composer AVI Configuration Register 3 (HDMI_FC_AVICONF3)	8	R/W	00h	33.5.70/ 1643
12_1018	Frame Composer GCP Packet Configuration Register (HDMI_FC_GCP)	8	R/W	00h	33.5.71/ 1643
12_1019	Frame Composer AVI Packet Configuration Register 0 (HDMI_FC_AVICONF0)	8	R/W	00h	33.5.72/ 1644
12_101A	Frame Composer AVI Packet Configuration Register 1 (HDMI_FC_AVICONF1)	8	R/W	00h	33.5.73/ 1645
12_101B	FC_AVICONFFrame Composer AVI Packet Configuration Register 2 (HDMI_FC_AVICONF2)	8	R/W	00h	33.5.74/ 1646
12_101C	Frame Composer AVI Packet VIC Register (HDMI_FC_AVIVID)	8	R/W	00h	33.5.75/ 1647

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_101D	Frame Composer AVI Packet End of Top Bar Register 0 (HDMI_FC_AVIETB0)	8	R/W	00h	33.5.76/ 1647
12_101E	Frame Composer AVI Packet End of Top Bar Register 1 (HDMI_FC_AVIETB1)	8	R/W	00h	33.5.77/ 1648
12_101F	Frame Composer AVI Packet Start of Bottom Bar Register 0 (HDMI_FC_AVISBB0)	8	R/W	00h	33.5.78/ 1648
12_1020	Frame Composer AVI Packet Start of Bottom Bar Register 1 (HDMI_FC_AVISBB1)	8	R/W	00h	33.5.79/ 1649
12_1021	Frame Composer AVI Packet End of Left Bar Register 0 (HDMI_FC_AVIELB0)	8	R/W	00h	33.5.80/ 1649
12_1022	Frame Composer AVI Packet End of Left Bar Register 1 (HDMI_FC_AVIELB1)	8	R/W	00h	33.5.81/ 1650
12_1023	Frame Composer AVI Packet Start of Right Bar Register 0 (HDMI_FC_AVISRB0)	8	R/W	00h	33.5.82/ 1650
12_1024	Frame Composer AVI Packet Start of Right Bar Register 1 (HDMI_FC_AVISRB1)	8	R/W	00h	33.5.83/ 1651
12_1025	Frame Composer AUD Packet Configuration Register 0 (HDMI_FC_AUDICONF0)	8	R/W	00h	33.5.84/ 1651
12_1026	Frame Composer AUD Packet Configuration Register 1 (HDMI_FC_AUDICONF1)	8	R/W	00h	33.5.85/ 1652
12_1027	Frame Composer AUD Packet Configuration Register 2 (HDMI_FC_AUDICONF2)	8	R/W	00h	33.5.86/ 1652
12_1028	Frame Composer AUD Packet Configuration Register 3 (HDMI_FC_AUDICONF3)	8	R/W	00h	33.5.87/ 1653
12_1029	Frame Composer VSI Packet Data IEEE Register 0 (HDMI_FC_VSDIEEEID0)	8	R/W	00h	33.5.88/ 1653
12_102A	Frame Composer VSI Packet Data Size Register (HDMI_FC_VSDSIZE)	8	R/W	1Bh	33.5.89/ 1654
12_1030	Frame Composer VSI Packet Data IEEE Register 1 (HDMI_FC_VSDIEEEID1)	8	R/W	00h	33.5.90/ 1654
12_1031	Frame Composer VSI Packet Data IEEE Register 2 (HDMI_FC_VSDIEEEID2)	8	R/W	00h	33.5.91/ 1655
12_1032	Frame Composer VSI Packet Data IEEE Register 0 (HDMI_FC_VSDPAYLOAD0)	8	R/W	00h	33.5.92/ 1655
12_1033	Frame Composer VSI Packet Data IEEE Register 1 (HDMI_FC_VSDPAYLOAD1)	8	R/W	00h	33.5.93/ 1656
12_1034	Frame Composer VSI Packet Data IEEE Register 2 (HDMI_FC_VSDPAYLOAD2)	8	R/W	00h	33.5.94/ 1656
12_1035	Frame Composer VSI Packet Data IEEE Register 3 (HDMI_FC_VSDPAYLOAD3)	8	R/W	00h	33.5.95/ 1657
12_1036	Frame Composer VSI Packet Data IEEE Register 4 (HDMI_FC_VSDPAYLOAD4)	8	R/W	00h	33.5.96/ 1657
12_1037	Frame Composer VSI Packet Data IEEE Register 5 (HDMI_FC_VSDPAYLOAD5)	8	R/W	00h	33.5.97/ 1658

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_1038	Frame Composer VSI Packet Data IEEE Register 6 (HDMI_FC_VSDPAYLOAD6)	8	R/W	00h	33.5.98/ 1658
12_1039	Frame Composer VSI Packet Data IEEE Register 7 (HDMI_FC_VSDPAYLOAD7)	8	R/W	00h	33.5.99/ 1659
12_103A	Frame Composer VSI Packet Data IEEE Register 8 (HDMI_FC_VSDPAYLOAD8)	8	R/W	00h	33.5.100/ 1659
12_103B	Frame Composer VSI Packet Data IEEE Register 9 (HDMI_FC_VSDPAYLOAD9)	8	R/W	00h	33.5.101/ 1660
12_103C	Frame Composer VSI Packet Data IEEE Register 10 (HDMI_FC_VSDPAYLOAD10)	8	R/W	00h	33.5.102/ 1660
12_103D	Frame Composer VSI Packet Data IEEE Register 11 (HDMI_FC_VSDPAYLOAD11)	8	R/W	00h	33.5.103/ 1661
12_103E	Frame Composer VSI Packet Data IEEE Register 12 (HDMI_FC_VSDPAYLOAD12)	8	R/W	00h	33.5.104/ 1661
12_103F	Frame Composer VSI Packet Data IEEE Register 13 (HDMI_FC_VSDPAYLOAD13)	8	R/W	00h	33.5.105/ 1662
12_1040	Frame Composer VSI Packet Data IEEE Register 14 (HDMI_FC_VSDPAYLOAD14)	8	R/W	00h	33.5.106/ 1662
12_1041	Frame Composer VSI Packet Data IEEE Register 15 (HDMI_FC_VSDPAYLOAD15)	8	R/W	00h	33.5.107/ 1663
12_1042	Frame Composer VSI Packet Data IEEE Register 16 (HDMI_FC_VSDPAYLOAD16)	8	R/W	00h	33.5.108/ 1663
12_1043	Frame Composer VSI Packet Data IEEE Register 17 (HDMI_FC_VSDPAYLOAD17)	8	R/W	00h	33.5.109/ 1664
12_1044	Frame Composer VSI Packet Data IEEE Register 18 (HDMI_FC_VSDPAYLOAD18)	8	R/W	00h	33.5.110/ 1664
12_1045	Frame Composer VSI Packet Data IEEE Register 19 (HDMI_FC_VSDPAYLOAD19)	8	R/W	00h	33.5.111/ 1665
12_1046	Frame Composer VSI Packet Data IEEE Register 20 (HDMI_FC_VSDPAYLOAD20)	8	R/W	00h	33.5.112/ 1665
12_1047	Frame Composer VSI Packet Data IEEE Register 21 (HDMI_FC_VSDPAYLOAD21)	8	R/W	00h	33.5.113/ 1666
12_1048	Frame Composer VSI Packet Data IEEE Register 22 (HDMI_FC_VSDPAYLOAD22)	8	R/W	00h	33.5.114/ 1666
12_1049	Frame Composer VSI Packet Data IEEE Register 23 (HDMI_FC_VSDPAYLOAD23)	8	R/W	00h	33.5.115/ 1667
12_104A	Frame Composer SPD Packet Data Vendor Name Register 0 (HDMI_FC_SPDVENDORNAME0)	8	R/W	00h	33.5.116/ 1667
12_1052	Frame Composer SPD Packet Data Product Name Register 0 (HDMI_FC_SPDPRODUCTNAME0)	8	R/W	00h	33.5.117/ 1668
12_1062	Frame Composer SPD Packet Data Source Product Descriptor Register (HDMI_FC_SPDDEVICEINF)	8	R/W	00h	33.5.118/ 1668
12_1063	Frame Composer Audio Sample Flat and Layout Configuration Register (HDMI_FC_AUDSCONF)	8	R/W	00h	33.5.119/ 1669

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_1064	Frame Composer Audio Packet Sample Present Status Register (HDMI_FC_AUDSSTAT)	8	R	00h	33.5.120/ 1669
12_1073	Frame Composer Number of High Priority Packets Attended Configuration Register (HDMI_FC_CTRLQHIGH)	8	R/W	0Fh	33.5.121/ 1670
12_1074	Frame Composer Number of Low Priority Packets Attended Configuration Register (HDMI_FC_CTRLQLOW)	8	R/W	03h	33.5.122/ 1671
12_1075	Frame Composer ACP Packet Type Configuration Register 0 (HDMI_FC_ACP0)	8	R/W	00h	33.5.123/ 1671
12_1091	Frame Composer ACP Packet Type Configuration Register 1 (HDMI_FC_ACP1)	8	R/W	00h	33.5.124/ 1672
12_1092	FC_ISCR1_Frame Composer Packet Status, Valid, and Continue Configuration Register (HDMI_FC_ISCR1_0)	8	R/W	00h	33.5.125/ 1672
12_1093	Frame Composer ISCR1 Packet Body Register 1 (HDMI_FC_ISCR1_1)	8	R/W	00h	33.5.126/ 1673
12_10A3	Frame Composer ISCR2 Packet Body Register 0 (HDMI_FC_ISCR2_0)	8	R/W	00h	33.5.127/ 1673
12_10B3	Frame Composer Data Island Auto Packet Scheduling Register 0 (HDMI_FC_DATAUTO0)	8	R/W	00h	33.5.128/ 1674
12_10B4	Frame Composer Data Island Auto Packet Scheduling Register 1 (HDMI_FC_DATAUTO1)	8	R/W	00h	33.5.129/ 1675
12_10B5	Frame Composer Data Island Auto Packet Scheduling Register 2 (HDMI_FC_DATAUTO2)	8	R/W	00h	33.5.130/ 1675
12_10B6	Frame Composer Data Island Manual Packet Request Register (HDMI_FC_DATMAN)	8	W	00h	33.5.131/ 1676
12_10B7	Frame Composer Data Island Auto Packet Scheduling Register 3 (HDMI_FC_DATAUTO3)	8	R/W	0Fh	33.5.132/ 1677
12_10B8	Frame Composer Round Robin ACR Packet Insertion Register 0 (HDMI_FC_RDRB0)	8	R/W	00h	33.5.133/ 1678
12_10B9	Frame Composer Round Robin ACR Packet Insertion Register 1 (HDMI_FC_RDRB1)	8	R/W	00h	33.5.134/ 1678
12_10BA	Frame Composer Round Robin ACR Packet Insertion Register 2 (HDMI_FC_RDRB2)	8	R/W	00h	33.5.135/ 1679
12_10BB	Frame Composer Round Robin ACR Packet Insertion Register 3 (HDMI_FC_RDRB3)	8	R/W	00h	33.5.136/ 1679
12_10BC	Frame Composer Round Robin ACR Packet Insertion Register 4 (HDMI_FC_RDRB4)	8	R/W	00h	33.5.137/ 1680
12_10BD	Frame Composer Round Robin ACR Packet Insertion Register 5 (HDMI_FC_RDRB5)	8	R/W	00h	33.5.138/ 1680
12_10BE	Frame Composer Round Robin ACR Packet Insertion Register 6 (HDMI_FC_RDRB6)	8	R/W	00h	33.5.139/ 1681
12_10BF	Frame Composer Round Robin ACR Packet Insertion Register 7 (HDMI_FC_RDRB7)	8	R/W	00h	33.5.140/ 1682
12_10D0	FC_STAT0 (HDMI_FC_STAT0)	8	R	00h	33.5.141/ 1682

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_10D1	FC_INT0 (HDMI_FC_INT0)	8	R/W	00h	33.5.142/ 1683
12_10D2	Frame Composer Packet Interrupt Mask Register 0 (HDMI_FC_MASK0)	8	R/W	25h	33.5.143/ 1684
12_10D3	FC_POL0 (HDMI_FC_POL0)	8	R/W	FFh	33.5.144/ 1685
12_10D4	FC_STAT1 (HDMI_FC_STAT1)	8	R/W	00h	33.5.145/ 1686
12_10D5	FC_INT1 (HDMI_FC_INT1)	8	R/W	00h	33.5.146/ 1686
12_10D6	Frame Composer Packet Interrupt Mask Register 1 (HDMI_FC_MASK1)	8	R/W	00h	33.5.147/ 1687
12_10D7	FC_POL1 (HDMI_FC_POL1)	8	R/W	FFh	33.5.148/ 1688
12_10D8	FC_STAT2 (HDMI_FC_STAT2)	8	R/W	00h	33.5.149/ 1689
12_10D9	FC_INT2 (HDMI_FC_INT2)	8	R/W	00h	33.5.150/ 1690
12_10DA	Frame Composer High/Low Priority Overflow Interrupt Mask Register 2 (HDMI_FC_MASK2)	8	R/W	00h	33.5.151/ 1690
12_10DB	FC_POL2 (HDMI_FC_POL2)	8	R/W	03h	33.5.152/ 1691
12_10E0	Frame Composer Pixel Repetition Configuration Register (HDMI_FC_PRCONF)	8	R/W	10h	33.5.153/ 1692
12_1100	Frame Composer GMD Packet Status Register (HDMI_FC_GMD_STAT)	8	R	00h	33.5.154/ 1693
12_1101	Frame Composer GMD Packet Enable Register (HDMI_FC_GMD_EN)	8	R/W	00h	33.5.155/ 1694
12_1102	Frame Composer GMD Packet Update Register (HDMI_FC_GMD_UP)	8	W	00h	33.5.156/ 1694
12_1103	Frame Composer GMD Packet Schedule Configuration Register (HDMI_FC_GMD_CONF)	8	R/W	10h	33.5.157/ 1695
12_1104	Frame Composer GMD Packet Profile and Gamut Sequence Configuration Register (HDMI_FC_GMD_HB)	8	R/W	00h	33.5.158/ 1696
12_1105	Frame Composer GMD Packet Body Register 0 (HDMI_FC_GMD_PB0)	8	R/W	00h	33.5.159/ 1696
12_1106	Frame Composer GMD Packet Body Register 1 (HDMI_FC_GMD_PB1)	8	R/W	00h	33.5.160/ 1697
12_1107	Frame Composer GMD Packet Body Register 2 (HDMI_FC_GMD_PB2)	8	R/W	00h	33.5.161/ 1697
12_1108	Frame Composer GMD Packet Body Register 3 (HDMI_FC_GMD_PB3)	8	R/W	00h	33.5.162/ 1698
12_1109	Frame Composer GMD Packet Body Register 4 (HDMI_FC_GMD_PB4)	8	R/W	00h	33.5.163/ 1698

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_110A	Frame Composer GMD Packet Body Register 5 (HDMI_FC_GMD_PB5)	8	R/W	00h	33.5.164/ 1699
12_110B	Frame Composer GMD Packet Body Register 6 (HDMI_FC_GMD_PB6)	8	R/W	00h	33.5.165/ 1699
12_110C	Frame Composer GMD Packet Body Register 7 (HDMI_FC_GMD_PB7)	8	R/W	00h	33.5.166/ 1700
12_110D	Frame Composer GMD Packet Body Register 8 (HDMI_FC_GMD_PB8)	8	R/W	00h	33.5.167/ 1700
12_110E	Frame Composer GMD Packet Body Register 9 (HDMI_FC_GMD_PB9)	8	R/W	00h	33.5.168/ 1701
12_110F	Frame Composer GMD Packet Body Register 10 (HDMI_FC_GMD_PB10)	8	R/W	00h	33.5.169/ 1701
12_1110	Frame Composer GMD Packet Body Register 11 (HDMI_FC_GMD_PB11)	8	R/W	00h	33.5.170/ 1702
12_1111	Frame Composer GMD Packet Body Register 12 (HDMI_FC_GMD_PB12)	8	R/W	00h	33.5.171/ 1702
12_1112	Frame Composer GMD Packet Body Register 13 (HDMI_FC_GMD_PB13)	8	R/W	00h	33.5.172/ 1703
12_1113	Frame Composer GMD Packet Body Register 14 (HDMI_FC_GMD_PB14)	8	R/W	00h	33.5.173/ 1703
12_1114	Frame Composer GMD Packet Body Register 15 (HDMI_FC_GMD_PB15)	8	R/W	00h	33.5.174/ 1704
12_1115	Frame Composer GMD Packet Body Register 16 (HDMI_FC_GMD_PB16)	8	R/W	00h	33.5.175/ 1704
12_1116	Frame Composer GMD Packet Body Register 17 (HDMI_FC_GMD_PB17)	8	R/W	00h	33.5.176/ 1705
12_1117	Frame Composer GMD Packet Body Register 18 (HDMI_FC_GMD_PB18)	8	R/W	00h	33.5.177/ 1705
12_1118	Frame Composer GMD Packet Body Register 19 (HDMI_FC_GMD_PB19)	8	R/W	00h	33.5.178/ 1706
12_1119	Frame Composer GMD Packet Body Register 20 (HDMI_FC_GMD_PB20)	8	R/W	00h	33.5.179/ 1706
12_111A	Frame Composer GMD Packet Body Register 21 (HDMI_FC_GMD_PB21)	8	R/W	00h	33.5.180/ 1707
12_111B	Frame Composer GMD Packet Body Register 22 (HDMI_FC_GMD_PB22)	8	R/W	00h	33.5.181/ 1707
12_111C	Frame Composer GMD Packet Body Register 23 (HDMI_FC_GMD_PB23)	8	R/W	00h	33.5.182/ 1708
12_111D	Frame Composer GMD Packet Body Register 24 (HDMI_FC_GMD_PB24)	8	R/W	00h	33.5.183/ 1708
12_111E	Frame Composer GMD Packet Body Register 25 (HDMI_FC_GMD_PB25)	8	R/W	00h	33.5.184/ 1709
12_111F	Frame Composer GMD Packet Body Register 26 (HDMI_FC_GMD_PB26)	8	R/W	00h	33.5.185/ 1709

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_1120	Frame Composer GMD Packet Body Register 27 (HDMI_FC_GMD_PB27)	8	R/W	00h	33.5.186/ 1710
12_1200	Frame Composer Video/Audio Force Enable Register (HDMI_FC_DBGFORCE)	8	R/W	00h	33.5.187/ 1710
12_1201	Frame Composer Audio Channel 0 Register 0 (HDMI_FC_DBGAUD0CH0)	8	R/W	00h	33.5.188/ 1711
12_1202	Frame Composer Audio Channel 0 Register 1 (HDMI_FC_DBGAUD1CH0)	8	R/W	00h	33.5.189/ 1712
12_1203	Frame Composer Audio Channel 0 Register 2 (HDMI_FC_DBGAUD2CH0)	8	R/W	00h	33.5.190/ 1712
12_1204	Frame Composer Audio Channel 1 Register 0 (HDMI_FC_DBGAUD0CH1)	8	R/W	00h	33.5.191/ 1713
12_1205	Frame Composer Audio Channel 1 Register 1 (HDMI_FC_DBGAUD1CH1)	8	R/W	00h	33.5.192/ 1713
12_1206	Frame Composer Audio Channel 1 Register 2 (HDMI_FC_DBGAUD2CH1)	8	R/W	00h	33.5.193/ 1714
12_1207	Frame Composer Debug Audio Channel 2 Register 0 (HDMI_FC_DBGAUD0CH2)	8	R/W	00h	33.5.194/ 1714
12_1208	Frame Composer Debug Audio Channel 2 Register 1 (HDMI_FC_DBGAUD1CH2)	8	R/W	00h	33.5.195/ 1715
12_1209	Frame Composer Audio Channel 2 Register 2 (HDMI_FC_DBGAUD2CH2)	8	R/W	00h	33.5.196/ 1715
12_120A	Frame Composer Audio Channel 3 Register 0 (HDMI_FC_DBGAUD0CH3)	8	R/W	00h	33.5.197/ 1716
12_120B	Frame Composer Audio Channel 3 Register 1 (HDMI_FC_DBGAUD1CH3)	8	R/W	00h	33.5.198/ 1716
12_120C	Frame Composer Audio Channel 3 Register 2 (HDMI_FC_DBGAUD2CH3)	8	R/W	00h	33.5.199/ 1717
12_120D	Frame Composer Audio Channel 4 Register 0 (HDMI_FC_DBGAUD0CH4)	8	R/W	00h	33.5.200/ 1717
12_120E	Frame Composer Audio Channel 4 Register 1 (HDMI_FC_DBGAUD1CH4)	8	R/W	00h	33.5.201/ 1718
12_120F	Frame Composer Audio Channel 4 Register 2 (HDMI_FC_DBGAUD2CH4)	8	R/W	00h	33.5.202/ 1718
12_1210	Frame Composer Audio Channel 5 Register 0 (HDMI_FC_DBGAUD0CH5)	8	R/W	00h	33.5.203/ 1719
12_1211	Frame Composer Audio Channel 5 Register 1 (HDMI_FC_DBGAUD1CH5)	8	R/W	00h	33.5.204/ 1719
12_1212	Frame Composer Audio Channel 5 Register 2 (HDMI_FC_DBGAUD2CH5)	8	R/W	00h	33.5.205/ 1720
12_1213	Frame Composer Audio Channel 6 Register 0 (HDMI_FC_DBGAUD0CH6)	8	R/W	00h	33.5.206/ 1720
12_1214	Frame Composer Audio Channel 6 Register 1 (HDMI_FC_DBGAUD1CH6)	8	R/W	00h	33.5.207/ 1721

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_1215	Frame Composer Audio Channel 6 Register 2 (HDMI_FC_DBGAUD2CH6)	8	R/W	00h	33.5.208/ 1721
12_1216	Frame Composer Audio Channel 7 Register 1 (HDMI_FC_DBGAUD0CH7)	8	R/W	00h	33.5.209/ 1722
12_1217	Frame Composer Audio Channel 7 Register 0 (HDMI_FC_DBGAUD1CH7)	8	R/W	00h	33.5.210/ 1722
12_1218	Frame Composer Audio Channel 7 Register 2 (HDMI_FC_DBGAUD2CH7)	8	R/W	00h	33.5.211/ 1723
12_1219	Frame Composer TMDS Channel 0 Register (HDMI_FC_DBGTMDS0)	8	R/W	00h	33.5.212/ 1723
12_121A	Frame Composer TMDS Channel 1 Register (HDMI_FC_DBGTMDS1)	8	R/W	00h	33.5.213/ 1724
12_121B	Frame Composer TMDS Channel 2 Register (HDMI_FC_DBGTMDS2)	8	R/W	00h	33.5.214/ 1724
12_3000	PHY Configuration Register (HDMI_PHY_CONF0)	8	R/W	06h	33.5.215/ 1725
12_3001	PHY Test Interface Register 0 (HDMI_PHY_TST0)	8	R/W	00h	33.5.216/ 1726
12_3002	PHY Test Interface Register 1 (HDMI_PHY_TST1)	8	R/W	00h	33.5.217/ 1726
12_3003	PHY Test Interface Register 2 (HDMI_PHY_TST2)	8	R	00h	33.5.218/ 1727
12_3004	PHY RXSENSE, PLL lock, and HPD Status Register (HDMI_PHY_STAT0)	8	R	00h	33.5.219/ 1727
12_3005	PHY RXSENSE, PLL lock, and HPD Interrupt Register (HDMI_PHY_INT0)	8	R	00h	33.5.220/ 1728
12_3006	PHY RXSENSE, PLL lock, and HPD Mask Register (HDMI_PHY_MASK0)	8	R/W	00h	33.5.221/ 1729
12_3007	PHY RXSENSE, PLL lock and HPD Polarity Register (HDMI_PHY_POL0)	8	R/W	F3h	33.5.222/ 1730
12_3020	PHY I2C Slave Address Configuration Register (HDMI_PHY_I2CM_SLAVE_ADDR)	8	R/W	00h	33.5.223/ 1731
12_3021	PHY I2C Address Configuration Register (HDMI_PHY_I2CM_ADDRESS_ADDR)	8	R/W	00h	33.5.224/ 1731
12_3022	PHY I2C Data Write Register 1 (HDMI_PHY_I2CM_DATAO_1_ADDR)	8	R/W	00h	33.5.225/ 1732
12_3023	PHY I2C Data Write Register 0 (HDMI_PHY_I2CM_DATAO_0_ADDR)	8	R/W	00h	33.5.226/ 1733
12_3024	PHY I2C Data Read Register 1 (HDMI_PHY_I2CM_DATAI_1_ADDR)	8	R	00h	33.5.227/ 1733
12_3025	PHY I2C Data Read Register 0 (HDMI_PHY_I2CM_DATAI_0_ADDR)	8	R/W	00h	33.5.228/ 1734
12_3026	PHY I2C Read/Write Operation (HDMI_PHY_I2CM_OPERATION_ADDR)	8	W	00h	33.5.229/ 1734

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_3027	PHY I2C Done Interrupt Register (HDMI_PHY_I2CM_INT_ADDR)	8	R/W	08h	33.5.230/ 1735
12_3028	PHY I2C Done Interrupt Register (HDMI_PHY_I2CM_CTLINT_ADDR)	8	R/W	88h	33.5.231/ 1736
12_3029	PHY I2C Speed Control Register (HDMI_PHY_I2CM_DIV_ADDR)	8	R/W	0Bh	33.5.232/ 1737
12_302A	PHY I2C Software Reset Register (HDMI_PHY_I2CM_SOFTRSTZ_ADDR)	8	R/W	01h	33.5.233/ 1737
12_302B	PHY I2C Slow Speed SCL High Level Control Register 1 (HDMI_PHY_I2CM_SS_SCL_HCNT_1_ADDR)	8	R/W	00h	33.5.234/ 1738
12_302C	PHY I2C Slow Speed SCL High Level Control Register 0 (HDMI_PHY_I2CM_SS_SCL_HCNT_0_ADDR)	8	R/W	6Ch	33.5.235/ 1739
12_302D	PHY I2C Slow Speed SCL Low Level Control Register 1 (HDMI_PHY_I2CM_SS_SCL_LCNT_1_ADDR)	8	R/W	00h	33.5.236/ 1739
12_302E	PHY I2C Slow Speed SCL Low Level Control Register 0 (HDMI_PHY_I2CM_SS_SCL_LCNT_0_ADDR)	8	R/W	7Fh	33.5.237/ 1740
12_302F	PHY I2C Fast Speed SCL High Level Control Register 1 (HDMI_PHY_I2CM_FS_SCL_HCNT_1_ADDR)	8	R/W	00h	33.5.238/ 1740
12_3030	PHY I2C Fast Speed SCL High Level Control Register 0 (HDMI_PHY_I2CM_FS_SCL_HCNT_0_ADDR)	8	R/W	11h	33.5.239/ 1741
12_3031	PHY I2C Fast Speed SCL Low Level Control Register 1 (HDMI_PHY_I2CM_FS_SCL_LCNT_1_ADDR)	8	R/W	00h	33.5.240/ 1741
12_3032	PHY I2C Fast Speed SCL Low Level Control Register 0 (HDMI_PHY_I2CM_FS_SCL_LCNT_0_ADDR)	8	R/W	24h	33.5.241/ 1742
12_3200	Audio Clock Regenerator N Value Register 1 (HDMI_AUD_N1)	8	R/W	00h	33.5.242/ 1742
12_3201	Audio Clock Regenerator N Value Register 2 (HDMI_AUD_N2)	8	R/W	00h	33.5.243/ 1743
12_3202	Audio Clock Regenerator N Value Register 3 (HDMI_AUD_N3)	8	R/W	00h	33.5.244/ 1743
12_3203	AUD_CTS1 (HDMI_AUD_CTS1)	8	R/W	00h	33.5.245/ 1744
12_3204	AUD_CTS2 (HDMI_AUD_CTS2)	8	R/W	00h	33.5.246/ 1744
12_3205	AUD_CTS3 (HDMI_AUD_CTS3)	8	R/W	00h	33.5.247/ 1745
12_3600	Audio DMA Start Register (HDMI_AHB_DMA_CONF0)	8	R/W	00h	33.5.248/ 1745
12_3601	AHB_DMA_START (HDMI_AHB_DMA_START)	8	R/W	00h	33.5.249/ 1746
12_3602	Audio DMA Stop Register (HDMI_AHB_DMA_STOP)	8	R/W	00h	33.5.250/ 1747
12_3603	Audio DMA FIFO Threshold Register (HDMI_AHB_DMA_THRSLD)	8	R/W	00h	33.5.251/ 1748

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_3604	Audio DMA Start Address Register 0 (HDMI_AHB_DMA_STRADDR0)	8	R/W	00h	33.5.252/ 1748
12_3605	Audio DMA Start Address Register 1 (HDMI_AHB_DMA_STRADDR1)	8	R/W	00h	33.5.253/ 1749
12_3606	Audio DMA Start Address Register 2 (HDMI_AHB_DMA_STRADDR2)	8	R/W	00h	33.5.254/ 1749
12_3607	Audio DMA Start Address Register 3 (HDMI_AHB_DMA_STRADDR3)	8	R/W	00h	33.5.255/ 1750
12_3608	Audio DMA Stop Address Register 0 (HDMI_AHB_DMA_STPADDR0)	8	R/W	00h	33.5.256/ 1750
12_3609	Audio DMA Stop Address Register 1 (HDMI_AHB_DMA_STPADDR1)	8	R/W	00h	33.5.257/ 1751
12_360A	Audio DMA Stop Address Register 2 (HDMI_AHB_DMA_STPADDR2)	8	R/W	00h	33.5.258/ 1751
12_360B	Audio DMA Stop Address Register 3 (HDMI_AHB_DMA_STPADDR3)	8	R/W	00h	33.5.259/ 1752
12_360C	Audio DMA Burst Start Address Register 0 (HDMI_AHB_DMA_BSTADDR0)	8	R	00h	33.5.260/ 1752
12_360D	Audio DMA Burst Start Address Register 1 (HDMI_AHB_DMA_BSTADDR1)	8	R	00h	33.5.261/ 1753
12_360E	Audio DMA Burst Start Address Register 2 (HDMI_AHB_DMA_BSTADDR2)	8	R	00h	33.5.262/ 1753
12_360F	Audio DMA Burst Start Address Register 3 (HDMI_AHB_DMA_BSTADDR3)	8	R	00h	33.5.263/ 1753
12_3610	Audio DMA Burst Length Register 0 (HDMI_AHB_DMA_MBLENGTH0)	8	R	00h	33.5.264/ 1754
12_3611	Audio DMA Burst Length Register 1 (HDMI_AHB_DMA_MBLENGTH1)	8	R	00h	33.5.265/ 1755
12_3612	Audio DMA Interrupt Status Register (HDMI_AHB_DMA_STAT)	8	R	00h	33.5.266/ 1755
12_3613	Audio DMA Interrupt Register (HDMI_AHB_DMA_INT)	8	R	00h	33.5.267/ 1756
12_3614	Audio DMA Mask Interrupt Register (HDMI_AHB_DMA_MASK)	8	R/W	00h	33.5.268/ 1757
12_3615	Audio DMA Polarity Interrupt Register (HDMI_AHB_DMA_POL)	8	R/W	00h	33.5.269/ 1758
12_3616	Audio DMA Channel Enable Configuration Register 1 (HDMI_AHB_DMA_CONF1)	8	R/W	00h	33.5.270/ 1759
12_3617	Audio DMA Buffer Interrupt Status Register (HDMI_AHB_DMA_BUFFSTAT)	8	R	00h	33.5.271/ 1760
12_3618	Audio DMA Buffer Interrupt Register (HDMI_AHB_DMA_BUFFINT)	8	R	00h	33.5.272/ 1761
12_3619	Audio DMA Buffer Mask Interrupt Register (HDMI_AHB_DMA_BUFFMASK)	8	R/W	00h	33.5.273/ 1762

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_361A	Audio DMA Buffer Polarity Interrupt Register (HDMI_AHB_DMA_BUFFPOL)	8	R/W	00h	33.5.274/ 1762
12_4001	Main Controller Synchronous Clock Domain Disable Register (HDMI_MC_CLKDIS)	8	R/W	00h	33.5.275/ 1763
12_4002	Main Controller Software Reset Register (HDMI_MC_SWRSTZREQ)	8	R/W	FFh	33.5.276/ 1764
12_4004	Main Controller Feed Through Control Register (HDMI_MC_FLOWCTRL)	8	R/W	00h	33.5.277/ 1765
12_4005	Main Controller PHY Reset Register (HDMI_MC_PHYRSTZ)	8	R/W	00h	33.5.278/ 1765
12_4006	Main Controller Clock Present Register (HDMI_MC_LOCKONCLOCK)	8	w1c	00h	33.5.279/ 1766
12_4007	Main Controller HEAC PHY Reset Register (HDMI_MC_HEACPHY_RST)	8	R/W	00h	33.5.280/ 1767
12_4100	Color Space Converter Interpolation and Decimation Configuration Register (HDMI_CSC_CFG)	8	R/W	00h	33.5.281/ 1767
12_4101	Color Space Converter Scale and Deep Color Configuration Register (HDMI_CSC_SCALE)	8	R/W	01h	33.5.282/ 1768
12_4102	CSC_COEF_A1_MSB (HDMI_CSC_COEF_A1_MSB)	8	R/W	20h	33.5.283/ 1769
12_4103	CSC_COEF_A1_LSB (HDMI_CSC_COEF_A1_LSB)	8	R/W	00h	33.5.284/ 1769
12_4104	CSC_COEF_A2_MSB (HDMI_CSC_COEF_A2_MSB)	8	R/W	00h	33.5.285/ 1770
12_4105	CSC_COEF_A2_LSB (HDMI_CSC_COEF_A2_LSB)	8	R/W	00h	33.5.286/ 1770
12_4106	CSC_COEF_A3_MSB (HDMI_CSC_COEF_A3_MSB)	8	R/W	00h	33.5.287/ 1771
12_4107	CSC_COEF_A3_LSB (HDMI_CSC_COEF_A3_LSB)	8	R/W	00h	33.5.288/ 1771
12_4108	CSC_COEF_A4_MSB (HDMI_CSC_COEF_A4_MSB)	8	R/W	00h	33.5.289/ 1772
12_4109	CSC_COEF_A4_LSB (HDMI_CSC_COEF_A4_LSB)	8	R/W	00h	33.5.290/ 1772
12_410A	CSC_COEF_B1_MSB (HDMI_CSC_COEF_B1_MSB)	8	R/W	00h	33.5.291/ 1773
12_410B	CSC_COEF_B1_LSB (HDMI_CSC_COEF_B1_LSB)	8	R/W	00h	33.5.292/ 1773
12_410C	CSC_COEF_B2_MSB (HDMI_CSC_COEF_B2_MSB)	8	R/W	20h	33.5.293/ 1774
12_410D	CSC_COEF_B2_LSB (HDMI_CSC_COEF_B2_LSB)	8	R/W	00h	33.5.294/ 1774
12_410E	CSC_COEF_B3_MSB (HDMI_CSC_COEF_B3_MSB)	8	R/W	00h	33.5.295/ 1775

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_410F	CSC_COEF_B3_LSB (HDMI_CSC_COEF_B3_LSB)	8	R/W	00h	33.5.296/ 1775
12_4110	CSC_COEF_B4_MSB (HDMI_CSC_COEF_B4_MSB)	8	R/W	00h	33.5.297/ 1776
12_4111	CSC_COEF_B4_LSB (HDMI_CSC_COEF_B4_LSB)	8	R/W	00h	33.5.298/ 1776
12_4112	CSC_COEF_C1_MSB (HDMI_CSC_COEF_C1_MSB)	8	R/W	00h	33.5.299/ 1777
12_4113	CSC_COEF_C1_LSB (HDMI_CSC_COEF_C1_LSB)	8	R/W	00h	33.5.300/ 1777
12_4114	CSC_COEF_C2_MSB (HDMI_CSC_COEF_C2_MSB)	8	R/W	00h	33.5.301/ 1778
12_4115	CSC_COEF_C2_LSB (HDMI_CSC_COEF_C2_LSB)	8	R/W	00h	33.5.302/ 1778
12_4116	CSC_COEF_C3_MSB (HDMI_CSC_COEF_C3_MSB)	8	R/W	20h	33.5.303/ 1779
12_4117	CSC_COEF_C3_LSB (HDMI_CSC_COEF_C3_LSB)	8	R/W	00h	33.5.304/ 1779
12_4118	CSC_COEFC4_MSB (HDMI_CSC_COEFC4_MSB)	8	R/W	00h	33.5.305/ 1780
12_4119	CSC_COEFC4_LSB (HDMI_CSC_COEFC4_LSB)	8	R/W	00h	33.5.306/ 1780
12_7D00	CEC_CTRL (HDMI_CEC_CTRL)	8	R/W	02h	33.5.307/ 1781
12_7D01	CEC_STAT (HDMI_CEC_STAT)	8	R	00h	33.5.308/ 1782
12_7D02	CEC_MASK (HDMI_CEC_MASK)	8	R/W	00h	33.5.309/ 1783
12_7D03	CEC_POLARITY (HDMI_CEC_POLARITY)	8	R/W	7Fh	33.5.310/ 1784
12_7D04	CEC_INT (HDMI_CEC_INT)	8	R	00h	33.5.311/ 1785
12_7D05	CEC_ADDR_L (HDMI_CEC_ADDR_L)	8	R/W	00h	33.5.312/ 1786
12_7D06	CEC_ADDR_H (HDMI_CEC_ADDR_H)	8	R/W	80h	33.5.313/ 1787
12_7D07	CEC_TX_CNT (HDMI_CEC_TX_CNT)	8	R/W	00h	33.5.314/ 1788
12_7D08	CEC_RX_CNT (HDMI_CEC_RX_CNT)	8	R	00h	33.5.315/ 1789
12_7D10	CEC_TX_DATA (HDMI_CEC_TX_DATA0)	8	R/W	00h	33.5.316/ 1790
12_7D11	CEC_TX_DATA (HDMI_CEC_TX_DATA1)	8	R/W	00h	33.5.316/ 1790

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_7D12	CEC_TX_DATA (HDMI_CEC_TX_DATA2)	8	R/W	00h	33.5.316/ 1790
12_7D13	CEC_TX_DATA (HDMI_CEC_TX_DATA3)	8	R/W	00h	33.5.316/ 1790
12_7D14	CEC_TX_DATA (HDMI_CEC_TX_DATA4)	8	R/W	00h	33.5.316/ 1790
12_7D15	CEC_TX_DATA (HDMI_CEC_TX_DATA5)	8	R/W	00h	33.5.316/ 1790
12_7D16	CEC_TX_DATA (HDMI_CEC_TX_DATA6)	8	R/W	00h	33.5.316/ 1790
12_7D17	CEC_TX_DATA (HDMI_CEC_TX_DATA7)	8	R/W	00h	33.5.316/ 1790
12_7D18	CEC_TX_DATA (HDMI_CEC_TX_DATA8)	8	R/W	00h	33.5.316/ 1790
12_7D19	CEC_TX_DATA (HDMI_CEC_TX_DATA9)	8	R/W	00h	33.5.316/ 1790
12_7D1A	CEC_TX_DATA (HDMI_CEC_TX_DATA10)	8	R/W	00h	33.5.316/ 1790
12_7D1B	CEC_TX_DATA (HDMI_CEC_TX_DATA11)	8	R/W	00h	33.5.316/ 1790
12_7D1C	CEC_TX_DATA (HDMI_CEC_TX_DATA12)	8	R/W	00h	33.5.316/ 1790
12_7D1D	CEC_TX_DATA (HDMI_CEC_TX_DATA13)	8	R/W	00h	33.5.316/ 1790
12_7D1E	CEC_TX_DATA (HDMI_CEC_TX_DATA14)	8	R/W	00h	33.5.316/ 1790
12_7D1F	CEC_TX_DATA (HDMI_CEC_TX_DATA15)	8	R/W	00h	33.5.316/ 1790
12_7D20	CEC_RX_DATA (HDMI_CEC_RX_DATA0)	8	R	00h	33.5.317/ 1790
12_7D21	CEC_RX_DATA (HDMI_CEC_RX_DATA1)	8	R	00h	33.5.317/ 1790
12_7D22	CEC_RX_DATA (HDMI_CEC_RX_DATA2)	8	R	00h	33.5.317/ 1790
12_7D23	CEC_RX_DATA (HDMI_CEC_RX_DATA3)	8	R	00h	33.5.317/ 1790
12_7D24	CEC_RX_DATA (HDMI_CEC_RX_DATA4)	8	R	00h	33.5.317/ 1790
12_7D25	CEC_RX_DATA (HDMI_CEC_RX_DATA5)	8	R	00h	33.5.317/ 1790
12_7D26	CEC_RX_DATA (HDMI_CEC_RX_DATA6)	8	R	00h	33.5.317/ 1790
12_7D27	CEC_RX_DATA (HDMI_CEC_RX_DATA7)	8	R	00h	33.5.317/ 1790

Table continues on the next page...

HDMI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_7D28	CEC_RX_DATA (HDMI_CEC_RX_DATA8)	8	R	00h	33.5.317/ 1790
12_7D29	CEC_RX_DATA (HDMI_CEC_RX_DATA9)	8	R	00h	33.5.317/ 1790
12_7D2A	CEC_RX_DATA (HDMI_CEC_RX_DATA10)	8	R	00h	33.5.317/ 1790
12_7D2B	CEC_RX_DATA (HDMI_CEC_RX_DATA11)	8	R	00h	33.5.317/ 1790
12_7D2C	CEC_RX_DATA (HDMI_CEC_RX_DATA12)	8	R	00h	33.5.317/ 1790
12_7D2D	CEC_RX_DATA (HDMI_CEC_RX_DATA13)	8	R	00h	33.5.317/ 1790
12_7D2E	CEC_RX_DATA (HDMI_CEC_RX_DATA14)	8	R	00h	33.5.317/ 1790
12_7D2F	CEC_RX_DATA (HDMI_CEC_RX_DATA15)	8	R	00h	33.5.317/ 1790
12_7D30	CEC_LOCK (HDMI_CEC_LOCK)	8	R/W	00h	33.5.318/ 1791
12_7D31	CEC_WKUPCTRL (HDMI_CEC_WKUPCTRL)	8	R/W	FFh	33.5.319/ 1791
12_7E00	I2CM_SLAVE (HDMI_I2CM_SLAVE)	8	R/W	00h	33.5.320/ 1792
12_7E01	I2CM_ADDRESS (HDMI_I2CM_ADDRESS)	8	R/W	00h	33.5.321/ 1793
12_7E02	I2CM_DATAO (HDMI_I2CM_DATAO)	8	R/W	00h	33.5.322/ 1793
12_7E03	I2CM_DATAI (HDMI_I2CM_DATAI)	8	R	00h	33.5.323/ 1794
12_7E04	I2CM_OPERATION (HDMI_I2CM_OPERATION)	8	W	00h	33.5.324/ 1794
12_7E05	I2CM_INT (HDMI_I2CM_INT)	8	R/W	08h	33.5.325/ 1795
12_7E06	I2CM_CTLINT (HDMI_I2CM_CTLINT)	8	R/W	88h	33.5.326/ 1796
12_7E07	I2CM_DIV (HDMI_I2CM_DIV)	8	R/W	0Bh	33.5.327/ 1796
12_7E08	I2CM_SEGADDR (HDMI_I2CM_SEGADDR)	8	R/W	00h	33.5.328/ 1797
12_7E09	I2CM_SOFTRSTZ (HDMI_I2CM_SOFTRSTZ)	8	R/W	01h	33.5.329/ 1798
12_7E0A	I2CM_SEGPTR (HDMI_I2CM_SEGPTR)	8	R/W	00h	33.5.330/ 1798
12_7E0B	I2CM_SS_SCL_HCNT_1_ADDR (HDMI_I2CM_SS_SCL_HCNT_1_ADDR)	8	R/W	00h	33.5.331/ 1799

Table continues on the next page...

HDMI memory map (continued)

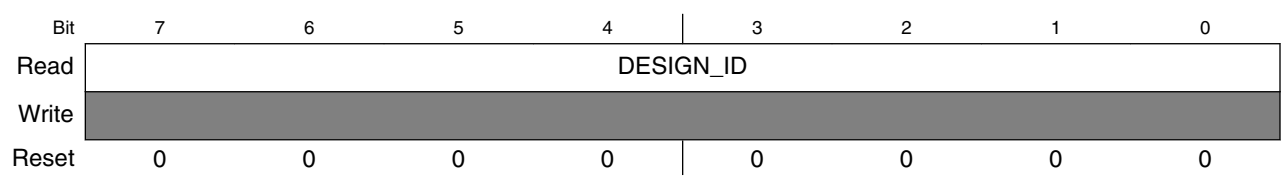
Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
12_7E0C	I2CM_SS_SCL_HCNT_0_ADDR (HDMI_I2CM_SS_SCL_HCNT_0_ADDR)	8	R/W	6Ch	33.5.332/ 1799
12_7E0D	I2CM_SS_SCL_LCNT_1_ADDR (HDMI_I2CM_SS_SCL_LCNT_1_ADDR)	8	R/W	00h	33.5.333/ 1800
12_7E0E	I2CM_SS_SCL_LCNT_0_ADDR (HDMI_I2CM_SS_SCL_LCNT_0_ADDR)	8	R/W	7Fh	33.5.334/ 1800
12_7E0F	I2CM_FS_SCL_HCNT_1_ADDR (HDMI_I2CM_FS_SCL_HCNT_1_ADDR)	8	R/W	00h	33.5.335/ 1801
12_7E10	I2CM_FS_SCL_HCNT_0_ADDR (HDMI_I2CM_FS_SCL_HCNT_0_ADDR)	8	R/W	11h	33.5.336/ 1801
12_7E11	I2CM_FS_SCL_LCNT_1_ADDR (HDMI_I2CM_FS_SCL_LCNT_1_ADDR)	8	R/W	00h	33.5.337/ 1802
12_7E12	I2CM_FS_SCL_LCNT_0_ADDR (HDMI_I2CM_FS_SCL_LCNT_0_ADDR)	8	R/W	24h	33.5.338/ 1802
12_7F00	BASE_POINTER_ADDR (HDMI_BASE_POINTER_ADDR)	8	R/W	00h	33.5.339/ 1803

33.5.1 Design Identification Register (HDMI_DESIGN_ID)

The following are the registers used to identify the HDMI TX controller.

- Name: Design Identification Register
- Address Offset: 0x0000
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 0h offset = 12_0000h



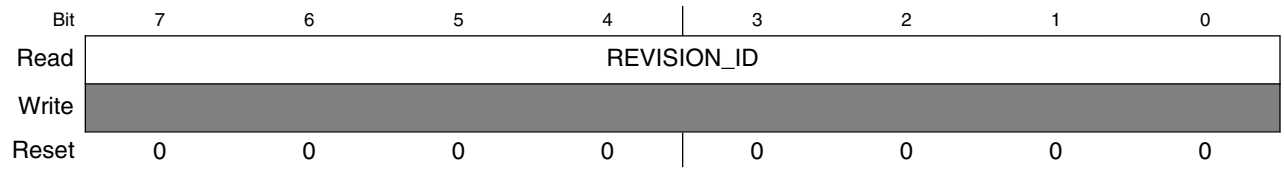
HDMI_DESIGN_ID field descriptions

Field	Description
DESIGN_ID	This is a 1 byte design ID code fixed by Freescale that Identifies the main revision of the HDMI TX controller. For example, HDMI TX 1.30a, DESIGN_ID = 11h; REVISION_ID = 0Ah

33.5.2 Revision Identification Register (HDMI_REVISION_ID)

- Name: Revision Identification Register
- Address Offset: 0x0001
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 1h offset = 12_0001h



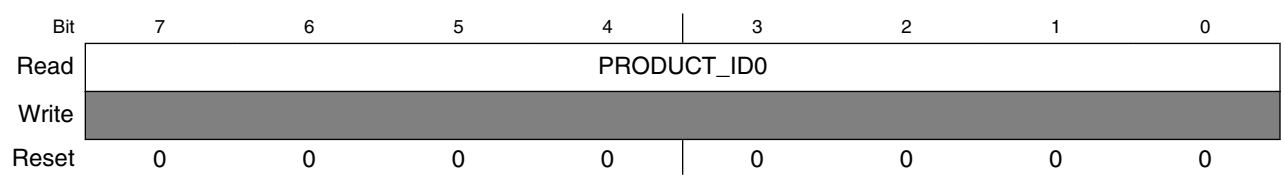
HDMI_REVISION_ID field descriptions

Field	Description
REVISION_ID	This is a one byte revision ID code fixed by Freescale that Identifies the main revision of the HDMI TX controller. For example, HDMI TX 1.30a, DESIGN_ID = 12h; REVISION_ID = 0Ah

33.5.3 Product Identification Register 0 (HDMI_PRODUCT_ID0)

- Name: Product Identification Register 0
- Address Offset: 0x0002
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 2h offset = 12_0002h



HDMI_PRODUCT_ID0 field descriptions

Field	Description
PRODUCT_ID0	This one byte fixed code Identifies Freescale's product line ("A0h" for HDMI TX products).

33.5.4 Product Identification Register 1 (HDMI_PRODUCT_ID1)

- Name: Product Identification Register 1
- Address Offset: 0x0003
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 3h offset = 12_0003h

Bit	7	6	5	4	3	2	1	0
Read	PRODUCT_ID1							
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_PRODUCT_ID1 field descriptions

Field	Description
PRODUCT_ID1	This one byte fixed code identifies Freescale's product line according to: 01h HDMI TX Controller C1h HDMI TX Controller with HDCP encryption engine

33.5.5 Configuration Identification Register 0 (HDMI_CONFIG0_ID)

- Name: Configuration Identification Register 0
- Address Offset: 0x0004
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 4h offset = 12_0004h

Bit	7	6	5	4	3	2	1	0
Read	prepen	audhbr	audspdif	audi2s	hdmi14	csc	cec	hdcp
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_CONFIG0_ID field descriptions

Field	Description
7 prepen	Indicates if it is possible to use internal pixel repetition
6 audhbr	Indicates if HBR interface is present
5 audspdif	Indicates if SPDIF interface is present
4 audi2s	Indicates if I2S interface is present
3 hdmi14	Indicates if HDMI 1.4 features are present
2 csc	Indicates if Color Space Conversion block is present
1 cec	Indicates if CEC is present
0 hdcp	Indicates if HDCP is present

33.5.6 Configuration Identification Register 1 (HDMI_CONFIG1_ID)

- Name: Configuration Identification Register 1
- Address Offset: 0x0005
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 5h offset = 12_0005h

Bit	7	6	5	4	3	2	1	0
Read	Reserved			confsfrdir	confi2c	confocp	confapb	confahb
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_CONFIG1_ID field descriptions

Field	Description
7–5 -	This field is reserved.
4 confsfrdir	Indicates that configuration interface is SFR interface
3 confi2c	Indicates that configuration interface is I2C interface

Table continues on the next page...

HDMI_CONFIG1_ID field descriptions (continued)

Field	Description
2 confocp	Indicates that configuration interface is OCP interface
1 confapb	Indicates that configuration interface is APB interface
0 confahb	Indicates that configuration interface is AHB interface

33.5.7 Configuration Identification Register 2 (HDMI_CONFIG2_ID)

- Name: Configuration Identification Register 2
- Address Offset: 0x0006
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 6h offset = 12_0006h

Bit	7	6	5	4	3	2	1	0
Read	phytype							
Write								
Reset	0	0	0	0	0	0	0	0

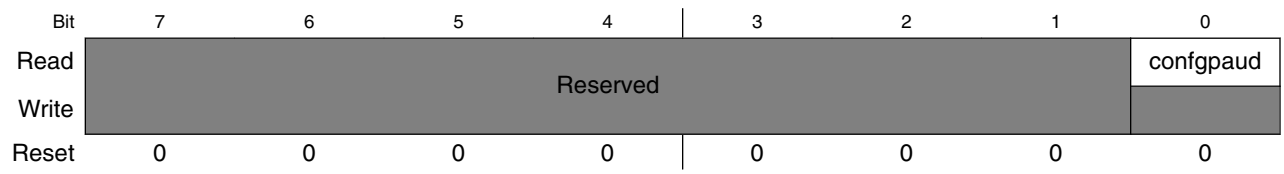
HDMI_CONFIG2_ID field descriptions

Field	Description
phytype	Indicates the type of PHY interface selected: 00h Legacy PHY (HDMI TX PHY) F2h PHY_Gen2 (HDMI 3D TX PHY) E2h PHY_Gen2 (HDMI 3D TX PHY) + HEAC PHY

33.5.8 Configuration Identification Register 3 (HDMI_CONFIG3_ID)

- Name: Configuration Identification Register 3
- Address Offset: 0x0007
- Size: 8 bits
- Value after Reset: Implementation Dependent
- Access: Read

Address: 12_0000h base + 7h offset = 12_0007h



HDMI_CONFIG3_ID field descriptions

Field	Description
7–1 -	This field is reserved.
0 confgpaud	Indicates that configuration interface is Generic Parallel Audio (GPAUD) interface

33.5.9 Frame Composer Interrupt Status Register 0 (HDMI_IH_FC_STAT0)

This section describes clear on write (1 to corresponding bit) status registers, which contain the following active-high, sticky bit interrupts.

HDMI TX introduces a new set of sticky bit mute control registers (IH_MUTE_FC_STAT0 to IH_MUTE_AHBDMAAUD_STAT0) that correspond to the interrupt registers. You can ignore a sticky bit interrupt by setting the corresponding mute control register bit to 1. This puts the global interrupt line on a higher priority than the sticky bit interrupt.

- Address Offset: 0x0100
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 100h offset = 12_0100h

Bit	7	6	5	4	3	2	1	0
Read	AUDI	ACP	HBR	DST	OBA	AUDS	ACR	NULL
Write	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0

HDMI_IH_FC_STAT0 field descriptions

Field	Description
7 AUDI	Active after successful transmission of an Audio InfoFrame packet.
6 ACP	Active after successful transmission of an Audio Content Protection packet.
5 HBR	Active after successful transmission of an Audio HBR packet.
4 DST	Reserved
3 OBA	Reserved
2 AUDS	Active after successful transmission of an Audio Sample packet. Due to high number of audio sample packets transmitted, this interrupt is by default masked at frame composer.
1 ACR	Active after successful transmission of an Audio Clock Regeneration (N/CTS transmission) packet.
0 NULL	Active after successful transmission of an Null packet. Due to high number of audio sample packets transmitted, this interrupt is by default masked at frame composer.

33.5.10 Frame Composer Interrupt Status Register 1 (HDMI_IH_FC_STAT1)

- Address Offset: 0x0101
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 101h offset = 12_0101h

Bit	7	6	5	4	3	2	1	0
Read	GMD	ISCR1	ISCR2	VSD	SPD	MPEG	AVI	GCP
Write	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0

HDMI_IH_FC_STAT1 field descriptions

Field	Description
7 GMD	Active after successful transmission of an Gamut metadata packet.
6 ISCR1	Active after successful transmission of an International Standard Recording Code 1 packet.
5 ISCR2	Active after successful transmission of an International Standard Recording Code 2 packet.
4 VSD	Active after successful transmission of an Vendor Specific Data infoFrame packet.
3 SPD	Active after successful transmission of an Source Product Descriptor infoFrame packet.
2 MPEG	Reserved
1 AVI	Active after successful transmission of an AVI infoFrame packet.
0 GCP	Active after successful transmission of an General Control Packet.

33.5.11 Frame Composer Interrupt Status Register 2 (HDMI_IH_FC_STAT2)

- Address Offset: 0x0102
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 102h offset = 12_0102h

Bit	7	6	5	4	3	2	1	0
Read	Reserved						LowPriority_ overflow	HighPriority _overflow
Write							w1c	w1c
Reset	0	0	0	0	0	0	0	0

HDMI_IH_FC_STAT2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	Frame Composer low priority packet queue descriptor overflow indication.

Table continues on the next page...

HDMI_IH_FC_STAT2 field descriptions (continued)

Field	Description
0 HighPriority_ overflow	Frame Composer high priority packet queue descriptor overflow indication.

33.5.12 Audio Sampler Interrupt Status Register (HDMI_IH_AS_STAT0)

- Address Offset: 0x0103
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 103h offset = 12_0103h

Bit	7	6	5	4	3	2	1	0
Read	Reserved					Aud_fifo_ underflow_ thr	Aud_fifo_ underflow	Aud_fifo_ overflow
Write						w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0

HDMI_IH_AS_STAT0 field descriptions

Field	Description
7–3 -	This field is reserved. Reserved
2 Aud_fifo_ underflow_thr	Audio Sampler audio FIFO empty threshold (four samples) indication. Only valid in HBR audio.
1 Aud_fifo_ underflow	Audio Sampler audio FIFO empty indication.
0 Aud_fifo_ overflow	Audio Sampler audio FIFO full indication.

33.5.13 PHY Interface Interrupt Status Register (HDMI_IH_PHY_STAT0)

- Address Offset: 0x0104
- Size: 8 bits
- Value after Reset: 0x00
- Access: Clear on Write/Read

Address: 12_0000h base + 104h offset = 12_0104h

Bit	7	6	5	4
Read	Reserved		RX_SENSE3	RX_SENSE2
Write			w1c	w1c
Reset	0	0	0	0

Bit	3	2	1	0
Read	RX_SENSE1	RX_SENSE0	TX_PHY_LOCK	HDP
Write	w1c	w1c	w1c	w1c
Reset	0	0	0	0

HDMI_IH_PHY_STAT0 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 RX_SENSE3	TX PHY RX_SENSE indication for driver 3. You may need to mask or change polarity of this interrupt after it has become active.
4 RX_SENSE2	TX PHY RX_SENSE indication for driver 2. You may need to mask or change polarity of this interrupt after it has become active.
3 RX_SENSE1	TX PHY RX_SENSE indication for driver 1. You may need to mask or change polarity of this interrupt after it has become active.
2 RX_SENSE0	TX PHY RX_SENSE indication for driver 0. You may need to mask or change polarity of this interrupt after it has become active.
1 TX_PHY_LOCK	TX PHY PLL lock indication. Please refer to PHY datasheet for more information. You may need to mask or change polarity of this interrupt after it has become active.
0 HDP	HDMI Hot Plug Detect indication. You may need to mask or change polarity of this interrupt after it has become active.

33.5.14 E-DDC I2C Master Interrupt Status Register (HDMI_IH_I2CM_STAT0)

- Address Offset: 0x0105
- Size: 8 bits
- Value after Reset: 0x00
- Access: Clear on Write/Read

Address: 12_0000h base + 105h offset = 12_0105h

Bit	7	6	5	4
Read	Reserved			
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved		I2Cmasterdone	I2CMASTER_ERROR
Write			w1c	w1c
Reset	0	0	0	0

HDMI_IH_I2CM_STAT0 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 I2Cmasterdone	I2C Master done indication
0 I2CMASTER_ERROR	I2C Master error indication

33.5.15 CEC Interrupt Status Register (HDMI_IH_CEC_STAT0)

- Address Offset: 0x0106
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 106h offset = 12_0106h

Bit	7	6	5	4	3	2	1	0
Read	Reserved	WAKEUP	ERROR_ FOLLOW	ERROR_ INITIATOR	ARB_LOST	NACK	EOM	DONE
Write		w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0

HDMI_IH_CEC_STAT0 field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 WAKEUP	CEC Wake-up indication
5 ERROR_ FOLLOW	CEC Error_follow indication
4 ERROR_ INITIATOR	CEC Error_follow indication
3 ARB_LOST	CEC Arb_Lost indication
2 NACK	CEC Nack indication
1 EOM	CEC End of Message Indication
0 DONE	CEC Done Indication

33.5.16 Video Packetizer Interrupt Status Register (HDMI_IH_VP_STAT0)

- Address Offset: 0x0107
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 107h offset = 12_0107h

Bit	7	6	5	4
Read	fifofullrepet	fifoemptyrepet	fifofullpp	fifoemptypp
Write	w1c	w1c	w1c	w1c
Reset	0	0	0	0

Bit	3	2	1	0
Read	fifofullremap	fifoemptyremap	fifofullbyp	fifoemptybyp
Write	w1c	w1c	w1c	w1c
Reset	0	0	0	0

HDMI_IH_VP_STAT0 field descriptions

Field	Description
7 fifofullrepet	Video packetizer pixel repeater FIFO full interrupt
6 fifoemptyrepet	Video packetizer pixel repeater FIFO empty interrupt
5 fifofullpp	Video packetizer pixel packing FIFO full interrupt
4 fifoemptypp	Video packetizer pixel packing FIFO empty interrupt
3 fifofullremap	Video packetizer pixel YCC 422 re-mapper FIFO full interrupt
2 fifoemptyremap	Video packetizer pixel YCC 422 re-mapper FIFO empty interrupt
1 fifofullbyp	Video packetizer 8-bit bypass fifo full interrupt
0 fifoemptybyp	Video packetizer 8-bit bypass fifo empty interrupt

33.5.17 PHY GEN2 I2C Master Interrupt Status Register (HDMI_IH_I2CMPHY_STAT0)

This clear on write (1 to corresponding bit) register contains the following active high sticky bit interrupts. That I2C Master PHY is the I2C Master block used to access the PHY I2C Slave.

- Address Offset: 0x0108
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 108h offset = 12_0108h

Bit	7	6	5	4
Read	Reserved			
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved		i2cmphydone	i2cmphyerror
Write			w1c	w1c
Reset	0	0	0	0

HDMI_IH_I2CMPHY_STAT0 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 i2cmphydone	I2C Master PHY done indication
0 i2cmphyerror	I2C Master PHY error indication

33.5.18 AHB Audio DMA Interrupt Status Register (HDMI_IH_AHBDMAAUD_STAT0)

Address Offset: 0x0109

Size: 8 bits

Value after Reset: 0x00

Access: Read/Clear on Write

Address: 12_0000h base + 109h offset = 12_0109h

Bit	7	6	5	4
Read	Reserved		ahbdmaaud_interror	ahbdmaaud_ intlostownership
Write			w1c	w1c
Reset	0	0	0	0

Bit	3	2	1	0
Read	ahbdmaaud_intretrysplit	ahbdmaaud_intdone	ahbdmaaud_intbufffull	ahbdmaaud_ intbuffempty
Write	w1c	w1c	w1c	w1c
Reset	0	0	0	0

HDMI_IH_AHBDMAAUD_STAT0 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 ahbdmaaud_interror	AHB audio DMA error interrupt
4 ahbdmaaud_ intlostownership	AHB audio DMA lost ownership interrupt
3 ahbdmaaud_ intretrysplit	AHB audio DMA RETRY/SPLIT interrupt
2 ahbdmaaud_intdone	AHB audio DMA done interrupt
1 ahbdmaaud_ intbufffull	AHB audio DMA Buffer full interrupt
0 ahbdmaaud_ intbuffempty	AHB audio DMA Buffer empty interrupt

33.5.19 Frame Composer Interrupt Mute Control Register 0 (HDMI_IH_MUTE_FC_STAT0)

- Address Offset: 0x0180
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 180h offset = 12_0180h

Bit	7	6	5	4	3	2	1	0
Read Write	AUDI	ACP	HBR	DST	OBA	AUDS	ACR	NULL
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_FC_STAT0 field descriptions

Field	Description
7 AUDI	When set to 1, mutes IH_ FC_STAT0[7]
6 ACP	When set to 1, mutes IH_ FC_STAT0[6]
5 HBR	When set to 1, mutes IH_ FC_STAT0[5]
4 DST	When set to 1, mutes IH_ FC_STAT0[4]
3 OBA	When set to 1, mutes IH_ FC_STAT0[3]
2 AUDS	When set to 1, mutes IH_ FC_STAT0[2]
1 ACR	When set to 1, mutes IH_ FC_STAT0[1]
0 NULL	When set to 1, mutes IH_ FC_STAT0[0]

33.5.20 Frame Composer Interrupt Mute Control Register 1 (HDMI_IH_MUTE_FC_STAT1)

- Address Offset: 0x0181
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 181h offset = 12_0181h

Bit	7	6	5	4	3	2	1	0
Read Write	GMD	ISCR1	ISCR2	VSD	SPD	MPEG	AVI	GCP
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_FC_STAT1 field descriptions

Field	Description
7 GMD	When set to 1, mutes IH_ FC_STAT1[7]
6 ISCR1	When set to 1, mutes IH_ FC_STAT1[6]
5 ISCR2	When set to 1, mutes IH_ FC_STAT1[5]
4 VSD	When set to 1, mutes IH_ FC_STAT1[4]
3 SPD	When set to 1, mutes IH_ FC_STAT1[3]
2 MPEG	When set to 1, mutes IH_ FC_STAT1[2]
1 AVI	When set to 1, mutes IH_ FC_STAT1[1]
0 GCP	When set to 1, mutes IH_ FC_STAT1[0]

33.5.21 Frame Composer Interrupt Mute Control Register 2 (HDMI_IH_MUTE_FC_STAT2)

- Address Offset: 0x0182
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 182h offset = 12_0182h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						LowPriority_ overflow	HighPriority _overflow
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_FC_STAT2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	When set to 1, mutes IH_ FC_STAT2[1]
0 HighPriority_ overflow	When set to 1, mutes IH_ FC_STAT2[0]

33.5.22 Audio Sampler Interrupt Mute Control Register 0 (HDMI_IH_MUTE_AS_STAT0)

- Address Offset: 0x0183
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 183h offset = 12_0183h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved					Aud_fifo_ underflow_ thr	Aud_fifo_ underflow	Aud_fifo_ overflow
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_AS_STAT0 field descriptions

Field	Description
7–3 -	This field is reserved. Reserved
2 Aud_fifo_ underflow_thr	When set to 1, mutes IH_ AS_STAT0[2]
1 Aud_fifo_ underflow	When set to 1, mutes IH_ AS_STAT0[1]
0 Aud_fifo_ overflow	When set to 1, mutes IH_ AS_STAT0[0]

33.5.23 PHY Interface Interrupt Mute Control Register (HDMI_IH_MUTE_PHY_STAT0)

- Address Offset: 0x0184
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 184h offset = 12_0184h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		RX_ SENSE3	RX_ SENSE2	RX_ SENSE1	RX_ SENSE0	TX_PHY_ LOCK	HDP
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_PHY_STAT0 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 RX_SENSE3	When set to 1, mutes IH_ PHY_STAT0[5]
4 RX_SENSE2	When set to 1, mutes IH_ PHY_STAT0[4]
3 RX_SENSE1	When set to 1, mutes IH_ PHY_STAT0[3]
2 RX_SENSE0	When set to 1, mutes IH_ PHY_STAT0[2]
1 TX_PHY_LOCK	When set to 1, mutes IH_ PHY_STAT0[1]
0 HDP	When set to 1, mutes IH_ PHY_STAT0[0]

33.5.24 E-DDC I2C Master Interrupt Mute Control Register (HDMI_IH_MUTE_I2CM_STAT0)

- Address Offset: 0x0185
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 185h offset = 12_0185h

Bit	7	6	5	4
Read Write	Reserved			
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	Reserved		I2Cmasterdone	I2CMASTER_ERROR
Reset	0	0	0	0

HDMI_IH_MUTE_I2CM_STAT0 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 I2Cmasterdone	When set to 1, mutes IH_ I2CM_STAT0[1]
0 I2CMASTER_ERROR	When set to 1, mutes IH_ I2CM_STAT0[0]

33.5.25 CEC Interrupt Mute Control Register (HDMI_IH_MUTE_CEC_STAT0)

- Address Offset: 0x0186
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 186h offset = 12_0186h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	WAKEUP	ERROR_ FOLLOW	ERROR_ INITIATOR	ARB_LOST	NACK	EOM	DONE
Reset	0	0	0	0	0	0	0	0

HDMI_IH_MUTE_CEC_STAT0 field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 WAKEUP	When set to 1, mutes IH_ CEC_STAT0[6]
5 ERROR_ FOLLOW	When set to 1, mutes IH_ CEC_STAT0[5]
4 ERROR_ INITIATOR	When set to 1, mutes IH_ CEC_STAT0[4]
3 ARB_LOST	When set to 1, mutes IH_ CEC_STAT0[3]
2 NACK	When set to 1, mutes IH_ CEC_STAT0[2]
1 EOM	When set to 1, mutes IH_ CEC_STAT0[1]
0 DONE	When set to 1, mutes IH_ CEC_STAT0[0]

33.5.26 Video Packetizer Interrupt Mute Control Register (HDMI_IH_MUTE_VP_STAT0)

- Address Offset: 0x0187
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 187h offset = 12_0187h

Bit	7	6	5	4
Read Write	fifofullrepet	fifoemptyrepet	fifofullpp	fifoemptypp
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	fifofullremap	fifoemptyremap	fifofullbyp	fifoemptybyp
Reset	0	0	0	0

HDMI_IH_MUTE_VP_STAT0 field descriptions

Field	Description
7 fifofullrepet	When set to 1, mutes IH_ VP_STAT0[7]

Table continues on the next page...

HDMI_IH_MUTE_VP_STAT0 field descriptions (continued)

Field	Description
6 fifoemptyrepet	When set to 1, mutes IH_ VP_STAT0[6]
5 fifofullpp	When set to 1, mutes IH_ VP_STAT0[5]
4 fifoemptypp	When set to 1, mutes IH_ VP_STAT0[4]
3 fifofullremap	When set to 1, mutes IH_ VP_STAT0[3]
2 fifoemptyremap	When set to 1, mutes IH_ VP_STAT0[2]
1 fifofullbyp	When set to 1, mutes IH_ VP_STAT0[1]
0 fifoemptybyp	When set to 1, mutes IH_ VP_STAT0[0]

33.5.27 PHY GEN 2 I2C Master Interrupt Mute Control Register (HDMI_IH_MUTE_I2CMPHY_STAT0)

- Address Offset: 0x0188
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 188h offset = 12_0188h

Bit	7	6	5	4
Read Write	Reserved			
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	Reserved		i2cmphydone	i2cmphyerror
Reset	0	0	0	0

HDMI_IH_MUTE_I2CMPHY_STAT0 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 i2cmphydone	When set to 1, mutes IH_ I2CMPHY_STAT0[1]
0 i2cmphyerror	When set to 1, mutes IH_ I2CMPHY_STAT0[0]

33.5.28 AHB Audio DMA Interrupt Mute Control Register (HDMI_IH_MUTE_AHBDMAAUD_STAT0)

- Address Offset: 0x0189
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 189h offset = 12_0189h

Bit	7	6	5	4
Read Write	Reserved		ahbdmaaud_interror	ahbdmaaud_ intlostownership
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	ahbdmaaud_intretrysplit	ahbdmaaud_intdone	ahbdmaaud_intbufffull	ahbdmaaud_ intbuffempty
Reset	0	0	0	0

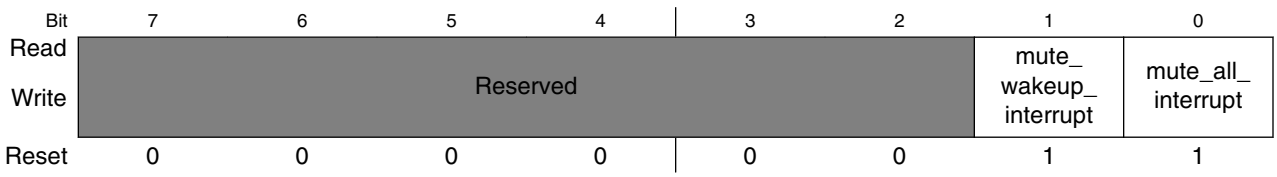
HDMI_IH_MUTE_AHBDMAAUD_STAT0 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 ahbdmaaud_interror	When set to 1, mutes IH_AHBDMAAUD_STAT0[5]
4 ahbdmaaud_ intlostownership	When set to 1, mutes IH_AHBDMAAUD_STAT0[4]
3 ahbdmaaud_ intretrysplit	When set to 1, mutes IH_AHBDMAAUD_STAT0[3]
2 ahbdmaaud_intdone	When set to 1, mutes IH_AHBDMAAUD_STAT0[2]
1 ahbdmaaud_ intbufffull	When set to 1, mutes IH_AHBDMAAUD_STAT0[1]
0 ahbdmaaud_ intbuffempty	When set to 1, mutes IH_AHBDMAAUD_STAT0[0]

33.5.29 Global Interrupt Mute Control Register (HDMI_IH_MUTE)

- Address Offset: 0x01FF
- Size: 8 bits
- Value after Reset: 0x03
- Access: Read/Write

Address: 12_0000h base + 1FFh offset = 12_01FFh



HDMI_IH_MUTE field descriptions

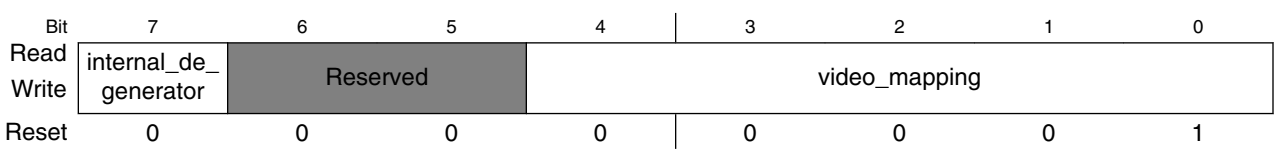
Field	Description
7–2 rsvd	This field is reserved.
1 mute_wakeup_ interrupt	When set to 1, mutes the wake-up interrupt line. The sticky bit interrupt continues with its state; only the wake up interrupt line is muted.
0 mute_all_ interrupt	When set to 1, mutes the main interrupt line (where all interrupts are ORed). The sticky bit interrupts continue with their state; only the main interrupt line will be muted.

33.5.30 Video Input Mapping and Internal Data Enable Configuration Register (HDMI_TX_INVID0)

This registers contains the input video mapping code as defined in Table 2-1.

- Address Offset: 0x0200
- Size: 8 bits
- Value after Reset: 0x01
- Access: Read/Write

Address: 12_0000h base + 200h offset = 12_0200h



HDMI_TX_INVID0 field descriptions

Field	Description
7 internal_de_ generator	Internal data enable (DE) generator enable. If data enable is not available for the input video the user may set this bit to one to activate the internal data enable generator. **NOTE:** This feature only works for input video modes that have native repetition (such as, all CEA videos). No desired pixel repetition can be used with this feature because these configurations only affect the Frame Composer and not this block.
6–5 -	This field is reserved. Reserved
video_mapping	video_mapping

33.5.31 Video Input Stuffing Enable Register (HDMI_TX_INSTUFFING)

This register enables the stuffing mechanism of the Video Sampler module in order to correctly perform Color Space Conversion of the ITU.601 standard YCC video. In this case, when "de" is low, the output video components gydata[15:0], rcrdata[15:0], and bcbdata[15:0] can be configured.

- Address Offset: 0x0201
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 201h offset = 12_0201h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved					BCBDATA_ STUFFING	RCRDATA_ STUFFING	GYDATA_ STUFFING
Reset	0	0	0	0	0	0	0	0

HDMI_TX_INSTUFFING field descriptions

Field	Description
7–3 -	This field is reserved. Reserved
2 BCBDATA_ STUFFING	BCBDATA stuffing bit 0 When the dataen signal is low, the value in the bcbdata[15:0] output is the one sampled from the corresponding input data. 1 When the dataen signal is low, the value in the bcbdata[15:0] output is given by the values in register TX_BCBDTA0 and TX_BCBDATA1.
1 RCRDATA_ STUFFING	RCRDATA stuffing bit

Table continues on the next page...

HDMI_TX_INSTUFFING field descriptions (continued)

Field	Description
	0 When the dataen signal is low, the value in the rcrdata[15:0] output is the one sampled from the corresponding input data. 1 When the dataen signal is low, the value in the rcrdata[15:0] output is given by the values in TX_RCRDTA0 and TX_RCRDATA1 registers.
0 GYDATA_ STUFFING	GYDATA stuffing bit 0 when the dataen signal is low, the value in the gydata[15:0] output is the one sampled from the corresponding input data. 1 When the dataen signal is low, the value in the gydata[15:0] output is given by the values in TX_GYDTA0 and TX_GYDATA1 registers.

33.5.32 Video Input GY Data Channel Stuffing Register 0 (HDMI_TX_GYDATA0)

- Address Offset: 0x0202
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 202h offset = 12_0202h

Bit	7	6	5	4	3	2	1	0
Read Write	gydata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_GYDATA0 field descriptions

Field	Description
gydata	gydata[7:0].This register defines the value of gydata[7:0] when TX_INSTUFFING[0] (gydata_stuffing) is set to 1b.

33.5.33 Video Input GY Data Channel Stuffing Register 1 (HDMI_TX_GYDATA1)

- Address Offset: 0x0203
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 203h offset = 12_0203h

Bit	7	6	5	4	3	2	1	0
Read Write	gydata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_GYDATA1 field descriptions

Field	Description
gydata	gydata[15:8].This register defines the value of gydata[15:8] when TX_INSTUFFING[0] (gydata_stuffing) is set to 1b.

33.5.34 Video Input RCR Data Channel Stuffing Register 0 (HDMI_TX_RCRDATA0)

- Address Offset: 0x0204
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 204h offset = 12_0204h

Bit	7	6	5	4	3	2	1	0
Read Write	rcrdata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_RCRDATA0 field descriptions

Field	Description
rcrdata	rcrdata[7:0]. This register defines the value of rcrydata[7:0] when TX_INSTUFFING[1] (rcrdata_stuffing) is set to 1b.

33.5.35 Video Input RCR Data Channel Stuffing Register 1 (HDMI_TX_RCRDATA1)

- Address Offset: 0x0205
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 205h offset = 12_0205h

Bit	7	6	5	4	3	2	1	0
Read Write	rcrdata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_RCRDATA1 field descriptions

Field	Description
rcrdata	rcrdata[15:8]. This register defines the value of rcrydata[15:8] when TX_INSTUFFING[1] (rcrdata_stuffing) is set to 1b.

33.5.36 Video Input RCB Data Channel Stuffing Register 0 (HDMI_TX_BCBDATA0)

- Address Offset: 0x0206
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 206h offset = 12_0206h

Bit	7	6	5	4	3	2	1	0
Read Write	bcbdata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_BCBDATA0 field descriptions

Field	Description
bcbdata	bcbdata[7:0]. This register defines the value of bcbdata[7:0] when TX_INSTUFFING[2] (bcbdata_stuffing) is set to 1b.

33.5.37 Video Input RCB Data Channel Stuffing Register 1 (HDMI_TX_BCBDATA1)

- Address Offset: 0x0207
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 207h offset = 12_0207h

Bit	7	6	5	4	3	2	1	0
Read Write	bcbdata							
Reset	0	0	0	0	0	0	0	0

HDMI_TX_BCBDATA1 field descriptions

Field	Description
bcbdata	bcbdata[15:8]. This register defines the value of bcbdata[15:8] when TX_INSTUFFING[2] (bcbdata_stuffing) is set to 1b.

33.5.38 Video Packetizer Packing Phase Status Register (HDMI_VP_STATUS)

- Address Offset: 0x0800
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 800h offset = 12_0800h

Bit	7	6	5	4	3	2	1	0
Read	Reserved				packing_phase			
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_VP_STATUS field descriptions

Field	Description
7–4 -	This field is reserved. Reserved

Table continues on the next page...

HDMI_VP_STATUS field descriptions (continued)

Field	Description
packing_phase	Read only register that holds the "packing phase" output by the Video packetizer block. For more information about "packing" video data, refer to the HDMI1.4a specification. The register is updated at tmds clock rate.

33.5.39 Video Packetizer Pixel Repetition and Color Depth Register (HDMI_VP_PR_CD)

This register configures the Color Depth of the input video and Pixel repetition to apply to video.

- Address Offset: 0x0801
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 801h offset = 12_0801h

Bit	7	6	5	4	3	2	1	0
Read Write	color_depth[3:0]				desired_pr_factor[3:0]			
Reset	0	0	0	0	0	0	0	0

HDMI_VP_PR_CD field descriptions

Field	Description
7–4 color_depth[3:0]	Color depth configuration: other Reserved. Not used. 0000 24 bits per pixel video (8 bit per component). 8-bit packing mode. 0100 24 bits per pixel video (8 bit per component). 8-bit packing mode. 0101 30 bits per pixel video (10 bit per component). 10-bit packing mode. 0110 36 bits per pixel video (12 bit per component). 12-bit packing mode. 0111 48 bits per pixel video (16 bit per component). 16-bit packing mode.
desired_pr_ factor[3:0]	Desired pixel repetition factor configuration. The configured value sets H13T PHY PLL to multiply pixel clock by the factor in order to obtain the desired repetition clock. For the CEA modes some are already defined with pixel repetition in the input video. So for CEA modes this shall be always 0. Shall only be used if the user wants to do pixel repetition using H13TCTRL core. other Reserved. Not used. 0000 No pixel repetition (pixel sent only once) 0001 Pixel sent 2 times (pixel repeated once) 0010 Pixel sent 3 times 0011 Pixel sent 4 times 0100 Pixel sent 5 times 0101 Pixel sent 6 times

Table continues on the next page...

HDMI_VP_PR_CD field descriptions (continued)

Field	Description
	0110 Pixel sent 7 times 0111 Pixel sent 8 times 1000 Pixel sent 9 times 1001 Pixel sent 10 times

33.5.40 Video Packetizer Stuffing and Default Packing Phase Register (HDMI_VP_STUFF)

This register controls the Pixel repetition, pixel packing and YCC422 stuffing.

- Address Offset: 0x0802
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 802h offset = 12_0802h

Bit	7	6	5	4
Read Write	Reserved		idefault_phase	ifix_pp_to_last
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	icx_goto_p0_st	ycc422_stuffing	pp_stuffing	pr_stuffing
Reset	0	0	0	0

HDMI_VP_STUFF field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 idefault_phase	Controls the default phase packing machine used according to: "If the transmitted video format has timing such that the phase of the first pixel of every Video Data Period corresponds to pixel packing phase 0 (for example, 10P0, 12P0, 16P0), the Source may set the Default_Phase bit in the GCP. The Sink may use this bit to optimize it's filtering or handling of the PP field." (HDMI specification version 1.4a) This means that for10 bit mode the Htotal must be dividable by 4 and for 12 bit mode the Htotal must be dividable by 2.
4 ifix_pp_to_last	Reserved. Controls packing machine strategy.
3 icx_goto_p0_st	Reserved. Controls packing machine strategy.
2 ycc422_stuffing	YCC 422 remap stuffing control. For horizontal blanking: 0 YCC 422 remap block in direct mode (input blanking data goes directly to output). 1 YCC 422 remap block in stuffing mode. When "de" goes to low the outputs are fixed to 0x00.

Table continues on the next page...

HDMI_VP_STUFF field descriptions (continued)

Field	Description
1 pp_stuffing	Pixel packing stuffing control 0 Pixel packing block in direct mode (input blanking data goes directly to output). 1 Pixel packing block in stuffing mode. When "de_rep" goes to low the outputs are fixed to 0x00.
0 pr_stuffing	Pixel repeater stuffing control 0 Pixel repeater block in direct mode (input blanking data goes directly to output). 1 Pixel repeater block in stuffing mode. When "de" goes to low the outputs are fixed to 0x00.

33.5.41 Video Packetizer YCC422 Remapping Register (HDMI_VP_REMAP)

This register controls YCC422 remap of the Video Packetizer. For more information about YCC422 remap refer to HDMI 1.4a specification.

- Address Offset: 0x0803
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 803h offset = 12_0803h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						ycc422_size[1:0]	
Reset	0	0	0	0	0	0	0	0

HDMI_VP_REMAP field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
ycc422_size[1:0]	YCC 422 remap input video size: 00 YCC 422 16-bit input video (8 bits per component). 01 YCC 422 20-bit input video (10 bits per component). 10 YCC 422 24-bit input video (12 bits per component). 11 Reserved. Not used.

33.5.42 Video Packetizer Output, Bypass, and Enable Configuration Register (HDMI_VP_CONF)

This register controls the Video Packetizer output selection, bypass select, YCC422 enable, Pixel repeater, and pixel packing enabling.

- Address Offset: 0x0804
- Size: 8 bits
- Value after Reset: 0x46
- Access: Read/Write

Address: 12_0000h base + 804h offset = 12_0804h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	bypass_en	pp_en	pr_en	ycc422_en	BYPASS_ SELECT	output_selector[1:0]	
Reset	0	1	0	0	0	1	1	0

HDMI_VP_CONF field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 bypass_en	Bypass enable. Disabling forces bypass module to output always zeros.
5 pp_en	Pixel packing enable. Disabling forces bypass module to output always zeros.
4 pr_en	Pixel repeater enable. Disabling forces bypass module to output always zeros.
3 ycc422_en	YCC 422 select enable. Disabling forces bypass module to output always zeros.
2 BYPASS_ SELECT	Bypass select bit 0 Data from pixel repeater block. 1 Data from input of video packetizer block.
output_ selector[1:0]	Video packetizer output selection. 00 Data from pixel packing block. 01 Data from YCC 422 remap block. 10 Data from 8-bit bypass block. 11 Data from 8-bit bypass block.

33.5.43 VP_STAT (HDMI_VP_STAT)

This register contains the following active high FIFO status indications:

- Address Offset: 0x0805
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 805h offset = 12_0805h

Bit	7	6	5	4
Read	ostfullrepet	ostemptyrepet	ostfullpp	ostemptypp
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	ostfullremap	ostemptyremap	ostfullbyp	ostemptybyp
Write				
Reset	0	0	0	0

HDMI_VP_STAT field descriptions

Field	Description
7 ostfullrepet	Video packetizer pixel repeater FIFO full status.
6 ostemptyrepet	Video packetizer pixel repeater FIFO empty status.
5 ostfullpp	Video packetizer pixel packing FIFO full status.
4 ostemptypp	Video packetizer pixel packing FIFO empty status.
3 ostfullremap	Video packetizer pixel YCC 422 re-mapper FIFO full status.
2 ostemptyremap	Video packetizer pixel YCC 422 re-mapper FIFO empty status.
1 ostfullbyp	Video packetizer 8-bit bypass FIFO full status.
0 ostemptybyp	Video packetizer 8-bit bypass FIFO empty status.

33.5.44 VP_INT (HDMI_VP_INT)

This register contains the interrupt indication of the VP_STAT status interrupts. Interrupt generation is accomplished in the following way:

interrupt = (mask == 1'b0) && (polarity == status);

All this interrupts are forwarded to the Interrupt Handler sticky bit registers and after ORed to a single main interrupt line to micro controller. Assertion of this interrupt implies that data related with the corresponding packet has been sent through the HDMI interface.

- Address Offset: 0x0806
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 806h offset = 12_0806h

Bit	7	6	5	4
Read	ointfullrepet	ointemptyrepet	ointfullpp	ointemptypp
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	ointfullremap	ointemptyremap	ointfullbyp	ointemptybyp
Write				
Reset	0	0	0	0

HDMI_VP_INT field descriptions

Field	Description
7 ointfullrepet	Video packetizer pixel repeater FIFO full status
6 ointemptyrepet	Video packetizer pixel repeater FIFO empty status
5 ointfullpp	Video packetizer pixel packing FIFO full status
4 ointemptypp	Video packetizer pixel packing FIFO empty status
3 ointfullremap	Video packetizer pixel YCC 422 re-mapper FIFO full status.
2 ointemptyremap	Video packetizer pixel YCC 422 re-mapper FIFO empty status.
1 ointfullbyp	Video packetizer 8-bit bypass FIFO full status.
0 ointemptybyp	Video packetizer 8-bit bypass FIFO empty status.

33.5.45 Video Packetizer Interrupt Mask Register (HDMI_VP_MASK)

Mask register for generation of VP_INT interrupts.

- Address Offset: 0x0807
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 807h offset = 12_0807h

Bit	7	6	5	4	3	2	1	0
Read Write	VPMASK7	VPMASK6	VPMASK5	VPMASK4	VPMASK3	VPMASK2	VPMASK1	VPMASK0
Reset	0	0	0	0	0	0	0	0

HDMI_VP_MASK field descriptions

Field	Description
7 VPMASK7	Mask bit for VP_INT[7] interrupt bit.
6 VPMASK6	Mask bit for VP_INT[6] interrupt bit.
5 VPMASK5	Mask bit for VP_INT[5] interrupt bit.
4 VPMASK4	Mask bit for VP_INT[4] interrupt bit.
3 VPMASK3	Mask bit for VP_INT[3] interrupt bit.
2 VPMASK2	Mask bit for VP_INT[2] interrupt bit.
1 VPMASK1	Mask bit for VP_INT[1] interrupt bit.
0 VPMASK0	Mask bit for VP_INT[0] interrupt bit.

33.5.46 VP_POL (HDMI_VP_POL)

Polarity register for generation of VP_INT interrupts.

- Address Offset: 0x0808
- Size: 8 bits
- Value after Reset: 0xFF
- Access: Read/Write

Address: 12_0000h base + 808h offset = 12_0808h

Bit	7	6	5	4	3	2	1	0
Read Write	VPPOL7	VPPOL6	VPPOL5	VPPOL4	VPPOL3	VPPOL2	VPPOL1	VPPOL0
Reset	1	1	1	1	1	1	1	1

HDMI_VP_POL field descriptions

Field	Description
7 VPPOL7	Polarity bit for VP_INT[7] interrupt bit.
6 VPPOL6	Polarity bit for VP_INT[6] interrupt bit.
5 VPPOL5	Polarity bit for VP_INT[5] interrupt bit.
4 VPPOL4	Polarity bit for VP_INT[4] interrupt bit.
3 VPPOL3	Polarity bit for VP_INT[3] interrupt bit.
2 VPPOL2	Polarity bit for VP_INT[2] interrupt bit.
1 VPPOL1	Polarity bit for VP_INT[1] interrupt bit.
0 VPPOL0	Polarity bit for VP_INT[0] interrupt bit.

33.5.47 Frame Composer Input Video Configuration and HDCP Keepout Register (HDMI_FC_INVIDCONF)

This register configures the Interlaced/progressive, Vblank variation and polarity of all video synchronism of the input video signal.

- Address Offset: 0x1000
- Size: 8 bits
- Value after Reset: 0x70
- Access: Read/Write

Address: 12_0000h base + 1000h offset = 12_1000h

Bit	7	6	5	4
Read Write	Reserved	vsync_in_polarity	hsync_in_polarity	de_in_polarity
Reset	0	1	1	1

Bit	3	2	1	0
Read Write	DVI_mode	Reserved	r_v_blank_in_osc	in_I_P
Reset	0	0	0	0

HDMI_FC_INVIDCONF field descriptions

Field	Description
7 -	This field is reserved. Reserved

Table continues on the next page...

HDMI_FC_INVIDCONF field descriptions (continued)

Field	Description
6 vsync_in_polarity	Vsync input polarity 1 Active high 0 Active low
5 hsync_in_polarity	Hsync input polarity 1 Active high 0 Active low
4 de_in_polarity	Data enable input polarity 1 Active high 0 Active low
3 DVI_mode	Active low 0 DVI mode selected 1 HDMI mode selected
2 -	This field is reserved. Reserved
1 r_v_blank_in_osc	Used for CEA861-D modes with fractional Vblank (for example, modes 5, 6, 7, 10, 11, 20, 21, and 22. For more modes, refer to CEA861-D specification. 1 Active high
0 in_I_P	Input video mode: 1 Interlaced 0 Progressive

33.5.48 Frame Composer Input Video HActive Pixels Register 0 (HDMI_FC_INHACTIV0)

- Address Offset: 0x1001
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1001h offset = 12_1001h

Bit	7	6	5	4	3	2	1	0
Read Write	H_in_activ[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INHACTIV0 field descriptions

Field	Description
H_in_activ[7:0]	Input video Horizontal active pixel region width. Number of Horizontal active pixels [0...8191].

33.5.49 Frame Composer Input Video HActive Pixels Register 1 (HDMI_FC_INHACTIV1)

- Address Offset: 0x1002
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1002h offset = 12_1002h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				H_in_activ[12:8]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INHACTIV1 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
H_in_activ[12:8]	Input video Horizontal active pixel region width. Dependencies: Value after Reset: 0000b • the higher bit of Horizontal active pixels; Number of Horizontal active pixels [0...8191].

33.5.50 Frame Composer Input Video HBlank Pixels Register 0 (HDMI_FC_INHBLANK0)

- Address Offset: 0x1003
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1003h offset = 12_1003h

Bit	7	6	5	4	3	2	1	0
Read Write	H_in_blank[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INHBLANK0 field descriptions

Field	Description
H_in_blank[7:0]	Input video Horizontal blanking pixel region width. Number of Horizontal blanking pixels [0...4095].

33.5.51 Frame Composer Input Video HBlank Pixels Register 1 (HDMI_FC_INHBLANK1)

- Address Offset: 0x1004
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1004h offset = 12_1004h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				H_in_blank[12:8]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INHBLANK1 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
H_in_blank[12:8]	Input video Horizontal blanking pixel region width. Dependencies: Value after Reset: 0000b • the higher bits of Horizontal blanking pixels; Number of Horizontal blanking pixels [0...8191].

33.5.52 Frame Composer Input Video VActive Pixels Register 0 (HDMI_FC_INVACTIV0)

- Address Offset: 0x1005
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1005h offset = 12_1005h

Bit	7	6	5	4	3	2	1	0
Read Write	V_in_activ[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INVACTIV0 field descriptions

Field	Description
V_in_activ[7:0]	Input video Vertical active pixel region width. Number of Vertical active lines [0...4095].

33.5.53 Frame Composer Input Video VActive Pixels Register 1 (HDMI_FC_INVACTIV1)

- Address Offset: 0x1006
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1006h offset = 12_1006h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			V_in_activ[12:8]				
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INVACTIV1 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
V_in_activ[12:8]	Input video Vertical active pixel region width. Dependencies: Value after Reset: 0000b

Table continues on the next page...

HDMI_FC_INVACTIV1 field descriptions (continued)

Field	Description
	• the higher 5 bits of Vertical active line; Number of Vertical active lines [0...8191].

33.5.54 Frame Composer Input Video VBlank Pixels Register (HDMI_FC_INVBLANK)

- Address Offset: 0x1007
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1007h offset = 12_1007h

Bit	7	6	5	4	3	2	1	0
Read Write	V_in_blank[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INVBLANK field descriptions

Field	Description
V_in_blank[7:0]	Input video Vertical blanking pixel region width. Number of Vertical blanking lines [0...255]. Value after Reset: 0x00

33.5.55 Frame Composer Input Video HSync Front Porch Register 0 (HDMI_FC_HSYNCINDELAY0)

- Address Offset: 0x1008
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1008h offset = 12_1008h

Bit	7	6	5	4	3	2	1	0
Read Write	H_in_delay[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_HSYNCINDELAY0 field descriptions

Field	Description
H_in_delay[7:0]	Input video Hsync active edge delay. Integer number of pixel clock cycles from "de" non active edge of the last "de" valid period [0...4095].

33.5.56 Frame Composer Input Video HSync Front Porch Register 1 (HDMI_FC_HSYNCINDELAY1)

- Address Offset: 0x1009
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1009h offset = 12_1009h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				H_in_delay[12:8]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_HSYNCINDELAY1 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
H_in_delay[12:8]	Input video Hsync active edge delay. Dependencies: Value after Reset: 0000b • the higher 5 bits of delay; Integer number of pixel clock cycles from "de" non active edge of the last "de" valid period [0...8191].

33.5.57 Frame Composer Input Video HSync Width Register 0 (HDMI_FC_HSYNCINWIDTH0)

- Address Offset: 0x100A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Ah offset = 12_100Ah

Bit	7	6	5	4	3	2	1	0
Read Write	H_in_width[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_HSYNCINWIDTH0 field descriptions

Field	Description
H_in_width[7:0]	Input video Hsync active pulse width. Integer number of pixel clock cycles [0...511].

33.5.58 Frame Composer Input Video HSync Width Register 1 (HDMI_FC_HSYNCINWIDTH1)

- Address Offset: 0x100B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Bh offset = 12_100Bh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						H_in_width[9:8]	
Reset	0	0	0	0	0	0	0	0

HDMI_FC_HSYNCINWIDTH1 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
H_in_width[9:8]	Input video Hsync active pulse width. Dependencies: Value after Reset after Reset: 0b

Table continues on the next page...

HDMI_FC_HSYNCINWIDTH1 field descriptions (continued)

Field	Description
	• Integer number of pixel clock cycles [0...1024].

33.5.59 Frame Composer Input Video VSync Front Porch Register (HDMI_FC_VSYNCINDELAY)

- Address Offset: 0x100C
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Ch offset = 12_100Ch

Bit	7	6	5	4	3	2	1	0
Read Write	V_in_delay[7..0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSYNCINDELAY field descriptions

Field	Description
V_in_delay[7..0]	Input video Vsync active edge delay. Integer number of Hsync pulses from "de" non active edge of the last "de" valid period. [0...255].

33.5.60 Frame Composer Input Video VSync Width Register (HDMI_FC_VSYNCINWIDTH)

- Address Offset: 0x100D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Dh offset = 12_100Dh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		V_in_width[5..0]					
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSYNCINWIDTH field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
V_in_width[5..0]	Value after Reset: 000000b Input video Vsync active pulse width: Integer number of pixel clock cycles [0...63].

33.5.61 Frame Composer Input Video Refresh Rate Register 0 (HDMI_FC_INFREQ0)

- Address Offset: 0x100E
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Eh offset = 12_100Eh

Bit	7	6	5	4	3	2	1	0
Read Write	infreq[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INFREQ0 field descriptions

Field	Description
infreq[7:0]	Video refresh rate in Hz*1E3 format. This registers are provided for debug and informative purposes. No data is written to this registers by the H13TCTRL and the data here written by software is not used in any way by the H13TCTRL.

33.5.62 Frame Composer Input Video Refresh Rate Register 1 (HDMI_FC_INFREQ1)

- Address Offset: 0x100F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 100Fh offset = 12_100Fh

Bit	7	6	5	4	3	2	1	0
Read Write	infreq[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INFREQ1 field descriptions

Field	Description
infreq[7:0]	Video refresh rate in Hz*1E3 format. This registers are provided for debug and informative purposes. No data is written to this registers by the H13TCTRL and the data here written by software is not used in any way by the H13TCTRL.

33.5.63 Frame Composer Input Video Refresh Rate Register 2 (HDMI_FC_INFREQ2)

- Address Offset: 0x1010
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1010h offset = 12_1010h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				infreq[19:16]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INFREQ2 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
infreq[19:16]	Video refresh rate in Hz*1E3 format. This registers are provided for debug and informative purposes. No data is written to this registers by the H13TCTRL and the data here written by software is not used in any way by the H13TCTRL. Value after Reset: 0000b

33.5.64 Frame Composer Control Period Duration Register (HDMI_FC_CTRLDUR)

- Address Offset: 0x1011
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1011h offset = 12_1011h

Bit	7	6	5	4	3	2	1	0
Read Write	ctrlperiodduration							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_CTRLDUR field descriptions

Field	Description
ctrlperiodduration	Configuration of the control period minimum duration (min. of 12 pixel clock cycles, refer to HDMI 1.4a specification). Integer number of pixel clocks cycles [0..255].

33.5.65 Frame Composer Extended Control Period Duration Register (HDMI_FC_EXCTRLDUR)

- Address Offset: 0x1012
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1012h offset = 12_1012h

Bit	7	6	5	4	3	2	1	0
Read Write	exctrlperiodduration							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_EXCTRLDUR field descriptions

Field	Description
exctrlperiodduration	Configuration of the extended control period minimum duration (min. of 32 pixel clock cycles, see HDMI 1.4a specification). Integer number of pixel clocks cycles [0..255].

33.5.66 Frame Composer Extended Control Period Maximum Spacing Register (HDMI_FC_EXCTRLSPAC)

- Address Offset: 0x1013
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1013h offset = 12_1013h

Bit	7	6	5	4	3	2	1	0
Read Write	exctrlperiodspacing							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_EXCTRLSPAC field descriptions

Field	Description
exctrlperiodspacing	Configuration of the maximum spacing between consecutive extended control periods (max of 50msec, see HDMI 1.4a specification): generated spacing = (1/freq tmds clock)*256*256*extctrlperiodspacing

33.5.67 Frame Composer Channel 0 Non-Preamble Data Register (HDMI_FC_CH0PREAM)

- Address Offset: 0x1014
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1014h offset = 12_1014h

Bit	7	6	5	4	3	2	1	0
Read Write	ch0_preamble_filter							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_CH0PREAM field descriptions

Field	Description
ch0_preamble_filter	When in control mode, configures 8-bits that are going to fill the channel 0 data lines not used to transmit the preamble (for more clarifications refer to HDMI 1.4a specification).

33.5.68 Frame Composer Channel 1 Non-Preamble Data Register (HDMI_FC_CH1PREAM)

- Address Offset: 0x1015
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1015h offset = 12_1015h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				ch1_preamble_filter			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_CH1PREAM field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
ch1_preamble_filter	When in control mode, configures 6-bits that are going to fill the channel 1 data lines not used to transmit the preamble (for more clarifications refer to HDMI 1.4a specification).

33.5.69 Frame Composer Channel 2 Non-Preamble Data Register (HDMI_FC_CH2PREAM)

- Address Offset: 0x1016
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1016h offset = 12_1016h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				ch2_preamble_filter			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_CH2PREAM field descriptions

Field	Description
7–6 -	This field is reserved. Reserved

Table continues on the next page...

HDMI_FC_CH2PREAM field descriptions (continued)

Field	Description
ch2_preamble_ filter	When in control mode, configures 6-bits that are going to fill the channel 2 data lines not used to transmit the preamble (for more clarifications, see HDMI 1.4a specification).

33.5.70 Frame Composer AVI Configuration Register 3 (HDMI_FC_AVICONF3)

- Address Offset: 0x1017
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

configuration of Quantization range and IT content type.

Address: 12_0000h base + 1017h offset = 12_1017h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				YQ1_YQ0_YCC		CN1_CN0	
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVICONF3 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
3–2 YQ1_YQ0_YCC	Quantization range according to CEA specification.
CN1_CN0	IT content type according to CEA specification

33.5.71 Frame Composer GCP Packet Configuration Register (HDMI_FC_GCP)

Configures the General Control Packet A/V mute indicators and the default phase.

- Address Offset: 0x1018
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1018h offset = 12_1018h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved					default_ phase	set_avmute	clear_ avmute
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GCP field descriptions

Field	Description
7–3 -	This field is reserved. Reserved
2 default_phase	Value of "default_phase" in the GCP packet. This data should be equal to the default phase used at Video packetizer packing machine. Value after Reset: 0b
1 set_avmute	Value of "set_avmute" in the GCP packet. Value after Reset: 0b
0 clear_avmute	Value of "clear_avmute" in the GCP packet. Value after Reset: 0b

33.5.72 Frame Composer AVI Packet Configuration Register 0 (HDMI_FC_AVICONF0)

Configures the following contents of the AVI infoFrame:

- RGB/YCC indication
- Bar information
- Scan information
- Active format present
- Progressive/Interlaced indicator
- Active aspect ratio
- Picture aspect ratio
- Colorimetry
- IT content
- Extended colorimetry
- Quantization range
- Non-uniform picture scaling

For more information, refer to HDMI 1.4a and CEA - 861D specifications.

- Address Offset: 0x1019
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1019h offset = 12_1019h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVICONF0_MISC	FC_AVICONF0_ACTIVE_FORMAT	FC_AVICONF0_SCAN		FC_AVICONF0_BAR		FC_AVICONF0_RGB_YCC	
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVICONF0 field descriptions

Field	Description
7 FC_AVICONF0_MISC	Frame composer AVI packet configuration bit
6 FC_AVICONF0_ACTIVE_FORMAT	Active format present
5–4 FC_AVICONF0_SCAN	Scan information
3–2 FC_AVICONF0_BAR	Bar information
FC_AVICONF0_RGB_YCC	RGB/YCC indication Value after Reset: 0b

33.5.73 Frame Composer AVI Packet Configuration Register 1 (HDMI_FC_AVICONF1)

- Address Offset: 0x101A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Ah offset = 12_101Ah

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVICONF0_COLOR		FC_AVICONF1_PICTURE_AR		FC_AVICONF1_ACTIVE_AR			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVICONF1 field descriptions

Field	Description
7–6 FC_AVICONF0_COLOR	Colorimetry
5–4 FC_AVICONF1_PICTURE_AR	Picture aspect ratio
FC_AVICONF1_ACTIVE_AR	Active aspect ratio Value after Reset: 0b

33.5.74 FC_AVICONFFrame Composer AVI Packet Configuration Register 2 (HDMI_FC_AVICONF2)

- Address Offset: 0x101B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Bh offset = 12_101Bh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVICONF2_IT	FC_AVICONF2_EXT_COLOR			-		FC_AVICONF2_SCALE	
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVICONF2 field descriptions

Field	Description
7 FC_AVICONF2_IT	IT content
6–4 FC_AVICONF2_EXT_COLOR	Extended colorimetry
3–2 -	Quantization range
FC_AVICONF2_SCALE	Non-uniform picture scaling Value after Reset: 0b

33.5.75 Frame Composer AVI Packet VIC Register (HDMI_FC_AVIVID)

Configures the AVI infoFrame Video Identification code. For more information, refer to the CEA-861-E specification.

- Address Offset: 0x101C
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Ch offset = 12_101Ch

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVIVID							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVIVID field descriptions

Field	Description
FC_AVIVID	the AVI infoFrame Video Identification code.

33.5.76 Frame Composer AVI Packet End of Top Bar Register 0 (HDMI_FC_AVIETB0)

These registers define the AVI infoFrame End of Top Bar value. For more information, refer to CEA-861-E specification.

- Address Offset: 0x101D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Dh offset = 12_101Dh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVIETB0 field descriptions

Field	Description
-	Line number of end of top bar (lower 8 bits)

33.5.77 Frame Composer AVI Packet End of Top Bar Register 1 (HDMI_FC_AVIETB1)

- Address Offset: 0x101E
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Eh offset = 12_101Eh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVIETB1 field descriptions

Field	Description
-	Line number of end of top bar (upper 8 bits)

33.5.78 Frame Composer AVI Packet Start of Bottom Bar Register 0 (HDMI_FC_AVISBB0)

These registers define the AVI infoFrame Start of Bottom Bar value. For more information, refer to CEA-861D specification.

- Address Offset: 0x101F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 101Fh offset = 12_101Fh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVISBB0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVISBB0 field descriptions

Field	Description
FC_AVISBB0	Line number of Start of Bottom Bar (lower 8 bits)

33.5.79 Frame Composer AVI Packet Start of Bottom Bar Register 1 (HDMI_FC_AVISBB1)

- Address Offset: 0x1020
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1020h offset = 12_1020h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVISBB1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVISBB1 field descriptions

Field	Description
FC_AVISBB1	Line number of Start of Bottom Bar (upper 8 bits)

33.5.80 Frame Composer AVI Packet End of Left Bar Register 0 (HDMI_FC_AVIELB0)

These registers define the AVI infoFrame End of Left Bar value. For more information, refer to CEA-861D specification.

- Address Offset: 0x1021
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1021h offset = 12_1021h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVIELB0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVIELB0 field descriptions

Field	Description
FC_AVIELB0	Pixel number of end of left Bar (lower 8 bits)

33.5.81 Frame Composer AVI Packet End of Left Bar Register 1 (HDMI_FC_AVIELB1)

- Address Offset: 0x1022
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1022h offset = 12_1022h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVIELB1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVIELB1 field descriptions

Field	Description
FC_AVIELB1	Pixel number of end of left Bar (lower 8 bits)

33.5.82 Frame Composer AVI Packet Start of Right Bar Register 0 (HDMI_FC_AVISRB0)

These registers define the AVI infoFrame Start of Right Bar value. For more information, refer to CEA-861D specification.

- Address Offset: 0x1023
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1023h offset = 12_1023h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVISRB0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVISRB0 field descriptions

Field	Description
FC_AVISRB0	Pixel number of start of right Bar (lower 8 bits)

33.5.83 Frame Composer AVI Packet Start of Right Bar Register 1 (HDMI_FC_AVISRB1)

- Address Offset: 0x1024
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1024h offset = 12_1024h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_AVISRB1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AVISRB1 field descriptions

Field	Description
FC_AVISRB1	Pixel number of start of right Bar (upper 8 bits)

33.5.84 Frame Composer AUD Packet Configuration Register 0 (HDMI_FC_AUDICONF0)

These registers configure the following contents of the AUDIO infoFrame:

- Coding type
- Channel count
- Sampling frequency
- Sampling size
- Channel allocation
- Audio level shift value
- Down mix enable

For more information, refer to CEA-861D specification.

- Address Offset: 0x1025 to 0x1028
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For the FC_AUDICONF0 register, bits [6:5] correspond to LFEPBL1, LFEPBL0 LFE playback level as compared to the other channels (from HDMI 1.4a specification).

Address: 12_0000h base + 1025h offset = 12_1025h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	CC[2:0]			CT[3:0]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDICONF0 field descriptions

Field	Description
7 -	This field is reserved. Reserved
6–4 CC[2:0]	Channel count
CT[3:0]	Coding Type

33.5.85 Frame Composer AUD Packet Configuration Register 1 (HDMI_FC_AUDICONF1)

Address: 12_0000h base + 1026h offset = 12_1026h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		SS[1:0]		Reserved	SF[2:0]		
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDICONF1 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5–4 SS[1:0]	Sampling size
3 -	This field is reserved. Reserved
SF[2:0]	Sampling frequency

33.5.86 Frame Composer AUD Packet Configuration Register 2 (HDMI_FC_AUDICONF2)

Address: 12_0000h base + 1027h offset = 12_1027h

Bit	7	6	5	4	3	2	1	0
Read Write	CA[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDICONF2 field descriptions

Field	Description
CA[7:0]	Channel allocation

33.5.87 Frame Composer AUD Packet Configuration Register 3 (HDMI_FC_AUDICONF3)

Address: 12_0000h base + 1028h offset = 12_1028h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	LFEPBL[1:0]		DM_INH	LSV[3:0]			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDICONF3 field descriptions

Field	Description
7 -	This field is reserved. Reserved
6–5 LFEPBL[1:0]	LFE playback information
4 DM_INH	Down mix enable
LSV[3:0]	Level shift value (for down mixing)

33.5.88 Frame Composer VSI Packet Data IEEE Register 0 (HDMI_FC_VSDIEEEID0)

These registers configure the Vendor Specific infoFrame IEEE registration identifier. For more information, refer to CEA-861D specification.

- Address Offset: 0x1029
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1029h offset = 12_1029h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDIEEEID0 field descriptions

Field	Description
-	the Vendor Specific infoFrame IEEE registration identifier byte 0

33.5.89 Frame Composer VSI Packet Data Size Register (HDMI_FC_VSDSIZE)

- Address Offset: 0x102A
- Size: 8 bits
- Value after Reset: 0x1B
- Access: Read/Write

configuration of Packet size.

Address: 12_0000h base + 102Ah offset = 12_102Ah

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			VSDSIZE				
Reset	0	0	0	1	1	0	1	1

HDMI_FC_VSDSIZE field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
VSDSIZE	Packet size as described in HDMI Vendor Specific InfoFrame (from HDMI specification).

33.5.90 Frame Composer VSI Packet Data IEEE Register 1 (HDMI_FC_VSDIEEEID1)

- Address Offset: 0x102a
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1030h offset = 12_1030h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDIEEEID1 field descriptions

Field	Description
-	the Vendor Specific infoFrame IEEE registration identifier byte 1

33.5.91 Frame Composer VSI Packet Data IEEE Register 2 (HDMI_FC_VSDIEEEID2)

- Address Offset: 0x102b
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1031h offset = 12_1031h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDIEEEID2 field descriptions

Field	Description
-	the Vendor Specific infoFrame IEEE registration identifier byte 2

33.5.92 Frame Composer VSI Packet Data IEEE Register 0 (HDMI_FC_VSDPAYLOAD0)

These registers configure the Vendor Specific infoFrame 24 bytes specific payload. For more information, refer to CEA-861D specification.

- Address Offset: 0x1032
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1032h offset = 12_1032h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD0 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte0

33.5.93 Frame Composer VSI Packet Data IEEE Register 1 (HDMI_FC_VSDPAYLOAD1)

- Address Offset: 0x1033
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1033h offset = 12_1033h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD1 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte1

33.5.94 Frame Composer VSI Packet Data IEEE Register 2 (HDMI_FC_VSDPAYLOAD2)

- Address Offset: 0x1034
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1034h offset = 12_1034h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD2 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte2

33.5.95 Frame Composer VSI Packet Data IEEE Register 3 (HDMI_FC_VSDPAYLOAD3)

- Address Offset: 0x1035
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1035h offset = 12_1035h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD3 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte3

33.5.96 Frame Composer VSI Packet Data IEEE Register 4 (HDMI_FC_VSDPAYLOAD4)

- Address Offset: 0x1036
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1036h offset = 12_1036h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD4 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte4

33.5.97 Frame Composer VSI Packet Data IEEE Register 5 (HDMI_FC_VSDPAYLOAD5)

- Address Offset: 0x1037
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1037h offset = 12_1037h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD5 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte5

33.5.98 Frame Composer VSI Packet Data IEEE Register 6 (HDMI_FC_VSDPAYLOAD6)

- Address Offset: 0x1038
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1038h offset = 12_1038h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD6 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte6

33.5.99 Frame Composer VSI Packet Data IEEE Register 7 (HDMI_FC_VSDPAYLOAD7)

- Address Offset: 0x1039
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1039h offset = 12_1039h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD7 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte7

33.5.100 Frame Composer VSI Packet Data IEEE Register 8 (HDMI_FC_VSDPAYLOAD8)

- Address Offset: 0x103a
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Ah offset = 12_103Ah

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD8 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte8

33.5.101 Frame Composer VSI Packet Data IEEE Register 9 (HDMI_FC_VSDPAYLOAD9)

- Address Offset: 0x103b
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Bh offset = 12_103Bh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD9 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte9

33.5.102 Frame Composer VSI Packet Data IEEE Register 10 (HDMI_FC_VSDPAYLOAD10)

- Address Offset: 0x103c
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Ch offset = 12_103Ch

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD10 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte10

33.5.103 Frame Composer VSI Packet Data IEEE Register 11 (HDMI_FC_VSDPAYLOAD11)

- Address Offset: 0x103d
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Dh offset = 12_103Dh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD11 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte11

33.5.104 Frame Composer VSI Packet Data IEEE Register 12 (HDMI_FC_VSDPAYLOAD12)

- Address Offset: 0x103e
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Eh offset = 12_103Eh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD12 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte12

33.5.105 Frame Composer VSI Packet Data IEEE Register 13 (HDMI_FC_VSDPAYLOAD13)

- Address Offset: 0x103f
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 103Fh offset = 12_103Fh

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD13 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte13

33.5.106 Frame Composer VSI Packet Data IEEE Register 14 (HDMI_FC_VSDPAYLOAD14)

- Address Offset: 0x1040
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1040h offset = 12_1040h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD14 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte14

33.5.107 Frame Composer VSI Packet Data IEEE Register 15 (HDMI_FC_VSDPAYLOAD15)

- Address Offset: 0x1041
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1041h offset = 12_1041h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD15 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte15

33.5.108 Frame Composer VSI Packet Data IEEE Register 16 (HDMI_FC_VSDPAYLOAD16)

- Address Offset: 0x1042
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1042h offset = 12_1042h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD16 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte16

33.5.109 Frame Composer VSI Packet Data IEEE Register 17 (HDMI_FC_VSDPAYLOAD17)

- Address Offset: 0x1043
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1043h offset = 12_1043h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD17 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte17

33.5.110 Frame Composer VSI Packet Data IEEE Register 18 (HDMI_FC_VSDPAYLOAD18)

- Address Offset: 0x1044
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1044h offset = 12_1044h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD18 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte18

33.5.111 Frame Composer VSI Packet Data IEEE Register 19 (HDMI_FC_VSDPAYLOAD19)

- Address Offset: 0x1045
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1045h offset = 12_1045h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD19 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte19

33.5.112 Frame Composer VSI Packet Data IEEE Register 20 (HDMI_FC_VSDPAYLOAD20)

- Address Offset: 0x1046
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1046h offset = 12_1046h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD20 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte20

33.5.113 Frame Composer VSI Packet Data IEEE Register 21 (HDMI_FC_VSDPAYLOAD21)

- Address Offset: 0x1047
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1047h offset = 12_1047h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD21 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte21

33.5.114 Frame Composer VSI Packet Data IEEE Register 22 (HDMI_FC_VSDPAYLOAD22)

- Address Offset: 0x1048
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1048h offset = 12_1048h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD22 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte22

33.5.115 Frame Composer VSI Packet Data IEEE Register 23 (HDMI_FC_VSDPAYLOAD23)

- Address Offset: 0x1049
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1049h offset = 12_1049h

Bit	7	6	5	4	3	2	1	0
Read Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_VSDPAYLOAD23 field descriptions

Field	Description
-	the Vendor Specific infoFrame 24 bytes specific payload byte23

33.5.116 Frame Composer SPD Packet Data Vendor Name Register 0 (HDMI_FC_SPDVENDORNAME0)

These registers configure the Source Product Descriptor infoFrame 8 bytes Vendor name. For more information, refer to CEA-861D specification.

- Address Offset: 0x104A to 0x1051
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 104Ah offset = 12_104Ah

Bit	7	6	5	4	3	2	1	0
Read Write	vendor_name							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_SPDVENDORNAME0 field descriptions

Field	Description
vendor_name	Vendor name

33.5.117 Frame Composer SPD Packet Data Product Name Register 0 (HDMI_FC_SPDPRODUCTNAME0)

These registers configure the Source Product Descriptor infoFrame 16 bytes Product name. For more information, refer to CEA-861D specification.

- Address Offset: 0x1052 to 0x1061
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1052h offset = 12_1052h

Bit	7	6	5	4	3	2	1	0
Read Write	product_name							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_SPDPRODUCTNAME0 field descriptions

Field	Description
product_name	Product name

33.5.118 Frame Composer SPD Packet Data Source Product Descriptor Register (HDMI_FC_SPDDEVICEINF)

This register configures Source Product Descriptor infoFrame description device field. For more information, refer to CEA-861D specification.

- Address Offset: 0x1062
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1062h offset = 12_1062h

Bit	7	6	5	4	3	2	1	0
Read Write	product_descriptor							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_SPDDEVICEINF field descriptions

Field	Description
product_ descriptor	Product descriptor

33.5.119 Frame Composer Audio Sample Flat and Layout Configuration Register (HDMI_FC_AUDSCONF)

Configures the Audio sample packet sample flat and layout configuration. For more information, refer to HDMI 1.4a specification.

- Address Offset: 0x1063
- Size: 8 bits
- Value after Reset: 0x00
- AEccess: Read/Write

Address: 12_0000h base + 1063h offset = 12_1063h

Bit	7	6	5	4	3	2	1	0
Read Write	aud_packet_sampfit[3:0]				Reserved			aud_packet_layout
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDSCONF field descriptions

Field	Description
7–4 aud_packet_sampfit[3:0]	Set the audio packet sample flat value to be sent on the packet.
3–1 -	This field is reserved. Reserved
0 aud_packet_layout	Set the audio packet layout to be sent in the packet: 1 layout 1 0 layout 0

33.5.120 Frame Composer Audio Packet Sample Present Status Register (HDMI_FC_AUDSSTAT)

Shows the data sample present indication of the last Audio sample packet sent by the HDMI TX Controller. For more information, refer to HDMI 1.4a specification.

- Address Offset: 0x1064
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 1064h offset = 12_1064h

Bit	7	6	5	4	3	2	1	0
Read	Reserved				packet_sampprs[3:0]			
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_FC_AUDSSTAT field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
packet_ sampprs[3:0]	Shows the data sample present indication of the last Audio sample packet sent by the HDMI TX Controller. This register information is at tmds clock rate.

33.5.121 Frame Composer Number of High Priority Packets Attended Configuration Register (HDMI_FC_CTRLQHIGH)

- Address Offset: 0x1073
- Size: 8 bits
- Value after Reset: 0x0F
- Access: Read/Write

Address: 12_0000h base + 1073h offset = 12_1073h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				onhighattended[4:0]			
Reset	0	0	0	0	1	1	1	1

HDMI_FC_CTRLQHIGH field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
onhighattended[4:0]	Configures the number of high priority packets or audio sample packets consecutively attended before checking low priority queue status. Integer number [0..31]

33.5.122 Frame Composer Number of Low Priority Packets Attended Configuration Register (HDMI_FC_CTRLQLOW)

- Address Offset: 0x1074
- Size: 8 bits
- Value after Reset: 0x03
- Access: Read/Write

Address: 12_0000h base + 1074h offset = 12_1074h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				onlowattended[4:0]			
Reset	0	0	0	0	0	0	1	1

HDMI_FC_CTRLQLOW field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
onlowattended[4:0]	Configures the number of low priority packets or null packets consecutively attended before checking high priority queue status or audio sample availability. Integer number [0..31]

33.5.123 Frame Composer ACP Packet Type Configuration Register 0 (HDMI_FC_ACP0)

Configures the following contents of the ACP packet. For more information, refer to the HDMI 1.4 specification.

- Address Offset: 0x1075
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1075h offset = 12_1075h

Bit	7	6	5	4	3	2	1	0
Read Write	acptype[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_ACP0 field descriptions

Field	Description
acptype[7:0]	Configures the ACP packet type.

33.5.124 Frame Composer ACP Packet Type Configuration Register 1 (HDMI_FC_ACP1)

Configures the following contents of the Audio Content Packet (ACP) body:

- Address Offset: 0x1091 to 0x1082
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1091h offset = 12_1091h

Bit	7	6	5	4	3	2	1	0
Read Write	Audio_contentpacket							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_ACP1 field descriptions

Field	Description
Audio_ contentpacket	Audio content packet

33.5.125 FC_ISCR1_Frame Composer Packet Status, Valid, and Continue Configuration Register (HDMI_FC_ISCR1_0)

Configures the following contents of the ISRC1 packet:

- Address Offset: 0x1092
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, see the HDMI 1.4 specification.

Address: 12_0000h base + 1092h offset = 12_1092h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			isrc_status[2:0]			isrc_valid	isrc_cont
Reset	0	0	0	0	0	0	0	0

HDMI_FC_ISCR1_0 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4–2 isrc_status[2:0]	Status of ISRC1.
1 isrc_valid	Valid of ISRC1.
0 isrc_cont	Indication of ISRC2.

33.5.126 Frame Composer ISCR1 Packet Body Register 1 (HDMI_FC_ISCR1_1)

Configures the following contents of the ISRC1 packet:

- ISRC1 packet body
- Address Offset: 0x10A2 to 0x1093
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, see the HDMI 1.4 specification.

Address: 12_0000h base + 1093h offset = 12_1093h

Bit	7	6	5	4	3	2	1	0
Read Write	isrc1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_ISCR1_1 field descriptions

Field	Description
isrc1	Configures the contents of the ISRC1 packet:

33.5.127 Frame Composer ISCR2 Packet Body Register 0 (HDMI_FC_ISCR2_0)

Configures the following contents of the ISRC2 packet:

- ISRC2 packet body
- Address Offset: 0x10B2 to 0x10A3
- Size: 8 bits

- Value after Reset: 0x00
- Access: Read/Write

For more information, see the HDMI 1.4 specification.

Address: 12_0000h base + 10A3h offset = 12_10A3h

Bit	7	6	5	4	3	2	1	0
Read Write	isrc2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_ISCR2_0 field descriptions

Field	Description
isrc2	Configures the contents of the ISRC1 packet:

33.5.128 Frame Composer Data Island Auto Packet Scheduling Register 0 (HDMI_FC_DATAUTO0)

Configures the Frame Composer RDRB(1)/Manual(0) data island packet insertion for SPD, VSD, ISRC2, ISRC1 and ACP packets. On RDRB mode the described packet scheduling is controlled by registers FC_DATAUTO1 and FC_DATAUTO2, while in Manual mode register FC_DATMAN requests to FC the insertion of the requested packet.

- Address Offset: 0x10B3
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10B3h offset = 12_10B3h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			spd_auto	vsd_auto	iscr2_auto	iscr1_auto	acp_auto
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DATAUTO0 field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4 spd_auto	Enables SPD automatic packet scheduling
3 vsd_auto	Enables VSD automatic packet scheduling

Table continues on the next page...

HDMI_FC_DATAUTO0 field descriptions (continued)

Field	Description
2 iscr2_auto	Enables ISRC2 automatic packet scheduling
1 iscr1_auto	Enables ISRC1 automatic packet scheduling
0 acp_auto	Enables ACP automatic packet scheduling

33.5.129 Frame Composer Data Island Auto Packet Scheduling Register 1 (HDMI_FC_DATAUTO1)

Configures the Frame Composer (FC) RDRB frame interpolation for SPD, VSD, ISRC2, ISRC1 and ACP packet insertion on data island when FC is on RDRB mode for the listed packets.

- Address Offset: 0x10B4
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10B4h offset = 12_10B4h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				AUTO_FRAME_INTERPOLATION			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DATAUTO1 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
AUTO_FRAME_ INTERPOLATION	Packet frame interpolation, for automatic packet scheduling

33.5.130 Frame Composer Data Island Auto Packet Scheduling Register 2 (HDMI_FC_DATAUTO2)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for SPD, VSD, ISRC2, ISRC1 and ACP packet insertion on data island when FC is on RDRB mode for the listed packets.

- Address Offset: 0x10B5

- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10B5h offset = 12_10B5h

Bit	7	6	5	4	3	2	1	0
Read Write	AUTO_FRAME_PACKETS				AUTO_LINE_SPACING			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DATAUTO2 field descriptions

Field	Description
7–4 AUTO_FRAME_ PACKETS	Packets per frame, for automatic packet scheduling
AUTO_LINE_ SPACING	Packets line spacing, for automatic packet scheduling

33.5.131 Frame Composer Data Island Manual Packet Request Register (HDMI_FC_DATMAN)

Requests to the Frame Composer the data island packet insertion for NULL, SPD, VSD, ISRC2, ISRC1 and ACP packets when FC_DATAUTO0 bit is in manual mode for the packet requested.

- Address Offset: 0x10B6
- Size: 8 bits
- Value after Reset: 0x00
- Access: Write

Address: 12_0000h base + 10B6h offset = 12_10B6h

Bit	7	6	5	4	3	2	1	0
Read	Reserved							
Write			null_tx	spd_tx	vsd_tx	iscr2_tx	isr1_tx	acp_tx
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DATMAN field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 null_tx	Null packet

Table continues on the next page...

HDMI_FC_DATMAN field descriptions (continued)

Field	Description
4 spd_tx	SPD packet
3 vsd_tx	VSD packet
2 iscr2_tx	ISRC2 packet
1 isr1_tx	ISRC1 packet
0 acp_tx	ACP packet

33.5.132 Frame Composer Data Island Auto Packet Scheduling Register 3 (HDMI_FC_DATAUTO3)

Configures the Frame Composer Automatic(1)/RDRB(0) data island packet insertion for AVI, GCP, AUDI and ACR packets. In Automatic mode, the packet will be inserted on Vblanking when first line with active Vsync appears.

- Address Offset: 0x10B7
- Size: 8 bits
- Value after Reset: 0x0F
- Access: Read/Write

Address: 12_0000h base + 10B7h offset = 12_10B7h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				avi_auto	gcp_auto	audi_auto	acr_auto
Reset	0	0	0	0	1	1	1	1

HDMI_FC_DATAUTO3 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
3 avi_auto	Enable AVI packet insertion
2 gcp_auto	Enable GCP packet insertion
1 audi_auto	Enable AUDI packet insertion
0 acr_auto	Enable ACR packet insertion

33.5.133 Frame Composer Round Robin ACR Packet Insertion Register 0 (HDMI_FC_RDRB0)

Configures the Frame Composer (FC) RDRB frame interpolation for ACR packet insertion on data island when FC is on RDRB mode for this packet.

- Address Offset: 0x10B8
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10B8h offset = 12_10B8h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				ACRframeinterpolation			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB0 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
ACRframeinterpolation	ACR frame interpolation

33.5.134 Frame Composer Round Robin ACR Packet Insertion Register 1 (HDMI_FC_RDRB1)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for the ACR packet insertion on data island when FC is on RDRB mode this packet.

- Address Offset: 0x10B9
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10B9h offset = 12_10B9h

Bit	7	6	5	4	3	2	1	0
Read Write	ACRpacketsinframe				ACRpacketlinespacing			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB1 field descriptions

Field	Description
7–4 ACRpacketsinframe	ACR packets in frame
ACRpacketlinespacing	ACR packet line spacing

33.5.135 Frame Composer Round Robin ACR Packet Insertion Register 2 (HDMI_FC_RDRB2)

Configures the Frame Composer (FC) RDRB frame interpolation for AUDI packet insertion on data island when FC is on RDRB mode for this packet.

- Address Offset: 0x10BA
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BAh offset = 12_10BAh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				AUDIframeinterpolation			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB2 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
AUDIframeinterpolation	Audio frame interpolation

33.5.136 Frame Composer Round Robin ACR Packet Insertion Register 3 (HDMI_FC_RDRB3)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for the AUDI packet insertion on data island when FC is on RDRB mode this packet.

- Address Offset: 0x10BB
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BBh offset = 12_10BBh

Bit	7	6	5	4	3	2	1	0
Read Write	AUDIpacketsinframe				AUDIpacketlinespacing			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB3 field descriptions

Field	Description
7–4 AUDIpacketsinframe	Audio packets per frame
AUDIpacketlinespacing	Audio packets line spacing

33.5.137 Frame Composer Round Robin ACR Packet Insertion Register 4 (HDMI_FC_RDRB4)

Configures the Frame Composer (FC) RDRB frame interpolation for GCP packet insertion on data island when FC is on RDRB mode for this packet.

- Address Offset: 0x10BC
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BCh offset = 12_10BCh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				GCPframeinterpolation			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB4 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
GCPframeinterpolation	GCP packets line spacing

33.5.138 Frame Composer Round Robin ACR Packet Insertion Register 5 (HDMI_FC_RDRB5)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for the GCP packet insertion on data island when FC is on RDRB mode this packet.

- Address Offset: 0x10BD

- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BDh offset = 12_10BDh

Bit	7	6	5	4	3	2	1	0
Read Write	GCPpacketsinframe				GCPpacketlinespacing			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB5 field descriptions

Field	Description
7–4 GCPpacketsinframe	GCP packets per frame
GCPpacketlinespacing	GCP packets line spacing

33.5.139 Frame Composer Round Robin ACR Packet Insertion Register 6 (HDMI_FC_RDRB6)

Configures the Frame Composer (FC) RDRB frame interpolation for AVI packet insertion on data island when FC is on RDRB mode for this packet.

- Address Offset: 0x10BE
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BEh offset = 12_10BEh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				AVIframeinterpolation			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB6 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
AVIframeinterpolation	GCP packets line spacing

33.5.140 Frame Composer Round Robin ACR Packet Insertion Register 7 (HDMI_FC_RDRB7)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for the AVI packet insertion on data island when FC is on RDRB mode this packet.

- Address Offset: 0x10BF
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10BFh offset = 12_10BFh

Bit	7	6	5	4	3	2	1	0
Read Write	AVIpacketsinframe				AVIpacketlinespacing			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_RDRB7 field descriptions

Field	Description
7–4 AVIpacketsinframe	AVI packets per frame
AVIpacketlinespacing	AVI packets line spacing

33.5.141 FC_STAT0 (HDMI_FC_STAT0)

Configures the Frame Composer (FC) RDRB line interpolation and number of packets in frame for the AVI packet insertion on data island when FC is on RDRB mode this packet.

- Address Offset: 0x10D0
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 10D0h offset = 12_10D0h

Bit	7	6	5	4	3	2	1	0
Read	AUDI	ACP	HBR	Reserved		AUDS	ACR	NULL
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_FC_STAT0 field descriptions

Field	Description
7 AUDI	Status bit Active after successful transmission of an Audio InfoFrame packet.
6 ACP	Status bit. Active after successful transmission of an Audio Content Protection Packet.
5 HBR	Status bit. Active after successful transmission of an Audio HBR packet
4–3 -	This field is reserved. Reserved
2 AUDS	Status bit Active after successful transmission of an Audio Sample packet.
1 ACR	Status bit Active after successful transmission of an Audio Clock Regeneration (N/CTS transmission) packet.
0 NULL	Status bit Active after successful transmission of an Null packet.

33.5.142 FC_INT0 (HDMI_FC_INT0)

This register contains the interrupt indication of the FC_STAT0 status interrupts. Interrupt generation is accomplished in the following way:

interrupt = (mask == 1'b0) && (polarity == status);

All this interrupts are forwarded to the Interrupt Handler sticky bit registers and after ORed to a single main interrupt line to micro controller. Assertion of this interrupt implies that data related with the corresponding packet has been sent through the HDMI interface.

- Address Offset: 0x10D1
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D1h offset = 12_10D1h

Bit	7	6	5	4	3	2	1	0
Read Write	AUDI	ACP	HBR	Reserved		AUDS	ACR	NULL
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INT0 field descriptions

Field	Description
7 AUDI	Interrupt indication bit

Table continues on the next page...

HDMI_FC_INT0 field descriptions (continued)

Field	Description
	Active after successful transmission of an Audio InfoFrame packet interrupt.
6 ACP	Interrupt indication bit Active after successful transmission of an Audio Content Protection packet interrupt.
5 HBR	Interrupt indication bit Active after successful transmission of a Audio HBR packet interrupt.
4–3 -	This field is reserved. Reserved
2 AUDS	Interrupt indication bit Active after successful transmission of an Audio Sample packet interrupt.
1 ACR	Interrupt indication bit Active after successful transmission of an Audio Clock Regeneration (N/CTS transmission) packet interrupt.
0 NULL	Interrupt indication bit Active after successful transmission of an Null packet interrupt.

33.5.143 Frame Composer Packet Interrupt Mask Register 0 (HDMI_FC_MASK0)

Mask register for generation of FC_INT0 interrupts.

- Address Offset: 0x10D2
- Size: 8 bits
- Value after Reset: 0x25
- Access: Read/Write

Address: 12_0000h base + 10D2h offset = 12_10D2h

Bit	7	6	5	4	3	2	1	0
Read Write	AUDI	ACP	HBR	Reserved		AUDS	ACR	NULL
Reset	0	0	1	0	0	1	0	1

HDMI_FC_MASK0 field descriptions

Field	Description
7 AUDI	Mask bit for FC_INT0.AUDI interrupt bit Value after Reset: 0b
6 ACP	Mask bit for FC_INT0.ACP interrupt bit Value after Reset: 0b
5 HBR	Mask bit for FC_INT0.HBR interrupt bit Value after Reset: 0b

Table continues on the next page...

HDMI_FC_MASK0 field descriptions (continued)

Field	Description
4–3 -	This field is reserved. Reserved
2 AUDS	Mask bit for FC_INT0.AUDS interrupt bit Value after Reset: 0b
1 ACR	Mask bit for FC_INT0.ACR interrupt bit Value after Reset: 0b
0 NULL	Mask bit for FC_INT0.NULL interrupt bit Value after Reset: 0b

33.5.144 FC_POL0 (HDMI_FC_POL0)

Polarity register for generation of FC_INT0 interrupts.

- Address Offset: 0x10D3
- Size: 8 bits
- Value after Reset: 0xFF
- Access: Read/Write

Address: 12_0000h base + 10D3h offset = 12_10D3h

Bit	7	6	5	4	3	2	1	0
Read Write	AUDI	ACP	HBR	Reserved		AUDS	ACR	NULL
Reset	1	1	1	1	1	1	1	1

HDMI_FC_POL0 field descriptions

Field	Description
7 AUDI	Polarity bit for FC_INT0.AUDI interrupt bit Value after Reset: 0b
6 ACP	Polarity bit for FC_INT0.ACP interrupt bit Value after Reset: 0b
5 HBR	Polarity bit for FC_INT0.HBR interrupt bit Value after Reset: 0b
4–3 -	This field is reserved. Reserved
2 AUDS	Polarity bit for FC_INT0.AUDS interrupt bit Value after Reset: 0b
1 ACR	Polarity bit for FC_INT0.ACR interrupt bit Value after Reset: 0b
0 NULL	Polarity bit for FC_INT0.NULL interrupt bit Value after Reset: 0b

33.5.145 FC_STAT1 (HDMI_FC_STAT1)

This register contains the following active high packet sent status indications:

- Address Offset: 0x10D4
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D4h offset = 12_10D4h

Bit	7	6	5	4	3	2	1	0
Read Write	GMD	ISCR1	ISCR2	VSD	SPD	Reserved	AVI	GCP
Reset	0	0	0	0	0	0	0	0

HDMI_FC_STAT1 field descriptions

Field	Description
7 GMD	Status bit Active after successful transmission of an Gamut metadata packet.
6 ISCR1	Status bit Active after successful transmission of an International Standard Recording Code 1 packet.
5 ISCR2	Active after successful transmission of an International Standard Recording Code 2 packet.
4 VSD	Active after successful transmission of an Vendor Specific Data infoFrame packet.
3 SPD	Active after successful transmission of an Source Product Descriptor infoFrame packet.
2 -	This field is reserved. Reserved
1 AVI	Status bit Active after successful transmission of an AVI infoFrame packet.
0 GCP	Status bit Active after successful transmission of an General Content Packet.

33.5.146 FC_INT1 (HDMI_FC_INT1)

This register contains the interrupt indication of the FC_STAT1 status interrupts. Interrupt generation is accomplished in the following way:

interrupt = (mask == 1'b0) && (polarity == status);

All this interrupts are forwarded to the Interrupt Handler sticky bit registers and after ORed to a single main interrupt line to micro controller. Assertion of this interrupt implies that data related with the corresponding packet has been sent through the HDMI interface.

- Address Offset: 0x10D5
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D5h offset = 12_10D5h

Bit	7	6	5	4	3	2	1	0
Read Write	GMD	ISCR1	ISCR2	VSD	SPD	Reserved	AVI	GCP
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INT1 field descriptions

Field	Description
7 GMD	Interrupt indication bit Active after successful transmission of an Gamut metadata packet interrupt.
6 ISCR1	Interrupt indication bit Active after successful transmission of an International Standard Recording Code 1 packet interrupt.
5 ISCR2	Interrupt indication bit Active after successful transmission of an International Standard Recording Code 2 packet interrupt.
4 VSD	Interrupt indication bit Active after successful transmission of an Vendor Specific Data infoFrame packet interrupt.
3 SPD	Interrupt indication bit Active after successful transmission of an Source Product Descriptor infoFrame packet interrupt.
2 -	This field is reserved. Reserved
1 AVI	Interrupt indication bit Active after successful transmission of an AVI infoFrame packet interrupt.
0 GCP	Interrupt indication bit Active after successful transmission of an General Content Packet interrupt.

33.5.147 Frame Composer Packet Interrupt Mask Register 1 (HDMI_FC_MASK1)

Mask register for generation of FC_INT1 interrupts.

- Address Offset: 0x10D6

- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D6h offset = 12_10D6h

Bit	7	6	5	4	3	2	1	0
Read Write	GMD	ISCR1	ISCR2	VSD	SPD	Reserved	AVI	GCP
Reset	0	0	0	0	0	0	0	0

HDMI_FC_MASK1 field descriptions

Field	Description
7 GMD	Mask bit for FC_INT1.GMD interrupt bit
6 ISCR1	Mask bit for FC_INT1.ISRC1 interrupt bit
5 ISCR2	Mask bit for FC_INT1.ISRC2 interrupt bit
4 VSD	Mask bit for FC_INT1.VSD interrupt bit
3 SPD	Mask bit for FC_INT1.SPD interrupt bit
2 -	This field is reserved. Reserved
1 AVI	Mask bit for FC_INT1.AVI interrupt bit
0 GCP	Mask bit for FC_INT1.GCP interrupt bit

33.5.148 FC_POL1 (HDMI_FC_POL1)

Polarity register for generation of FC_INT1 interrupts.

- Address Offset: 0x10D7
- Size: 8 bits
- Value after Reset: 0xFF
- Access: Read/Write

Address: 12_0000h base + 10D7h offset = 12_10D7h

Bit	7	6	5	4	3	2	1	0
Read Write	GMD	ISCR1	ISCR2	VSD	SPD	Reserved	AVI	GCP
Reset	1	1	1	1	1	1	1	1

HDMI_FC_POL1 field descriptions

Field	Description
7 GMD	Polarity bit for FC_INT1.GMD interrupt bit
6 ISCR1	Polarity bit for FC_INT1.ISRC1 interrupt bit
5 ISCR2	Polarity bit for FC_INT1.ISRC2 interrupt bit
4 VSD	Polarity bit for FC_INT1.VSD interrupt bit
3 SPD	Polarity bit for FC_INT1.SPD interrupt bit
2 -	This field is reserved. Reserved
1 AVI	Polarity bit for FC_INT1.AVI interrupt bit
0 GCP	Polarity bit for FC_INT1.GCP interrupt bit

33.5.149 FC_STAT2 (HDMI_FC_STAT2)

This register contains the following active high packet sent status indications:

- Address Offset: 0x10D8
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D8h offset = 12_10D8h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						LowPriority_ overflow	HighPriority _overflow
Reset	0	0	0	0	0	0	0	0

HDMI_FC_STAT2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	Status bit Frame Composer low priority packet queue descriptor overflow indication.
0 HighPriority_ overflow	Status bit Frame Composer high priority packet queue descriptor overflow indication.

33.5.150 FC_INT2 (HDMI_FC_INT2)

This register contains the interrupt indication of the FC_STAT2 status interrupts. Interrupt generation is accomplished in the following way:

interrupt = (mask == 1'b0) && (polarity == status);

All this interrupts are forwarded to the Interrupt Handler sticky bit registers and after ORed to a single main interrupt line to micro controller. Assertion of this interrupt implies that data related with the corresponding packet has been sent through the HDMI interface.

- Address Offset: 0x10D9
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10D9h offset = 12_10D9h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						LowPriority_ overflow	HighPriority _overflow
Reset	0	0	0	0	0	0	0	0

HDMI_FC_INT2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	Interrupt indication bit Frame Composer low priority packet queue descriptor overflow indication interrupt.
0 HighPriority_ overflow	Interrupt indication bit Frame Composer high priority packet queue descriptor overflow indication interrupt.

33.5.151 Frame Composer High/Low Priority Overflow Interrupt Mask Register 2 (HDMI_FC_MASK2)

Mask register for generation of FC_INT2 interrupts.

- Address Offset: 0x10DA
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 10DAh offset = 12_10DAh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						LowPriority_ overflow	HighPriority _overflow
Reset	0	0	0	0	0	0	0	0

HDMI_FC_MASK2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	Mask bit for FC_INT1.LowPriority_overflow interrupt bit Value after Reset: 0b
0 HighPriority_ overflow	Mask bit for FC_INT1.HighPriority_overflow interrupt bit Value after Reset: 0b

33.5.152 FC_POL2 (HDMI_FC_POL2)

Polarity register for generation of FC_INT2 interrupts.

- Address Offset: 0x10DB
- Size: 8 bits
- Value after Reset: 0x03
- Access: Read/Write

Address: 12_0000h base + 10DBh offset = 12_10DBh

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						LowPriority_ overflow	HighPriority _overflow
Reset	0	0	0	0	0	0	1	1

HDMI_FC_POL2 field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 LowPriority_ overflow	Polarity bit for FC_INT1.LowPriority_overflow interrupt bit Value after Reset: 1b
0 HighPriority_ overflow	Polarity bit for FC_INT1.HighPriority_overflow interrupt bit Value after Reset: 1b

33.5.153 Frame Composer Pixel Repetition Configuration Register (HDMI_FC_PRCONF)

Defines the Pixel Repetition ratio factor of the input and output video signal.

- Address Offset: 0x10E0
- Size: 8 bits
- Value after Reset: 0x10
- Access: Read/Write

Address: 12_0000h base + 10E0h offset = 12_10E0h

Bit	7	6	5	4	3	2	1	0
Read Write	incoming_pr_factor[3:0]				output_pr_factor[3:0]			
Reset	0	0	0	1	0	0	0	0

HDMI_FC_PRCONF field descriptions

Field	Description
7–4 incoming_pr_ factor[3:0]	Configures the input video pixel repetition. A plus 1 factor should be added in this register configuration. For CEA modes this value should be extracted from the CEA spec for the video mode being inputted. **NOTE:** When working in YCC422 video the actual repetition of the stream will be Incoming_pr_factor * (desired_pr_factor + 1). This calculation is done internally in the H13TCTRL and no HW overflow protection is available. Care must be taken to avoid this result passes the maximum number of 10 pixels repeated since no HDMI support is available for this in the spec and the H13TPHY does not support this higher repetition values. other: Reserved. Not used. 0000 No action. Shall not be used. 0001 No pixel repetition (pixel sent only once). 0010 Pixel sent twice (pixel repeated once). 0011 Pixel sent 3 times. 0100 Pixel sent 4 times. 0101 Pixel sent 5 times. 0110 Pixel sent 6 times. 0111 Pixel sent 7 times. 1000 Pixel sent 8 times. 1001 Pixel sent 9 times. 1010 Pixel sent 10 times.
output_pr_ factor[3:0]	Configures the video pixel repetition ratio to be sent on the AVI infoFrame. This value must be valid according to HDMI spec. The output_pr_factor = incoming_pr_factor(without the + 1 factor) * desired_pr_factor. other: Reserved. Not used. 0000 No action. Shall not be used. 0001 Pixel sent twice (pixel repeated once). 0010 Pixel sent 3 times. 0011 Pixel sent 4 times.

Table continues on the next page...

HDMI_FC_PRCONF field descriptions (continued)

Field	Description
	0100 Pixel sent 5 times. 0101 Pixel sent 6 times. 0110 Pixel sent 7 times. 0111 Pixel sent 8 times. 1000 Pixel sent 9 times. 1001 Pixel sent 10 times.

33.5.154 Frame Composer GMD Packet Status Register (HDMI_FC_GMD_STAT)

Gamut metadata packet status bit information for no_current_gmd, next_gmd_field, gmd_packet_sequence and current_gamut_seq_num. For more information, refer to the HDMI 1.4a specification.

- Address Offset: 0x1100
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 1100h offset = 12_1100h

Bit	7	6	5	4
Read	igmdno_crnt_gbd	igmddnext_field	igmdpacket_seq[1:0]	
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	igmdcurrent_gamut_seq_num[3:0]			
Write				
Reset	0	0	0	0

HDMI_FC_GMD_STAT field descriptions

Field	Description
7 igmdno_crnt_gbd	Gamut scheduling: No current gamut data
6 igmddnext_field	Gamut scheduling: Gamut Next field
5–4 igmdpacket_seq[1:0]	Gamut scheduling: Gamut packet sequence
igmdcurrent_gamut_seq_num[3:0]	Gamut scheduling: Current Gamut packet sequence number

33.5.155 Frame Composer GMD Packet Enable Register (HDMI_FC_GMD_EN)

This register enables Gamut metadata (GMD) packet transmission. Packets are inserted in the incoming frame, starting in the line where active Vsync indication starts. After enable of GMD packets the outgoing packet is sent with no_current_gmd active indication until update GMD request is performed in the controller.

- Address Offset: 0x1101
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1101h offset = 12_1101h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							gmdenabletx
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_EN field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 gmdenabletx	Gamut Metadata packet transmission enable (1b).

33.5.156 Frame Composer GMD Packet Update Register (HDMI_FC_GMD_UP)

This register performs an GMD packet content update according to the configured packet body (FC_GMD_PB0 to FC_GMD_PB27) and packet header (FC_GMD_HB). This active high auto clear register reflects the body and header configurations on the GMD packets sent arbitrating the current_gamut_seq_num, gmd_packet_sequence and next_gmd_field bits on packet to correctly indicate to source the Gamut change to be performed. After enable GMD packets the first update request is also responsible for deactivating the no_current_gmd indication bit. Attention packet update request must only be done after correct configuration of GMD packet body and header registers. Correct affected_gamut_seq_num and gmd_profile configuration is user responsibility and must convey with HDMI 1.4a standard gamut rules.

- Address Offset: 0x1102

- Size: 8 bits
- Value after Reset: 0x00
- Access: Write

Address: 12_0000h base + 1102h offset = 12_1102h

Bit	7	6	5	4
Read	Reserved			
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved			
Write				gmdupdatepacket
Reset	0	0	0	0

HDMI_FC_GMD_UP field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 gmdupdatepacket	Gamut Metadata packet update.

33.5.157 Frame Composer GMD Packet Schedule Configuration Register (HDMI_FC_GMD_CONF)

This register configures the number of GMD packets to be inserted per frame (starting always in the line where the active Vsync appears) and the line spacing between the transmitted GMD packets. Note that for profile P0 (refer to HDMI 1.4a spec) this register should only indicate one GMD packet to be inserted per video field.

- Address Offset: 0x1103
- Size: 8 bits
- Value after Reset: 0x10
- Access: Read/Write

Address: 12_0000h base + 1103h offset = 12_1103h

Bit	7	6	5	4	3	2	1	0
Read Write	gmdpacketsinframe[3:0]				gmdpacketlinespacing[3:0]			
Reset	0	0	0	1	0	0	0	0

HDMI_FC_GMD_CONF field descriptions

Field	Description
7–4 gmdpacketsinframe[3:0]	Number of GMD packets per frame or video field (profile P0)
gmdpacketlinespacing[3:0]	Number of line spacing between the transmitted GMD packets

33.5.158 Frame Composer GMD Packet Profile and Gamut Sequence Configuration Register (HDMI_FC_GMD_HB)

This register configures the GMD packet header affected_gamut_seq_num and gmd_profile bits. For more information, refer to the HDMI 1.4a specification.

- Address Offset: 0x1104
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1104h offset = 12_1104h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	gmdgbd_profile			gmdaffected_gamut_seq_num			
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_HB field descriptions

Field	Description
7 -	This field is reserved. Reserved
6–4 gmdgbd_profile	GMD profile bits
gmdaffected_ gamut_seq_num	Affected gamut sequence number

33.5.159 Frame Composer GMD Packet Body Register 0 (HDMI_FC_GMD_PB0)

Configures the following contents of the GMD packet:

- GMD packet body byte0
- Address Offset: 0x1105
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1105h offset = 12_1105h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB0 field descriptions

Field	Description
FC_GMD_PB0	Gamut Metadata packet byte0

33.5.160 Frame Composer GMD Packet Body Register 1 (HDMI_FC_GMD_PB1)

- GMD packet body byte1
- Address Offset: 0x1106
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1106h offset = 12_1106h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB1 field descriptions

Field	Description
FC_GMD_PB1	Gamut Metadata packet byte1

33.5.161 Frame Composer GMD Packet Body Register 2 (HDMI_FC_GMD_PB2)

- GMD packet body byte2
- Address Offset: 0x1107
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1107h offset = 12_1107h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB2 field descriptions

Field	Description
FC_GMD_PB2	Gamut Metadata packet byte2

33.5.162 Frame Composer GMD Packet Body Register 3 (HDMI_FC_GMD_PB3)

- GMD packet body byte3
- Address Offset: 0x1108
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1108h offset = 12_1108h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB3							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB3 field descriptions

Field	Description
FC_GMD_PB3	Gamut Metadata packet byte3

33.5.163 Frame Composer GMD Packet Body Register 4 (HDMI_FC_GMD_PB4)

- GMD packet body byte4
- Address Offset: 0x1109
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1109h offset = 12_1109h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB4							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB4 field descriptions

Field	Description
FC_GMD_PB4	Gamut Metadata packet byte4

33.5.164 Frame Composer GMD Packet Body Register 5 (HDMI_FC_GMD_PB5)

- GMD packet body byte5
- Address Offset: 0x110a
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Ah offset = 12_110Ah

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB5							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB5 field descriptions

Field	Description
FC_GMD_PB5	Gamut Metadata packet byte5

33.5.165 Frame Composer GMD Packet Body Register 6 (HDMI_FC_GMD_PB6)

- GMD packet body byte6
- Address Offset: 0x110b
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Bh offset = 12_110Bh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB6							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB6 field descriptions

Field	Description
FC_GMD_PB6	Gamut Metadata packet byte6

33.5.166 Frame Composer GMD Packet Body Register 7 (HDMI_FC_GMD_PB7)

- GMD packet body byte7
- Address Offset: 0x110c
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Ch offset = 12_110Ch

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB7 field descriptions

Field	Description
FC_GMD_PB2	Gamut Metadata packet byte7

33.5.167 Frame Composer GMD Packet Body Register 8 (HDMI_FC_GMD_PB8)

- GMD packet body byte8
- Address Offset: 0x110d
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Dh offset = 12_110Dh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB8							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB8 field descriptions

Field	Description
FC_GMD_PB8	Gamut Metadata packet byte8

33.5.168 Frame Composer GMD Packet Body Register 9 (HDMI_FC_GMD_PB9)

- GMD packet body byte9
- Address Offset: 0x110e
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Eh offset = 12_110Eh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB9							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB9 field descriptions

Field	Description
FC_GMD_PB9	Gamut Metadata packet byte9

33.5.169 Frame Composer GMD Packet Body Register 10 (HDMI_FC_GMD_PB10)

- GMD packet body byte10
- Address Offset: 0x110f
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 110Fh offset = 12_110Fh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB10							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB10 field descriptions

Field	Description
FC_GMD_PB10	Gamut Metadata packet byte10

33.5.170 Frame Composer GMD Packet Body Register 11 (HDMI_FC_GMD_PB11)

- GMD packet body byte11
- Address Offset: 0x1110
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1110h offset = 12_1110h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB11							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB11 field descriptions

Field	Description
FC_GMD_PB11	Gamut Metadata packet byte11

33.5.171 Frame Composer GMD Packet Body Register 12 (HDMI_FC_GMD_PB12)

- GMD packet body byte12
- Address Offset: 0x1111
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1111h offset = 12_1111h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB12							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB12 field descriptions

Field	Description
FC_GMD_PB12	Gamut Metadata packet byte12

33.5.172 Frame Composer GMD Packet Body Register 13 (HDMI_FC_GMD_PB13)

- GMD packet body byte13
- Address Offset: 0x1112
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1112h offset = 12_1112h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB13							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB13 field descriptions

Field	Description
FC_GMD_PB13	Gamut Metadata packet byte13

33.5.173 Frame Composer GMD Packet Body Register 14 (HDMI_FC_GMD_PB14)

- GMD packet body byte14
- Address Offset: 0x1113
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1113h offset = 12_1113h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB14							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB14 field descriptions

Field	Description
FC_GMD_PB14	Gamut Metadata packet byte14

33.5.174 Frame Composer GMD Packet Body Register 15 (HDMI_FC_GMD_PB15)

- GMD packet body byte15
- Address Offset: 0x1114
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1114h offset = 12_1114h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB15							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB15 field descriptions

Field	Description
FC_GMD_PB15	Gamut Metadata packet byte15

33.5.175 Frame Composer GMD Packet Body Register 16 (HDMI_FC_GMD_PB16)

- GMD packet body byte16
- Address Offset: 0x1115
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1115h offset = 12_1115h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB16							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB16 field descriptions

Field	Description
FC_GMD_PB16	Gamut Metadata packet byte16

33.5.176 Frame Composer GMD Packet Body Register 17 (HDMI_FC_GMD_PB17)

- GMD packet body byte17
- Address Offset: 0x1116
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1116h offset = 12_1116h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB17							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB17 field descriptions

Field	Description
FC_GMD_PB17	Gamut Metadata packet byte17

33.5.177 Frame Composer GMD Packet Body Register 18 (HDMI_FC_GMD_PB18)

- GMD packet body byte18
- Address Offset: 0x1117
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1117h offset = 12_1117h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB18							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB18 field descriptions

Field	Description
FC_GMD_PB18	Gamut Metadata packet byte18

33.5.178 Frame Composer GMD Packet Body Register 19 (HDMI_FC_GMD_PB19)

- GMD packet body byte19
- Address Offset: 0x1118
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1118h offset = 12_1118h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB18							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB19 field descriptions

Field	Description
FC_GMD_PB18	Gamut Metadata packet byte18

33.5.179 Frame Composer GMD Packet Body Register 20 (HDMI_FC_GMD_PB20)

- GMD packet body byte20
- Address Offset: 0x1119
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1119h offset = 12_1119h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB20							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB20 field descriptions

Field	Description
FC_GMD_PB20	Gamut Metadata packet byte20

33.5.180 Frame Composer GMD Packet Body Register 21 (HDMI_FC_GMD_PB21)

- GMD packet body byte21
- Address Offset: 0x111a
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Ah offset = 12_111Ah

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB21							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB21 field descriptions

Field	Description
FC_GMD_PB21	Gamut Metadata packet byte21

33.5.181 Frame Composer GMD Packet Body Register 22 (HDMI_FC_GMD_PB22)

- GMD packet body byte22
- Address Offset: 0x111b
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Bh offset = 12_111Bh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB22							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB22 field descriptions

Field	Description
FC_GMD_PB22	Gamut Metadata packet byte22

33.5.182 Frame Composer GMD Packet Body Register 23 (HDMI_FC_GMD_PB23)

- GMD packet body byte23
- Address Offset: 0x111c
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Ch offset = 12_111Ch

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB23							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB23 field descriptions

Field	Description
FC_GMD_PB23	Gamut Metadata packet byte23

33.5.183 Frame Composer GMD Packet Body Register 24 (HDMI_FC_GMD_PB24)

- GMD packet body byte24
- Address Offset: 0x111d
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Dh offset = 12_111Dh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB24							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB24 field descriptions

Field	Description
FC_GMD_PB24	Gamut Metadata packet byte24

33.5.184 Frame Composer GMD Packet Body Register 25 (HDMI_FC_GMD_PB25)

- GMD packet body byte25
- Address Offset: 0x111e
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Eh offset = 12_111Eh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB25							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB25 field descriptions

Field	Description
FC_GMD_PB25	Gamut Metadata packet byte25

33.5.185 Frame Composer GMD Packet Body Register 26 (HDMI_FC_GMD_PB26)

- GMD packet body byte26
- Address Offset: 0x111f
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 111Fh offset = 12_111Fh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB26							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB26 field descriptions

Field	Description
FC_GMD_PB26	Gamut Metadata packet byte26

33.5.186 Frame Composer GMD Packet Body Register 27 (HDMI_FC_GMD_PB27)

- GMD packet body byte27
- Address Offset: 0x1120
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

For more information, refer to the HDMI 1.4a specification.

Address: 12_0000h base + 1120h offset = 12_1120h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_GMD_PB27							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_GMD_PB27 field descriptions

Field	Description
FC_GMD_PB27	Gamut Metadata packet byte27

33.5.187 Frame Composer Video/Audio Force Enable Register (HDMI_FC_DBGFORCE)

This register allows to force the controller to output audio and video data the values configured in the FC_DBGAUD and FC_DBGTMDS registers.

- Address Offset: 0x1200
- Size: 8 bits

- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1200h offset = 12_1200h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			forceaudio	Reserved			forcevideo
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGFORCE field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4 forceaudio	Force fixed audio output with FC_DBGAUDxCHx registers contain.
3–1 -	This field is reserved. Reserved
0 forcevideo	Force fixed video output with FC_DBGTMDSx registers contain.

33.5.188 Frame Composer Audio Channel 0 Register 0 (HDMI_FC_DBGAUD0CH0)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x1201
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1201h offset = 12_1201h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH0 field descriptions

Field	Description
FC_ DBGAUD0CH0	the audio fixed data byte0 to be used in channel 0 when in fixed audio selection

33.5.189 Frame Composer Audio Channel 0 Register 1 (HDMI_FC_DBGAUD1CH0)

- Address Offset: 0x1202
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1202h offset = 12_1202h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH0 field descriptions

Field	Description
FC_ DBGAUD1CH0	the audio fixed data byte1 to be used in channel 0 when in fixed audio selection

33.5.190 Frame Composer Audio Channel 0 Register 2 (HDMI_FC_DBGAUD2CH0)

- Address Offset: 0x1203
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1203h offset = 12_1203h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH0 field descriptions

Field	Description
FC_ DBGAUD2CH0	the audio fixed data byte2 to be used in channel 0 when in fixed audio selection

33.5.191 Frame Composer Audio Channel 1 Register 0 (HDMI_FC_DBGAUD0CH1)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x1204
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1204h offset = 12_1204h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH1 field descriptions

Field	Description
FC_ DBGAUD0CH1	the audio fixed data byte2 to be used in channel 0 when in fixed audio selection

33.5.192 Frame Composer Audio Channel 1 Register 1 (HDMI_FC_DBGAUD1CH1)

- Address Offset: 0x1205
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1205h offset = 12_1205h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH1 field descriptions

Field	Description
FC_ DBGAUD1CH1	the audio fixed data byte1 to be used in channel 1 when in fixed audio selection

33.5.193 Frame Composer Audio Channel 1 Register 2 (HDMI_FC_DBGAUD2CH1)

- Address Offset: 0x1206
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1206h offset = 12_1206h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH1 field descriptions

Field	Description
FC_ DBGAUD2CH1	the audio fixed data byte2 to be used in channel 1 when in fixed audio selection

33.5.194 Frame Composer Debug Audio Channel 2 Register 0 (HDMI_FC_DBGAUD0CH2)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x1207
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1207h offset = 12_1207h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH2 field descriptions

Field	Description
FC_ DBGAUD0CH2	the audio fixed data byte0 to be used in channel 2 when in fixed audio selection

33.5.195 Frame Composer Debug Audio Channel 2 Register 1 (HDMI_FC_DBGAUD1CH2)

- Address Offset: 0x1208
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1208h offset = 12_1208h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH2 field descriptions

Field	Description
FC_ DBGAUD1CH2	the audio fixed data byte1 to be used in channel 2 when in fixed audio selection

33.5.196 Frame Composer Audio Channel 2 Register 2 (HDMI_FC_DBGAUD2CH2)

- Address Offset: 0x1209
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1209h offset = 12_1209h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH2 field descriptions

Field	Description
FC_ DBGAUD2CH2	the audio fixed data byte2 to be used in channel 2 when in fixed audio selection

33.5.197 Frame Composer Audio Channel 3 Register 0 (HDMI_FC_DBGAUD0CH3)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x120A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Ah offset = 12_120Ah

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH3							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH3 field descriptions

Field	Description
FC_ DBGAUD0CH3	the audio fixed data byte0 to be used in channel 3 when in fixed audio selection

33.5.198 Frame Composer Audio Channel 3 Register 1 (HDMI_FC_DBGAUD1CH3)

- Address Offset: 0x120B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Bh offset = 12_120Bh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH3							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH3 field descriptions

Field	Description
FC_ DBGAUD1CH3	the audio fixed data byte1 to be used in channel 3 when in fixed audio selection

33.5.199 Frame Composer Audio Channel 3 Register 2 (HDMI_FC_DBGAUD2CH3)

- Address Offset: 0x120C
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Ch offset = 12_120Ch

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH3							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH3 field descriptions

Field	Description
FC_ DBGAUD2CH3	the audio fixed data byte2 to be used in channel 3 when in fixed audio selection

33.5.200 Frame Composer Audio Channel 4 Register 0 (HDMI_FC_DBGAUD0CH4)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x120D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Dh offset = 12_120Dh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH4							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH4 field descriptions

Field	Description
FC_ DBGAUD0CH4	the audio fixed data byte0 to be used in channel 4 when in fixed audio selection

33.5.201 Frame Composer Audio Channel 4 Register 1 (HDMI_FC_DBGAUD1CH4)

- Address Offset: 0x120E
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Eh offset = 12_120Eh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH4							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH4 field descriptions

Field	Description
FC_ DBGAUD1CH4	the audio fixed data byte1 to be used in channel 4 when in fixed audio selection

33.5.202 Frame Composer Audio Channel 4 Register 2 (HDMI_FC_DBGAUD2CH4)

- Address Offset: 0x120F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 120Fh offset = 12_120Fh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH4							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH4 field descriptions

Field	Description
FC_ DBGAUD2CH4	the audio fixed data byte2 to be used in channel 4 when in fixed audio selection

33.5.203 Frame Composer Audio Channel 5 Register 0 (HDMI_FC_DBGAUD0CH5)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x1210
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1210h offset = 12_1210h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH5							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH5 field descriptions

Field	Description
FC_ DBGAUD0CH5	the audio fixed data byte0 to be used in channel 5 when in fixed audio selection

33.5.204 Frame Composer Audio Channel 5 Register 1 (HDMI_FC_DBGAUD1CH5)

- Address Offset: 0x1211
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1211h offset = 12_1211h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH5							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH5 field descriptions

Field	Description
FC_ DBGAUD1CH5	the audio fixed data byte1 to be used in channel 5 when in fixed audio selection

33.5.205 Frame Composer Audio Channel 5 Register 2 (HDMI_FC_DBGAUD2CH5)

- Address Offset: 0x1212
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1212h offset = 12_1212h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH5							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH5 field descriptions

Field	Description
FC_ DBGAUD2CH5	the audio fixed data byte2 to be used in channel 5 when in fixed audio selection

33.5.206 Frame Composer Audio Channel 6 Register 0 (HDMI_FC_DBGAUD0CH6)

Configures the audio fixed data to be used in channel 0 when in fixed audio selection.

- Address Offset: 0x1213
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1213h offset = 12_1213h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH6							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH6 field descriptions

Field	Description
FC_ DBGAUD0CH6	The audio fixed data byte0 to be used in channel 6 when in fixed audio selection

33.5.207 Frame Composer Audio Channel 6 Register 1 (HDMI_FC_DBGAUD1CH6)

- Address Offset: 0x1214
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1214h offset = 12_1214h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH6							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH6 field descriptions

Field	Description
FC_ DBGAUD1CH6	the audio fixed data byte1 to be used in channel 6 when in fixed audio selection

33.5.208 Frame Composer Audio Channel 6 Register 2 (HDMI_FC_DBGAUD2CH6)

- Address Offset: 0x1215
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1215h offset = 12_1215h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH6							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH6 field descriptions

Field	Description
FC_ DBGAUD2CH6	the audio fixed data byte2 to be used in channel 6 when in fixed audio selection

33.5.209 Frame Composer Audio Channel 7 Register 1 (HDMI_FC_DBGAUD0CH7)

Configures the audio fixed data to be used in channel 7 when in fixed audio selection.

- Address Offset: 0x1216
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1216h offset = 12_1216h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD0CH7							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD0CH7 field descriptions

Field	Description
FC_ DBGAUD0CH7	the audio fixed data byte0 to be used in channel 7 when in fixed audio selection

33.5.210 Frame Composer Audio Channel 7 Register 0 (HDMI_FC_DBGAUD1CH7)

- Address Offset: 0x1217
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1217h offset = 12_1217h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD1CH7							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD1CH7 field descriptions

Field	Description
FC_ DBGAUD1CH7	the audio fixed data byte1 to be used in channel 0 when in fixed audio selection

33.5.211 Frame Composer Audio Channel 7 Register 2 (HDMI_FC_DBGAUD2CH7)

- Address Offset: 0x1218
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1218h offset = 12_1218h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGAUD2CH7							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGAUD2CH7 field descriptions

Field	Description
FC_ DBGAUD2CH7	the audio fixed data byte2 to be used in channel 0 when in fixed audio selection

33.5.212 Frame Composer TMDS Channel 0 Register (HDMI_FC_DBGTMDS0)

Configures the video fixed data to be used in tmds channel 0 when in fixed video selection. This equals to set B pixel component value in RGB video or Cb pixel component value in YCbCr.

- Address Offset: 0x1219
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 1219h offset = 12_1219h

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGTMDS0							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGTMDS0 field descriptions

Field	Description
FC_DBGTMDS0	set B pixel component value in RGB video or Cb pixel component value in YCbCr

33.5.213 Frame Composer TMDS Channel 1 Register (HDMI_FC_DBGTMDS1)

Configures the video fixed data to be used in tmds channel 1 when in fixed video selection. This equals to set G pixel component value in RGB video or Y pixel component value in YCbCr.

- Address Offset: 0x121A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 121Ah offset = 12_121Ah

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGTMDS1							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGTMDS1 field descriptions

Field	Description
FC_DBGTMDS1	set G pixel component value in RGB video or Y pixel component value in YCbCr

33.5.214 Frame Composer TMDS Channel 2 Register (HDMI_FC_DBGTMDS2)

Configures the video fixed data to be used in tmds channel 2 when in fixed video selection. This equals to set R pixel component value in RGB video or Cr pixel component value in YCbCr.

- Address Offset: 0x121B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 121Bh offset = 12_121Bh

Bit	7	6	5	4	3	2	1	0
Read Write	FC_DBGTMDS2							
Reset	0	0	0	0	0	0	0	0

HDMI_FC_DBGTMDS2 field descriptions

Field	Description
FC_DBGTMDS2	set R pixel component value in RGB video or Cr pixel component value in YCbCr

33.5.215 PHY Configuration Register (HDMI_PHY_CONF0)

This register holds the power down, data enable polarity and interface control of the HDMI Source PHY control. For more information, refer to the DesignWare Cores HDMI TX PHY Databook.

- Address Offset: 0x3000
- Size: 8 bits
- Value after Reset: 0x06
- Access: Read/Write

Address: 12_0000h base + 3000h offset = 12_3000h

Bit	7	6	5	4
Read Write	PDZ	ENTMDS	sparectrl	gen2_pddq
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	gen2_txpwron	gen2_enhpdrxsense	seldataenpol	seldipif
Reset	0	1	1	0

HDMI_PHY_CONF0 field descriptions

Field	Description
7 PDZ	Power-down enable (active low 0b). Value after Reset: 0b
6 ENTMDS	Enable TMDS drivers, bias, and TMDS digital logic. Value after Reset: 0b
5 sparectrl	Reserved. Spare pin control. Value after Reset: 0b
4 gen2_pddq	PHY_Gen2 PDDQ signal Value after Reset: 0b
3 gen2_txpwron	PHY_Gen2 TXPWRON signal Value after Reset: 0b
2 gen2_ enhpdrxsense	PHY_Gen2 ENHPDRXSENSE signal Value after Reset: 1b
1 seldataenpol	Select data enable polarity. Value after Reset: 1b

Table continues on the next page...

HDMI_PHY_CONF0 field descriptions (continued)

Field	Description
0 seldipif	Select interface control. Value after Reset: 0b

33.5.216 PHY Test Interface Register 0 (HDMI_PHY_TST0)

PHY TX mapped text interface (control). For more information, refer to the DesignWare Cores HDMI TX PHY Databook.

- Address Offset: 0x3001
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3001h offset = 12_3001h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		testclr	testen	Reserved			testclk
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_TST0 field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5 testclr	Enable TMDS drivers, bias and tmds digital logic. Value after Reset: 0b
4 testen	Reserved. Spare control pins. Value after Reset: 0b
3–1 -	This field is reserved. Reserved
0 testclk	Test clock signal. Value after Reset: 0b

33.5.217 PHY Test Interface Register 1 (HDMI_PHY_TST1)

PHY TX mapped text interface (data in). For more information, refer to the DesignWare Cores HDMI TX PHY Databook.

- Address Offset: 0x3002
- Size: 8 bits

- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3002h offset = 12_3002h

Bit	7	6	5	4	3	2	1	0
Read Write	testdin[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_TST1 field descriptions

Field	Description
testdin[7:0]	Test data input.

33.5.218 PHY Test Interface Register 2 (HDMI_PHY_TST2)

PHY TX mapped text interface (data out). For more information, refer to the DesignWare Cores HDMI TX PHY Databook.

- Address Offset: 0x3003
- Size: 8 bits
- Value after Reset: N/A
- Access: Read

Address: 12_0000h base + 3003h offset = 12_3003h

Bit	7	6	5	4	3	2	1	0
Read	testdout[7:0]							
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_TST2 field descriptions

Field	Description
testdout[7:0]	Test data output.

33.5.219 PHY RXSENSE, PLL lock, and HPD Status Register (HDMI_PHY_STAT0)

This register contains the following active high packet sent status indications. For more information, see Overview

- Address Offset: 0x3004
- Size: 8 bits

- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3004h offset = 12_3004h

Bit	7	6	5	4
Read	RX_SENSE3	RX_SENSE2	RX_SENSE1	RX_SENSE0
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved		HPD	TX_PHY_LOCK
Write				
Reset	0	0	0	0

HDMI_PHY_STAT0 field descriptions

Field	Description
7 RX_SENSE3	Status bit. TX PHY RX_SENSE indication for TMDS CLK driver. User may need to mask or change polarity of this interrupt after it has became active.
6 RX_SENSE2	Status bit. TX PHY RX_SENSE indication for TMDS channel 2 driver. User may need to mask or change polarity of this interrupt after it has became active.
5 RX_SENSE1	Status bit. TX PHY RX_SENSE indication for TMDS channel 1 driver. User may need to mask or change polarity of this interrupt after it has became active.
4 RX_SENSE0	Status bit. TX PHY RX_SENSE indication for TMDS channel 0 driver. User may need to mask or change polarity of this interrupt after it has became active.
3–2 -	This field is reserved. Reserved
1 HPD	Status bit. HDMI Hot Plug Detect indication. User may need to mask or change polarity of this interrupt after it has became active.
0 TX_PHY_LOCK	Status bit. TX PHY PLL lock indication. Please refer to PHY datasheet for more information. User may need to mask or change polarity of this interrupt after it has became active.

33.5.220 PHY RXSENSE, PLL lock, and HPD Interrupt Register (HDMI_PHY_INT0)

This register contains the interrupt indication of the PHY_STAT0 status interrupts. Interrupt generation is accomplished in the following way:

interrupt = (mask == 1'b0) && (polarity == status);

All this interrupts are forwarded to the Interrupt Handler sticky bit registers and after ORed to a single main interrupt line to micro controller. Assertion of this interrupt implies that data related with the corresponding packet has been sent through the HDMI interface.

- Address Offset: 0x3005

- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3005h offset = 12_3005h

Bit	7	6	5	4
Read	RX_SENSE3	RX_SENSE2	RX_SENSE1	RX_SENSE0
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved		HPD	TX_PHY_LOCK
Write				
Reset	0	0	0	0

HDMI_PHY_INT0 field descriptions

Field	Description
7 RX_SENSE3	Interrupt indication bit TX PHY RX_SENSE indication interrupt for TMDS CLK driver.
6 RX_SENSE2	Interrupt indication bit TX PHY RX_SENSE indication interrupt for TMDS channel 2 driver.
5 RX_SENSE1	Interrupt indication bit TX PHY RX_SENSE indication interrupt for TMDS channel 1 driver.
4 RX_SENSE0	Interrupt indication bit TX PHY RX_SENSE indication interrupt for TMDS channel 0 driver.
3–2 -	This field is reserved. Reserved
1 HPD	Interrupt indication bit HDMI Hot Plug Detect indication interrupt.
0 TX_PHY_LOCK	Interrupt indication bit TX PHY PLL lock indication interrupt.

33.5.221 PHY RXSENSE, PLL lock, and HPD Mask Register (HDMI_PHY_MASK0)

Mask register for generation of PHY_INT0 interrupts.

- Address Offset: 0x3006
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3006h offset = 12_3006h

Bit	7	6	5	4	3	2	1	0
Read Write	RX_ SENSE3	RX_ SENSE2	RX_ SENSE1	RX_ SENSE0	Reserved		HPD	TX_PHY_ LOCK
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_MASK0 field descriptions

Field	Description
7 RX_SENSE3	Mask bit for PHY_INT0.RX_SENSE3 interrupt bit
6 RX_SENSE2	Mask bit for PHY_INT0.RX_SENSE2 interrupt bit
5 RX_SENSE1	Mask bit for PHY_INT0.RX_SENSE1 interrupt bit
4 RX_SENSE0	Mask bit for PHY_INT0.RX_SENSE0 interrupt bit
3–2 -	This field is reserved. Reserved
1 HPD	Mask bit for PHY_INT0.HPD interrupt bit
0 TX_PHY_LOCK	Mask bit for PHY_INT0.TX_PHY_LOCK interrupt bit

33.5.222 PHY RXSENSE, PLL lock and HPD Polarity Register (HDMI_PHY_POL0)

Polarity register for generation of PHY_INT0 interrupts.

- Address Offset: 0x3007
- Size: 8 bits
- Value after Reset: 0xF3
- Access: Read/Write

Address: 12_0000h base + 3007h offset = 12_3007h

Bit	7	6	5	4	3	2	1	0
Read Write	RX_ SENSE3	RX_ SENSE2	RX_ SENSE1	RX_ SENSE0	Reserved		HPD	TX_PHY_ LOCK
Reset	1	1	1	1	0	0	1	1

HDMI_PHY_POL0 field descriptions

Field	Description
7 RX_SENSE3	Polarity bit for PHY_INT0.RX_SENSE3 interrupt bit

Table continues on the next page...

HDMI_PHY_POL0 field descriptions (continued)

Field	Description
6 RX_SENSE2	Polarity bit for PHY_INT0.RX_SENSE2 interrupt bit
5 RX_SENSE1	Polarity bit for PHY_INT0.RX_SENSE1 interrupt bit
4 RX_SENSE0	Polarity bit for PHY_INT0.RX_SENSE0 interrupt bit
3–2 -	This field is reserved. Reserved
1 HPD	Polarity bit for PHY_INT0.HPD interrupt bit
0 TX_PHY_LOCK	Polarity bit for PHY_INT0.TX_PHY_LOCK interrupt bit

33.5.223 PHY I2C Slave Address Configuration Register (HDMI_PHY_I2CM_SLAVE_ADDR)

This register writes the slave address of the I2C Master PHY.

- Address Offset: 0x3020
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3020h offset = 12_3020h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	-						
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_SLAVE_ADDR field descriptions

Field	Description
7 -	This field is reserved. Reserved
-	Slave address to be sent during read and write operations. The PHY Gen2 slave address is: 7'h69 The HEAC PHY slave address is: 7'h49

33.5.224 PHY I2C Address Configuration Register (HDMI_PHY_I2CM_ADDRESS_ADDR)

This register writes the address for read and writer operations.

- Address Offset: 0x3021
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3021h offset = 12_3021h

Bit	7	6	5	4	3	2	1	0
Read Write	address							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_ADDRESS_ADDR field descriptions

Field	Description
address	Register address for read and write operations.

33.5.225 PHY I2C Data Write Register 1 (HDMI_PHY_I2CM_DATAO_1_ADDR)

- Address Offset: 0x3022
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3022h offset = 12_3022h

Bit	7	6	5	4	3	2	1	0
Read Write	datao[15:8]							
Reset	0	0	0	0	0	0	0	0

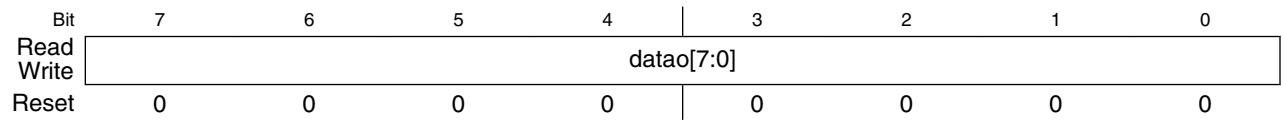
HDMI_PHY_I2CM_DATAO_1_ADDR field descriptions

Field	Description
datao[15:8]	MSB's of data to be written on register pointed by address [7:0].

33.5.226 PHY I2C Data Write Register 0 (HDMI_PHY_I2CM_DATAO_0_ADDR)

- Address Offset: 0x3023
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3023h offset = 12_3023h



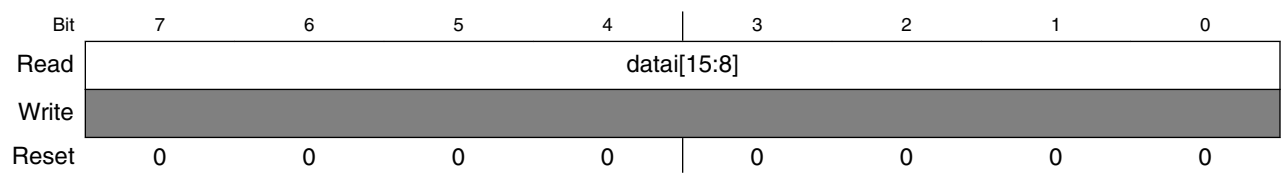
HDMI_PHY_I2CM_DATAO_0_ADDR field descriptions

Field	Description
datao[7:0]	LSB's of data to be written on register pointed by address [7:0].

33.5.227 PHY I2C Data Read Register 1 (HDMI_PHY_I2CM_DATAI_1_ADDR)

- Address Offset: 0x3024
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3024h offset = 12_3024h



HDMI_PHY_I2CM_DATAI_1_ADDR field descriptions

Field	Description
datai[15:8]	MSB's of data read from the register pointed by address [7:0].

33.5.228 PHY I2C Data Read Register 0 (HDMI_PHY_I2CM_DATAI_0_ADDR)

- Address Offset: 0x3025
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3025h offset = 12_3025h

Bit	7	6	5	4	3	2	1	0
Read Write	datai[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_DATAI_0_ADDR field descriptions

Field	Description
datai[7:0]	LSB's of data read from the register pointed by address [7:0].

33.5.229 PHY I2C Read/Write Operation (HDMI_PHY_I2CM_OPERATION_ADDR)

This register requests read and write operations from the I2C Master PHY. This register can only be written; reading this register always results in 00h. Writing 1'b1 simultaneously to read and write requests is considered a read request.

- Address Offset: 0x3026
- Size: 8 bits
- Value after Reset: 0x00
- Access: Write

Address: 12_0000h base + 3026h offset = 12_3026h

Bit	7	6	5	4	3	2	1	0
Read	Reserved				Reserved			
Write				write				read
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_OPERATION_ADDR field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4 write	Write operation request
3–1 -	This field is reserved. Reserved
0 read	Read operation request.

33.5.230 PHY I2C Done Interrupt Register (HDMI_PHY_I2CM_INT_ADDR)

This register contains and configures I2C master PHY done interrupt.

- Address Offset: 0x3027
- Size: 8 bits
- Value after Reset: 0x08
- Access: Read/Write

Address: 12_0000h base + 3027h offset = 12_3027h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				done_pol	done_mask	done_ interrupt	done_status
Reset	0	0	0	0	1	0	0	0

HDMI_PHY_I2CM_INT_ADDR field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
3 done_pol	Done interrupt polarity configuration Value after Reset: 1b
2 done_mask	Done interrupt mask signal Value after Reset: 0b
1 done_interrupt	Operation done interrupt bit.{done_interrupt =(done_mask==0b)&& (done_status==done_pol)}. Value after Reset: 0b
0 done_status	Operation done status bit.Marks the end of a rd or write operation. Value after Reset: 0b

33.5.231 PHY I2C Done Interrupt Register (HDMI_PHY_I2CM_CTLINT_ADDR)

This register contains and configures the I2C master PHY error interrupts.

- Address Offset: 0x3028
- Size: 8 bits
- Value after Reset: 0x88
- Access: Read/Write

Address: 12_0000h base + 3028h offset = 12_3028h

Bit	7	6	5	4
Read Write	nack_pol	nack_mask	nack_interrupt	nack_status
Reset	1	0	0	0

Bit	3	2	1	0
Read Write	arbitration_pol	arbitration_mask	arbitration_interrupt	arbitration_status
Reset	1	0	0	0

HDMI_PHY_I2CM_CTLINT_ADDR field descriptions

Field	Description
7 nack_pol	Not acknowledge error interrupt polarity configuration. Value after Reset: 1b
6 nack_mask	Not acknowledge error interrupt mask signal Value after Reset: 0b
5 nack_interrupt	Not acknowledge error interrupt bit.{nack_interrupt = nack_mask==0b) && (nack_status==nack_pol)}. Value after Reset: 0b
4 nack_status	Not acknowledge error status bit.Error on I2C not acknowledge. Value after Reset: 0b
3 arbitration_pol	Arbitration error interrupt polarity configuration. Value after Reset: 1b
2 arbitration_mask	Arbitration error interrupt mask signal. Value after Reset: 0b
1 arbitration_ interrupt	Arbitration error interrupt bit.{arbitration_interrupt = (arbitration_mask==0b)&& (arbitration_status==arbitration_pol)}. Value after Reset: 0b
0 arbitration_status	Arbitration error status bit. Error on master I2C protocol arbitration. Value after Reset: 0b

33.5.232 PHY I2C Speed Control Register (HDMI_PHY_I2CM_DIV_ADDR)

This register wets the I2C Master PHY to work in either Fast or Standard mode.

- Address Offset: 0x3029
- Size: 8 bits
- Value after Reset: 0x0B
- Access: Read/Write

Address: 12_0000h base + 3029h offset = 12_3029h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				fast_mode			
Reset	0	0	0	0	1	0	1	1

HDMI_PHY_I2CM_DIV_ADDR field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
fast_mode	Sets the I2C Master to work in Fast Mode or Standard Mode (x implies that it can take any value) Value after Reset: 1011b 1xxxb Fast Mode 0xxxb Standard Mode

33.5.233 PHY I2C Software Reset Register (HDMI_PHY_I2CM_SOFTRSTZ_ADDR)

This register sets the I2C Master PHY software reset.

- Address Offset: 0x302A
- Size: 8 bits
- Value after Reset: 0x01
- Access: Read/Write

The following *CNT registers must be set before any I2C bus transaction can take place to ensure proper I/O timing.

The following are the I2C Master SCL clock settings:

- SS: Standard Speed

- FS: Fast Speed
- HCNT: SCL High Level counter
- LCNT: SCL Low Level counter

Address: 12_0000h base + 302Ah offset = 12_302Ah

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							i2c_softrst
Reset	0	0	0	0	0	0	0	1

HDMI_PHY_I2CM_SOFTRSTZ_ADDR field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 i2c_softrst	I2C Master PHY Software Reset. Active by writing a zero and auto cleared to one in the following cycle. Value after Reset: 1b

33.5.234 PHY I2C Slow Speed SCL High Level Control Register 1 (HDMI_PHY_I2CM_SS_SCL_HCNT_1_ADDR)

- Address Offset: 0x302B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 302Bh offset = 12_302Bh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_hcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_SS_SCL_HCNT_1_ADDR field descriptions

Field	Description
i2cmp_ss_scl_ hcnt[15:8]	Value after Reset: 8'h00

33.5.235 PHY I2C Slow Speed SCL High Level Control Register 0 (HDMI_PHY_I2CM_SS_SCL_HCNT_0_ADDR)

- Address Offset: 0x302C
- Size: 8 bits
- Value after Reset: 0x6C
- Access: Read/Write

Address: 12_0000h base + 302Ch offset = 12_302Ch

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_hcnt[7:0]							
Reset	0	1	1	0	1	1	0	0

HDMI_PHY_I2CM_SS_SCL_HCNT_0_ADDR field descriptions

Field	Description
i2cmp_ss_scl_ hcnt[7:0]	Value after Reset: 8'h6C

33.5.236 PHY I2C Slow Speed SCL Low Level Control Register 1 (HDMI_PHY_I2CM_SS_SCL_LCNT_1_ADDR)

- Address Offset: 0x302D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 302Dh offset = 12_302Dh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_lcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_SS_SCL_LCNT_1_ADDR field descriptions

Field	Description
i2cmp_ss_scl_ lcnt[15:8]	Value after Reset: 8'h00

33.5.237 PHY I2C Slow Speed SCL Low Level Control Register 0 (HDMI_PHY_I2CM_SS_SCL_LCNT_0_ADDR)

- Address Offset: 0x302E
- Size: 8 bits
- Value after Reset: 0x7F
- Access: Read/Write

Address: 12_0000h base + 302Eh offset = 12_302Eh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_lcnt[7:0]							
Reset	0	1	1	1	1	1	1	1

HDMI_PHY_I2CM_SS_SCL_LCNT_0_ADDR field descriptions

Field	Description
i2cmp_ss_scl_ lcnt[7:0]	Value after Reset: 8'h7F

33.5.238 PHY I2C Fast Speed SCL High Level Control Register 1 (HDMI_PHY_I2CM_FS_SCL_HCNT_1_ADDR)

- Address Offset: 0x302F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 302Fh offset = 12_302Fh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_hcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_FS_SCL_HCNT_1_ADDR field descriptions

Field	Description
i2cmp_fs_scl_ hcnt[15:8]	Value after Reset: 8'h00

33.5.239 PHY I2C Fast Speed SCL High Level Control Register 0 (HDMI_PHY_I2CM_FS_SCL_HCNT_0_ADDR)

- Address Offset: 0x3030
- Size: 8 bits
- Value after Reset: 0x11
- Access: Read/Write

Address: 12_0000h base + 3030h offset = 12_3030h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_hcnt[7:0]							
Reset	0	0	0	1	0	0	0	1

HDMI_PHY_I2CM_FS_SCL_HCNT_0_ADDR field descriptions

Field	Description
i2cmp_fs_scl_ hcnt[7:0]	Value after Reset: 8'h11

33.5.240 PHY I2C Fast Speed SCL Low Level Control Register 1 (HDMI_PHY_I2CM_FS_SCL_LCNT_1_ADDR)

- Address Offset: 0x3031
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3031h offset = 12_3031h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_lcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_I2CM_FS_SCL_LCNT_1_ADDR field descriptions

Field	Description
i2cmp_fs_scl_ lcnt[15:8]	Value after Reset: 8'h00

33.5.241 PHY I2C Fast Speed SCL Low Level Control Register 0 (HDMI_PHY_I2CM_FS_SCL_LCNT_0_ADDR)

- Address Offset: 0x3032
- Size: 8 bits
- Value after Reset: 0x24
- Access: Read/Write

Address: 12_0000h base + 3032h offset = 12_3032h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_lcnt[7:0]							
Reset	0	0	1	0	0	1	0	0

HDMI_PHY_I2CM_FS_SCL_LCNT_0_ADDR field descriptions

Field	Description
i2cmp_fs_scl_ lcnt[7:0]	Value after Reset: 8'h24

33.5.242 Audio Clock Regenerator N Value Register 1 (HDMI_AUD_N1)

For N expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3200
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3200h offset = 12_3200h

Bit	7	6	5	4	3	2	1	0
Read Write	AudN[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_N1 field descriptions

Field	Description
AudN[7:0]	HDMI Audio Clock Regenerator N value

33.5.243 Audio Clock Regenerator N Value Register 2 (HDMI_AUD_N2)

For N expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3201
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3201h offset = 12_3201h

Bit	7	6	5	4	3	2	1	0
Read Write	AudN[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_N2 field descriptions

Field	Description
AudN[15:8]	HDMI Audio Clock Regenerator N value

33.5.244 Audio Clock Regenerator N Value Register 3 (HDMI_AUD_N3)

For N expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3202
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3202h offset = 12_3202h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				AudN[19:16]			
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_N3 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
AudN[19:16]	HDMI Audio Clock Regenerator N value

33.5.245 AUD_CTS1 (HDMI_AUD_CTS1)

For CTS expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3203
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3203h offset = 12_3203h

Bit	7	6	5	4	3	2	1	0
Read Write	audCTS[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_CTS1 field descriptions

Field	Description
audCTS[7:0]	HDMI Audio Clock Regenerator CTS calculated value. This value can be manually set using the CTS_manual (AUD_CTS3) mechanism.

33.5.246 AUD_CTS2 (HDMI_AUD_CTS2)

For CTS expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3204
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3204h offset = 12_3204h

Bit	7	6	5	4	3	2	1	0
Read Write	audCTS[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_CTS2 field descriptions

Field	Description
audCTS[15:8]	HDMI Audio Clock Regenerator CTS calculated value. This value can be manually set using the CTS_manual (AUD_CTS3) mechanism.

33.5.247 AUD_CTS3 (HDMI_AUD_CTS3)

For CTS expected values, refer to the HDMI 1.4a specification.

- Address Offset: 0x3205
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3205h offset = 12_3205h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				audCTS[19:16]			
Reset	0	0	0	0	0	0	0	0

HDMI_AUD_CTS3 field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
audCTS[19:16]	HDMI Audio Clock Regenerator CTS calculated value. This value can be manually set using the CTS_manual (AUD_CTS3) mechanism.

33.5.248 Audio DMA Start Register (HDMI_AHB_DMA_CONF0)

This register contains the software reset bit for the audio FIFOs. It also configures operating modes of the AHB master.

- Address Offset: 0x3600
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3600h offset = 12_3600h

Bit	7	6	5	4	3	2	1	0
Read Write	sw_fifo_rst	Reserved		hbr	enable_ hlock	incr_type[1:0]		burst_mode
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_CONF0 field descriptions

Field	Description
7 sw_fifo_rst	This is the software reset bit for the audio and FIFOs clear. Writing 0'b does not result in any action.

Table continues on the next page...

HDMI_AHB_DMA_CONF0 field descriptions (continued)

Field	Description
	Writing 1'b to this register resets all audio FIFOs. Reading from this register always returns 0'b.
6–5 -	This field is reserved. Reserved
4 hbr	HBR packets enable. The HDMI TX sends the HBR packets. This bit is enabled when the audio frequency is higher than 192 KHz. If this bit is enabled, the number of channels configured in AHB_DMA_CONF1 is always 8.
3 enable_hlock	Enable request of locked burst AHB mechanism. 1 Enables the usage of ohlock for master request to arbiter of a locked complete burst.\ 0 Disables request of locked burst AHB mechanism
2–1 incr_type[1:0]	Forced size burst mode. 00 Corresponds to INCR4 fixed four beat incremental AHB burst mode. Only valid when burst_mode is high. 01 Corresponds to INCR8 fixed eight beat incremental AHB burst mode. Only valid when burst_mode is high. 10 Corresponds to INCR16 fixed 16 beat incremental AHB burst mode. Only valid when burst_mode is high. 11 Corresponds to INCR16 fixed 16 beat incremental AHB burst mode. Only valid when burst_mode is high.
0 burst_mode	Burst mode bit 1 Forces the burst mode to be fixed beat incremental burst mode designated by the incr_type[1:0] signal. 0 Normal operation is unspecified length incremental burst. It corresponds to INCR AHB burst mode.

33.5.249 AHB_DMA_START (HDMI_AHB_DMA_START)

The data_ buffer_ready bit field signals the AHB audio DMA to start accessing system memory in order to fetch data samples to store in the FIFO. After the operation starts, a new request for a DMA start is ignored until the DMA is stopped or it reaches the end address. Only in one of these situations will a new start request be acknowledged.

The first DMA burst request after data_buffer_ready configuration uses the initial_addr[31:0] as the ohaddr[31:0] and the MBURSTLENGTH[10:0] = AUDIO_FIFO_DEPTH if AUDIO_FIFO_DEPTH < 1024 or MBURSTLENGTH[10:0] = 1024 if AUDIO_FIFO_DEPTH >= 1024.

- Address Offset: 0x3601
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3601h offset = 12_3601h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							data_buffer_ready
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_START field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 data_buffer_ready	Data buffer ready

33.5.250 Audio DMA Stop Register (HDMI_AHB_DMA_STOP)

The stop_dma_transaction bit field signals the AHB audio DMA to stop current memory access. After it stops, if a new start DMA operation is requested, the DMA engine restarts the memory access assuming the initial_addr[31:0] is programmed at AHB_DMA_STRADDR0 to AHB_DMA_STRADDR3.

- Address Offset: 0x3602
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3602h offset = 12_3602h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							stop_dma_transaction
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STOP field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 stop_dma_transaction	Stop DMA transaction

33.5.251 Audio DMA FIFO Threshold Register (HDMI_AHB_DMA_THRSLD)

This register defines the FIFO medium threshold occupation value.

After the AHB master successfully completes a burst transaction, the FIFO may stay remain full until the data fetch interface requests samples. The sample request from the FIFO using the data fetch mechanism drops the number of samples stored in the audio FIFO.

As soon as the number of samples in the FIFO drops lower than the fifo_threshold[7:0], the DMA engine requests a new burst of samples to the AHB master with a size (MBURSTLENGTH[10:0]) equal to AUDIO_FIFO_DEPTH minus fifo_threshold[7:0].

Therefore, the fifo_threshold[7:0] is the medium number of samples that should be available in the audio FIFO across the DMA operation.

- Address Offset: 0x3603
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3603h offset = 12_3603h

Bit	7	6	5	4	3	2	1	0
Read Write	fifo_threshold[7]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_THRSLD field descriptions

Field	Description
fifo_threshold[7]	FIFO medium threshold occupation value

33.5.252 Audio DMA Start Address Register 0 (HDMI_AHB_DMA_STRADDR0)

These registers define the initial_addr[31:0] used to initiate the DMA burst read transactions upon data_buffer_ready configuration.

- Address Offset: 0x3604 to 0x3607
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3604h offset = 12_3604h

Bit	7	6	5	4	3	2	1	0
Read Write	initial_addr[7]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STRADDR0 field descriptions

Field	Description
initial_addr[7]	Defines init_addr[7:0] for bits 7-0 to initiate DMA burst transactions

33.5.253 Audio DMA Start Address Register 1 (HDMI_AHB_DMA_STRADDR1)

Address: 12_0000h base + 3605h offset = 12_3605h

Bit	7	6	5	4	3	2	1	0
Read Write	initial_addr[15]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STRADDR1 field descriptions

Field	Description
initial_addr[15]	Defines init_addr[15:8] for bits 7-0 to initiate DMA burst transactions

33.5.254 Audio DMA Start Address Register 2 (HDMI_AHB_DMA_STRADDR2)

Address: 12_0000h base + 3606h offset = 12_3606h

Bit	7	6	5	4	3	2	1	0
Read Write	initial_addr[23]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STRADDR2 field descriptions

Field	Description
initial_addr[23]	Defines init_addr[23:16] for bits 7-0 to initiate DMA burst transactions

33.5.255 Audio DMA Start Address Register 3 (HDMI_AHB_DMA_STRADDR3)

Address: 12_0000h base + 3607h offset = 12_3607h

Bit	7	6	5	4	3	2	1	0
Read Write	initial_addr[31]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STRADDR3 field descriptions

Field	Description
initial_addr[31]	Defines init_addr[31:24] for bits 7-0 to initiate DMA burst transactions

33.5.256 Audio DMA Stop Address Register 0 (HDMI_AHB_DMA_STPADDR0)

This registers define the final_addr[31:0] used as the final point to the DMA burst read transactions.

Upon data_buffer_ready configuration, the DMA engine starts requesting burst reads from the external system memory. Each burst read can have a maximum theoretical length of 1024 words (due to the AMBA AHB specification restriction). As an example, if the first burst transaction of the AHB audio DMA has a length of 16, then the second burst starts at address ohaddr[31:0] = initial_addr[31:0] + 16 and has a length of MBURSTLENGTH[10:0] = AUDIO_FIFO_DEPTH - fifo_threshold[7:0].

The DMA engine is responsible for incrementing the burst starting address and defining its corresponding burst length to reach the final_addr[31:0] address. The last burst request issued by the DMA engine takes into account that it should only request data until the final_addr[31:0] address (included) and for that should calculate the correct burst length.

After reaching the final_addr[31:0] address, the done interrupt is active to signal completion of DMA operation.

- Address Offset: 0x3608 to 0x360B
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3608h offset = 12_3608h

Bit	7	6	5	4	3	2	1	0
Read Write	final_addr[7]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STPADDR0 field descriptions

Field	Description
final_addr[7]	Defines final_addr[7:0] for bits 7-0 to initiate DMA burst transactions

33.5.257 Audio DMA Stop Address Register 1 (HDMI_AHB_DMA_STPADDR1)

Address: 12_0000h base + 3609h offset = 12_3609h

Bit	7	6	5	4	3	2	1	0
Read Write	final_addr[15]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STPADDR1 field descriptions

Field	Description
final_addr[15]	Defines final_addr[15:8] for bits 7-0 to initiate DMA burst transactions

33.5.258 Audio DMA Stop Address Register 2 (HDMI_AHB_DMA_STPADDR2)

Address: 12_0000h base + 360Ah offset = 12_360Ah

Bit	7	6	5	4	3	2	1	0
Read Write	final_addr[23]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STPADDR2 field descriptions

Field	Description
final_addr[23]	Defines final_addr[23:16] for bits 7-0 to initiate DMA burst transactions

33.5.259 Audio DMA Stop Address Register 3 (HDMI_AHB_DMA_STPADDR3)

Address: 12_0000h base + 360Bh offset = 12_360Bh

Bit	7	6	5	4	3	2	1	0
Read Write	final_addr[31]							
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_STPADDR3 field descriptions

Field	Description
final_addr[31]	Defines final_addr[31:24] for bits 7-0 to initiate DMA burst transactions

33.5.260 Audio DMA Burst Start Address Register 0 (HDMI_AHB_DMA_BSTADDR0)

This read-only register composes the start address of the current burst operation. As an example, if the first burst transaction of the AHB audio DMA as a length of 16, then the second burst should start at address ohaddr[31:0] = initial_addr[31:0] + 16. While this burst is being executed, burst_start_addr[31:0] = haddr[31:0] = initial_addr[31:0] + 16.

- Address Offset: 0x360C to 0x360F
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 360Ch offset = 12_360Ch

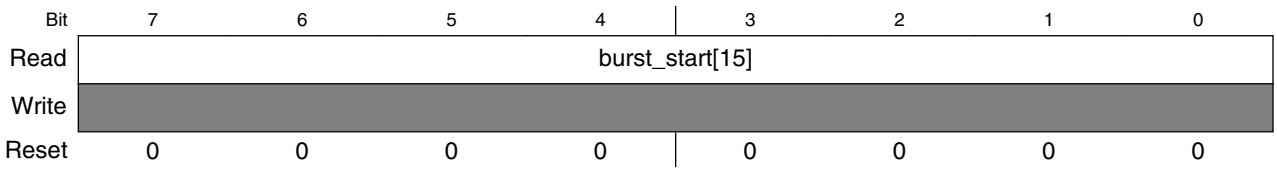
Bit	7	6	5	4	3	2	1	0
Read	burst_start[7]							
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_BSTADDR0 field descriptions

Field	Description
burst_start[7]	Start address for the current burst operation

33.5.261 Audio DMA Burst Start Address Register 1 (HDMI_AHB_DMA_BSTADDR1)

Address: 12_0000h base + 360Dh offset = 12_360Dh

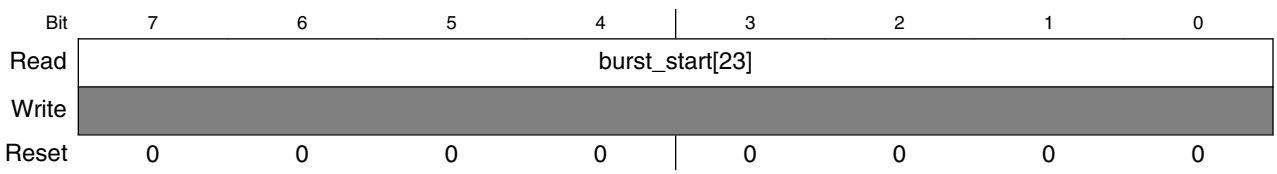


HDMI_AHB_DMA_BSTADDR1 field descriptions

Field	Description
burst_start[15]	Start address for the current burst operation

33.5.262 Audio DMA Burst Start Address Register 2 (HDMI_AHB_DMA_BSTADDR2)

Address: 12_0000h base + 360Eh offset = 12_360Eh

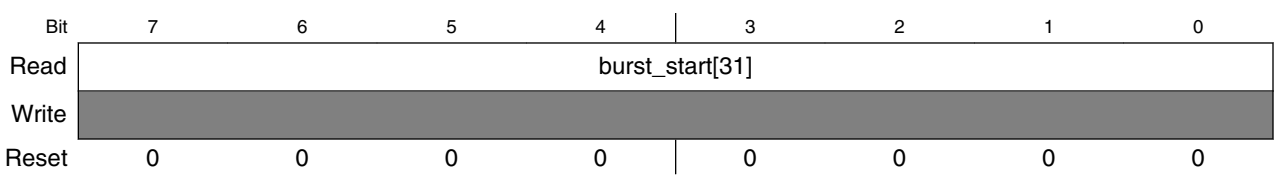


HDMI_AHB_DMA_BSTADDR2 field descriptions

Field	Description
burst_start[23]	Start address for the current burst operation

33.5.263 Audio DMA Burst Start Address Register 3 (HDMI_AHB_DMA_BSTADDR3)

Address: 12_0000h base + 360Fh offset = 12_360Fh



HDMI_AHB_DMA_BSTADDR3 field descriptions

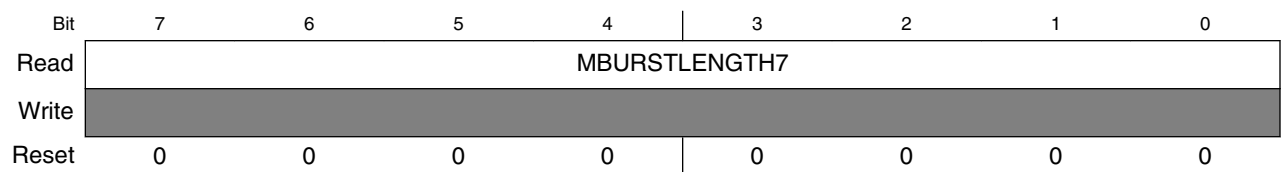
Field	Description
burst_start[31]	Start address for the current burst operation

33.5.264 Audio DMA Burst Length Register 0 (HDMI_AHB_DMA_MBLENGTH0)

These registers hold the length of the current burst operation. As an example, if the first burst transaction of the AHB audio DMA is a length of 8, then the second burst should start at address ohaddr[31:0] = initial_addr[31:0] + 8. It will also have length MBURSTLENGTH[10:0] = AUDIO_FIFO_DEPTH - fifo_threshold[7:0] while this burst is being executed, MBURSTLENGTH[10:0] = AUDIO_FIFO_DEPTH - fifo_threshold[7:0].

- Address Offset: 0x3610 to 0x3611
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3610h offset = 12_3610h



HDMI_AHB_DMA_MBLENGTH0 field descriptions

Field	Description
MBURSTLENGTH7	Requested burst length

33.5.265 Audio DMA Burst Length Register 1 (HDMI_AHB_DMA_MBLENGTH1)

Address: 12_0000h base + 3611h offset = 12_3611h

Bit	7	6	5	4
Read	Reserved			
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved	MBURSTLENGTH10	MBURSTLENGTH9	MBURSTLENGTH8
Write				
Reset	0	0	0	0

HDMI_AHB_DMA_MBLENGTH1 field descriptions

Field	Description
7–3 -	This field is reserved. Reserved
2 MBURSTLENGTH10	Requested burst length
1 MBURSTLENGTH9	Requested burst length
0 MBURSTLENGTH8	Requested burst length

33.5.266 Audio DMA Interrupt Status Register (HDMI_AHB_DMA_STAT)

This register contains the status bits of the following interrupts:

- Address Offset: 0x3612
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3612h offset = 12_3612h

Bit	7	6	5	4
Read	statdone	statretrysplit	statlostownership	staterror
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved	statthrfifoempty	statfifofull	statfifoempty
Write				
Reset	0	0	0	0

HDMI_AHB_DMA_STAT field descriptions

Field	Description
7 statdone	Status of DMA end of operation interrupt. Active when DMA engine reaches final_addr[15:0] or when stop DMA operation is activated.
6 statretrysplit	Status of retry/split interrupt. Active when AHB master receives a RETRY or SPLIT response from slave.
5 statlostownership	Status of master lost ownership when in burst transfer. Active when AHB master loses BUS ownership within the course of a burst transfer.
4 staterror	Status of error interrupt. Active when slave indicates error through the isresp[1:0].
3 -	This field is reserved. Reserved
2 statthrfifoempty	Status of audio FIFO empty when audio FIFO has less than four samples.
1 statfifofull	Status of audio FIFO full interrupt.
0 statfifoempty	Status of audio FIFO empty interrupt.

33.5.267 Audio DMA Interrupt Register (HDMI_AHB_DMA_INT)

This register contains the interrupt bits of the following interrupts:

- Address Offset: 0x3613
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3613h offset = 12_3613h

Bit	7	6	5	4
Read	intdone	intretrysplit	intlostownership	interror
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	Reserved	intthrfifoempty	intfifofull	intfifoempty
Write				
Reset	0	0	0	0

HDMI_AHB_DMA_INT field descriptions

Field	Description
7 intdone	DMA end of operation interrupt. Active when DMA engine reaches final_addr[15:0] or when stop DMA operation is activated.
6 intretrysplit	Retry/split interrupt. Active when AHB master receives a RETRY or SPLIT response from slave.
5 intlostownership	Master lost ownership interrupt when in burst transfer. Active when AHB master loses BUS ownership within the course of a burst transfer.
4 interror	Error interrupt. Active when slave indicates error through the isresp[1:0].
3 -	This field is reserved. Reserved
2 intthrfifoempty	Audio FIFO empty interrupt when audio FIFO has less than four samples.
1 intfifofull	Audio FIFO full interrupt.
0 intfifoempty	Audio FIFO empty interrupt.

33.5.268 Audio DMA Mask Interrupt Register (HDMI_AHB_DMA_MASK)

Mask for each of the interrupts present in the AHB audio DMA module. For usage information, see Audio DMA Interrupt Register (HDMI_AHB_DMA_INT) ."

- Address Offset: 0x3614
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3614h offset = 12_3614h

Bit	7	6	5	4
Read Write	done_mask	retrysplit_mask	lostownership_mask	error_mask
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	Reserved	fifo_thrempty_mask	fifo_full_mask	fifo_empty_mask
Reset	0	0	0	0

HDMI_AHB_DMA_MASK field descriptions

Field	Description
7 done_mask	DMA end of operation interrupt mask. Active when DMA engine reaches final_addr[15:0] or when stop DMA operation is activated.

Table continues on the next page...

HDMI_AHB_DMA_MASK field descriptions (continued)

Field	Description
6 retrysplit_mask	Retry/split interrupt mask. Active when AHB master receives a RETRY or SPLIT response from slave.
5 lostownership_ mask	Master lost ownership interrupt mask when in burst transfer. Active when AHB master loses BUS ownership within the course of a burst transfer.
4 error_mask	Error interrupt mask. Active when slave indicates error through the isresp[1:0].
3 -	This field is reserved. Reserved
2 fifo_thrempty_ mask	Audio FIFO empty interrupt mask when audio FIFO has less than four samples.
1 fifo_full_mask	Audio FIFO full interrupt mask.
0 fifo_empty_mask	Audio FIFO empty interrupt mask.

33.5.269 Audio DMA Polarity Interrupt Register (HDMI_AHB_DMA_POL)

Polarity for each of the interrupts present in the AHB audio DMA module. For usage information, see Audio DMA Interrupt Register (HDMI_AHB_DMA_INT) ."

- Address Offset: 0x3615
- Size: 8 bits per register
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3615h offset = 12_3615h

Bit	7	6	5	4
Read Write	done_polarity	retrysplit_polarity	lostownership_polarity	error_polarity
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	Reserved	fifo_thrfifoempty_ polarity	fifo_full_polarity	fifo_empty_polarity
Reset	0	0	0	0

HDMI_AHB_DMA_POL field descriptions

Field	Description
7 done_polarity	DMA end of operation interrupt mask. Active when DMA engine reaches final_addr[15:0] or when stop DMA operation is activated.

Table continues on the next page...

HDMI_AHB_DMA_POL field descriptions (continued)

Field	Description
6 retrysplit_polarity	Retry/split interrupt mask. Active when AHB master receives a RETRY or SPLIT response from slave.
5 lostownership_ polarity	Master lost ownership interrupt mask when in burst transfer. Active when AHB master loses BUS ownership within the course of a burst transfer.
4 error_polarity	Error interrupt mask. Active when slave indicates error through the isresp[1:0].
3 -	This field is reserved. Reserved
2 fifo_thrfifoempty_ polarity	Audio FIFO empty interrupt mask when audio FIFO has less than four samples.
1 fifo_full_polarity	Audio FIFO full interrupt mask.
0 fifo_empty_ polarity	Audio FIFO empty interrupt mask.

33.5.270 Audio DMA Channel Enable Configuration Register 1 (HDMI_AHB_DMA_CONF1)

- Address Offset: 0x3616
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3616h offset = 12_3616h

Bit	7	6	5	4	3	2	1	0
Read Write	CH_IN_EN7	CH_IN_EN6	CH_IN_EN5	CH_IN_EN4	CH_IN_EN3	CH_IN_EN2	CH_IN_EN1	CH_IN_EN0
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_CONF1 field descriptions

Field	Description
7 CH_IN_EN7	Channel 7 enable bit 1 Channel enabled 0 Channel disabled
6 CH_IN_EN6	Channel 6 enable bit 1 Channel enabled 0 Channel disabled

Table continues on the next page...

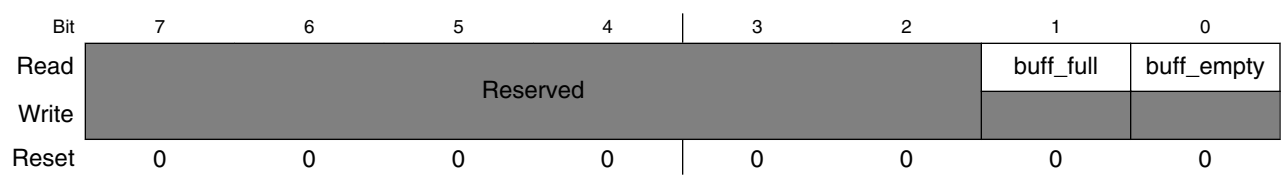
HDMI_AHB_DMA_CONF1 field descriptions (continued)

Field	Description
5 CH_IN_EN5	Channel 5 enable bit 1 Channel enabled 0 Channel disabled
4 CH_IN_EN4	Channel 4 enable bit 1 Channel enabled 0 Channel disabled
3 CH_IN_EN3	Channel 3 enable bit 1 Channel enabled 0 Channel disabled
2 CH_IN_EN2	Channel 2 enable bit 1 Channel enabled 0 Channel disabled
1 CH_IN_EN1	Channel 1 is always enabled.
0 CH_IN_EN0	Channel 0 is always enabled.

33.5.271 Audio DMA Buffer Interrupt Status Register (HDMI_AHB_DMA_BUFFSTAT)

- Address Offset: 0x3617
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3617h offset = 12_3617h



HDMI_AHB_DMA_BUFFSTAT field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 buff_full	Buffer full flag status

Table continues on the next page...

HDMI_AHB_DMA_BUFFSTAT field descriptions (continued)

Field	Description
0 buff_empty	Buffer empty flag status

33.5.272 Audio DMA Buffer Interrupt Register (HDMI_AHB_DMA_BUFFINT)

- Address Offset: 0x3618
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 3618h offset = 12_3618h

Bit	7	6	5	4	3	2	1	0
Read	Reserved						int_buff_full	int_buff_empty
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_BUFFINT field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 int_buff_full	Buffer full flag interrupt
0 int_buff_empty	Buffer empty flag interrupt

33.5.273 Audio DMA Buffer Mask Interrupt Register (HDMI_AHB_DMA_BUFFMASK)

- Address Offset: 0x3619
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 3619h offset = 12_3619h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						int_buff_full	int_buff_ empty
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_BUFFMASK field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 int_buff_full	Buffer full flag mask
0 int_buff_empty	Buffer empty flag mask

33.5.274 Audio DMA Buffer Polarity Interrupt Register (HDMI_AHB_DMA_BUFFPOL)

- Address Offset: 0x361A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 361Ah offset = 12_361Ah

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						int_buff_full	int_buff_ empty
Reset	0	0	0	0	0	0	0	0

HDMI_AHB_DMA_BUFFPOL field descriptions

Field	Description
7–2 -	This field is reserved. Reserved
1 int_buff_full	Buffer full flag polarity
0 int_buff_empty	Buffer empty flag polarity

33.5.275 Main Controller Synchronous Clock Domain Disable Register (HDMI_MC_CLKDIS)

Main controller synchronous disable control per clock domain. Upon release of synchronous disable the corresponding sw reset NRZ request signal, to that domain, is toggled asking to the output for a synchronized active low reset to be generated to that domain.

- Address Offset: 0x4001
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4001h offset = 12_4001h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	Reserved	cecclk_ disable	cscclk_ disable	audclk_ disable	prepclk_ disable	tmdsclk_ disable	pixelclk_ disable
Reset	0	0	0	0	0	0	0	0

HDMI_MC_CLKDIS field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 hdcpclk_disable	This field is reserved. Reserved
5 cecclk_disable	CEC Engine clock synchronous disable signal.
4 cscclk_disable	Color Space Converter clock synchronous disable signal.
3 audclk_disable	Audio Sampler clock synchronous disable signal.
2 prepclk_disable	Pixel Repetition clock synchronous disable signal.

Table continues on the next page...

HDMI_MC_CLKDIS field descriptions (continued)

Field	Description
1 tmdsclk_disable	TMDS clock synchronous disable signal.
0 pixelclk_disable	Pixel clock synchronous disable signal.

33.5.276 Main Controller Software Reset Register (HDMI_MC_SWRSTZREQ)

Main controller software reset request per clock domain. Writing zero to a bit of this register results in an NRZ signal toggle at sfrclk rate to an output signal that indicates a software reset request. This toggle must be used to generate a synchronized reset to de corresponding domain, with at least 1 clock cycle. Register defaults back to 0xFF.

- Address Offset: 0x4002
- Size: 8 bits
- Value after Reset: 0xFF
- Access: Read/Write

Address: 12_0000h base + 4002h offset = 12_4002h

Bit	7	6	5	4
Read Write	Reserved	cecswrst_req	Reserved	
Reset	1	1	1	1

Bit	3	2	1	0
Read Write	Reserved	prepswrst_req	tmdsswrst_req	pixelswrst_req
Reset	1	1	1	1

HDMI_MC_SWRSTZREQ field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 cecswrst_req	CEC software reset request. Defaults back to 1b after reset request.
5–3 -	This field is reserved. Reserved
2 prepswrst_req	Pixel Repetition clock synchronous disable signal.
1 tmdsswrst_req	TMDS software reset request. Defaults back to 1b after reset request.
0 pixelswrst_req	Pixel software reset request. Defaults back to 1b after reset request.

33.5.277 Main Controller Feed Through Control Register (HDMI_MC_FLOWCTRL)

- Address Offset: 0x4004
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4004h offset = 12_4004h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							Feed_ through_off
Reset	0	0	0	0	0	0	0	0

HDMI_MC_FLOWCTRL field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 Feed_through_off	Video path Feed Through enable bit: 1 Color Space Converter is in the video data path. 0 Color Space Converter is bypassed (not in the video data path).

33.5.278 Main Controller PHY Reset Register (HDMI_MC_PHYRSTZ)

- Address Offset: 0x4005
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4005h offset = 12_4005h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							phyrstz
Reset	0	0	0	0	0	0	0	0

HDMI_MC_PHYRSTZ field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 phyrstz	HDMI Source PHY active low reset control.

33.5.279 Main Controller Clock Present Register (HDMI_MC_LOCKONCLOCK)

- Address Offset: 0x4006
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Clear on Write

Address: 12_0000h base + 4006h offset = 12_4006h

Bit	7	6	5	4	3	2	1	0
Read	Reserved	pclk	tclktclk	prepclk	Reserved			cecclk
Write		w1c	w1c	w1c				w1c
Reset	0	0	0	0	0	0	0	0

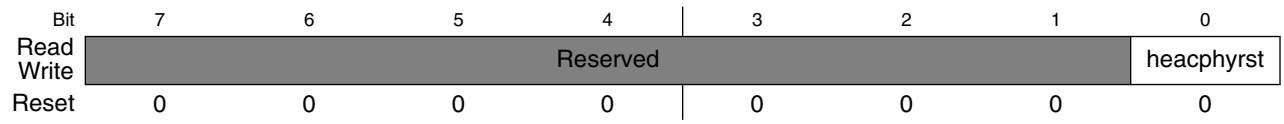
HDMI_MC_LOCKONCLOCK field descriptions

Field	Description
7 -	This field is reserved. Reserved.
6 pclk	Pixel clock status. Indicates that the clock is present in the system. Cleared by WR 1 to this position.
5 tclktclk	TMDS clock status. Indicates that the clock is present in the system. Cleared by WR 1 to this position
4 prepclk	Pixel repetition clock status. Indicates that the clock is present in the system. Cleared by WR 1 to this position.
3–1 -	This field is reserved. Reserved.
0 cecclk	CEC clock status. Indicates that the clock is present in the system. Cleared by WR 1 to this position.

33.5.280 Main Controller HEAC PHY Reset Register (HDMI_MC_HEACPHY_RST)

- Address Offset: 0x4007
- Size: 8 bits
- Value after Reset: N/A
- Access: Read/Write

Address: 12_0000h base + 4007h offset = 12_4007h



HDMI_MC_HEACPHY_RST field descriptions

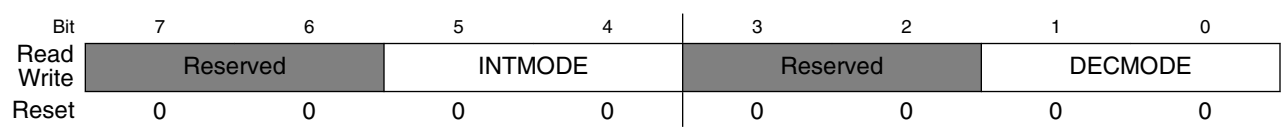
Field	Description
7–1 -	This field is reserved. Reserved
0 heacphyrst	HEAC PHY reset (active high)

33.5.281 Color Space Converter Interpolation and Decimation Configuration Register (HDMI_CSC_CFG)

Color Space Conversion configuration register. Configures YCC422 to YCC444 interpolation mode and YCC444 to YCC422 decimation mode.

- Address Offset: 0x4100
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4100h offset = 12_4100h



HDMI_CSC_CFG field descriptions

Field	Description
7–6 -	This field is reserved. Reserved
5–4 INTMODE	Chroma interpolation configuration: 00 interpolation disabled 01 $H_u(z) = 1 + z^{-1}$ 10 $H_u(z)=1/2 + Z^{-1} +1/2\ z^{-2}$ 11 interpolation disabled.
3–2 -	This field is reserved. Reserved
DECMODE	Chroma decimation configuration: DECMODE[1:0] Chroma decimation 00 decimation disabled 01 H d? z?=1 10 H d? z?=1/ 4?1/ 2 z?1?1/4 z?2 11 H d? z?Þ211=?5?12 z?2?22 z?4?39 z?6?65 z?8?109 z?10?204 z?12?648 z?14?1024 z?15?648 z?16?204 z?18?109 z?20?65 z?22?39 z?24?22 z?26?12 z?28?5 z?30 00 decimation disabled 01 $H_d(z) = 1$ 10 $H_d(Z)=1/4 + 1/2z^{-1} 1+1/4z^{-2}$ $11H_d(z)x2^{11} =-5+12z^{-2} +22z^{-4} +39z^{-8} +109z^{-10} -204z^{-12} +648z^{-14} +1024z^{-15} +648z^{-16} -204z^{-18} +109z^{-20} -65z^{-22} +39z^{-24} -22z^{-26} +12z^{-28} -5z^{-30}$

33.5.282 Color Space Converter Scale and Deep Color Configuration Register (HDMI_CSC_SCALE)

- Address Offset: 0x4101
- Size: 8 bits
- Value after Reset: 0x01
- Access: Read/Write

$$\begin{bmatrix} G \\ R \\ B \end{bmatrix} = 2^{\text{csc scale} - 12} \times \begin{bmatrix} A_1 & A_2 & A_3 \\ B_1 & B_2 & B_3 \\ C_1 & C_2 & C_3 \end{bmatrix} + 2^{\text{csc scale}} \times \begin{bmatrix} A_4 \\ B_4 \\ C_4 \end{bmatrix}$$

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = 2^{\text{cscscale} - 12} \times \begin{bmatrix} A_1 & A_2 & A_3 \\ B_1 & B_2 & B_3 \\ C_1 & C_2 & C_3 \end{bmatrix} + 2^{\text{cscscale}} \times \begin{bmatrix} A_4 \\ B_4 \\ C_4 \end{bmatrix}$$

Figure 33-299. CSC Conversion Functions

Address: 12_0000h base + 4101h offset = 12_4101h

Bit	7	6	5	4	3	2	1	0
Read Write	csc_colorde_pth[3:0]				Reserved		-	
Reset	0	0	0	0	0	0	0	1

HDMI_CSC_SCALE field descriptions

Field	Description
7–4 csc_colorde_pth[3:0]	Color space converter color depth configuration: Other: Reserved. 0000 24 bit per pixel video (8 bit per component). 0100 24 bit per pixel video (8 bit per component). 0101 30 bit per pixel video (10 bit per component). 0110 36 bit per pixel video (12 bit per component). 0111 48 bit per pixel video (16 bit per component).
3–2 -	This field is reserved. Reserved
-	Defines the cscscale[1:0] scale factor to apply to all coefficients in Color Space Conversion. This scale factor is expressed in the number of left shifts to apply to each of the coefficients, ranging from 0 to 2.

33.5.283 CSC_COEF_A1_MSB (HDMI_CSC_COEF_A1_MSB)

Color Space Conversion A1 coefficient.

- Address Offset: 0x4102
- Size: 8 bits
- Value after Reset: 0x20
- Access: Read/Write

Address: 12_0000h base + 4102h offset = 12_4102h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A1_MSB							
Reset	0	0	1	0	0	0	0	0

HDMI_CSC_COEF_A1_MSB field descriptions

Field	Description
CSC_COEF_A1_MSB	Color Space Conversion A1 MSB coefficient.

33.5.284 CSC_COEF_A1_LSB (HDMI_CSC_COEF_A1_LSB)

Color Space Conversion A1 coefficient.

- Address Offset: 0x4103
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4103h offset = 12_4103h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A1_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A1_LSB field descriptions

Field	Description
CSC_COEF_A1_LSB	Color Space Conversion A1 LSB coefficient

33.5.285 CSC_COEF_A2_MSB (HDMI_CSC_COEF_A2_MSB)

Color Space Conversion A2 coefficient.

- Address Offset: 0x4104
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4104h offset = 12_4104h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A2_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A2_MSB field descriptions

Field	Description
CSC_COEF_A2_MSB	Color Space Conversion A2 MSB coefficient.

33.5.286 CSC_COEF_A2_LSB (HDMI_CSC_COEF_A2_LSB)

Color Space Conversion A2 coefficient.

- Address Offset: 0x4105
- Size: 8 bits

- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4105h offset = 12_4105h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A2_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A2_LSB field descriptions

Field	Description
CSC_COEF_A2_LSB	Color Space Conversion A2 LSB coefficient.

33.5.287 CSC_COEF_A3_MSB (HDMI_CSC_COEF_A3_MSB)

Color Space Conversion A3 coefficient.

- Address Offset: 0x4106
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4106h offset = 12_4106h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A3_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A3_MSB field descriptions

Field	Description
CSC_COEF_A3_MSB	Color Space Conversion A3 MSB coefficient.

33.5.288 CSC_COEF_A3_LSB (HDMI_CSC_COEF_A3_LSB)

Color Space Conversion A3 coefficient.

- Address Offset: 0x4107
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4107h offset = 12_4107h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A3_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A3_LSB field descriptions

Field	Description
CSC_COEF_A3_LSB	Color Space Conversion A3 LSB coefficient.

33.5.289 CSC_COEF_A4_MSB (HDMI_CSC_COEF_A4_MSB)

Color Space Conversion A4 coefficient.

- Address Offset: 0x4108
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4108h offset = 12_4108h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A4_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A4_MSB field descriptions

Field	Description
CSC_COEF_A4_MSB	Color Space Conversion A4 MSB coefficient.

33.5.290 CSC_COEF_A4_LSB (HDMI_CSC_COEF_A4_LSB)

Color Space Conversion A4 coefficient.

- Address Offset: 0x4109
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4109h offset = 12_4109h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_A4_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_A4_LSB field descriptions

Field	Description
CSC_COEF_A4_LSB	Color Space Conversion A4 LSB coefficient.

33.5.291 CSC_COEF_B1_MSB (HDMI_CSC_COEF_B1_MSB)

Color Space Conversion B1 coefficient.

- Address Offset: 0x410A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 410Ah offset = 12_410Ah

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B1_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B1_MSB field descriptions

Field	Description
CSC_COEF_B1_MSB	Color Space Conversion B1 MSB coefficient.

33.5.292 CSC_COEF_B1_LSB (HDMI_CSC_COEF_B1_LSB)

Color Space Conversion B1 coefficient.

- Address Offset: 0x410B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 410Bh offset = 12_410Bh

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B1_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B1_LSB field descriptions

Field	Description
CSC_COEF_B1_LSB	Color Space Conversion B1 LSB coefficient.

33.5.293 CSC_COEF_B2_MSB (HDMI_CSC_COEF_B2_MSB)

Color Space Conversion B2 coefficient.

- Address Offset: 0x410C
- Size: 8 bits
- Value after Reset: 0x20
- Access: Read/Write

Address: 12_0000h base + 410Ch offset = 12_410Ch

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B2_MSB							
Reset	0	0	1	0	0	0	0	0

HDMI_CSC_COEF_B2_MSB field descriptions

Field	Description
CSC_COEF_B2_MSB	Color Space Conversion B2 MSB coefficient.

33.5.294 CSC_COEF_B2_LSB (HDMI_CSC_COEF_B2_LSB)

Color Space Conversion B2 coefficient.

- Address Offset: 0x410D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 410Dh offset = 12_410Dh

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B2_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B2_LSB field descriptions

Field	Description
CSC_COEF_B2_LSB	Color Space Conversion B2 LSB coefficient.

33.5.295 CSC_COEF_B3_MSB (HDMI_CSC_COEF_B3_MSB)

Color Space Conversion B3 coefficient.

- Address Offset: 0x410E
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 410Eh offset = 12_410Eh

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B3_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B3_MSB field descriptions

Field	Description
CSC_COEF_B3_MSB	Color Space Conversion B3 MSB coefficient.

33.5.296 CSC_COEF_B3_LSB (HDMI_CSC_COEF_B3_LSB)

Color Space Conversion B3 coefficient.

- Address Offset: 0x410F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 410Fh offset = 12_410Fh

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B3_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B3_LSB field descriptions

Field	Description
CSC_COEF_B3_LSB	Color Space Conversion B3 LSB coefficient.

33.5.297 CSC_COEF_B4_MSB (HDMI_CSC_COEF_B4_MSB)

Color Space Conversion B4 coefficient.

- Address Offset: 0x4110
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4110h offset = 12_4110h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B4_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B4_MSB field descriptions

Field	Description
CSC_COEF_B4_MSB	Color Space Conversion B4 MSB coefficient.

33.5.298 CSC_COEF_B4_LSB (HDMI_CSC_COEF_B4_LSB)

Color Space Conversion B4 coefficient.

- Address Offset: 0x4111
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4111h offset = 12_4111h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_B4_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_B4_LSB field descriptions

Field	Description
CSC_COEF_B4_LSB	Color Space Conversion B4 LSB coefficient.

33.5.299 CSC_COEF_C1_MSB (HDMI_CSC_COEF_C1_MSB)

Color Space Conversion C1 coefficient.

- Address Offset: 0x4112
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4112h offset = 12_4112h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C1_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_C1_MSB field descriptions

Field	Description
CSC_COEF_C1_MSB	Color Space Conversion C1 MSB coefficient.

33.5.300 CSC_COEF_C1_LSB (HDMI_CSC_COEF_C1_LSB)

Color Space Conversion C1 coefficient.

- Address Offset: 0x4113
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4113h offset = 12_4113h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C1_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_C1_LSB field descriptions

Field	Description
CSC_COEF_C1_LSB	Color Space Conversion C1 LSB coefficient.

33.5.301 CSC_COEF_C2_MSB (HDMI_CSC_COEF_C2_MSB)

Color Space Conversion C2 coefficient.

- Address Offset: 0x4114
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4114h offset = 12_4114h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C2_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_C2_MSB field descriptions

Field	Description
CSC_COEF_C2_MSB	Color Space Conversion C2 MSB coefficient.

33.5.302 CSC_COEF_C2_LSB (HDMI_CSC_COEF_C2_LSB)

Color Space Conversion C2 coefficient.

- Address Offset: 0x4115
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4115h offset = 12_4115h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C2_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_C2_LSB field descriptions

Field	Description
CSC_COEF_C2_LSB	Color Space Conversion C2 LSB coefficient.

33.5.303 CSC_COEF_C3_MSB (HDMI_CSC_COEF_C3_MSB)

Color Space Conversion C3 coefficient.

- Address Offset: 0x4116
- Size: 8 bits
- Value after Reset: 0x20
- Access: Read/Write

Address: 12_0000h base + 4116h offset = 12_4116h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C3_MSB							
Reset	0	0	1	0	0	0	0	0

HDMI_CSC_COEF_C3_MSB field descriptions

Field	Description
CSC_COEF_C3_MSB	Color Space Conversion C3 MSB coefficient.

33.5.304 CSC_COEF_C3_LSB (HDMI_CSC_COEF_C3_LSB)

Color Space Conversion C3 coefficient.

- Address Offset: 0x4117
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4117h offset = 12_4117h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEF_C3_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEF_C3_LSB field descriptions

Field	Description
CSC_COEF_C3_LSB	Color Space Conversion C3 LSB coefficient.

33.5.305 CSC_COEFC4_MSB (HDMI_CSC_COEFC4_MSB)

Color Space Conversion C4 coefficient.

- Address Offset: 0x4118
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4118h offset = 12_4118h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEFC4_MSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEFC4_MSB field descriptions

Field	Description
CSC_COEFC4_MSB	Color Space Conversion C4 MSB coefficient.

33.5.306 CSC_COEFC4_LSB (HDMI_CSC_COEFC4_LSB)

Color Space Conversion C4 coefficient.

- Address Offset: 0x4119
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 4119h offset = 12_4119h

Bit	7	6	5	4	3	2	1	0
Read Write	CSC_COEFC4_LSB							
Reset	0	0	0	0	0	0	0	0

HDMI_CSC_COEFC4_LSB field descriptions

Field	Description
CSC_COEFC4_ LSB	Color Space Conversion C4 LSB coefficient.

33.5.307 CEC_CTRL (HDMI_CEC_CTRL)

CEC registers control the CEC feature that is implemented in HDMI TX. They perform various functions like controlling, monitoring, and buffering data for the transmitter and the receiver.

This register handles the main control of the CEC initiator.

- Address Offset: 0x7D00
- Size: 8 bits
- Value after Reset: 0x02
- Access: Read/Write

Address: 12_0000h base + 7D00h offset = 12_7D00h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			STANDBY	BC_NACK	FRAME_TYP		SEND
Reset	0	0	0	0	0	0	1	0

HDMI_CEC_CTRL field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4 STANDBY	Standby bit 0 CEC controller responds the ACK to all messages. 1 CEC controller responds with ACK to all ping messages (only when the EOM is received) and responds with NACK to all other messages, generating wake-up status for selected opcodes. Attention that the NACK will only be posted on the last block of a frame.
3 BC_NACK	Broadcast NACK bit 0 Reset by software to ACK the received broadcast message. 1 Set by software to NACK the received broadcast message. This bit holds till software resets. The broadcasts will be answered with 1'b0. It means the follower reject the message.

Table continues on the next page...

HDMI_CEC_CTRL field descriptions (continued)

Field	Description
2–1 FRAME_TYP	Frame Type bit 00 Signal Free Time = 3-bit periods. Previous attempt to send frame is unsuccessful. 01 Signal Free Time = 5-bit periods. New initiator wants to send a frame. 10 Signal Free Time = 7-bit periods. Present initiator wants to send another frame immediately after its previous frame. (spec CEC 9.1) 11 Illegal value. If software write this value, hardware will set the value to the default 2'b01.
0 SEND	Send bit 0 Reset to 0 by hardware when the CEC transmission is done (no matter successful or failed). It can also work as an indicator checked by software to see whether the transmission is finished. 1 Set by software to trigger CEC sending a frame as an initiator. This bit keeps at 1 while the transmission is going on.

33.5.308 CEC_STAT (HDMI_CEC_STAT)

This register indicates the status of CEC line. All bits are read only. When an event occurs, the corresponding bit will set to 1 for one SFR clock cycle only. Then, the bit automatically resets to 0. No software reset is required. Software can read the "stable" interrupts on IH_CEC_STAT0 register (this register has the same bit arrangement as CEC_STAT register).

- Address Offset: 0x7D01
- Size: 8 bits
- Value after Reset: N/A
- Access: Read

Address: 12_0000h base + 7D01h offset = 12_7D01h

Bit	7	6	5	4	3	2	1	0
Read	Reserved	WAKEUP	ERROR_ FOLL	ERROR_ INIT	ARB_LOST	NACK	EOM	DONE
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_CEC_STAT field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 WAKEUP	Follower received wake-up command (for follower only).
5 ERROR_FOLL	An error is notified by a follower. Abnormal logic data bit error (for follower).

Table continues on the next page...

HDMI_CEC_STAT field descriptions (continued)

Field	Description
4 ERROR_INIT	An error is detected on cec line (for initiator only).
3 ARB_LOST	The initiator losses the CEC line arbitration to a second initiator. (specification CEC 9).
2 NACK	A frame is not acknowledged in a directly addressed message. Or a frame is negatively acknowledged in a broadcast message (for initiator only).
1 EOM	EOM is detected so that the received data is ready in the receiver data buffer (for follower only).
0 DONE	The current transmission is successful (for initiator only).

33.5.309 CEC_MASK (HDMI_CEC_MASK)

This read/write register masks/unmasks the interrupt events. When the bit is set to 1 (masked), the corresponding event will not trigger an interrupt signal at the system interface. When the bit is reset to 0, the interrupt event is unmasked.

- Address Offset: 0x7D02
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7D02h offset = 12_7D02h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	WAKEUP_ MASK	ERROR_ FOLL_ MASK	ERROR_ INIT_MASK	ARB_ LOST_ MASK	NACK_ MASK	EOM_MASK	DONE_ MASK
Reset	0	0	0	0	0	0	0	0

HDMI_CEC_MASK field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 WAKEUP_MASK	Follower wake-up signal mask
5 ERROR_FOLL_ MASK	An error is notified by a follower. Abnormal logic data bit error (for follower).
4 ERROR_INIT_ MASK	An error is detected on cec line (for initiator only).

Table continues on the next page...

HDMI_CEC_MASK field descriptions (continued)

Field	Description
3 ARB_LOST_ MASK	The initiator losses the CEC line arbitration to a second initiator. (specification CEC 9).
2 NACK_MASK	A frame is not acknowledged in a directly addressed message. Or a frame is negatively acknowledged in a broadcast message (for initiator only).
1 EOM_MASK	EOM is detected so that the received data is ready in the receiver data buffer (for follower only).
0 DONE_MASK	The current transmission is successful (for initiator only).

33.5.310 CEC_POLARITY (HDMI_CEC_POLARITY)

This register is readable and writable, which controls the polarity of the interrupt status register as well as the polarity of the interrupt signals at system interface.

- Address Offset: 0x7D03
- Size: 8 bits
- Value after Reset: 0x7F
- Access: Read/Write

Address: 12_0000h base + 7D03h offset = 12_7D03h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	WAKEUP_ POL	ERROR_ FOLL_POL	ERROR_ INIT_POL	ARB_ LOST_POL	NACK_POL	EOM_POL	DONE_POL
Reset	0	1	1	1	1	1	1	1

HDMI_CEC_POLARITY field descriptions

Field	Description
7 - Reserved	This field is reserved.
6 WAKEUP_POL	Follower wakeup signal polarity
5 ERROR_FOLL_ POL	CEC line error polarity (for follower only)
4 ERROR_INIT_ POL	CEC line error polarity (for initiator only)
3 ARB_LOST_POL	Initiator Arbitration lost signal polarity
2 NACK_POL	Frame NACK signal polarity

Table continues on the next page...

HDMI_CEC_POLARITY field descriptions (continued)

Field	Description
1 EOM_POL	EOM detect signal polarity (follower only)
0 DONE_POL	Current transmission success or not signal polarity

33.5.311 CEC_INT (HDMI_CEC_INT)

This register is read only. Each bit of the register is output at the system interface. The output signals hold the active interrupt state (high or low) for only one SFR clock cycle. Then the hardware resets the bit to an inactive state. Software can read the "stable" interrupts on the IH_CEC_STAT0 register at address 0x0106 (this register has the same bit arrangement as the CEC_STAT register).

The functional formula for the interrupts is:

CEC_INT = (CEC_MASK == 0b) && (CEC_STATUS == CEC_POLARITY)

- Address Offset: 0x7D04
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 7D04h offset = 12_7D04h

Bit	7	6	5	4
Read	Reserved	WAKEUP_INT	ERROR_FOLL_INT	ERROR_INIT_INT
Write				
Reset	0	0	0	0

Bit	3	2	1	0
Read	ARB_LOST_INT	NACK_INT	EOM_INT	DONE_INT
Write				
Reset	0	0	0	0

HDMI_CEC_INT field descriptions

Field	Description
7 -	This field is reserved. Reserved
6 WAKEUP_INT	Follower wakeup signal polarity

Table continues on the next page...

HDMI_CEC_INT field descriptions (continued)

Field	Description
5 ERROR_FOLL_INT	Follower wakeup interrupt
4 ERROR_INIT_INT	CEC line error interrupt (for follower only)
3 ARB_LOST_INT	CEC line error interrupt (for initiator only)
2 NACK_INT	Initiator Arbitration lost interrupt
1 EOM_INT	Frame NACK interrupt
0 DONE_INT	EOM detect interrupt (for follower only)

33.5.312 CEC_ADDR_L (HDMI_CEC_ADDR_L)

CEC_ADDR_L and CEC_ADDR_H registers indicate the logical address(es) allocated to the CEC device. The logical address mappings are shown in CEC_ADDR_L (HDMI_CEC_ADDR_L) and CEC_ADDR_H (HDMI_CEC_ADDR_H) . This register is written by software when the logical allocation is finished. Bit value 1 means the corresponding logical address is allocated to this device. Bit value 0 means the corresponding logical address is not allocated to this device.

- Address Offset: 0x7D05
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7D05h offset = 12_7D05h

Bit	7	6	5	4
Read Write	CEC_ADDR_L7	CEC_ADDR_L6	CEC_ADDR_L5	CEC_ADDR_L4
Reset	0	0	0	0

Bit	3	2	1	0
Read Write	CEC_ADDR_L3	CEC_ADDR_L2	CEC_ADDR_L1	CEC_ADDR_L0
Reset	0	0	0	0

HDMI_CEC_ADDR_L field descriptions

Field	Description
7 CEC_ADDR_L7	Logical address 7 - Tuner 3
6 CEC_ADDR_L6	Logical address 6 - Tuner 2
5 CEC_ADDR_L5	Logical address 5 - Audio System
4 CEC_ADDR_L4	Logical address 4 - Playback Device 1
3 CEC_ADDR_L3	Logical address 3 - Tuner 1
2 CEC_ADDR_L2	Logical address 2 - Recording Device 2
1 CEC_ADDR_L1	Logical address 1 - Recording Device 1
0 CEC_ADDR_L0	Logical address 0 - Device TV

33.5.313 CEC_ADDR_H (HDMI_CEC_ADDR_H)

CEC_ADDR_L and CEC_ADDR_H registers indicate the logical address(es) allocated to the CEC device. The logical address mappings are shown in CEC_ADDR_L (HDMI_CEC_ADDR_L) and CEC_ADDR_H (HDMI_CEC_ADDR_H) . This register is written by software when the logical allocation is finished. Bit value 1 means the corresponding logical address is allocated to this device. Bit value 0 means the corresponding logical address is not allocated to this device.

- Address Offset: 0x7D06
- Size: 8 bits
- Value after Reset: 0x80
- Access: Read/Write

Address: 12_0000h base + 7D06h offset = 12_7D06h

Bit	7	6	5	4
Read Write	CEC_ADDR_H7	CEC_ADDR_H6	CEC_ADDR_H5	CEC_ADDR_H4
Reset	1	0	0	0

Bit	3	2	1	0
Read Write	CEC_ADDR_H3	CCEC_ADDR_H2	CEC_ADDR_H1	CEC_ADDR_H0
Reset	0	0	0	0

HDMI_CEC_ADDR_H field descriptions

Field	Description
7 CEC_ADDR_H7	Logical address 15 - Unregistered (as initiator address), Broadcast (as destination address)
6 CEC_ADDR_H6	Logical address 14 - Free use
5 CEC_ADDR_H5	Logical address 13 - Reserved
4 CEC_ADDR_H4	Logical address 12 - Reserved
3 CEC_ADDR_H3	Logical address 11 - Playback Device 3
2 CCEC_ADDR_H2	Logical address 10 - Tuner 4
1 CEC_ADDR_H1	Logical address 9 - Playback Device 3
0 CEC_ADDR_H0	Logical address 8 - Playback Device 2

33.5.314 CEC_TX_CNT (HDMI_CEC_TX_CNT)

This register indicates the size of the frame in bytes (including header and data blocks), which are available in the transmitter data buffer.

When the value is zero, the CEC controller ignores the send command triggered by software. When the transmission is done (no matter success or not), the current value is held until it is overwritten by software.

- Address Offset: 0x7D07
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7D07h offset = 12_7D07h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				CEC_TX_CNT			
Reset	0	0	0	0	0	0	0	0

HDMI_CEC_TX_CNT field descriptions

Field	Description
7–5 -	This field is reserved. Reserved

Table continues on the next page...

HDMI_CEC_TX_CNT field descriptions (continued)

Field	Description
CEC_TX_CNT	CEC Transmitter Counter register: Value after Reset: 5'b00000 0 No data needs to be transmitted. 1 Frame size is 1 byte. 16 Frame size is 16 byte.

33.5.315 CEC_RX_CNT (HDMI_CEC_RX_CNT)

This register indicates the size of the frame in bytes (including header and data blocks), which are available in the receiver data buffer.

Only after the whole receiving process is finished successfully, the counter is refreshed to the value which indicates the total number of data bytes in the Receiver Data Register.

- Address Offset: 0x7d08
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 7D08h offset = 12_7D08h

Bit	7	6	5	4	3	2	1	0
Read	-			CEC_RX_CNT				
Write	-							
Reset	0	0	0	0	0	0	0	0

HDMI_CEC_RX_CNT field descriptions

Field	Description
7–5 -	
CEC_RX_CNT	CEC Receiver Counter register. Value after Reset: 5‘b00000 0 No data received 1 1-byte data is received. 16 16-byte data is received.

33.5.316 CEC_TX_DATA (HDMI_CEC_TX_DATA*n*)

These registers (8 bit each) are the buffers used for storing the data waiting for transmission(including header and data blocks).

- Address Offset: 0x7D10 .. 0x7D1F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 7D10h offset + (1d × i), where i=0d to 15d

Bit	7	6	5	4	3	2	1	0
Read Write	CEC_TX_DATA							
Reset	0	0	0	0	0	0	0	0

HDMI_CEC_TX_DATA*n* field descriptions

Field	Description
CEC_TX_DATA	Header block in CEC_TX_DATA0 Data blockn in CEC_TX_DATAn

33.5.317 CEC_RX_DATA (HDMI_CEC_RX_DATA*n*)

These registers (8 bit each) are the buffers used for storing the received data (including header and data blocks).

- Address Offset: 0x7D20 .. 0x7D2F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 7D20h offset + (1d × i), where i=0d to 15d

Bit	7	6	5	4	3	2	1	0
Read	CEC_RX_DATA							
Write								
Reset	0	0	0	0	0	0	0	0

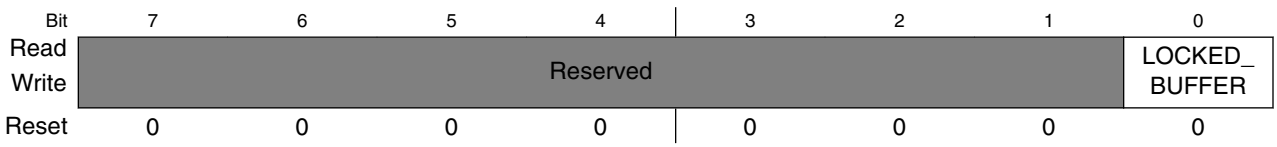
HDMI_CEC_RX_DATA*n* field descriptions

Field	Description
CEC_RX_DATA	Header block in CEC_RX_DATA0 Data blockn in CEC_RX_DATAn

33.5.318 CEC_LOCK (HDMI_CEC_LOCK)

- Address Offset: 0x7D30
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7D30h offset = 12_7D30h



HDMI_CEC_LOCK field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 LOCKED_ BUFFER	When a frame is received, this bit would be active. The CEC controller answers to all the messages with NACK until the CPU writes it to '0'.

33.5.319 CEC_WKUPCTRL (HDMI_CEC_WKUPCTRL)

- Address Offset: 0x7D31
- Size: 8 bits
- Value after Reset: 0xFF
- Access: Read/Write

After receiving a message in the CEC_RX_DATA1 (OPCODE) registers, the CEC engine verifies the message opcode[7:0] against one of the previously defined values to generate the wake-up status:

Wakeupstatus is 1 when:

received opcode is 0x04 and opcode0x04en is 1 or received opcode is 0x0D and opcode0x0Den is 1 or received opcode is 0x41 and opcode0x41en is 1 or received opcode is 0x42 and opcode0x42en is 1 or received opcode is 0x44 and opcode0x44en is 1 or received opcode is 0x70 and opcode0x70en is 1 or received opcode is 0x82 and opcode0x82en is 1 or received opcode is 0x86 and opcode0x86en is 1

Wakeupstatus is 0 when none of the previous conditions are true.

This formula means that the wake-up status (on CEC_STAT[6] register) is only '1' if the opcode[7:0] received is equal to one of the defined values and the corresponding enable bit of that defined value is set to '1'.

Address: 12_0000h base + 7D31h offset = 12_7D31h

Bit	7	6	5	4
Read Write	OPCODE0x86en	OPCODE0x82en	OPCODE0x70en	OPCODE0x44en
Reset	1	1	1	1

Bit	3	2	1	0
Read Write	OPCODE0x42en	OPCODE0x41en	OPCODE0x0Den	OPCODE0x04en
Reset	1	1	1	1

HDMI_CEC_WKUPCTRL field descriptions

Field	Description
7 OPCODE0x86en	OPCODE 0x86 wake up enable
6 OPCODE0x82en	OPCODE 0x82 wake up enable
5 OPCODE0x70en	OPCODE 0x70 wake up enable
4 OPCODE0x44en	OPCODE 0x44 wake up enable
3 OPCODE0x42en	OPCODE 0x42 wake up enable
2 OPCODE0x41en	OPCODE 0x41 wake up enable
1 OPCODE0x0Den	OPCODE 0x0D wake up enable
0 OPCODE0x04en	OPCODE 0x04 wake up enable

33.5.320 I2CM_SLAVE (HDMI_I2CM_SLAVE)

I2C Master Registers (E-DDC) registers are responsible for the Master's coordination with the Slave, by coordinating the Slave address, data identification, transaction status, acknowledgement, and reset functions.

- Address Offset: 0x7E00
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E00h offset = 12_7E00h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	slaveaddr[6:0]						
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_SLAVE field descriptions

Field	Description
7 -	This field is reserved. Reserved
slaveaddr[6:0]	Slave address to be sent during read and write normal operations.

33.5.321 I2CM_ADDRESS (HDMI_I2CM_ADDRESS)

- Address Offset: 0x7E01
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E01h offset = 12_7E01h

Bit	7	6	5	4	3	2	1	0
Read Write	address[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_ADDRESS field descriptions

Field	Description
address[7:0]	Register address for read and write operations.

33.5.322 I2CM_DATAO (HDMI_I2CM_DATAO)

- Address Offset: 0x7E02
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E02h offset = 12_7E02h

Bit	7	6	5	4	3	2	1	0
Read Write	datao[7:0]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_DATAO field descriptions

Field	Description
datao[7:0]	Data to be written on register pointed by address[7:0].

33.5.323 I2CM_DATAI (HDMI_I2CM_DATAI)

- Address Offset: 0x7E03
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read

Address: 12_0000h base + 7E03h offset = 12_7E03h

Bit	7	6	5	4	3	2	1	0
Read	datai[7:0]							
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_DATAI field descriptions

Field	Description
datai[7:0]	Data read from register pointed by address[7:0].

33.5.324 I2CM_OPERATION (HDMI_I2CM_OPERATION)

Read and write operation request. This register can only be written, reading this register will always result in 00h. Writing 1'b1 simultaneously to rd, rd_ext and wr requests is considered as a read (rd) request.

- Address Offset: 0x7E04
- Size: 8 bits
- Value after Reset: 0x00
- Access: Write

Address: 12_0000h base + 7E04h offset = 12_7E04h

Bit	7	6	5	4	3	2	1	0
Read	Reserved				Reserved			
Write				wr			rd_ext	rd
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_OPERATION field descriptions

Field	Description
7–5 -	This field is reserved. Reserved
4 wr	Write operation request.
3–2 -	This field is reserved. Reserved
1 rd_ext	After writing 1'b1 to rd_ext bit a extended data read operation is started (E- DDC read operation).
0 rd	Read operation request.

33.5.325 I2CM_INT (HDMI_I2CM_INT)

This register contains and configures I2C master done interrupt.

- Address Offset: 0x7E05
- Size: 8 bits
- Value after Reset: 0x08
- Access: Read/Write

Address: 12_0000h base + 7E05h offset = 12_7E05h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				done_pol	done_mask	done_ interrupt	done_status
Reset	0	0	0	0	1	0	0	0

HDMI_I2CM_INT field descriptions

Field	Description
7–4 -	This field is reserved. Reserved
3 done_pol	Done interrupt polarity configuration.
2 done_mask	Done interrupt mask signal.
1 done_interrupt	Operation done interrupt bit. Only lasts for 1 SFR clock cycle and is auto cleaned after it. {done_interrupt = (done_mask==0b) && (done_status==done_pol)}.
0 done_status	Operation done status bit. Marks the end of a rd or write operation.

33.5.326 I2CM_CTLINT (HDMI_I2CM_CTLINT)

This register contains and configures I2C master arbitration error and not acknowledge error interrupt.

- Address Offset: 0x7E06
- Size: 8 bits
- Value after Reset: 0x88
- Access: Read/Write

Address: 12_0000h base + 7E06h offset = 12_7E06h

Bit	7	6	5	4
Read Write	nack_pol	nack_mask	nack_interrupt	nack_status
Reset	1	0	0	0

Bit	3	2	1	0
Read Write	arbitration_pol	arbitration_mask	arbitration_interrupt	arbitration_status
Reset	1	0	0	0

HDMI_I2CM_CTLINT field descriptions

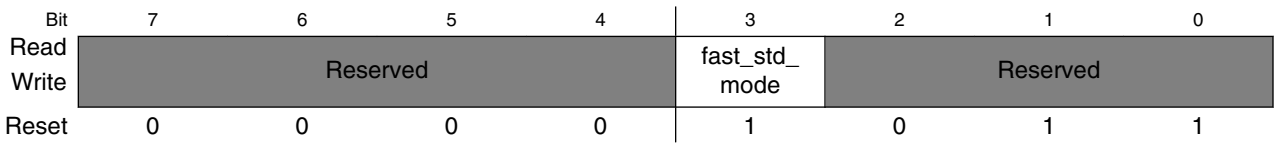
Field	Description
7 nack_pol	Not acknowledge error interrupt polarity configuration.
6 nack_mask	Not acknowledge error interrupt mask signal.
5 nack_interrupt	Not acknowledge error interrupt bit. Only lasts for 1 SFR clock cycle and is auto cleaned after it. {nack_interrupt = (nack_mask==0b) && (nack_status==nack_pol)}.
4 nack_status	Not acknowledge error status bit. Error on I2C not acknowledge.
3 arbitration_pol	Arbitration error interrupt polarity configuration.
2 arbitration_mask	Arbitration error interrupt mask signal.
1 arbitration_ interrupt	Arbitration error interrupt bit. Only lasts for 1 SFR clock cycle and is auto cleaned after it. {arbitration_interrupt = (arbitration_mask==0b) && (arbitration_status==arbitration_pol)}.
0 arbitration_status	Arbitration error status bit. Error on master I2C protocol arbitration.

33.5.327 I2CM_DIV (HDMI_I2CM_DIV)

This register configures the division relation between master and scl clock.

- Address Offset: 0x7E07
- Size: 8 bits
- Value after Reset: 0x0B
- Access: Read/Write

Address: 12_0000h base + 7E07h offset = 12_7E07h



HDMI_I2CM_DIV field descriptions

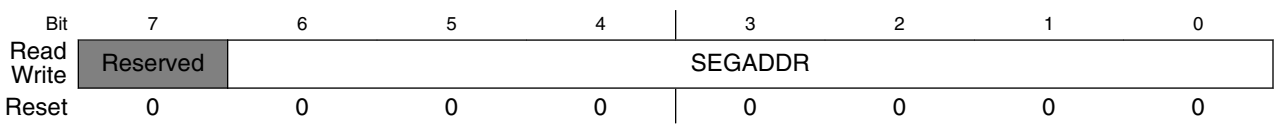
Field	Description
7–4 -	This field is reserved. Reserved
3 fast_std_mode	Sets the I2C Master to work in Fast Mode or Standard Mode: 1 Fast Mode 0 Standard Mode
-	This field is reserved. Reserved

33.5.328 I2CM_SEGADDR (HDMI_I2CM_SEGADDR)

This register configures the segment address for extended RD/WR destination.

- Address Offset: 0x7E08
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E08h offset = 12_7E08h



HDMI_I2CM_SEGADDR field descriptions

Field	Description
7 -	This field is reserved. Reserved
SEGADDR	E-DDC Extended read segment address

33.5.329 I2CM_SOFTRSTZ (HDMI_I2CM_SOFTRSTZ)

This register resets the I2C master.

- Address Offset: 0x7E09
- Size: 8 bits
- Value after Reset: 0x01
- Access: Read/Write

Address: 12_0000h base + 7E09h offset = 12_7E09h

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved							i2c_softrst
Reset	0	0	0	0	0	0	0	1

HDMI_I2CM_SOFTRSTZ field descriptions

Field	Description
7–1 -	This field is reserved. Reserved
0 i2c_softrst	I2C Master Software Reset. Active by writing a zero and auto cleared to one in the following cycle. Value after Reset: 1b

33.5.330 I2CM_SEGPTR (HDMI_I2CM_SEGPTR)

This register configures the segment pointer for extended RD/WR request.

- Address Offset: 0x7E0A
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

The following *CNT registers must be set before any I2C bus transaction can take place to ensure proper I/O timing.

Address: 12_0000h base + 7E0Ah offset = 12_7E0Ah

Bit	7	6	5	4	3	2	1	0
Read Write	I2CM_SEGPTR							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_SEGPTR field descriptions

Field	Description
I2CM_SEGPTR	I2CM_SEGPTR is used for EDID reading operations, particularly for the Extended Data Read Operation (See I²C Master Interface Extended Read Mode ") which is used for Enhanced DDC. This is all described in the VESA Enhanced Display Data Channel Standard v1.1 spec. (addresses A0h/A1h pairs and a segment pointer - 60h).

33.5.331 I2CM_SS_SCL_HCNT_1_ADDR (HDMI_I2CM_SS_SCL_HCNT_1_ADDR)

- Address Offset: 0x7E0B
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E0Bh offset = 12_7E0Bh

Bit	7	6	5	4	3	2	1	0
Read / Write	i2cmp_ss_scl_hcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_SS_SCL_HCNT_1_ADDR field descriptions

Field	Description
i2cmp_ss_scl_hcnt[15:8]	Value after Reset: 8'h00

33.5.332 I2CM_SS_SCL_HCNT_0_ADDR (HDMI_I2CM_SS_SCL_HCNT_0_ADDR)

- Address Offset: 0x7E0C
- Size: 8 bits
- Value after Reset: 0x6C
- Access: Read/Write

Address: 12_0000h base + 7E0Ch offset = 12_7E0Ch

Bit	7	6	5	4	3	2	1	0
Read / Write	i2cmp_ss_scl_hcnt[7:0]							
Reset	0	1	1	0	1	1	0	0

HDMI_I2CM_SS_SCL_HCNT_0_ADDR field descriptions

Field	Description
i2cmp_ss_scl_hcnt[7:0]	Value after Reset: 8'h6C

33.5.333 I2CM_SS_SCL_LCNT_1_ADDR (HDMI_I2CM_SS_SCL_LCNT_1_ADDR)

- Address Offset: 0x7E0D
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E0Dh offset = 12_7E0Dh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_lcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_SS_SCL_LCNT_1_ADDR field descriptions

Field	Description
i2cmp_ss_scl_lcnt[15:8]	Value after Reset: 8'h00

33.5.334 I2CM_SS_SCL_LCNT_0_ADDR (HDMI_I2CM_SS_SCL_LCNT_0_ADDR)

- Address Offset: 0x7E0E
- Size: 8 bits
- Value after Reset: 0x7F
- Access: Read/Write

Address: 12_0000h base + 7E0Eh offset = 12_7E0Eh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_ss_scl_lcnt[7:0]							
Reset	0	1	1	1	1	1	1	1

HDMI_I2CM_SS_SCL_LCNT_0_ADDR field descriptions

Field	Description
i2cmp_ss_scl_lcnt[7:0]	Value after Reset: 8'h7F

33.5.335 I2CM_FS_SCL_HCNT_1_ADDR (HDMI_I2CM_FS_SCL_HCNT_1_ADDR)

- Address Offset: 0x7E0F
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E0Fh offset = 12_7E0Fh

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_hcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_FS_SCL_HCNT_1_ADDR field descriptions

Field	Description
i2cmp_fs_scl_hcnt[15:8]	Value after Reset: 8'h00

33.5.336 I2CM_FS_SCL_HCNT_0_ADDR (HDMI_I2CM_FS_SCL_HCNT_0_ADDR)

- Address Offset: 0x7E10
- Size: 8 bits
- Value after Reset: 0x11
- Access: Read/Write

Address: 12_0000h base + 7E10h offset = 12_7E10h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_hcnt[7:0]							
Reset	0	0	0	1	0	0	0	1

HDMI_I2CM_FS_SCL_HCNT_0_ADDR field descriptions

Field	Description
i2cmp_fs_scl_hcnt[7:0]	Value after Reset: 8'h11

33.5.337 I2CM_FS_SCL_LCNT_1_ADDR (HDMI_I2CM_FS_SCL_LCNT_1_ADDR)

- Address Offset: 0x7E11
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7E11h offset = 12_7E11h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_lcnt[15:8]							
Reset	0	0	0	0	0	0	0	0

HDMI_I2CM_FS_SCL_LCNT_1_ADDR field descriptions

Field	Description
i2cmp_fs_scl_lcnt[15:8]	Value after Reset: 8'h00

33.5.338 I2CM_FS_SCL_LCNT_0_ADDR (HDMI_I2CM_FS_SCL_LCNT_0_ADDR)

- Address Offset: 0x7E12
- Size: 8 bits
- Value after Reset: 0x24
- Access: Read/Write

Address: 12_0000h base + 7E12h offset = 12_7E12h

Bit	7	6	5	4	3	2	1	0
Read Write	i2cmp_fs_scl_lcnt[7:0]							
Reset	0	0	1	0	0	1	0	0

HDMI_I2CM_FS_SCL_LCNT_0_ADDR field descriptions

Field	Description
i2cmp_fs_scl_ lcnt[7:0]	Value after Reset: 8'h24

33.5.339 BASE_POINTER_ADDR (HDMI_BASE_POINTER_ADDR)

The I2C Slave Registers allow register memory pagination, and function in the incremental burst operation mode that increases the data throughput when consecutive addressed registers need to be read or write.

The I2C base pointer operation mode is a aimed to allow register memory pagination. As long as this operational mode is enabled the value written to this register will be used as the seven most significant bits of the internal Special Function Register address interface (sfraddr[14:8]) for all read or write operations. I2C data transfer protocol used shall be the 7-bit addressed as defined in the section 9 of the I2C-bus Specification, version 2.1.

- Address Offset: 0x7F00
- Size: 8 bits
- Value after Reset: 0x00
- Access: Read/Write

Address: 12_0000h base + 7F00h offset = 12_7F00h

Bit	7	6	5	4	3	2	1	0
Read Write	en_base_ pointer_addr	base_pointer_base_addr[6:0]						
Reset	0	0	0	0	0	0	0	0

HDMI_BASE_POINTER_ADDR field descriptions

Field	Description
7 en_base_ pointer_addr	Enables the base pointer operation mode.
base_pointer_ base_addr[6:0]	Defines the base address for base pointer operation mode. They represent the address bits [14:8]

Chapter 34
HDMI 3D Tx PHY (HDMI_PHY)

34.1 Overview

34.1.1 General Description

The HDMI 3D Tx PHY (HDMI_PHY) is the physical layer of a single-link HDMI transmitter interface and comprises three line drivers for data transmission and an additional line driver for clock transmission. The HDMI_PHY is designed to perform the serialization and transmission of video data and control information through an HDMI interface. It interfaces with an HDMI Tx link controller through a common graphic controller interface supporting 30- or 60-bit data transfers. A clock line driver is used for reference clock transmission.

34.1.2 Applications

The HDMI_PHY is targeted to digital video/audio transmission for high resolution display applications, supporting major display formats up to 1080 i/p in DTV applications and QXGA in graphic display applications, with true-color or deep-color resolutions. The HDMI_PHY supports 3D video formats. At maximum pixel rate, the HDMI channel bit rate is 3.4 Gbps, providing a maximum throughput of 9.2 Gbps.

The HDMI_PHY is an ideal solution for implementing the physical layer of a high-reliability digital video/audio interface based on HDMI technology, with reduced line count and minimum cable-wire and EMI shielding requirements. The HDMI_PHY delivers the bandwidth needed in every time mode that demands high resolution panels, while keeping clock sources at a low frequency.

The figure below shows a typical application for HDMI_PHY.

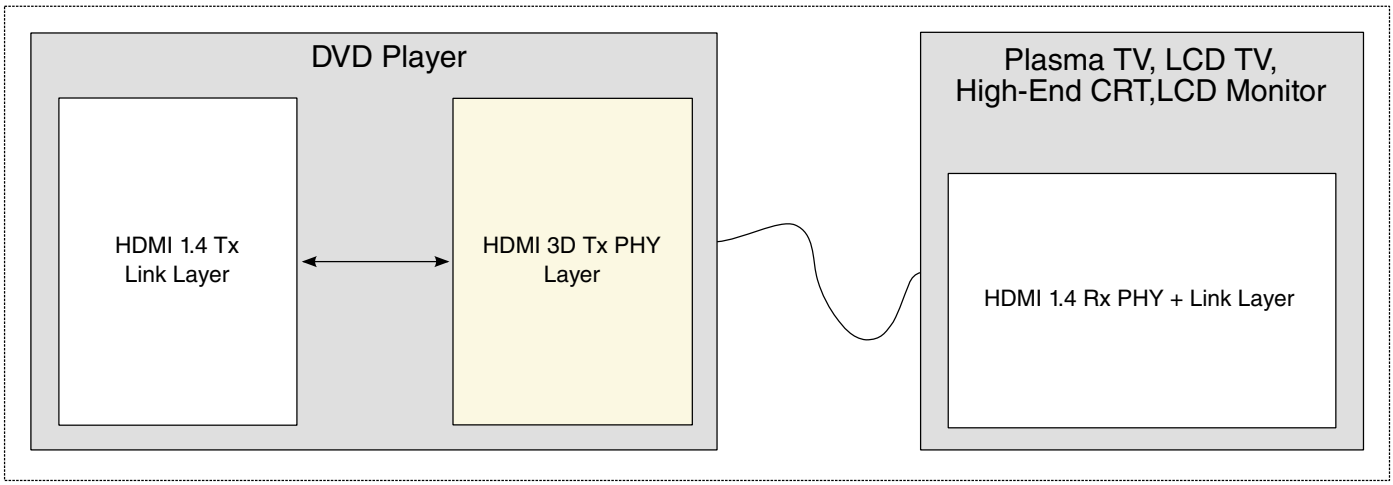


Figure 34-1. Typical Application for HDMI 3D Tx PHY

34.1.3 Standards Compliance

The HDMI_PHY is fully compliant with all the required specifications of the following standards:

- *High-Definition Multimedia Interface Specification*, Version 1.4a
- *Digital Visual Interface*, Revision 1.0
- *HDMI Compliance Test Specification*, Version 1.4a

34.1.4 Features

The HDMI_PHY provides the following features:

- Support for up to 720p at 100 Hz and 720i at 200 Hz or 1080p at 60 Hz and 1080i/720i at 120 Hz HDTV display resolutions and up to QXGA graphic display resolutions
- Support for 3D video formats
- Link controller flexible interface with 30- or 60-bit SDR data access
- Up to 9.2 Gbps aggregate bandwidth
- Driver with pre-emphasis and edge rate control for extra-long cable support
- Programmable source terminations
- HPD input analog comparator
- Rx sensing
- 13.5-266 MHz input reference clock

- 50% duty-cycle output clock
- Embedded Tx scope
- Embedded A/D converter and analog testbus for ATE testing
- Built-in pattern generator
- Small core area
- Low power consumption

34.1.5 HDMI 3D Tx PHY System-Level Overview

This section provides a system-level overview of the HDMI 3D Tx PHY.

34.1.5.1 System-Level Block Diagram

The figure below shows a system-level block diagram of the HDMI Tx System.

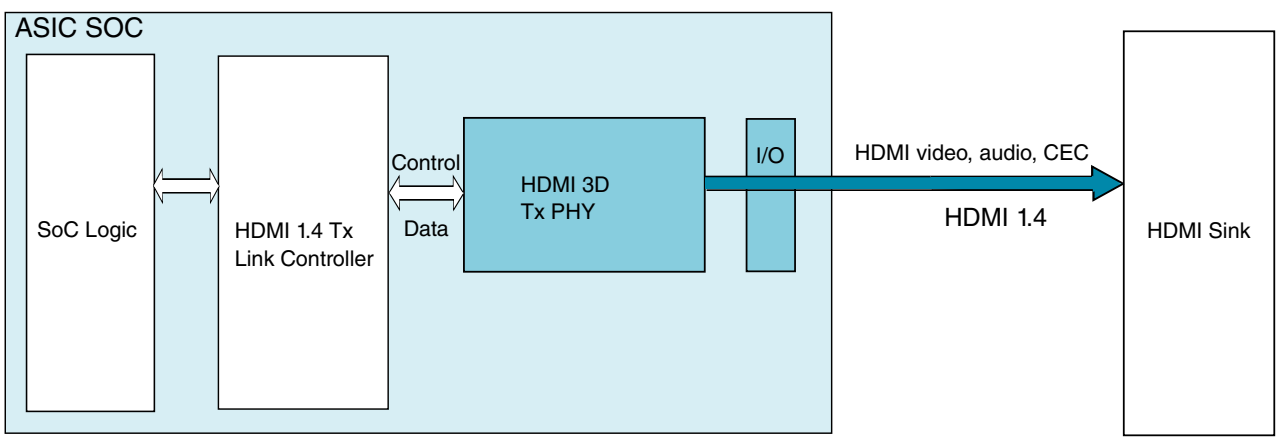


Figure 34-2. HDMI Tx System-Level Block Diagram

The HDMI_PHY macro interfaces with the HDMI Tx link controller, which provides control signals and video data to be transmitted through the TMDS lanes. Analog I/O's interface with the external world.

34.1.5.2 HDMI 3D Tx PHY

The HDMI 3D Tx PHY is the physical layer of an HDMI/DVI-compliant digital transmitter (source).

For a functional block diagram and a functional description of the HDMI 3D Tx PHY, see Functional Description.

34.1.5.2.1 Interfaces

The HDMI interface on the HDMI_PHY block is defined by the TMDS differential lanes and the HPD/DDCCEC connections. HDMI video data is provided through three differential TMDS pairs for data and one TMDS differential pair for clock. HPD/DDC_CEC enables detection of the HDMI sink. These signals interface the external world and connect to the HDMI connector.

On the system interface, the HDMI_PHY connects to the HDMI Tx link controller. This block provides an input reference clock to the PHY (pixel clock). Based on the defined video mode, a pixel repetition clock and a TMDS word clock are output to the controller. The controller then generates video data on this TMDS word clock and outputs both data and clock to the HDMI_PHY. Data is then serialized and sent to the HDMI interface.

In addition, the system side includes one standard I^2C interface for configuration and testability of the PHY. Other signals correspond to control logic and are either input to the PHY or provided by the PHY to the controller block for observability.

34.1.5.3 HDMI 1.4 I/O Pads

HDMI 3D Tx PHY provides a set of analog ESD pads for use.

34.1.5.3.1 I/O Pads Description

HDMI 3D Tx PHY I/O pads are provided as follows:

- Pad order: VPH, VP, GD, VPH, VP, GD, VPH, VP, GD, HDMI_TX_CLK_P, HDMI_TX_CLK_N, GD, HDMI_TX_DATA0_N, HDMI_TX_DATA0_P, GD, HDMI_TX_DATA1_N, HDMI_TX_DATA1_P, GD, HDMI_TX_DATA2_N, HDMI_TX_DATA2_P, GD, VP, VPH, REXT, HDMI_TX_HPD, HDMI_TX_DDC_CEC

As shown in the figure below, for maximum performance, HDMI 3D Tx PHY I/O pads contain decoupling capacitance between VP and GD and a filter between VP and VP_FILT (one for each VP_FILT signal).

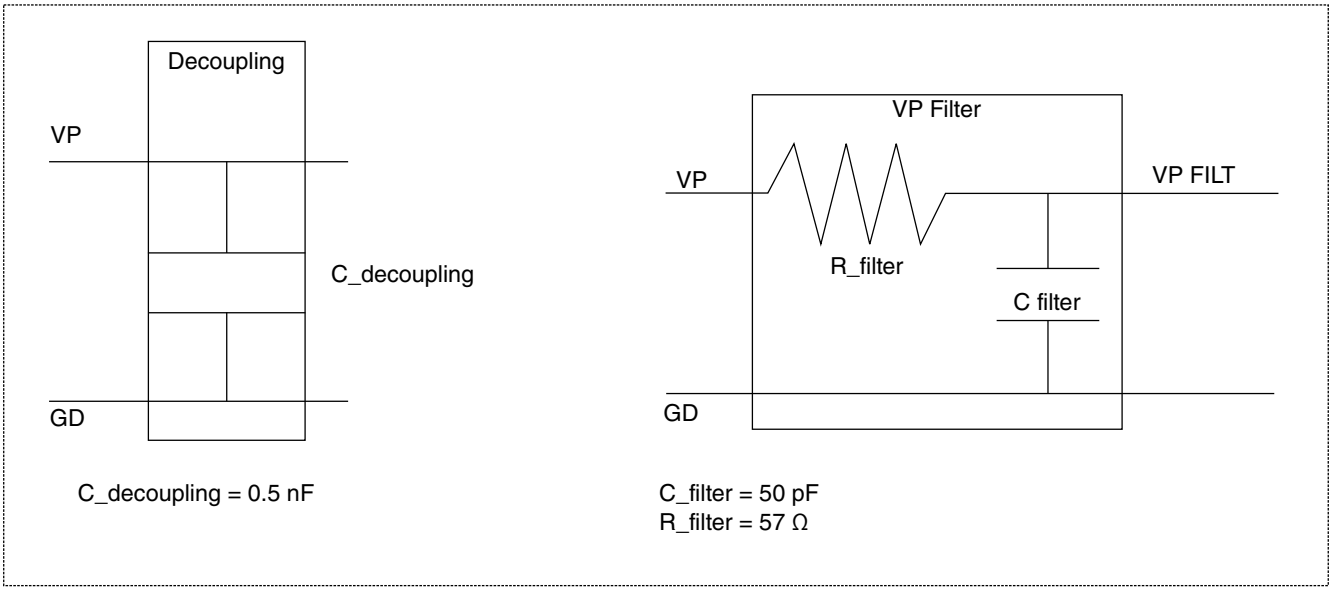


Figure 34-3. I/O Pads: Decoupling Capacitance Between VP and GD, Filter Between VP and VP_FILT

34.2 External Signals

The table found here describes the external signals of HDMI.

Table 34-1. HDMI External Signals

Signal	Description	Pad	Mode	Direction
HDMI_TX_CEC_LINE (CEC_LINE)	CEC line between source and sink	EIM_A25	ALT6	I/O
		KEY_ROW2	ALT6	
HDMI_TX_CLK_N (CLK_N)	Negative Clock Signal	HDMI_CLKM	No Muxing	I
HDMI_TX_CLK_P (CLK_P)	Positive Clock Signal	HDMI_CLKP	No Muxing	I
HDMI_TX_DATA0_N (DATA0_N)	Negative Data Signal 0	HDMI_D0M	No Muxing	I/O
HDMI_TX_DATA0_P (DATA0_P)	Positive Data Signal 0	HDMI_D0P	No Muxing	I/O
HDMI_TX_DATA1_N (DATA1_N)	Negative Data Signal 1	HDMI_D1M	No Muxing	I/O
HDMI_TX_DATA1_P (DATA1_P)	Positive Data Signal 1	HDMI_D1P	No Muxing	I/O
HDMI_TX_DATA2_N (DATA2_N)	Negative Data Signal 2	HDMI_D2M	No Muxing	I/O

Table continues on the next page...

Table 34-1. HDMI External Signals (continued)

Signal	Description	Pad	Mode	Direction
HDMI_TX_DATA2_P (DATA0_P)	Positive Data Signal 2	HDMI_D2P	No Muxing	I/O
HDMI_TX_DDC_CEC (DDC_CEC)	CEC Signal	HDMI_DDCCEC	No Muxing	I/O
HDMI_TX_DDC_SCL (DDC_SCL)	SCL Signal	EIM_EB2	ALT4	I/O
		KEY_COL3	ALT2	
HDMI_TX_DDC_SDA (DDC_SDA)	SDA Signal	EIM_D16	ALT4	I/O
		KEY_ROW3	ALT2	
HDMI_TX_HPD (HPD)	HPD Signal	HDMI_HPD	No Muxing	I/O

34.2.1 Top-Level I/O Diagram

The figure below shows the HDMI 3D Tx PHY top-level signals.

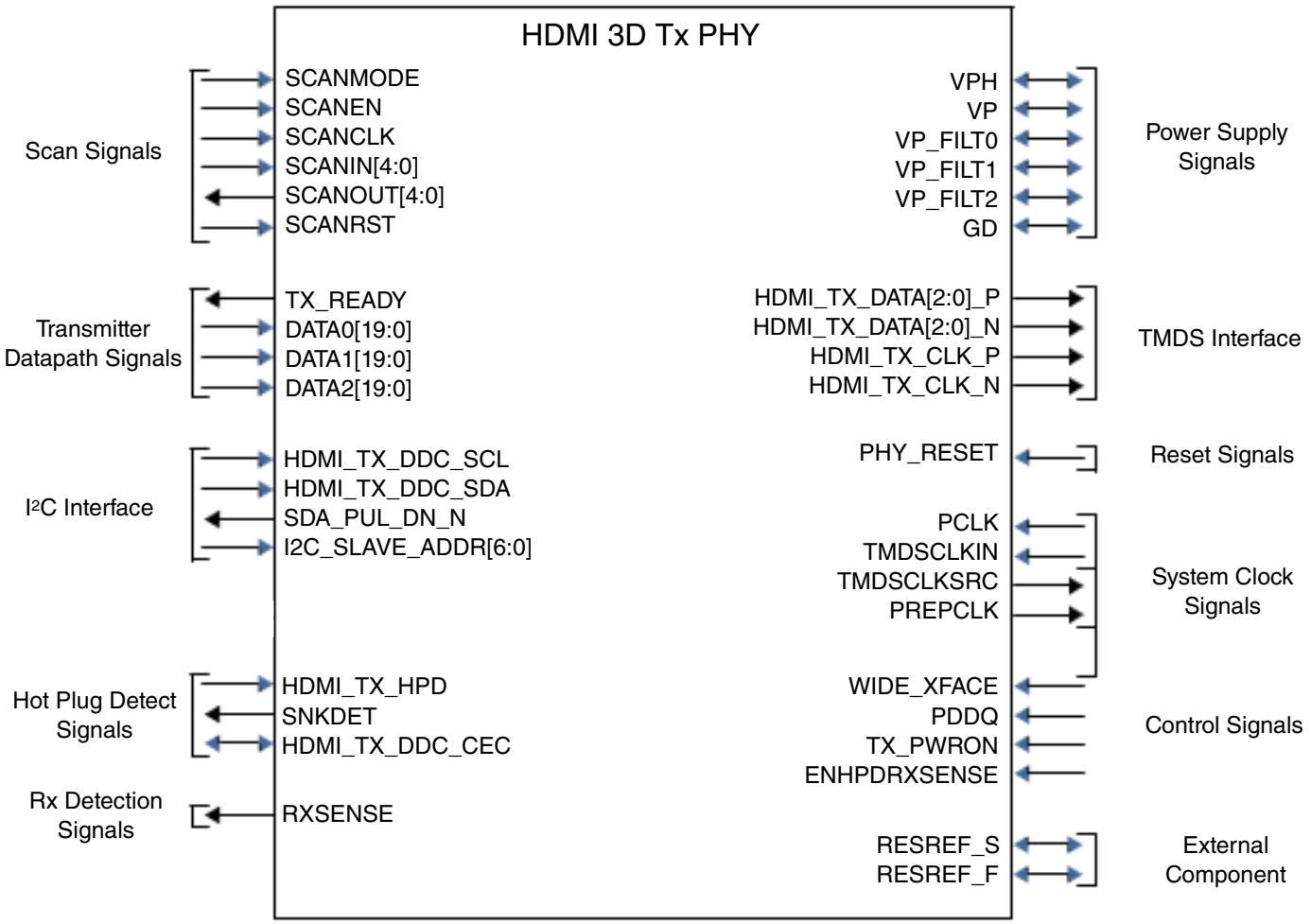


Figure 34-4. Top-Level I/O Diagram

34.2.2 Top -Level Signal Descriptions

This section describes the top-level signals.

In addition to describing the function of each signal, the signal descriptions include the following information:

- **Voltage range**: Describes the voltage range expected on this pin.
- **Synchronous**: Indicates that the signal is asserted or deasserted with respect to a clock edge.
- **Asynchronous**: Indicates that the signal is not asserted or deasserted with respect to a clock edge.

34.2.2.1 TMDS Interface

The table below describes the Transmission Minimized Differential Signaling (TMDS) interface signals.

Table 34-2. TMDS Interface

Signal	I/O	Description
HDMI_TX_DATA[2:0]_P	O	**Function**: Positive TMDS differential line driver data output for data channels 0,1, and 2. **Voltage range**: For the permitted voltage range, refer to *High-Definition Interface Specification*, Version 1.4, "Electrical Specification" section. **Active state**: N/A **Synchronous to**: N/A
HDMI_TX_DATA[2:0]_N	O	**Function**: Negative TMDS differential line driver data output for data channels 0,1, and 2. **Voltage range**: For the permitted voltage range, refer to *High-Definition Interface Specification*, Version 1.4, "Electrical Specification" section. **Active state**: N/A **Synchronous to**: N/A
HDMI_TX_CLK_P	O	**Function**: Positive TMDS differential line driver clock output **Voltage range**: For the permitted voltage range, refer to *High-Definition Multimedia Interface Specification*, Version 1.4, "Electrical Specification" section. **Active state**: N/A **Synchronous to**: N/A
HDMI_TX_CLK_N	O	**Function**: Negative TMDS differential line driver clock output **Voltagerange**: For the permitted voltage range, refer to *High-DefinitionMultimediaInterfaceSpecification*, Version 1.4, "Electrical Specification" section. **Activestate**: N/A **Synchronousto**: N/A

34.2.2.2 Reset Signals

The table blow describes the reset signals.

Table 34-3. Reset Signals

Signal	I/O	Description
PHY_RESET	I	**Function**: PHY reset. This signal places the digital section of the macro into a reset state. **Voltagerange**: 0-VP

Table 34-3. Reset Signals

Signal	I/O	Description
		Activestate: High **Synchronousto**: Asynchronous

34.2.2.3 External Component

The table below describes the top-level signals that connect to an external component.

Table 34-4. External Component

Signal	I/O	Description
RESREF_S	I/O	**Function**: 1600- precision resistor to ground. During resistor tuning, current is forced through the external resistor using RESREF_F, while the induced voltage is sensed on RESREF_S. **Voltage range**: N/A **Active state**: N/A **Synchronous to**: Asynchronous
RESREF_F	I/O	**Function**: Reference resistor connection **Voltage range**: N/A **Active state**: N/A **Synchronous to**: Asynchronous

34.3 Functional Description

This section describes the functional architecture of the HDMI 3D Tx PHY.

34.3.1 Functional Overview

The figure found in this section shows a functional block diagram of the HDMI_PHY.

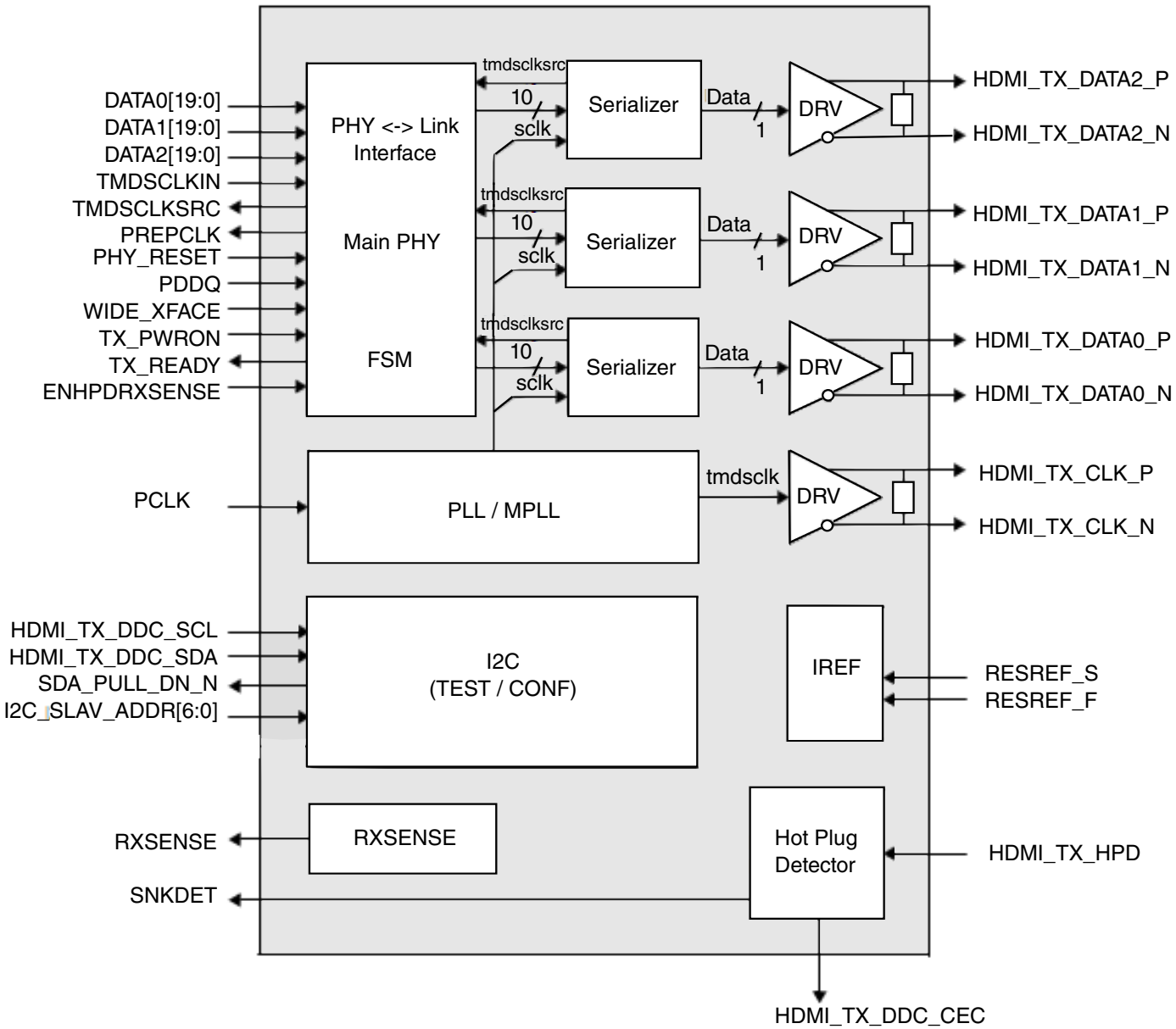


Figure 34-5. HDMI 3D Tx PHY Functional Block Diagram

The HDMI_PHY is the physical layer of an HDMI/DVI-compliant digital transmitter (source), capable of encoding and transmitting high-speed data streams carrying RGB video, audio, and control information. The HDMI Link Controller interface conforms to a 30- or 60-bit synchronous data interface (DATA0[19:0],DATA1[19:0],DATA2[19:0]).

The PHY includes two PLLs that synthesize the high-speed serial bit clock (required by the transmitter) from a reference pixel clock with a frequency of 13.5-266 MHz, for a transmitter capable of transmitting up to 9.2 Gbps using three lanes.

The HDMI_PHY drives audio and video across three TMDS data channels. Each serial TMDS link has a data rate range of 0.25-3.4 Gbps. The HDMI_PHY also drives the TMDS clock at 1/10th of the serial data rate with a frequency range of 25-266 MHz.

Within the HDMI_PHY, additional support blocks exist: the Bandgap block (for blocks biasing), the Resistor Calibration block, the ADC, and the analog test bus.

The HDMI_PHY accepts an input pixel clock (PCLK) with a frequency range of 13.5-266 MHz and accepts three channels of parallel data (TXDATA0[19:0], TXDATA1[19:0], TXDATA2[19:0]) that is synchronous with the TMDSCLKIN input clock. The TMDSCLKIN clock has a frequency range of 12.5-266MHz.

The TMDSCLKSRC clock output should be used by the HDMI transmitter controller as the source for TMDSCLKIN. The relation between TMDSCLKSRC and TMDSCLKIN should be constant between resets.

The HDMI_PHY's status and configuration is accessed through its I^2C interface. In Scan mode of operation, SCANCLK is bypassed to all PHY output clocks.

PLL/MPLL Operation

The PLL/MPLL can be configured in Coherent mode or Non-Coherent mode (default). In Coherent mode, the TMDS clock is the MPLL feedback clock, which is coherent with the MPLL's high-speed output clock, because both clocks are shaped by the MPLL response. In Non-Coherent mode, the TMDS clock is the MPLL reference clock, which is not coherent with the MPLL's high-speed output clock.

Driver Operation

The driver differential source termination, edge rate, pre-emphasis, and voltage level can be configured for maximum performance.

34.3.2 Operating Modes

This section describes various operating modes of the HDMI_PHY.

The HDMI_PHY can be placed in two different operating modes: Power-Down and Active.

The PHY_RESET signal is used to place the digital section of the IP in a well-defined state. Reset is active when the PHY_RESET signal is asserted high. PHY_RESET assertion also clears the configuration registers. Through the control registers, assertion of a soft reset to the macro is possible. This soft reset clears all system FSMs except I^2C and control registers.

For each separate video mode in which the HDMI_PHY is set to transmit, due to different operating frequency, color depth and pixel repetition that characterizes each one, you might need to configure the PHY block for correct operation and optimized performance. It is recommended that PHY configuration through the I^2C interface be done while the PHY is in Power-down mode. Configuration involves programming the PLL/MPLL internal blocks as well as the analog drivers' source termination value and edge rate control, for example. This programming is done through the I^2C interface. For more information about the I^2C interface, see I2C Interface Signals, for information about the PLL/MPLL configurations that must be made for each video mode of operation, see PLL/MPLL Generic Configuration Settings , and for information about the driver configurations, see Control Registers and Appendix A: Driver Voltage Level Configuration.

34.3.2.1 Power-Down Mode

This mode is the lowest power consumption mode. The PHY enters this mode when the TX_PWRON and ENHPDRXSENSE signals are set to 1'b0 and the PDDQ and PHY_RESET signals are set to 1'b1.

This mode is characterized by having all analog blocks disabled and digital logic quiet. Current consumption in this mode corresponds to the analog blocks standby current and digital logic leakage current.

To enable the HDMI 3D Tx PHY to enter Power-down mode, after TX_PWRON is set to 1'b0, the PCLK input must continue toggling until the TX_READY output signal is set to 1'b0 (indicating that the PHY has been correctly set in Power-down mode). If this sequence is not followed, TX_READY is not set to 1'b0, and the PHY remains in Active mode.

During Power-down mode, the ENHPDRXSENSE control signal can be set to 1'b1, enabling the HPD and RXSENSE-related circuitry. This setting enables the link controller to observe when an HDMI Rx is connected to the Tx and to power up the HDMI 3D Tx PHY after detecting the Rx.

During Power-down mode, if ENHPDRXSENSE is enabled, it is recommended that source terminations on both data and clock lines are enabled by setting tx_rescal[6:0] and ck_rescal[6:0] to 7'b1111111 on registers

0x04 and 0x05, respectively. Because the analog driver within the macro is disabled, enabling the source terminations will help pull-down the floating TMDS data/clock lines to a voltage level that will guarantee proper RXSENSE operation and output signaling when the Rx connects/disconnects from the PHY.

34.3.2.2 Active Mode

Active mode is the normal operational mode. To start processing received data from the link controller interface, the macro must first reach this state. The PHY is ready to transmit when TX_READY is asserted high.

In this state, the interface signals (DATA0[19:0], DATA1[19:0], DATA2[19:0]) are sent to the HDMI interface. Data is transferred according to the HDMI specification.

After entering Active mode, the HDMI Tx macro starts serially transmitting data received from the link controller to the TMDS interface. The way in which input data is encoded (depends on whether data corresponds to Video Data Period, Data Island Period, or Control Periods) depends on the HDMI protocol sequence. Data encoding is performed by the Link controller layer.

While operating in Active mode, the HDMI_PHY operates in one of the following modes (as defined in the HDMI specification, and as shown in the figure below):

- Video Data Period
- Data Island Period
- Control Period

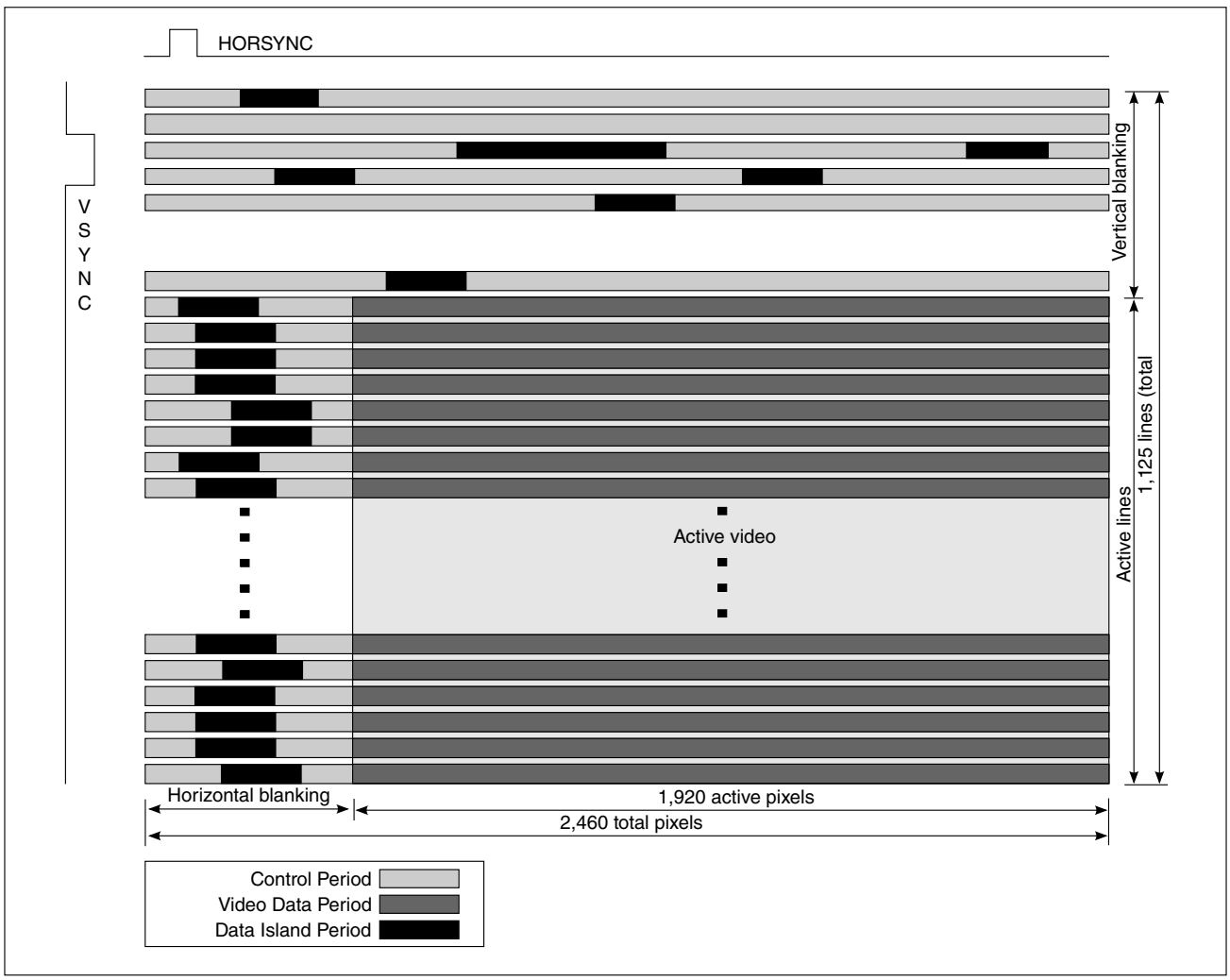


Figure 34-6. Active Mode

Video Data Period is used for the transmission of active pixel data. Transition Minimized (TMDS) data coding (8B10B) is used for 8-bit data transfers. In Deep-color modes, pixel data words are first packed in 8-bit multiple data groups. These groups are then fragmented into 8-bit length words for TMDS encoding.

Data Island Period is used for audio or other auxiliary data information transmission. Data is transferred in packet format using a similar Transition Minimized coding, in this case a TMDS Error Reduction Coding (TERC4). This is a 4B10B code.

Finally, Control Period is used when non-active video/audio or other auxiliary information needs to be transmitted. Only control information is transmitted in this mode. In Control Periods, data is transferred using a Transition Maximized (also TMDS) data

coding, in a form of 2B10B code. This property is very useful to enable proper word alignment on the sink (receiver) side. A Control Period is always entered between any other two periods that are not Control Periods.

34.3.2.2.1 Wide Interface Mode

To support the full HDMI bandwidth requirement, each of the three channels accepts 9-bit data and clock at a maximum rate of 340 MHz. In this case, the WIDE_XFACE input is set to 1'b0.

When WIDE_XFACE is set to 1'b1, each channel accepts 20-bit data and clock at a maximum rate of 170 MHz. This mode is meant to ease interface timing while maintaining the maximum HDMI bandwidth requirement.

When WIDE_XFACE is set to 1'b1, the TMDSCLKSRC clock frequency is not reduced to half simply through the WIDE_XFACE control. The frequency can be reduced to half through the I^2C interface, register 0x1E, cko_word_div_enb control bit. This control is independent from the WIDE_XFACE selection.

34.3.2.3 Power Sequence

The HDMI_PHY power sequence implementation controls the PLL/MPLL start of operation, resistor calibration, and clock alignment.

As shown in the figure below, during power-up, the Power Sequence finite state machine (FSM) advances through the PLL and MPLL power-up states by loading a counter at the beginning of each state (pwr_cnt) and advancing when the counter (pwr_cnt) decrements to 0. After the Resistor Calibration FSM is enabled in the RES_CAL state, the FSM waits for the assertion of rcal_adc_done before advancing. After the Clock Alignment FSM is enabled in the TX_CLK_ALIGN state, the Power Sequence FSM waits for the assertion of tx_ck_align_done before advancing to the TX_READY state. The TX_READY state asserts the TX_READY output signal, which indicates that the HDMI_PHY is ready to transmit the TMDS clock and data and that normal operation can begin.

If TX_PWRON is deasserted in the TX_READY state, the Power Sequence FSM moves to the DISABLE_TX state. The power-down sequence is initiated. This state disables the TMDS clock and data outputs. Next, the PWR_DN_PLL resets the PLL and MPLL blocks. Finally, the FSM returns to the PWR_DN state and waits for TX_PWRON assertion to begin the power-up sequence.

The HDMI_PHY enters Power-down mode only when the TX_READY output is deasserted. To enable the power sequence to reach this state, when TX_PWRON is set to 1'b0 while the HDMI_PHY is in the "TX_READY" state, the PCLK input must continue

toggling until TX_READY is deasserted. This sequence is required to enable the Power Sequence FSM to continue its execution and move from TX_READY to DISABLE_TX, later to PWR_DN_PLL, and finally to PWR_DN that corresponds to the power-down state.

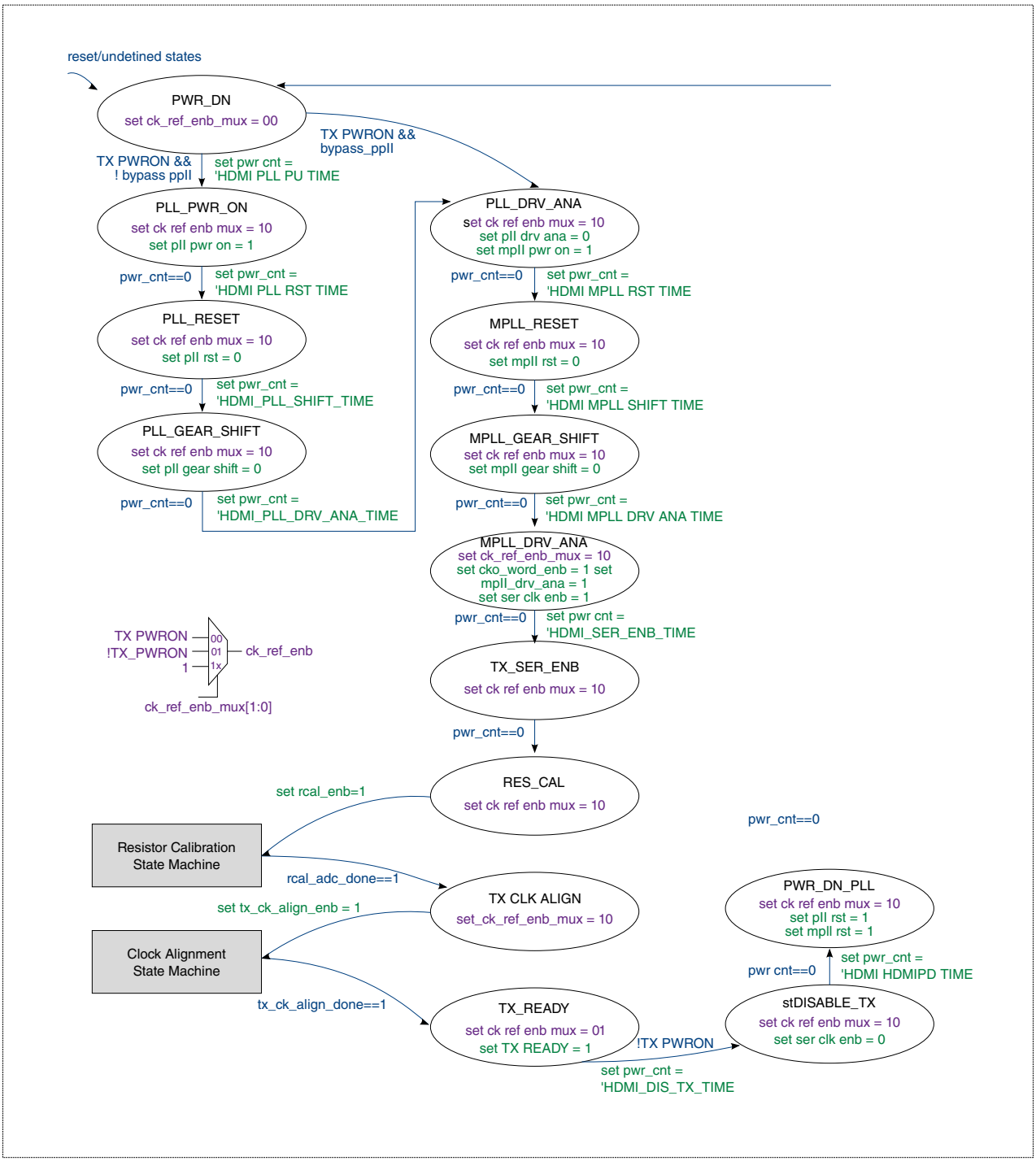


Figure 34-7. Power Sequence Finite State Machine

The following signals can be overridden through the control registers: bypass_ppll, pll_pwr_on, pll_rst, pll_gear_shift, pll_drv_ana, mpll_rst, mpll_gear_shift, cko_word_enb, mpll_drv_ana, ser_ckl_enb, rcal_enb, tx_clk_align_enb, ser_clk_enb, and ck_ref_enb.

The rcal_adc_done and tx_ck_align_done signals can be observed through the control registers or through the digital test bus (dtb[1:0], controlled through control registers).

The TX_PWRON and TX_READY signals belong to the HDMI 3D Tx PHY macro interface.

The power sequence can also be started through control registers, using the tx_pwron0, tx_pwron1, tx_pwron2, and ck_pwron signals.

TX_READY can also be overridden through control registers.

The clock source that clocks the Power Sequence FSM is controlled by the refclk_enb signal. This signal is asserted when TX_PWRON is asserted or when the state machine is in a state other than PWR_DN. This signal can be overridden through the control registers.

NOTE

Due to a potential different state on some analog nodes, it is recommended that the power-up sequence be run twice (after the first TX_READY assertion, the PHY can be powered down, then powered up again).

The clock alignment procedure can be done differently for each of the power-up sequence runs. For the first run, the clock alignment can be set to be performed channel-by-channel (tx_ck_align_mode = 1'b1, register 0x1C); for the second power-up, the clock alignment can be set to be performed for all channels at the same time (tx_ck_align_mode = 1'b0, register 0x1C) to ensure inter-pair skew close to 0 UI.

34.3.2.3.1 PLL/MPLL

During power-up, the PLL block and (optionally) MPLL block receive the pwron, rst, gear_shift, and drv_ana signals, as shown in the figure below. The minimum and maximum timing for these signals is listed in the table below. The power sequence then initiates the resistor calibration and clock alignment steps.

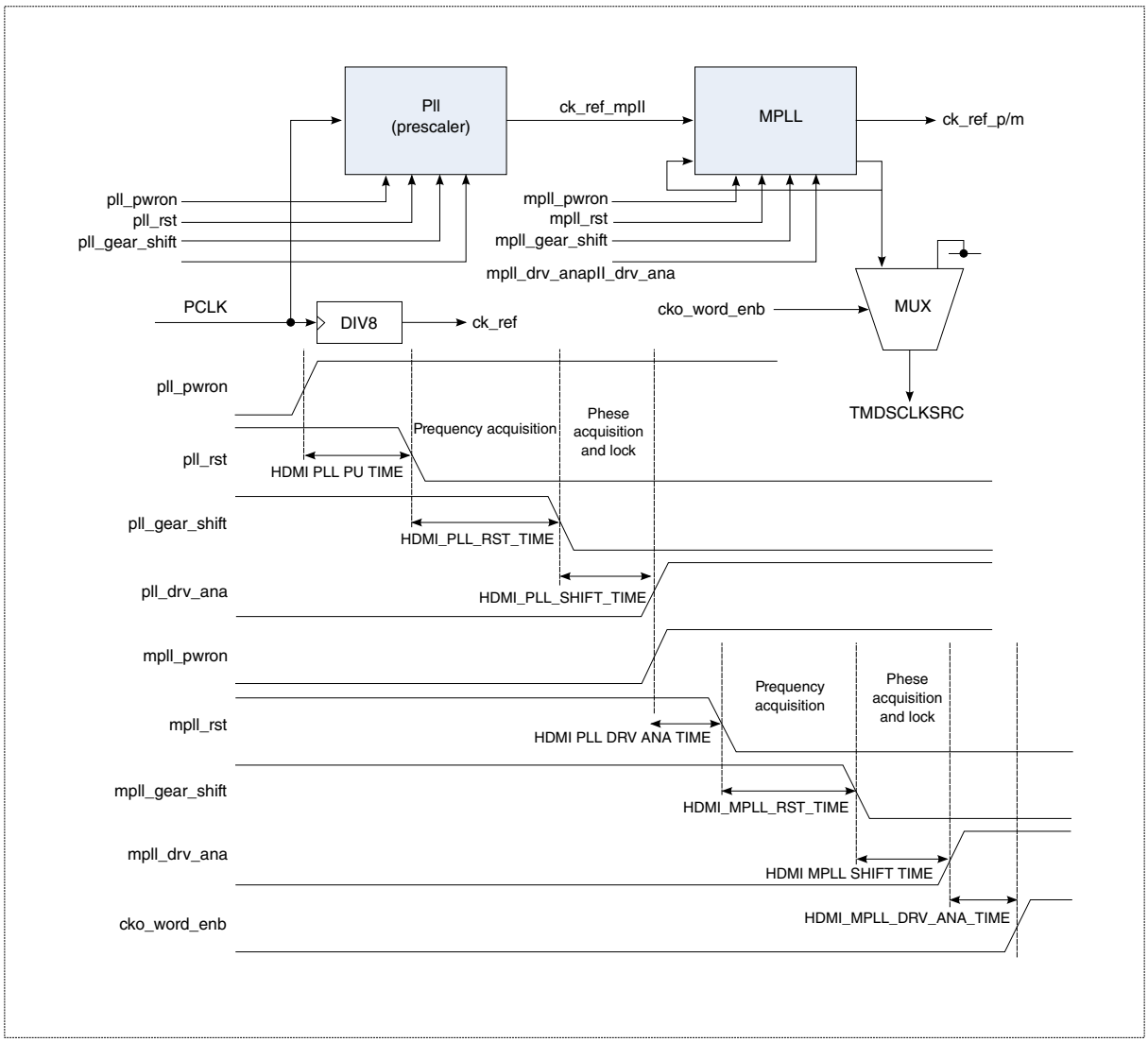


Figure 34-8. PLL/MPLL Power-Up Sequence

The following signals can be overridden through the control registers: pll_pwron, pll_rst, pll_gear_shift, pll_drv_ana, mpll_pwron, mpll_rst, mpll_gear_shift, mpll_drv_ana, and cko_word_enb.

ck_ref, ck_ref_p/m, and ck_ref_mpll are internal signals, while PCLK and TMDSCLKSRC belong to the HDMI 3D Tx PHY macro interface.

The table below provides the PLL/MPLL power-up times.

Table 34-5. PLL/MPLL Power-Up Time

Count Name	Count Value	Minimum Time (µs) (T(refclk) = 0.00294 µs)	Maximum Time (µs) (T(refclk = 0.07407 µs)
HDMI_PLL_PU_TIME	125	2.294	74.074
HDMI_PLL_RST_TIME	1250	22.940	740.740
HDMI_PLL_SHIFT_TIME	50	0.918	29.630
HDMI_PLL_DRV_ANA_TIME	125	2.294	74.074
HDMI_MPLL_RST_TIME	2500	45.880	1481.480
HDMI_MPLL_SHIFT_TIME	1000	18.352	592.592
HDMI_MPLL_DRV_ANA_TIME	50	0.918	29.630
HDMI_SER_ENB_TIME	50	0.918	29.630
Total PLL and MPLL Power-up Time		**94.513**	**3,051.849**

34.3.2.3.2 Resistor, ADC Calibration

Users can set the single-ended source termination resistance via the d_tx_term[2:0] control register bits, according to the table below.

Table 34-6. Single-Ended Source Termination Resistance Settings

N	d_tx_term[2:0]	R(Ω)
0	000	50
1	001	56.14
2	010	66.67
3	011	80
4	100	100
5	101	133.33
6	110	200
7	111	Open

To accurately set the specified termination, the resistor calibration measures the actual resistance of the external resistor. The resistance is measured using successive approximations of a 7-bit termination value by enabling each bit (one-hot) from MSB to LSB.

Users select the termination value by setting the d_tx_term[2:0] control register bits. The HDMI 3D Tx PHY includes differential source termination on the drivers.

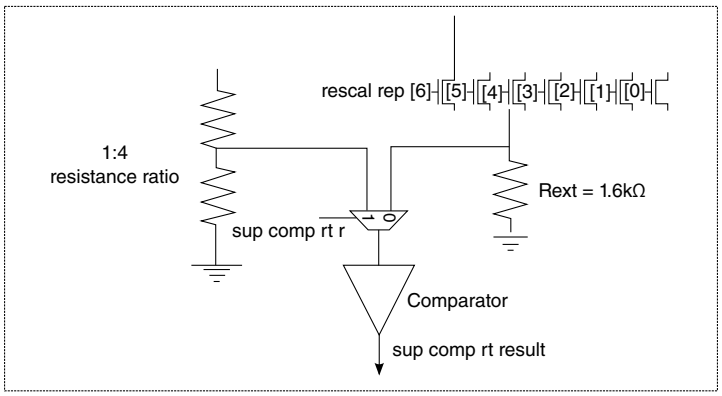


Figure 34-9. External Resistor Measurement

The exact value of Rext is determined to create an exact termination.

The analog support block applies a current to the external resistor and compares the resultant voltage. The result (sup_comp_rt_result) is registered by the Calibration block.

The source termination is then set according to the following equation:

tx_rescal = rescal_rep x (1 - 0.125 x *N*)

Instead of subtracting by (1/8th x rescal_rep x *N*), rescal_rep is initially multiplied by 8 (left shift by 3 bits), then tx_rescal is subtracted *N* number of times. *N* sets clk_cnt, which sets the number of subtraction loops, as shown in the table below. This table represents the clk_cnt required to trigger tx_rescal_done. The final result is divided by 8 (right shift by 3 bits) and rounded up if necessary.

Table 34-7. clk_cnt Necessary to Trigger tx_rescal_done

N/d_tx_term	clk_cnt to Trigger tx_rescal_done
0	<all>
1	0
2	1
3	2
4	3
5	4
6	5
7	<all>

The figure below shows the FSM for the resistor and ADC calibration.

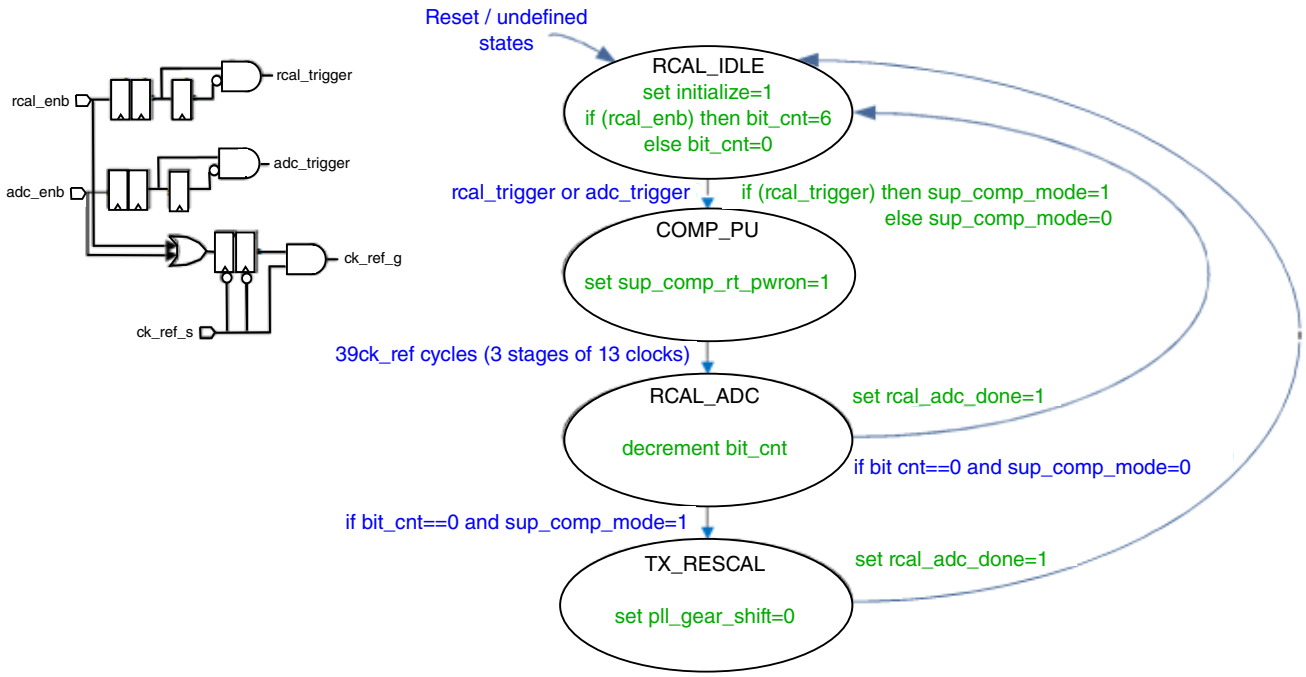


Figure 34-10. Finite State Machine for the Resistor and ADC Calibration

The rcal_enb signal is asserted by the Power Sequence FSM, but this signal can be overridden through the control registers.

The rescal_rep, sup_comp_mode, sup_comp_rt_r, sup_comp_rt_result, tx_rescal, sup_comp_rt_pwron, and adc_enb signals can be overridden through the control registers.

The rcal_adc_done signal can be observed through the control registers or through the digital test bus (dtb[1:0], controlled through control registers).

34.3.2.3.3 Clock Alignment

There are two clocks in the transmit clock path, TMDSCLKIN (from the parallel HDMI controller interface), and internal ck_tx_out (a divided-down clock, generated from the MPLL clock). These two clocks must be properly synchronized, aligning them before transmitting data through the HDMI 3D Tx PHY.

The alignment results in the internal tx_ser_clk (buffered version of TMDSCLKIN) being delayed in the range of 0.6-0.8 UI with respect to the TMDSCLKIN clock.

The Clock Alignment module adjusts the alignment between TMDSCLKIN and tx_ser_clk by generating internal clock kill signals called tx_ser_clk_kill[2:0]. On the rising edge of each tx_ser_clk_kill[2:0] signal, the corresponding tx_ser_clk[2:0] is delayed by 0.2 UI.

The clock alignment enable input, tx_ck_align_enb, is generated by the power sequence, but this input can also be overridden through the control registers.

Because it is assumed that channel 1 (the middle channel) has the median delays among the three channels, the state machine will compare TMDSCLKIN and tx_ser_clk[1] only when the register bit, tx_ck_align_mode, is low (default). In this case, tx_ser_clk_kill[2:0] is asserted simultaneously. When tx_ck_align_mode is high, each channel is aligned individually. Channel 2 is aligned first, followed by channel 0 and lastly channel 1. The tx_ck_align_mode signal is controlled through the control registers.

The internal clock kill signals, tx_ser_clk_kill2, tx_ser_clk_kill1, and tx_ser_clk_kill0, are generated by the Clock Alignment state machine, but these signals can be overridden through the control registers.

34.3.2.4 Color Depth and Color Mode Selection

This section describes HDMI 3D Tx PHY color depth and color mode selection.

34.3.2.4.1 Pixel Repetition Clock Generation Selection

For small displays operating a TMDS clock below 25 MHz or applications that require an increase in available bandwidth for audio and auxiliary information transmission, the HDMI_PHY can generate pixel repetition clocks.

Pixel repetition signals are dedicated to configure pixel repetition rates. These signals affect only the relation between PCLK and PREPCLK.

PCLK 1 / (pixel repetition) x PREPCLK

34.3.3 Configuration and Test Mode

This section describes HDMI 3D Tx PHY configuration and test mode operation.

The HDMI 3D Tx PHY can be configured and set in test mode through the supported I^2C interface.

To avoid transient periods in the macro operation during reconfiguration procedures, it is highly recommended that the I^2C interface be used while the HDMI 3D Tx PHY is in Power-down mode.

Control Registers describes all the registers and corresponding fields that are available through the I^2C interface.

34.3.3.1 Power-Up Configuration

Before powering up the HDMI 3D Tx PHY, configure the PHY for proper operation and performance.

Configuring the PHY through the I^2C interface is as follows.

NOTE

The recommended settings for the PLL/MPLL mode of operation and driver configurations provided in this section are presented as the default value that should be considered and initially configured. Unless stated otherwise, these recommended settings apply to all operating frequencies.

PLL/MPLL Mode of Operation

Configuration summary: Configure PLL/MPLL mode of operation as Single or Two-PLL in Coherent or Non-Coherent mode. For more information about these modes of operation, see Power-Up Requirements.

Table 34-8. PLL/MPLL Mode of Operation

Fields	Bits	Default
Register0x13		
bypass_ppll	11	1'b0
Register0x17		
cko_sel	2-1	2'b11

PLL/MPLL Clock Dividers and Analog Configuration

Configuration summary: Configured for each video mode. For information about the PLL and MPLL divider settings, see Table B-2 .

Table 34-9. PLL/MPLL Clock Dividers and Analog Configuration

Fields	Bits
Register 0x06	
prep_div[1:0]	14-13
mpll_n_ctrl[1:0]	8-7
pll_n_ctrl[1:0]	6-5
pixel_rep[2:0]	4-2
clr_dpth[1:0]	1-0

Table continues on the next page...

Table 34-9. PLL/MPLL Clock Dividers and Analog Configuration (continued)

Fields	Bits
Register 0x10	
mpll_prop_cntrl[2:0]	11-9
mpll_int_cntrl[2:0]	8-6
pll_prop_cntrl[2:0]	5-3
pll_int_cntrl[2:0]	2-0
Register 0x15	
pll_gmp_cntrl[1:0]	3-2
mpll_gmp_cntrl[1:0]	1-0

Driver Edge Rate Control

Configuration summary: This bus controls the slew rate of the clock and data output drivers.

Table 34-10. Driver Edge Rate Control

Fields	Bits	Default
Register 0x06		
tx_edgerate[1:0]	12-11	2'b00
ck_edgerate[1:0]	10-9	2'b00

Driver Single-Ended Source Termination

Configuration summary: This bus controls the driver single-ended source termination.

Table 34-11. Driver Single-Ended Source Termination

d-tx_term	R (ohm)
000	50
001	56.14
010	66.67
011	80
100	100
101	133.33
110	200
111	Open circuit

Fields	Bits	Default
Register 0x19		
d_tx_term[2:0]	2-0	3'b101

Driver Voltage Level

Configuration summary: For information about the driver voltage level settings, see Appendix A: Driver Voltage Level Configuration. Driver voltage level configuration is dependent on driver differential source termination and driver pre-emphasis settings.

Table 34-13. Driver Voltage Level

Fields	Bits	Default
Register 0x0E		
sup_tx_lvl[4:0]	9-5	25-148.5 MHz: 5'b10000 222.75-297 MHz: 5'b01111 340 MHz: 5'b01110
sup_ck_lvl[4:0]	4-0	25-148.5 MHz: 5'b10000 222.75-297 MHz: 5'b01111 340 MHz: 5'b01110

Driver Pre-Emphasis

Configuration summary: Enables trailer drivers to enable pre-emphasis operation. Pre-emphasis is achieved by reducing the drive level of a non-transition bit with respect to a transition bit. The following table summarizes the total output current configured with pre-emphasis (I is current of unit current source).

Table 34-14. Driver Pre-Emphasis

Total Output Current	tx_symon	tx_traon	tx_trbon
0 * I	0	0	0
36 * I	1	0	0
36 * I + 3 * I	1	0	1
36 * I + 6 * I	1	1	0
36 * I + 6 * I + 3 * I	1	1	1

The amount of pre-emphasis is programmed through the tx_traon and tx_trbon signals in the control register, address 0x09 as follows.

Field	Bit	Default

Table continues on the next page...

tx_trbon	1	25-148.5 MHz: 1'b0 222.75-340 MHz: 1'b1
tx_traon	2	1'b0

34.4 System-Level Implementation

34.4.1 System Operation

This section describes HDMI 3D Tx PHY system operation including power-up/power-down and power- on reset.

34.4.1.1 Powering Up and Powering Down

To set the HDMI_PHY in Power-down mode, set the TX_PWRON signal to 1'b0 and the PDDQ signal to 1'b1. To power up the HDMI 3D Tx PHY and place it in Active mode, set TX_PWRON to 1'b1 and PDDQ to 1'b0.

Any configuration programmed on the HDMI_PHY must be done in Power-down mode. To configure the PHY through the I^2C interface, set PHY_RESET to 1'b0 (to move the digital core from a reset state and to enable programming).

The figure below shows the power-down and power-up sequences.

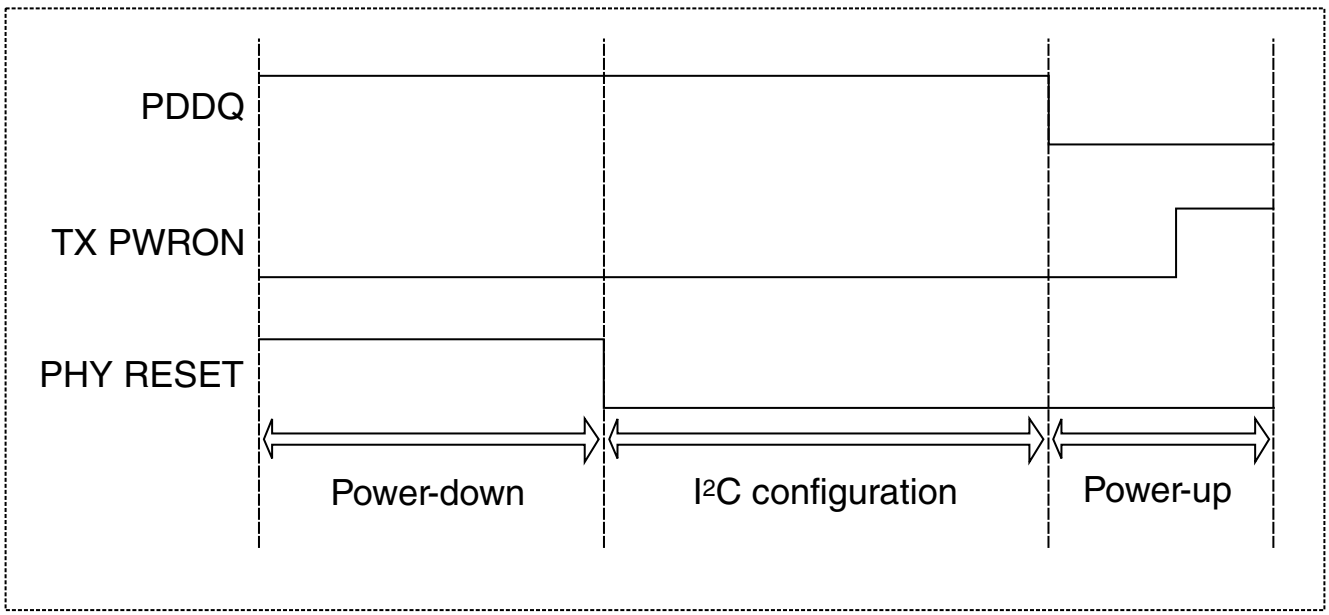


Figure 34-11. Power-Down and Power-Up Sequence

The power-up sequence starts by advancing one finite state machine (FSM) through the Phase-Locked Loop (PLL) and Multiplexed Phase-Locked Loop (MPLL) power-up states. Afterwards, resistor calibration is performed and clock alignment is enabled. When clock alignment is completed, the TX_READY signal is set to 1'b1 to indicate that the PHY is ready to transmit the TMDS clock and that data transmission and normal operation can start.

In Power-down mode, TMDS clock and data lines are disabled.

Power-Up Configuration

Before powering up the HDMI_ PHY, configure the PHY for proper operation and performance. The following PHY characteristics can be configured:

- PLL/MPLL mode of operation: Configure PLL/MPLL mode of operation as Single or Two-PLL in Coherent or Non-Coherent mode.
- PLL/MPLL clock dividers and analog configuration: Configure for each video mode.
- Driver edge rate control: Slew rate of clock and data output drivers
- Driver differential source termination: Differential source termination of clock and data output drivers
- Driver voltage level: Voltage reference level for clock and data channels
- Driver pre-emphasis: Enable and control pre-emphasis operation

For more information about this configuration and the associated control registers and signals, see Power-Up Configuration.

34.4.1.2 Active Mode Requirements

During normal operation, the RXSENSE circuitry must always be enabled (ENHPDRXSENSE asserted high).

During normal operation, if ENHPDRXSENSE is deactivated (asserted low), the PHY's analog portion is turned off.

34.4.1.3 Power-Up Requirements

Before setting the TX_PWRON signal high and the PDDQ signal low to power up the HDMI 3D Tx PHY, the PCLK and TMDSCLKIN input reference clocks must be stable and within their specified parameters, and power supply rails must be stable and within the specifications.

34.4.1.4 Power Supply Sequence When the HDMI 3D Tx PHY is Not Used

The HDMI 3D Tx PHY supports power collapsing.

There is no constraint on the power supply sequence when the HDMI 3D Tx PHY is not used. However, it is recommended that you short the data/clock lines to ground (or leave them floating). Activity at the data/clock lines or shorting the data/clock lines to the 3v3 rail (from Rx) should be avoided.

34.4.1.5 Power-Down Requirements

To enable the HDMI 3D Tx PHY to enter Power-down mode, after TX_PWRON is set to 1'b0, the PCLK input must continue toggling until the TX_READY output signal is set to 1'b0 (indicating that the PHY has been correctly set in Power-down mode). This sequence is required to enable the power-sequence machine to continue its execution and move from its states until the machine reaches the power-down state.

If this sequence is not followed, TX_READY is not set to 1'b0, and the HDMI 3D Tx PHY remains in Active mode.

34.5 Reference Clock

This chapter describes the reference clock that the HDMI_PHY supports.

The HDMI_PHY system requires an input clock signal that must be applied to the PCLK input with the following constraints:

- Minimum PCLK period/frequency (pixel repetition): 74 ns/13.5 MHz
- Minimum PCLK period/frequency (no pixel repetition): 39.7 ns/24.175 MHz
- PCLK duty cycle: 40-60%
- Maximum PCLK long-term RMS jitter: 100 ps

34.6 Control Registers

34.6.1 Control Registers Module Design Architecture

The Control Registers module is designed to provide the HDMI 3D Tx PHY with a file register component (36 words where each word comprises 16 bits).

Each register is designed to correspond to one of the following access types.

- **Read-only**: The HDMI controller can only read from this register. No writing by the HDMI controller is permitted.
- **Read/write**: The HDMI controller can read from and write to this register.
- **Read/write/override**: The HDMI controller can read from and write to this register. However, if the MSB of the register is set to 0, the value read by the HDMI controller will not be the value stored in the register; instead, the value will be the internal value of the HDMI PHY. In particular, this event occurs for 0x14, 0x18, 0x0A, and 0x0F where the value is read from the I^2C interface when PHY_RESET and TX_PWRON are low.
- **Read/write/asynchronous set-on-done**: The HDMI controller can read from and write to this register synchronously. However, three internal modules (Resistance Calibration, Clock Alignment, and Tx Scope) can write to this register asynchronously.

34.7 HDMI_PHY Memory Map/Register Definition

NOTE

HDMI_PHY registers are accessed through the I2C Master Interface of the HDMI Controller. See HDMI Transmitter (HDMI) for more information.

HDMI_PHY memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
0	Power Control (HDMI_PHY_PWRCTRL)	16	R/W	0000h	34.7.1/1836
1	Serializer Divider Control (HDMI_PHY_SERDIVCTRL)	16	R/W	0000h	34.7.2/1838
2	Serializer Clock Control (HDMI_PHY_SERCKCTRL)	16	R/W	0000h	34.7.3/1838
3	Serializer Clock Kill Control (HDMI_PHY_SERCKKILLCTRL)	16	R/W	0000h	34.7.4/1839
4	Transmitter and Resistance Calibration Control (HDMI_PHY_TXRESCTRL)	16	R/W	0000h	34.7.5/1840
5	Clock Calibration Control (HDMI_PHY_CKCALCTRL)	16	R/W	0000h	34.7.6/1841
6	Color Depth, Pixel Repetition, Clock Divider for PLL and MPLL, and Edge Rate Control (HDMI_PHY_CPCE_CTRL)	16	R/W	0400h	34.7.7/1842
7	Tx and Clock Measure Control (HDMI_PHY_TXCLKMEASCTRL)	16	R/W	0000h	34.7.8/1844
8	Tx Measure Control (HDMI_PHY_TXMEASCTRL)	16	R/W	0000h	34.7.9/1845
9	Clock Symbol and Transmitter Control (HDMI_PHY_CKSYMTXCTRL)	16	R/W	0000h	34.7.10/ 1847
A	Comparator Sequence Control (HDMI_PHY_CMPSEQCTRL)	16	R/W	0000h	34.7.11/ 1848
B	Comparator Power Control (HDMI_PHY_CMPPWRCTRL)	16	R/W	0000h	34.7.12/ 1849
C	Comparator Mode Control (HDMI_PHY_CMPMODECTRL)	16	R/W	0000h	34.7.13/ 1849
D	Measure Control (HDMI_PHY_MEASCTRL)	16	R/W	0000h	34.7.14/ 1850
E	Voltage Level Control (HDMI_PHY_VLEVCTRL)	16	R/W	0000h	34.7.15/ 1851
F	Digital-to-Analog Control (HDMI_PHY_D2ACTRL)	16	R/W	0000h	34.7.16/ 1852
10	Current Control (HDMI_PHY_CURRCTRL)	16	R/W	08ABh	34.7.17/ 1853
11	Drive Analog Control (HDMI_PHY_DRVANACTRL)	16	R/W	0003h	34.7.18/ 1853
12	PLL Measure Control (HDMI_PHY_PLLMEASCTRL)	16	R/W	0000h	34.7.19/ 1854
13	PLL Phase and Bypass Control (HDMI_PHY_PLLPHBYCTRL)	16	R/W	0000h	34.7.20/ 1856

Table continues on the next page...

HDMI_PHY memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
14	Gear Shift, Reset Mode, and Power State Control (HDMI_PHY_GRP_CTRL)	16	R/W	0000h	34.7.21/ 1857
15	Gmp Control (HDMI_PHY_GMPCTRL)	16	R/W	0000h	34.7.22/ 1858
16	MPLL Measure Control (HDMI_PHY_MPLLMEASCTRL)	16	R/W	0000h	34.7.23/ 1859
17	MPLL and PLL Phase, Scope Clock Select, and MUX Clock Control (HDMI_PHY_MSM_CTRL)	16	R/W	0000h	34.7.24/ 1861
18	Scope, Comparator Result and Power Bad Status (HDMI_PHY_SCRPB_STATUS)	16	R	0000h	34.7.25/ 1862
19	Transmission Termination (HDMI_PHY_TXTERM)	16	R/W	0007h	34.7.26/ 1864
1A	Power Sequence, TX Clock Alignment, Resistance Calibration, Pattern Generator Skip Bit, and TMDS Encoder Enable (HDMI_PHY_PTRPT_ENBL)	16	R/W	0000h	34.7.27/ 1865
1B	Pattern Generator Mode (HDMI_PHY_PATTERNGEN)	16	R/W	0000h	34.7.28/ 1867
1C	The Soft-Reset and DAC Enable, Clock Alignment and PG Mode (HDMI_PHY_SDCAP_MODE)	16	R/W	0000h	34.7.29/ 1868
1D	Scope Mode register (HDMI_PHY_SCOPEMODE)	16	R/W	0000h	34.7.30/ 1870
1E	Digital Transmission Mode (HDMI_PHY_DIGTXMODE)	16	R/W	0000h	34.7.31/ 1871
1F	Scope, Transmission Clock Alignment, and Resistance Calibration Set-on-Done Status (HDMI_PHY_STR_STATUS)	16	R/W	0000h	34.7.32/ 1875
20	Scope Counter on Channel 0 (HDMI_PHY_SCOPECNT0)	16	R	0000h	34.7.33/ 1877
21	Scope Counter on Channel 1 (HDMI_PHY_SCOPECNT1)	16	R	0000h	34.7.34/ 1877
22	Scope Counter on Channel 2 (HDMI_PHY_SCOPECNT2)	16	R	0000h	34.7.35/ 1878
23	Scope Counter on Clock Channel (HDMI_PHY_SCOPECNTCLK)	16	R	0000h	34.7.36/ 1878
24	Scope Sample Count MSB, Scope Sample Repetition (HDMI_PHY_SCOPESAMPLE)	16	R/W	13C0h	34.7.37/ 1879
25	Scope Counter MSB Channel 0 and Channel 1 (HDMI_PHY_SCOPECNTMSB01)	16	R	0000h	34.7.38/ 1880
26	Scope Counter MSB Channel 2 and Clock Channel (HDMI_PHY_SCOPECNTMSB2CK)	16	R	0000h	34.7.39/ 1880

34.7.1 Power Control (HDMI_PHY_PWRCTRL)

Register name: PWRCTRL

Access type: Read/write/override

Address: 0x00

Value at reset: 0x0000

Address: 0h base + 0h offset = 0h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved			tx_pwron	tx_pwron0	tx_pwron1	tx_pwron2	ck_pwron
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_PWRCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–5 -	This field is reserved. Reserved
4 tx_pwron	Transmitter Power-On This bit enables users to power down the entire PHY through the I2C interface. 0 Power off the PHY, if the Override bit is 1. 1 Power on the PHY, if the Override bit is 1.
3 tx_pwron0	Transmitter Power-On 0 This bit powers on or powers off the transmitter driver for channel 0. 0 Power off the transmitter driver for the first channel, if the Override bit is 0. 1 Power on the transmitter driver for the first channel, if the Override bit is 0.
2 tx_pwron1	Transmitter Power-On 1 This bit powers on or powers off the transmitter driver for channel 1. 0 Power off the transmitter driver for the second channel, if the Override bit is 0. 1 Power on the transmitter driver for the second channel, if the Override bit is 0.
1 tx_pwron2	Transmitter Power-On 2 This bit powers on or powers off the transmitter driver for channel 2. 0 Power off the transmitter driver for the third channel, if the Override bit is 0. 1 Power on the transmitter driver for the third channel, if the Override bit is 0.
0 ck_pwron	Clock Power-On This bit powers on or powers off the clock driver. 0 Power off the clock driver, if the Override bit is 0. 1 Power on the clock driver, if the Override bit is 0.

34.7.2 Serializer Divider Control (HDMI_PHY_SERDIVCTRL)

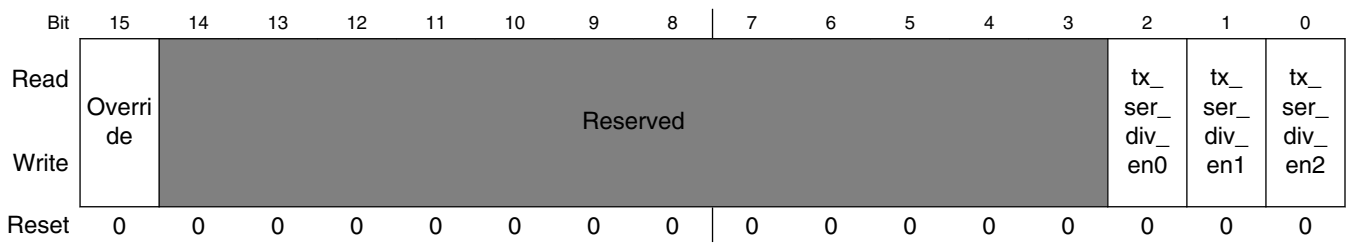
Register name: SERDIVCTRL

Access type: Read/write/override

Address: 0x01

Value at reset: 0x0000

Address: 0h base + 1h offset = 1h



HDMI_PHY_SERDIVCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–3 -	This field is reserved. Reserved
2 tx_ser_div_en0	Transmitter Serializer Divider Enable 0 This bit enables or disables the low-speed clock in serializer 0. 0 Disable the low-speed clock in the first serializer, if the Override bit is 0. 1 Enable the low-speed clock in the first serializer, if the Override bit is 0.
1 tx_ser_div_en1	Transmitter Serializer Divider Enable 1 This bit enables or disables the low-speed clock in serializer 1. 0 Disable the low-speed clock in the second serializer, if the Override bit is 0. 1 Enable the low-speed clock in the second serializer, if the Override bit is 0.
0 tx_ser_div_en2	Transmitter Serializer Divider Enable 2 This bit enables or disables the low-speed clock in serializer 2. 0 Disable the low-speed clock in the third serializer, if the Override bit is 0. 1 Enable the low-speed clock in the third serializer, if the Override bit is 0.

34.7.3 Serializer Clock Control (HDMI_PHY_SERCKCTRL)

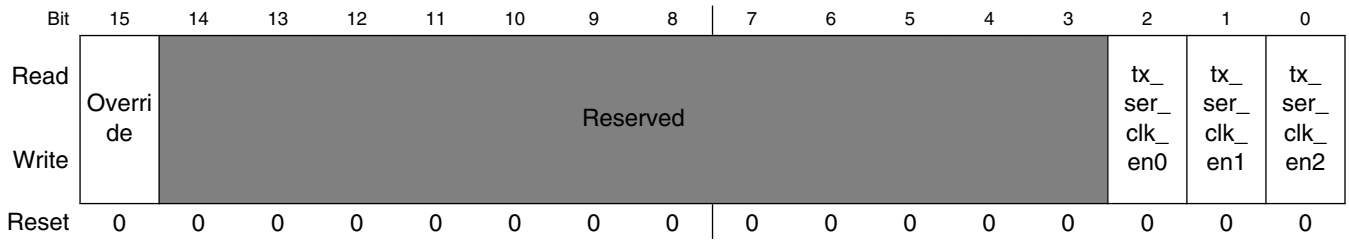
Register name: SERCKCTRL

Access type: Read/write/override

Address: 0x02

Value at reset: 0x0000

Address: 0h base + 2h offset = 2h



HDMI_PHY_SERCKCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–3 -	This field is reserved. Reserved
2 tx_ser_clk_en0	Transmitter Serializer Clock Enable 0 This bit enables or disables the high-speed clock in serializer 0. 0 Disable the high-speed clock in the first serializer, if the Override bit is 0. 1 Enable the high-speed clock in the first serializer, if the Override bit is 0.
1 tx_ser_clk_en1	Transmitter Serializer Clock Enable 1 This bit enables or disables the high-speed clock in serializer 1. 0 Disable the high-speed clock in the second serializer, if the Override bit is 0. 1 Enable the high-speed clock in the second serializer, if the Override bit is 0.
0 tx_ser_clk_en2	Transmitter Serializer Clock Enable 2 This bit enables or disables the high-speed clock in serializer two. 0 Disable the high-speed clock in the third serializer, if the Override bit is 0. 1 Enable the high-speed clock in the third serializer, if the Override bit is 0.

34.7.4 Serializer Clock Kill Control (HDMI_PHY_SERCKKILLCTRL)

Register name: SERCKKILLCTRL

Access type: Read/write/override

Address: 0x03

Value at reset: 0x0000

Address: 0h base + 3h offset = 3h

Bit	15	14	13	12	11	10	9	8
Read / Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read / Write	Reserved					tx_ser_clk_kill0	tx_ser_clk_kill1	tx_ser_clk_kill2
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_SERCKKILLCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–3 -	This field is reserved. Reserved
2 tx_ser_clk_kill0	Transmitter Serializer Clock Kill 0 This bit is used to delay the tx_ck_out0 sampling clock by a time $t_{delay,}$ where t_{delay} equals one period of refclk (340 MHz). This delay equals 0.2 UI of the tx_ck_out0 clock. 0 No effect, if the Override bit is 0. 1 Delay the tx_ck_out0 clock by 0.2 UI, if the Override bit is 0.
1 tx_ser_clk_kill1	Transmitter Serializer Clock Kill 1 This bit is used to delay the tx_ck_out1 sampling clock by a time $t_{delay,}$ where t_{delay} equals one period of refclk (340 MHz). This delay equals 0.2 UI of the tx_ck_out1 clock. 0 No effect, if the Override bit is 0. 1 Delay the tx_ck_out1 clock by 0.2 UI, if the Override bit is 0.
0 tx_ser_clk_kill2	Transmitter Serializer Clock Kill 2 This bit is used to delay the tx_ck_out2 sampling clock by a time $t_{delay,}$ where t_{delay} equals one period of refclk (340 MHz). This delay equals 0.2 UI of the tx_ck_out2 clock. 0 No effect, if the Override bit is 0. 1 Delay the tx_ck_out2 clock by 0.2 UI, if the Override bit is 0.

34.7.5 Transmitter and Resistance Calibration Control (HDMI_PHY_TXRESCTRL)

Register name: TXRESCALCTRL

Access type: Read/write/override

Address: 0x04

Value at reset: 0x0000

Address: 0h base + 4h offset = 4h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved	tx_rescal[6:0]					
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	tx_ rescal[6:0]	rescal_rep[6:0]						
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_TXRESCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14 - Reserved	This field is reserved.
13–7 tx_rescal[6:0]	Transmitter Resistance Calibration This bus controls the parallel termination of the transmitter drivers and sets termination to a value based on the calibration algorithm performed in the support resistance calibration module. tx_rescal[6:0] provides 64 termination levels; the specific values are defined from lab test results.
rescal_rep[6:0]	Resistance Calibration Replica This bus controls the bias voltage of the transmitter driver. rescal_rep[6:0] provides 64 voltage levels; the specific values are defined from lab test results.

34.7.6 Clock Calibration Control (HDMI_PHY_CKCALCTRL)

Register name: CKCALCTRL

Access type: Read/write/override

Address: 0x05

Value at reset: 0x0000

Address: 0h base + 5h offset = 5h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved	ck_rescal[6:0]						
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_CKCALCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–7 -	This field is reserved. Reserved
ck_rescal[6:0]	Clock Resistance Calibration This bus controls the termination of the clock driver and sets the termination to a value based on the calibration algorithm performed in the support resistance calibration module. ck_rescal[6:0] provides 64 termination levels; the specific values are defined from lab test results.

34.7.7 Color Depth, Pixel Repetition, Clock Divider for PLL and MPLL, and Edge Rate Control (HDMI_PHY_CPCE_CTRL)

Register name: -

Access type: Read/write

Address: 0x06

Value at reset: 0x0400

Address: 0h base + 6h offset = 6h

Bit	15	14	13	12	11	10	9	8
Read Write	Reserved	prep_div[1:0]		tx_edgerate[1:0]		ck_edgerate[1:0]		mpll_n_ cntrl[1:0]
Reset	0	0	0	0	0	1	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	mpll_n_ cntrl[1:0]	pll_n_cntrl[1:0]		pixel_rep[2:0]			clr_dpth[1:0]	
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_CPCE_CTRL field descriptions

Field	Description
15 -	This field is reserved. Reserved
14–13 prep_div[1:0]	Digital Pixel Repetition Divider Controls the ratio by which the internal TMDS clock is divided to generate PREPCLK. 00 Divide by 1 (8 bit). 01 Divide by 1.25 (10 bits). 10 Divide by 1.5 (12 bits). 11 Divide by 2 (16 bits).
12–11 tx_edgerate[1:0]	Transmitter Edge Rate

Table continues on the next page...

HDMI_PHY_CPCE_CTRL field descriptions (continued)

Field	Description
	This bus controls the slew rate of the transmitter output driver. 0 tx_edgerate[0]: Slow edges 1 tx_edgerate[0]: Fast edges 0 tx_edgerate[1]: Disable edge rate override. 1 tx_edgerate[1]: Enable edge rate override.
10–9 ck_edgerate[1:0]	Clock Edge Rate This bus controls the slew rate of the clock output driver. 0 ck_edgerate[0]: Slow edges 1 ck_edgerate[0]: Fast edges 0 ck_edgerate[1]: Disable edge rate override. 1 ck_edgerate[1]: Enable edge rate override.
8–7 mpll_n_cntrl[1:0]	Programmable Divider Control This bus controls the programmable divider modulus, which are set based on the ck_ref_mpll_p/m (TMDS rate) input reference frequency to keep the ring oscillator within the required range (925 MHz through 1.85 GHz in MPLL). 00 N = 1 (for TMDS rate of 184.1-370 MHz) 01 N = 2 (for TMDS rate of 92.51-185 MHz) 10 N = 4 (for TMDS rate of 45.26-92.5 MHz) 11 N = 8 (for TMDS rate up to 45.25 MHz)
6–5 pll_n_cntrl[1:0]	Programmable Divider Control This bus controls the programmable divider modules, which are set based on the refclk_p/m (pixel rate) input reference frequency to keep the ring oscillator within the required range (740 MHz through 1.48 GHz in PLL). 00 N = 1 (for TMDS rate of 184.1-370 MHz) 01 N = 2 (for TMDS rate of 92.51-185 MHz) 10 N = 4 (for TMDS rate of 45.26-92.5 MHz) 11 N = 8 (for TMDS rate (not the pixel rate) up to 45.25 MHz)
4–2 pixel_rep[2:0]	Pixel Repetition This bus controls another factor by which to divide the input frequency (refclk) by the output TMDS rate (ck_ref_mpll_p/m). FTMDS/Fin = (clr_depth[1:0] x pixel_rep[2]) / (pixel_rep[1:0]) 0 **Pixel_rep[2]** Divide by 1. 1 **Pixel_rep[2]**Divide by 2. 00 **Pixel_rep[1:0]**Divide by 4. 01 **Pixel_rep[1:0]**Divide by 2. 1x **Pixel_rep[1:0]**Divide by 1.
clr_dpth[1:0]	Color Depth This bus controls the factor by which to divide the reference clock (PCLK) by the output TMDS rate (ck_ref_mpll_p/m). 00 Divide by 4. 01 Divide by 5.

Table continues on the next page...

HDMI_PHY_CPCE_CTRL field descriptions (continued)

Field	Description
	10 Divide by 6. 11 Divide by 8.

34.7.8 Tx and Clock Measure Control (HDMI_PHY_TXCLKMEASCTRL)

Register name: TXCKMEASCTRL

Access type: Read/write

Address: 0x07

Value at reset: 0x0000

Table 34-25. tx_meas_iv2[7:0]

Effective Bit	Register Value	Description
tx_meas_iv2[0]	00000001	Connect V_{cm_p} (common DC voltage of the positive side of scope) to the analog test bus.
tx_meas_iv2[1]	00000010	Connect V_{cm_m} (common DC voltage of the negative side of scope) to the analog test bus.
tx_meas_iv2[2]	00000100	Connect V_{bg3by4_reg} (output node of tx_vreg_vbgby2 block) to the analog test bus.
tx_meas_iv2[3]	00001000	Connect t_{x_vref} (reference voltage of tx_biasgen block) to the analog test bus.
tx_meas_iv2[4]	00010000	Connect V_{rep_fb} (feedback voltage of the replica circuit of tx_biasgen block) to the analog test bus.
tx_meas_iv2[5]	00100000	In the event that bleed current is too large, this bit can be used to force vb closer to gnd and disable/reduce the bleed current. (This bit is a debug feature, which does not pull vb to gnd properly.)
tx_meas_iv2[6]	01000000	Connect V_p (low power supply) to the analog test bus.
tx_meas_iv2[7]	10000000	Connect V_{cm} (common DC voltage of the scope) to the analog test bus.

Table 34-26. ck_meas_iv[7:0]

Effective Bit	Register Value	Description
ck_meas_iv[0]	00000001	Connect V_{cm_p} (common DC voltage of the positive side of scope) to the analog test bus.
ck_meas_iv[1]	00000010	Connect V_{cm_m} (common DC voltage of the negative side of scope) to the analog test bus.
ck_meas_iv[2]	00000100	Connect V_{bg3by4_reg} (output node of tx_vreg_vbgby2 block) to the analog test bus.

Table continues on the next page...

Table 34-26. ck_meas_iv[7:0] (continued)

ck_meas_iv[3]	00001000	Connect t_{x_vref} (reference voltage of tx_biasgen block) to the analog test bus.
ck_meas_iv[4]	00010000	Connect V_{rep_fb} (feedback voltage of the replica circuit of tx_biasgen block) to the analog test bus.
ck_meas_iv[5]	00100000	In the event that bleed current is too large, this bit can be used to force v_b closer to gnd and disable/reduce the bleed current. (This bit is a debug feature, which does not pull v_b to gnd properly.)
ck_meas_iv[6]	01000000	Connect V_p (low power supply) to the analog test bus.
ck_meas_iv[7]	10000000	Connect V_{cm} (common DC voltage of scope) to the analog test bus.

Address: 0h base + 7h offset = 7h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	tx_meas_iv2[7:0]								ck_meas_iv[7:0]							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_TXCLKMEASCTRL field descriptions

Field	Description
15–8 tx_meas_iv2[7:0]	Transmitter Measure Internal Voltage 2 This bus is used to test specific voltages for third transmitter lane/channel by applying voltages on the atb_sense port based on configured currents, as described in the tx_meas_iv2[7:0] table.
ck_meas_iv[7:0]	Clock Measure Internal Voltage This bus is used to test specific voltages for the clock lane/channel by applying voltages on the atb_sense port based on configured currents, as described in the ck_meas_iv[7:0] table.

34.7.9 Tx Measure Control (HDMI_PHY_TXMEASCTRL)

Register name: TXMEASCTRL

Access type: Read/write

Address: 0x08

Value at reset: 0x0000

Table 34-28. tx_meas_iv1[7:0]

Effective Bit	Register Value	Description
tx_meas_iv1[0]	00000001	Connect V_{cm_p} (common DC voltage of positive side of scope) to the analog test bus.
tx_meas_iv1[1]	00000010	Connect V_{cm_m} (common DC voltage of negative side of scope) to the analog test bus.
tx_meas_iv1[2]	00000100	Connect V_{bg3by4_reg} (output node of tx_vreg_vbgby2 block) to the analog test bus.

Table continues on the next page...

Table 34-28. tx_meas_iv1[7:0] (continued)

tx_meas_iv1[3]	00001000	Connect t_{x_vref} (reference voltage of tx_biasgen block) to the analog test bus.
tx_meas_iv1[4]	00010000	Connect V_{rep_fb} (feedback voltage of replica circuit of tx_biasgen block) to the analog test bus.
tx_meas_iv1[5]	00100000	In case bleed current is too large, this bit can be used to force v_b closer to gnd and disable/reduce the bleed current. (This bit is a debug feature, which does not pull v_b to gnd properly.)
tx_meas_iv1[6]	01000000	Connect V_p (low power supply) to the analog test bus.
tx_meas_iv1[7]	10000000	Connect V_{cm} (common DC voltage of scope) to the analog test bus.

Table 34-29. tx_meas_iv0[7:0]

Effective Bit	Register Value	Description
tx_meas_iv0[0]	00000001	Connect V_{cm_p} (common DC voltage of positive side of scope) to the analog test bus.
tx_meas_iv0[1]	00000010	Connect V_{cm_m} (common DC voltage of negative side of scope) to the analog test bus.
tx_meas_iv0[2]	00000100	Connect V_{bg3by4_reg} (output node of tx_vreg_vbgby2 block) to the analog test bus.
tx_meas_iv0[3]	00001000	Connect t_{x_vref} (reference voltage of tx_biasgen block) to the analog test bus
tx_meas_iv0[4]	00010000	Connect V_{rep_fb} (feedback voltage of replica circuit of tx_biasgen block) to the analog test bus.
tx_meas_iv0[5]	00100000	In the event that bleed current is too large, this bit can be used to force v_b closer to gnd and disable/reduce the bleed current. (This bit is a debug feature, which does not pull v_b to gnd properly.)
tx_meas_iv0[6]	01000000	Connect V_p (low power supply) to the analog test bus.
tx_meas_iv0[7]	10000000	Connect V_{cm} (common DC voltage of scope) to the analog test bus.

Address: 0h base + 8h offset = 8h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	tx_meas_iv0[7:0]								tx_meas_iv1[7:0]							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_TXMEASCTRL field descriptions

Field	Description
15–8 tx_meas_iv0[7:0]	Transmitter Measure Internal Voltage 0 This bus is used to test specific voltages for the first transmitter lane/channel by applying voltages on the atb_sense port based on configured currents, as described in the tx_meas_iv0[7:0] table.
tx_meas_iv1[7:0]	Transmitter Measure Internal Voltage 1 This bus is used to test specific voltages for the second transmitter lane/channel by applying voltages on the atb_sense port based on configured currents, as described in the tx_meas_iv1[7:0] table.

34.7.10 Clock Symbol and Transmitter Control (HDMI_PHY_CKSYMTXCTRL)

Register name: CKSYMTXCTRL

Access type: Read/write/override

Address: 0x09

Value at reset: 0x0009

Address: 0h base + 9h offset = 9h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved				tx_symon	tx_traon	tx_trbon	ck_symon
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_CKSYMTXCTRL field descriptions

Field	Description
15 Override	Writing a 0 to the override bit causes the register to set the value 0x0F, regardless of the other bit settings. Setting the override bit to 1 causes tx_symon, tx_traon, tx_trbon, and ck_symon bits to work per the register settings. These bits must be set at the same time the override bit is set to 1, or while the override bit remains at 1.
14–4 -	This field is reserved. Reserved
3 tx_symon	Transmitter Symbol On This bit enables the transmitter symbol driver(s), To enable the transmitter driver(s), the tx_pwron bit for each channel must be high. 0 Disable the transmitter symbol driver(s). 1 Enable the transmitter symbol driver(s).
2 tx_traon	Transmitter Trailer A On This bit enables the transmitter trailer A driver(s). To enable the transmitter trailer A driver(s) and to enable pre-emphasis, the tx_pwron bit for each channel must be high. 0 Disable the transmitter trailer A driver(s). 1 Enable the transmitter trailer A driver(s).
1 tx_trbon	Transmitter Trailer B On This bit enables the transmitter trailer B driver(s). To enable the transmitter trailer B driver(s) and to enable pre-emphasis, the tx_pwron bit for each channel must be high. 0 Disable the transmitter trailer B driver(s). 1 Enable the transmitter trailer B driver(s).

Table continues on the next page...

HDMI_PHY_CKSYMTXCTRL field descriptions (continued)

Field	Description
0 ck_symon	Clock Symbol On This bit enables the clock symbol driver. To enable the clock driver, the ck_powon bit must be high. In addition, there is no pre-emphasis enable for the clock driver. 0 Disable the clock symbol driver. 1 Enable the clock symbol driver.

34.7.11 Comparator Sequence Control (HDMI_PHY_CMPSEQCTRL)

Register name: CMPSEQCTRL

Access type: Read/write/override

Address: 0x0A

Value at reset: 0x0000

Address: 0h base + Ah offset = Ah

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Override	Reserved														sup_comp_rt_r
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_CMPSEQCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–1 -	This field is reserved. Reserved
0 sup_comp_rt_r	Support Comparator Resistance Termination This bit controls the comparator sequence. 0 Latch the first input, if the Override bit is 0. 1 Latch the second input, then set the comparator's output pin by the result of comparison, if the Override bit is 0.

34.7.12 Comparator Power Control (HDMI_PHY_CMPPWRCTRL)

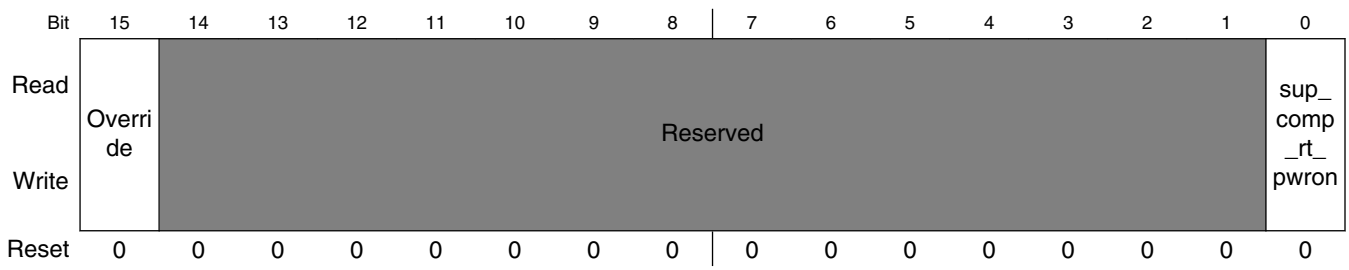
Register name: CMPPWRCTRL

Access type: Read/write/override

Address: 0x0B

Value at reset: 0x0000

Address: 0h base + Bh offset = Bh



HDMI_PHY_CMPPWRCTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–1 -	This field is reserved. Reserved
0 sup_comp_rt_pwron	Support Comparator Resistance Termination Power-On This bit powers on the Comparator module. 0 Power off the Comparator module and connect the comparator's output to ground, if the Override bit is 0. 1 Power on the Comparator module, if the Override bit is 0.

34.7.13 Comparator Mode Control (HDMI_PHY_CMPMODECTRL)

Register name: CMPMODECTRL

Access type: Read/write/override

Address: 0x0C

Value at reset: 0x0000

Address: 0h base + Ch offset = Ch

Bit	15	14	13	12	11	10	9	8
Read / Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read / Write	Reserved							sup_comp_mode
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_CMPMODECTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–1 -	This field is reserved. Reserved
0 sup_comp_mode	Support Comparator Mode This bit selects the comparator mode. 0 Testing mode (ADC mode) 1 Calibration mode

34.7.14 Measure Control (HDMI_PHY_MEASCTRL)

Register name: MEASCTRL

Access type: Read/write

Address: 0x0D

Value at reset: 0x0000

NOTE

Two or more of the previous register bits must not be set to 1 simultaneously; doing so can lead to a hardware problem.

Address: 0h base + Dh offset = Dh

Bit	15	14	13	12	11	10	9	8
Read / Write	Reserved							
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read / Write	Reserved				sup_atb_on_rext	sup_por_meas_iv[1:0]		sup_dac_on_atb
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_MEASCTRL field descriptions

Field	Description
15–4 -	This field is reserved. Reserved
3 sup_atb_on_rext	Support Analog Test Bus On External Calibration Resistance This bit connects or disconnects the V_{rext} node to/from the analog test bus. 0 Disconnect the V_{rext} node from the analog test bus. 1 Connect the V_{rext} node to the analog test bus.
2–1 sup_por_meas_iv[1:0]	Support Power Measure Internal Voltage This bus connects or disconnects a single output signal on the analog test bus to measure the voltage of two nodes of the support power block. 01 Connect V_{be} (Bipolar transistor voltage) to the analog test bus. 10 Connect V_{bg} (Band-gap voltage) to the analog test bus.
0 sup_dac_on_atb	Support Digital-to-Analog Converter On Analog Test Bus This bit connects or disconnects the DAC's output on the analog test bus to test the performance of the DAC through Integral Non-Linearity (INL) and Differential Non-Linearity (DNL). 0 Disconnect the DAC's output from the analog test bus. 1 Connect the DAC's output to the analog test bus.

34.7.15 Voltage Level Control (HDMI_PHY_VLEVCTRL)

Register name: VLEVCTRL

Access type: Read/write

Address: 0x0E

Value at reset: 0x0000

Address: 0h base + Eh offset = Eh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	Reserved						sup_tx_lvl[4:0]					sup_ck_lvl[4:0]				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_VLEVCTRL field descriptions

Field	Description
15–10 -	This field is reserved. Reserved
9–5 sup_tx_lvl[4:0]	Support Transmitter Level This bus controls the reference voltage level of the three transmitter channel modules. This voltage reference has a direct relation with the output signal voltage level. For more information about the driver voltage level configuration, see Appendix A: Driver Voltage Level Configuration.

Table continues on the next page...

HDMI_PHY_VLEVCTRL field descriptions (continued)

Field	Description
sup_ck_lvl[4:0]	Support Clock Level This bus controls the reference voltage level of the Clock Channel module. This voltage reference has a direct relation with the output signal voltage level. For more information about the driver voltage level configuration, see Appendix A: Driver Voltage Level Configuration.

34.7.16 Digital-to-Analog Control (HDMI_PHY_D2ACTRL)

Register name: D2ACTRL

Access type: Read/write/override

Address: 0x0F

Value at reset: 0x0000

Address: 0h base + Fh offset = Fh

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved				sup_dac_n[7:0]		
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	sup_dac_n[7:0]					sup_dac_th_n[2:0]		
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_D2ACTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–11 - Reserved	This field is reserved.
10–3 sup_dac_n[7:0]	Support Analog-to-Digital Inverted This bus represents the LSB's of the 9-bit DAC value.
sup_dac_th_n[2:0]	Support Digital-to-Analog Thermometer Inverted This bus value is inverted and defined in the thermometer code to represent the binary code of the two MSB's of the 9-bit DAC value. **Note**: To increase the stability of the DAC block, the two MSB's of the 9-bit DAC value are represented in thermometer code, not in binary code. The MSB's of the 9-bit DAC value is split into two thermometer-code bits. A transition from 0 to 1 of the 9-bit DAC value's MSB is represented by "00" -> "01" -> "11" in thermometer code.

34.7.17 Current Control (HDMI_PHY_CURRCTRL)

Register name: CURRCTRL

Access type: Read/write

Address: 0x10

Value at reset: 0x08AB

Address: 0h base + 10h offset = 10h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	Reserved				mpll_prop_cntrl[2:0]			mpll_int_cntrl[2:0]			pll_prop_cntrl[2:0]			pll_int_cntrl[2:0]		
Reset	0	0	0	0	1	0	0	0	1	0	1	0	1	0	1	1

HDMI_PHY_CURRCTRL field descriptions

Field	Description
15–12 -	This field is reserved. Reserved
11–9 mpll_prop_ cntrl[2:0]	MPLL Proportional Control This bus controls the MPLL charge pump proportional current. Eight levels of charge pump proportional current value are possible. The specific values are defined in PLL/MPLL Generic Configuration Settings. Default (reset) value of pll_prop_cntrl[1:0] is 100.
8–6 mpll_int_cntrl[2:0]	MPLL Integral Control This bus controls the charge pump integral current. Eight levels of charge pump integral current value are possible. The specific values are defined in PLL/MPLL Generic Configuration Settings. Default (reset) value of pll_int_cntrl[1:0] is 100.
5–3 pll_prop_ cntrl[2:0]	PLL Proportional Control This bus controls the PLL charge pump proportional current. Eight levels of charge pump proportional current value are possible. The specific values are defined in PLL/MPLL Generic Configuration Settings. Default (reset) value of pll_int_cntrl is 011.
pll_int_cntrl[2:0]	PLL Charge Pump Integral Control This bus controls the PLL charge pump integral current. Eight levels of charge pump integral current value are possible. The specific values are defined in PLL/MPLL Generic Configuration Settings. Default (reset) value of pll_int_cntrl[2:0] is 100.

34.7.18 Drive Analog Control (HDMI_PHY_DRVANACTRL)

Register name: DRVANACTRL

Access type: Read/write/override

Address: 0x11

Value at reset: 0x0003

Address: 0h base + 11h offset = 11h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved						pll_drv_ana	mpll_drv_ana
Reset	0	0	0	0	0	0	1	1

HDMI_PHY_DRVANACTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–2 -	This field is reserved. Reserved
1 pll_drv_ana	PLL Drive Analog This bit enables or disables driving the ck_ref_mpll_p/m clocks to the MPLL module. Default (reset) value of pll_drv_ana is 1. 0 Set ck_ref_mpll low, if the Override bit is 0. 1 Enable 25-340 MHz output clock (ck_ref_mpll_p/m) to be driven to MPLL, if the Override bit is 0.
0 mpll_drv_ana	MPLL Drive Analog This bit enables or disables driving the ck_ref_p/m clocks to all transmitters (tx_topa). Default (reset) value of mpll_drv_ana is 1. 0 Set ck_ref_p high; ck_ref_m low, if the Override bit is 0. 1 Enable ck_ref_mpll output clock (125-1700 MHz ck_ref_p/m) to be driven to all transmitters (tx_topa), if the Override bit is 0.

34.7.19 PLL Measure Control (HDMI_PHY_PLLMEASCTRL)

Register name: PLLMEASCTRL

Access type: Read/write

Address: 0x12

Value at reset: 0x0000

NOTE

With the exception of pll_atb_sense_sel and pll_meas_iv[9], two or more of the previous register bits must not be set to 1 simultaneously; doing so can lead to a hardware problem.

The table below describes the pll_meas_iv[10:0] bit settings.

Table 34-40. pll_meas_iv[10:0]

Effective Bit	Pll_meas_iv[10:0] Value	Description
Pll_meas_iv[0]	00000000001	Not used.
Pll_meas_iv[1]	00000000010	Not used in pll_top. Connected to fast_tech pins of PLL and MPLL in hdmi_topa.
Pll_meas_iv[2]	00000000100	Connects VP supply voltage to the atb_sense line.
Pll_meas_iv[3]	00000001000	Connects vp_cp to the atb_sense line.
Pll_meas_iv[4]	00000010000	Connects internal supply voltage of the VCO (ivco) to the atb_sense line.
Pll_meas_iv[5]	00000100000	Connects vp_cko voltage to the atb_sense line.
Pll_meas_iv[6]	00001000000	Connects node vpsf to the atb_sense line.
Pll_meas_iv[7]	00010000000	Connects vref to atb_sense line.
Pll_meas_iv[8]	00100000000	Connects vcntrl to atb_sense line.
Pll_meas_iv[9]	01000000000	Enables the phase mixer and pll_cko_pm_p/m.
Pll_meas_iv[10]	10000000000	Measures the voltage corresponding to the output phase of the clr_dpth divider (fb_clk) with respect to refclk.

Address: 0h base + 12h offset = 12h

Bit	15	14	13	12	11	10	9	8
Read Write	Reserved			pll_atb_ sense_sel	pll_meas_iv[10:0]			
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	pll_meas_iv[10:0]							pll_meas_gd
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_PLLMEASCTRL field descriptions

Field	Description
15–13 -	This field is reserved. Reserved
12 pll_atb_sense_ sel	PLL Analog Test Bus Sense Select This bit enables or disables internal signals of the PLL to be connected to the analog test bus. Without setting this bit, no measurements can be made on the atb_sense line. Default (reset) value of pll_atb_sense_sel is 0. 0 Disable the ability to measure internal DC signals on the atb_sense line in the PLL. 1 Enable the ability to measure internal DC signals on the atb_sense line in the PLL.

Table continues on the next page...

HDMI_PHY_PLLMEASCTRL field descriptions (continued)

Field	Description
11–1 pll_meas_iv[10:0]	PLL Measure Internal Voltage This bus enables or disables measuring various PLL node voltages and branch currents. For information about the bit settings, see the pll_meas_iv[10:0] table.
0 pll_meas_gd	PLL Measure Ground This bit connects or disconnects the ground signal to the atb_sense (analog test bus) bus. 0 Disconnect the ground signal from the atb_sense bus. 1 Connect the ground signal to the atb_sense bus.

34.7.20 PLL Phase and Bypass Control (HDMI_PHY_PLLPHBYCTRL)

Register name: PLLPHBYCTRL

Access type: Read/write

Address: 0x13

Value at reset: 0x0000

Address: 0h base + 13h offset = 13h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved				bypass_ppll	pll_ph_sel_ck	pll_ph_sel[9:0]									
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_PLLPHBYCTRL field descriptions

Field	Description
15–12 -	This field is reserved. Reserved
11 bypass_ppll	Bypass Pre-PLL This bit enables or disables bypassing the pre-PLL. 0 Disable bypassing pll_top by forcing refclk_mpll_ref low. 1 Enable bypassing pll_top by buffering refclk to refclk_mpll_ref.
10 pll_ph_sel_ck	PLL Phase Select Clock This bit enables or disables latching the ph_sel[9:0] into a 9-bit DAC used in the phase mixer.

Table continues on the next page...

HDMI_PHY_PLLPHBYCTRL field descriptions (continued)

Field	Description
	0 Disable latching the pll_ph_sel[9:0] into the 9-bit DAC. 1 Enable latching the pll_ph_sel[9:0] into the 9-bit DAC.
pll_ph_sel[9:0]	PLL Phase Select This bus is a control word for the PLL's phase mixer that enables the phase of pll_cko_pm_p/m to be varied ± 0.5 UI of the VCO frequency, which is 740-1,480 MHz.

34.7.21 Gear Shift, Reset Mode, and Power State Control (HDMI_PHY_GRP_CTRL)

Register name: -

Access type: Read/write/override

Address: 0x14

Value at reset: 0x0000

Address: 0h base + 14h offset = 14h

Bit	15	14	13	12	11	10	9	8
Read Write	Override	Reserved						
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		pll_pwr_on	pll_rst	mpll_pwr_on	mpll_rst	pll_gear_shift	mpll_gear_shift
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_GRP_CTRL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–6 -	This field is reserved. Reserved
5 pll_pwr_on	PLL Power-On This bit is used to power on/off the PLL module. Note: If the Override bit is set to 1, the working value is the Override bit value, not the registered value. 0 Power off PLL and draw minimal current, if the Override bit is 0. 1 Power on PLL and enable it to operate normally, if the Override bit is 0.
4 pll_rst	PLL Reset This bit is used to place the MPLL in Reset mode.

Table continues on the next page...

HDMI_PHY_GRP_CTRL field descriptions (continued)

Field	Description
	0 Enable PLL to operate normally, if the Override bit is 0. 1 Place the PLL in Reset mode, if the Override bit is 0.
3 mpll_pwr_on	MPLL Power-On This bit is used to power-on/off the MPLL module. 0 Power on MPLL and set all output clocks to DC levels, if the Override bit is 0. 1 Power off MPLL and enable it to operate normally, if the Override bit is 0.
2 mpll_rst	MPLL Reset This bit is used to place the MPLL in Reset mode. 0 Enable MPLL to operate normally, if the Override bit is 0. 1 Place the MPLL in Reset mode, if the Override bit is 0.
1 pll_gear_shift	PLL Gear Shift This bit enables or disables Rapid Locking mode, where the pll_gear_shift bit is asserted for 25 µs when coming out of reset, then deasserted before clocks are valid. 0 Disable Rapid Locking mode, if the Override bit is 0. 1 Enable Rapid Locking mode, if the Override bit is 0.
0 mpll_gear_shift	MPLL Gear Shift This bit enables or disables Rapid Locking mode, where the mpll_gear_shift bit is asserted for 25 µs when coming out of reset, then deasserted before clocks are valid. 0 Disable Rapid Locking mode, if the Override bit is 0. 1 Enable Rapid Locking mode, if the Override bit is 0.

34.7.22 Gmp Control (HDMI_PHY_GMPCTRL)

Register name: GMPCTRL

Access type: Read/write

Address: 0x15

Value at reset: 0x0000

Address: 0h base + 15h offset = 15h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	Reserved												pll_gmp_cntrl[1:0]		mpll_gmp_cntrl[1:0]	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_GMPCTRL field descriptions

Field	Description
15–4 -	This field is reserved. Reserved
3–2 pll_gmp_cntrl[1:0]	PLL gmp Control This bus controls the effective loop-filter resistance (equal) to increase or decrease PLL bandwidth and to compensate for changes in the Divider module (n_cntrl). 00 TMDS rate up to 45.25 MHz 01 TMDS rate of 45.26-92.5 MHz 10 TMDS rate of 92.51-185 MHz 11 TMDS rate of 184.1-370 MHz
mpll_gmp_ cntrl[1:0]	MPLL gmp Control This bus controls the effective loop-filter resistance (= 1/gmp) to increase or decrease MPLL bandwidth and to compensate for changes in the Divider module (n_cntrl). 00 TMDS rate up to 45.25 MHz 01 TMDS rate of 45.26-92.5 MHz 10 TMDS rate of 92.51-185 MHz 11 TMDS rate of 184.1-370 MHz

34.7.23 MPLL Measure Control (HDMI_PHY_MPLLMEASCTRL)

Register name: MPLLMEASCTRL

Access type: Read/write

Address: 0x16

Value at reset: 0x0000

NOTE

With the exception of mpll_atb_sense_sel, mpll_meas_iv[9], and mpll_meas_iv[11], two or more of the previous register bits must not be set to 1 simultaneously, because doing so can lead to a hardware problem.

The table below describes the mpll_meas_iv[11:0] bit settings.

Table 34-45. mpll_meas_iv[11:0]

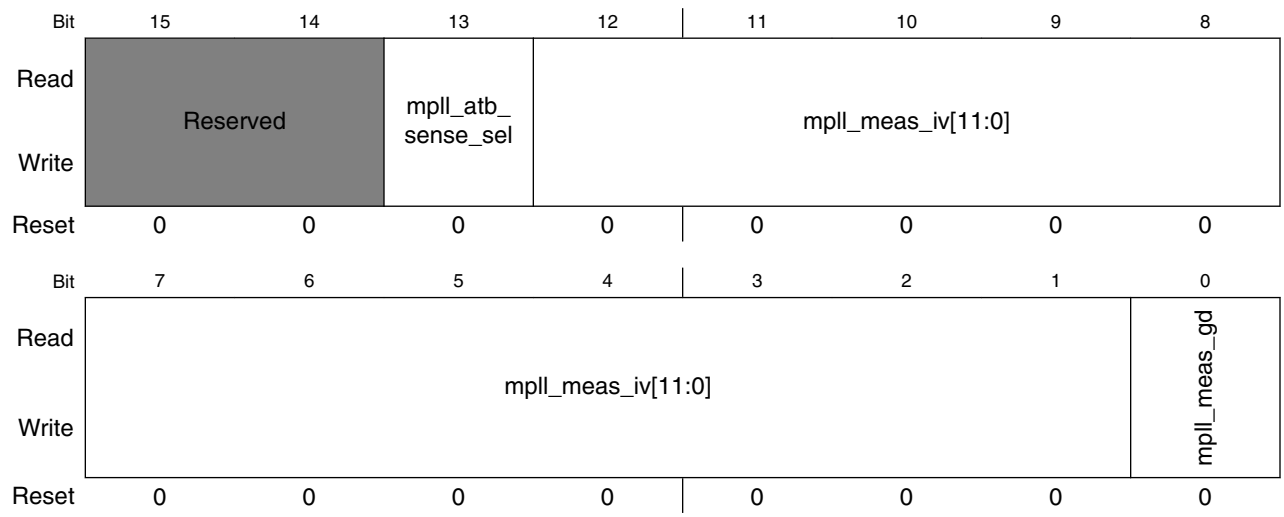
Effective Bit	mpll_meas_iv[11:0] Value	Description
mpll_meas_iv[0]	000000000001	Connects the internal positive DCC control line (vc0p) to the atb_sense line.
mpll_meas_iv[1]	000000000010	Connects the internal negative DCC control line (vc0m) to the atb_sense line.

Table continues on the next page...

Table 34-45. mpll_meas_iv[11:0] (continued)

Effective Bit	mpll_meas_iv[11:0] Value	Description
mpll_meas_iv[2]	000000000100	Connects VP supply voltage to the atb_sense line.
mpll_meas_iv[3]	000000001000	Connects vp_cp to the atb_sense line.
mpll_meas_iv[4]	000000010000	Connects the internal supply voltage of the VCO (ivco) to the atb_sense line through a low-pass filter.
mpll_meas_iv[5]	000000100000	Connects the vp_cko voltage to the atb_sense line.
mpll_meas_iv[6]	000001000000	Connects the vpsf node to the atb_sense line.
mpll_meas_iv[7]	000010000000	Connects vref to the atb_sense line.
mpll_meas_iv[8]	000100000000	Connects vcntrl to the atb_sense line.
mpll_meas_iv[9]	001000000000	Enables the phase mixer and cko_pm_p/m.
mpll_meas_iv[10]	010000000000	Measures the voltage corresponding to the phase of the divide-by-5 output (fb_clk) with respect to the phase of the mpll_ana input (ck_refclk_p).
mpll_meas_iv[11]	100000000000	Forces div_x1 low; therefore, bitclk_p/m is multiplexed to ck_ref_p/m (normally, if n_cntrl = 00, ck0_p/m is multiplexed to ck_ref_p/m and DCC loop is enabled.

Address: 0h base + 16h offset = 16h



HDMI_PHY_MPLLMEASCTRL field descriptions

Field	Description
15–14 -	This field is reserved. Reserved
13 mpll_atb_sense_ sel	MPLL Analog Test Bus Sense Select This bit enables or disables internal signals of the PLL to be connected to the analog test bus. Without setting this bit, no measurements can be made on the atb_sense line. Default (reset) value of mpll_atb_sense_sel is 0. 0 Disable the ability to measure internal DC signals on the atb_sense line in the PLL. 1 Enable the ability to measure internal DC signals on the atb_sense line in the PLL.

Table continues on the next page...

HDMI_PHY_MPLLMEASCTRL field descriptions (continued)

Field	Description
12–1 mpll_meas_iv[11:0]	MPLL Measure Internal Voltage This bus enables or disables measuring various PLL node voltages and branch currents. For information about the bit settings, see the mpll_meas_iv[11:0] table.
0 mpll_meas_gd	MPLL Measure Ground This bit connects or disconnects the ground signal to the atb_sense (analog test bus) bus. 0 Disconnect the ground signal from the atb_sense bus. 1 Connect the ground signal to the atb_sense bus.

34.7.24 MPLL and PLL Phase, Scope Clock Select, and MUX Clock Control (HDMI_PHY_MSM_CTRL)

Register name: -

Access type: Read/write

Address: 0x17

Value at reset: 0x0000

The table below describes the cko_sel[1:0] bit settings. (The cko_sel[1:0] default value is 00.)

Table 34-47. cko_sel[1:0]

cko_sel[1:0]	TMDS Clock Mode	PLL and MPLL Mode	Output TMDS Clock on TMDS Channel
00	Non-Coherent	Normal	ck_ref_mpll_p/m (Pre-PLL out)
01	Off (PLL and MPLL are both off)	Normal	Off (No output TMDS clock)
10	Test (PHY in test mode)	Normal	PCLK (Input reference clock to HDMI 3D Tx PHY)
11	Coherent	Normal	fb_clk (MPLL feedback clock)
00	Off (PLL and MPLL are both off)	Bypass	Off (No output TMDS clock)
01	Off (PLL and MPLL are both off)	Bypass	Off (No output TMDS clock)
10	Non-Coherent	Bypass	PCLK (Input reference clock to HDMI 3D Tx PHY)
11	Coherent	Bypass	fb_clk (MPLL feedback clock)

Address: 0h base + 17h offset = 17h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved		mpll_ph_sel_ck	mpll_ph_sel[9:0]										cko_sel[1:0]		scope_ck_sel
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_MSM_CTRL field descriptions

Field	Description
15–14 -	This field is reserved. Reserved
13 mpll_ph_sel_ck	MPLL Phase Select Clock This bit enables or disables latching ph_sel[9:0] into a 9-bit DAC used in the phase mixer. 0 Disable latching mpll_ph_sel[9:0] into the 9-bit DAC. 1 Enable latching mpll_ph_sel[9:0] into the 9-bit DAC.
12–3 mpll_ph_sel[9:0]	MPLL Phase Select This bus is a control word for the MPLL's phase mixer and enables the phase of pll_cko_pm_p /m to be varied ± 0.5 UI of the VCO frequency, which has a range of 925-1,850 MHz.
2–1 cko_sel[1:0]	Clock Output Select This bus selects the clock to be connected to the output TMDS clock channel. **Notes:** • Normal mode: The color depth or pixel repetition is required, which means that the PLL is powered on (pll_pwr_on bit is set to 1) and the bypass_ppll bit is set to 0. • Bypass mode: The color depth and the pixel repetition is not required, which means that the PLL is powered off (pll_pwr_on bit is set to 0) and the bypass_ppll bit is set to 1. For information about the cko_sel[1:0] bit settings and corresponding PLL/MPLL modes, see the cko_sel[1:0] table.
0 scope_ck_sel	Scope Clock Select Selects the clock to connect to the scope clock signal: the differential pll_cko_p/m or the differential mpll_cko_p/m. 0 Connect the mpll_cko_pm_p/m to the clock scope (this clock has a range of 23.125-45.25 MHz) where the mpll_meas_iv[9] bit must be high. 1 Connect the pll_cko_pm_p/m to the clock scope (this clock has a range of 23.125-45.25 MHz) where the pll_meas_iv[9] bit must be high.

34.7.25 Scope, Comparator Result and Power Bad Status (HDMI_PHY_SCRPB_STATUS)

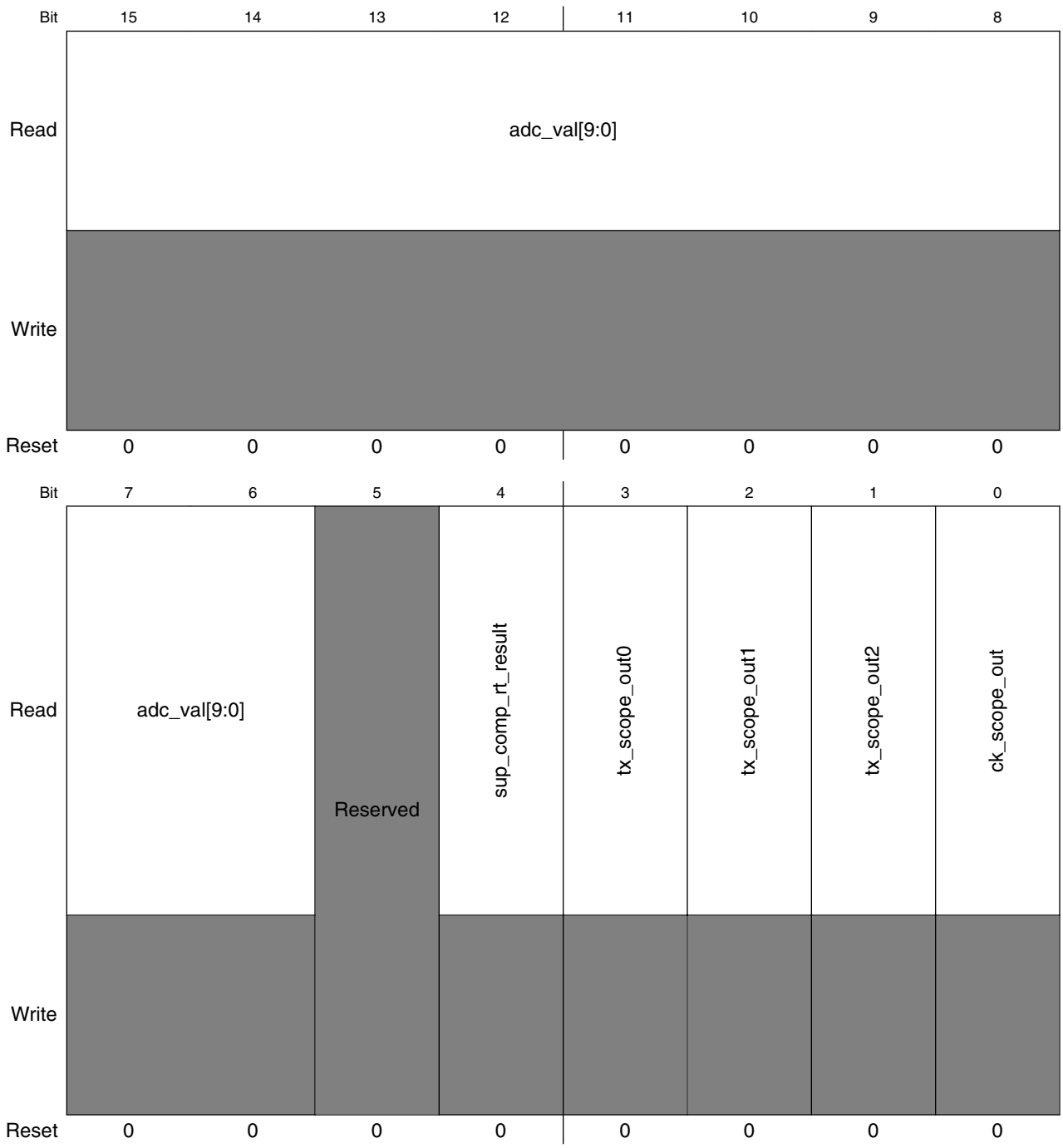
Register name: -

Access type: Read-only

Address: 0x18

Value at reset: N/A

Address: 0h base + 18h offset = 18h



HDMI_PHY_SCRPB_STATUS field descriptions

Field	Description
15–6 adc_val[9:0]	ADC/DAC bit word (analog signal value)
5 -	This field is reserved. Reserved
4 sup_comp_rt_result	Support Comparator Resistance Termination Result This bit represents the result of the comparison process. 0 The first input is greater than or equal to the second input. 1 The first input is less than the second input.
3 tx_scope_out0	Scoping Value for Lane 0 0 The driver output of lane 0 is not differential. 1 The driver output of lane 0 is differential.
2 tx_scope_out1	Scoping Value for Lane 1 0 The driver output of lane 1 is not differential. 1 The driver output of lane 1 is differential.
1 tx_scope_out2	Scoping Value for Lane 2 0 The driver output of lane 2 is not differential. 1 The driver output of lane 2 is differential.
0 ck_scope_out	Clock Scope Output Signal 0 The clock driver output is not differential. 1 The clock driver output is differential.

34.7.26 Transmission Termination (HDMI_PHY_TXTERM)

Register name: TXTERM

Access type: Read/write

Address: 0x19

Value at reset: 0x0007

Address: 0h base + 19h offset = 19h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	Reserved													d_tx_term[2:0]		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

HDMI_PHY_TXTERM field descriptions

Field	Description
15–3 -	This field is reserved. Reserved
d_tx_term[2:0]	Digital Transmission Termination This bus defines the transmission termination (resistance) value, which is set by the HDMI controller. The formula for the resistance value is: R = 50 / (1 - 0.125 x d_tx_term) This equation is valid only when d_tx_term equals 0-6. • 000: 50 Ω • 001: 56.14 Ω • 010: 66.67 Ω • 011: 80 Ω • 100: 100 Ω • 101: 133.33 Ω • 110: 200 Ω • 111: Open circuit

34.7.27 Power Sequence, TX Clock Alignment, Resistance Calibration, Pattern Generator Skip Bit, and TMDS Encoder Enable (HDMI_PHY_PTRPT_ENBL)

Register name: -

Access type: Read/write/override

Address: 0x1A

Value at reset: 0x0000

Address: 0h base + 1Ah offset = 1Ah

Bit	15	14	13	12	11	10	9	8
Read	Override	Reserved						pg_skip_bit2
Write								
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read	pg_skip_bit1	pg_skip_bit0	ck_ref_enb	rcal_enb	tx_ck_align_enb	tx_ready	cko_word_enb	refclk_enb
Write								
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_PTRPT_ENBL field descriptions

Field	Description
15 Override	If the Override bit is set to 1, the working value is the Override bit value, not the registered value.
14–9 -	This field is reserved. Reserved
8 pg_skip_bit2	Pattern Generator Skip Bit 2 This bit enables or disables pattern generator skip bit feature for channel 2. 0 Disable the pattern generator skip bit feature for the third transmitting channel, if the Override bit is 0. 1 Enable the pattern generator skip bit feature for the third transmitting channel, if the Override bit is 0.
7 pg_skip_bit1	Pattern Generator Skip Bit 1 This bit enables or disables pattern generator skip bit feature for channel 1. 0 Disable the pattern generator skip bit feature for the second transmitting channel, if the Override bit is 0. 1 Enable the pattern generator skip bit feature for the second transmitting channel, if the Override bit is 0.
6 pg_skip_bit0	Pattern Generator Skip Bit 0 This bit enables or disables pattern generator skip bit feature for channel 0. 0 Disable the pattern generator skip bit feature for the first transmitting channel, if the Override bit is 0. 1 Enable the pattern generator skip bit feature for the first transmitting channel, if the Override bit is 0.
5 ck_ref_enb	Clock Reference Enable This bit powers up the clock alignment and the resistance calibration modules. 0 Powers down the clock alignment and the resistance calibration modules, if the Override bit is 0. 1 Powers up the clock alignment and the resistance calibration modules, if the Override bit is 0.
4 rcal_enb	Resistance Calibration Enable This bit enables or disables the resistance clock alignment process. 0 Disable the resistance calibration FSM, if the Override bit is 0. 1 Enable the resistance calibration FSM, if the Override bit is 0.
3 tx_ck_align_enb	Transmission Clock Alignment Enable This bit disables or enables the clock alignment FSM. 0 Disable transmission clock alignment FSM, if the Override bit is 0. 1 Enable transmission clock alignment FSM, if the Override bit is 0.
2 tx_ready	Transmission Ready This bit indicates whether the PHY transmit driver is ready to transmit data. 0 PHY transmit driver is not ready to transmit data, if the Override bit is 0. 1 PHY transmit driver is ready to transmit data, if the Override bit is 0.
1 cko_word_enb	Output Clock Word Enable This bit enables the output word clock. 0 Disable the output clock word, if the Override bit is 0. 1 Enable the output clock word, if the Override bit is 0

Table continues on the next page...

HDMI_PHY_PTRPT_ENBL field descriptions (continued)

Field	Description
0 refclk_enb	Reference Clock Enable This bit enables or disables the input reference clock. 0 Disable the input reference clock, if the Override bit is 0. 1 Enable the input reference clock, if the Override bit is 0.

34.7.28 Pattern Generator Mode (HDMI_PHY_PATTERNGEN)

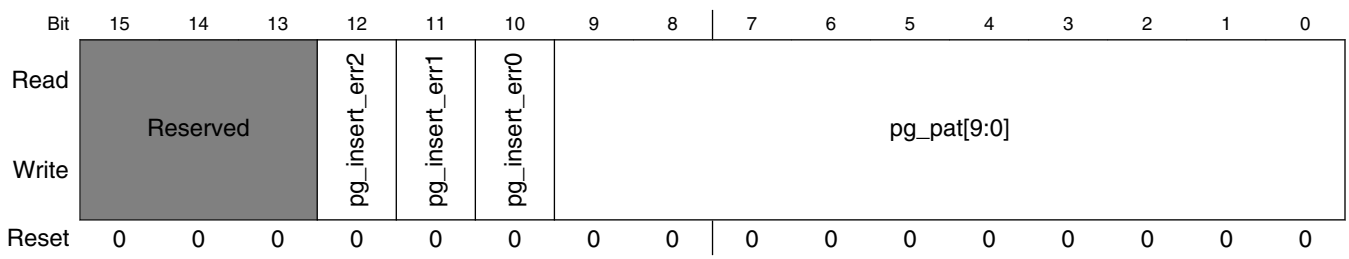
Register name: PATTERNGEN

Access type: Read/write

Address: 0x1B

Value at reset: 0x0000

Address: 0h base + 1Bh offset = 1Bh



HDMI_PHY_PATTERNGEN field descriptions

Field	Description
15–13 -	This field is reserved. Reserved
12 pg_insert_err2	Pattern Generator Insert Error Two This bit enables or disables error insertion inside the generated pattern for channel 2. 0 Do not insert error inside generated pattern for the third transmit channel. 1 Insert error inside generated pattern for the third transmit channel.
11 pg_insert_err1	Pattern Generator Insert Error One This bit enables or disables error insertion inside the generated pattern for channel 1. 0 Do not insert error inside generated pattern for the second transmit channel. 1 Insert error inside generated pattern for the second transmit channel.
10 pg_insert_err0	Pattern Generator Insert Error Zero This bit enables or disables error insertion inside the generated pattern for channel 0.

Table continues on the next page...

HDMI_PHY_PATTERNGEN field descriptions (continued)

Field	Description
	0 Do not insert error inside generated pattern for the first transmit channel. 1 Insert error inside generated pattern for the first transmit channel.
pg_pat[9:0]	Pattern Generator Generated Pattern This bus carries the generated pattern from the Pattern Generator module.

34.7.29 The Soft-Reset and DAC Enable, Clock Alignment and PG Mode (HDMI_PHY_SDCAP_MODE)

Register name: -

Access type: Read/write

Address: 0x1C

Value at reset: 0x0000

Address: 0h base + 1Ch offset = 1Ch

Bit	15	14	13	12	11	10	9	8
Read Write	Reserved				pg_mode2[2:0]			pg_ mode1[2:0]
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	pg_mode1[2:0]		pg_mode0[2:0]			tx_ck_align_ mode	adc_enb	soft_reset
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_SDCAP_MODE field descriptions

Field	Description
15–12 -	This field is reserved. Reserved
11–9 pg_mode2[2:0]	Pattern Generator Mode 2 This bus is used to select the mode of the Pattern Generator module for channel 2. 000 Disable the Pattern Generator module. 001 Enable the Pattern Generator module and generate a sequence of patterns for the third channel using LFSR 15 equation (for example, x15 + x14 + 1). 010 Enable the Pattern Generator module and generate a sequence of patterns for the third channel using LFSR 7 equation (for example, x7 + x6 + 1). 011 Enable the Pattern Generator module and generate a sequence of patterns for the third channel with the fixed word, which is saved inside the pg_pat[9:0] field of the PATTERNGEN register. 100 Enable the Pattern Generator module and generate a sequence of patterns for the third channel with DC-balanced word (for example, 0's or 1's and their inverted values).

Table continues on the next page...

HDMI_PHY_SDCAP_MODE field descriptions (continued)

Field	Description
	101 Enable the Pattern Generator module and generate a sequence of patterns for the third channel with fixed pattern using word of 0's word, 1's word, random word, the saved pattern inside the pg_pat[9:0] field of the PATTERNGEN register, and the inverted pg_pat[9:0] values (for example, 9 bits of 0's, 9 bits of 1's, 3FF, pg_pat[9:0], ~ pg_pat[9:0]). 110 Reserved 111 Reserved
8–6 pg_mode1[2:0]	Pattern Generator Mode 1 This bus is used to select the mode of the Pattern Generator module for channel 1. 000 Disable the Pattern Generator module. 001 Enable the Pattern Generator module and generate a sequence of patterns for the second channel using LFSR 15 equation (for example, x15 + x14 + 1). 010 second the Pattern Generator module and generate a sequence of patterns for the first channel using LFSR 7 equation (for example, x7 + x6 + 1). 011 Enable the Pattern Generator module and generate a sequence of patterns for the second channel with the fixed word, which is saved inside the pg_pat[9:0] field of the PATTERNGEN register. 100 Enable the Pattern Generator module and generate a sequence of patterns for the second channel with DC-balanced word (for example, 0's or 1's and their inverted values). 101 Enable the Pattern Generator module and generate a sequence of patterns for the second channel with fixed pattern using word of 0's word, 1's word, random word, the saved pattern inside the pg_pat[9:0] field of the PATTERNGEN register, and the inverted pg_pat[9:0] values (for example, 9 bits of 0's, 9 bits of 1's, 3FF, pg_pat[9:0], ~ pg_pat[9:0]). 110 Reserved 111 Reserved
5–3 pg_mode0[2:0]	Pattern Generator Mode 0 This bus is used to select the mode of the Pattern Generator module for channel 0. 000 Disable the Pattern Generator module. 001 Enable the Pattern Generator module and generate a sequence of patterns for the first channel using LFSR 15 equation (for example, x15 + x14 + 1). 010 Enable the Pattern Generator module and generate a sequence of patterns for the first channel using LFSR 7 equation (for example, x7 + x6 + 1). 011 Enable the Pattern Generator module and generate a sequence of patterns for the first channel with the fixed word, which is saved inside the pg_pat[9:0] field of the PATTERNGEN register. 100 Enable the Pattern Generator module and generate a sequence of patterns for the first channel with DC-balanced word (for example, 0's or 1's and their inverted values). 101 Enable the Pattern Generator module and generate a sequence of patterns for the first channel with fixed pattern using word of 0's word, 1's word, random word, the saved pattern inside the pg_pat[9:0] field of the PATTERNGEN register, and the inverted pg_pat[9:0] values (for example, 9 bits of 0's, 9 bits of 1's, 3FF, pg_pat[9:0], ~ pg_pat[9:0]). 110 Reserved 111 Reserved
2 tx_ck_align_mode	Transmission Clock Alignment Mode This bit selects the Transmission Clock Alignment mode. 0 Align the three lanes based on lane 1. 1 Align each of the three lanes separately in the following order: lane 2, lane 0, then lane 1.
1 adc_enb	Analog-to-Digital Converter Enable This bit enables or disables the analog-to-digital converter.

Table continues on the next page...

HDMI_PHY_SDCAP_MODE field descriptions (continued)

Field	Description
	0 Disable the analog-to-digital converter. 1 Enable the analog-to-digital converter, which is used in the Resistance Calibration module.
0 soft_reset	Soft Reset This bit enables or disables the soft-reset feature. 0 Do not perform a soft reset. 1 Perform a soft reset by resetting all the system FSMs except the I^2C and Control Register modules.

34.7.30 Scope Mode register (HDMI_PHY_SCOPEMODE)

Register name: SCOPEMODE

Access type: Read/write

Address: 0x1D

Value at reset: 0x0000

Address: 0h base + 1Dh offset = 1Dh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	Reserved		ck_scope_enb	scope_enb2	scope_enb1	scope_enb0	scope_sample_cnt[9:0]									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_SCOPEMODE field descriptions

Field	Description
15–14 -	This field is reserved. Reserved
13 ck_scope_enb	Clock Scope Enable This bit enables or disables the tracing of 1's on the clock channel. 0 Disable the tracing of 1's on the clock. 1 Enable the tracing of 1's on the clock.
12 scope_enb2	Scope Enable 2 This bit enables or disables the tracing of 1's on channel 2. 0 Disable the tracing of 1's on the third channel. 1 Enable the tracing of 1's on the third channel.
11 scope_enb1	Scope Enable 1 This bit enables or disables the tracing of 1's on channel 1.

Table continues on the next page...

HDMI_PHY_SCOPEMODE field descriptions (continued)

Field	Description
	0 Disable the tracing of 1's on the second channel. 1 Enable the tracing of 1's on the second channel.
10 scope_enb0	Scope Enable 0 This bit enables or disables the tracing of 1's on channel 0. 0 Disable the tracing of 1's on the first channel. 1 Enable the tracing of 1's on the first channel.
scope_sample_cnt[9:0]	Scope Sample Counter Indicates the number of samples that will be counted (should be multiple of the LFSR length). This count includes only the LSB bits; if the LFSR15 was used, you must program the new bits under the 0x24 register.

34.7.31 Digital Transmission Mode (HDMI_PHY_DIGTXMODE)

Register name: DIGTXMODE

Access type: Read/write

Address: 0x1E

Value at reset: 0x0000

dtb_select[6:0]

The dtb_select[6:0] encodings for the bit pairs are as follows:

Power Sequence

- 0x00: dtb[1] = pll_pwr_on and dtb[0] = pll_rst
- 0x01: dtb[1] = pll_pwr_on and dtb[0] = pll_gear_shift
- 0x02: dtb[1] = pll_pwr_on and dtb[0] = pll_drv_ana
- 0x03: dtb[1] = pll_pwr_on and dtb[0] = cko_word_enb
- 0x04: dtb[1] = pll_pwr_on and dtb[0] = tx_ready
- 0x05: dtb[1] = mpll_pwr_on and dtb[0] = mpll_rst
- 0x06: dtb[1] = mpll_pwr_on and dtb[0] = mpll_gear_shift
- 0x07: dtb[1] = mpll_pwr_on and dtb[0] = mpll_drv_ana
- 0x08: dtb[1] = mpll_pwr_on and dtb[0] = cko_word_enb
- 0x09: dtb[1] = mpll_pwr_on and dtb[0] = tx_ready
- 0x0A: dtb[1] = mpll_pwr_on and dtb[0] = tx_ser_div_en1
- 0x0B: dtb[1] = mpll_pwr_on and dtb[0] = tx_ser_clk_en1
- 0x0C: dtb[1] = ck_ref_enb and dtb[0] = refclk_enb
- 0x0D: dtb[1] = cko_word_enb and dtb[0] = refclk_enb

Transmission Clock Alignment

- 0x0E: Reserved
- 0x0F: dtb[1] = tx_ck_align_enb and dtb[0] = tx_ser_clk_kill0
- 0x10: dtb[1] = tx_ck_align_enb and dtb[0] = tx_ser_clk_kill1
- 0x11: dtb[1] = tx_ck_align_enb and dtb[0] = tx_ser_clk_kill2
- 0x12: dtb[1] = tx_ck_align_enb and dtb[0] = tx_ck_align_done

Resistance Calibration and Analog-to-Digital Converter

- 0x13: Reserved
- 0x14: dtb[1] = ck_rescal[6] and dtb[0] = ck_rescal[5]
- 0x15: dtb[1] = ck_rescal[4] and dtb[0] = ck_rescal[3]
- 0x16: dtb[1] = ck_rescal[2] and dtb[0] = ck_rescal[1]
- 0x17: dtb[1] = ck_rescal[0] and dtb[0] = sup_comp_mode
- 0x18: dtb[1] = sup_comp_rt_result and dtb[0] = sup_comp_rt_r
- 0x19: dtb[1] = sup_dac_n[7] and dtb[0] = sup_dac_n[6]
- 0x1A: dtb[1] = sup_dac_n[5] and dtb[0] = sup_dac_n[4]
- 0x1B: dtb[1] = sup_dac_n[3] and dtb[0] = sup_dac_n[2]
- 0x1C: dtb[1] = sup_dac_n[1] and dtb[0] = sup_dac_n[0]
- 0x1D: dtb[1] = sup_dac_th_n[2] and dtb[0] = sup_dac_th_n[1]
- 0x1E: dtb[1] = sup_dac_th_n[0] and dtb[0] = rescal_rep[6]
- 0x1F: dtb[1] = rescal_rep[5] and dtb[0] = rescal_rep[4]
- 0x20: dtb[1] = rescal_rep[3] and dtb[0] = rescal_rep[2]
- 0x21: dtb[1] = rescal_rep[1] and dtb[0] = rescal_rep[0]
- 0x22: dtb[1] = rcal_enb and dtb[0] = rcal_adc_done
- 0x23: dtb[1] = sup_comp_rt_pwron and dtb[0] = rcal_adc_done
- 0x24: Reserved
- 0x25: Reserved
- 0x26: Reserved
- 0x27: Reserved

TMDS Data Pattern

- 0x28: dtb[1] = tmds_data2[9] and dtb[0] = tmds_data2[8]
- 0x29: dtb[1] = tmds_data2[8] and dtb[0] = tmds_data2[7]
- 0x2A: dtb[1] = tmds_data2[7] and dtb[0] = tmds_data2[6]
- 0x2B: dtb[1] = tmds_data2[6] and dtb[0] = tmds_data2[5]
- 0x2C: dtb[1] = tmds_data2[5] and dtb[0] = tmds_data2[4]
- 0x2D: dtb[1] = tmds_data2[4] and dtb[0] = tmds_data2[3]
- 0x2E: dtb[1] = tmds_data2[3] and dtb[0] = tmds_data2[2]
- 0x2F: dtb[1] = tmds_data2[2] and dtb[0] = tmds_data2[1]
- 0x30: dtb[1] = tmds_data2[1] and dtb[0] = tmds_data2[0]
- 0x31: dtb[1] = tmds_data2[0] and dtb[0] = tmds_data1[9]

- 0x32: dtb[1] = tmds_data1[9] and dtb[0] = tmds_data1[8]
- 0x33: dtb[1] = tmds_data1[8] and dtb[0] = tmds_data1[7]
- 0x34: dtb[1] = tmds_data1[7] and dtb[0] = tmds_data1[6]
- 0x35: dtb[1] = tmds_data1[6] and dtb[0] = tmds_data1[5]
- 0x36: dtb[1] = tmds_data1[5] and dtb[0] = tmds_data1[4]
- 0x37: dtb[1] = tmds_data1[4] and dtb[0] = tmds_data1[3]
- 0x38: dtb[1] = tmds_data1[3] and dtb[0] = tmds_data1[2]
- 0x39: dtb[1] = tmds_data1[2] and dtb[0] = tmds_data1[1]
- 0x3A: dtb[1] = tmds_data1[1] and dtb[0] = tmds_data1[0]
- 0x3B: dtb[1] = tmds_data1[0] and dtb[0] = tmds_data0[9]
- 0x3C: dtb[1] = tmds_data0[9] and dtb[0] = tmds_data0[8]
- 0x3D: dtb[1] = tmds_data0[8] and dtb[0] = tmds_data0[7]
- 0x3E: dtb[1] = tmds_data0[7] and dtb[0] = tmds_data0[6]
- 0x3F: dtb[1] = tmds_data0[6] and dtb[0] = tmds_data0[5]
- 0x40: dtb[1] = tmds_data0[5] and dtb[0] = tmds_data0[4]
- 0x41: dtb[1] = tmds_data0[4] and dtb[0] = tmds_data0[3]
- 0x42: dtb[1] = tmds_data0[3] and dtb[0] = tmds_data0[2]
- 0x43: dtb[1] = tmds_data0[2] and dtb[0] = tmds_data0[1]
- 0x44: dtb[1] = tmds_data0[1] and dtb[0] = tmds_data0[0]
- 0x45: dtb[1] = tmds_data0[0] and dtb[0] = tx_ready

Pattern Generator

- 0x46: dtb[1] = tmds_data2[0] and dtb[0] = pg_dtb2
- 0x47: dtb[1] = tmds_data1[0] and dtb[0] = pg_dtb1
- 0x48: dtb[1] = tmds_data0[0] and dtb[0] = pg_dtb0
- 0x49: dtb[1] = pg_dtb2 and dtb[0] = pg_dtb0
- 0x4A: dtb[1] = pg_dtb1 and dtb[0] = pg_dtb0
- 0x4B: Reserved
- 0x4C: Reserved
- 0x4D: Reserved
- 0x4E: Reserved
- 0x4F: Reserved

Scope

- 0x50: dtb[1] = scope_enb2 and dtb[0] = scope_done2
- 0x51: dtb[1] = scope_enb1 and dtb[0] = scope_done1
- 0x52: dtb[1] = scope_enb0 and dtb[0] = scope_done0
- 0x53: dtb[1] = ck_scope_enb and dtb[0] = ck_scope_done
- 0x54: dtb[1] = scope_enb2 and dtb[0] = tx_scope_out2
- 0x55: dtb[1] = scope_enb1 and dtb[0] = tx_scope_out1
- 0x56: dtb[1] = scope_enb0 and dtb[0] = tx_scope_out0

- 0x57: Reserved
- 0x58: Reserved
- 0x59: Reserved

I2C

- 0x5A: dtb[1] = start_cond and dtb[0] = cregs_write (Not stored in the control register.)
- 0x5B: dtb[1] = sda_pull_dn_n and dtb[0] = cregs_ack (Not stored in the control register.)
- 0x5C: dtb[1] = cregs_read and dtb[0] = cregs_rd_data[0] (Not stored in the control register.)
- 0x5D: Reserved
- 0x5E: Reserved
- 0x5F: Reserved
- 0x60: Reserved
- 0x61: Reserved
- 0x62: Reserved
- 0x63: Reserved

Rx Sense

- 0x64: dtb[1] = rx_sense of clock driver and dtb[0] = rx_sense of CH2's driver
- 0x65: dtb[1] = rx_sense of CH1's driver and dtb[0] = rx_sense of CH0's driver
- 0x66: -

Address: 0h base + 1Eh offset = 1Eh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved			dtb_select[6:0]						invert_data2	invert_data1	invert_data0	cko_word_div_enb	skip_bit2	skip_bit1	skip_bit0
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_DIGTXMODE field descriptions

Field	Description
15–13 -	This field is reserved. Reserved
12–7 dtb_select[6:0]	Debug Test Bus Select This field determines the pair of bits placed on the dtb[1:0] bus. These selected pairs of bits come from the control register. The values that appear on dtb[1:0] are the current values actually stored in the control register (not the override values) with the exception of I^2C values, which are not stored in the control register.

Table continues on the next page...

HDMI_PHY_DIGTXMODE field descriptions (continued)

Field	Description
	For information about the bit pairs, see dtb_select[6:0] .
6 invert_data2	Inverter Data 2 This bit enables or disables the inverting feature for the transmitted pattern on channel 2. 0 Disable the inverting feature on the third channel. 1 Enable the inverting feature on the third channel.
5 invert_data1	Inverter Data 1 This bit enables or disables the inverting feature for the transmitted pattern on channel 1. 0 Disable the inverting feature on the second channel. 1 Enable the inverting feature on the second channel.
4 invert_data0	Inverter Data 0 This bit enables or disables the inverting feature for the transmitted pattern on channel 0. 0 Disable the inverting feature on the first channel. 1 Enable the inverting feature on the first channel.
3 cko_word_div_enb	Clock Output Word Divider Enable This bit enables or disables the output clock word divider. 0 Disable the output clock divider. 1 Enable the output clock divider.
2 skip_bit2	Skip Bit 2 This bit enables or disables skipping of the ninth bit of the transmitted pattern on channel 2. 0 Disable the skipping feature on the third channel. 1 Enable the skipping feature on the third channel.
1 skip_bit1	Skip Bit 1 This bit enables or disables skipping of the ninth bit of the transmitted pattern on channel 1. 0 Disable the skipping feature on the second channel. 1 Enable the skipping feature on the second channel.
0 skip_bit0	Skip Bit 0 This bit enables or disables skipping of the ninth bit of the transmitted pattern on channel 0. 0 Disable the skipping feature on the first channel. 1 Enable the skipping feature on the first channel.

34.7.32 Scope, Transmission Clock Alignment, and Resistance Calibration Set-on-Done Status (HDMI_PHY_STR_STATUS)

Register name: -

Access type: Read/write/asynchronous set-on-done

Address: 0x1F

Value at reset: 0x0000

Address: 0h base + 1Fh offset = 1Fh

Bit	15	14	13	12	11	10	9	8
Read Write	Reserved							
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read Write	Reserved		rcal_adc_done	tx_ck_align_done	ck_scope_done	scope_done2	scope_done1	scope_done0
Reset	0	0	0	0	0	0	0	0

HDMI_PHY_STR_STATUS field descriptions

Field	Description
15–6 -	This field is reserved. Reserved
5 rcal_adc_done	Resistance Calibration Analog-to-Digital Converter Done This bit indicates the status of completing the resistance calibration FSM. 0 The resistance calibration FSM is not complete. 1 The resistance calibration FSM is complete.
4 tx_ck_align_done	Transmission Clock Alignment Done This bit indicates the status of completing the transmission clock alignment FSM. 0 The transmission clock alignment FSM is not complete. 1 The transmission clock alignment FSM is complete.
3 ck_scope_done	Clock Scope Done This bit indicates the status of tracing of 1's on the clock channel. 0 The tracing process on the clock channel is not complete. 1 The tracing process on the clock channel is complete.
2 scope_done2	Scope Done 2 This bit indicates the status of tracing of 1's on channel 2. 0 The tracing process on channel 2 is not complete. 1 The tracing process on channel 2 is complete.
1 scope_done1	Scope Done 1 This bit indicates the status of tracing of 1's on channel 1. 0 The tracing process on channel 1 is not complete. 1 The tracing process on channel 1 is complete.
0 scope_done0	Scope Done 0 This bit indicates the status of tracing of 1's on channel 0. 0 The tracing process on channel 0 is not complete. 1 The tracing process on channel 0 is complete.

34.7.33 Scope Counter on Channel 0 (HDMI_PHY_SCOPECNT0)

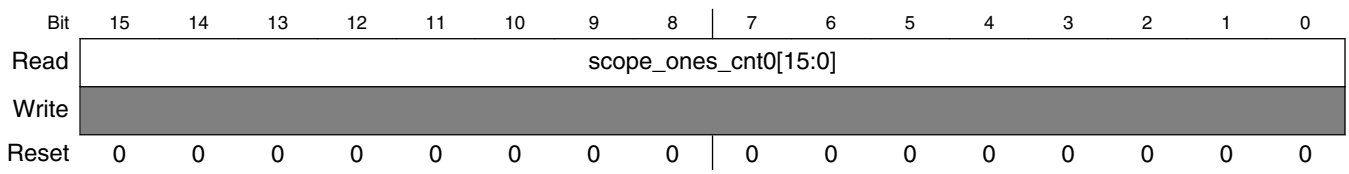
Register name: SCOPECNT0

Access type: Read-only

Address: 0x20

Value at reset: 0x0000

Address: 0h base + 20h offset = 20h



HDMI_PHY_SCOPECNT0 field descriptions

Field	Description
scope_ones_ cnt0[15:0]	Scope 1's Counter 0 This register carries the number of counted 1's on channel 0. If the LFSR15 was used to generate the scope patterns, you must read the MSB bits under 0x25 register.

34.7.34 Scope Counter on Channel 1 (HDMI_PHY_SCOPECNT1)

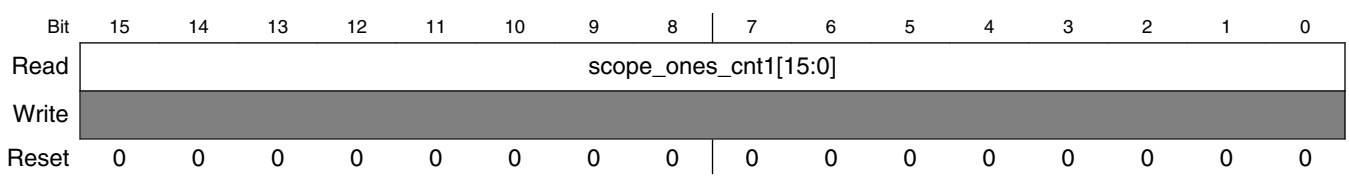
Register name: SCOPECNT1

Access type: Read-only

Address: 0x21

Value at reset: 0x0000

Address: 0h base + 21h offset = 21h



HDMI_PHY_SCOPECNT1 field descriptions

Field	Description
scope_ones_ cnt1[15:0]	Scope 1's Counter 1

HDMI_PHY_SCOPECNT1 field descriptions (continued)

Field	Description
	This register carries the number of counted 1's on channel 1. If the LFSR15 was used to generate the scope patterns, you must read the MSB bits under 0x25 register.

34.7.35 Scope Counter on Channel 2 (HDMI_PHY_SCOPECNT2)

Register name: SCOPECNT2

Access type: Read-only

Address: 0x22

Value at reset: 0x0000

Address: 0h base + 22h offset = 22h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	scope_ones_cnt2[15:0]															
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_SCOPECNT2 field descriptions

Field	Description
scope_ones_ cnt2[15:0]	Scope 1's Counter 2 This register carries the number of counted 1's on channel 2. If the LFSR15 was used to generate the scope patterns, you must read the MSB bits under 0x26 register.

34.7.36 Scope Counter on Clock Channel (HDMI_PHY_SCOPECNTCLK)

Register name: SCOPECNTCK

Access type: Read-only

Address: 0x23

Value at reset: 0x0000

Address: 0h base + 23h offset = 23h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	ck_scope_ones_cnt[15:0]															
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_SCOPECNTCLK field descriptions

Field	Description
ck_scope_ones_cnt[15:0]	Clock Scope 1's Counter This register carries the number of counted 1's on the clock channel. If the LFSR15 was used to generate the scope patterns, you must read the MSB bits under 0x26 register.

34.7.37 Scope Sample Count MSB, Scope Sample Repetition (HDMI_PHY_SCOPESAMPLE)

Register name: SCOPESAMPLE

Access type: Read/write

Address: 0x24

Value at reset: 0x13c0

Address: 0h base + 24h offset = 24h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved			scope_sample_rep[6:0]					scope_sample_cnt[15:9]							
Write																
Reset	0	0	0	1	0	0	1	1	1	1	0	0	0	0	0	0

HDMI_PHY_SCOPESAMPLE field descriptions

Field	Description
15–13 -	This field is reserved. Reserved
12–6 scope_sample_rep[6:0]	Scope Sample Repetition Number of repetitions made by the scope FSM. The total samples captured is scope_sample_rep x scope_sample_cnt.
scope_sample_cnt[15:9]	Scope Sample Counter Indicates the number of samples that will be counted (should be multiple of the LFSR length). These samples are the MSB bits only.

34.7.38 Scope Counter MSB Channel 0 and Channel 1 (HDMI_PHY_SCOPECNTMSB01)

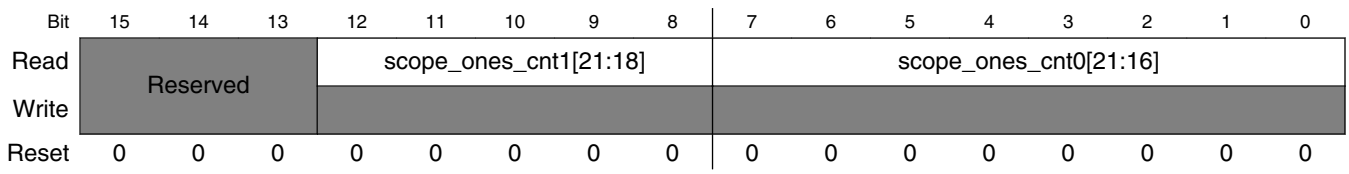
Register name: SCOPECNTMSB01

Access type: Read-only

Address: 0x25

Value at reset: 0x0000

Address: 0h base + 25h offset = 25h



HDMI_PHY_SCOPECNTMSB01 field descriptions

Field	Description
15–13 -	This field is reserved. Reserved
12–8 scope_ones_cnt1[21:18]	Scope 1's Counter 1 This register carries the number of counted 1's on channel 1. These 1's are the MSB bits only.
scope_ones_cnt0[21:16]	Scope 1's Counter 0 This register carries the number of counted 1's on channel 0. These 1's are the MSB bits only.

34.7.39 Scope Counter MSB Channel 2 and Clock Channel (HDMI_PHY_SCOPECNTMSB2CK)

Register name: SCOPECNTMSB2CK

Access type: Read-only

Address: 0x26

Value at reset: 0x0000

Address: 0h base + 26h offset = 26h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved			ck_scope_ones_cnt[21:16]					scope_ones_cnt2[21:16]							
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

HDMI_PHY_SCOPECNTMSB2CK field descriptions

Field	Description
15–13 - Reserved	This field is reserved.
12–6 ck_scope_ones_cnt[21:16]	Clock Scope 1's Counter This register carries the number of counted 1's on the clock channel. These 1's are the MSB bits only.
scope_ones_cnt2[21:16]	Scope 1's Counter 2 This register carries the number of counted 1's on channel 2. These 1's are the MSB bits only.

34.8 Appendix A: Driver Voltage Level Configuration

This appendix describes the driver voltage level configuration.

This configuration depends on the source termination value, the driver pre-emphasis settings, and the target signal voltage level swing.

A correct configuration must be set to meet both eye diagram mask and the specified high and low signal voltage levels.

To correctly configure the driver voltage level, the following parameters and signals (represented through their symbol) must be taken into consideration:

- VPH_{RXTERM} = 3.3 V -> 3.3-V supply rail connected to HDMI PHY sink termination resistors
- RXTERM = 50 Ω -> HDMI PHY sink termination resistors

Control Register	Signal	Symbol
0x19	d_tx_temp[2:0]	RTERM
0x0E	sup_tx_lvl[4.:0]	TXLVL
0x0E	sup_ck_lvl[4.:0]	CKLVL
0x09	tx_symon	SYMON
0x09	tx_traon	TRAON
0x09	tx_trbon	TRBON

The following table defines the pre-emphasis factor (PREEMPH) to be used in the information that follows.

SYMON	TRAON	TRBON	PREEMPH
0	0	0	-
1	0	0	0.00
1	0	1	0.08
1	1	0	0.17
1	1	1	0.25

The following equations can be used to calculate-for a certain signal voltage swing (VSWING) and PHY configuration (PREEMPH, RTERM)-the signal's high and low voltage levels (VHI, VLO).

VHI = VLO + VSWING

TXLVL = CKLVL = [0.772 - (VPHRXTERM - VLO)] / 0.01405

For a certain termination value and pre-emphasis configuration (factor), users input only the VSWING value. With this data, users can obtain the respective VHI and VLO DC levels and the TXLVL and CKLVL configuration to apply to the PHY.

Lower TXLVL/CKLVL values result in higher signal amplitudes, while higher TXLVL/ CKLVL values result in lower signal amplitudes.

Values for VSWING, VHI, and VLO must be within HDMI 1.4a specification limits. The table below provides driver voltage level settings for some example scenarios.

Table 34-65. Example Driver Voltage Level Settings

RTERM	SYMON	TRAON	TRBON	VHI (V)	VLO (V)	VSWING (V)	TXLVL	TXLVL (BIN)	CKLVL	CKLVL (BIN)
100	1'b1	1'b0	1'b0	3.200	2.800	0.400	19	10011	19	10011
100	1'b1	1'b0	1'b0	3.175	2.675	0.500	10	01010	10	01010
100	1'b1	1'b0	1'b1	3.107	2.607	0.500	6	00110	6	00110
133	1'b1	1'b0	1'b0	3.225	2.825	0.400	21	10101	21	10101
133	1'b1	1'b0	1'b0	3.206	2.706	0.500	13	01101	13	01101
133	1'b1	1'b0	1'b1	3.143	2.643	0.500	8	01000	8	01000

34.9 Appendix B

This appendix describes the PLL/MPLL configurations that must be made for each video mode of operation.

The PLL/MPLL configurations are divided in two distinct sections. Power-Up Requirements describes how to configure the PLL/MPLL mode of operation, and PLL/MPLL Generic Configuration Settings describes all the settings required to configure the PLL/MPLL for each specific video mode.

34.9.1 Single or Two-PLL in Coherent or Non-Coherent Mode of Operation

Table 34-66. Single or Two-PLL in Coherent or Non-Coherent Mode of Operation

Mode of Operation	bypass_ppll Register address: 0013: Bit 11	cko_sel<1> Register address: 0017: Bit 2	cko_sel<0> Register address: 0017: Bit 1
Two-PLL Non-Coherent (default)	0	0	0
Two-PLL Coherent	0	1	1
Single-PLL Non-Coherent	1	1	0
Single-PLL Coherent	1	1	1

NOTE

Single-PLL Coherent and Non-Coherent modes can be set for only video formats with no pixel repetition and color depth equal to 8 bits.

34.9.2 PLL/MPLL Generic Configuration Settings

Each supported video format can be referred to by its input pixel clock frequency, pixel repetition, and color depth in bits.

The table beblow provides all the PLL/MPLL settings necessary to completely configure the blocks for any supported video format.

Table B-2 PLL/MPLL Generic Configuration Settings

Divider Settings

Table 34-67. PLL/MPLL Generic Configuration Settings

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			0000:Bit14	0000:Bit13	0000:Bit8	0000:Bit7	0000:Bit6	0000:Bit5	0000:Bit4	0000:Bit3	0000:Bit2	0000:Bit1	0000:Bit0	0010:Bit5	0010:Bit4	0010:Bit3	0010:Bit2	0010:Bit1	0010:Bit0	0015:Bit3	0015:Bit2	0010:Bit11	0010:Bit10	0010:Bit9	0010:Bit8	0010:Bit7	0010:Bit6	0015:Bit1	0015:Bit0
PixelClock(MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
13.5	2	8	0	0	1	1	1	1	0	0	0	1	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
13.5	2	10	0	1	1	1	1	1	1	0	0	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
13.5	2	12	1	0	1	1	1	1	1	0	0	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	0
13.5	2	16	1	1	1	0	1	1	0	0	1	1	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1
13.5	4	8	0	0	1	0	1	1	0	0	1	1	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1
13.5	4	10	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1
13.5	4	12	1	0	1	0	1	1	1	0	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	1
13.5	4	16	1	1	0	1	1	1	0	1	0	1	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	1	0
18	3	8	0	0	1	0	1	1	0	0	1	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1
18	3	16	1	1	0	1	1	1	0	1	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	1	0
24.175	1	8	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
24.175	1	10	0	1	1	1	1	1	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0

Table continues on the next page...

Table 34-67. PLL/MPLL Generic Configuration Settings (continued)

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			0006: Bit 14	0006: Bit 13	0006: Bit 8	0006: Bit 7	0006: Bit 6	0006: Bit 5	0006: Bit 4	0006: Bit 3	0006: Bit 2	0006: Bit 1	0006: Bit 0	0010: Bit 5	0010: Bit 4	0010: Bit 3	0010: Bit 2	0010: Bit 1	0010: Bit 0	0015: Bit 3	0015: Bit 2	0010: Bit 11	0010: Bit 10	0010: Bit 9	0010: Bit 8	0010: Bit 7	0010: Bit 6	0015: Bit 1	0015: Bit 0
PixelClock (MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
24.175	1	12	1	0	1	1	1	1	0	0	0	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	0
24.175	1	16	1	1	1	0	1	0	0	0	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
27	1	8	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
27	1	10	0	1	1	1	1	1	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
27	1	12	1	0	1	1	1	1	0	0	0	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	0
27	1	16	1	1	1	0	1	0	0	0	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
27	2	8	0	0	1	0	1	0	0	0	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
27	2	10	0	1	1	0	1	1	0	0	1	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	1
27	2	12	1	0	1	0	1	1	0	0	1	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	1
27	2	16	1	1	0	1	1	0	0	0	1	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	1	0
27	4	8	0	0	0	1	1	0	0	0	1	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	1	0
27	4	10	0	1	0	1	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	1	0
27	4	12	1	0	0	1	1	1	0	1	0	1	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	1	0
27	4	16	1	1	0	0	1	0	0	1	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	1	1
36	1	8	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0
36	1	16	1	1	1	0	1	0	0	0	0	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
50, 35	1	8	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1

Table continues on the next page...

Table 34-67. PLL/MPLL Generic Configuration Settings (continued)

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			0006: Bit 14	0006: Bit 13	0006: Bit 8	0006: Bit 7	0006: Bit 6	0006: Bit 5	0006: Bit 4	0006: Bit 3	0006: Bit 2	0006: Bit 1	0006: Bit 0	0010: Bit 5	0010: Bit 4	0010: Bit 3	0010: Bit 2	0010: Bit 1	0010: Bit 0	0015: Bit 3	0015: Bit 2	0010: Bit 11	0010: Bit 10	0010: Bit 9	0010: Bit 8	0010: Bit 7	0010: Bit 6	0015: Bit 1	0015: Bit 0
PixelClock (MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
50, 35	1	10	0	1	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
50, 35	1	12	1	0	1	0	1	0	0	0	0	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
50, 35	1	16	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
50, 35	2	8	0	0	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
50, 35	2	10	0	1	0	1	1	0	0	0	1	0	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	1	0
50, 35	2	12	1	0	0	1	1	0	0	0	1	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	1	0
50, 35	2	16	1	1	0	0	0	1	0	0	1	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	1
54	1	8	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
54	1	10	0	1	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
54	1	12	1	0	1	0	1	0	0	0	0	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
54	1	16	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
54	2	8	0	0	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
54	2	10	0	1	0	1	1	0	0	0	1	0	1	0	1	1	1	0	0	0	1	1	0	0	1	0	0	1	0

Table continues on the next page...

Table 34-67. PLL/MPLL Generic Configuration Settings (continued)

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			0006: Bit 14	0006: Bit 13	0006: Bit 8	0006: Bit 7	0006: Bit 6	0006: Bit 5	0006: Bit 4	0006: Bit 3	0006: Bit 2	0006: Bit 1	0006: Bit 0	0010: Bit 5	0010: Bit 4	0010: Bit 3	0010: Bit 2	0010: Bit 1	0010: Bit 0	0015: Bit 3	0015: Bit 2	0010: Bit 11	0010: Bit 10	0010: Bit 9	0010: Bit 8	0010: Bit 7	0010: Bit 6	0015: Bit 1	0015: Bit 0
PixelClock (MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
54	2	12	1	0	0	1	1	0	0	0	1	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	1	0
54	2	16	1	1	0	0	0	1	0	0	1	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	1
58.4	1	8	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0	0	1	0	0	0	1
58.4	1	10	0	1	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
58.4	1	12	1	0	1	0	1	0	0	0	0	1	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
58.4	1	16	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
72	1	8	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
72	1	10	0	1	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
72	1	12	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
72	1	16	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
74.25	1	8	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	1	1	0	1	1	0	1
74.25	1	10	0	1	0	1	0	1	0	0	0	0	1	0	1	1	1	0	0	1	0	1	0	1	1	0	1	1	0
74.25	1	12	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
74.25	1	16	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0

Table continues on the next page...

Table 34-67. PLL/MPLL Generic Configuration Settings (continued)

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			00006:Bit14	00006:Bit13	00006:Bit8	00006:Bit7	00006:Bit6	00006:Bit5	00006:Bit4	00006:Bit3	00006:Bit2	00006:Bit1	00006:Bit0	00010:Bit5	00010:Bit4	00010:Bit3	00010:Bit2	00010:Bit1	00010:Bit0	00015:Bit3	00015:Bit2	00010:Bit11	00010:Bit10	00010:Bit9	00010:Bit8	00010:Bit7	00010:Bit6	00015:Bit1	00015:Bit0
PixelClock(MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
108	1	8	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
108	1	10	0	1	0	1	0	1	0	0	0	0	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
108	1	12	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
108	1	16	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	1
118,8	1	8	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
118,8	1	10	0	1	0	1	0	1	0	0	0	0	1	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0
118,8	1	12	1	0	0	1	0	1	0	0	0	1	0	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
118,8	1	16	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	1
144	1	8	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
144	1	10	0	1	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	1	1	0	1	1	1
144	1	12	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	1
144	1	16	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	1	1	0	1	1	0	1	1	1	1

Table continues on the next page...

Table 34-67. PLL/MPLL Generic Configuration Settings (continued)

			Divider Settings											PLL Charge Pump Settings								MPLL Charge Pump Settings							
			0000:Bit14	0000:Bit13	0000:Bit8	0000:Bit7	0000:Bit6	0000:Bit5	0000:Bit4	0000:Bit3	0000:Bit2	0000:Bit1	0000:Bit0	0010:Bit5	0010:Bit4	0010:Bit3	0010:Bit2	0010:Bit1	0010:Bit0	0015:Bit3	0015:Bit2	0010:Bit11	0010:Bit10	0010:Bit9	0010:Bit8	0010:Bit7	0010:Bit6	0015:Bit1	0015:Bit0
PixelClock(MHz)	PixelRepetition	ColorDepthinBits	prep_div<1>	prep_div<0>	mpll_n_cntrl<1>	mpll_n_cntrl<0>	pll_n_cntrl<1>	pll_n_cntrl<0>	pixel_rep<2>	pixel_rep<1>	pixel_rep<0>	clr_dpth<1>	clr_dpth<0>	pll_prop_cntrl<2>	pll_prop_cntrl<1>	pll_prop_cntrl<0>	pll_int_cntrl<2>	pll_int_cntrl<1>	pll_int_cntrl<0>	pll_gmp_cntrl<1>	pll_gmp_cntrl<0>	mpll_prop_cntrl<2>	mpll_prop_cntrl<1>	mpll_prop_cntrl<0>	mpll_int_cntrl<2>	mpll_int_cntrl<1>	mpll_int_cntrl<0>	mpll_gmp_cntrl<1>	mpll_gmp_cntrl<0>
148.5	1	8	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
148.5	1	10	0	1	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	1	1	0	1	1	1
148.5	1	12	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	1
148.5	1	16	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0	1	1	0	1	1	0	1	1	1	1
216	1	8	0	0	0	1	0	1	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	1	0	1	1	1	0
216	1	10	0	1	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	1	1	0	1	1	1
216	1	12	1	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	1	1	1	0	0	1	0	0	1	1

34.10 Appendix C: 3D Video Formats

3D video formats are created from any of the supported VICs. The 3D structure in which the frames are organized determines the way data is transferred.

The 3D video format is indicated using the VIC in the AVI InfoFrame (indicating one of the 2D formats defined in the CEA-861-D standard) in conjunction with the 3D_Structure field in the HDMI vendor-specific InfoFrame.

For Side-by-Side (Half) and Top-and-Bottom 3D structures, pixel clock frequency is equal to the original VIC pixel clock frequency.

For the L+depth+GFX+GFX-depth 3D structure, pixel clock frequency is equal to 4x the original VIC pixel clock frequency.

For other 3D structures, pixel clock frequency is equal to 2x the original VIC pixel clock frequency.

The following table summarizes the original 2D VIC pixel clock frequencies required to support the 3D structures.

Table 34-68. 2D VIC Pixel Clock Frequencies Required to Support 3D Structures

3D Structure	2D Pixel Clock Frequency
0000 (Frame Packing)	2x 2D pixel clock frequency
0001 (Field alternative)	2x 2D pixel clock frequency
0010 (Line alternative)	2x 2D pixel clock frequency
0011 (Side-by-Side (Full))	2x 2D pixel clock frequency
0100 (L + depth)	2x 2D pixel clock frequency
0101 (L + depth + GPX + GFX-depth)	4x 2D pixel clock frequency
0110 (Top-and-Bottom)	2D pixel clock frequency
1000 (Side-by-Side (Half))	2D pixel clock frequency

VIC-specific PLL/MPLL configurations apply to Side-by-Side (Half) and Top-and-Bottom 3D structures. To obtain the PLL/MPLL configuration for a specific VIC used within a 3D structure, see PLL/MPLL Generic Configuration Settings. Each configuration corresponds to a specific video format's pixel clock, pixel repetition, and color depth combination.

Example:

VIC 32: No pixel repetition, PCLK = 74.25 MHz, CD = 8,10,12,16

VIC 32 within 3D structure:

- Side-by-Side (Half) 3D structure (same pixel clock frequency): Same PLL/MPLL configuration as for VIC 32.
- Frame Packing (doubled pixel clock frequency): PLL/MPLL configuration corresponds to a row in Table 34-67 with the following: no pixel repetition, PCLK = 148.5 MHz, CD = 8,10,12,16.

For VICs where PCLK = 148.5 MHz, no color depth is supported for 3D video format, because the pixel clock frequency would be greater than 340 MHz.

Table 34-69 provides examples of primary 3D video format timings, and Table 34-70 provides examples of other 3D video format timings. Secondary 3D video format timings can be found in the HDMI specification.

Table 34-69. Primary 3D Video Format Timings

VIC	Video	Vertical Refresh Rate (Hz)	2D Pixel Clock (MHz)	3D Structure	3D Pixel Clock (MHz)
4	1,280x720p	59.94/60	74.25	0000 (Frame Packing)	148.50
4	1,280x720p	59.94/60	74.25	0110 (Top-and-Bottom)	74.25
4	1,280x720p	59.94/60	74.25	1000 (Side-by-Side (Half))	74.25
19	1,280x720p	50	74.25	0000 (Frame Packing)	148.50
19	1,280x720p	50	74.25	0110 (Top-and-Bottom)	74.25
19	1,280x720p	50	74.25	1000 (Side-by-Side (Half))	74.25
60	1,280x720p	23.97/24	58.40	0000 (Frame Packing)	118.8
62	1,280x720p	29.97/30	74.25	0000 (Frame Packing)	148.50
5	1,920x1,080i	59.94/60	74.25	0000 (Frame Packing)	148.50
5	1,920x1,080i	59.94/60	74.25	1000 (Side-by-Side (Half))	74.25
20	1,920x1,080i	50	74.25	0000 (Frame Packing)	148.50
20	1,920x1,080i	50	74.25	1000 (Side-by-Side (Half))	74.25
32	1,920x1,080p	23.98/24	74.25	0000 (Frame Packing)	148.50
32	1,920x1,080p	23.98/24	74.25	0110 (Top-and-Bottom)	74.25
32	1,920x1,080p	23.98/24	74.25	1000 (Side-by-Side (Half))	74.25
34	1,920x1,080p	29.98/30	74.25	0000 (Frame Packing)	148.50
34	1,920x1,080p	29.98/30	74.25	0110 (Top-and-Bottom)	74.25
16	1,920x1,080p	59.94/60	148.50	0110 (Top-and-Bottom)	148.50
31	1,920x1,080p	50	148.50	0110 (Top-and-Bottom)	148.50

Table 34-70. Examples of Other 3D Video Formats

VIC	Video	Vertical Refresh Rate (Hz)	2D Pixel Clock (MHz)	3D Structure	3D Pixel Clock (MHz)
5	1,920x1,080i	59.94/60	74.25	0001 (Field alternative)	148.5
20	1,920x1,080i	50	74.25	0001 (Field alternative)	148.5
16	1,920x1,080p	59.94/60	148.50	0010 (Line alternative)	297
31	1.920x1,080p	50	148.50	0010 (Line alternative)	297
16	1,920x1,080p	59.94/60	148.50	0011 (Side-by-Side (Full))	297
31	1,920x1,080p	50	148.50	0011 (Side-by-Side (Full))	297
19	1,280x720p	50	74.25	0100 (L + depth)	148.5
19	1,280x720p	50	74.25	0101 (L + depth + GFX + GFX-depth)	297

Chapter 35
I2C Controller (I2C)

35.1 Overview

This chapter describes block-level operation and programming of I2C. The chapter is intended for a block-driver software developer. To understand how the block is integrated at the SoC level, a system software developer should see discussions of the block in the appropriate SoC-level chapter(s).

References: This document assumes an understanding of the following document:

- *The I2C Bus Specification*, Version 2.1, by Philips Semiconductor

The Inter IC (I2C) provides functionality of a standard I2C slave and master. The I2C is designed to be compatible with the standard NXP I2C bus protocol.

NOTE

Three independent I2C channels are available.

I2C is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange, minimizing the interconnection between devices. This bus is suitable for applications requiring occasional communications over a short distance between many devices. The flexible I2C standard allows additional devices to be connected to the bus for expansion and system development. See the connection diagram in the figure below.

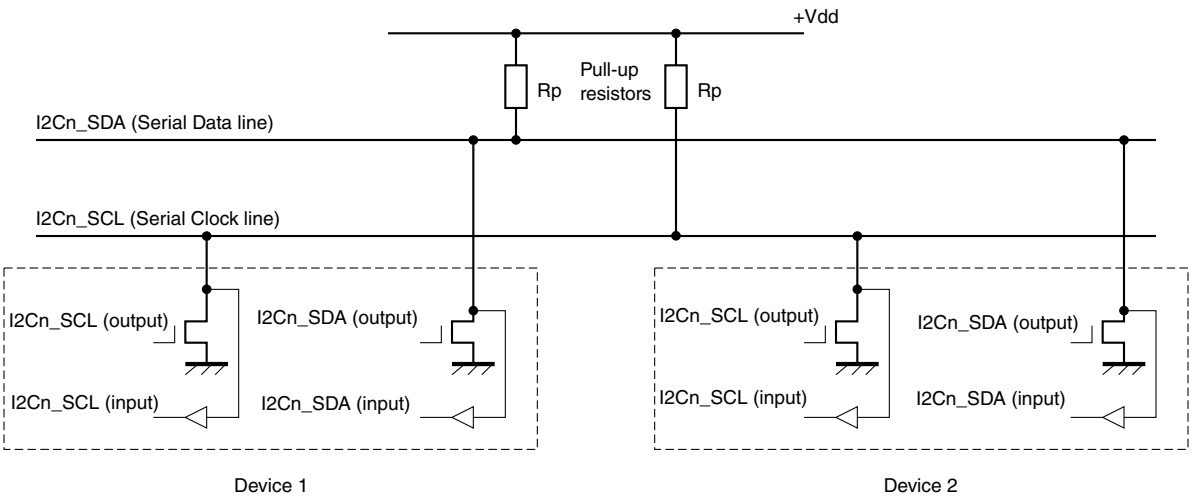


Figure 35-1. Connection of devices to I2C bus

The I2C interface speed is dependent on the I2C bus loading and timing characteristics. For pin requirement details, see *The I2C Bus Specification*. The I2C system is a true multimaster bus including arbitration and collision detection that prevents data corruption if multiple devices attempt to control the bus simultaneously. This feature supports complex applications with multiprocessor control and can be used for rapid testing and alignment of end products through external connections to an assembly-line computer. The figure below shows the block diagram of I2C.

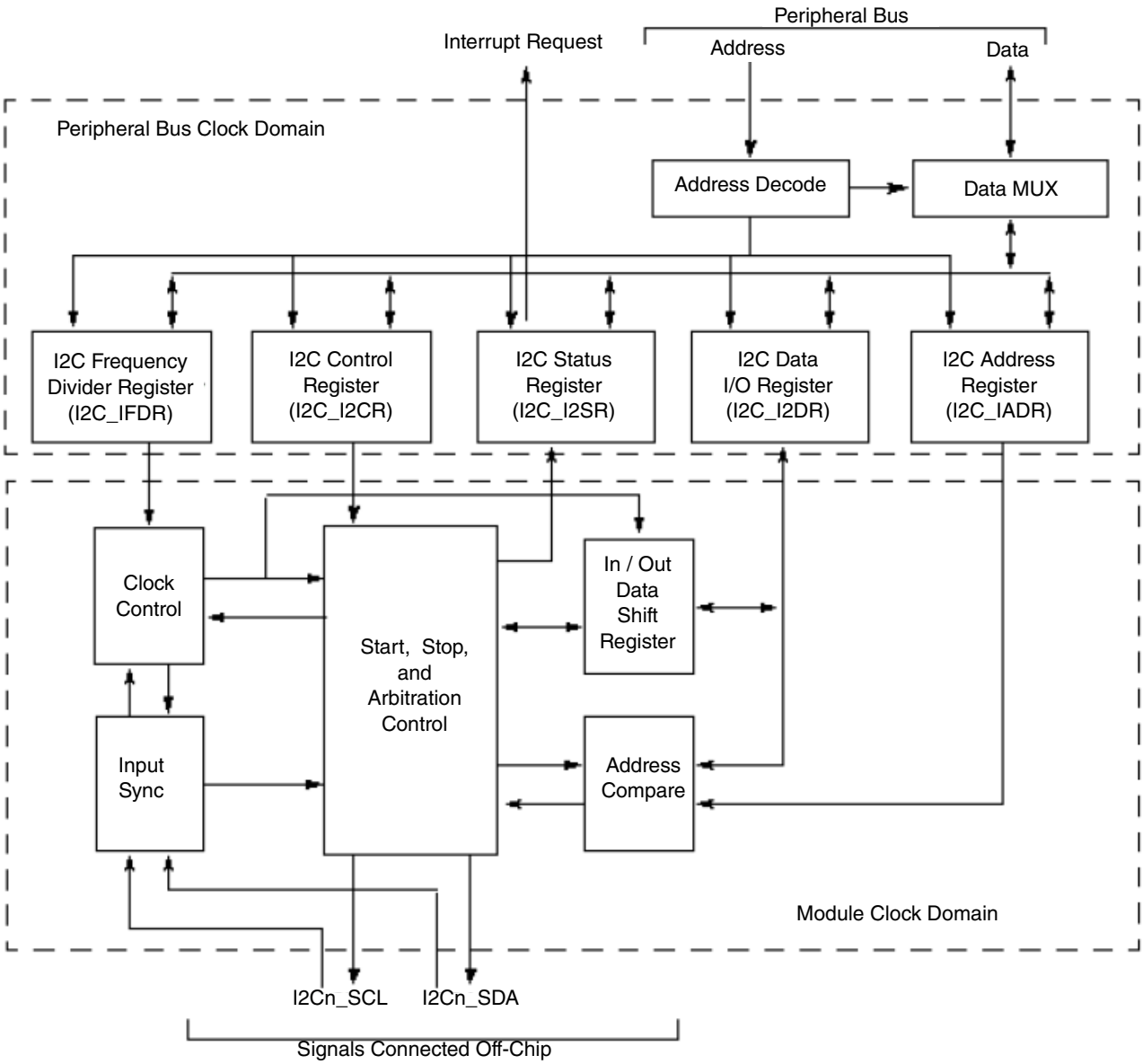


Figure 35-2. I2C block diagram

35.1.1 Features

The I2C has the following key features:

- Compatibility with I2C bus standard
- Multimaster operation
- Software programmability for one of 64 different serial clock frequencies
- Software-selectable acknowledge bit
- Interrupt-driven, byte-by-byte data transfer
- Arbitration-lost interrupt with automatic mode switching from master to slave

- Calling address identification interrupt
- Start and stop signal generation/detection
- Repeated Start signal generation
- Acknowledge bit generation/detection
- Bus-busy detection

35.1.2 Modes and operations

The I2C operates primarily in two functional modes: Standard mode and Fast mode.

- In Standard mode, I2C supports the data transfer rates up to 100 kbits/s.
- In Fast mode, data transfer rates up to 400 kbits/s can be achieved. Per block operation, there is no special configuration required for Fast or Standard mode. It is the data transfer rate that distinguishes Standard and Fast mode.

35.2 External Signals

This section discusses I2C signals that connect off-chip.

For I2C compliance, all devices connected to the I2Cn_SCL and I2Cn_SDA signals must have open-drain or open-collector outputs. The logic AND function is implemented on both lines with external pull-up resistors.

Inputs of I2Cn_SCL and I2Cn_SDA also need to be manually enabled by setting the SION bit in the IOMUX after the corresponding PADs are selected as I2C function.

The table below describes all I2C signals that connect off-chip.

Table 35-1. I2C External Signals

Signal	Description	Pad	Mode	Direction
I2C1_SCL (SCL)	Serial Clock	CSI0_DAT9	ALT4	IO
		EIM_D21	ALT6	
I2C1_SDA (SDA)	Serial Data	CSI0_DAT8	ALT4	IO
		EIM_D28	ALT1	
I2C2_SCL (SCL)	Serial Clock	EIM_EB2	ALT6	IO
		KEY_COL3	ALT4	
I2C2_SDA (SDA)	Serial Data	EIM_D16	ALT6	IO
		KEY_ROW3	ALT4	
I2C3_SCL (SCL)	Serial Clock	EIM_D17	ALT6	IO
		GPIO_3	ALT2	

Table continues on the next page...

Table 35-1. I2C External Signals (continued)

Signal	Description	Pad	Mode	Direction
		GPIO_5	ALT6	
I2C3_SDA (SDA)	Serial Data	EIM_D18	ALT6	IO
		GPIO_6	ALT2	
		GPIO_16	ALT6	

35.3 Clocks

There are two input clocks for I2C.

The following table describes the clock sources for I2C. Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 35-2. I2C Clocks

Clock name	Clock Root	Description
ipg_clk_patref	perclk_clk_root	Module clock
ipg_clk_s	ipg_clk_root	Peripheral access clock

- Peripheral clock (ipg_clk_s): This clock is used for peripheral bus register read/ writes.
- Module clock (ipg_clk_patref): This is the functional clock of the I2C. The serial bit clock frequency is derived from the module clock. The module clock and peripheral clocks are synchronous with each other. The minimum frequency of the module clock should be 12.8 MHz for Fast mode to achieve 400-kbps operation.

35.4 Functional description

This section provides a complete functional description of the block.

35.4.1 I2C system configuration

After a reset, the I2C defaults to Slave Receive operations. Thus, when not operating as a master or responding to a slave transmit address, the I2C defaults to the Slave Receive state.

For exceptions, see Initialization sequence.

NOTE

The I2C is designed to be compatible with the Philips™ I2C bus protocol. For information on system configuration, protocol, and restrictions, see the *I2C Bus Specification*, version 2.1, by Philips Semiconductors. The I2C supports Standard and Fast modes only.

35.4.2 Arbitration procedure

If multiple devices simultaneously request the bus, the bus clock is determined by a synchronization procedure in which the low period equals the longest clock-low period among the devices, and the high period equals the shortest. A data arbitration procedure determines the relative priority of competing devices.

A device loses arbitration if it sends logic high while another sends logic low; it immediately switches to Slave Receive mode and stops driving I2Cn_SDA. In this case, the transition from master to Slave mode does not generate a Stop condition. Meanwhile, hardware sets the arbitration lost bit in the I2C Status register (I2C_I2SR[IAL] to indicate loss of arbitration).

35.4.3 Clock synchronization

Because wire-AND logic is used, a high-to-low transition on SCL affects devices connected to the bus. Devices start counting their low period when the master drives SCL low. When a device clock goes low, it holds SCL low until the Clock High state is reached. However, the low-to-high change in this device clock may not change the state of SCL if another device clock is still in its low period. Therefore, the device with the longest low period holds the synchronized clock SCL low.

Devices with shorter low periods enter a High Wait state during this time (see Figure 35-3). When all devices involved have counted off their low periods, the synchronized clock SCL is released and pulled high. There is then no difference between device clocks and the state of SCL, so all of the devices start counting their high periods. The first device to complete its high period pulls SCL low again.

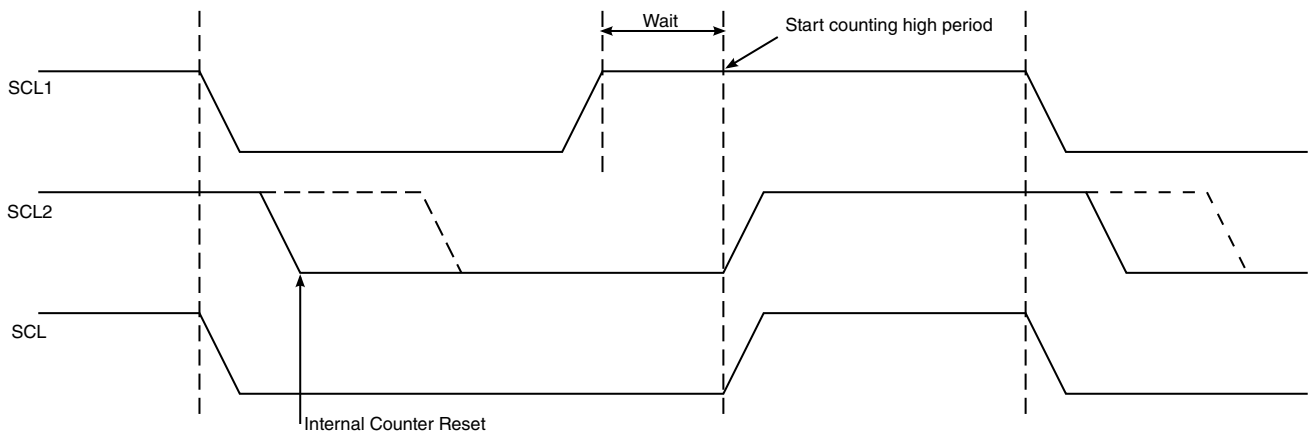


Figure 35-3. Synchronized clock SCL

35.4.4 Handshaking

The clock synchronization mechanism can be used as a handshake in data transfers. Slave devices can hold SCL low after completing one byte transfer (9 bits). In such a case, the clock mechanism halts the bus clock and forces the master clock into a Wait state until the slave releases SCL.

35.4.5 Clock stretching

Slaves can use the clock synchronization mechanism to slow down the transfer bit rate. After the master has driven SCL low, the slave can drive SCL low for the required period and then release it. If the slave SCL low period is longer than the master SCL low period, the resulting SCL bus signal low period is stretched.

35.4.6 Peripheral bus accesses

I2C is a 16-bit block. Only half-word accesses should be performed to the block.

35.4.7 Generation of transfer error on IP bus

If an address is received on the peripheral slave bus interface but it is not implemented, an access error is generated.

35.4.8 Reset

The I2C can be reset in the following ways:

- Global reset: A hard asynchronous reset of the whole I2C
- Software reset: An internal reset for the whole I2C (except for I2C_IADR and I2C_IFDR registers) initiated by deasserting the I2C_I2CR[IEN] bit

35.4.9 Interrupts

There is only one interrupt from the block, which is enabled by setting the I2C_I2CR[IIEN] bit.

The interrupt is generated in any one of the following conditions:

- One byte transfer is completed (the interrupt is set at the falling edge of the ninth clock).
- An address is received that matches its own specific address in Slave Receive mode.
- Arbitration is lost.

35.4.10 Byte order

The block only supports the Little-Endian mode.

35.5 Initialization

NOTE

Ensure the input select pins for IOMUXC are configured correctly for I2C.

35.5.1 Initialization sequence

Before the interface can transfer serial data, registers must be initialized, as listed here.

1. Set the data sampling rate (I2C_IFDR[IC] to obtain SCL frequency from the system bus clock.
2. Update the address in the (I2C_IADR) to define its slave address (address can range from 0 to 0x7f).
3. Set the I2C enable bit (I2C_I2CR[IEN]) to enable the I2C bus interface system.
4. Modify the bits in the I2C_I2CR to select Master/Slave mode, Transmit/Receive mode, and Interrupt-Enable or not.

35.5.2 Generation of Start

After completion of the initialization procedure, serial data can be transmitted by selecting the Master Transmit mode. On a multimaster bus system, the busy bus (I2C_I2SR[IBB]) must be tested to determine whether the serial bus is free. If the bus is free (IBB = 0), the Start signal and the first byte (the slave address) can be sent. The data written to the data register comprises the address of the desired slave and the LSB indicates the transfer direction.

The free time between a Stop and the next Start condition is built into the hardware that generates the Start cycle. Depending on the relative frequencies of the system clock and the SCL period, it may be necessary to wait until the I2C is busy after writing the calling address to the data register (I2C_I2DR), before proceeding to load data into the data register (I2C_I2DR).

35.5.3 Post-transfer software response

Sending or receiving a byte sets the data transferring bit (I2C_I2SR[ICF]), which indicates one byte of communication is finished. Upon completion, the interrupt status (I2C_I2SR[IIF]) is also set. An external interrupt is generated if the interrupt enable (I2C_I2CR[IIEN]) is set. The software must first clear the interrupt status (I2C_I2SR[IIF]) in the interrupt routine.

See the flow chart in Figure 35-5.

The data transferring bit (I2C_I2SR[ICF]) is cleared either by reading from I2C_I2DR in Receive mode or by writing to this register in Transmit mode.

The software can service the I2C I/O in the main program by monitoring the interrupt status (I2C_I2SR[IIF]) if the interrupt enable is deasserted. In this case, the interrupt status should be polled in the data transferring bit (I2C_I2SR[ICF]) because the operation is different when arbitration is lost.

When an interrupt occurs at the end of the address cycle, the master is always in Transmit mode; that is, the address is sent. If Master Receive mode is required, then I2C_I2CR[MTX] should be toggled and a dummy read of the I2C_I2DR register must be executed to trigger receive data.

During Slave-mode address cycles (I2C_I2SR[IAAS] = 1), the slave read/write bit I2C_I2SR[SRW] is read to determine the direction of the next transfer. The transmit/receive bit (I2C_I2CR[MTX]) should also be programmed accordingly. For Slave-mode data cycles (IAAS = 0), SRW is invalid. MTX should be read to determine the current transfer direction.

35.5.4 Generation of Stop

A data transfer ends when the master signals a Stop, which can occur after all data is sent.

For a master receiver to terminate a data transfer, it must inform the slave transmitter by not acknowledging the last data byte. This is done by setting the transmit acknowledge bit (I2C_I2CR[TXAK]) before reading the next-to-last byte. Before the last byte is read, a Stop signal must be generated.

35.5.5 Generation of Repeated Start

After the data transfer, if the master still requires the bus, it can signal another Start followed by another slave address without signaling a Stop.

35.5.6 Slave mode

In the slave interrupt service routine (see Figure 35-5), the block addressed as slave bit (IAAS) should be tested to check if a calling of its own address has just been received. If IAAS is set, software should set the Transmit/Receive mode select bit (I2C_I2CR[MTX]) according to the I2C_I2SR[SRW]. Writing to the I2C_I2CR clears the IAAS automatically. The only time IAAS is read as set is from the interrupt at the end of the address cycle where an address match occurred; interrupts resulting from subsequent data transfers will have IAAS cleared. A data transfer can now be initiated by writing information to I2C_I2DR for slave transmits, or read from I2C_I2DR in Slave Receive mode. A dummy read of I2C_I2DR in Slave Receive mode releases SCL, allowing the master to send data.

In the slave transmitter routine, the receive acknowledge bit (I2C_I2SR[RXAK]) must be tested before sending the next byte of data. Setting RXAK means an end-of-data signal from the master receiver, after which the software must switch it from Transmit to Receiver mode. Reading the data register (I2C_I2DR) then releases SCL so the master can generate a Stop signal.

35.5.7 Arbitration lost

If several devices try to engage the bus at the same time, one becomes master. Hardware immediately switches devices that lose arbitration to Slave Receive mode. Data output to 12Cn_SDA stops, but 12Cn_SCL is still generated until the end of the byte during which arbitration is lost. An interrupt occurs at the falling edge of the ninth clock of this transfer if the arbitration is lost (I2C_I2SR[IAL] = 1), and the Slave mode is selected (I2C_I2CR[MSTA] = 0).

See the flow chart in Figure 35-5.

If a device that is not a master tries to transmit or do a Start, hardware inhibits the transmission, clears MSTA without signaling a Stop, generates an interrupt to the ARM platform, and sets I2C_I2SR[IAL] to indicate a failed attempt to engage the bus. When considering these cases, the slave service routine should first test I2C_I2SR[IAL], and the software should clear it if it is set.

For Multimaster mode, when an I2C is enabled when the bus is busy and asserts Start, the I2C_I2SR[IAL] bit gets set only for 12Cn_SDA=0, 12Cn_SCL=0/1, 12Cn_SDA=1, and 12Cn_SCL=0; but not for 12Cn_SDA=1 and I2Cn_SCL=1, which is the equivalent of Bus Idle state.

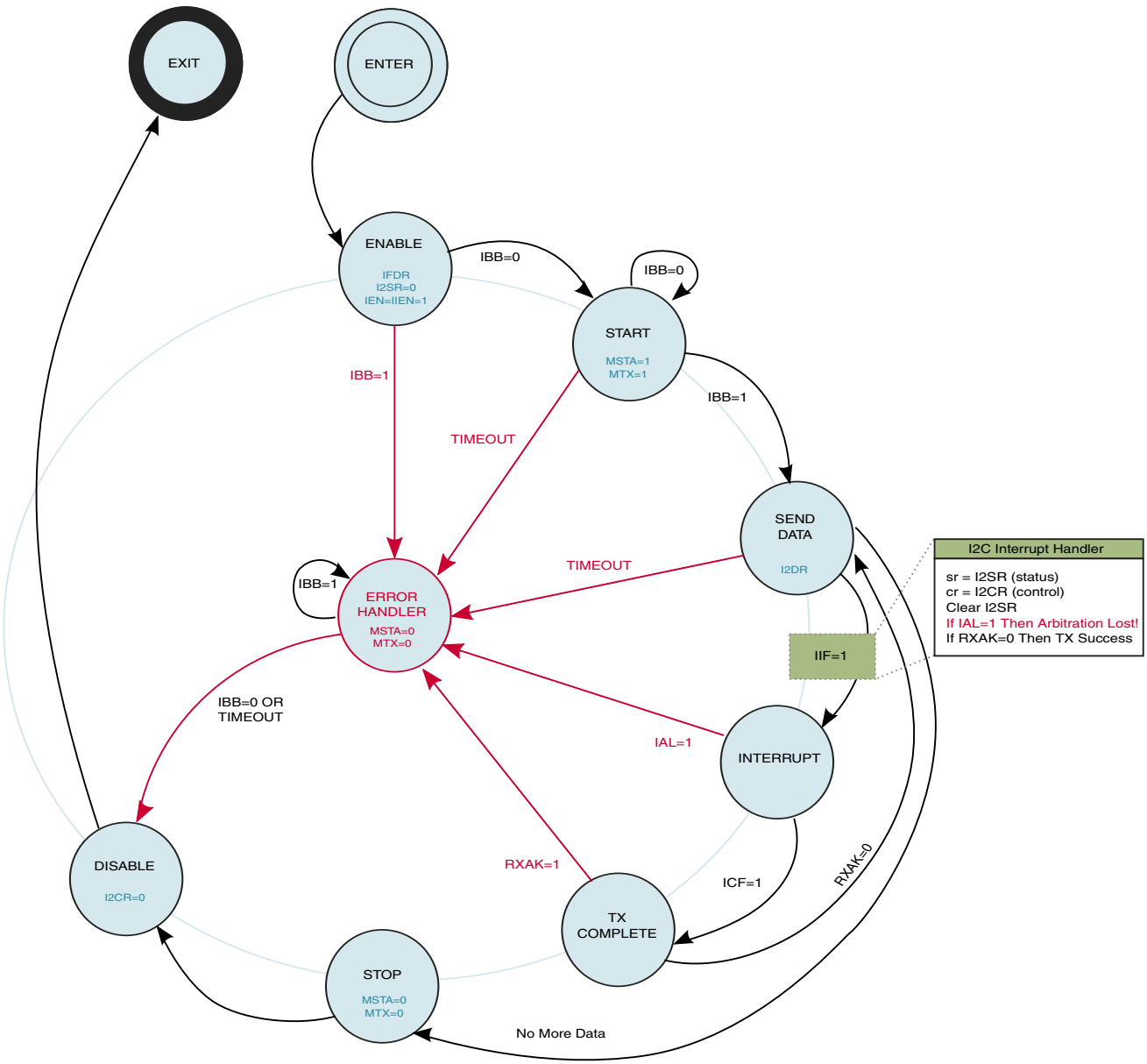


Figure 35-4. I2C Programming state diagram

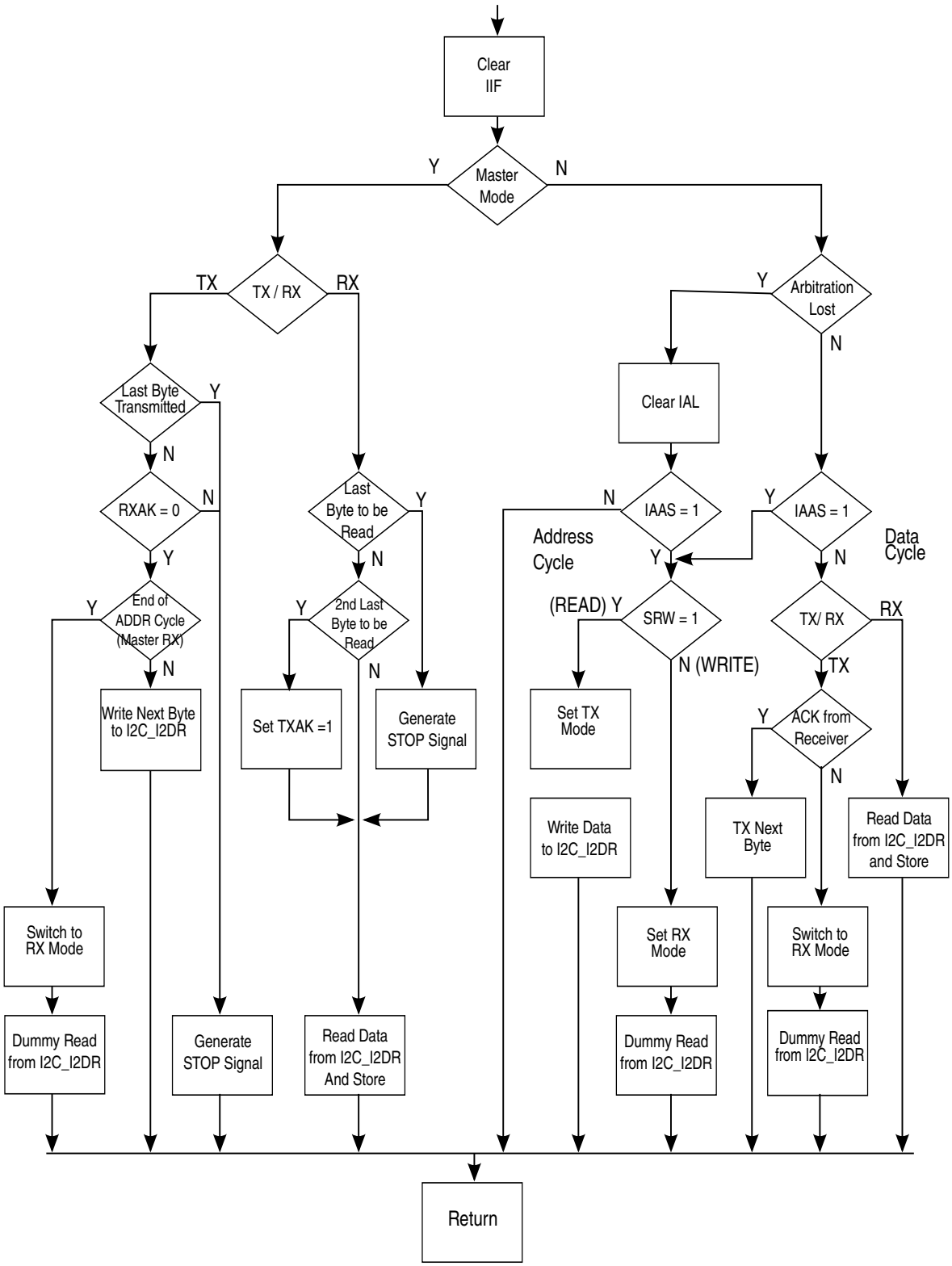


Figure 35-5. Flowchart of typical I2C interrupt routine

NOTE

For a Repeated Start only, the Stop-generation stage does not occur in Master mode. A loop repeats itself without stopping for the next start.

For Master Receive mode, I2C is programmed as Master Transmit during Address mode and after slave address transfer; the MTX bit should be cleared and a dummy read on the I2C_I2DR register should be performed so I2C can read the next receive data.

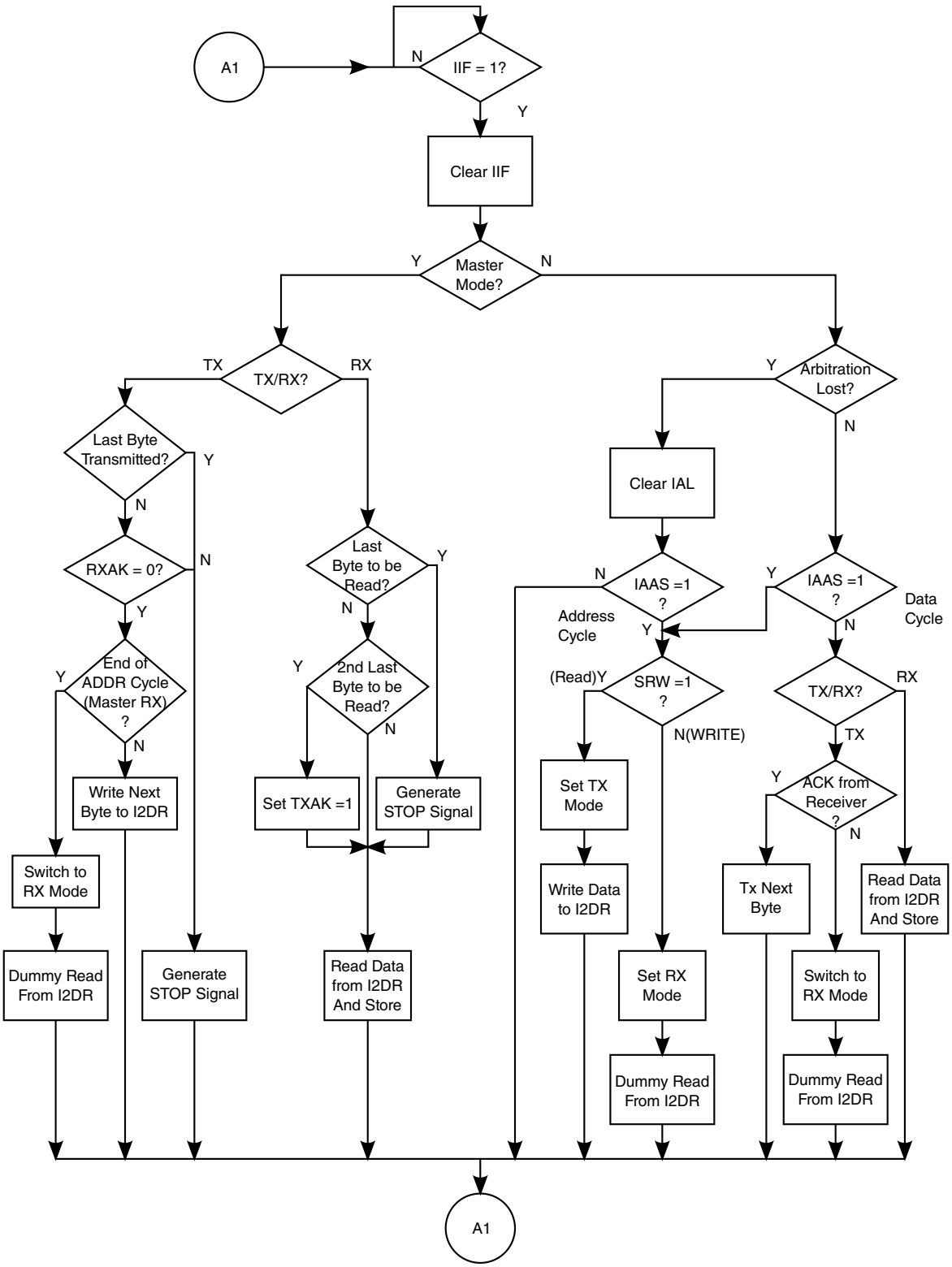


Figure 35-6. Flowchart for typical I2C polling routine

NOTE

The timeout value depends on the bus frequency at which I2C is operating. The minimum timeout for polling the IIF bit at a maximum I2C bus frequency of 400 kHz is T_{min} = 25 µs (=2.5 x 10 µs). This value can be calculated for any bus frequency. The formula is $T_{min} = 10/F_{SCL}$, where F_{SCL} is the frequency of the I2C clock (SCL).

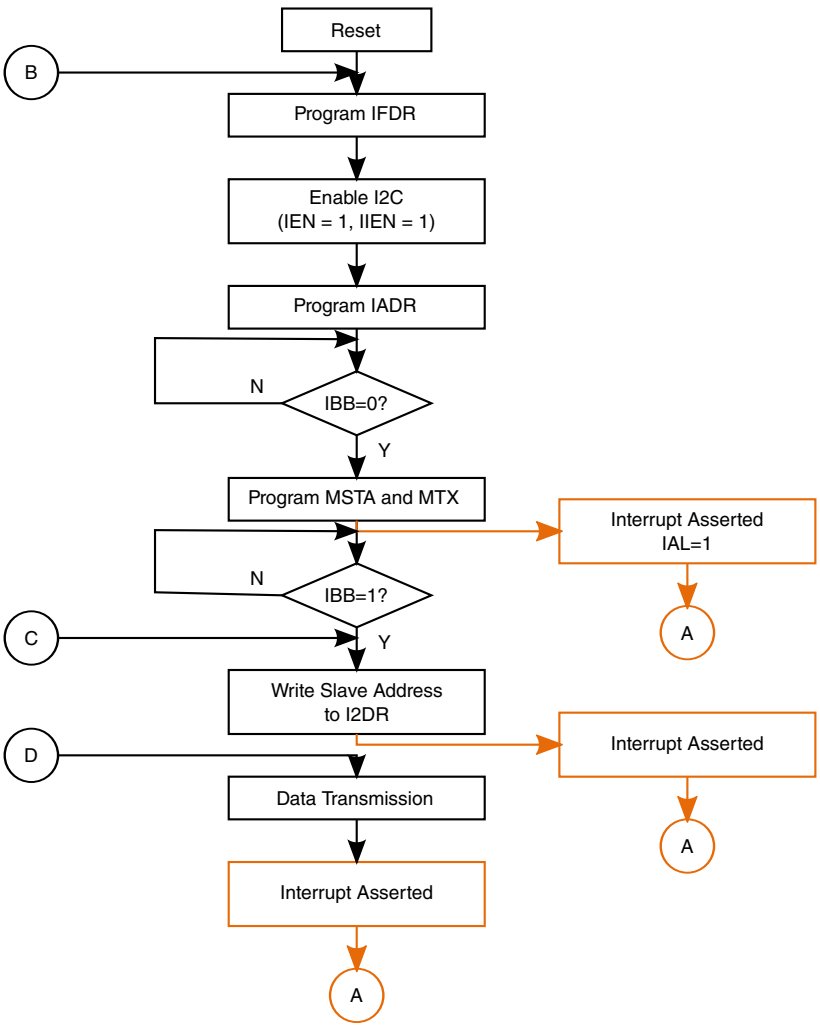


Figure 35-7. Detailed flowchart of a typical I2C Master Transmit mode, part 1

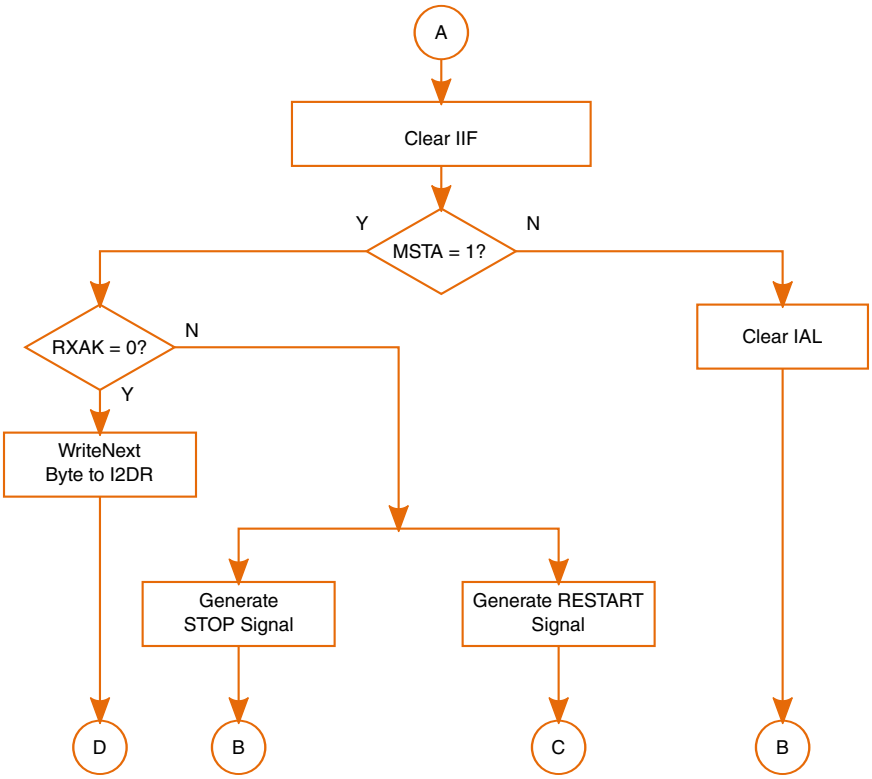


Figure 35-8. Detailed flowchart of a typical I2C Master Transmit mode, part 2

Figure 35-7 and Figure 35-8 show the Master Transmit mode operation with interrupt subroutine. If an interrupt is generated and the MSTA bit is 0, then bus arbitration is lost and IAL is set. Software can clear the IAL bit and reprogram I2C. If the MSTA bit is 1, then it is a transfer-generated interrupt. In this case, software can check the RXAK bit for a data receive acknowledgement by the slave and, accordingly, decide to do one of the following:

- Generate a STOP
- Generate a REPEATED START by writing to the I2C_I2CR register
- Perform the next data transfer by writing to the I2C_I2DR register

NOTE

The IBB bit is asserted by a Start condition on the bus, and it is deasserted by a Stop condition on the bus. Therefore, if arbitration is lost due to an unexpected Stop condition during transfer, then IBB is cleared. If arbitration is lost due to a data mismatch, then it is not cleared. Software should always clear the IEN bit and then set it if arbitration is lost.

35.6 Software restriction

Software should ensure that there is a delay of at least two module clock cycles after it sets the I2C_I2CR[RSTA] bit and before writing to the I2C_I2DR register. The maximum possible clock period of the module clock is 78 ns.

35.7 I2C Memory Map/Register Definition

The I2C contains five 16-bit registers.

NOTE

Registers at offsets 0x0002, 0x0006, 0x000A, and 0x000E are reserved for future additions.

I2C memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21A_0000	I2C Address Register (I2C1_IADR)	16	R/W	0000h	35.7.1/1911
21A_0004	I2C Frequency Divider Register (I2C1_IFDR)	16	R/W	0000h	35.7.2/1911
21A_0008	I2C Control Register (I2C1_I2CR)	16	R/W	0000h	35.7.3/1913
21A_000C	I2C Status Register (I2C1_I2SR)	16	R/W	0081h	35.7.4/1914
21A_0010	I2C Data I/O Register (I2C1_I2DR)	16	R/W	0000h	35.7.5/1916
21A_4000	I2C Address Register (I2C2_IADR)	16	R/W	0000h	35.7.1/1911
21A_4004	I2C Frequency Divider Register (I2C2_IFDR)	16	R/W	0000h	35.7.2/1911
21A_4008	I2C Control Register (I2C2_I2CR)	16	R/W	0000h	35.7.3/1913
21A_400C	I2C Status Register (I2C2_I2SR)	16	R/W	0081h	35.7.4/1914
21A_4010	I2C Data I/O Register (I2C2_I2DR)	16	R/W	0000h	35.7.5/1916
21A_8000	I2C Address Register (I2C3_IADR)	16	R/W	0000h	35.7.1/1911
21A_8004	I2C Frequency Divider Register (I2C3_IFDR)	16	R/W	0000h	35.7.2/1911
21A_8008	I2C Control Register (I2C3_I2CR)	16	R/W	0000h	35.7.3/1913
21A_800C	I2C Status Register (I2C3_I2SR)	16	R/W	0081h	35.7.4/1914
21A_8010	I2C Data I/O Register (I2C3_I2DR)	16	R/W	0000h	35.7.5/1916

35.7.1 I2C Address Register (I2Cx_IADR)

Address: Base address + 0h offset

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	0								ADR							0
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

I2Cx_IADR field descriptions

Field	Description
15–8 Reserved	This read-only field is reserved and always has the value 0.
7–1 ADR	Slave address. Contains the specific slave address to be used by the I2C. Slave mode is the default I2C mode for an address match on the bus. **NOTE:** The I2C_IADR holds the address to which the I2C responds when addressed as a slave. The slave address is not the address sent on the bus during the address transfer. The register is not reset by a software reset.
0 Reserved	This read-only field is reserved and always has the value 0.

35.7.2 I2C Frequency Divider Register (I2Cx_IFDR)

The I2C_IFDR provides a programmable prescaler to configure the clock for bit-rate selection. The register does not get reset by a software reset.

I2C clock is sourced from PERCLK_ROOT which is routed from IPG_CLK_ROOT. I2C clock frequency can easily obtained by using the following formula:

I2C clock Frequency = (PERCLK_ROOT frequency)/(division factor corresponding to IFDR)

By default, IPG_CLK_ROOT and PERCLK_ROOT frequencies are set to 49.5MHz, where the root clock is sourced from PLL2's PFD2. Obtaining the frequencies can be accomplished by:

PLL2 = 528MHz

PLL2_PFD2 = 528MHz * 18 / 24 = 396MHz

IPG_CLK_ROOT = (PLL2_PFD2 / ahb_podf)/ ipg_podf = (396MHz/4)/2 = 49.5MHz

PER_CLK_ROOT = IPG_CLK_ROOT/perclk_podf = 49.5MHz/1 = 49.5MHz

NOTE

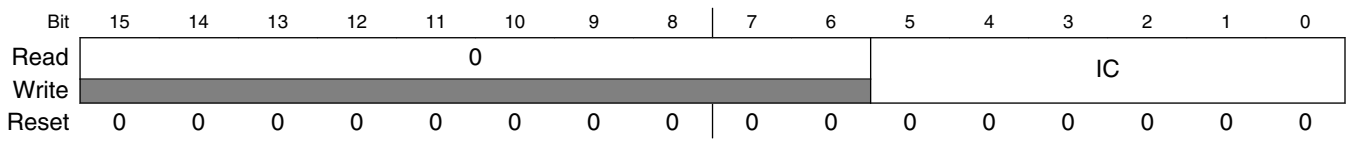
The above calculation assumes that the default CCM register settings, routing, and division factors are used. If different routing, PFD values, and/or division factors are used, the user must adjust the parameters accordingly to calculate the correct clock frequency.

The following table describes the divider and register values for the register field "IC."

Table 35-13. I2C_IFDR Register Field Values

IC	Divider		IC	Divider		IC	Divider		IC	Divider
0x00	30		0x10	288		0x20	22		0x30	160
0x01	32		0x11	320		0x21	24		0x31	192
0x02	36		0x12	384		0x22	26		0x32	224
0x03	42		0x13	480		0x23	28		0x33	256
0x04	48		0x14	576		0x24	32		0x34	320
0x05	52		0x15	640		0x25	36		0x35	384
0x06	60		0x16	768		0x26	40		0x36	448
0x07	72		0x17	960		0x27	44		0x37	512
0x08	80		0x18	1152		0x28	48		0x38	640
0x09	88		0x19	1280		0x29	56		0x39	768
0x0A	104		0x1A	1536		0x2A	64		0x3A	896
0x0B	128		0x1B	1920		0x2B	72		0x3B	1024
0x0C	144		0x1C	2304		0x2C	80		0x3C	1280
0x0D	160		0x1D	2560		0x2D	96		0x3D	1536
0x0E	192		0x1E	3072		0x2E	112		0x3E	1792
0x0F	240		0x1F	3840		0x2F	128		0x3F	2048

Address: Base address + 4h offset



I2Cx_IFDR field descriptions

Field	Description
15–6 Reserved	This read-only field is reserved and always has the value 0.
IC	I2C clock rate. Prescales the clock for bit-rate selection. Due to potentially slow I2Cn_SCL and I2Cn_SDA rise and fall times, bus signals are sampled at the prescaler frequency. The serial bit clock frequency may be lower than IPG_CLK_ROOT divided by the divider shown in the I2C Data I/O Register. **NOTE:** The IC value should not be changed during the data transfer, however, it can be changed before a Repeat Start or Start programming sequence in I2C. The I2C protocol supports bit rates of up to 400 kbps. The IC bits need to be programmed in accordance with this constraint.

35.7.3 I2C Control Register (I2Cx_I2CR)

The I2C_I2CR is used to enable the I2C and the I2C interrupt. It also contains bits that govern operation as a slave or a master.

Address: Base address + 8h offset

Bit	15	14	13	12	11	10	9	8
Read	0							
Write								
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read	IEN	IIEN	MSTA	MTX	TXAK	0	0	
Write						RSTA		
Reset	0	0	0	0	0	0	0	0

I2Cx_I2CR field descriptions

Field	Description
15–8 Reserved	This read-only field is reserved and always has the value 0.
7 IEN	I2C enable. Also controls the software reset of the entire I2C. Resetting the bit generates an internal reset to the block. If the block is enabled in the middle of a byte transfer, Slave mode ignores the current bus transfer and starts operating when the next Start condition is detected. Master mode is not aware that the bus is busy, so initiating a start cycle may corrupt the current bus cycle, ultimately causing either the current master or the I2C to lose arbitration. Subsequently, bus operation returns to normal. 0 The block is disabled, but registers can still be accessed. 1 The I2C is enabled. This bit must be set before any other I2C_I2CR bits have an effect.
6 IIEN	I2C interrupt enable. **NOTE:** If data is written during the Start condition, that is, just after setting the I2C_I2CR[MSTA] and I2C_I2CR[MTX] bits, then the ICF bit is cleared at the falling edge of SCLK after Start. If data is written after the Start condition and falling edge of SCLK, then the ICF bit is cleared as soon as data is written. 0 I2C interrupts are disabled, but the status flag I2C_I2SR[IIF] continues to be set when an Interrupt condition occurs. 1 I2C interrupts are enabled. An I2C interrupt occurs if I2C_I2SR[IIF] is also set.
5 MSTA	Master/Slave mode select bit. If the master loses arbitration, MSTA is cleared without generating a Stop signal. **NOTE:** The module clock should be on for writing to the MSTA bit. **NOTE:** The MSTA bit is cleared by software to generate a Stop condition; it can also be cleared by hardware when the I2C loses the bus arbitration.

Table continues on the next page...

I2Cx_I2CR field descriptions (continued)

Field	Description
	0 Slave mode. Changing MSTA from 1 to 0 generates a Stop and selects Slave mode. 1 Master mode. Changing MSTA from 0 to 1 signals a Start on the bus and selects Master mode.
4 MTX	Transmit/Receive mode select bit. Selects the direction of master and slave transfers. 0 Receive. When a slave is addressed, the software should set MTX according to the slave read/write bit in the I2C status register (I2C_I2SR[SRW]). 1 Transmit. In Master mode, MTX should be set according to the type of transfer required. Therefore, for address cycles, MTX is always 1.
3 TXAK	Transmit acknowledge enable. Specifies the value driven onto I2Cn_SDA during acknowledge cycles for both master and slave receivers. **NOTE:** Writing TXAK applies only when the I2C bus is a receiver. 0 An acknowledge signal is sent to the bus at the ninth clock bit after receiving one byte of data. 1 No acknowledge signal response is sent (that is, the acknowledge bit = 1).
2 RSTA	Repeat start. Always reads as 0. Attempting a repeat start without bus mastership causes loss of arbitration. 0 No repeat start 1 Generates a Repeated Start condition
Reserved	This read-only field is reserved and always has the value 0.

35.7.4 I2C Status Register (I2Cx_I2SR)

The I2C_I2SR contains bits that indicate transaction direction and status.

Address: Base address + Ch offset

Bit	15	14	13	12	11	10	9	8
Read	0							
Write								
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read	ICF	IAAS	IBB	IAL	0	SRW	IIF	RXAK
Write								
Reset	1	0	0	0	0	0	0	1

I2Cx_I2SR field descriptions

Field	Description
15–8 Reserved	This read-only field is reserved and always has the value 0.
7 ICF	Data transferring bit. While one byte of data is transferred, ICF is cleared. 0 Transfer is in progress. 1 Transfer is complete. This bit is set by the falling edge of the ninth clock of the last byte transfer.
6 IAAS	I2C addressed as a slave bit. The ARM platform is interrupted if the interrupt enable (I2C_I2CR[IIEN]) is set. The ARM platform must check the slave read/write bit (SRW) and set its Transfer/Receive mode accordingly. Writing to I2C_I2CR clears this bit. 0 Not addressed 1 Addressed as a slave. Set when its own address (I2C_IADR) matches the calling address.
5 IBB	I2C bus busy bit. Indicates the status of the bus. **NOTE:** When I2C is enabled (I2C_I2CR[IEN] = 1), it continuously polls the bus data (SDA) and clock (SCL) signals to determine a Start or Stop condition. 0 Bus is idle. If a Stop signal is detected, IBB is cleared. 1 Bus is busy. When Start is detected, IBB is set.
4 IAL	Arbitration lost. Set by hardware in the following circumstances (IAL must be cleared by software by writing a "0" to it at the start of the interrupt service routine): • I2Cn_SDA input samples low when the master drives high during an address or data-transmit cycle. • I2Cn_SDA input samples low when the master drives high during the acknowledge bit of a data-receive cycle. For the above two cases, the bit is set at the falling edge of the ninth I2Cn_SCL clock during the ACK cycle. • A Start cycle is attempted when the bus is busy. • A Repeated Start cycle is requested in Slave mode. • A Stop condition is detected when the master did not request it. **NOTE:** Software cannot set the bit. 0 No arbitration lost. 1 Arbitration is lost.
3 Reserved	This read-only field is reserved and always has the value 0.
2 SRW	Slave read/write. When the I2C is addressed as a slave, IAAS is set, and the slave read/write bit (SRW) indicates the value of the R/W command bit of the calling address sent from the master. SRW is valid only when a complete transfer has occurred, no other transfers have been initiated, and the I2C is a slave and has an address match. 0 Slave receive, master writing to slave 1 Slave transmit, master reading from slave
1 IIF	I2C interrupt. Must be cleared by the software by writing a "0" to it in the interrupt routine. **NOTE:** The software cannot set the bit. 0 No I2C interrupt pending. 1 An interrupt is pending.

Table continues on the next page...

I2Cx_I2SR field descriptions (continued)

Field	Description
	This causes a processor interrupt request (if the interrupt enable is asserted [IIEN = 1]). The interrupt is set when one of the following occurs: • One byte transfer is completed (the interrupt is set at the falling edge of the ninth clock). • An address is received that matches its own specific address in Slave Receive mode. • Arbitration is lost.
0 RXAK	Received acknowledge. This is the value received from the I2Cn_SDA input for the acknowledge bit during a bus cycle. 0 An "acknowledge" signal was received after the completion of an 8-bit data transmission on the bus. 1 A "No acknowledge" signal was detected at the ninth clock.

35.7.5 I2C Data I/O Register (I2Cx_I2DR)

In Master Receive mode, reading the data register allows a read to occur and initiates the next byte to be received. In Slave mode, the same function is available after it is addressed.

Address: Base address + 10h offset

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	0								DATA							
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

I2Cx_I2DR field descriptions

Field	Description
15–8 Reserved	This read-only field is reserved and always has the value 0.
DATA	Data Byte. Holds the last data byte received or the next data byte to be transferred. Software writes the next data byte to be transmitted or reads the data byte received. **NOTE:** The core-written value in I2C_I2DR cannot be read back by the core. Only data written by the I2C bus side can be read.

Chapter 36
IOMUX Controller (IOMUXC)

36.1 Overview

The IOMUX Controller (IOMUXC), together with the IOMUX, enables the IC to share one pad to several functional blocks. This sharing is done by multiplexing the pad's input and output signals.

Every module requires a specific pad setting (such as pull up or keeper), and for each pad, there are up to 8 muxing options (called ALT modes). The pad settings parameters are controlled by the IOMUXC.

The IOMUX consists only of combinatorial logic combined from several basic IOMUX cells. Each basic IOMUX cell handles only one pad signal's muxing.

Figure 36-1 illustrates the IOMUX/IOMUXC connectivity in the system.

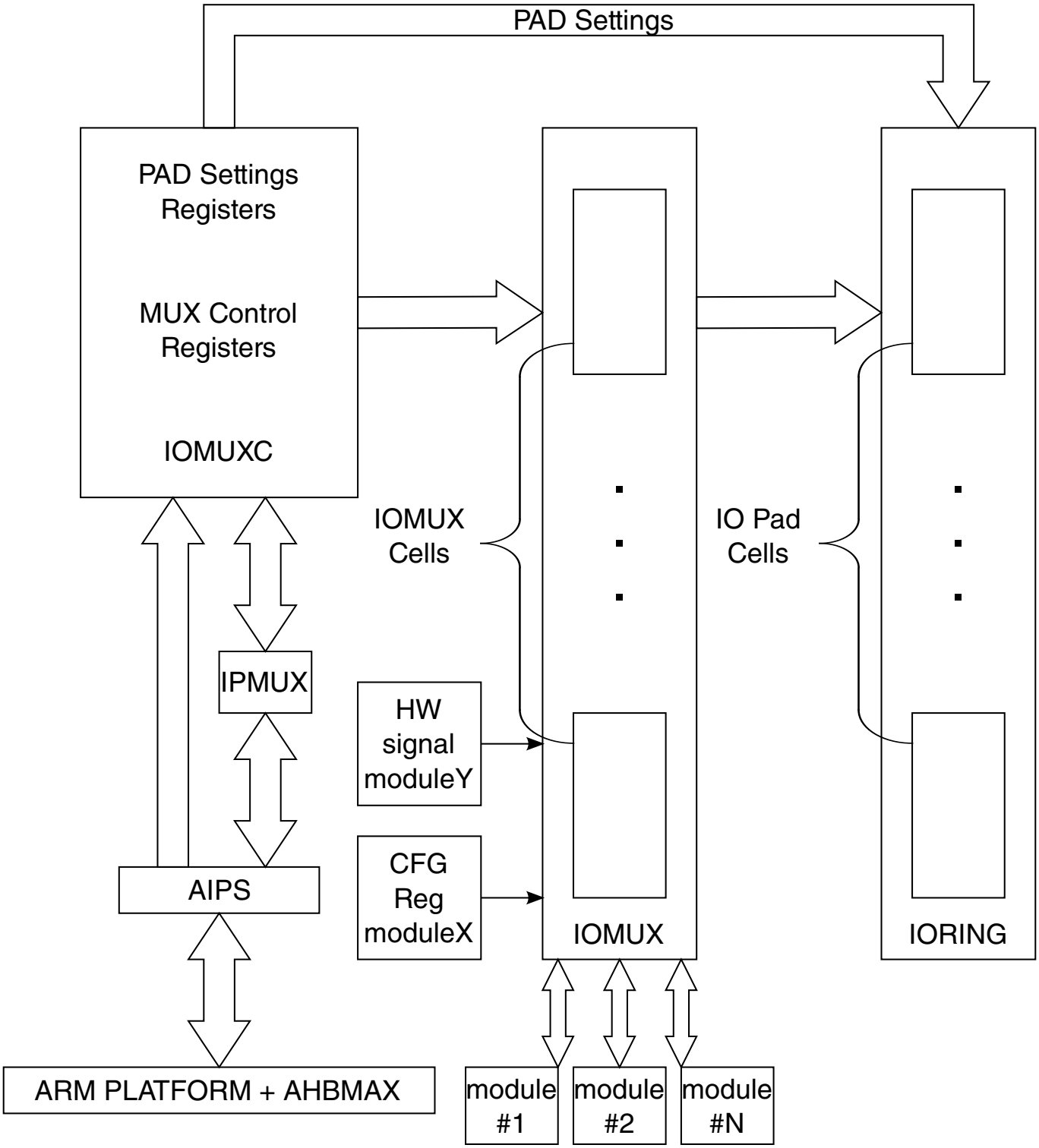


Figure 36-1. IOMUX SoC Level Block Diagram

36.1.1 Features

The IOMUXC features are:

- 32-bit software mux control registers (IOMUXC_SW_MUX_CTL_PAD_<PAD NAME> or IOMUXC_SW_MUX_CTL_GRP_<GROUP NAME>) to configure 1 of 8 alternate (ALT) MUX_MODE feilds of each pad or a predefined group of pads and to enable the forcing of an input path of the pad(s) (SION bit).
- 32-bit software pad control registers (IOMUXC_SW_PAD_CTL_PAD_<PAD_NAME> or IOMUXC_SW_PAD_CTL_GRP_<GROUP NAME>) to configure specific pad settings of each pad, or a predefined group of pads.
- 32-bit general purpose registers - 14 (GPR0 to GPR13) 32-bit registers according to SoC requirements for any usage.
- 32-bit input select control registers to control the input path to a module when more than one pad drives this module input.

Each SW MUX/PAD CTL IOMUXC register handles only one pad or one pad's group.

Only the minimum number of registers required by software are implemented by hardware. For example, if only ALT0 and ALT1 modes are used on Pad x then only one bit register will be generated as the MUX_MODE control field in the software mux control register of Pad x.

The software mux control registers may allow the forcing of pads to become input (input path enabled) regardless of the functional direction driven. This may be useful for loopback and GPIO data capture.

36.2 Clocks

The table found here describes the clock sources for IOMUXC.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 36-1. IOMUXC Clocks

Clock name	Clock Root	Description
ipt_clk_io	enfc_clk_root	IO clock
ipg_clk_s	ipg_clk_root	Peripheral access clock

36.3 Functional description

This section provides a complete functional description of the block.

The IOMUXC consists of two sub-blocks:

- IOMUXC_REGISTERS includes all of the IOMUXC registers (see Features).
- IOMUXC_LOGIC includes all of the IOMUXC combinatorial logic (IP interface controls, address decoder, observability muxes).

The IOMUX consists of a number (about the number of pads in the SoC) of basic iomux_cell units. If only one functional mode is required for a specific pad, there is no need for IOMUX and the signals can be connected directly from the module to the I/O. The IOMUX cell is required whenever two or more functional modes are required for a specific pad or when one functional mode and the one test mode are required.

The basic iomux_cell design, which allows two levels of HW signal control (in ALT6 and ALT7 modes - ALT7 gets highest priority) is shown in Figure 36-2.

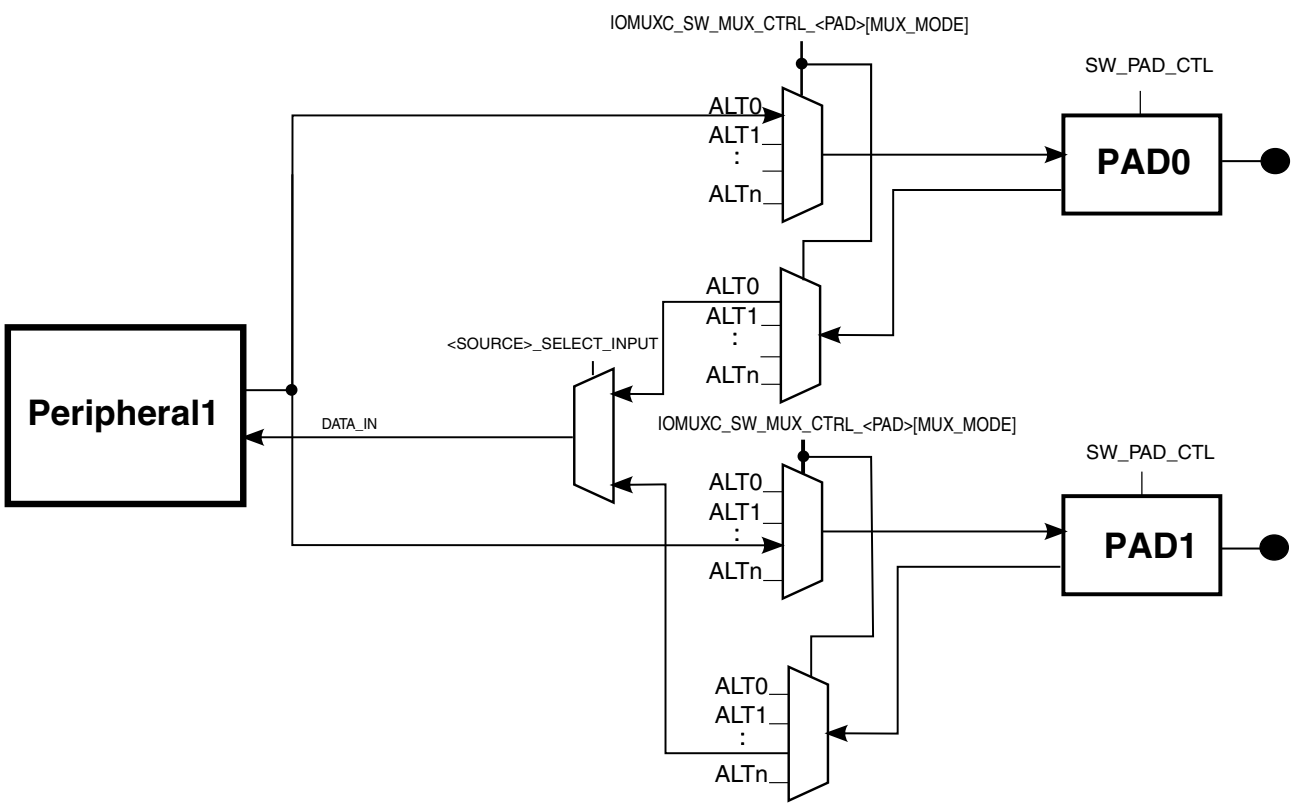


Figure 36-2. IOMUX Cell Block Diagram

36.3.1 ALT6 and ALT7 extended muxing modes

The ALT7 and ALT6 extended muxing modes allow any signal in the system (such as fuse, pad input, JTAG, or software register) to override any software configuration and to force the ALT6/ALT7 muxing mode.

It also allows an IOMUX software register to control a group of pads.

36.3.2 SW Loopback through SION bit

A limited option exists to override the default pad functionality and force the input path to be active (ipp_ibe==1'b1) regardless of the value driven by the corresponding module. This can be done by setting the SION (Software Input On) bit in the IOMUXC_SW_MUX_CTL register (when available) to "1".

Uses include:

- LoopBack - Module x drives the pad and also receives pad value as an input.
- GPIO Capture - Module x drives the pad and the value is captured by GPIO.

36.3.3 Daisy chain - multi pads driving same module input pin

In some cases, more than one pad may drive a single module input pin. Such cases require the addition of one more level of IOMUXing; all of these input signals are muxed, and a dedicated software controlled register controls the mux in order to select the required input path.

A block port involved in "daisy chain" requires two software configuration commands, one for selecting the mode for this pad (programable via the IOMUXC_SW_MUX_CTL_<PAD> registers) and one for defining it as the input path (via the daisy chain registers).

This means that a block port involved in "daisy chain" requires two software configuration commands, one for selecting the mode for this pad (programable via the IOMUXC_SW_MUX_CTL_<PAD> registers) and one for defining it as the input path (via the daisy chain registers). The daisy chain is illustrated in the figure below.

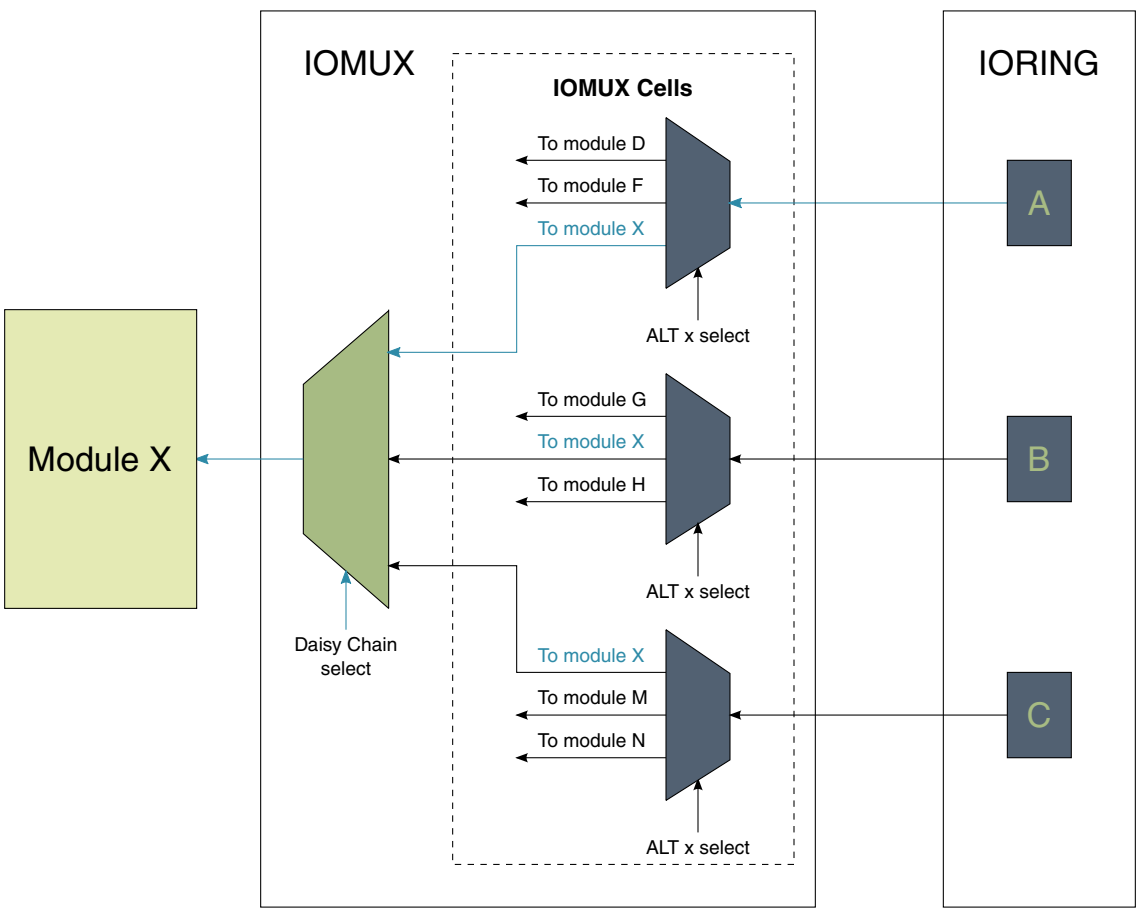


Figure 36-3. Daisy chain illustration

36.4 IOMUXC Memory Map/Register Definition

The main groups of IOMUXC registers are:

The General Purpose Registers IOMUXC_GPR[13:0] are used to select operating modes for general features in the SoC, usually not related to the IOMUX itself.

The Software MUX Control Registers are used to configure the IOMUX muxing, and "connect" the pad to a given port in a module.

The PAD Settings Registers are used to control the pad settings configuration. For some pads (in order to save chip route) the pad settings are grouped in one register; changing the group register will affect the settings for all pads in the group.

The following table shows the IOMUXC register summary.

IOMUXC memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0000	GPR (IOMUXC_GPR0)	32	R/W	0000_0000h	36.4.1/1950
20E_0004	GPR (IOMUXC_GPR1)	32	R/W	4840_0005h	36.4.2/1953
20E_0008	GPR (IOMUXC_GPR2)	32	R/W	0000_0000h	36.4.3/1956
20E_000C	GPR (IOMUXC_GPR3)	32	R/W	01E0_0000h	36.4.4/1958
20E_0010	GPR (IOMUXC_GPR4)	32	R/W	0000_0000h	36.4.5/1962
20E_0014	GPR (IOMUXC_GPR5)	32	R/W	0000_0000h	36.4.6/1965
20E_0018	GPR (IOMUXC_GPR6)	32	R/W	2222_2222h	36.4.7/1966
20E_001C	GPR (IOMUXC_GPR7)	32	R/W	2222_2222h	36.4.8/1967
20E_0020	GPR (IOMUXC_GPR8)	32	R/W	0000_0000h	36.4.9/1968
20E_0024	GPR (IOMUXC_GPR9)	32	R/W	0000_0000h	36.4.10/ 1969
20E_0028	GPR (IOMUXC_GPR10)	32	R/W	0000_3800h	36.4.11/ 1970
20E_002C	GPR (IOMUXC_GPR11)	32	R/W	0000_3800h	36.4.12/ 1972
20E_0030	GPR (IOMUXC_GPR12)	32	R/W	0F00_0000h	36.4.13/ 1972
20E_0034	GPR (IOMUXC_GPR13)	32	R/W	0591_24C4h	36.4.14/ 1974
20E_004C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1)	32	R/W	0000_0005h	36.4.15/ 1977
20E_0050	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2)	32	R/W	0000_0005h	36.4.16/ 1978
20E_0054	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0)	32	R/W	0000_0005h	36.4.17/ 1979

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0058	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC)	32	R/W	0000_0005h	36.4.18/ 1980
20E_005C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0)	32	R/W	0000_0005h	36.4.19/ 1981
20E_0060	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1)	32	R/W	0000_0005h	36.4.20/ 1982
20E_0064	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2)	32	R/W	0000_0005h	36.4.21/ 1983
20E_0068	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3)	32	R/W	0000_0005h	36.4.22/ 1984
20E_006C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL)	32	R/W	0000_0005h	36.4.23/ 1985
20E_0070	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0)	32	R/W	0000_0005h	36.4.24/ 1986
20E_0074	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL)	32	R/W	0000_0005h	36.4.25/ 1987
20E_0078	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1)	32	R/W	0000_0005h	36.4.26/ 1988
20E_007C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2)	32	R/W	0000_0005h	36.4.27/ 1989
20E_0080	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3)	32	R/W	0000_0005h	36.4.28/ 1990
20E_0084	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC)	32	R/W	0000_0005h	36.4.29/ 1991
20E_0088	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25)	32	R/W	0000_0000h	36.4.30/ 1992
20E_008C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B)	32	R/W	0000_0005h	36.4.31/ 1993
20E_0090	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16)	32	R/W	0000_0005h	36.4.32/ 1994
20E_0094	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17)	32	R/W	0000_0005h	36.4.33/ 1995
20E_0098	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18)	32	R/W	0000_0005h	36.4.34/ 1996
20E_009C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19)	32	R/W	0000_0005h	36.4.35/ 1997
20E_00A0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20)	32	R/W	0000_0005h	36.4.36/ 1998
20E_00A4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21)	32	R/W	0000_0005h	36.4.37/ 1999
20E_00A8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22)	32	R/W	0000_0005h	36.4.38/ 2000
20E_00AC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23)	32	R/W	0000_0005h	36.4.39/ 2001

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_00B0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B)	32	R/W	0000_0005h	36.4.40/ 2002
20E_00B4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24)	32	R/W	0000_0005h	36.4.41/ 2003
20E_00B8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25)	32	R/W	0000_0005h	36.4.42/ 2004
20E_00BC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26)	32	R/W	0000_0005h	36.4.43/ 2005
20E_00C0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27)	32	R/W	0000_0005h	36.4.44/ 2006
20E_00C4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28)	32	R/W	0000_0005h	36.4.45/ 2007
20E_00C8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29)	32	R/W	0000_0005h	36.4.46/ 2008
20E_00CC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30)	32	R/W	0000_0005h	36.4.47/ 2009
20E_00D0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31)	32	R/W	0000_0005h	36.4.48/ 2010
20E_00D4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24)	32	R/W	0000_0000h	36.4.49/ 2011
20E_00D8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23)	32	R/W	0000_0000h	36.4.50/ 2012
20E_00DC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22)	32	R/W	0000_0000h	36.4.51/ 2013
20E_00E0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21)	32	R/W	0000_0000h	36.4.52/ 2014
20E_00E4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20)	32	R/W	0000_0000h	36.4.53/ 2015
20E_00E8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19)	32	R/W	0000_0000h	36.4.54/ 2016
20E_00EC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18)	32	R/W	0000_0000h	36.4.55/ 2017
20E_00F0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17)	32	R/W	0000_0000h	36.4.56/ 2018
20E_00F4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16)	32	R/W	0000_0000h	36.4.57/ 2019
20E_00F8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B)	32	R/W	0000_0000h	36.4.58/ 2020
20E_00FC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B)	32	R/W	0000_0000h	36.4.59/ 2021
20E_0100	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B)	32	R/W	0000_0000h	36.4.60/ 2022
20E_0104	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_RW)	32	R/W	0000_0000h	36.4.61/ 2023

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0108	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B)	32	R/W	0000_0000h	36.4.62/ 2024
20E_010C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B)	32	R/W	0000_0000h	36.4.63/ 2025
20E_0110	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B)	32	R/W	0000_0000h	36.4.64/ 2026
20E_0114	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD00)	32	R/W	0000_0000h	36.4.65/ 2027
20E_0118	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD01)	32	R/W	0000_0000h	36.4.66/ 2028
20E_011C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD02)	32	R/W	0000_0000h	36.4.67/ 2029
20E_0120	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD03)	32	R/W	0000_0000h	36.4.68/ 2030
20E_0124	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD04)	32	R/W	0000_0000h	36.4.69/ 2031
20E_0128	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD05)	32	R/W	0000_0000h	36.4.70/ 2032
20E_012C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD06)	32	R/W	0000_0000h	36.4.71/ 2033
20E_0130	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD07)	32	R/W	0000_0000h	36.4.72/ 2034
20E_0134	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD08)	32	R/W	0000_0000h	36.4.73/ 2035
20E_0138	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD09)	32	R/W	0000_0000h	36.4.74/ 2036
20E_013C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD10)	32	R/W	0000_0000h	36.4.75/ 2037
20E_0140	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD11)	32	R/W	0000_0000h	36.4.76/ 2038
20E_0144	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD12)	32	R/W	0000_0000h	36.4.77/ 2039
20E_0148	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD13)	32	R/W	0000_0000h	36.4.78/ 2040
20E_014C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD14)	32	R/W	0000_0000h	36.4.79/ 2041
20E_0150	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD15)	32	R/W	0000_0000h	36.4.80/ 2042
20E_0154	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B)	32	R/W	0000_0000h	36.4.81/ 2043
20E_0158	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK)	32	R/W	0000_0000h	36.4.82/ 2044
20E_015C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK)	32	R/W	0000_0005h	36.4.83/ 2045

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0160	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15)	32	R/W	0000_0005h	36.4.84/ 2046
20E_0164	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02)	32	R/W	0000_0005h	36.4.85/ 2047
20E_0168	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03)	32	R/W	0000_0005h	36.4.86/ 2048
20E_016C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04)	32	R/W	0000_0005h	36.4.87/ 2049
20E_0170	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00)	32	R/W	0000_0005h	36.4.88/ 2050
20E_0174	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01)	32	R/W	0000_0005h	36.4.89/ 2051
20E_0178	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02)	32	R/W	0000_0005h	36.4.90/ 2052
20E_017C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03)	32	R/W	0000_0005h	36.4.91/ 2053
20E_0180	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04)	32	R/W	0000_0005h	36.4.92/ 2054
20E_0184	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05)	32	R/W	0000_0005h	36.4.93/ 2055
20E_0188	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06)	32	R/W	0000_0005h	36.4.94/ 2056
20E_018C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07)	32	R/W	0000_0005h	36.4.95/ 2057
20E_0190	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08)	32	R/W	0000_0005h	36.4.96/ 2058
20E_0194	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09)	32	R/W	0000_0005h	36.4.97/ 2059
20E_0198	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10)	32	R/W	0000_0005h	36.4.98/ 2060
20E_019C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11)	32	R/W	0000_0005h	36.4.99/ 2061
20E_01A0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12)	32	R/W	0000_0005h	36.4.100/ 2062
20E_01A4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13)	32	R/W	0000_0005h	36.4.101/ 2063
20E_01A8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14)	32	R/W	0000_0005h	36.4.102/ 2064
20E_01AC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15)	32	R/W	0000_0005h	36.4.103/ 2065
20E_01B0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16)	32	R/W	0000_0005h	36.4.104/ 2066
20E_01B4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17)	32	R/W	0000_0005h	36.4.105/ 2067

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_01B8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18)	32	R/W	0000_0005h	36.4.106/ 2068
20E_01BC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19)	32	R/W	0000_0005h	36.4.107/ 2069
20E_01C0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20)	32	R/W	0000_0005h	36.4.108/ 2070
20E_01C4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21)	32	R/W	0000_0005h	36.4.109/ 2071
20E_01C8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22)	32	R/W	0000_0005h	36.4.110/ 2072
20E_01CC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23)	32	R/W	0000_0005h	36.4.111/ 2073
20E_01D0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO)	32	R/W	0000_0005h	36.4.112/ 2074
20E_01D4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK)	32	R/W	0000_0005h	36.4.113/ 2075
20E_01D8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER)	32	R/W	0000_0005h	36.4.114/ 2076
20E_01DC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV)	32	R/W	0000_0005h	36.4.115/ 2077
20E_01E0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1)	32	R/W	0000_0005h	36.4.116/ 2078
20E_01E4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0)	32	R/W	0000_0005h	36.4.117/ 2079
20E_01E8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN)	32	R/W	0000_0005h	36.4.118/ 2080
20E_01EC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1)	32	R/W	0000_0005h	36.4.119/ 2081
20E_01F0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0)	32	R/W	0000_0005h	36.4.120/ 2082
20E_01F4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_MDC)	32	R/W	0000_0005h	36.4.121/ 2083
20E_01F8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL0)	32	R/W	0000_0005h	36.4.122/ 2084
20E_01FC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0)	32	R/W	0000_0005h	36.4.123/ 2085
20E_0200	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL1)	32	R/W	0000_0005h	36.4.124/ 2086
20E_0204	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1)	32	R/W	0000_0005h	36.4.125/ 2087
20E_0208	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL2)	32	R/W	0000_0005h	36.4.126/ 2088
20E_020C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2)	32	R/W	0000_0005h	36.4.127/ 2089

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0210	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL3)	32	R/W	0000_0005h	36.4.128/ 2090
20E_0214	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3)	32	R/W	0000_0005h	36.4.129/ 2091
20E_0218	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL4)	32	R/W	0000_0005h	36.4.130/ 2092
20E_021C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4)	32	R/W	0000_0005h	36.4.131/ 2093
20E_0220	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO00)	32	R/W	0000_0005h	36.4.132/ 2094
20E_0224	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO01)	32	R/W	0000_0005h	36.4.133/ 2095
20E_0228	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO09)	32	R/W	0000_0005h	36.4.134/ 2096
20E_022C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO03)	32	R/W	0000_0005h	36.4.135/ 2097
20E_0230	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO06)	32	R/W	0000_0005h	36.4.136/ 2098
20E_0234	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO02)	32	R/W	0000_0005h	36.4.137/ 2099
20E_0238	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO04)	32	R/W	0000_0005h	36.4.138/ 2100
20E_023C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO05)	32	R/W	0000_0005h	36.4.139/ 2101
20E_0240	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO07)	32	R/W	0000_0005h	36.4.140/ 2102
20E_0244	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO08)	32	R/W	0000_0005h	36.4.141/ 2103
20E_0248	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO16)	32	R/W	0000_0005h	36.4.142/ 2104
20E_024C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO17)	32	R/W	0000_0005h	36.4.143/ 2105
20E_0250	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO18)	32	R/W	0000_0005h	36.4.144/ 2106
20E_0254	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO19)	32	R/W	0000_0005h	36.4.145/ 2107
20E_0258	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK)	32	R/W	0000_0005h	36.4.146/ 2108
20E_025C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC)	32	R/W	0000_0005h	36.4.147/ 2109
20E_0260	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN)	32	R/W	0000_0005h	36.4.148/ 2110
20E_0264	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC)	32	R/W	0000_0005h	36.4.149/ 2111

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0268	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04)	32	R/W	0000_0005h	36.4.150/ 2112
20E_026C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05)	32	R/W	0000_0005h	36.4.151/ 2113
20E_0270	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06)	32	R/W	0000_0005h	36.4.152/ 2114
20E_0274	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07)	32	R/W	0000_0005h	36.4.153/ 2115
20E_0278	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08)	32	R/W	0000_0005h	36.4.154/ 2116
20E_027C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09)	32	R/W	0000_0005h	36.4.155/ 2117
20E_0280	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10)	32	R/W	0000_0005h	36.4.156/ 2118
20E_0284	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11)	32	R/W	0000_0005h	36.4.157/ 2119
20E_0288	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12)	32	R/W	0000_0005h	36.4.158/ 2120
20E_028C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13)	32	R/W	0000_0005h	36.4.159/ 2121
20E_0290	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14)	32	R/W	0000_0005h	36.4.160/ 2122
20E_0294	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15)	32	R/W	0000_0005h	36.4.161/ 2123
20E_0298	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16)	32	R/W	0000_0005h	36.4.162/ 2124
20E_029C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17)	32	R/W	0000_0005h	36.4.163/ 2125
20E_02A0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18)	32	R/W	0000_0005h	36.4.164/ 2126
20E_02A4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19)	32	R/W	0000_0005h	36.4.165/ 2127
20E_02A8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7)	32	R/W	0000_0005h	36.4.166/ 2128
20E_02AC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6)	32	R/W	0000_0005h	36.4.167/ 2129
20E_02B0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5)	32	R/W	0000_0005h	36.4.168/ 2130
20E_02B4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4)	32	R/W	0000_0005h	36.4.169/ 2131
20E_02B8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_CMD)	32	R/W	0000_0005h	36.4.170/ 2132
20E_02BC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_CLK)	32	R/W	0000_0005h	36.4.171/ 2133

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_02C0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0)	32	R/W	0000_0005h	36.4.172/ 2134
20E_02C4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1)	32	R/W	0000_0005h	36.4.173/ 2135
20E_02C8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA2)	32	R/W	0000_0005h	36.4.174/ 2136
20E_02CC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3)	32	R/W	0000_0005h	36.4.175/ 2136
20E_02D0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_RESET)	32	R/W	0000_0005h	36.4.176/ 2137
20E_02D4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CLE)	32	R/W	0000_0005h	36.4.177/ 2138
20E_02D8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_ALE)	32	R/W	0000_0005h	36.4.178/ 2139
20E_02DC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B)	32	R/W	0000_0005h	36.4.179/ 2140
20E_02E0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B)	32	R/W	0000_0005h	36.4.180/ 2141
20E_02E4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS0_B)	32	R/W	0000_0005h	36.4.181/ 2142
20E_02E8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B)	32	R/W	0000_0005h	36.4.182/ 2142
20E_02EC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B)	32	R/W	0000_0005h	36.4.183/ 2143
20E_02F0	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B)	32	R/W	0000_0005h	36.4.184/ 2144
20E_02F4	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_CMD)	32	R/W	0000_0005h	36.4.185/ 2145
20E_02F8	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_CLK)	32	R/W	0000_0005h	36.4.186/ 2146
20E_02FC	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00)	32	R/W	0000_0005h	36.4.187/ 2147
20E_0300	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01)	32	R/W	0000_0005h	36.4.188/ 2148
20E_0304	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02)	32	R/W	0000_0005h	36.4.189/ 2149
20E_0308	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03)	32	R/W	0000_0005h	36.4.190/ 2150
20E_030C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04)	32	R/W	0000_0005h	36.4.191/ 2151
20E_0310	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05)	32	R/W	0000_0005h	36.4.192/ 2152
20E_0314	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06)	32	R/W	0000_0005h	36.4.193/ 2153

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0318	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07)	32	R/W	0000_0005h	36.4.194/ 2154
20E_031C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0)	32	R/W	0000_0005h	36.4.195/ 2155
20E_0320	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1)	32	R/W	0000_0005h	36.4.196/ 2156
20E_0324	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2)	32	R/W	0000_0005h	36.4.197/ 2157
20E_0328	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA3)	32	R/W	0000_0005h	36.4.198/ 2158
20E_032C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4)	32	R/W	0000_0005h	36.4.199/ 2158
20E_0330	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5)	32	R/W	0000_0005h	36.4.200/ 2159
20E_0334	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6)	32	R/W	0000_0005h	36.4.201/ 2160
20E_0338	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7)	32	R/W	0000_0005h	36.4.202/ 2161
20E_033C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1)	32	R/W	0000_0005h	36.4.203/ 2162
20E_0340	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0)	32	R/W	0000_0005h	36.4.204/ 2163
20E_0344	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3)	32	R/W	0000_0005h	36.4.205/ 2164
20E_0348	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_CMD)	32	R/W	0000_0005h	36.4.206/ 2165
20E_034C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2)	32	R/W	0000_0005h	36.4.207/ 2166
20E_0350	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_CLK)	32	R/W	0000_0005h	36.4.208/ 2167
20E_0354	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_CLK)	32	R/W	0000_0005h	36.4.209/ 2168
20E_0358	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_CMD)	32	R/W	0000_0005h	36.4.210/ 2169
20E_035C	Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3)	32	R/W	0000_0005h	36.4.211/ 2170
20E_0360	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA1)	32	R/W	0001_B0B0h	36.4.212/ 2171
20E_0364	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA2)	32	R/W	0001_B0B0h	36.4.213/ 2172
20E_0368	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA0)	32	R/W	0001_B0B0h	36.4.214/ 2174
20E_036C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TXC)	32	R/W	0001_3030h	36.4.215/ 2176

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0370	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD0)	32	R/W	0001_B030h	36.4.216/ 2178
20E_0374	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD1)	32	R/W	0001_B030h	36.4.217/ 2179
20E_0378	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD2)	32	R/W	0001_B030h	36.4.218/ 2181
20E_037C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD3)	32	R/W	0001_B030h	36.4.219/ 2183
20E_0380	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RX_CTL)	32	R/W	0001_3030h	36.4.220/ 2184
20E_0384	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD0)	32	R/W	0001_B030h	36.4.221/ 2186
20E_0388	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TX_CTL)	32	R/W	0001_3030h	36.4.222/ 2188
20E_038C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD1)	32	R/W	0001_B030h	36.4.223/ 2189
20E_0390	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD2)	32	R/W	0001_B030h	36.4.224/ 2191
20E_0394	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD3)	32	R/W	0001_B030h	36.4.225/ 2193
20E_0398	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RXC)	32	R/W	0001_3030h	36.4.226/ 2194
20E_039C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR25)	32	R/W	0000_B0B1h	36.4.227/ 2196
20E_03A0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB2_B)	32	R/W	0001_B0B0h	36.4.228/ 2198
20E_03A4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA16)	32	R/W	0001_B0B0h	36.4.229/ 2199
20E_03A8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA17)	32	R/W	0001_B0B0h	36.4.230/ 2201
20E_03AC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA18)	32	R/W	0001_B0B0h	36.4.231/ 2203
20E_03B0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA19)	32	R/W	0001_B0B0h	36.4.232/ 2205
20E_03B4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA20)	32	R/W	0001_B0B0h	36.4.233/ 2206
20E_03B8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA21)	32	R/W	0001_B0B0h	36.4.234/ 2208
20E_03BC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA22)	32	R/W	0001_B0B0h	36.4.235/ 2210
20E_03C0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA23)	32	R/W	0001_B0B0h	36.4.236/ 2212
20E_03C4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB3_B)	32	R/W	0001_B0B0h	36.4.237/ 2213

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_03C8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA24)	32	R/W	0001_B0B0h	36.4.238/ 2215
20E_03CC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA25)	32	R/W	0001_B0B0h	36.4.239/ 2217
20E_03D0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA26)	32	R/W	0001_B0B0h	36.4.240/ 2219
20E_03D4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA27)	32	R/W	0001_B0B0h	36.4.241/ 2220
20E_03D8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA28)	32	R/W	0001_B0B0h	36.4.242/ 2222
20E_03DC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA29)	32	R/W	0001_B0B0h	36.4.243/ 2224
20E_03E0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA30)	32	R/W	0001_B0B0h	36.4.244/ 2226
20E_03E4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA31)	32	R/W	0001_B0B0h	36.4.245/ 2227
20E_03E8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR24)	32	R/W	0000_B0B1h	36.4.246/ 2229
20E_03EC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR23)	32	R/W	0000_B0B1h	36.4.247/ 2231
20E_03F0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR22)	32	R/W	0000_B0B1h	36.4.248/ 2233
20E_03F4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR21)	32	R/W	0000_B0B1h	36.4.249/ 2234
20E_03F8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR20)	32	R/W	0000_B0B1h	36.4.250/ 2236
20E_03FC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR19)	32	R/W	0000_B0B1h	36.4.251/ 2238
20E_0400	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR18)	32	R/W	0000_B0B1h	36.4.252/ 2240
20E_0404	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR17)	32	R/W	0000_B0B1h	36.4.253/ 2241
20E_0408	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR16)	32	R/W	0000_B0B1h	36.4.254/ 2243
20E_040C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_CS0_B)	32	R/W	0000_B0B1h	36.4.255/ 2245
20E_0410	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_CS1_B)	32	R/W	0000_B0B1h	36.4.256/ 2247
20E_0414	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_OE_B)	32	R/W	0000_B0B1h	36.4.257/ 2248
20E_0418	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_RW)	32	R/W	0000_B0B1h	36.4.258/ 2250
20E_041C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_LBA_B)	32	R/W	0000_B0B1h	36.4.259/ 2252

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0420	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB0_B)	32	R/W	0000_B0B1h	36.4.260/ 2254
20E_0424	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB1_B)	32	R/W	0000_B0B1h	36.4.261/ 2255
20E_0428	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD00)	32	R/W	0000_B0B1h	36.4.262/ 2257
20E_042C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD01)	32	R/W	0000_B0B1h	36.4.263/ 2259
20E_0430	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD02)	32	R/W	0000_B0B1h	36.4.264/ 2261
20E_0434	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD03)	32	R/W	0000_B0B1h	36.4.265/ 2262
20E_0438	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD04)	32	R/W	0000_B0B1h	36.4.266/ 2264
20E_043C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD05)	32	R/W	0000_B0B1h	36.4.267/ 2266
20E_0440	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD06)	32	R/W	0000_B0B1h	36.4.268/ 2268
20E_0444	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD07)	32	R/W	0000_B0B1h	36.4.269/ 2269
20E_0448	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD08)	32	R/W	0000_B0B1h	36.4.270/ 2271
20E_044C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD09)	32	R/W	0000_B0B1h	36.4.271/ 2273
20E_0450	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD10)	32	R/W	0000_B0B1h	36.4.272/ 2275
20E_0454	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD11)	32	R/W	0000_B0B1h	36.4.273/ 2276
20E_0458	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD12)	32	R/W	0000_B0B1h	36.4.274/ 2278
20E_045C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD13)	32	R/W	0000_B0B1h	36.4.275/ 2280
20E_0460	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD14)	32	R/W	0000_B0B1h	36.4.276/ 2282
20E_0464	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD15)	32	R/W	0000_B0B1h	36.4.277/ 2283
20E_0468	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_WAIT_B)	32	R/W	0000_B060h	36.4.278/ 2285
20E_046C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_BCLK)	32	R/W	0000_B0B1h	36.4.279/ 2287
20E_0470	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_DISP_CLK)	32	R/W	0001_B0B0h	36.4.280/ 2289
20E_0474	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN15)	32	R/W	0001_B0B0h	36.4.281/ 2290

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0478	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN02)	32	R/W	0001_B0B0h	36.4.282/ 2292
20E_047C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN03)	32	R/W	0001_B0B0h	36.4.283/ 2294
20E_0480	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN04)	32	R/W	0001_B0B0h	36.4.284/ 2296
20E_0484	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA00)	32	R/W	0001_B0B0h	36.4.285/ 2297
20E_0488	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA01)	32	R/W	0001_B0B0h	36.4.286/ 2299
20E_048C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA02)	32	R/W	0001_B0B0h	36.4.287/ 2301
20E_0490	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA03)	32	R/W	0001_B0B0h	36.4.288/ 2303
20E_0494	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA04)	32	R/W	0001_B0B0h	36.4.289/ 2304
20E_0498	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA05)	32	R/W	0001_B0B0h	36.4.290/ 2306
20E_049C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA06)	32	R/W	0001_B0B0h	36.4.291/ 2308
20E_04A0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA07)	32	R/W	0001_B0B0h	36.4.292/ 2310
20E_04A4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA08)	32	R/W	0001_B0B0h	36.4.293/ 2311
20E_04A8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA09)	32	R/W	0001_B0B0h	36.4.294/ 2313
20E_04AC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA10)	32	R/W	0001_B0B0h	36.4.295/ 2315
20E_04B0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA11)	32	R/W	0001_B0B0h	36.4.296/ 2317
20E_04B4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA12)	32	R/W	0001_B0B0h	36.4.297/ 2318
20E_04B8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA13)	32	R/W	0001_B0B0h	36.4.298/ 2320
20E_04BC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA14)	32	R/W	0001_B0B0h	36.4.299/ 2322
20E_04C0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA15)	32	R/W	0001_B0B0h	36.4.300/ 2324
20E_04C4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA16)	32	R/W	0001_B0B0h	36.4.301/ 2325
20E_04C8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA17)	32	R/W	0001_B0B0h	36.4.302/ 2327
20E_04CC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA18)	32	R/W	0001_B0B0h	36.4.303/ 2329

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_04D0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA19)	32	R/W	0001_B0B0h	36.4.304/ 2331
20E_04D4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA20)	32	R/W	0001_B0B0h	36.4.305/ 2332
20E_04D8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA21)	32	R/W	0001_B0B0h	36.4.306/ 2334
20E_04DC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA22)	32	R/W	0001_B0B0h	36.4.307/ 2336
20E_04E0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA23)	32	R/W	0001_B0B0h	36.4.308/ 2338
20E_04E4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_MDIO)	32	R/W	0001_B0B0h	36.4.309/ 2339
20E_04E8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_REF_CLK)	32	R/W	0001_B0B0h	36.4.310/ 2341
20E_04EC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_ER)	32	R/W	0001_B0B0h	36.4.311/ 2343
20E_04F0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_CRS_DV)	32	R/W	0001_B0B0h	36.4.312/ 2345
20E_04F4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA1)	32	R/W	0001_B0B0h	36.4.313/ 2346
20E_04F8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA0)	32	R/W	0001_B0B0h	36.4.314/ 2348
20E_04FC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_EN)	32	R/W	0001_B0B0h	36.4.315/ 2350
20E_0500	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA1)	32	R/W	0001_B0B0h	36.4.316/ 2352
20E_0504	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA0)	32	R/W	0001_B0B0h	36.4.317/ 2353
20E_0508	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_MDC)	32	R/W	0001_B0B0h	36.4.318/ 2355
20E_050C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS5_P)	32	R/W	0000_2030h	36.4.319/ 2357
20E_0510	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM5)	32	R/W	0000_8030h	36.4.320/ 2359
20E_0514	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM4)	32	R/W	0000_8030h	36.4.321/ 2361
20E_0518	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS4_P)	32	R/W	0000_2030h	36.4.322/ 2363
20E_051C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS3_P)	32	R/W	0000_2030h	36.4.323/ 2365
20E_0520	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM3)	32	R/W	0000_8030h	36.4.324/ 2367
20E_0524	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2_P)	32	R/W	0000_2030h	36.4.325/ 2369

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0528	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM2)	32	R/W	0000_8030h	36.4.326/ 2371
20E_052C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR00)	32	R/W	0000_8000h	36.4.327/ 2373
20E_0530	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR01)	32	R/W	0000_8000h	36.4.328/ 2375
20E_0534	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR02)	32	R/W	0000_8000h	36.4.329/ 2377
20E_0538	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR03)	32	R/W	0000_8000h	36.4.330/ 2379
20E_053C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR04)	32	R/W	0000_8000h	36.4.331/ 2381
20E_0540	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR05)	32	R/W	0000_8000h	36.4.332/ 2383
20E_0544	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR06)	32	R/W	0000_8000h	36.4.333/ 2385
20E_0548	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR07)	32	R/W	0000_8000h	36.4.334/ 2387
20E_054C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR08)	32	R/W	0000_8000h	36.4.335/ 2389
20E_0550	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR09)	32	R/W	0000_8000h	36.4.336/ 2391
20E_0554	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR10)	32	R/W	0000_8000h	36.4.337/ 2393
20E_0558	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR11)	32	R/W	0000_8000h	36.4.338/ 2395
20E_055C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR12)	32	R/W	0000_8000h	36.4.339/ 2397
20E_0560	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR13)	32	R/W	0000_8000h	36.4.340/ 2399
20E_0564	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR14)	32	R/W	0000_8000h	36.4.341/ 2401
20E_0568	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR15)	32	R/W	0000_8000h	36.4.342/ 2403
20E_056C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CAS_B)	32	R/W	0000_8030h	36.4.343/ 2405
20E_0570	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CS0_B)	32	R/W	0000_8000h	36.4.344/ 2407
20E_0574	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CS1_B)	32	R/W	0000_8000h	36.4.345/ 2409
20E_0578	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_RAS_B)	32	R/W	0000_8030h	36.4.346/ 2411
20E_057C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_RESET)	32	R/W	0008_3030h	36.4.347/ 2413

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0580	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA0)	32	R/W	0000_8000h	36.4.348/ 2415
20E_0584	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA1)	32	R/W	0000_8000h	36.4.349/ 2417
20E_0588	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK0_P)	32	R/W	0000_8030h	36.4.350/ 2419
20E_058C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA2)	32	R/W	0000_B000h	36.4.351/ 2421
20E_0590	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE0)	32	R/W	0000_3000h	36.4.352/ 2423
20E_0594	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK1_P)	32	R/W	0000_8030h	36.4.353/ 2425
20E_0598	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE1)	32	R/W	0000_3000h	36.4.354/ 2427
20E_059C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT0)	32	R/W	0000_3030h	36.4.355/ 2429
20E_05A0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT1)	32	R/W	0000_3030h	36.4.356/ 2431
20E_05A4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDWE_B)	32	R/W	0000_8000h	36.4.357/ 2433
20E_05A8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0_P)	32	R/W	0000_2030h	36.4.358/ 2435
20E_05AC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM0)	32	R/W	0000_8030h	36.4.359/ 2437
20E_05B0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1_P)	32	R/W	0000_2030h	36.4.360/ 2439
20E_05B4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM1)	32	R/W	0000_8030h	36.4.361/ 2441
20E_05B8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS6_P)	32	R/W	0000_2030h	36.4.362/ 2443
20E_05BC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM6)	32	R/W	0000_8030h	36.4.363/ 2445
20E_05C0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS7_P)	32	R/W	0000_2030h	36.4.364/ 2447
20E_05C4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM7)	32	R/W	0000_8030h	36.4.365/ 2449
20E_05C8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL0)	32	R/W	0001_B0B0h	36.4.366/ 2451
20E_05CC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW0)	32	R/W	0001_B0B0h	36.4.367/ 2452
20E_05D0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL1)	32	R/W	0001_B0B0h	36.4.368/ 2454
20E_05D4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW1)	32	R/W	0001_B0B0h	36.4.369/ 2456

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_05D8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL2)	32	R/W	0001_B0B0h	36.4.370/ 2458
20E_05DC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW2)	32	R/W	0001_B0B0h	36.4.371/ 2459
20E_05E0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL3)	32	R/W	0001_B0B0h	36.4.372/ 2461
20E_05E4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW3)	32	R/W	0001_B0B0h	36.4.373/ 2463
20E_05E8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL4)	32	R/W	0001_B0B0h	36.4.374/ 2465
20E_05EC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW4)	32	R/W	0001_B0B0h	36.4.375/ 2466
20E_05F0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO00)	32	R/W	0001_B0B0h	36.4.376/ 2468
20E_05F4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO01)	32	R/W	0001_B0B0h	36.4.377/ 2470
20E_05F8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO09)	32	R/W	0001_B0B0h	36.4.378/ 2472
20E_05FC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO03)	32	R/W	0001_B0B0h	36.4.379/ 2473
20E_0600	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO06)	32	R/W	0001_B0B0h	36.4.380/ 2475
20E_0604	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO02)	32	R/W	0001_B0B0h	36.4.381/ 2477
20E_0608	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO04)	32	R/W	0001_B0B0h	36.4.382/ 2479
20E_060C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO05)	32	R/W	0001_B0B0h	36.4.383/ 2480
20E_0610	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO07)	32	R/W	0001_B0B0h	36.4.384/ 2482
20E_0614	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO08)	32	R/W	0001_B0B0h	36.4.385/ 2484
20E_0618	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO16)	32	R/W	0001_B0B0h	36.4.386/ 2486
20E_061C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO17)	32	R/W	0001_B0B0h	36.4.387/ 2487
20E_0620	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO18)	32	R/W	0001_B0B0h	36.4.388/ 2489
20E_0624	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO19)	32	R/W	0001_B0B0h	36.4.389/ 2491
20E_0628	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_PIXCLK)	32	R/W	0001_B0B0h	36.4.390/ 2492
20E_062C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_HSYNC)	32	R/W	0001_B0B0h	36.4.391/ 2494

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0630	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA_EN)	32	R/W	0001_B0B0h	36.4.392/ 2496
20E_0634	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_VSYNC)	32	R/W	0001_B0B0h	36.4.393/ 2498
20E_0638	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA04)	32	R/W	0001_B0B0h	36.4.394/ 2499
20E_063C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA05)	32	R/W	0001_B0B0h	36.4.395/ 2501
20E_0640	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA06)	32	R/W	0001_B0B0h	36.4.396/ 2503
20E_0644	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA07)	32	R/W	0001_B0B0h	36.4.397/ 2505
20E_0648	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA08)	32	R/W	0001_B0B0h	36.4.398/ 2506
20E_064C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA09)	32	R/W	0001_B0B0h	36.4.399/ 2508
20E_0650	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA10)	32	R/W	0001_B0B0h	36.4.400/ 2510
20E_0654	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA11)	32	R/W	0001_B0B0h	36.4.401/ 2512
20E_0658	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA12)	32	R/W	0001_B0B0h	36.4.402/ 2513
20E_065C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA13)	32	R/W	0001_B0B0h	36.4.403/ 2515
20E_0660	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA14)	32	R/W	0001_B0B0h	36.4.404/ 2517
20E_0664	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA15)	32	R/W	0001_B0B0h	36.4.405/ 2519
20E_0668	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA16)	32	R/W	0001_B0B0h	36.4.406/ 2520
20E_066C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA17)	32	R/W	0001_B0B0h	36.4.407/ 2522
20E_0670	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA18)	32	R/W	0001_B0B0h	36.4.408/ 2524
20E_0674	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA19)	32	R/W	0001_B0B0h	36.4.409/ 2526
20E_0678	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TMS)	32	R/W	0000_7060h	36.4.410/ 2527
20E_067C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_MOD)	32	R/W	0000_B060h	36.4.411/ 2529
20E_0680	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TRSTB)	32	R/W	0000_7060h	36.4.412/ 2531
20E_0684	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TDI)	32	R/W	0000_7060h	36.4.413/ 2532

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0688	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TCK)	32	R/W	0000_7060h	36.4.414/ 2534
20E_068C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TDO)	32	R/W	0000_90B1h	36.4.415/ 2536
20E_0690	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA7)	32	R/W	0001_B0B0h	36.4.416/ 2537
20E_0694	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA6)	32	R/W	0001_B0B0h	36.4.417/ 2539
20E_0698	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA5)	32	R/W	0001_B0B0h	36.4.418/ 2541
20E_069C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA4)	32	R/W	0001_B0B0h	36.4.419/ 2542
20E_06A0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_CMD)	32	R/W	0001_B0B0h	36.4.420/ 2544
20E_06A4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_CLK)	32	R/W	0001_B0B0h	36.4.421/ 2546
20E_06A8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA0)	32	R/W	0001_B0B0h	36.4.422/ 2548
20E_06AC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA1)	32	R/W	0001_B0B0h	36.4.423/ 2549
20E_06B0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA2)	32	R/W	0001_B0B0h	36.4.424/ 2551
20E_06B4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA3)	32	R/W	0001_B0B0h	36.4.425/ 2553
20E_06B8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_RESET)	32	R/W	0001_B0B0h	36.4.426/ 2555
20E_06BC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CLE)	32	R/W	0001_B0B0h	36.4.427/ 2556
20E_06C0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_ALE)	32	R/W	0001_B0B0h	36.4.428/ 2558
20E_06C4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_WP_B)	32	R/W	0001_B0B0h	36.4.429/ 2560
20E_06C8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_READY_B)	32	R/W	0001_B0B0h	36.4.430/ 2562
20E_06CC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS0_B)	32	R/W	0001_B0B0h	36.4.431/ 2563
20E_06D0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS1_B)	32	R/W	0001_B0B0h	36.4.432/ 2565
20E_06D4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS2_B)	32	R/W	0001_B0B0h	36.4.433/ 2567
20E_06D8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS3_B)	32	R/W	0001_B0B0h	36.4.434/ 2569
20E_06DC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_CMD)	32	R/W	0001_B0B0h	36.4.435/ 2570

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_06E0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_CLK)	32	R/W	0001_B0B0h	36.4.436/ 2572
20E_06E4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA00)	32	R/W	0001_B0B0h	36.4.437/ 2574
20E_06E8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA01)	32	R/W	0001_B0B0h	36.4.438/ 2576
20E_06EC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA02)	32	R/W	0001_B0B0h	36.4.439/ 2577
20E_06F0	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA03)	32	R/W	0001_B0B0h	36.4.440/ 2579
20E_06F4	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA04)	32	R/W	0001_B0B0h	36.4.441/ 2581
20E_06F8	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA05)	32	R/W	0001_B0B0h	36.4.442/ 2583
20E_06FC	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA06)	32	R/W	0001_B0B0h	36.4.443/ 2584
20E_0700	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA07)	32	R/W	0001_B0B0h	36.4.444/ 2586
20E_0704	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA0)	32	R/W	0001_B0B0h	36.4.445/ 2588
20E_0708	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA1)	32	R/W	0001_B0B0h	36.4.446/ 2590
20E_070C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA2)	32	R/W	0001_B0B0h	36.4.447/ 2591
20E_0710	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA3)	32	R/W	0001_B0B0h	36.4.448/ 2593
20E_0714	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA4)	32	R/W	0001_B0B0h	36.4.449/ 2595
20E_0718	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA5)	32	R/W	0001_B0B0h	36.4.450/ 2597
20E_071C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA6)	32	R/W	0001_B0B0h	36.4.451/ 2598
20E_0720	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA7)	32	R/W	0001_B0B0h	36.4.452/ 2600
20E_0724	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA1)	32	R/W	0001_B0B0h	36.4.453/ 2602
20E_0728	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA0)	32	R/W	0001_B0B0h	36.4.454/ 2604
20E_072C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA3)	32	R/W	0001_B0B0h	36.4.455/ 2605
20E_0730	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_CMD)	32	R/W	0001_B0B0h	36.4.456/ 2607
20E_0734	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA2)	32	R/W	0001_B0B0h	36.4.457/ 2609

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0738	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_CLK)	32	R/W	0001_B0B0h	36.4.458/ 2611
20E_073C	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_CLK)	32	R/W	0001_B0B0h	36.4.459/ 2612
20E_0740	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_CMD)	32	R/W	0001_B0B0h	36.4.460/ 2614
20E_0744	Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA3)	32	R/W	0001_B0B0h	36.4.461/ 2616
20E_0748	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B7DS)	32	R/W	0000_0030h	36.4.462/ 2618
20E_074C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_ADDDS)	32	R/W	0000_0030h	36.4.463/ 2618
20E_0750	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL)	32	R/W	0000_0000h	36.4.464/ 2619
20E_0754	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL0)	32	R/W	0000_0000h	36.4.465/ 2620
20E_0758	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRPKE)	32	R/W	0000_1000h	36.4.466/ 2621
20E_075C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL1)	32	R/W	0000_0000h	36.4.467/ 2621
20E_0760	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL2)	32	R/W	0000_0000h	36.4.468/ 2622
20E_0764	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL3)	32	R/W	0000_0000h	36.4.469/ 2623
20E_0768	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRPK)	32	R/W	0000_2000h	36.4.470/ 2624
20E_076C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL4)	32	R/W	0000_0000h	36.4.471/ 2624
20E_0770	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRHYS)	32	R/W	0000_0000h	36.4.472/ 2625
20E_0774	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRMODE)	32	R/W	0000_0000h	36.4.473/ 2626
20E_0778	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL5)	32	R/W	0000_0000h	36.4.474/ 2627
20E_077C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL6)	32	R/W	0000_0000h	36.4.475/ 2628
20E_0780	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL7)	32	R/W	0000_0000h	36.4.476/ 2628
20E_0784	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B0DS)	32	R/W	0000_0030h	36.4.477/ 2629
20E_0788	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B1DS)	32	R/W	0000_0030h	36.4.478/ 2630
20E_078C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_CTLDS)	32	R/W	0000_0030h	36.4.479/ 2630

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0790	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII)	32	R/W	0008_0000h	36.4.480/ 2631
20E_0794	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B2DS)	32	R/W	0000_0030h	36.4.481/ 2632
20E_0798	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE)	32	R/W	0008_0000h	36.4.482/ 2633
20E_079C	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B3DS)	32	R/W	0000_0030h	36.4.483/ 2634
20E_07A0	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B4DS)	32	R/W	0000_0030h	36.4.484/ 2634
20E_07A4	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B5DS)	32	R/W	0000_0030h	36.4.485/ 2635
20E_07A8	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B6DS)	32	R/W	0000_0030h	36.4.486/ 2636
20E_07AC	Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM)	32	R/W	0000_0000h	36.4.487/ 2636
20E_07B0	Select Input Register (IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT)	32	R/W	0000_0000h	36.4.488/ 2637
20E_07B4	Select Input Register (IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.489/ 2638
20E_07B8	Select Input Register (IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.490/ 2639
20E_07BC	Select Input Register (IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.491/ 2640
20E_07C0	Select Input Register (IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.492/ 2641
20E_07C4	Select Input Register (IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.493/ 2642
20E_07C8	Select Input Register (IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.494/ 2643
20E_07CC	Select Input Register (IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.495/ 2644
20E_07D0	Select Input Register (IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.496/ 2645
20E_07D4	Select Input Register (IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.497/ 2646
20E_07D8	Select Input Register (IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.498/ 2647
20E_07DC	Select Input Register (IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.499/ 2648
20E_07E0	Select Input Register (IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.500/ 2649
20E_07E4	Select Input Register (IOMUXC_FLEXCAN1_RX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.501/ 2649

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_07E8	Select Input Register (IOMUXC_FLEXCAN2_RX_SELECT_INPUT)	32	R/W	0000_0000h	36.4.502/ 2650
20E_07F0	Select Input Register (IOMUXC_CCM_PMIC_READY_SELECT_INPUT)	32	R/W	0000_0000h	36.4.503/ 2651
20E_07F4	Select Input Register (IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.504/ 2651
20E_07F8	Select Input Register (IOMUXC_ECSPI1_MISO_SELECT_INPUT)	32	R/W	0000_0000h	36.4.505/ 2652
20E_07FC	Select Input Register (IOMUXC_ECSPI1_MOSI_SELECT_INPUT)	32	R/W	0000_0000h	36.4.506/ 2653
20E_0800	Select Input Register (IOMUXC_ECSPI1_SS0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.507/ 2653
20E_0804	Select Input Register (IOMUXC_ECSPI1_SS1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.508/ 2654
20E_0808	Select Input Register (IOMUXC_ECSPI1_SS2_SELECT_INPUT)	32	R/W	0000_0000h	36.4.509/ 2655
20E_080C	Select Input Register (IOMUXC_ECSPI1_SS3_SELECT_INPUT)	32	R/W	0000_0000h	36.4.510/ 2656
20E_0810	Select Input Register (IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.511/ 2656
20E_0814	Select Input Register (IOMUXC_ECSPI2_MISO_SELECT_INPUT)	32	R/W	0000_0000h	36.4.512/ 2657
20E_0818	Select Input Register (IOMUXC_ECSPI2_MOSI_SELECT_INPUT)	32	R/W	0000_0000h	36.4.513/ 2658
20E_081C	Select Input Register (IOMUXC_ECSPI2_SS0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.514/ 2658
20E_0820	Select Input Register (IOMUXC_ECSPI2_SS1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.515/ 2659
20E_0824	Select Input Register (IOMUXC_ECSPI4_SS0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.516/ 2660
20E_0828	Select Input Register (IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.517/ 2661
20E_082C	Select Input Register (IOMUXC_ECSPI5_MISO_SELECT_INPUT)	32	R/W	0000_0000h	36.4.518/ 2662
20E_0830	Select Input Register (IOMUXC_ECSPI5_MOSI_SELECT_INPUT)	32	R/W	0000_0000h	36.4.519/ 2663
20E_0834	Select Input Register (IOMUXC_ECSPI5_SS0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.520/ 2664
20E_0838	Select Input Register (IOMUXC_ECSPI5_SS1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.521/ 2665
20E_083C	Select Input Register (IOMUXC_ENET_REF_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.522/ 2666
20E_0840	Select Input Register (IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT)	32	R/W	0000_0000h	36.4.523/ 2667

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0844	Select Input Register (IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.524/ 2668
20E_0848	Select Input Register (IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.525/ 2669
20E_084C	Select Input Register (IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.526/ 2670
20E_0850	Select Input Register (IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT)	32	R/W	0000_0000h	36.4.527/ 2671
20E_0854	Select Input Register (IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT)	32	R/W	0000_0000h	36.4.528/ 2672
20E_0858	Select Input Register (IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.529/ 2673
20E_085C	Select Input Register (IOMUXC_ESAI_RX_FS_SELECT_INPUT)	32	R/W	0000_0000h	36.4.530/ 2674
20E_0860	Select Input Register (IOMUXC_ESAI_TX_FS_SELECT_INPUT)	32	R/W	0000_0000h	36.4.531/ 2675
20E_0864	Select Input Register (IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.532/ 2676
20E_0868	Select Input Register (IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.533/ 2677
20E_086C	Select Input Register (IOMUXC_ESAI_RX_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.534/ 2678
20E_0870	Select Input Register (IOMUXC_ESAI_TX_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.535/ 2679
20E_0874	Select Input Register (IOMUXC_ESAI_SDO0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.536/ 2680
20E_0878	Select Input Register (IOMUXC_ESAI_SDO1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.537/ 2681
20E_087C	Select Input Register (IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT)	32	R/W	0000_0000h	36.4.538/ 2682
20E_0880	Select Input Register (IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT)	32	R/W	0000_0000h	36.4.539/ 2683
20E_0884	Select Input Register (IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.540/ 2684
20E_0888	Select Input Register (IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT)	32	R/W	0000_0000h	36.4.541/ 2685
20E_088C	Select Input Register (IOMUXC_HDMI_ICECIN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.542/ 2686
20E_0890	Select Input Register (IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.543/ 2687
20E_0894	Select Input Register (IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.544/ 2688
20E_0898	Select Input Register (IOMUXC_I2C1_SCL_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.545/ 2689

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_089C	Select Input Register (IOMUXC_I2C1_SDA_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.546/ 2690
20E_08A0	Select Input Register (IOMUXC_I2C2_SCL_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.547/ 2691
20E_08A4	Select Input Register (IOMUXC_I2C2_SDA_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.548/ 2692
20E_08A8	Select Input Register (IOMUXC_I2C3_SCL_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.549/ 2692
20E_08AC	Select Input Register (IOMUXC_I2C3_SDA_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.550/ 2693
20E_08B0	Select Input Register (IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT)	32	R/W	0000_0000h	36.4.551/ 2694
20E_08B4	Select Input Register (IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT)	32	R/W	0000_0000h	36.4.552/ 2695
20E_08B8	Select Input Register (IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT)	32	R/W	0000_0000h	36.4.553/ 2696
20E_08BC	Select Input Register (IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT)	32	R/W	0000_0000h	36.4.554/ 2697
20E_08C0	Select Input Register (IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT)	32	R/W	0000_0000h	36.4.555/ 2698
20E_08C4	Select Input Register (IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT)	32	R/W	0000_0000h	36.4.556/ 2699
20E_08C8	Select Input Register (IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT)	32	R/W	0000_0000h	36.4.557/ 2700
20E_08CC	Select Input Register (IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT)	32	R/W	0000_0000h	36.4.558/ 2701
20E_08D0	Select Input Register (IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT)	32	R/W	0000_0000h	36.4.559/ 2702
20E_08D4	Select Input Register (IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT)	32	R/W	0000_0000h	36.4.560/ 2703
20E_08D8	Select Input Register (IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.561/ 2704
20E_08DC	Select Input Register (IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT)	32	R/W	0000_0000h	36.4.562/ 2705
20E_08E0	Select Input Register (IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT)	32	R/W	0000_0000h	36.4.563/ 2706
20E_08E4	Select Input Register (IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT)	32	R/W	0000_0000h	36.4.564/ 2707
20E_08E8	Select Input Register (IOMUXC_KEY_COL5_SELECT_INPUT)	32	R/W	0000_0000h	36.4.565/ 2707
20E_08EC	Select Input Register (IOMUXC_KEY_COL6_SELECT_INPUT)	32	R/W	0000_0000h	36.4.566/ 2708
20E_08F0	Select Input Register (IOMUXC_KEY_COL7_SELECT_INPUT)	32	R/W	0000_0000h	36.4.567/ 2709

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_08F4	Select Input Register (IOMUXC_KEY_ROW5_SELECT_INPUT)	32	R/W	0000_0000h	36.4.568/ 2709
20E_08F8	Select Input Register (IOMUXC_KEY_ROW6_SELECT_INPUT)	32	R/W	0000_0000h	36.4.569/ 2710
20E_08FC	Select Input Register (IOMUXC_KEY_ROW7_SELECT_INPUT)	32	R/W	0000_0000h	36.4.570/ 2711
20E_0900	Select Input Register (IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.571/ 2711
20E_0904	Select Input Register (IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.572/ 2712
20E_0908	Select Input Register (IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT)	32	R/W	0000_0000h	36.4.573/ 2713
20E_090C	Select Input Register (IOMUXC_SDMA_EVENTS14_SELECT_INPUT)	32	R/W	0000_0000h	36.4.574/ 2714
20E_0910	Select Input Register (IOMUXC_SDMA_EVENTS15_SELECT_INPUT)	32	R/W	0000_0000h	36.4.575/ 2715
20E_0914	Select Input Register (IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT)	32	R/W	0000_0000h	36.4.576/ 2715
20E_0918	Select Input Register (IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT)	32	R/W	0000_0000h	36.4.577/ 2716
20E_091C	Select Input Register (IOMUXC_UART1_UART_RTS_B_SELECT_INPUT)	32	R/W	0000_0000h	36.4.578/ 2717
20E_0920	Select Input Register (IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT)	32	R/W	0000_0000h	36.4.579/ 2717
20E_0924	Select Input Register (IOMUXC_UART2_UART_RTS_B_SELECT_INPUT)	32	R/W	0000_0000h	36.4.580/ 2718
20E_0928	Select Input Register (IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT)	32	R/W	0000_0000h	36.4.581/ 2719
20E_092C	Select Input Register (IOMUXC_UART3_UART_RTS_B_SELECT_INPUT)	32	R/W	0000_0000h	36.4.582/ 2719
20E_0930	Select Input Register (IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT)	32	R/W	0000_0000h	36.4.583/ 2720
20E_0934	Select Input Register (IOMUXC_UART4_UART_RTS_B_SELECT_INPUT)	32	R/W	0000_0000h	36.4.584/ 2721
20E_0938	Select Input Register (IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT)	32	R/W	0000_0000h	36.4.585/ 2721
20E_093C	Select Input Register (IOMUXC_UART5_UART_RTS_B_SELECT_INPUT)	32	R/W	0000_0000h	36.4.586/ 2722
20E_0940	Select Input Register (IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT)	32	R/W	0000_0000h	36.4.587/ 2723
20E_0944	Select Input Register (IOMUXC_USB_OTG_OC_SELECT_INPUT)	32	R/W	0000_0000h	36.4.588/ 2724
20E_0948	Select Input Register (IOMUXC_USB_H1_OC_SELECT_INPUT)	32	R/W	0000_0000h	36.4.589/ 2725

Table continues on the next page...

IOMUXC memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_094C	Select Input Register (IOMUXC_USDHC1_WP_ON_SELECT_INPUT)	32	R/W	0000_0000h	36.4.590/ 2726

36.4.1 GPR (IOMUXC_GPR0)

Address: 20E_0000h base + 0h offset = 20E_0000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CLOCK_8_MUX_SEL		CLOCK_0_MUX_SEL		CLOCK_B_MUX_SEL		CLOCK_3_MUX_SEL		CLOCK_A_MUX_SEL		CLOCK_2_MUX_SEL		CLOCK_9_MUX_SEL		CLOCK_1_MUX_SEL	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	TX_CLK2_MUX_SEL		AUDIO_VIDEO_MUXING						DMAREQ_MUX_SEL7	DMAREQ_MUX_SEL6	DMAREQ_MUX_SEL5	DMAREQ_MUX_SEL4	DMAREQ_MUX_SEL3	DMAREQ_MUX_SEL2	DMAREQ_MUX_SEL1	DMAREQ_MUX_SEL0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_GPR0 field descriptions

Field	Description
31–30 CLOCK_8_MUX_SEL	Selects the source of asrck_clock_8 in ASRC according to clock muxing scheme 00 audmux.amx_output_rxclk_p7 muxed with ssi3.ssi_srck 01 audmux.amx_output_rxclk_p7 10 ssi3.ssi_srck 11 ssi3.rx_bit_clk
29–28 CLOCK_0_MUX_SEL	Selects the source of asrck_clock_0 in ASRC according to clock muxing scheme: 00 esai.ipp_ind_sckr muxed with esai.ipp_do_sckr 01 esai.ipp_ind_sckr 10 esai.ipp_do_sckr 11 Reserved
27–26 CLOCK_B_MUX_SEL	Selects the source of asrck_clock_b in ASRC according to clock muxing scheme: 00 audmux.amx_output_txclk_p7 muxed with ssi3.ssi_stck 01 audmux.amx_output_txclk_p7 10 ssi3.ssi_stck 11 ssi3.tx_bit_clk

Table continues on the next page...

IOMUXC_GPR0 field descriptions (continued)

Field	Description
25–24 CLOCK_3_MUX_SEL	Selects the source of asrck_clock_3 in ASRC according to clock muxing scheme: 00 audmux.amx_output_rxclk_p7 muxed with ssi3.ssi_srck 01 audmux.amx_output_rxclk_p7 10 ssi3.ssi_srck 11 ssi3.rx_bit_clk
23–22 CLOCK_A_MUX_SEL	Selects the source of asrck_clock_a in ASRC according to clock muxing scheme: 00 audmux.amx_output_txclk_p2 muxed with ssi2.ssi_stck 01 audmux.amx_output_txclk_p2 10 ssi2.ssi_stck 11 ssi2.tx_bit_clk
21–20 CLOCK_2_MUX_SEL	Selects the source of asrck_clock_2 in ASRC according to clock muxing scheme: 00 audmux.amx_output_rxclk_p2 muxed with ssi2.ssi_srck 01 audmux.amx_output_rxclk_p2 10 ssi2.ssi_srck 11 ssi2.rx_bit_clk
19–18 CLOCK_9_MUX_SEL	Selects the source of asrck_clock_9 in ASRC according to clock muxing scheme: 00 audmux.amx_output_txclk_p1 muxed with ssi1.ssi_stck 01 audmux.amx_output_txclk_p1 10 ssi1.ssi_stck 11 ssi1.tx_bit_clk
17–16 CLOCK_1_MUX_SEL	Selects the source of asrck_clock_1 in ASRC according to clock muxing scheme: 00 audmux.amx_output_rxclk_p1 muxed with ssi1.ssi_srck 01 audmux.amx_output_rxclk_p1 10 ssi1.ssi_srck 11 ssi1.rx_bit_clk
15–14 TX_CLK2_MUX_SEL	Selects the source of tx_clk2 in SPDIF according to ASRC clock muxing scheme: 00 same source as for asrc.asrck_clock_1 01 same source as for asrc.asrck_clock_2 10 same source as for asrc.asrck_clock_3 11 Reserved
13–8 AUDIO_VIDEO_MUXING	See section (TBD) for details.
7 DMAREQ_MUX_SEL7	Selects between two possible sources for SDMA_EVENT[14]: 0 spdif.drq0_spdif_b 1 iomux.sdma_ext_events[1] - External DMA Request via DISP0_DAT17 or GPIO_18
6 DMAREQ_MUX_SEL6	Selects between two possible sources for SDMA_EVENT[23]: 0 esai. 1 i2c3.ipi_int_b

Table continues on the next page...

IOMUXC_GPR0 field descriptions (continued)

Field	Description
5 DMAREQ_MUX_SEL5	Selects between two possible sources for SDMA_EVENT[9]: 0 ecspi4.ipd_req_cspi_rdma_b 1 epit2.ipi_int_epit_oc
4 DMAREQ_MUX_SEL4	Selects between two possible sources for SDMA_EVENT[10]: 0 ecspi4.ipd_req_cspi_tdma_b 1 i2c1.ipi_int_b
3 DMAREQ_MUX_SEL3	Selects between two possible sources for SDMA_EVENT[5]: 0 ecspi2.ipd_req_cspi_rdma_b 1 i2c1.ipi_int_b
2 DMAREQ_MUX_SEL2	Selects between two possible sources for SDMA_EVENT[4]: 0 ecspi1.ipd_req_cspi_tdma_b 1 i2c2.ipi_int_b
1 DMAREQ_MUX_SEL1	Selects between two possible sources for SDMA_EVENT[3]: 0 ecspi1.ipd_req_cspi_rdma_b 1 i2c3.ipi_int_b
0 DMAREQ_MUX_SEL0	Selects between two possible sources for SDMA_EVENT[2]: 0 ipu1.ipu_sdma_event 1 hdmi_tx.hdmi_tx_ophydtb[0]

36.4.2 GPR (IOMUXC_GPR1)

Address: 20E_0000h base + 4h offset = 20E_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	CFG_L1_CLK_REMOVAL_EN	APP_CLK_REQ_N	0	APP_REQ_EXIT_L1	APP_READY_ENTR_L23	APP_REQ_ENTR_L1	MIPI_COLOR_SW	MIPI_DPI_OFF	0	EXC_MON	ENET_CLK_SEL	MIPI_IPU2_MUX	MIPI_IPU1_MUX	TEST_POWERDOWN	IPU_VPU_MUX	REF_SSP_EN
W																
Reset	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	USB_EXP_MODE	SYS_INT	USB_OTG_ID_SEL	GINT	ADDRS3[10]		ACT_CS3	ADDRS2[10]		ACT_CS2	ADDRS1[10]		ACT_CS1	ADDRS0[10]		ACT_CS0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_GPR1 field descriptions

Field	Description
31 CFG_L1_CLK_REMOVAL_EN	PCIe_CTL (CLK LOGIC CONTROLLER GLUE) - Enable the reference clock removal in L1 state. This is a bit from application register.
30 APP_CLK_REQ_N	PCIe_CTL (CLK LOGIC CONTROLLER GLUE) - Indicates that application logic is ready to have reference clock removed.
29 Reserved	This read-only field is reserved and always has the value 0.
28 APP_REQ_EXIT_L1	PCIe_CTL - Application Request to Exit L1. Request from the application to exit ASPM state L1. 0 PCIe application request is not set 1 PCIe application request is set
27 APP_READY_ENTR_L23	PCIe_CTL - Application Ready to Enter L23. Indication from the application that it is ready to enter the L23 state. 0 PCIe application is not ready to enter L23 1 PCIe application is ready to enter L23
26 APP_REQ_ENTR_L1	PCIe_CTL - Application Request to Enter L1. Request from the application to enter ASPM state L1. 0 PCIe application request is not set 1 PCIe application request is set

Table continues on the next page...

IOMUXC_GPR1 field descriptions (continued)

Field	Description
25 MIPI_COLOR_SW	MIPI color switch control 0 MIPI color switch request is not set 1 MIPI color switch request is set
24 MIPI_DPI_OFF	MIPI DPI shutdown request 0 MIPI DPI shutdown request is not set 1 MIPI DPI shutdown request is set
23 Reserved	This read-only field is reserved and always has the value 0.
22 EXC_MON	Exclusive monitor response select of illegal command (of lal gaskets, except MMDC) 0 OKEY response 1 SLVError (default)
21 ENET_CLK_SEL	ENET TX reference clock 0 get enet tx reference clk from pad (external OSC for both external PHY and Internal Controller) 1 get enet tx reference clk from internal clock from anatop (loopback through pad), this clock also sent out to external PHY
20 MIPI_IPU2_MUX	MIPI sensor to IPU-2 mux control 0 Enable mipi to IPU2 CSI1 - virtual channel is fixed to 3. 1 Enable parallel interface to IPU2 CSI1.
19 MIPI_IPU1_MUX	MIPI sensor to IPU-1 mux control 0 Enable mipi to IPU1 CSI0 - virtual channel is fixed to 0. 1 Enable parallel interface to IPU1 CSI0.
18 TEST_POWERDOWN	PCIe_PHY - All Circuits Power-Down Control Function: Powers down all circuitry in the PHY for IDDQ testing. 0 Power down is not requested 1 Power down is requested
17 IPU_VPU_MUX	IPU-1/IPU-2 to VPU signals control. This control selects between IPU-1 and IPU-2 outputs that are going to the VPU (current buffer, new frame, end of line) 0 IPU-1 is selected 1 IPU-2 is selected
16 REF_SSP_EN	PCIe_PHY - Reference Clock Enable for SS function. Function: Enables the reference clock to the prescaler. The phy_ref_ssp_en signal must remain deasserted until the reference clock is running at the appropriate frequency, at which point phy_ref_ssp_en can be asserted. For lower power states, phy_ref_ssp_en can also be deasserted. 0 PCIe PHY reference clock is disabled 1 PCIe PHY reference clock is enabled
15 USB_EXP_MODE	USB Exposure mode 0 Exposure mode is disabled. 1 Exposure mode is enabled.
14 SYS_INT	PCIe_CTL - When SYS_INT goes from low to high, the core generates an Assert_INTx Message. When sys_int goes from high to low, the core generates a Deassert_INTx Message.

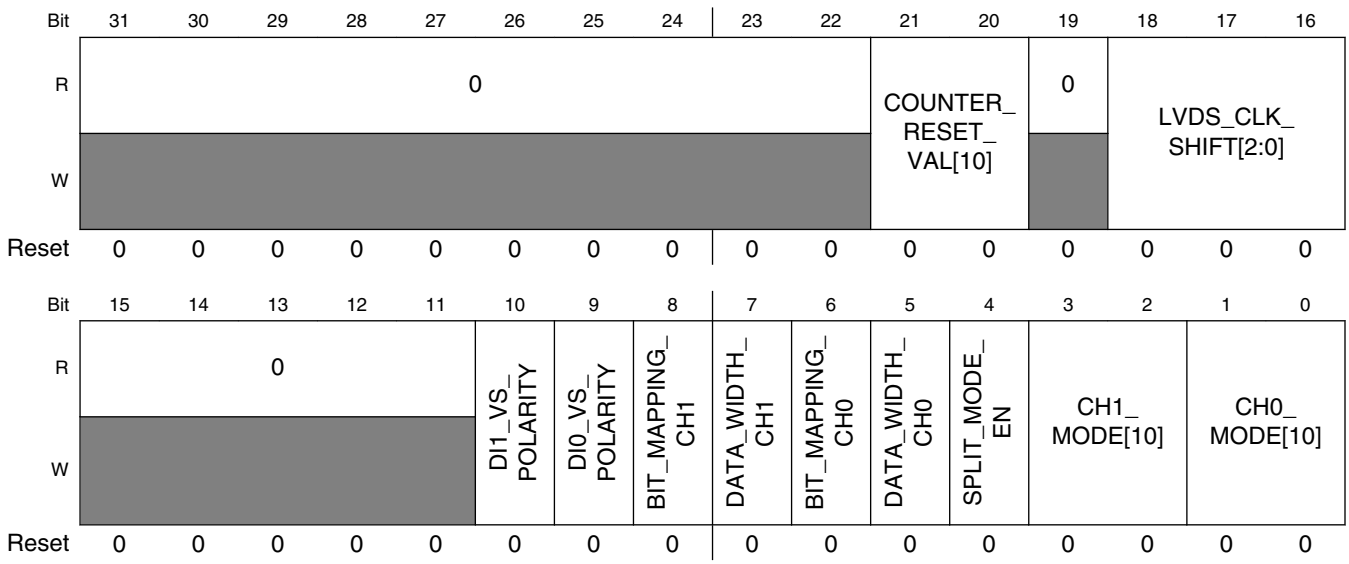
Table continues on the next page...

IOMUXC_GPR1 field descriptions (continued)

Field	Description
	0 PCIe system interrupt request is not asserted 1 PCIe system interrupt request is asserted
13 USB_OTG_ID_SEL	''usb_otg_id' pin iomux select control. (It functions as the 'daisy chain' mux control) 0 selects ENET_RX_ER 1 selects GPIO_1.
12 GINT	Global interrupt "0" bit (connected to ARM IRQ#0 and GPC) 0 Global interrupt request is not asserted 1 Global interrupt request is asserted
11–10 ADDRS3[10]	Active Chip Select and Address Space. Each of the ACT_CSx represents one of the four chip selects of the EIM. When ACT_CSx=1'b1, the corresponding chip select is active and has a valid address space according to its address space configuration determined by ADDRSx[10] bits ADDRSx[10] is setting the space for each chip select which is active. The address space of the first active chip select must be the largest one, the following active chip select address spaces may be equal or smaller. Total address space size is 128 MByte. The supported configurations are: CS0(128M), CS1 (0M), CS2 (0M), CS3(0M) [default configuration] CS0(64M), CS1(64M), CS2(0M), CS3(0M) CS0(64M), CS1(32M), CS2(32M), CS3(0M) CS0(32M), CS1(32M), CS2(32M), CS3(32M) Address Space Configuration options (ADDRSx[10]): 00 32 MByte 01 64 MByte 10 128 MByte 11 Reserved
9 ACT_CS3	See description for ADDRS3[10]
8–7 ADDRS2[10]	See description for ADDRS3[10]
6 ACT_CS2	See description for ADDRS3[10]
5–4 ADDRS1[10]	See description for ADDRS3[10]
3 ACT_CS1	See description for ADDRS3[10]
2–1 ADDRS0[10]	See description for ADDRS3[10]
0 ACT_CS0	See description for ADDRS3[10]

36.4.3 GPR (IOMUXC_GPR2)

Address: 20E_0000h base + 8h offset = 20E_0008h



IOMUXC_GPR2 field descriptions

Field	Description
31–22 Reserved	This read-only field is reserved and always has the value 0.
21–20 COUNTER_ RESET_VAL[10]	Reset value for the LDB counter which determines when the shift registers are loaded with data. **NOTE:** Used for debug purposes only. In normal functional operation must be '00' 00 Reset value is 5 01 Reset value is 3 10 Reset value is 4 11 Reset value is 6
19 Reserved	This read-only field is reserved and always has the value 0.
18–16 LVDS_CLK_ SHIFT[2:0]	Shifts the LVDS output clock in relation to the data. **NOTE:** Used for debug purposes only. In normal functional operation must be '000' 000 Output clock is '1100011' (normal operation) 001 Output clock is '1110001' 010 Output clock is '1111000' 011 Output clock is '1000111' 100 Output clock is '0001111' 101 Output clock is '0011111' 110 Output clock is '0111100' 111 Output clock is '1100011'
15–11 Reserved	This read-only field is reserved and always has the value 0.

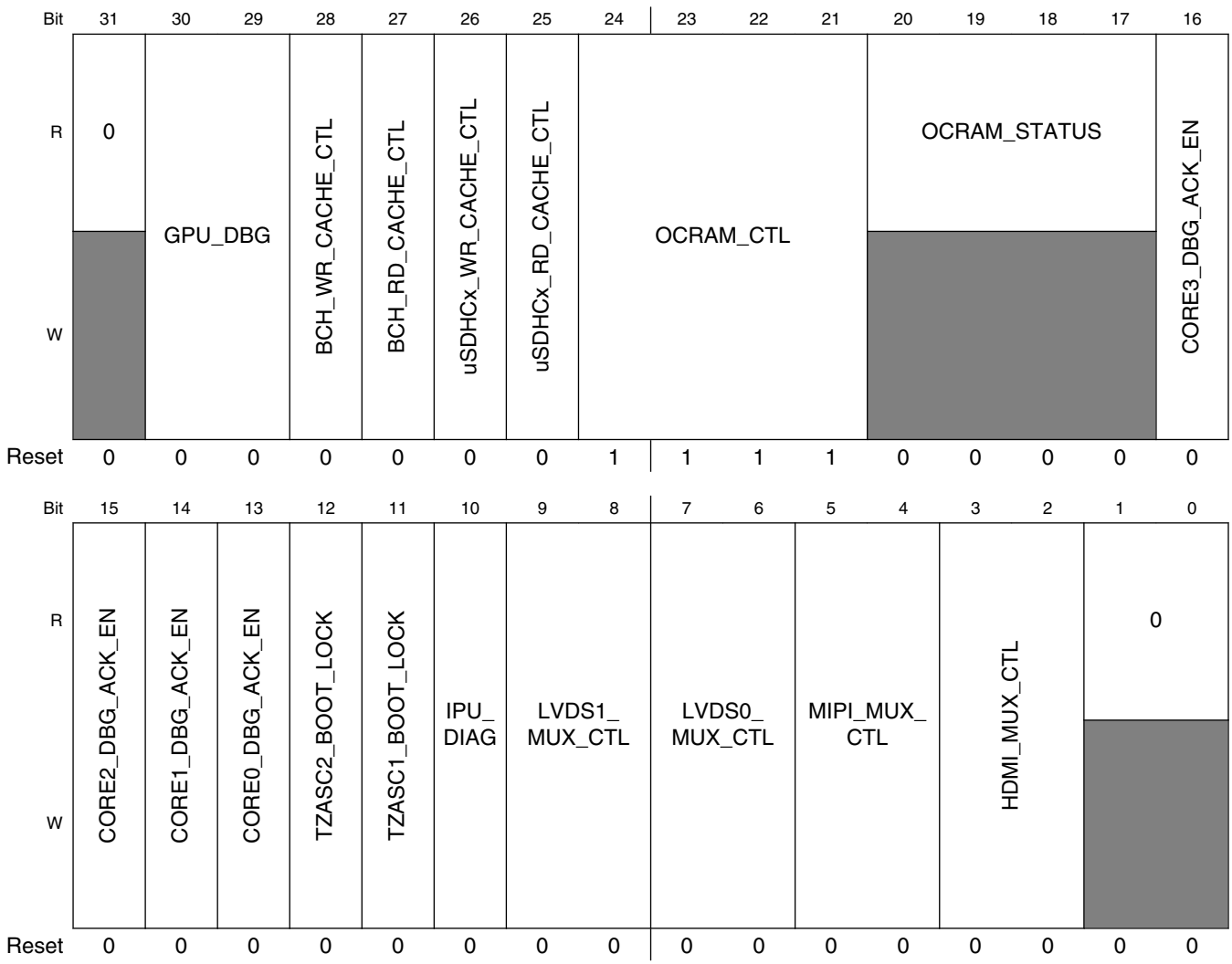
Table continues on the next page...

IOMUXC_GPR2 field descriptions (continued)

Field	Description
10 DI1_VS_ POLARITY	Vsync polarity for IPU's DI1 interface. 0 ipu_di1_vsync is active high. 1 ipu_di1_vsync is active low.
9 DI0_VS_ POLARITY	Vsync polarity for IPU's DI0 interface. 0 ipu_di0_vsync is active high. 1 ipu_di0_vsync is active low.
8 BIT_MAPPING_ CH1	Data mapping for LVDS channel 1. 0 Use SPWG standard. 1 Use JEIDA standard.
7 DATA_WIDTH_ CH1	Data width for LVDS channel 1. **NOTE:** This bit must be set when using JEIDA standard (bit_mapping_ch1 is set) 0 Data width is 18 bits wide (lvds1_tx3 is not used) 1 Data width is 24 bits wide.
6 BIT_MAPPING_ CH0	Data mapping for LVDS channel 0. 0 Use SPWG standard. 1 Use JEIDA standard.
5 DATA_WIDTH_ CH0	Data width for LVDS channel 0. **NOTE:** This bit must be set when using JEIDA standard (bit_mapping_ch0 is set) 0 Data width is 18 bits wide (lvds0_tx3 is not used) 1 Data width is 24 bits wide.
4 SPLIT_MODE_ EN	Enable split mode. 0 Split mode is disabled. 1 Split mode is enabled. In this mode both channels should be enabled and working with the same DI (ch0_mode and ch1_mode should both be either '01' or '11')
3–2 CH1_MODE[10]	LVDS channel 1 operation mode 00 Channel disabled. 01 Channel enabled, routed to DI0 10 Channel disabled. 11 Channel enabled, routed to DI1.
CH0_MODE[10]	LVDS channel 0 operation mode 00 Channel disabled. 01 Channel enabled, routed to DI0 10 Channel disabled. 11 Channel enabled, routed to DI1.

36.4.4 GPR (IOMUXC_GPR3)

Address: 20E_0000h base + Ch offset = 20E_000Ch



IOMUXC_GPR3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 GPU_DBG	GPU debug busses to IOMUX 00 GPU3D 01 GPU2D 10 OpenVG 11 Reserved

Table continues on the next page...

IOMUXC_GPR3 field descriptions (continued)

Field	Description
28 BCH_WR_ CACHE_CTL	Control BCH block cacheable attribute of AXI write transactions [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
27 BCH_RD_ CACHE_CTL	Control BCH block cacheable attribute of AXI read transactions [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.
26 uSDHCx_WR_ CACHE_CTL	Control uSDHCx [1-4] blocks cacheable attribute of AXI write transactions [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
25 uSDHCx_RD_ CACHE_CTL	Control uSDHCx [1-4] blocks cacheable attribute of AXI read transactions [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.
24–21 OCRAM_CTL	OCRAM_CTL[24] write address pipeline control bit. When this feature is enabled, the write address from the AXI master would be delayed 1 cycle before it can be accepted by the on-chip RAM. This can avoid any setup time issue for the write access on the memory cell at high frequency. Enabling this feature would cost at most 1 more clock cycle for each AXI write transaction, i.e., at most 1 more clock cycle for each write burst with multiple beats of data. When this feature is disabled, the write address from the AXI master can be accepted by the on-chip RAM without delay, and data can be written to memory at this cycle (if no other access and write data is also ready at this cycle). 0 write address pipeline is disabled 1 write address pipeline is enabled OCRAM_CTL[23] - write data pipeline control bit When this feature is enabled, the write data from the AXI master would be delayed 1 cycle before it can be accepted by the on-chip RAM. This can avoid any setup time issue for the write access on the memory cell at high frequency. Enabling this feature would cost at most 1 more clock cycle for each AXI write transaction, i.e., at most 1 more clock cycle for each write burst with multiple beats of data. When this feature is disabled, the write data from the AXI master can be accepted by the on-chip RAM without delay, and data can be written to memory at this cycle (if no other access and write address is also ready at this cycle). 0 write data pipeline is disabled 1 write data pipeline is enabled OCRAM_CTL[22] read address pipeline control bit. When this feature is enabled, the read address from the AXI master would be delayed 1 cycle before it can be accepted by the on-chip RAM. This can avoid any setup time issue for the read access on the memory cell at high frequency. Enabling this feature would cost at most 1 more clock cycle for each AXI read transaction, i.e., at most 1 more clock cycle for each read burst with multiple beats of data. When this feature is disabled, the read address from the AXI master can be accepted by the on-chip RAM without delay, and data can become ready for master at next clock cycle (if no other access and no read data wait). 0 read address pipeline is disabled 1 read address pipeline is enabled OCRAM_CTL[21] - read data wait state control bit

Table continues on the next page...

IOMUXC_GPR3 field descriptions (continued)

Field	Description
	When the read data wait state is enabled, it will cost 2 cycles for each read access, (each beat of a read burst). This can avoid the potential timing problem caused by the relatively longer memory access time at higher frequency. When this feature is disabled, it only costs 1 clock cycle to finish a read transaction, i.e., get read data back in the next cycle of read request becomes valid on the bus. 0 read data pipeline is disabled 1 read data pipeline is enabled
20–17 OCRAM_ STATUS	This field shows the OCRAM pipeline settings status, controlled by OCRAM_CTL[24:21] bits respectively. When the control bit is changed, the corresponding status bit goes high and keeps high until this new configuration is applied the internal logic. This provides a way for software to detect that the configuration has become valid. The suggested flow for changing the configuration in software is: • set/clear the control bit • poll the status bit until it goes to 0 OCRAM_STATUS[17] shows the write address pipeline status. This bit value reflects the propagation of the respective control bit to OCRAM memory. OCRAM_STATUS[18] shows the write data pipeline status. This bit value reflects the propagation of the respective control bit to OCRAM memory. OCRAM_STATUS[19] shows the read address pipeline status. This bit value reflects the propagation of the respective control bit to OCRAM memory. OCRAM_STATUS[20] shows the read data pipeline status. This bit value reflects the propagation of the respective control bit to OCRAM memory. 0 read data pipeline configuration valid 1 read data pipeline control bit changed
16 CORE3_DBG_ ACK_EN	Mask control of Core 3 debug acknowledge to global debug acknowledge 0 Core 3 debug acknowledge is part of global acknowledge. 1 Core 3 debug acknowledge is masked by this bit, and it is not part of global acknowledge.
15 CORE2_DBG_ ACK_EN	Mask control of Core 2 debug acknowledge to global debug acknowledge 0 Core 2 debug acknowledge is part of global acknowledge. 1 Core 2 debug acknowledge is masked by this bit, and it is not part of global acknowledge.
14 CORE1_DBG_ ACK_EN	Mask control of Core 1 debug acknowledge to global debug acknowledge. 0 Core 1 debug acknowledge is part of global acknowledge. 1 Core 1 debug acknowledge is masked by this bit, and it is not part of global acknowledge.
13 CORE0_DBG_ ACK_EN	Mask control of Core 1 debug acknowledge to global debug acknowledge 0 Core 0 debug acknowledge is part of global acknowledge. 1 Core 0 debug acknowledge is masked by this bit, and it is not part of global acknowledge.
12 TZASC2_BOOT_ LOCK	TZASC-2 secure boot lock 0 secure boot lock is disabled. 1 secure boot lock is enabled
11 TZASC1_BOOT_ LOCK	TZASC-1 secure boot lock 0 secure boot lock is disabled. 1 secure boot lock is enabled

Table continues on the next page...

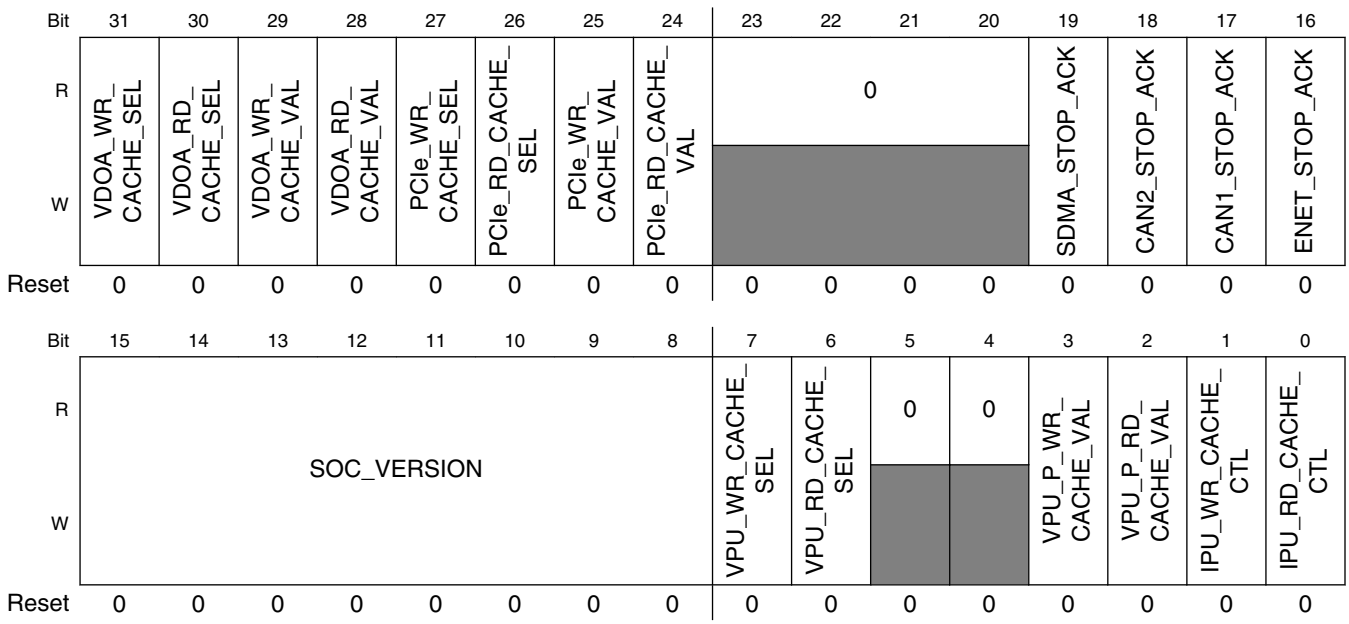
IOMUXC_GPR3 field descriptions (continued)

Field	Description
10 IPU_DIAG	IPU diagnostic debug bus mux 0 IPU1 diagnostic bus is selected 1 IPU2 diagnostic bus is selected
9–8 LVDS1_MUX_CTL	LVDS1 MUX control 00 LVDS1 source is IPU1 DI0 port 01 LVDS1 source is IPU1 DI1 port 10 LVDS1 source is IPU2 DI0 port 11 LVDS1 source is IPU2 DI1 port
7–6 LVDS0_MUX_CTL	LVDS0 MUX control 00 LVDS0 source is IPU1 DI0 port 01 LVDS0 source is IPU1 DI1 port 10 LVDS0 source is IPU2 DI0 port 11 LVDS0 source is IPU2 DI1 port
5–4 MIPI_MUX_CTL	MIPI MUX control 00 MIPI source is IPU1 DI0 port 01 MIPI source is IPU1 DI1 port 10 MIPI source is IPU2 DI0 port 11 MIPI source is IPU2 DI1 port
3–2 HDMI_MUX_CTL	HDMI MUX control 00 HDMI source is IPU1 DI0 port 01 HDMI source is IPU1 DI1 port 10 HDMI source is IPU2 DI0 port 11 HDMI source is IPU2 DI1 port
Reserved	This read-only field is reserved and always has the value 0.

1. Set of the cache bits, enable packet optimization through the bus system to DDR controller. The only side effect is that bus may change the nature of the accesses, which may lead to problems when accessing FIFO type address. In most typical cases, these bits should be set. For the GPU3D, GPU2D and OpenVG, such settings are possible through the IP programming model. For few peripherals, for these bits to take effect, it is required to also select set '1' to 'cache-mux' control bit.

36.4.5 GPR (IOMUXC_GPR4)

Address: 20E_0000h base + 10h offset = 20E_0010h



IOMUXC_GPR4 field descriptions

Field	Description
31 VDOA_WR_CACHE_SEL	This bit selects the cacheable attribute of VDOA AXI write transcations [1] 0 The write transactions cacheable attribute is driven by the VDOA core 1 The write transactions cacheable attribute is driven by VDOA_WR_CACHE_VAL.
30 VDOA_RD_CACHE_SEL	This bit selects the cacheable attribute of VDOA AXI read transcations) [1] 0 The read transaction cacheable attribute is driven by the VDOA core 1 The read transaction cacheable attribute is driven by VDOA_RD_CACHE_VAL.
29 VDOA_WR_CACHE_VAL	VDOA block cacheable attribute value of AXI write transactions The value of VDOA_WR_CACHE_VAL is affecting the transactions only if VDOA_WR_CACHE_SEL is set. [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
28 VDOA_RD_CACHE_VAL	VDOA block cacheable attribute value of AXI read transactions The value of VDOA_RD_CACHE_VAL is affecting the transactions only if VDOA_RD_CACHE_SEL is set. [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.
27 PCIe_WR_CACHE_SEL	This bit selects the cacheable attribute of PCIe AXI write transcations [1]

Table continues on the next page...

IOMUXC_GPR4 field descriptions (continued)

Field	Description
	0 The write transactions cacheable attribute is driven by the PCIe core 1 The write transactions cacheable attribute is driven by PCIe_WR_CACHE_VAL.
26 PCIe_RD_ CACHE_SEL	This bit selects the cacheable attribute of PCIe AXI read transcations) [1] 0 The read transaction cacheable attribute is driven by the PCIe core 1 The read transaction cacheable attribute is driven by PCIe_RD_CACHE_VAL.
25 PCIe_WR_ CACHE_VAL	PCIe block cacheable attribute value of AXI write transactions The value of PCIe_WR_CACHE_VAL is affecting the transactions only if PCIe_WR_CACHE_SEL is set. [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
24 PCIe_RD_ CACHE_VAL	PCIe block cacheable attribute value of AXI read transactions The value of PCIe_RD_CACHE_VAL is affecting the transactions only if PCIe_RD_CACHE_SEL is set. [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.
23–20 Reserved	This read-only field is reserved and always has the value 0.
19 SDMA_STOP_ ACK	SDMA stop acknowledge. This is status (read only) bit. 0 SDMA stop acknowledge is not asserted. 1 SDMA stop acknowledge is asserted, SDMA is in STOP mode.
18 CAN2_STOP_ ACK	CAN-2 stop acknowledge. This is status (read only) bit. 0 CAN-2 stop acknowledge is not asserted. 1 CAN-2 stop acknowledge is asserted, CAN-2 is in STOP mode.
17 CAN1_STOP_ ACK	CAN-1 stop acknowledge. This is status (read only) bit. 0 CAN-1 stop acknowledge is not asserted. 1 CAN-1 stop acknowledge is asserted, CAN-1 is in STOP mode.
16 ENET_STOP_ ACK	ENET stop acknowledge. This is status (read only) bit. 0 ENET stop acknowledge is not asserted. 1 ENET stop acknowledge is asserted, ENET is in STOP mode.
15–8 SOC_VERSION	This is status (read only) field.
7 VPU_WR_ CACHE_SEL	This bit selects the cacheable attribute of VPU AXI write transcations (both primary and secondary AXI buses) [1] 0 The write transactions cacheable attribute is driven by the VPU core 1 The write transactions cacheable attribute is driven by VPU_SEC_WR_CACHE_VAL for secondary bus and VPU_P_WR_CACHE_VAL for primary bus.
6 VPU_RD_ CACHE_SEL	This bit selects the cacheable attribute of VPU AXI read transcations (both primary and secondary AXI buses) [1] 0 The read transaction cacheable attribute is driven by the VPU core 1 The read transaction cacheable attribute is driven by VPU_SEC_RD_CACHE_VAL for secondary bus and VPU_P_RD_CACHE_VAL for primary bus.

Table continues on the next page...

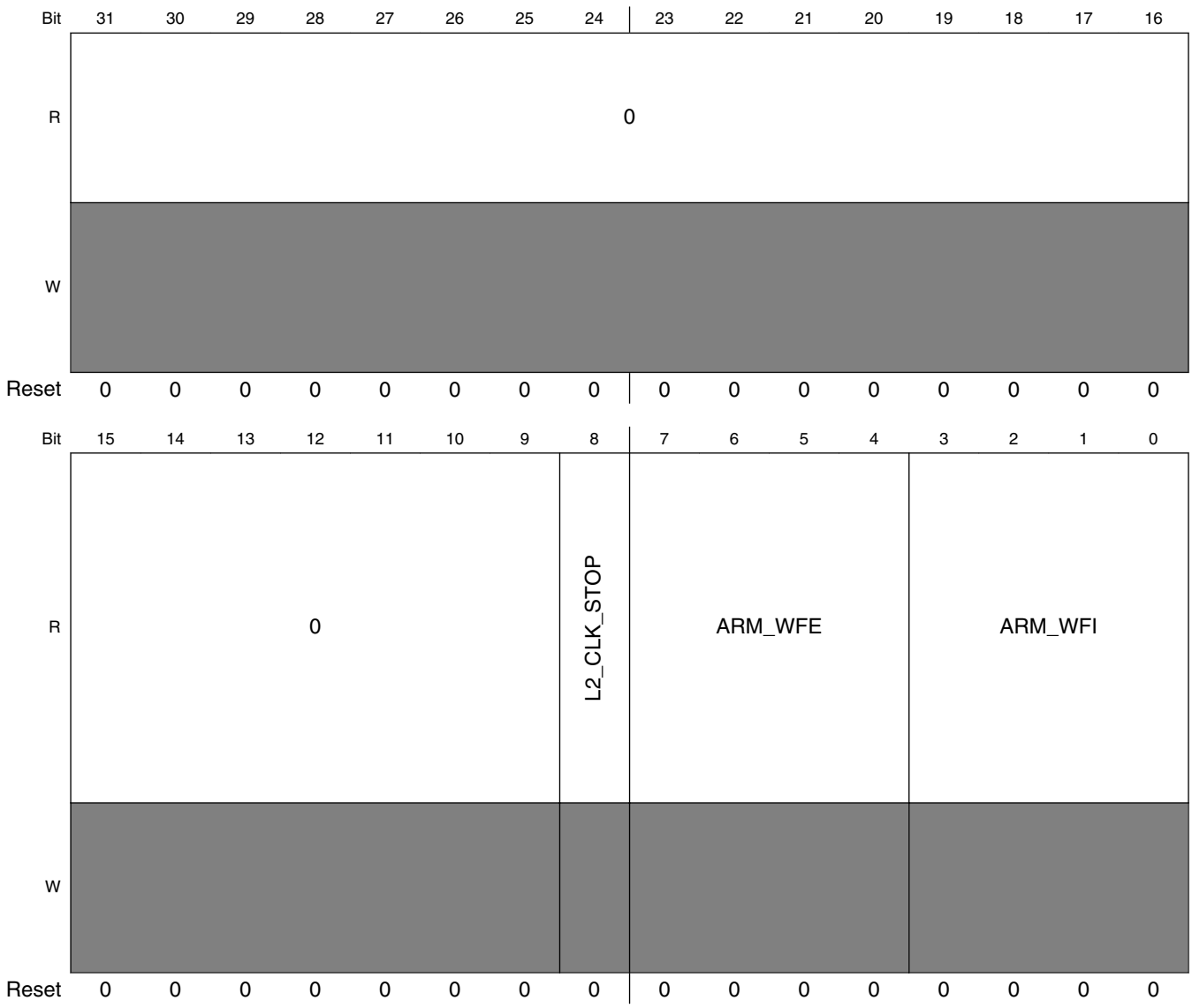
IOMUXC_GPR4 field descriptions (continued)

Field	Description
5 Reserved	This read-only field is reserved and always has the value 0.
4 Reserved	This read-only field is reserved and always has the value 0.
3 VPU_P_WR_CACHE_VAL	VPU (primary bus) block cacheable attribute value of AXI write transactions The value of VPU_P_WR_CACHE_VAL is affecting the transactions only if VPU_WR_CACHE_SEL is set. [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
2 VPU_P_RD_CACHE_VAL	VPU (primary bus) block cacheable attribute value of AXI read transactions The value of VPU_P_RD_CACHE_VAL is affecting the transactions only if VPU_RD_CACHE_SEL is set. [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.
1 IPU_WR_CACHE_CTL	Control IPU-1 and IPU-2 block cacheable attribute of AXI write transactions [1] 0 Cacheable attribute is off for write transactions. 1 Cacheable attribute is on for write transactions.
0 IPU_RD_CACHE_CTL	Control IPU-1 and IPU-2 block cacheable attribute of AXI read transactions [1] 0 Cacheable attribute is off for read transactions. 1 Cacheable attribute is on for read transactions.

1. Set of the cache bits, enable packet optimization through the bus system to DDR controller. The only side effect is that bus may change the nature of the accesses, which may lead to problems when accessing FIFO type address. In most typical cases, these bits should be set. For the GPU3D, GPU2D and OpenVG, such settings are possible through the IP programming model. For few peripherals, for these bits to take effect, it is required to also select set '1' to 'cache-mux' control bit.

36.4.6 GPR (IOMUXC_GPR5)

Address: 20E_0000h base + 14h offset = 20E_0014h



IOMUXC_GPR5 field descriptions

Field	Description
31–9 Reserved	This read-only field is reserved and always has the value 0.
8 L2_CLK_STOP	L2 cache clock stop indication (this is a status, read only bit) 0 L2 cache clock is running 1 L2 cache clock stopped
7–4 ARM_WFE	ARM WFE event out indication on WFE state of the cores (these are status, read only bits)

Table continues on the next page...

IOMUXC_GPR5 field descriptions (continued)

Field	Description
	0 ARM Core[GPR5-index - 4] is not in “Wait for Event” mode 1 ARM Core[GPR5-index - 4] is in “Wait for Event” mode
ARM_WFI	ARM WFI event out indicating on WFI state of the cores (these are status, read only bits) 0 ARM Core[GPR5-index] is not in “Wait for Interrupt” mode 1 ARM Core[GPR5-index] is in “Wait for Interrupt” mode

36.4.7 GPR (IOMUXC_GPR6)

Address: 20E_0000h base + 18h offset = 20E_0018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IPU1_ID11_RD_QoS				IPU1_ID10_RD_QoS				IPU1_ID01_RD_QoS				IPU1_ID00_RD_QoS				IPU1_ID11_WR_QoS				IPU1_ID10_WR_QoS				IPU1_ID01_WR_QoS				IPU1_ID00_WR_QoS			
Reset	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0

IOMUXC_GPR6 field descriptions

Field	Description
31–28 IPU1_ID11_RD_QoS	IPU1 Read AXI ID=11 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
27–24 IPU1_ID10_RD_QoS	IPU1 Read AXI ID=10 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
23–20 IPU1_ID01_RD_QoS	IPU1 Read AXI ID=01 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
19–16 IPU1_ID00_RD_QoS	IPU1 Read AXI ID=00 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
15–12 IPU1_ID11_WR_QoS	IPU1 Write AXI ID=11 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
11–8 IPU1_ID10_WR_QoS	IPU1 Write AXI ID=10 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111
7–4 IPU1_ID01_WR_QoS	IPU1 Write AXI ID=01 Quality of Service (QoS) priority 0xxx - 3 lsb’s will be passed as configured 1xxx - 1111

Table continues on the next page...

IOMUXC_GPR6 field descriptions (continued)

Field	Description
IPU1_ID00_WR_QoS	IPU1 Write AXI ID=00 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111

36.4.8 GPR (IOMUXC_GPR7)

Address: 20E_0000h base + 1Ch offset = 20E_001Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IPU2_ID11_RD_QoS				IPU2_ID10_RD_QoS				IPU2_ID01_RD_QoS				IPU2_ID00_RD_QoS				IPU2_ID11_WR_QoS				IPU2_ID10_WR_QoS				IPU2_ID01_WR_QoS				IPU2_ID00_WR_QoS			
Reset	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0

IOMUXC_GPR7 field descriptions

Field	Description
31–28 IPU2_ID11_RD_QoS	IPU2 Read AXI ID=11 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
27–24 IPU2_ID10_RD_QoS	IPU2 Read AXI ID=10 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
23–20 IPU2_ID01_RD_QoS	IPU2 Read AXI ID=01 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
19–16 IPU2_ID00_RD_QoS	IPU2 Read AXI ID=00 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
15–12 IPU2_ID11_WR_QoS	IPU2 Write AXI ID=11 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
11–8 IPU2_ID10_WR_QoS	IPU2 Write AXI ID=10 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
7–4 IPU2_ID01_WR_QoS	IPU2 Write AXI ID=01 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111
IPU2_ID00_WR_QoS	IPU2 Write AXI ID=00 Quality of Service (QoS) priority 0xxx - 3 lsb's will be passed as configured 1xxx - 1111

36.4.9 GPR (IOMUXC_GPR8)

Address: 20E_0000h base + 20h offset = 20E_0020h

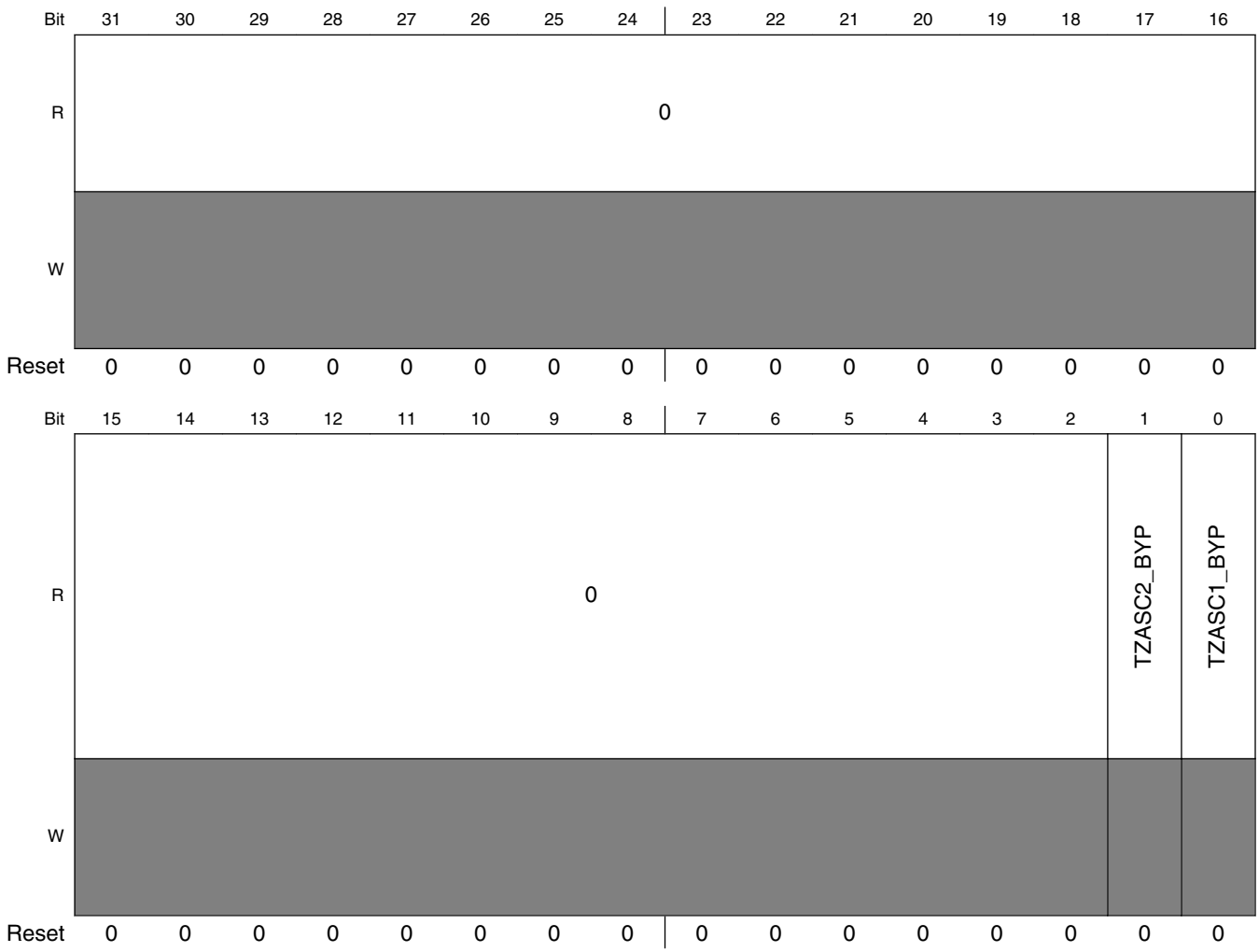
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PCS_TX_SWING_LOW							PCS_TX_SWING_FULL							PCS_TX_DEEMPH_GEN2_6DB						PCS_TX_DEEMPH_GEN2_3P5DB						PCS_TX_DEEMPH_GEN1					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_GPR8 field descriptions

Field	Description
31–25 PCS_TX_ SWING_LOW	PCIe_PHY - This static value sets the launch amplitude of the transmitter when pipe0_tx_swing is set to 1'b0 (default state). 7'hxx - TX launch amplitude swing_low value.
24–18 PCS_TX_ SWING_FULL	PCIe_PHY - This static value sets the Tx driver SWING_FULL value. 7'hxx - Gen2 TX SWING FULL value.
17–12 PCS_TX_ DEEMPH_ GEN2_6DB	PCIe_PHY - This static value sets the Tx driver de-emphasis value in the case where pipe0_tx_deemph is set to 1'b0 and the PHY is running at the Gen2 (6db) rate. 6'hxx - Gen2 (6db) De-emphasis value.
11–6 PCS_TX_ DEEMPH_ GEN2_3P5DB	PCIe_PHY - This static value sets the Tx driver de-emphasis value in the case where pipe0_tx_deemph is set to 1'b1 (the default setting) and the PHY is running at the Gen2 (3p5db) rate. 6'hxx - Gen2 De-emphasis value.
PCS_TX_ DEEMPH_GEN1	PCIe_PHY - This static value sets the Tx driver de-emphasis value in the case where pipe0_tx_deemph is set to 1'b1 (the default setting) and the PHY is running at the Gen1 rate. 6'hxx - Gen1 De-emphasis value.

36.4.10 GPR (IOMUXC_GPR9)

Address: 20E_0000h base + 24h offset = 20E_0024h

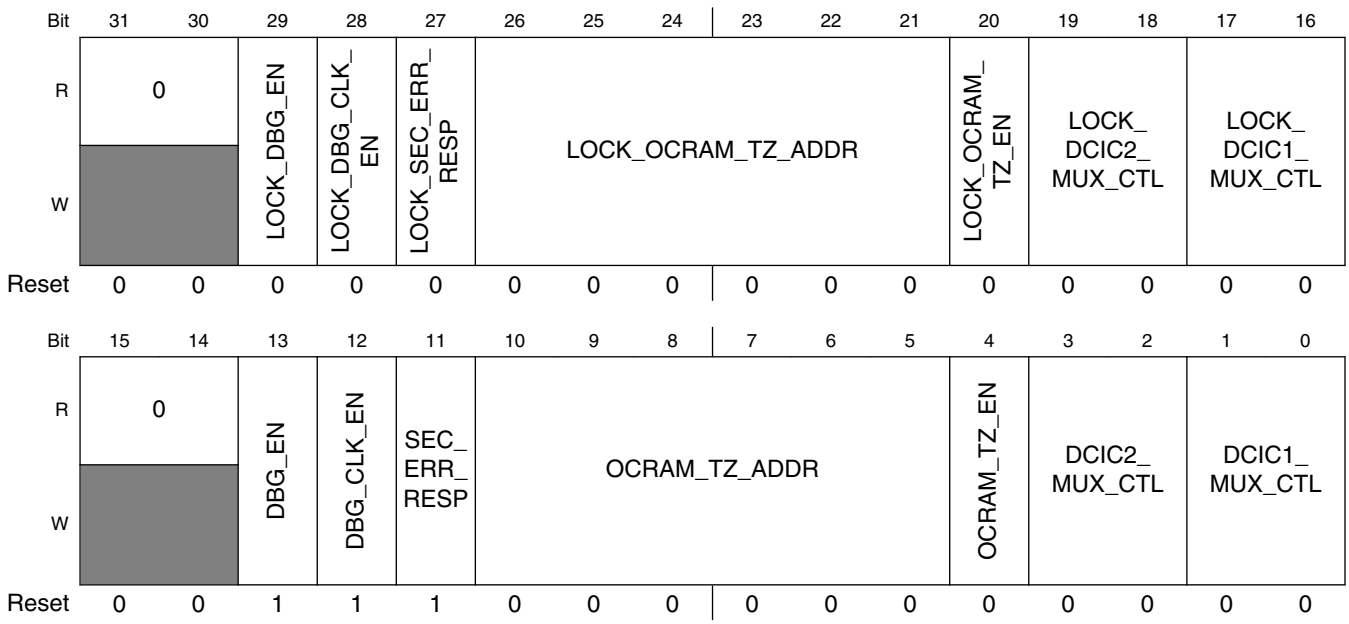


IOMUXC_GPR9 field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
1 TZASC2_BYP	TZASC-2 BYPASS MUX control 0 The TZASC-2 is bypassed and the transactions to DDR are not being checked. 1 The TZASC-2 is not bypassed and the transactions to DDR are being monitored / checked.
0 TZASC1_BYP	TZASC-1 BYPASS MUX control 0 The TZASC-1 is bypassed and the transactions to DDR are not being checked. 1 The TZASC-1 is not bypassed and the transactions to DDR are being monitored / checked.

36.4.11 GPR (IOMUXC_GPR10)

Address: 20E_0000h base + 28h offset = 20E_0028h



IOMUXC_GPR10 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
29 LOCK_DBG_EN	Lock DBG_EN field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
28 LOCK_DBG_CLK_EN	Lock DBG_CLK_EN field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
27 LOCK_SEC_ERR_RESP	Lock SEC_ERR_RESP field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
26–21 LOCK_OCRAM_TZ_ADDR	Lock OCRAM_TZ_ADDR field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
20 LOCK_OCRAM_TZ_EN	Lock OCRAM_TZ_EN field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)

Table continues on the next page...

IOMUXC_GPR10 field descriptions (continued)

Field	Description
19–18 LOCK_DCIC2_ MUX_CTL	Lock DCIC2_MUX_CTL field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
17–16 LOCK_DCIC1_ MUX_CTL	Lock DCIC1_MUX_CTL field for changes. This is a sticky field, once set it can't be cleared (only by reset). 0 Field is not locked 1 Field is locked (read access only)
15–14 Reserved	This read-only field is reserved and always has the value 0.
13 DBG_EN	ARM non secure (non-invasive) debug enable 0 Debug turned off. 1 Debug enabled (default).
12 DBG_CLK_EN	ARM Debug clock enable 0 Debug turned off. 1 Debug enabled (default).
11 SEC_ERR_ RESP	Security error response enable for all security gaskets (on both AHB and AXI busses) 0 OKEY response 1 SLVError (default)
10–5 OCRAM_TZ_ ADDR	OCRAM TrustZone (TZ) start address. This is the start address of the secure memory region within the OCRAM memory space is 4KB granularity. The start address affects the OCRAM transactions only if OCRAM_TZ_EN bit is set. The OCRAM TZ ENDADDR is not configurable and is set to the end of OCRAM memory space.
4 OCRAM_TZ_EN	OCRAM TrustZone (TZ) enable. 0 The TrustZone feature is disabled. Entire OCRAM space is available for all access types (secure/non-secure/user/supervisor). 1 The TrustZone feature is enabled. Access to address in the range specified by [ENDADDR:STARTADDR] follows the execution mode access policy described in CSU chapter.
3–2 DCIC2_MUX_ CTL	DCIC-2 MUX control 00 DCIC-2 source is IPU1 DI1 port 01 DCIC-2 source is LVDS0 10 DCIC-2 source is LVDS1 11 DCIC-2 source is MIPI DPI
DCIC1_MUX_ CTL	DCIC-1 MUX control 00 DCIC-1 source is IPU1 or IPU2 DI0 port 01 DCIC-1 source is LVDS0 10 DCIC-1 source is LVDS1 11 DCIC-1 source is HDMI

36.4.12 GPR (IOMUXC_GPR11)

Address: 20E_0000h base + 2Ch offset = 20E_002Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															0
W																
Reset	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_GPR11 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 Reserved	This read-only field is reserved and always has the value 0.
15–1 Reserved	This read-only field is reserved and always has the value 0.
0 Reserved	This read-only field is reserved and always has the value 0.

36.4.13 GPR (IOMUXC_GPR12)

Address: 20E_0000h base + 30h offset = 20E_0030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0				ARMP_IPG_CLK_EN	ARMP_AHB_CLK_EN	ARMP_ATB_CLK_EN	ARMP_APB_CLK_EN	PCIe_CTL_7			DIA_STATUS_BUS_SELECT				APPS_PM_XMT_TURNOFF
W																
Reset	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DEVICE_TYPE				APP_INIT_RST	APP_LTSSM_ENABLE	APPS_PM_XMT_PME	LOS_LEVEL					uSDHC_DBG_MUX		0	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_GPR12 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27 ARMP_IPG_ CLK_EN	ARM platform IPG clock enable 0 IPG clock is not running (gated). 1 IPG clock is running (enabled).
26 ARMP_AHB_ CLK_EN	ARM platform AHB clock enable 0 IPG clock is not running (gated). 1 IPG clock is running (enabled).
25 ARMP_ATB_ CLK_EN	ARM platform ATB clock enable 0 IPG clock is not running (gated). 1 IPG clock is running (enabled).
24 ARMP_APB_ CLK_EN	ARM platform APB clock enable 0 IPG clock is not running (gated). 1 IPG clock is running (enabled).
23–21 PCIe_CTL_7	PCIe control of diagnostic bus select (Drive 'cxpl_diag_ctrl' PCI controller input)
20–17 DIA_STATUS_ BUS_SELECT	PCIe_CTL - used for debug to select what part of diag_status_bus will be reflected on the 32 bits of the iomux
16 APPS_PM_ XMT_TURNOFF	PCIe_CTL - Request from the application to generate a PM_Turn_Off Message.
15–12 DEVICE_TYPE	PCIe_CTL - Device/Port Type. Indicates the specific type of this PCI Express Function (EP or RC) DEVICE_TYPE field values 0011-1111 are reserved. 0000 **PCIE_EP** — EP Mode 0010 **PCIE_RC** — RC Mode
11 APP_INIT_RST	PCIe_CTL - Request from the application to send a Hot Reset to the downstream device.
10 APP_LTSSM_ ENABLE	PCIe_CTL - Driven low by the application after reset to hold the LTSSM in the Detect state until the application is ready. When the application has finished initializing the core configuration registers, it asserts app_ltssm_enable to allow the LTSSM to continue Link establishment. 0 Application is not ready. 1 Application is ready.
9 APPS_PM_ XMT_PME	PCIe_CTL - Wake Up. Used by application logic to wake up the PMC state machine from a D1, D2 or D3 power state. Upon wake-up, the core sends a PM_PME Message
8–4 LOS_LEVEL	PCIe_PHY - Loss-of-Signal Detector Sensitivity Level Control Function: Sets the sensitivity level for the Loss-of-Signal detector. This signal must be set to 0x9
3–2 uSDHC_DBG_ MUX	uSDHC debug bus IO mux control '00' - uSDHC1 debug '01' - uSDHC2 debug

Table continues on the next page...

IOMUXC_GPR12 field descriptions (continued)

Field	Description
	'10' - uSDHC3 debug '11' - uSDHC4 debug
Reserved	This read-only field is reserved and always has the value 0.

36.4.14 GPR (IOMUXC_GPR13)

SATA_PHY_6	PHUG	FRUG	fast_startup	Frequency Tolerance (ppm)
000	1	1	None	780
001	2	2	None	780
010	1	4	None	6,250
011	2	4	None	6,250
1xx	Reserved			

Address: 20E_0000h base + 34h offset = 20E_0034h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	SDMA_STOP_REQ	CAN2_STOP_REQ	CAN1_STOP_REQ	ENET_STOP_REQ	SATA_PHY_8			SATA_PHY_7					SATA_PHY_6		
W																
Reset	0	0	0	0	0	1	0	1	1	0	0	1	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	SATA_SPEED	SATA_PHY_5	SATA_PHY_4			SATA_PHY_3			SATA_PHY_2						SATA_PHY_1	SATA_PHY_0
W																
Reset	0	0	1	0	0	1	0	0	1	1	0	0	0	1	0	0

IOMUXC_GPR13 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30 SDMA_STOP_REQ	SDMA stop request 0 Stop request off. 1 Stop request on.

Table continues on the next page...

IOMUXC_GPR13 field descriptions (continued)

Field	Description
29 CAN2_STOP_REQ	CAN2 stop request 0 Stop request off. 1 Stop request on.
28 CAN1_STOP_REQ	CAN1 stop request 0 Stop request off. 1 Stop request on.
27 ENET_STOP_REQ	ENET stop request 0 Stop request off. 1 Stop request on.
26–24 SATA_PHY_8	SATA _PHY Rx - Receiver Equalization control 000 0.5 dB 001 1.0 dB 010 1.5 dB 011 2.0 dB 100 2.5 dB 101 3.0 dB (default) 110 3.5 dB 111 4.0 dB
23–19 SATA_PHY_7	SATA PHY Rx - Loss of signal detector level. Below the recommended value are shown 10000 SATA1i 10000 SATA1m 11010 SATA1x 10010 SATA2i 10010 (default) SATA2m 11010 SATA2x
18–16 SATA_PHY_6	SATA PHY Rx - DPLL mode control, sets phase and frequency gain of receiver DPLL For bits encoding see GPR (IOMUXC_GPR13) below.
15 SATA_SPEED	Indicates SATA PHY speed mode 0 1.5 GHz 1 3.0 GHz
14 SATA_PHY_5	SATA PHY - Spread Spectrum Enable. Enables spead spectrum clock production. If the applied RefClk is already spread spectrum, this bit must be deasserted. 0 Spread Spectrum disabled 1 Spread spectrum enabled
13–11 SATA_PHY_4	SATA PHY -Transmit Attenuation control, provides discrete driver attenuation factors (from full driver level). 000 16/16 001 14/16 010 12/16 011 10/16

Table continues on the next page...

IOMUXC_GPR13 field descriptions (continued)

Field	Description
	100 9/16 (default) 101 8/16 110 Reserved 111 Reserved
10–7 SATA_PHY_3	SATA PHY Tx -Transmit Boost Control, ratio of drive level of transmission bit to non transmission bit. 0000 0dB 0001 0.37 dB 0010 0.74 dB 0011 1.11 dB 0100 1.48 dB 0101 1.85 dB 0110 2.22 dB 0111 2.59 dB 1000 2.96 dB 1001 3.33 dB (default) 1010 3.70 dB 1011 4.07 dB 1100 4.44 dB 1101 4.81 dB 1110 5.28 dB 1111 5.75 dB
6–2 SATA_PHY_2	SATA PHY - Transmit level settings. Fine resolution settings of transmit signal level, common to all lanes connected to one clock module. 00000 0.937 V 00001 0.947 V 00010 0.957 V 00011 0.966 V 00100 0.976 V 00101 0.986 V 00110 0.996 V 00111 1.005 V 01000 1.015 V 01001 1.025 V 01010 1.035 V 01011 1.045 V 01100 1.054 V 01101 1.064 V 01110 1.074 V 01111 1.084 V 10000 1.094 V 10001 1.104 V (default) 10010 1.113 V 10011 1.123 V 10100 1.133 V 10101 1.143 V

Table continues on the next page...

IOMUXC_GPR13 field descriptions (continued)

Field	Description
	10110 1.152 V 10111 1.162 V 11000 1.172 V 11001 1.182 V 11010 1.191 V 11011 1.201 V 11100 1.211 V 11101 1.221 V 11110 1.230 V 11111 1.240 V
1 SATA_PHY_1	SATA PHY internal PLL Reference Clock Enable 0 Disable the reference clock to the internal PLL of SATA PHY 1 Enable the reference clock to the internal PLL of SATA PHY
0 SATA_PHY_0	SATA PHY - Tx Edge rate control enables the SATA PHY to meet the edge rate requirements for all SATA variants 0 Fast edge rate 1 Medium edge rate

36.4.15 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1)

Address: 20E_0000h base + 4Ch offset = 20E_004Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_DAT1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA1 field descriptions (continued)

Field	Description
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: SD2_DAT1. NOTE: Pad SD2_DAT1 is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_DATA1. 001 **ALT1** — Select signal ECSPI5_SS0. - Configure register IOMUXC_ECSPI5_SS0_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal EIM_CS2_B. 011 **ALT3** — Select signal AUD4_TXFS. - Configure register IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal KEY_COL7. - Configure register IOMUXC_KEY_COL7_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO1_IO14.

36.4.16 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2)

Address: 20E_0000h base + 50h offset = 20E_0050h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_DAT2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA2 field descriptions (continued)

Field	Description
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: SD2_DAT2. NOTE: Pad SD2_DAT2 is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_DATA2. 001 **ALT1** — Select signal ECSPI5_SS1. - Configure register IOMUXC_ECSPI5_SS1_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal EIM_CS3_B. 011 **ALT3** — Select signal AUD4_TXD. - Configure register IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal KEY_ROW6. - Configure register IOMUXC_KEY_ROW6_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO1_IO13.

36.4.17 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0)

Address: 20E_0000h base + 54h offset = 20E_0054h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_DAT0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: SD2_DAT0.

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA0 field descriptions (continued)

Field	Description
	NOTE: Pad SD2_DAT0 is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_DATA0. 001 **ALT1** — Select signal ECSPI5_MISO. - Configure register IOMUXC_ECSPI5_MISO_SELECT_INPUT for mode ALT1. 011 **ALT3** — Select signal AUD4_RXD. - Configure register IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal KEY_ROW7. - Configure register IOMUXC_KEY_ROW7_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO1_IO15. 110 **ALT6** — Select signal DCIC2_OUT.

36.4.18 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC)

Address: 20E_0000h base + 58h offset = 20E_0058h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TXC. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: RGMII_TXC. NOTE: Pad RGMII_TXC is involved in Daisy Chain. 000 **ALT0** — Select signal USB_H2_DATA.

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_RGMII_TXC field descriptions (continued)

Field	Description
	001 **ALT1** — Select signal RGMII_TXC. 010 **ALT2** — Select signal SPDIF_EXT_CLK. - Configure register IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO6_IO19. 111 **ALT7** — Select signal XTALOSC_REF_CLK_24M.

36.4.19 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0)

Address: 20E_0000h base + 5Ch offset = 20E_005Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_TD0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TD0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_TD0. 000 **ALT0** — Select signal HSI_TX_READY. 001 **ALT1** — Select signal RGMII_TD0. 101 **ALT5** — Select signal GPIO6_IO20.

36.4.20 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1)

Address: 20E_0000h base + 60h offset = 20E_0060h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

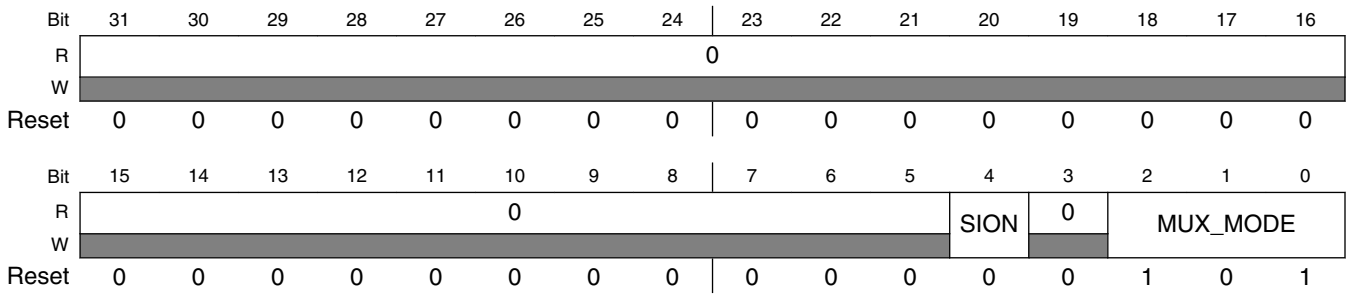
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_TD1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TD1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_TD1. 000 **ALT0** — Select signal HSI_RX_FLAG. 001 **ALT1** — Select signal RGMII_TD1. 101 **ALT5** — Select signal GPIO6_IO21.

36.4.21 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2)

Address: 20E_0000h base + 64h offset = 20E_0064h



IOMUXC_SW_MUX_CTL_PAD_RGMII_TD2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TD2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_TD2. 000 **ALT0** — Select signal HSI_RX_DATA. 001 **ALT1** — Select signal RGMII_TD2. 101 **ALT5** — Select signal GPIO6_IO22.

36.4.22 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3)

Address: 20E_0000h base + 68h offset = 20E_0068h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

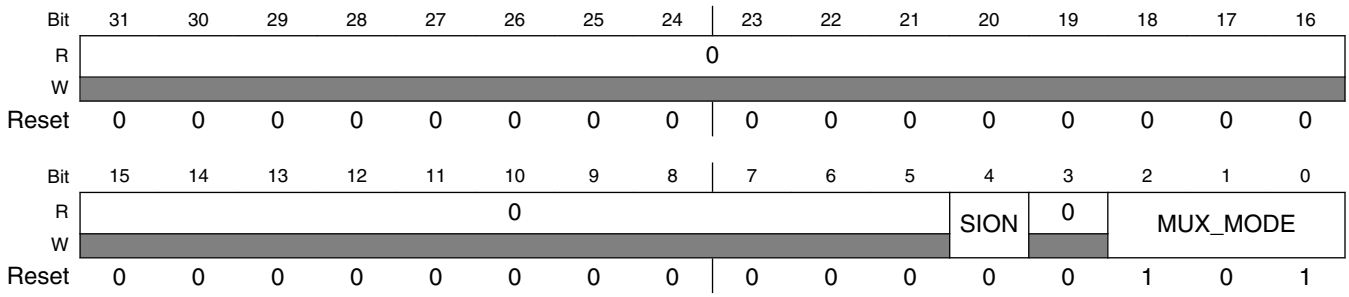
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_TD3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TD3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_TD3. 000 **ALT0** — Select signal HSI_RX_WAKE. 001 **ALT1** — Select signal RGMII_TD3. 101 **ALT5** — Select signal GPIO6_IO23.

36.4.23 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL)

Address: 20E_0000h base + 6Ch offset = 20E_006Ch



IOMUXC_SW_MUX_CTL_PAD_RGMII_RX_CTL field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RX_CTL. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RX_CTL. NOTE: Pad RGMII_RX_CTL is involved in Daisy Chain. 000 **ALT0** — Select signal USB_H3_DATA. 001 **ALT1** — Select signal RGMII_RX_CTL. - Configure register IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO24.

36.4.24 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0)

Address: 20E_0000h base + 70h offset = 20E_0070h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_RD0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RD0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RD0. NOTE: Pad RGMII_RD0 is involved in Daisy Chain. 000 **ALT0** — Select signal HSI_RX_READY. 001 **ALT1** — Select signal RGMII_RD0. - Configure register IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO25.

36.4.25 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL)

Address: 20E_0000h base + 74h offset = 20E_0074h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_TX_CTL field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_TX_CTL. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: RGMII_TX_CTL. NOTE: Pad RGMII_TX_CTL is involved in Daisy Chain. 000 **ALT0** — Select signal USB_H2_STROBE. 001 **ALT1** — Select signal RGMII_TX_CTL. 101 **ALT5** — Select signal GPIO6_IO26. 111 **ALT7** — Select signal ENET_REF_CLK. - Configure register IOMUXC_ENET_REF_CLK_SELECT_INPUT for mode ALT7.

36.4.26 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1)

Address: 20E_0000h base + 78h offset = 20E_0078h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

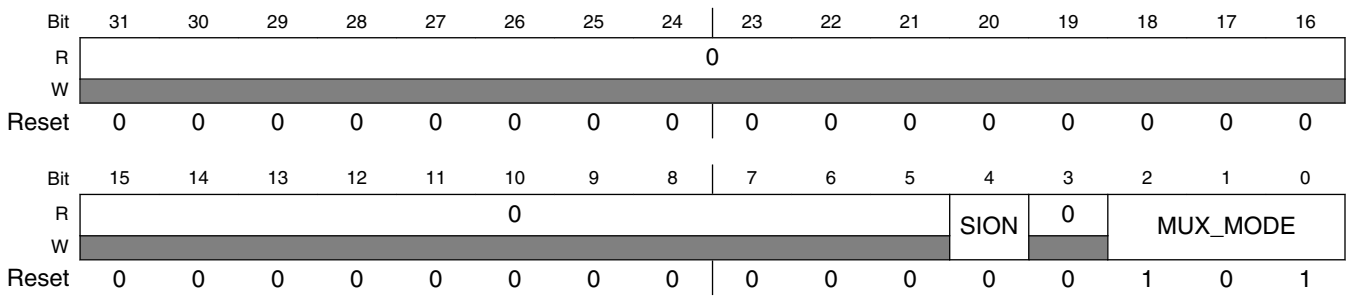
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_RD1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RD1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RD1. NOTE: Pad RGMII_RD1 is involved in Daisy Chain. 000 **ALT0** — Select signal HSI_TX_FLAG. 001 **ALT1** — Select signal RGMII_RD1. - Configure register IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO27.

36.4.27 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2)

Address: 20E_0000h base + 7Ch offset = 20E_007Ch



IOMUXC_SW_MUX_CTL_PAD_RGMII_RD2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RD2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RD2. NOTE: Pad RGMII_RD2 is involved in Daisy Chain. 000 **ALT0** — Select signal HSI_TX_DATA. 001 **ALT1** — Select signal RGMII_RD2. - Configure register IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO28.

36.4.28 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3)

Address: 20E_0000h base + 80h offset = 20E_0080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

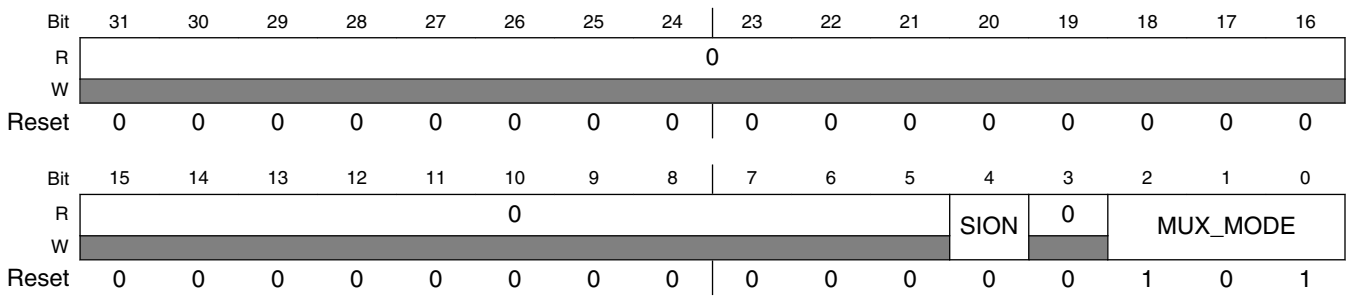
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_RGMII_RD3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RD3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RD3. NOTE: Pad RGMII_RD3 is involved in Daisy Chain. 000 **ALT0** — Select signal HSI_TX_WAKE. 001 **ALT1** — Select signal RGMII_RD3. - Configure register IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO29.

36.4.29 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC)

Address: 20E_0000h base + 84h offset = 20E_0084h

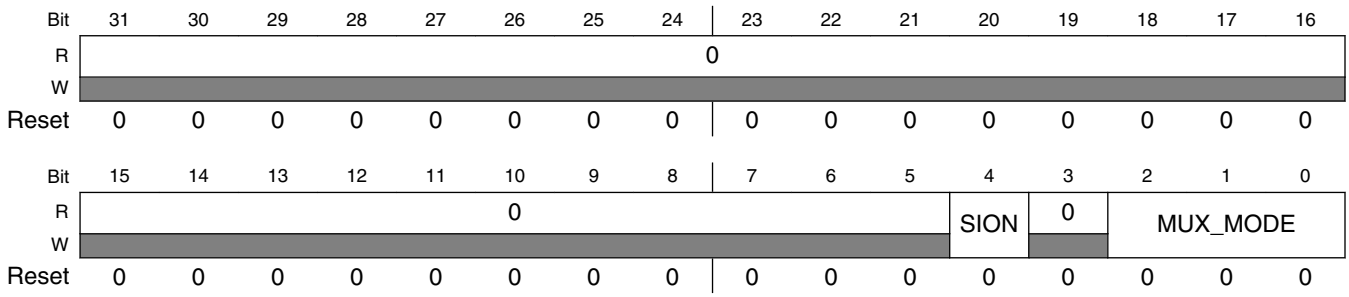


IOMUXC_SW_MUX_CTL_PAD_RGMII_RXC field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad RGMII_RXC. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: RGMII_RXC. NOTE: Pad RGMII_RXC is involved in Daisy Chain. 000 **ALT0** — Select signal USB_H3_STROBE. 001 **ALT1** — Select signal RGMII_RXC. - Configure register IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO30.

36.4.30 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25)

Address: 20E_0000h base + 88h offset = 20E_0088h

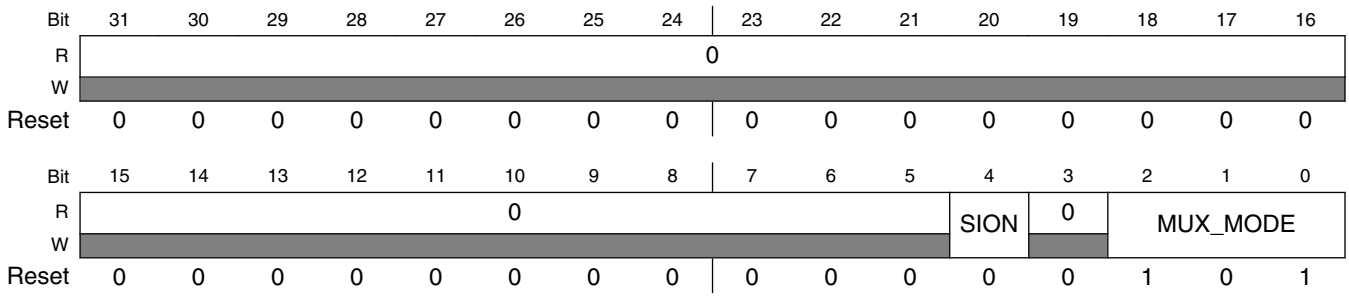


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR25 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A25. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_A25. NOTE: Pad EIM_A25 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR25. 001 **ALT1** — Select signal ECSPI4_SS1. 010 **ALT2** — Select signal ECSPI2_RDY. 011 **ALT3** — Select signal IPU1_DI1_PIN12. 100 **ALT4** — Select signal IPU1_DI0_D1_CS. 101 **ALT5** — Select signal GPIO5_IO02. 110 **ALT6** — Select signal HDMI_TX_CEC_LINE. - Configure register IOMUXC_HDMI_ICECIN_SELECT_INPUT for mode ALT6.

36.4.31 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B)

Address: 20E_0000h base + 8Ch offset = 20E_008Ch

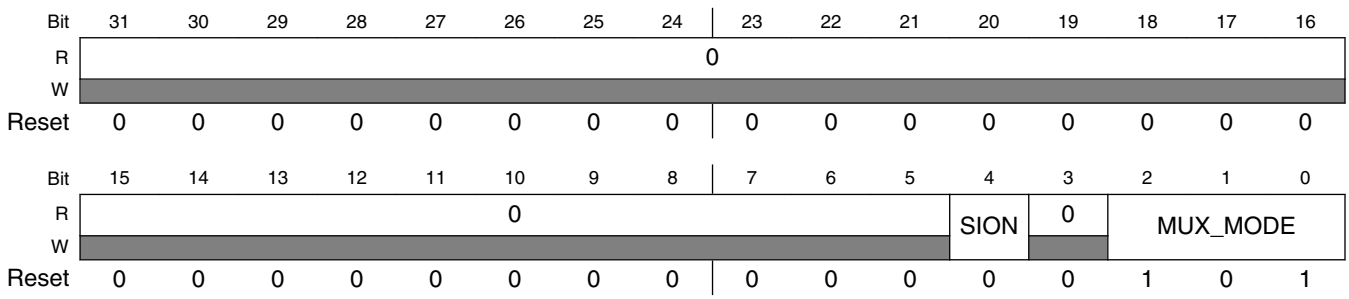


IOMUXC_SW_MUX_CTL_PAD_EIM_EB2_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_EB2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_EB2. NOTE: Pad EIM_EB2 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_EB2_B. 001 **ALT1** — Select signal ECSPI1_SS0. - Configure register IOMUXC_ECSPI1_SS0_SELECT_INPUT for mode ALT1. 011 **ALT3** — Select signal IPU2_CSI1_DATA19. - Configure register IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal HDMI_TX_DDC_SCL. - Configure register IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO2_IO30. 110 **ALT6** — Select signal I2C2_SCL. - Configure register IOMUXC_I2C2_SCL_IN_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal SRC_BOOT_CFG30.

36.4.32 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16)

Address: 20E_0000h base + 90h offset = 20E_0090h

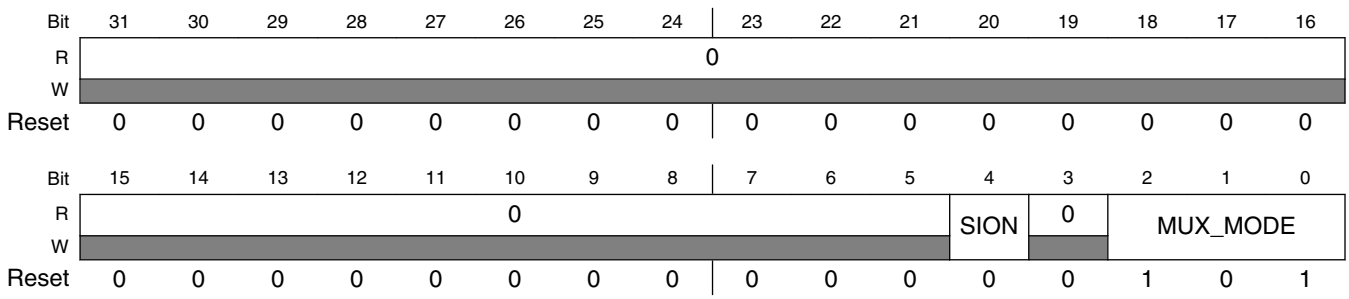


IOMUXC_SW_MUX_CTL_PAD_EIM_DATA16 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D16. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D16. NOTE: Pad EIM_D16 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA16. 001 **ALT1** — Select signal ECSPI1_SCLK. - Configure register IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal IPU1_DI0_PIN05. 011 **ALT3** — Select signal IPU2_CSI1_DATA18. - Configure register IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal HDMI_TX_DDC_SDA. - Configure register IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO16. 110 **ALT6** — Select signal I2C2_SDA. - Configure register IOMUXC_I2C2_SDA_IN_SELECT_INPUT for mode ALT6.

36.4.33 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17)

Address: 20E_0000h base + 94h offset = 20E_0094h



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA17 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D17. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D17. NOTE: Pad EIM_D17 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA17. 001 **ALT1** — Select signal ECSPI1_MISO. - Configure register IOMUXC_ECSPI1_MISO_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal IPU1_DI0_PIN06. 011 **ALT3** — Select signal IPU2_CSI1_PIXCLK. - Configure register IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal DCIC1_OUT. 101 **ALT5** — Select signal GPIO3_IO17. 110 **ALT6** — Select signal I2C3_SCL. - Configure register IOMUXC_I2C3_SCL_IN_SELECT_INPUT for mode ALT6.

36.4.34 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18)

Address: 20E_0000h base + 98h offset = 20E_0098h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

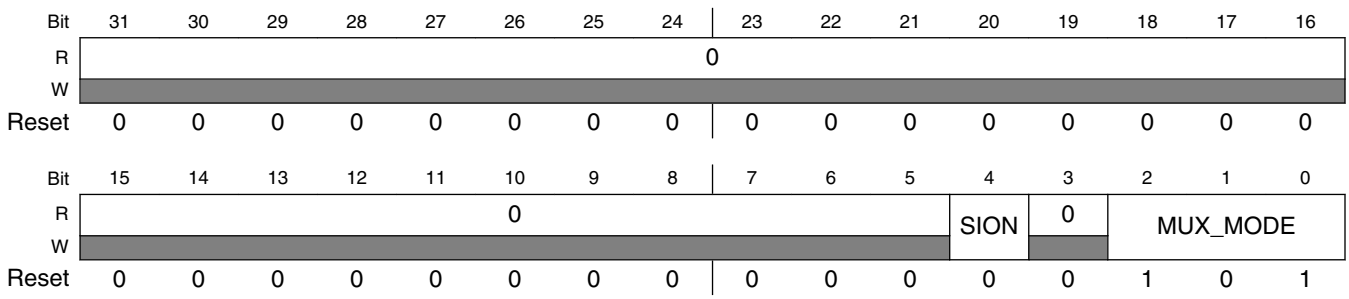
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA18 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D18. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D18. NOTE: Pad EIM_D18 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA18. 001 **ALT1** — Select signal ECSPI1_MOSI. - Configure register IOMUXC_ECSPI1_MOSI_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal IPU1_DI0_PIN07. 011 **ALT3** — Select signal IPU2_CSI1_DATA17. - Configure register IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal IPU1_DI1_D0_CS. 101 **ALT5** — Select signal GPIO3_IO18. 110 **ALT6** — Select signal I2C3_SDA. - Configure register IOMUXC_I2C3_SDA_IN_SELECT_INPUT for mode ALT6.

36.4.35 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19)

Address: 20E_0000h base + 9Ch offset = 20E_009Ch



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA19 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D19. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D19. NOTE: Pad EIM_D19 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA19. 001 **ALT1** — Select signal ECSPI1_SS1. - Configure register IOMUXC_ECSPI1_SS1_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal IPU1_DI0_PIN08. 011 **ALT3** — Select signal IPU2_CSI1_DATA16. - Configure register IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART1_CTS_B. - Configure register IOMUXC_UART1_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO19. 110 **ALT6** — Select signal EPIT1_OUT.

36.4.36 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20)

Address: 20E_0000h base + A0h offset = 20E_00A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

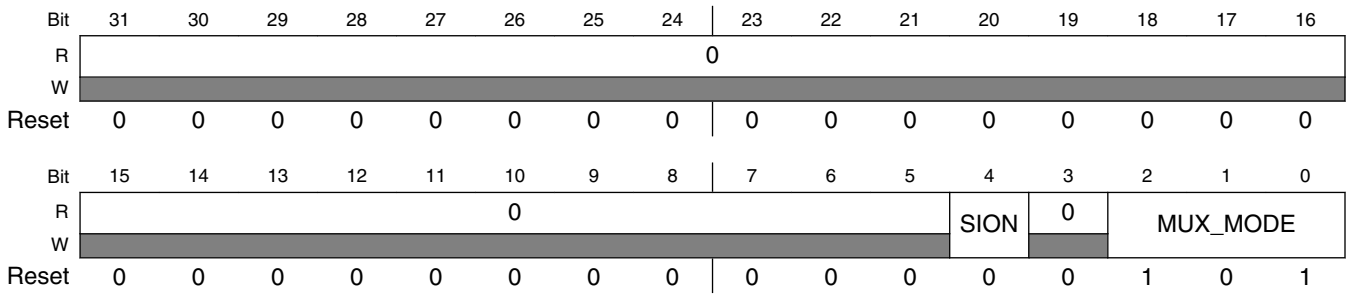
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA20 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D20. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D20. NOTE: Pad EIM_D20 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA20. 001 **ALT1** — Select signal ECSPI4_SS0. - Configure register IOMUXC_ECSPI4_SS0_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal IPU1_DI0_PIN16. 011 **ALT3** — Select signal IPU2_CSI1_DATA15. - Configure register IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART1_RTS_B. - Configure register IOMUXC_UART1_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO20. 110 **ALT6** — Select signal EPIT2_OUT.

36.4.37 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21)

Address: 20E_0000h base + A4h offset = 20E_00A4h



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA21 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D21. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D21. NOTE: Pad EIM_D21 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA21. 001 **ALT1** — Select signal ECSPI4_SCLK. 010 **ALT2** — Select signal IPU1_DI0_PIN17. 011 **ALT3** — Select signal IPU2_CSI1_DATA11. - Configure register IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal USB_OTG_OC. - Configure register IOMUXC_USB_OTG_OC_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO21. 110 **ALT6** — Select signal I2C1_SCL. - Configure register IOMUXC_I2C1_SCL_IN_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal SPDIF_IN. - Configure register IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT for mode ALT7.

36.4.38 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22)

Address: 20E_0000h base + A8h offset = 20E_00A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

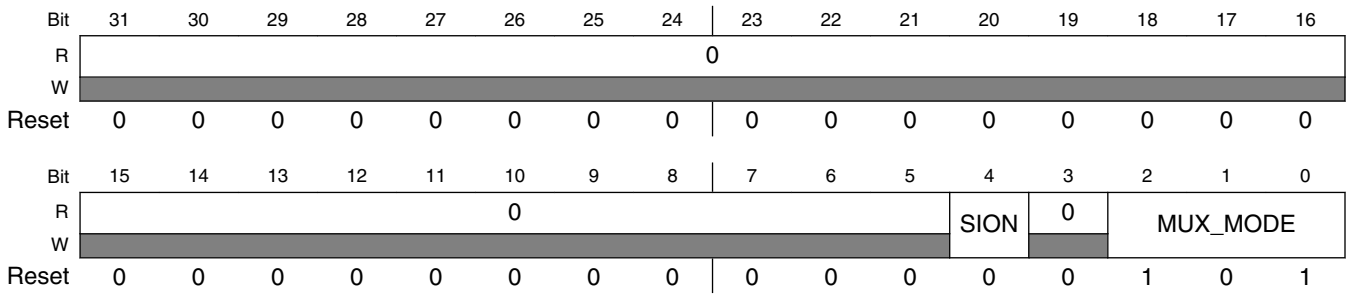
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA22 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D22. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D22. NOTE: Pad EIM_D22 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA22. 001 **ALT1** — Select signal ECSPI4_MISO. 010 **ALT2** — Select signal IPU1_DI0_PIN01. 011 **ALT3** — Select signal IPU2_CSI1_DATA10. - Configure register IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal USB_OTG_PWR. 101 **ALT5** — Select signal GPIO3_IO22. 110 **ALT6** — Select signal SPDIF_OUT.

36.4.39 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23)

Address: 20E_0000h base + ACh offset = 20E_00ACh



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA23 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D23. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D23. NOTE: Pad EIM_D23 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA23. 001 **ALT1** — Select signal IPU1_DI0_D0_CS. 010 **ALT2** — Select signal UART3_CTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal UART1_DCD_B. 100 **ALT4** — Select signal IPU2_CSI1_DATA_EN. - Configure register IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO23. 110 **ALT6** — Select signal IPU1_DI1_PIN02. 111 **ALT7** — Select signal IPU1_DI1_PIN14.

36.4.40 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B)

Address: 20E_0000h base + B0h offset = 20E_00B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

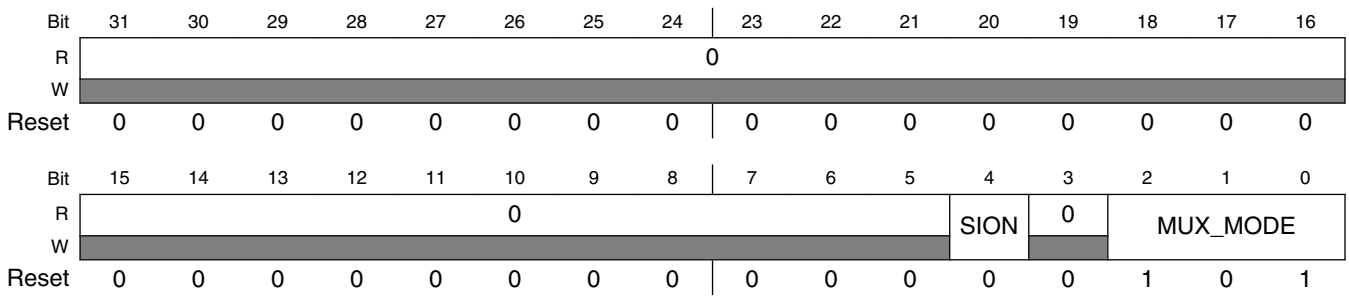
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_EB3_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_EB3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_EB3. NOTE: Pad EIM_EB3 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_EB3_B. 001 **ALT1** — Select signal ECSPI4_RDY. 010 **ALT2** — Select signal UART3_RTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal UART1_RI_B. 100 **ALT4** — Select signal IPU2_CSI1_HSYNC. - Configure register IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO2_IO31. 110 **ALT6** — Select signal IPU1_DI1_PIN03. 111 **ALT7** — Select signal SRC_BOOT_CFG31.

36.4.41 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24)

Address: 20E_0000h base + B4h offset = 20E_00B4h



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA24 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D24. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D24. NOTE: Pad EIM_D24 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA24. 001 **ALT1** — Select signal ECSPI4_SS2. 010 **ALT2** — Select signal UART3_TX_DATA. - Configure register IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal ECSPI1_SS2. - Configure register IOMUXC_ECSPI1_SS2_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal ECSPI2_SS2. 101 **ALT5** — Select signal GPIO3_IO24. 110 **ALT6** — Select signal AUD5_RXFS. - Configure register IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal UART1_DTR_B.

36.4.42 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25)

Address: 20E_0000h base + B8h offset = 20E_00B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

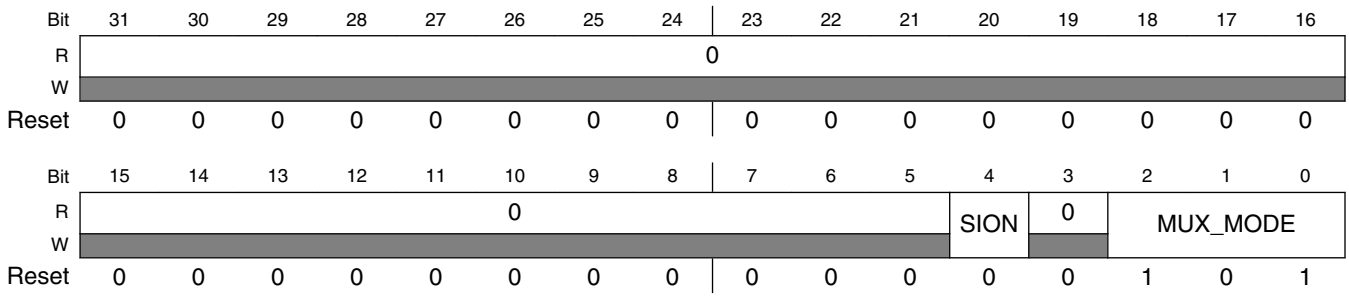
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA25 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D25. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D25. NOTE: Pad EIM_D25 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA25. 001 **ALT1** — Select signal ECSPI4_SS3. 010 **ALT2** — Select signal UART3_RX_DATA. - Configure register IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal ECSPI1_SS3. - Configure register IOMUXC_ECSPI1_SS3_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal ECSPI2_SS3. 101 **ALT5** — Select signal GPIO3_IO25. 110 **ALT6** — Select signal AUD5_RXC. - Configure register IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal UART1_DSR_B.

36.4.43 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26)

Address: 20E_0000h base + BCh offset = 20E_00BCh



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA26 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D26. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D26. NOTE: Pad EIM_D26 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA26. 001 **ALT1** — Select signal IPU1_DI1_PIN11. 010 **ALT2** — Select signal IPU1_CSI0_DATA01. 011 **ALT3** — Select signal IPU2_CSI1_DATA14. - Configure register IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART2_TX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO26. 110 **ALT6** — Select signal IPU1_SISG2. 111 **ALT7** — Select signal IPU1_DISP1_DATA22.

36.4.44 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27)

Address: 20E_0000h base + C0h offset = 20E_00C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

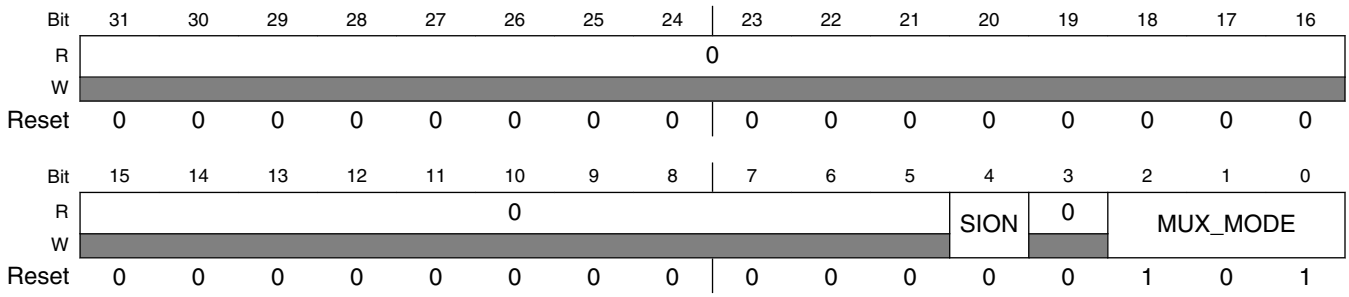
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA27 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D27. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D27. NOTE: Pad EIM_D27 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA27. 001 **ALT1** — Select signal IPU1_DI1_PIN13. 010 **ALT2** — Select signal IPU1_CSI0_DATA00. 011 **ALT3** — Select signal IPU2_CSI1_DATA13. - Configure register IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART2_RX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO27. 110 **ALT6** — Select signal IPU1_SISG3. 111 **ALT7** — Select signal IPU1_DISP1_DATA23.

36.4.45 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28)

Address: 20E_0000h base + C4h offset = 20E_00C4h



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA28 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D28. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: EIM_D28. NOTE: Pad EIM_D28 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA28. 001 **ALT1** — Select signal I2C1_SDA. - Configure register IOMUXC_I2C1_SDA_IN_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal ECSPI4_MOSI. 011 **ALT3** — Select signal IPU2_CSI1_DATA12. - Configure register IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART2_CTS_B. - Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO28. 110 **ALT6** — Select signal IPU1_EXT_TRIG. 111 **ALT7** — Select signal IPU1_DI0_PIN13.

36.4.46 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29)

Address: 20E_0000h base + C8h offset = 20E_00C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

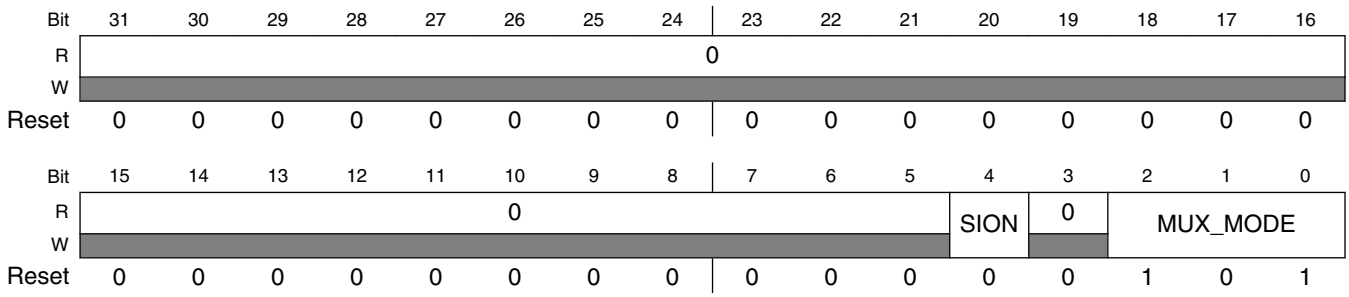
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA29 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D29. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D29. NOTE: Pad EIM_D29 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA29. 001 **ALT1** — Select signal IPU1_DI1_PIN15. 010 **ALT2** — Select signal ECSPI4_SS0. - Configure register IOMUXC_ECSPI4_SS0_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal UART2_RTS_B. - Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO29. 110 **ALT6** — Select signal IPU2_CSI1_VSYNC. - Configure register IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal IPU1_DI0_PIN14.

36.4.47 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30)

Address: 20E_0000h base + CCh offset = 20E_00CCh



IOMUXC_SW_MUX_CTL_PAD_EIM_DATA30 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D30. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D30. NOTE: Pad EIM_D30 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA30. 001 **ALT1** — Select signal IPU1_DISP1_DATA21. 010 **ALT2** — Select signal IPU1_DI0_PIN11. 011 **ALT3** — Select signal IPU1_CSI0_DATA03. 100 **ALT4** — Select signal UART3_CTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO30. 110 **ALT6** — Select signal USB_H1_OC. - Configure register IOMUXC_USB_H1_OC_SELECT_INPUT for mode ALT6.

36.4.48 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31)

Address: 20E_0000h base + D0h offset = 20E_00D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

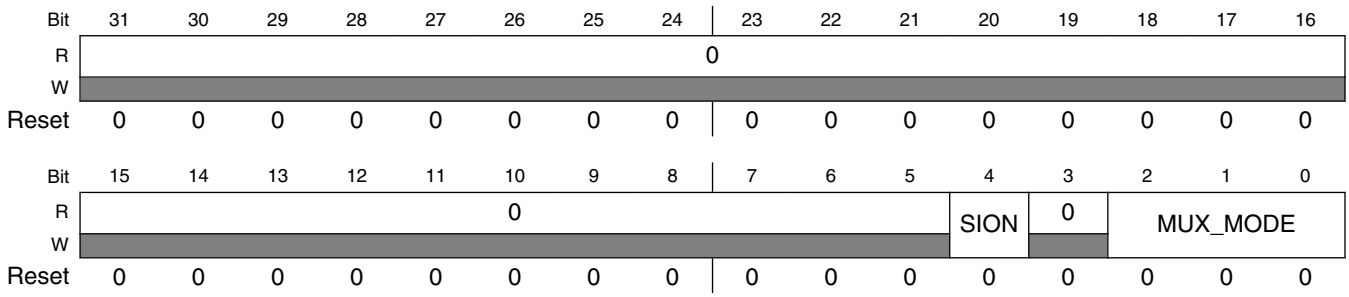
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_EIM_DATA31 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_D31. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_D31. NOTE: Pad EIM_D31 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_DATA31. 001 **ALT1** — Select signal IPU1_DISP1_DATA20. 010 **ALT2** — Select signal IPU1_DI0_PIN12. 011 **ALT3** — Select signal IPU1_CSI0_DATA02. 100 **ALT4** — Select signal UART3_RTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO3_IO31. 110 **ALT6** — Select signal USB_H1_PWR.

36.4.49 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24)

Address: 20E_0000h base + D4h offset = 20E_00D4h

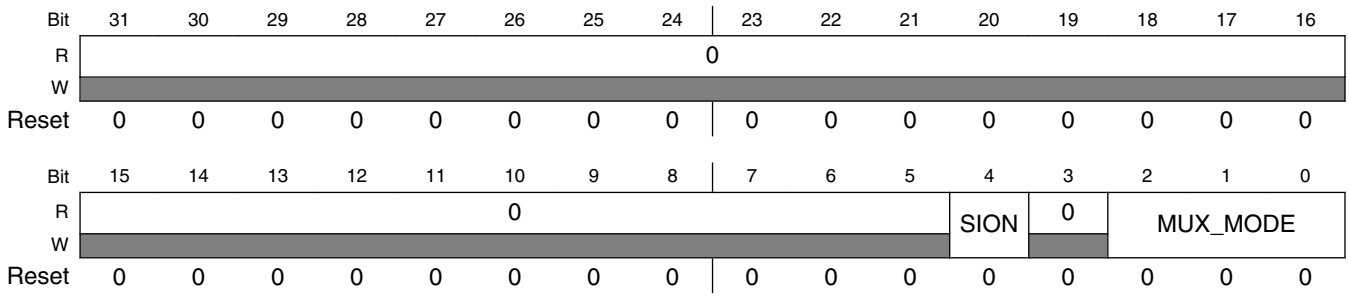


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR24 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A24. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_A24. NOTE: Pad EIM_A24 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR24. 001 **ALT1** — Select signal IPU1_DISP1_DATA19. 010 **ALT2** — Select signal IPU2_CSI1_DATA19. - Configure register IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal IPU2_SISG2. 100 **ALT4** — Select signal IPU1_SISG2. 101 **ALT5** — Select signal GPIO5_IO04. 111 **ALT7** — Select signal SRC_BOOT_CFG24.

36.4.50 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23)

Address: 20E_0000h base + D8h offset = 20E_00D8h

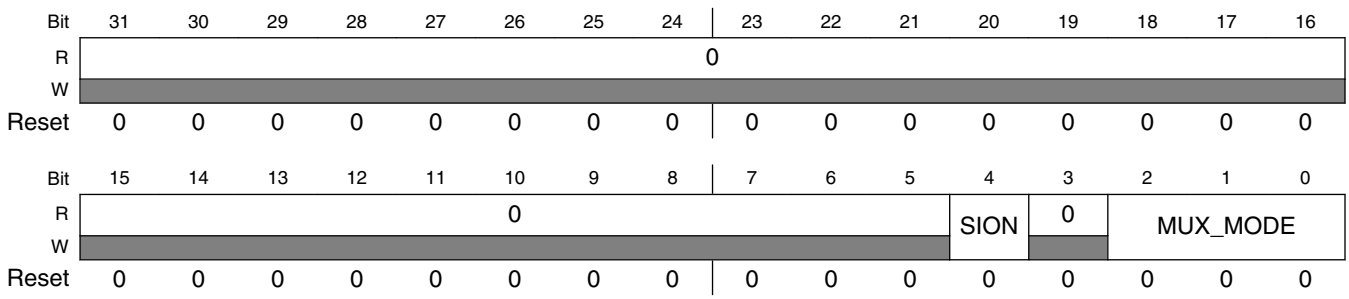


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR23 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A23. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: EIM_A23. NOTE: Pad EIM_A23 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR23. 001 **ALT1** — Select signal IPU1_DISP1_DATA18. 010 **ALT2** — Select signal IPU2_CSI1_DATA18. - Configure register IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal IPU2_SISG3. 100 **ALT4** — Select signal IPU1_SISG3. 101 **ALT5** — Select signal GPIO6_IO06. 111 **ALT7** — Select signal SRC_BOOT_CFG23.

36.4.51 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22)

Address: 20E_0000h base + DCh offset = 20E_00DCh

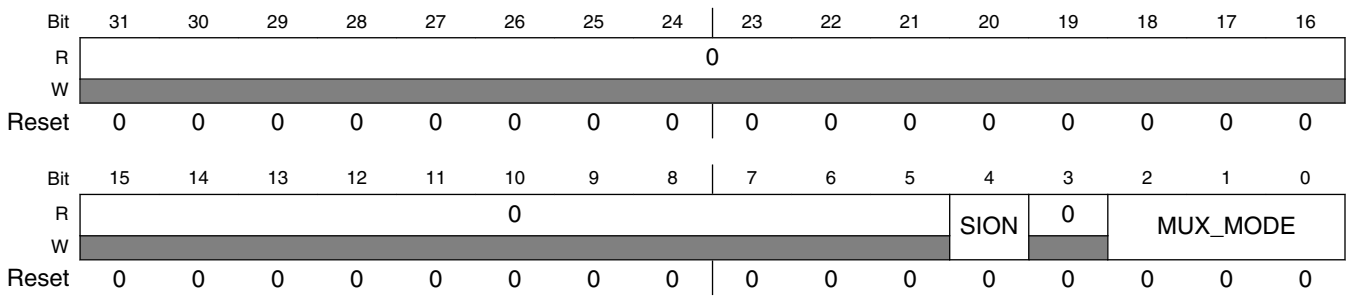


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR22 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A22. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A22. NOTE: Pad EIM_A22 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR22. 001 **ALT1** — Select signal IPU1_DISP1_DATA17. 010 **ALT2** — Select signal IPU2_CSI1_DATA17. - Configure register IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO16. 111 **ALT7** — Select signal SRC_BOOT_CFG22.

36.4.52 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21)

Address: 20E_0000h base + E0h offset = 20E_00E0h

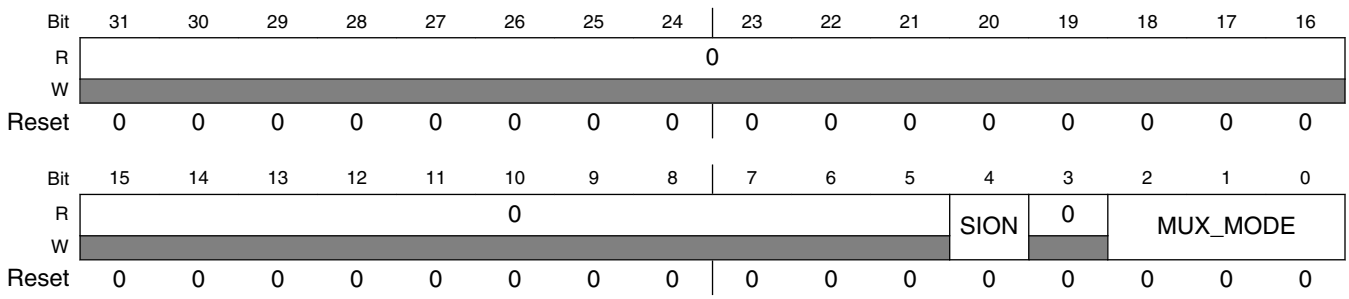


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR21 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A21. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A21. NOTE: Pad EIM_A21 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR21. 001 **ALT1** — Select signal IPU1_DISP1_DATA16. 010 **ALT2** — Select signal IPU2_CSI1_DATA16. - Configure register IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO17. 111 **ALT7** — Select signal SRC_BOOT_CFG21.

36.4.53 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20)

Address: 20E_0000h base + E4h offset = 20E_00E4h



IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR20 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A20. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A20. NOTE: Pad EIM_A20 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR20. 001 **ALT1** — Select signal IPU1_DISP1_DATA15. 010 **ALT2** — Select signal IPU2_CSI1_DATA15. - Configure register IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO18. 111 **ALT7** — Select signal SRC_BOOT_CFG20.

36.4.54 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19)

Address: 20E_0000h base + E8h offset = 20E_00E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

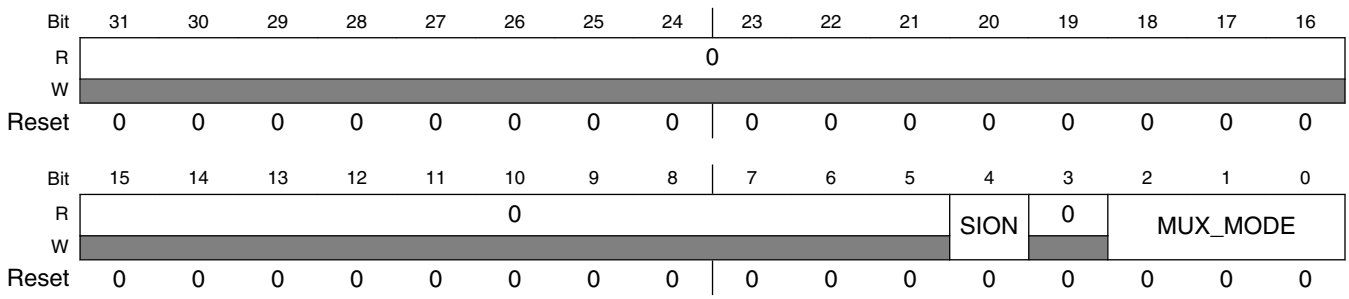
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR19 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A19. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A19. NOTE: Pad EIM_A19 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR19. 001 **ALT1** — Select signal IPU1_DISP1_DATA14. 010 **ALT2** — Select signal IPU2_CSI1_DATA14. - Configure register IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO19. 111 **ALT7** — Select signal SRC_BOOT_CFG19.

36.4.55 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18)

Address: 20E_0000h base + ECh offset = 20E_00ECh



IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR18 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A18. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A18. NOTE: Pad EIM_A18 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR18. 001 **ALT1** — Select signal IPU1_DISP1_DATA13. 010 **ALT2** — Select signal IPU2_CSI1_DATA13. - Configure register IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO20. 111 **ALT7** — Select signal SRC_BOOT_CFG18.

36.4.56 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17)

Address: 20E_0000h base + F0h offset = 20E_00F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

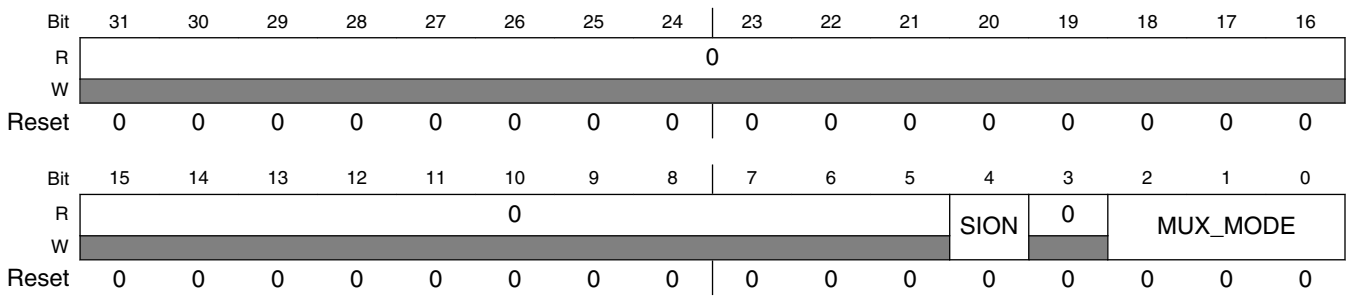
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR17 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A17. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A17. NOTE: Pad EIM_A17 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR17. 001 **ALT1** — Select signal IPU1_DISP1_DATA12. 010 **ALT2** — Select signal IPU2_CSI1_DATA12. - Configure register IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO21. 111 **ALT7** — Select signal SRC_BOOT_CFG17.

36.4.57 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16)

Address: 20E_0000h base + F4h offset = 20E_00F4h

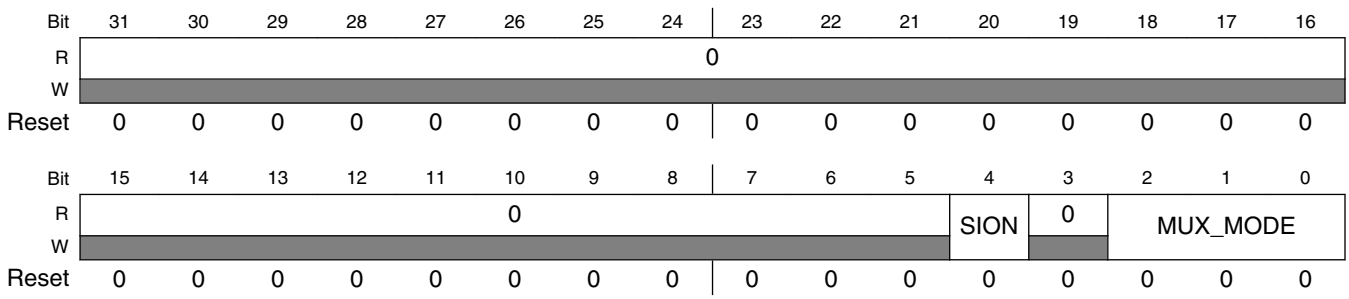


IOMUXC_SW_MUX_CTL_PAD_EIM_ADDR16 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_A16. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_A16. NOTE: Pad EIM_A16 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_ADDR16. 001 **ALT1** — Select signal IPU1_DI1_DISP_CLK. 010 **ALT2** — Select signal IPU2_CSI1_PIXCLK. - Configure register IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO22. 111 **ALT7** — Select signal SRC_BOOT_CFG16.

36.4.58 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B)

Address: 20E_0000h base + F8h offset = 20E_00F8h



IOMUXC_SW_MUX_CTL_PAD_EIM_CS0_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_CS0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_CS0. NOTE: Pad EIM_CS0 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_CS0_B. 001 **ALT1** — Select signal IPU1_DI1_PIN05. 010 **ALT2** — Select signal ECSPI2_SCLK. - Configure register IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO23.

36.4.59 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B)

Address: 20E_0000h base + FCh offset = 20E_00FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

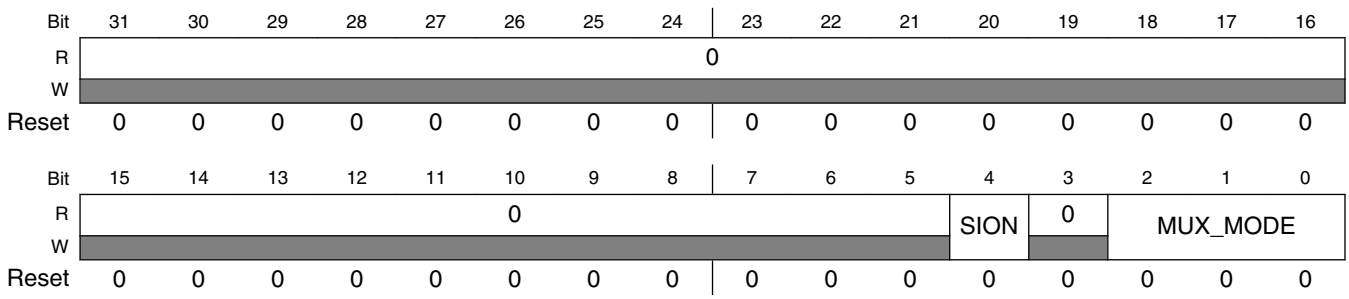
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_CS1_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_CS1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_CS1. NOTE: Pad EIM_CS1 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_CS1_B. 001 **ALT1** — Select signal IPU1_DI1_PIN06. 010 **ALT2** — Select signal ECSPI2_MOSI. - Configure register IOMUXC_ECSPI2_MOSI_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO24.

36.4.60 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B)

Address: 20E_0000h base + 100h offset = 20E_0100h

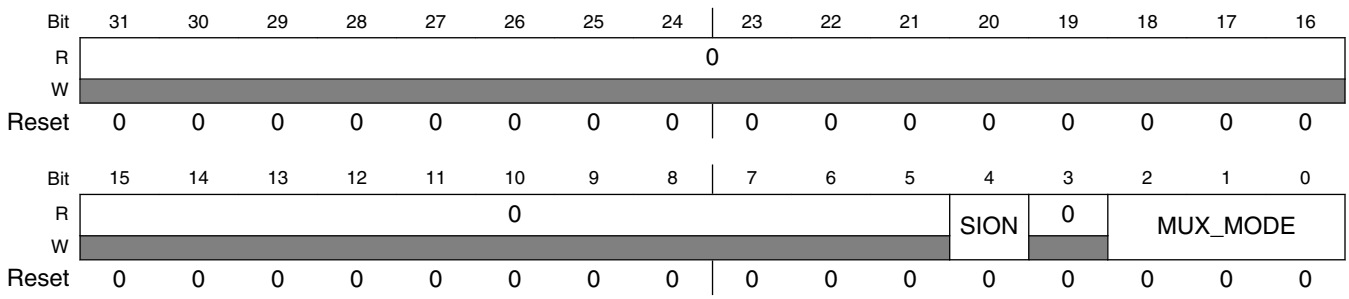


IOMUXC_SW_MUX_CTL_PAD_EIM_OE_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_OE. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_OE. NOTE: Pad EIM_OE is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_OE_B. 001 **ALT1** — Select signal IPU1_DI1_PIN07. 010 **ALT2** — Select signal ECSPI2_MISO. - Configure register IOMUXC_ECSPI2_MISO_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO25.

36.4.61 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_RW)

Address: 20E_0000h base + 104h offset = 20E_0104h

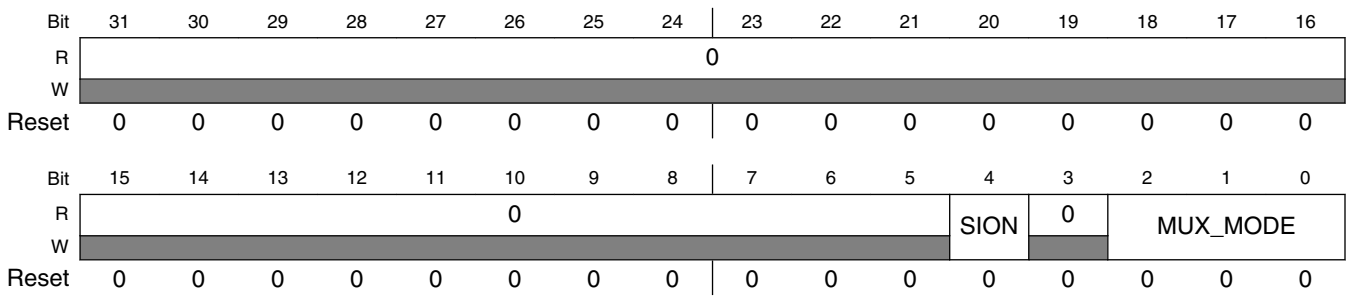


IOMUXC_SW_MUX_CTL_PAD_EIM_RW field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_RW. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_RW. NOTE: Pad EIM_RW is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_RW. 001 **ALT1** — Select signal IPU1_DI1_PIN08. 010 **ALT2** — Select signal ECSPI2_SS0. - Configure register IOMUXC_ECSPI2_SS0_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO26. 111 **ALT7** — Select signal SRC_BOOT_CFG29.

36.4.62 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B)

Address: 20E_0000h base + 108h offset = 20E_0108h



IOMUXC_SW_MUX_CTL_PAD_EIM_LBA_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_LBA. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_LBA. NOTE: Pad EIM_LBA is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_LBA_B. 001 **ALT1** — Select signal IPU1_DI1_PIN17. 010 **ALT2** — Select signal ECSPI2_SS1. - Configure register IOMUXC_ECSPI2_SS1_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO27. 111 **ALT7** — Select signal SRC_BOOT_CFG26.

36.4.63 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B)

Address: 20E_0000h base + 10Ch offset = 20E_010Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

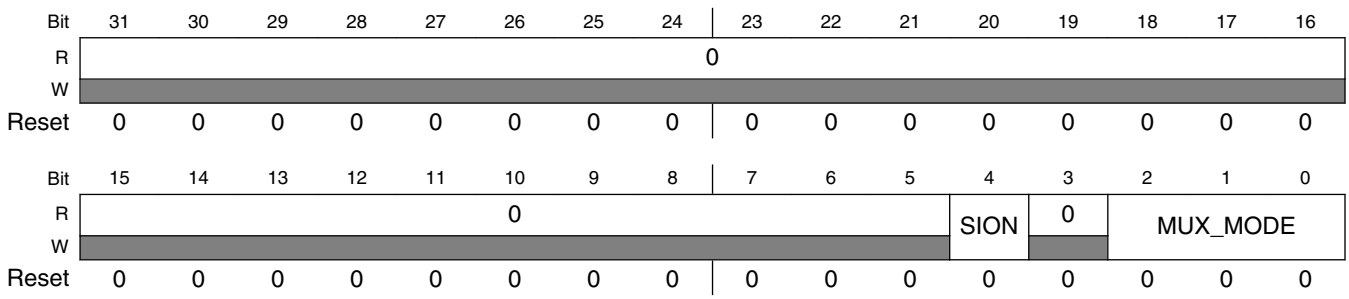
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_EB0_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_EB0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: EIM_EB0. NOTE: Pad EIM_EB0 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_EB0_B. 001 **ALT1** — Select signal IPU1_DISP1_DATA11. 010 **ALT2** — Select signal IPU2_CSI1_DATA11. - Configure register IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal CCM_PMIC_READY. - Configure register IOMUXC_CCM_PMIC_READY_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO2_IO28. 111 **ALT7** — Select signal SRC_BOOT_CFG27.

36.4.64 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B)

Address: 20E_0000h base + 110h offset = 20E_0110h



IOMUXC_SW_MUX_CTL_PAD_EIM_EB1_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_EB1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_EB1. NOTE: Pad EIM_EB1 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_EB1_B. 001 **ALT1** — Select signal IPU1_DISP1_DATA10. 010 **ALT2** — Select signal IPU2_CSI1_DATA10. - Configure register IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO29. 111 **ALT7** — Select signal SRC_BOOT_CFG28.

36.4.65 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD00)

Address: 20E_0000h base + 114h offset = 20E_0114h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

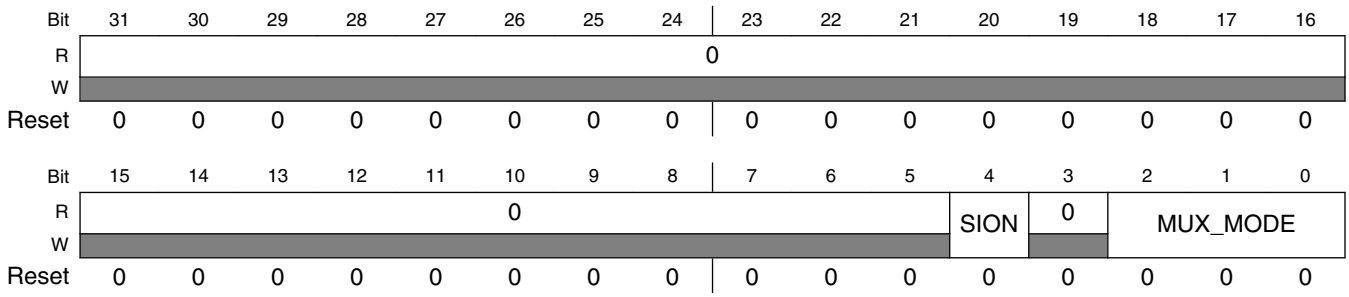
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD00 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA0. 000 **ALT0** — Select signal EIM_AD00. 001 **ALT1** — Select signal IPU1_DISP1_DATA09. 010 **ALT2** — Select signal IPU2_CSI1_DATA09. 101 **ALT5** — Select signal GPIO3_IO00. 111 **ALT7** — Select signal SRC_BOOT_CFG00.

36.4.66 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD01)

Address: 20E_0000h base + 118h offset = 20E_0118h



IOMUXC_SW_MUX_CTL_PAD_EIM_AD01 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA1. 000 **ALT0** — Select signal EIM_AD01. 001 **ALT1** — Select signal IPU1_DISP1_DATA08. 010 **ALT2** — Select signal IPU2_CSI1_DATA08. 101 **ALT5** — Select signal GPIO3_IO01. 111 **ALT7** — Select signal SRC_BOOT_CFG01.

36.4.67 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD02)

Address: 20E_0000h base + 11Ch offset = 20E_011Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

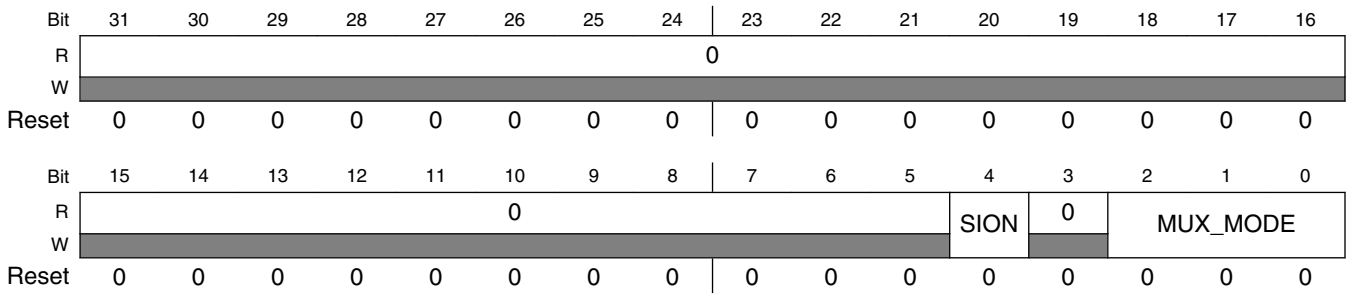
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD02 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA2. 000 **ALT0** — Select signal EIM_AD02. 001 **ALT1** — Select signal IPU1_DISP1_DATA07. 010 **ALT2** — Select signal IPU2_CSI1_DATA07. 101 **ALT5** — Select signal GPIO3_IO02. 111 **ALT7** — Select signal SRC_BOOT_CFG02.

36.4.68 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD03)

Address: 20E_0000h base + 120h offset = 20E_0120h



IOMUXC_SW_MUX_CTL_PAD_EIM_AD03 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA3. 000 **ALT0** — Select signal EIM_AD03. 001 **ALT1** — Select signal IPU1_DISP1_DATA06. 010 **ALT2** — Select signal IPU2_CSI1_DATA06. 101 **ALT5** — Select signal GPIO3_IO03. 111 **ALT7** — Select signal SRC_BOOT_CFG03.

36.4.69 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD04)

Address: 20E_0000h base + 124h offset = 20E_0124h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

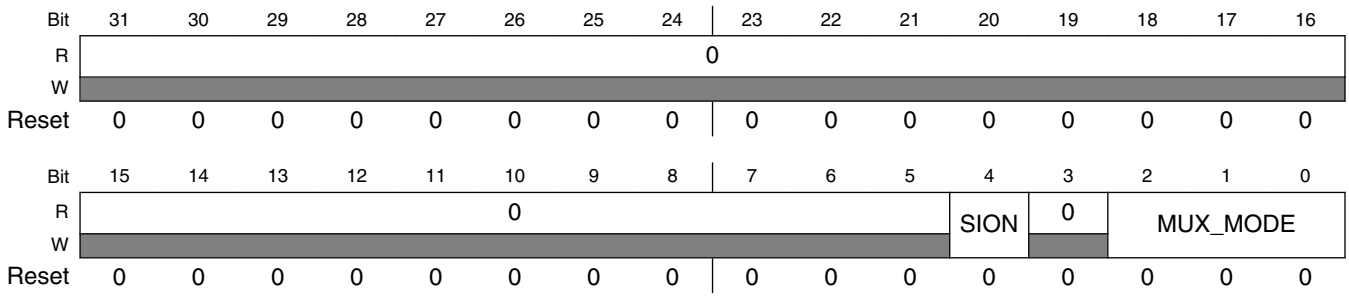
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA4. 000 **ALT0** — Select signal EIM_AD04. 001 **ALT1** — Select signal IPU1_DISP1_DATA05. 010 **ALT2** — Select signal IPU2_CSI1_DATA05. 101 **ALT5** — Select signal GPIO3_IO04. 111 **ALT7** — Select signal SRC_BOOT_CFG04.

36.4.70 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD05)

Address: 20E_0000h base + 128h offset = 20E_0128h

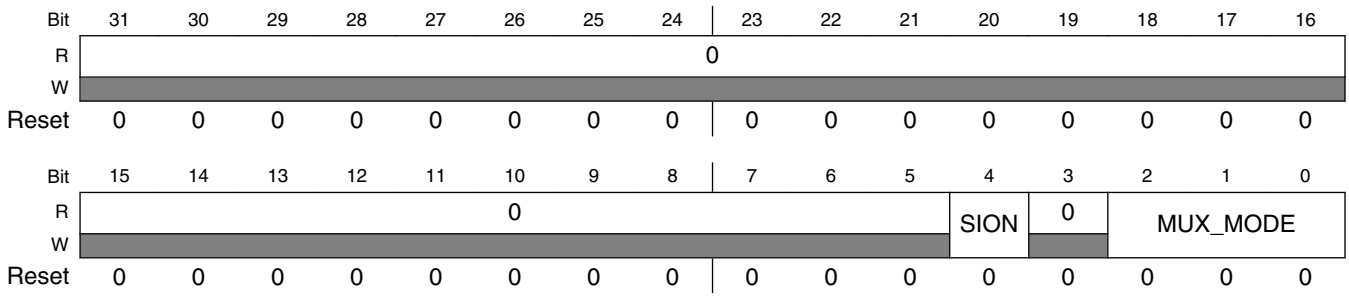


IOMUXC_SW_MUX_CTL_PAD_EIM_AD05 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA5. 000 **ALT0** — Select signal EIM_AD05. 001 **ALT1** — Select signal IPU1_DISP1_DATA04. 010 **ALT2** — Select signal IPU2_CSI1_DATA04. 101 **ALT5** — Select signal GPIO3_IO05. 111 **ALT7** — Select signal SRC_BOOT_CFG05.

36.4.71 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD06)

Address: 20E_0000h base + 12Ch offset = 20E_012Ch



IOMUXC_SW_MUX_CTL_PAD_EIM_AD06 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA6. 000 **ALT0** — Select signal EIM_AD06. 001 **ALT1** — Select signal IPU1_DISP1_DATA03. 010 **ALT2** — Select signal IPU2_CSI1_DATA03. 101 **ALT5** — Select signal GPIO3_IO06. 111 **ALT7** — Select signal SRC_BOOT_CFG06.

36.4.72 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD07)

Address: 20E_0000h base + 130h offset = 20E_0130h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

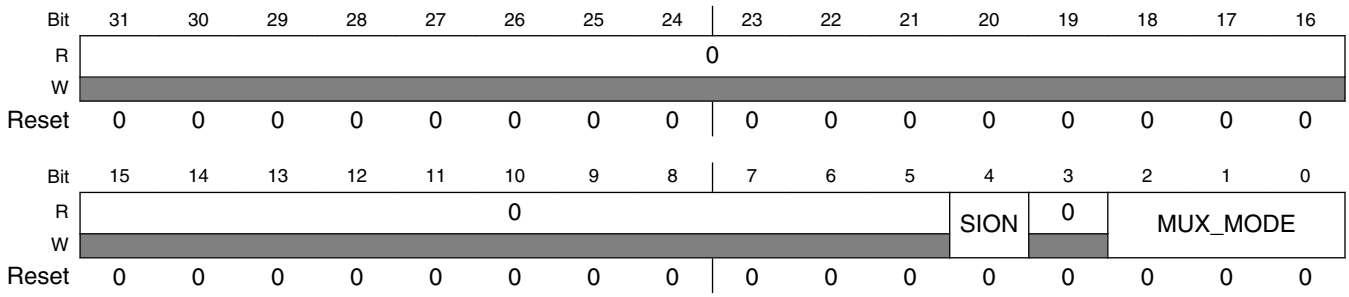
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD07 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA7. 000 **ALT0** — Select signal EIM_AD07. 001 **ALT1** — Select signal IPU1_DISP1_DATA02. 010 **ALT2** — Select signal IPU2_CSI1_DATA02. 101 **ALT5** — Select signal GPIO3_IO07. 111 **ALT7** — Select signal SRC_BOOT_CFG07.

36.4.73 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD08)

Address: 20E_0000h base + 134h offset = 20E_0134h

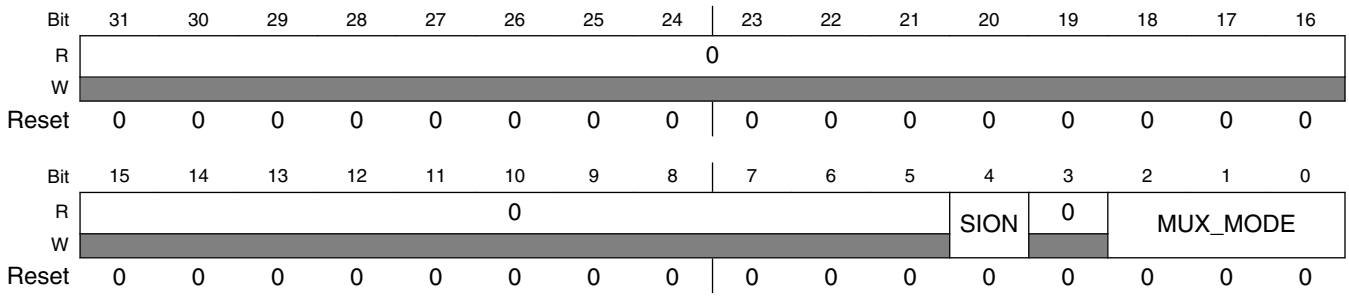


IOMUXC_SW_MUX_CTL_PAD_EIM_AD08 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA8. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA8. 000 **ALT0** — Select signal EIM_AD08. 001 **ALT1** — Select signal IPU1_DISP1_DATA01. 010 **ALT2** — Select signal IPU2_CSI1_DATA01. 101 **ALT5** — Select signal GPIO3_IO08. 111 **ALT7** — Select signal SRC_BOOT_CFG08.

36.4.74 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD09)

Address: 20E_0000h base + 138h offset = 20E_0138h

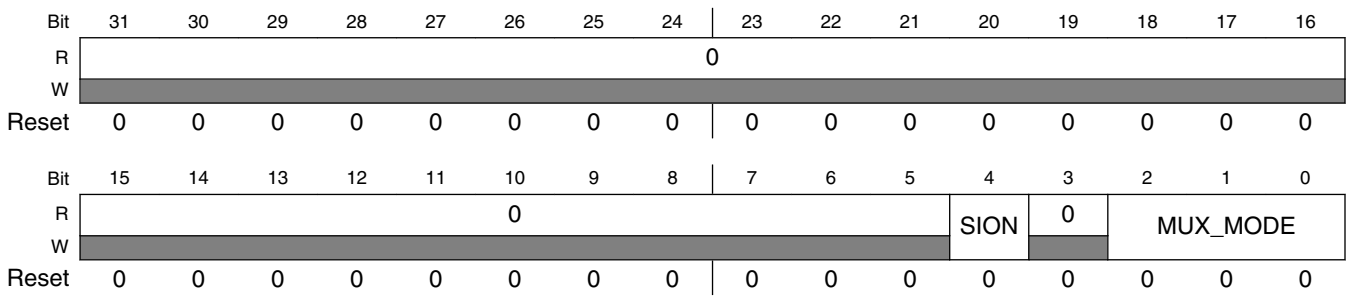


IOMUXC_SW_MUX_CTL_PAD_EIM_AD09 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA9. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA9. 000 **ALT0** — Select signal EIM_AD09. 001 **ALT1** — Select signal IPU1_DISP1_DATA00. 010 **ALT2** — Select signal IPU2_CSI1_DATA00. 101 **ALT5** — Select signal GPIO3_IO09. 111 **ALT7** — Select signal SRC_BOOT_CFG09.

36.4.75 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD10)

Address: 20E_0000h base + 13Ch offset = 20E_013Ch

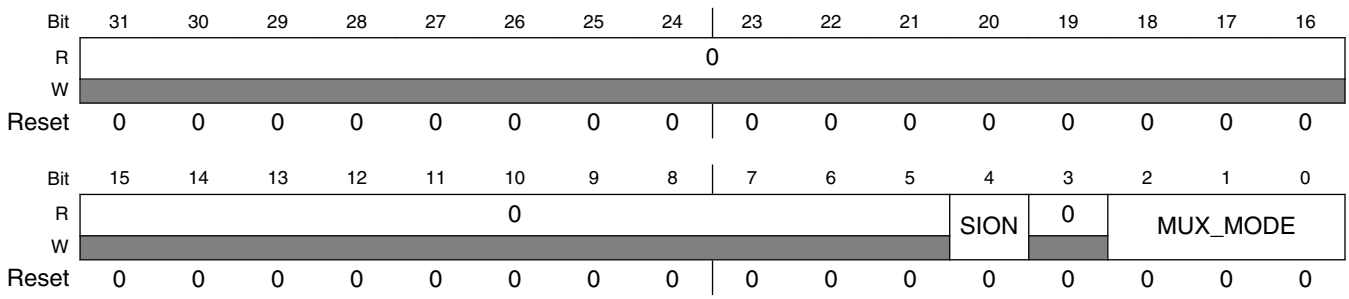


IOMUXC_SW_MUX_CTL_PAD_EIM_AD10 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA10. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA10. NOTE: Pad EIM_DA10 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_AD10. 001 **ALT1** — Select signal IPU1_DI1_PIN15. 010 **ALT2** — Select signal IPU2_CSI1_DATA_EN. - Configure register IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO3_IO10. 111 **ALT7** — Select signal SRC_BOOT_CFG10.

36.4.76 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD11)

Address: 20E_0000h base + 140h offset = 20E_0140h



IOMUXC_SW_MUX_CTL_PAD_EIM_AD11 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA11. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA11. NOTE: Pad EIM_DA11 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_AD11. 001 **ALT1** — Select signal IPU1_DI1_PIN02. 010 **ALT2** — Select signal IPU2_CSI1_HSYNC. - Configure register IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO3_IO11. 111 **ALT7** — Select signal SRC_BOOT_CFG11.

36.4.77 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD12)

Address: 20E_0000h base + 144h offset = 20E_0144h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD12 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA12. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA12. NOTE: Pad EIM_DA12 is involved in Daisy Chain. 000 **ALT0** — Select signal EIM_AD12. 001 **ALT1** — Select signal IPU1_DI1_PIN03. 010 **ALT2** — Select signal IPU2_CSI1_VSYNC. - Configure register IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO3_IO12. 111 **ALT7** — Select signal SRC_BOOT_CFG12.

36.4.78 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD13)

Address: 20E_0000h base + 148h offset = 20E_0148h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

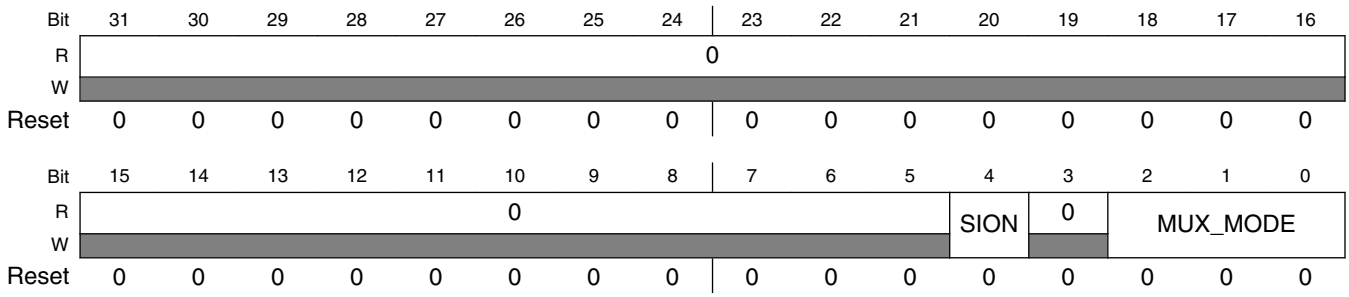
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD13 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA13. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_DA13. 000 **ALT0** — Select signal EIM_AD13. 001 **ALT1** — Select signal IPU1_DI1_D0_CS. 101 **ALT5** — Select signal GPIO3_IO13. 111 **ALT7** — Select signal SRC_BOOT_CFG13.

36.4.79 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD14)

Address: 20E_0000h base + 14Ch offset = 20E_014Ch



IOMUXC_SW_MUX_CTL_PAD_EIM_AD14 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA14. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_DA14. 000 **ALT0** — Select signal EIM_AD14. 001 **ALT1** — Select signal IPU1_DI1_D1_CS. 101 **ALT5** — Select signal GPIO3_IO14. 111 **ALT7** — Select signal SRC_BOOT_CFG14.

36.4.80 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_AD15)

Address: 20E_0000h base + 150h offset = 20E_0150h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_AD15 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_DA15. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: EIM_DA15. 000 **ALT0** — Select signal EIM_AD15. 001 **ALT1** — Select signal IPU1_DI1_PIN01. 010 **ALT2** — Select signal IPU1_DI1_PIN04. 101 **ALT5** — Select signal GPIO3_IO15. 111 **ALT7** — Select signal SRC_BOOT_CFG15.

36.4.81 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B)

Address: 20E_0000h base + 154h offset = 20E_0154h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_WAIT_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_WAIT. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: EIM_WAIT. 000 **ALT0** — Select signal EIM_WAIT_B. 001 **ALT1** — Select signal EIM_DTACK_B. 101 **ALT5** — Select signal GPIO5_IO00. 111 **ALT7** — Select signal SRC_BOOT_CFG25.

36.4.82 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK)

Address: 20E_0000h base + 158h offset = 20E_0158h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_MUX_CTL_PAD_EIM_BCLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad EIM_BCLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: EIM_BCLK. 000 **ALT0** — Select signal EIM_BCLK. 001 **ALT1** — Select signal IPU1_DI1_PIN16. 101 **ALT5** — Select signal GPIO6_IO31.

36.4.83 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK)

Address: 20E_0000h base + 15Ch offset = 20E_015Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DI0_DISP_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DI0_DISP_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: DI0_DISP_CLK. 000 **ALT0** — Select signal IPU1_DI0_DISP_CLK. 001 **ALT1** — Select signal IPU2_DI0_DISP_CLK. 101 **ALT5** — Select signal GPIO4_IO16.

36.4.84 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15)

Address: 20E_0000h base + 160h offset = 20E_0160h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DI0_PIN15 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DI0_PIN15. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DI0_PIN15. 000 **ALT0** — Select signal IPU1_DI0_PIN15. 001 **ALT1** — Select signal IPU2_DI0_PIN15. 010 **ALT2** — Select signal AUD6_TXC. 101 **ALT5** — Select signal GPIO4_IO17.

36.4.85 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02)

Address: 20E_0000h base + 164h offset = 20E_0164h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DI0_PIN02 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DI0_PIN2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DI0_PIN2. 000 **ALT0** — Select signal IPU1_DI0_PIN02. 001 **ALT1** — Select signal IPU2_DI0_PIN02. 010 **ALT2** — Select signal AUD6_TXD. 101 **ALT5** — Select signal GPIO4_IO18.

36.4.86 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03)

Address: 20E_0000h base + 168h offset = 20E_0168h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

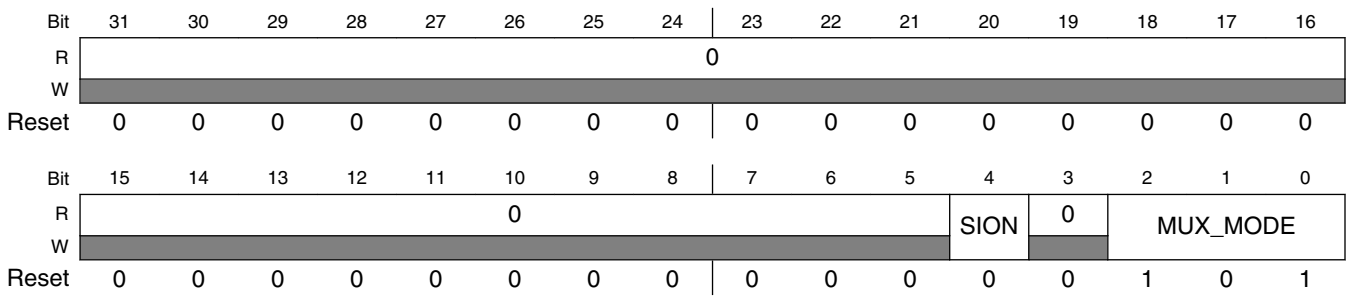
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DI0_PIN03 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DI0_PIN3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DI0_PIN3. 000 **ALT0** — Select signal IPU1_DI0_PIN03. 001 **ALT1** — Select signal IPU2_DI0_PIN03. 010 **ALT2** — Select signal AUD6_TXFS. 101 **ALT5** — Select signal GPIO4_IO19.

36.4.87 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04)

Address: 20E_0000h base + 16Ch offset = 20E_016Ch



IOMUXC_SW_MUX_CTL_PAD_DI0_PIN04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DI0_PIN4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DI0_PIN4. NOTE: Pad DI0_PIN4 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DI0_PIN04. 001 **ALT1** — Select signal IPU2_DI0_PIN04. 010 **ALT2** — Select signal AUD6_RXD. 011 **ALT3** — Select signal SD1_WP. - Configure register IOMUXC_USDHC1_WP_ON_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO4_IO20.

36.4.88 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00)

Address: 20E_0000h base + 170h offset = 20E_0170h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA00 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT0. 000 **ALT0** — Select signal IPU1_DISP0_DATA00. 001 **ALT1** — Select signal IPU2_DISP0_DATA00. 010 **ALT2** — Select signal ECSPI3_SCLK. 101 **ALT5** — Select signal GPIO4_IO21.

36.4.89 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01)

Address: 20E_0000h base + 174h offset = 20E_0174h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA01 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT1. 000 **ALT0** — Select signal IPU1_DISP0_DATA01. 001 **ALT1** — Select signal IPU2_DISP0_DATA01. 010 **ALT2** — Select signal ECSPI3_MOSI. 101 **ALT5** — Select signal GPIO4_IO22.

36.4.90 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02)

Address: 20E_0000h base + 178h offset = 20E_0178h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA02 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT2. 000 **ALT0** — Select signal IPU1_DISP0_DATA02. 001 **ALT1** — Select signal IPU2_DISP0_DATA02. 010 **ALT2** — Select signal ECSPI3_MISO. 101 **ALT5** — Select signal GPIO4_IO23.

36.4.91 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03)

Address: 20E_0000h base + 17Ch offset = 20E_017Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA03 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT3. 000 **ALT0** — Select signal IPU1_DISP0_DATA03. 001 **ALT1** — Select signal IPU2_DISP0_DATA03. 010 **ALT2** — Select signal ECSPI3_SS0. 101 **ALT5** — Select signal GPIO4_IO24.

36.4.92 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04)

Address: 20E_0000h base + 180h offset = 20E_0180h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT4. 000 **ALT0** — Select signal IPU1_DISP0_DATA04. 001 **ALT1** — Select signal IPU2_DISP0_DATA04. 010 **ALT2** — Select signal ECSPI3_SS1. 101 **ALT5** — Select signal GPIO4_IO25.

36.4.93 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05)

Address: 20E_0000h base + 184h offset = 20E_0184h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA05 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT5. 000 **ALT0** — Select signal IPU1_DISP0_DATA05. 001 **ALT1** — Select signal IPU2_DISP0_DATA05. 010 **ALT2** — Select signal ECSPI3_SS2. 011 **ALT3** — Select signal AUD6_RXFS. 101 **ALT5** — Select signal GPIO4_IO26.

36.4.94 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06)

Address: 20E_0000h base + 188h offset = 20E_0188h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA06 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT6. 000 **ALT0** — Select signal IPU1_DISP0_DATA06. 001 **ALT1** — Select signal IPU2_DISP0_DATA06. 010 **ALT2** — Select signal ECSPI3_SS3. 011 **ALT3** — Select signal AUD6_RXC. 101 **ALT5** — Select signal GPIO4_IO27.

36.4.95 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07)

Address: 20E_0000h base + 18Ch offset = 20E_018Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA07 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT7. 000 **ALT0** — Select signal IPU1_DISP0_DATA07. 001 **ALT1** — Select signal IPU2_DISP0_DATA07. 010 **ALT2** — Select signal ECSPI3_RDY. 101 **ALT5** — Select signal GPIO4_IO28.

36.4.96 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08)

Address: 20E_0000h base + 190h offset = 20E_0190h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA08 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT8. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT8. 000 **ALT0** — Select signal IPU1_DISP0_DATA08. 001 **ALT1** — Select signal IPU2_DISP0_DATA08. 010 **ALT2** — Select signal PWM1_OUT. 011 **ALT3** — Select signal WDOG1_B. 101 **ALT5** — Select signal GPIO4_IO29.

36.4.97 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09)

Address: 20E_0000h base + 194h offset = 20E_0194h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA09 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT9. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT9. 000 **ALT0** — Select signal IPU1_DISP0_DATA09. 001 **ALT1** — Select signal IPU2_DISP0_DATA09. 010 **ALT2** — Select signal PWM2_OUT. 011 **ALT3** — Select signal WDOG2_B. 101 **ALT5** — Select signal GPIO4_IO30.

36.4.98 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10)

Address: 20E_0000h base + 198h offset = 20E_0198h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

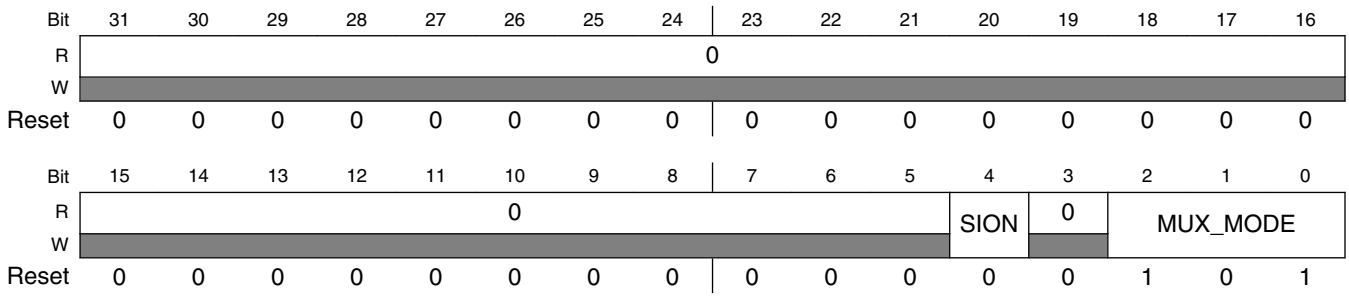
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA10 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT10. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: DISP0_DAT10. 000 **ALT0** — Select signal IPU1_DISP0_DATA10. 001 **ALT1** — Select signal IPU2_DISP0_DATA10. 101 **ALT5** — Select signal GPIO4_IO31.

36.4.99 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11)

Address: 20E_0000h base + 19Ch offset = 20E_019Ch



IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA11 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT11. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: DISP0_DAT11. 000 **ALT0** — Select signal IPU1_DISP0_DATA11. 001 **ALT1** — Select signal IPU2_DISP0_DATA11. 101 **ALT5** — Select signal GPIO5_IO05.

36.4.100 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12)

Address: 20E_0000h base + 1A0h offset = 20E_01A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

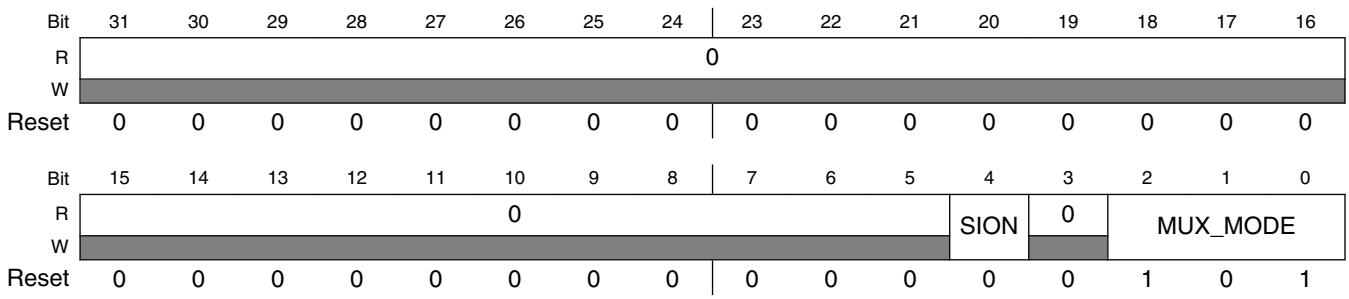
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA12 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT12. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: DISP0_DAT12. 000 **ALT0** — Select signal IPU1_DISP0_DATA12. 001 **ALT1** — Select signal IPU2_DISP0_DATA12. 101 **ALT5** — Select signal GPIO5_IO06.

36.4.101 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13)

Address: 20E_0000h base + 1A4h offset = 20E_01A4h



IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA13 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT13. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT13. NOTE: Pad DISP0_DAT13 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA13. 001 **ALT1** — Select signal IPU2_DISP0_DATA13. 011 **ALT3** — Select signal AUD5_RXFS. - Configure register IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO07.

36.4.102 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14)

Address: 20E_0000h base + 1A8h offset = 20E_01A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

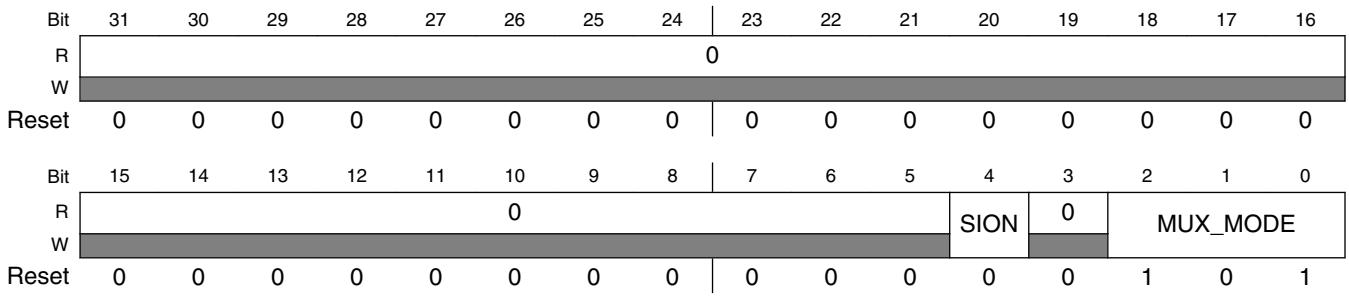
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA14 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT14. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: DISP0_DAT14. NOTE: Pad DISP0_DAT14 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA14. 001 **ALT1** — Select signal IPU2_DISP0_DATA14. 011 **ALT3** — Select signal AUD5_RXC. - Configure register IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO08.

36.4.103 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15)

Address: 20E_0000h base + 1ACh offset = 20E_01ACh



IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA15 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT15. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT15. NOTE: Pad DISP0_DAT15 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA15. 001 **ALT1** — Select signal IPU2_DISP0_DATA15. 010 **ALT2** — Select signal ECSPI1_SS1. - Configure register IOMUXC_ECSPI1_SS1_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal ECSPI2_SS1. - Configure register IOMUXC_ECSPI2_SS1_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO09.

36.4.104 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16)

Address: 20E_0000h base + 1B0h offset = 20E_01B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA16 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT16. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: DISP0_DAT16. NOTE: Pad DISP0_DAT16 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA16. 001 **ALT1** — Select signal IPU2_DISP0_DATA16. 010 **ALT2** — Select signal ECSPI2_MOSI. - Configure register IOMUXC_ECSPI2_MOSI_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD5_TXC. - Configure register IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal SDMA_EXT_EVENT0. - Configure register IOMUXC_SDMA_EVENTS14_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO10.

36.4.105 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17)

Address: 20E_0000h base + 1B4h offset = 20E_01B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA17 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT17. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: DISP0_DAT17. NOTE: Pad DISP0_DAT17 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA17. 001 **ALT1** — Select signal IPU2_DISP0_DATA17. 010 **ALT2** — Select signal ECSPI2_MISO. - Configure register IOMUXC_ECSPI2_MISO_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD5_TXD. - Configure register IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal SDMA_EXT_EVENT1. - Configure register IOMUXC_SDMA_EVENTS15_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO11.

36.4.106 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18)

Address: 20E_0000h base + 1B8h offset = 20E_01B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA18 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT18. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: DISP0_DAT18. NOTE: Pad DISP0_DAT18 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA18. 001 **ALT1** — Select signal IPU2_DISP0_DATA18. 010 **ALT2** — Select signal ECSPI2_SS0. - Configure register IOMUXC_ECSPI2_SS0_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD5_TXFS. - Configure register IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD4_RXFS. - Configure register IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO12. 111 **ALT7** — Select signal EIM_CS2_B.

36.4.107 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19)

Address: 20E_0000h base + 1BCh offset = 20E_01BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA19 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT19. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: DISP0_DAT19. NOTE: Pad DISP0_DAT19 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA19. 001 **ALT1** — Select signal IPU2_DISP0_DATA19. 010 **ALT2** — Select signal ECSPI2_SCLK. - Configure register IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD5_RXD. - Configure register IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD4_RXC. - Configure register IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO13. 111 **ALT7** — Select signal EIM_CS3_B.

36.4.108 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20)

Address: 20E_0000h base + 1C0h offset = 20E_01C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA20 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT20. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT20. NOTE: Pad DISP0_DAT20 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA20. 001 **ALT1** — Select signal IPU2_DISP0_DATA20. 010 **ALT2** — Select signal ECSPI1_SCLK. - Configure register IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_TXC. - Configure register IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO14.

36.4.109 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21)

Address: 20E_0000h base + 1C4h offset = 20E_01C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA21 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT21. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT21. NOTE: Pad DISP0_DAT21 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA21. 001 **ALT1** — Select signal IPU2_DISP0_DATA21. 010 **ALT2** — Select signal ECSPI1_MOSI. - Configure register IOMUXC_ECSPI1_MOSI_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_TXD. - Configure register IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO15.

36.4.110 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22)

Address: 20E_0000h base + 1C8h offset = 20E_01C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA22 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT22. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT22. NOTE: Pad DISP0_DAT22 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA22. 001 **ALT1** — Select signal IPU2_DISP0_DATA22. 010 **ALT2** — Select signal ECSPI1_MISO. - Configure register IOMUXC_ECSPI1_MISO_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_TXFS. - Configure register IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO16.

36.4.111 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23)

Address: 20E_0000h base + 1CCh offset = 20E_01CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_DISP0_DATA23 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad DISP0_DAT23. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: DISP0_DAT23. NOTE: Pad DISP0_DAT23 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_DISP0_DATA23. 001 **ALT1** — Select signal IPU2_DISP0_DATA23. 010 **ALT2** — Select signal ECSPI1_SS0. - Configure register IOMUXC_ECSPI1_SS0_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_RXD. - Configure register IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO17.

36.4.112 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO)

Address: 20E_0000h base + 1D0h offset = 20E_01D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

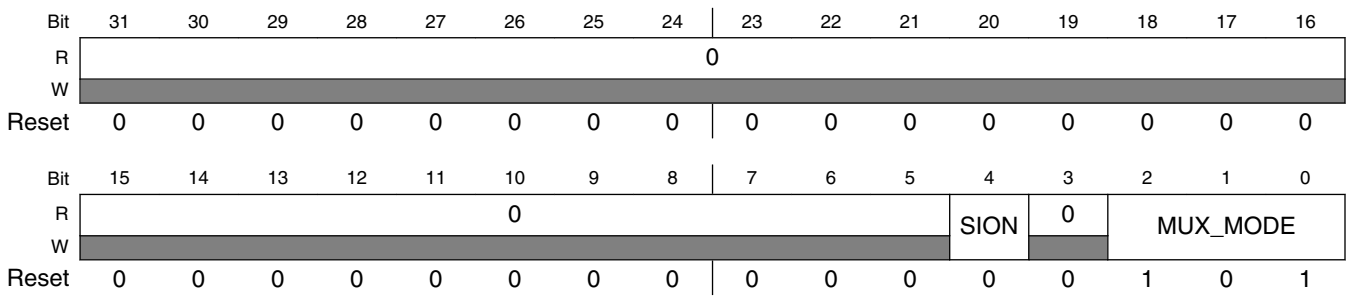
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_MDIO field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_MDIO. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: ENET_MDIO. NOTE: Pad ENET_MDIO is involved in Daisy Chain. 001 **ALT1** — Select signal ENET_MDIO. - Configure register IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal ESAI_RX_CLK. - Configure register IOMUXC_ESAI_RX_CLK_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal ENET_1588_EVENT1_OUT. 101 **ALT5** — Select signal GPIO1_IO22. 110 **ALT6** — Select signal SPDIF_LOCK.

36.4.113 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK)

Address: 20E_0000h base + 1D4h offset = 20E_01D4h



IOMUXC_SW_MUX_CTL_PAD_ENET_REF_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_REF_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: ENET_REF_CLK. NOTE: Pad ENET_REF_CLK is involved in Daisy Chain. 001 **ALT1** — Select signal ENET_TX_CLK. 010 **ALT2** — Select signal ESAI_RX_FS. - Configure register IOMUXC_ESAI_RX_FS_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO23. 110 **ALT6** — Select signal SPDIF_SR_CLK.

36.4.114 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER)

Address: 20E_0000h base + 1D8h offset = 20E_01D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

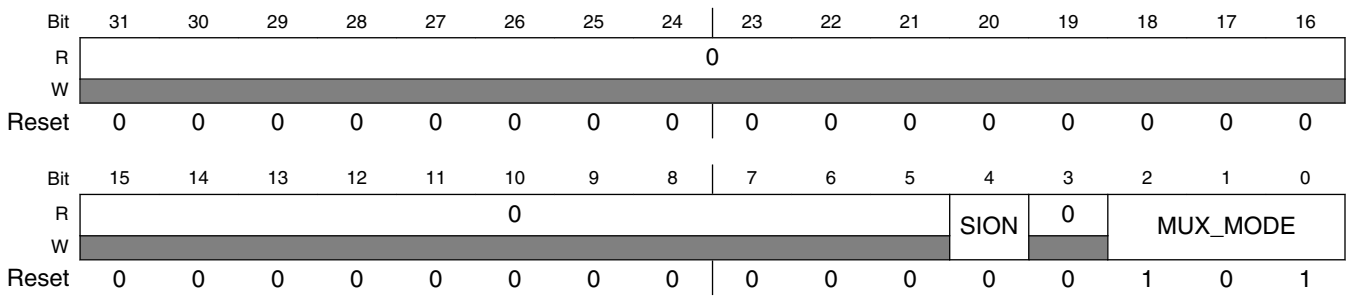
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_RX_ER field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_RX_ER. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: ENET_RX_ER. NOTE: Pad ENET_RX_ER is involved in Daisy Chain. 000 **ALT0** — Select signal USB_OTG_ID. 001 **ALT1** — Select signal ENET_RX_ER. 010 **ALT2** — Select signal ESAI_RX_HF_CLK. - Configure register IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal SPDIF_IN. - Configure register IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal ENET_1588_EVENT2_OUT. 101 **ALT5** — Select signal GPIO1_IO24.

36.4.115 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV)

Address: 20E_0000h base + 1DCh offset = 20E_01DCh



IOMUXC_SW_MUX_CTL_PAD_ENET_CRS_DV field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_CRS_DV. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: ENET_CRS_DV. NOTE: Pad ENET_CRS_DV is involved in Daisy Chain. 001 **ALT1** — Select signal ENET_RX_EN. - Configure register IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal ESAI_TX_CLK. - Configure register IOMUXC_ESAI_TX_CLK_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal SPDIF_EXT_CLK. - Configure register IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO1_IO25.

36.4.116 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1)

Address: 20E_0000h base + 1E0h offset = 20E_01E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

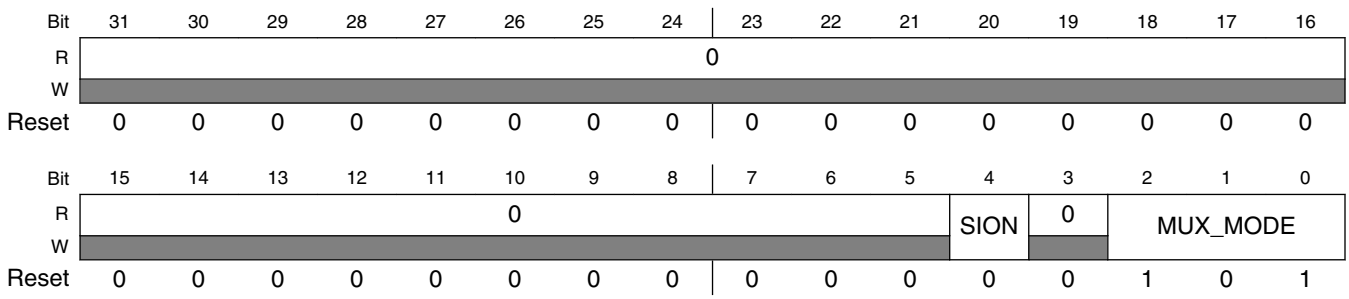
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_RXD1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: ENET_RXD1. NOTE: Pad ENET_RXD1 is involved in Daisy Chain. 000 **ALT0** — Select signal MLB_SIG. - Configure register IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_RX_DATA1. - Configure register IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal ESAI_TX_FS. - Configure register IOMUXC_ESAI_TX_FS_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal ENET_1588_EVENT3_OUT. 101 **ALT5** — Select signal GPIO1_IO26.

36.4.117 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0)

Address: 20E_0000h base + 1E4h offset = 20E_01E4h



IOMUXC_SW_MUX_CTL_PAD_ENET_RX_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_RXD0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: ENET_RXD0. NOTE: Pad ENET_RXD0 is involved in Daisy Chain. 000 **ALT0** — Select signal XTALOSC_OSC32K_32K_OUT. 001 **ALT1** — Select signal ENET_RX_DATA0. - Configure register IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal ESAI_TX_HF_CLK. - Configure register IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal SPDIF_OUT. 101 **ALT5** — Select signal GPIO1_IO27.

36.4.118 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN)

Address: 20E_0000h base + 1E8h offset = 20E_01E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_TX_EN field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_TX_EN. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: ENET_TX_EN. NOTE: Pad ENET_TX_EN is involved in Daisy Chain. 001 **ALT1** — Select signal ENET_TX_EN. 010 **ALT2** — Select signal ESAI_TX3_RX2. - Configure register IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO28.

36.4.119 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1)

Address: 20E_0000h base + 1ECh offset = 20E_01ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_TXD1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: ENET_TXD1. NOTE: Pad ENET_TXD1 is involved in Daisy Chain. 000 **ALT0** — Select signal MLB_CLK. - Configure register IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_TX_DATA1. 010 **ALT2** — Select signal ESAI_TX2_RX3. - Configure register IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal ENET_1588_EVENT0_IN. 101 **ALT5** — Select signal GPIO1_IO29.

36.4.120 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0)

Address: 20E_0000h base + 1F0h offset = 20E_01F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_TX_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_TXD0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: ENET_TXD0. NOTE: Pad ENET_TXD0 is involved in Daisy Chain. 001 **ALT1** — Select signal ENET_TX_DATA0. 010 **ALT2** — Select signal ESAI_TX4_RX1. - Configure register IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO30.

36.4.121 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_ENET_MDC)

Address: 20E_0000h base + 1F4h offset = 20E_01F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_ENET_MDC field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad ENET_MDC. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: ENET_MDC. NOTE: Pad ENET_MDC is involved in Daisy Chain. 000 **ALT0** — Select signal MLB_DATA. - Configure register IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_MDC. 010 **ALT2** — Select signal ESAI_TX5_RX0. - Configure register IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT for mode ALT2. 100 **ALT4** — Select signal ENET_1588_EVENT1_IN. 101 **ALT5** — Select signal GPIO1_IO31.

36.4.122 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL0)

Address: 20E_0000h base + 1F8h offset = 20E_01F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_COL0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_COL0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_COL0. NOTE: Pad KEY_COL0 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_SCLK. - Configure register IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_RX_DATA3. - Configure register IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal AUD5_TXC. - Configure register IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL0. 100 **ALT4** — Select signal UART4_TX_DATA. - Configure register IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO06. 110 **ALT6** — Select signal DCIC1_OUT.

36.4.123 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0)

Address: 20E_0000h base + 1FCh offset = 20E_01FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

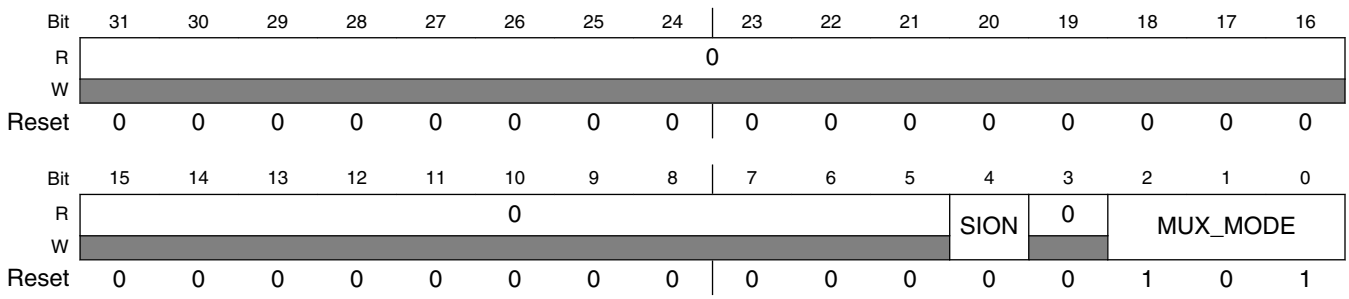
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_ROW0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_ROW0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_ROW0. NOTE: Pad KEY_ROW0 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_MOSI. - Configure register IOMUXC_ECSPI1_MOSI_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_TX_DATA3. 010 **ALT2** — Select signal AUD5_TXD. - Configure register IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW0. 100 **ALT4** — Select signal UART4_RX_DATA. - Configure register IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO07. 110 **ALT6** — Select signal DCIC2_OUT.

36.4.124 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL1)

Address: 20E_0000h base + 200h offset = 20E_0200h



IOMUXC_SW_MUX_CTL_PAD_KEY_COL1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_COL1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_COL1. NOTE: Pad KEY_COL1 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_MISO. - Configure register IOMUXC_ECSPI1_MISO_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_MDIO. - Configure register IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal AUD5_TXFS. - Configure register IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL1. 100 **ALT4** — Select signal UART5_TX_DATA. - Configure register IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO08. 110 **ALT6** — Select signal SD1_VSELECT.

36.4.125 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1)

Address: 20E_0000h base + 204h offset = 20E_0204h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_ROW1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_ROW1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_ROW1. NOTE: Pad KEY_ROW1 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_SS0. - Configure register IOMUXC_ECSPI1_SS0_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_COL. 010 **ALT2** — Select signal AUD5_RXD. - Configure register IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW1. 100 **ALT4** — Select signal UART5_RX_DATA. - Configure register IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO09. 110 **ALT6** — Select signal SD2_VSELECT.

36.4.126 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL2)

Address: 20E_0000h base + 208h offset = 20E_0208h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

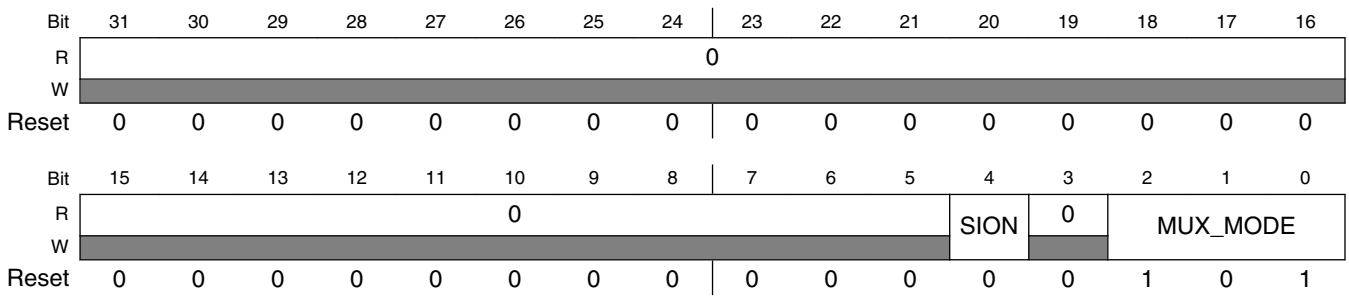
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_COL2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_COL2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_COL2. NOTE: Pad KEY_COL2 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_SS1. - Configure register IOMUXC_ECSPI1_SS1_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_RX_DATA2. - Configure register IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal FLEXCAN1_TX. 011 **ALT3** — Select signal KEY_COL2. 100 **ALT4** — Select signal ENET_MDC. 101 **ALT5** — Select signal GPIO4_IO10. 110 **ALT6** — Select signal USB_H1_PWR_CTL_WAKE.

36.4.127 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2)

Address: 20E_0000h base + 20Ch offset = 20E_020Ch



IOMUXC_SW_MUX_CTL_PAD_KEY_ROW2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_ROW2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_ROW2. NOTE: Pad KEY_ROW2 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_SS2. - Configure register IOMUXC_ECSPI1_SS2_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_TX_DATA2. 010 **ALT2** — Select signal FLEXCAN1_RX. - Configure register IOMUXC_FLEXCAN1_RX_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW2. 100 **ALT4** — Select signal SD2_VSELECT. 101 **ALT5** — Select signal GPIO4_IO11. 110 **ALT6** — Select signal HDMI_TX_CEC_LINE. - Configure register IOMUXC_HDMI_ICECIN_SELECT_INPUT for mode ALT6.

36.4.128 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL3)

Address: 20E_0000h base + 210h offset = 20E_0210h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_COL3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_COL3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_COL3. NOTE: Pad KEY_COL3 is involved in Daisy Chain. 000 **ALT0** — Select signal ECSPI1_SS3. - Configure register IOMUXC_ECSPI1_SS3_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_CRS. 010 **ALT2** — Select signal HDMI_TX_DDC_SCL. - Configure register IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL3. 100 **ALT4** — Select signal I2C2_SCL. - Configure register IOMUXC_I2C2_SCL_IN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO12. 110 **ALT6** — Select signal SPDIF_IN. - Configure register IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT for mode ALT6.

36.4.129 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3)

Address: 20E_0000h base + 214h offset = 20E_0214h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_ROW3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_ROW3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: KEY_ROW3. NOTE: Pad KEY_ROW3 is involved in Daisy Chain. 000 **ALT0** — Select signal XTALOSC_OSC32K_32K_OUT. 001 **ALT1** — Select signal ASRC_EXT_CLK. - Configure register IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal HDMI_TX_DDC_SDA. - Configure register IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW3. 100 **ALT4** — Select signal I2C2_SDA. - Configure register IOMUXC_I2C2_SDA_IN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO13. 110 **ALT6** — Select signal SD1_VSELECT.

36.4.130 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_COL4)

Address: 20E_0000h base + 218h offset = 20E_0218h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

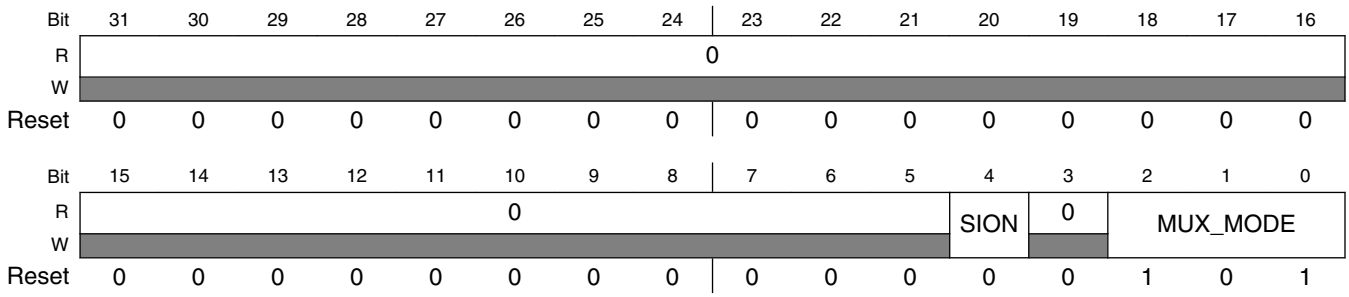
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_KEY_COL4 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_COL4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: KEY_COL4. NOTE: Pad KEY_COL4 is involved in Daisy Chain. 000 **ALT0** — Select signal FLEXCAN2_TX. 001 **ALT1** — Select signal IPU1_SISG4. 010 **ALT2** — Select signal USB_OTG_OC. - Configure register IOMUXC_USB_OTG_OC_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL4. 100 **ALT4** — Select signal UART5_RTS_B. - Configure register IOMUXC_UART5_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO14.

36.4.131 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4)

Address: 20E_0000h base + 21Ch offset = 20E_021Ch



IOMUXC_SW_MUX_CTL_PAD_KEY_ROW4 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad KEY_ROW4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: KEY_ROW4. NOTE: Pad KEY_ROW4 is involved in Daisy Chain. 000 **ALT0** — Select signal FLEXCAN2_RX. - Configure register IOMUXC_FLEXCAN2_RX_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal IPU1_SISG5. 010 **ALT2** — Select signal USB_OTG_PWR. 011 **ALT3** — Select signal KEY_ROW4. 100 **ALT4** — Select signal UART5_CTS_B. - Configure register IOMUXC_UART5_UART_RTS_B_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO4_IO15.

36.4.132 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO00)

Address: 20E_0000h base + 220h offset = 20E_0220h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

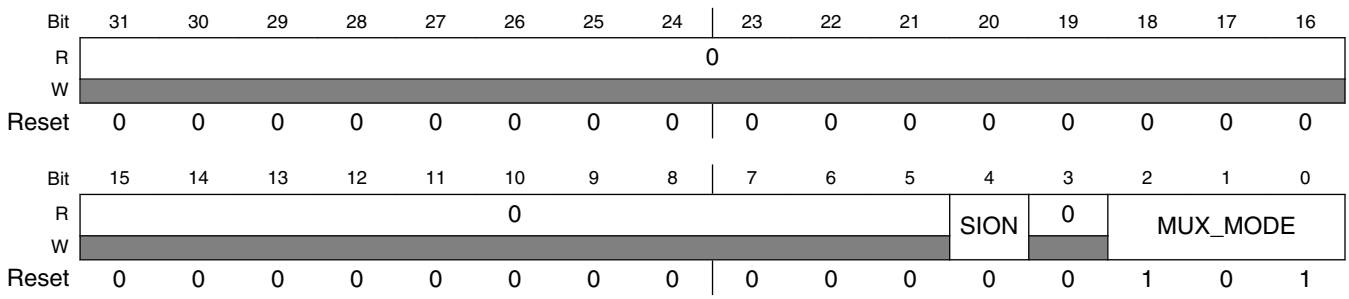
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO00 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_0. NOTE: Pad GPIO_0 is involved in Daisy Chain. 000 **ALT0** — Select signal CCM_CLKO1. 010 **ALT2** — Select signal KEY_COL5. - Configure register IOMUXC_KEY_COL5_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal ASRC_EXT_CLK. - Configure register IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal EPIT1_OUT. 101 **ALT5** — Select signal GPIO1_IO00. 110 **ALT6** — Select signal USB_H1_PWR. 111 **ALT7** — Select signal SNVS_VIO_5.

36.4.133 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO01)

Address: 20E_0000h base + 224h offset = 20E_0224h



IOMUXC_SW_MUX_CTL_PAD_GPIO01 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_1. NOTE: Pad GPIO_1 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_RX_CLK. - Configure register IOMUXC_ESAI_RX_CLK_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal WDOG2_B. 010 **ALT2** — Select signal KEY_ROW5. - Configure register IOMUXC_KEY_ROW5_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal USB_OTG_ID. 100 **ALT4** — Select signal PWM2_OUT. 101 **ALT5** — Select signal GPIO1_IO01. 110 **ALT6** — Select signal SD1_CD_B.

36.4.134 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO09)

Address: 20E_0000h base + 228h offset = 20E_0228h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

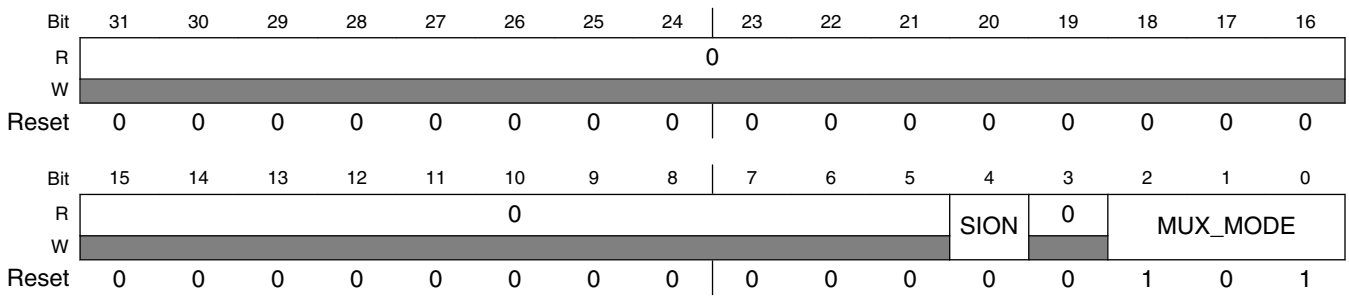
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO09 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_9. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_9. NOTE: Pad GPIO_9 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_RX_FS. - Configure register IOMUXC_ESAI_RX_FS_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal WDOG1_B. 010 **ALT2** — Select signal KEY_COL6. - Configure register IOMUXC_KEY_COL6_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal CCM_REF_EN_B. 100 **ALT4** — Select signal PWM1_OUT. 101 **ALT5** — Select signal GPIO1_IO09. 110 **ALT6** — Select signal SD1_WP. - Configure register IOMUXC_USDHC1_WP_ON_SELECT_INPUT for mode ALT6.

36.4.135 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO03)

Address: 20E_0000h base + 22Ch offset = 20E_022Ch



IOMUXC_SW_MUX_CTL_PAD_GPIO03 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_3. NOTE: Pad GPIO_3 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_RX_HF_CLK. - Configure register IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT for mode ALT0. 010 **ALT2** — Select signal I2C3_SCL. - Configure register IOMUXC_I2C3_SCL_IN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal XTALOSC_REF_CLK_24M. 100 **ALT4** — Select signal CCM_CLKO2. 101 **ALT5** — Select signal GPIO1_IO03. 110 **ALT6** — Select signal USB_H1_OC. - Configure register IOMUXC_USB_H1_OC_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal MLB_CLK. - Configure register IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT for mode ALT7.

36.4.136 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO06)

Address: 20E_0000h base + 230h offset = 20E_0230h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

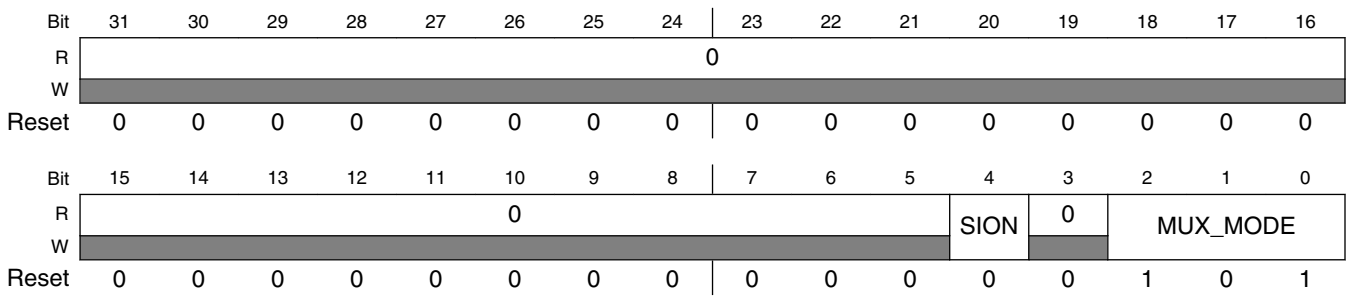
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO06 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: GPIO_6. NOTE: Pad GPIO_6 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX_CLK. - Configure register IOMUXC_ESAI_TX_CLK_SELECT_INPUT for mode ALT0. 010 **ALT2** — Select signal I2C3_SDA. - Configure register IOMUXC_I2C3_SDA_IN_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO06. 110 **ALT6** — Select signal SD2_LCTL. 111 **ALT7** — Select signal MLB_SIG. - Configure register IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT for mode ALT7.

36.4.137 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO02)

Address: 20E_0000h base + 234h offset = 20E_0234h



IOMUXC_SW_MUX_CTL_PAD_GPIO02 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: GPIO_2. NOTE: Pad GPIO_2 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX_FS. - Configure register IOMUXC_ESAI_TX_FS_SELECT_INPUT for mode ALT0. 010 **ALT2** — Select signal KEY_ROW6. - Configure register IOMUXC_KEY_ROW6_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO02. 110 **ALT6** — Select signal SD2_WP. 111 **ALT7** — Select signal MLB_DATA. - Configure register IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT for mode ALT7.

36.4.138 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO04)

Address: 20E_0000h base + 238h offset = 20E_0238h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

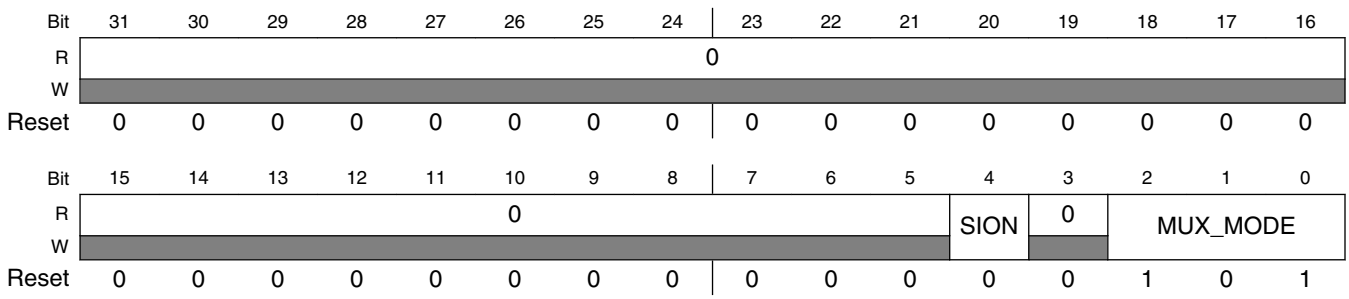
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: GPIO_4. NOTE: Pad GPIO_4 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX_HF_CLK. - Configure register IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT for mode ALT0. 010 **ALT2** — Select signal KEY_COL7. - Configure register IOMUXC_KEY_COL7_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO1_IO04. 110 **ALT6** — Select signal SD2_CD_B.

36.4.139 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO05)

Address: 20E_0000h base + 23Ch offset = 20E_023Ch



IOMUXC_SW_MUX_CTL_PAD_GPIO05 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: GPIO_5. NOTE: Pad GPIO_5 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX2_RX3. - Configure register IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT for mode ALT0. 010 **ALT2** — Select signal KEY_ROW7. - Configure register IOMUXC_KEY_ROW7_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal CCM_CLKO1. 101 **ALT5** — Select signal GPIO1_IO05. 110 **ALT6** — Select signal I2C3_SCL. - Configure register IOMUXC_I2C3_SCL_IN_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal ARM_EVENTI.

36.4.140 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO07)

Address: 20E_0000h base + 240h offset = 20E_0240h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO07 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: GPIO_7. NOTE: Pad GPIO_7 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX4_RX1. - Configure register IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ECSPI5_RDY. 010 **ALT2** — Select signal EPIT1_OUT. 011 **ALT3** — Select signal FLEXCAN1_TX. 100 **ALT4** — Select signal UART2_TX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO1_IO07. 110 **ALT6** — Select signal SPDIF_LOCK. 111 **ALT7** — Select signal USB_OTG_HOST_MODE.

36.4.141 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO08)

Address: 20E_0000h base + 244h offset = 20E_0244h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO08 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_8. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: GPIO_8. NOTE: Pad GPIO_8 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX5_RX0. - Configure register IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal XTALOSC_REF_CLK_32K. 010 **ALT2** — Select signal EPIT2_OUT. 011 **ALT3** — Select signal FLEXCAN1_RX. - Configure register IOMUXC_FLEXCAN1_RX_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal UART2_RX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO1_IO08. 110 **ALT6** — Select signal SPDIF_SR_CLK. 111 **ALT7** — Select signal USB_OTG_PWR_CTL_WAKE.

36.4.142 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO16)

Address: 20E_0000h base + 248h offset = 20E_0248h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

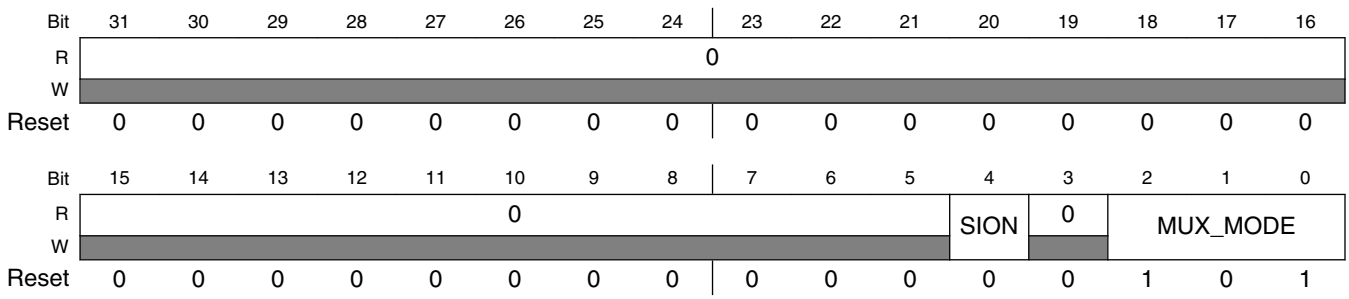
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO16 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_16. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 8 iomux modes to be used for pad: GPIO_16. NOTE: Pad GPIO_16 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX3_RX2. - Configure register IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_1588_EVENT2_IN. 010 **ALT2** — Select signal ENET_REF_CLK. - Configure register IOMUXC_ENET_REF_CLK_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal SD1_LCTL. 100 **ALT4** — Select signal SPDIF_IN. - Configure register IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO7_IO11. 110 **ALT6** — Select signal I2C3_SDA. - Configure register IOMUXC_I2C3_SDA_IN_SELECT_INPUT for mode ALT6. 111 **ALT7** — Select signal JTAG_DE_B.

36.4.143 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO17)

Address: 20E_0000h base + 24Ch offset = 20E_024Ch



IOMUXC_SW_MUX_CTL_PAD_GPIO17 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_17. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: GPIO_17. NOTE: Pad GPIO_17 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX0. - Configure register IOMUXC_ESAI_SDO0_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_1588_EVENT3_IN. 010 **ALT2** — Select signal CCM_PMIC_READY. - Configure register IOMUXC_CCM_PMIC_READY_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal SDMA_EXT_EVENT0. - Configure register IOMUXC_SDMA_EVENTS14_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal SPDIF_OUT. 101 **ALT5** — Select signal GPIO7_IO12.

36.4.144 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO18)

Address: 20E_0000h base + 250h offset = 20E_0250h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

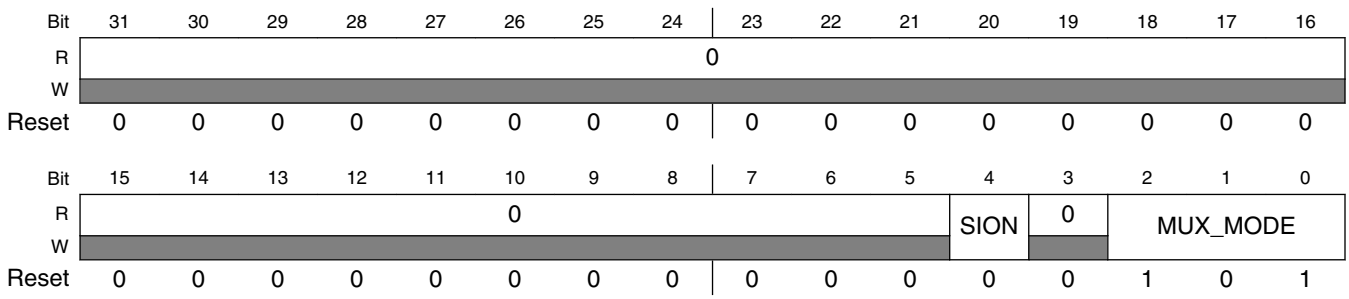
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_GPIO18 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_18. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_18. NOTE: Pad GPIO_18 is involved in Daisy Chain. 000 **ALT0** — Select signal ESAI_TX1. - Configure register IOMUXC_ESAI_SDO1_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_RX_CLK. - Configure register IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal SD3_VSELECT. 011 **ALT3** — Select signal SDMA_EXT_EVENT1. - Configure register IOMUXC_SDMA_EVENTS15_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal ASRC_EXT_CLK. - Configure register IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO7_IO13. 110 **ALT6** — Select signal SNVS_VIO_5_CTL.

36.4.145 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_GPIO19)

Address: 20E_0000h base + 254h offset = 20E_0254h



IOMUXC_SW_MUX_CTL_PAD_GPIO19 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad GPIO_19. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: GPIO_19. NOTE: Pad GPIO_19 is involved in Daisy Chain. 000 **ALT0** — Select signal KEY_COL5. - Configure register IOMUXC_KEY_COL5_SELECT_INPUT for mode ALT0. 001 **ALT1** — Select signal ENET_1588_EVENT0_OUT. 010 **ALT2** — Select signal SPDIF_OUT. 011 **ALT3** — Select signal CCM_CLKO1. 100 **ALT4** — Select signal ECSPI1_RDY. 101 **ALT5** — Select signal GPIO4_IO05. 110 **ALT6** — Select signal ENET_TX_ER.

36.4.146 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK)

Address: 20E_0000h base + 258h offset = 20E_0258h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_PIXCLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_PIXCLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: CSI0_PIXCLK. 000 **ALT0** — Select signal IPU1_CSI0_PIXCLK. 101 **ALT5** — Select signal GPIO5_IO18. 111 **ALT7** — Select signal ARM_EVENTO.

36.4.147 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC)

Address: 20E_0000h base + 25Ch offset = 20E_025Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_HSYNC field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_MCLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: CSI0_MCLK. 000 **ALT0** — Select signal IPU1_CSI0_HSYNC. 011 **ALT3** — Select signal CCM_CLKO1. 101 **ALT5** — Select signal GPIO5_IO19. 111 **ALT7** — Select signal ARM_TRACE_CTL.

36.4.148 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN)

Address: 20E_0000h base + 260h offset = 20E_0260h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA_EN field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DATA_EN. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: CSI0_DATA_EN. 000 **ALT0** — Select signal IPU1_CSI0_DATA_EN. 001 **ALT1** — Select signal EIM_DATA00. 101 **ALT5** — Select signal GPIO5_IO20. 111 **ALT7** — Select signal ARM_TRACE_CLK.

36.4.149 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC)

Address: 20E_0000h base + 264h offset = 20E_0264h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_VSYNC field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_VSYNC. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: CSI0_VSYNC. 000 **ALT0** — Select signal IPU1_CSI0_VSYNC. 001 **ALT1** — Select signal EIM_DATA01. 101 **ALT5** — Select signal GPIO5_IO21. 111 **ALT7** — Select signal ARM_TRACE00.

36.4.150 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04)

Address: 20E_0000h base + 268h offset = 20E_0268h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

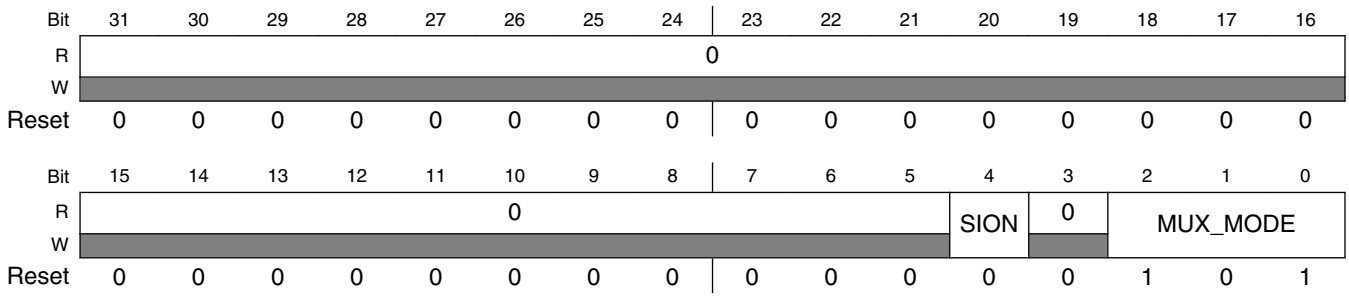
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT4. NOTE: Pad CSI0_DAT4 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA04. 001 **ALT1** — Select signal EIM_DATA02. 010 **ALT2** — Select signal ECSPI1_SCLK. - Configure register IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL5. - Configure register IOMUXC_KEY_COL5_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD3_TXC. 101 **ALT5** — Select signal GPIO5_IO22. 111 **ALT7** — Select signal ARM_TRACE01.

36.4.151 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05)

Address: 20E_0000h base + 26Ch offset = 20E_026Ch



IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA05 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT5. NOTE: Pad CSI0_DAT5 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA05. 001 **ALT1** — Select signal EIM_DATA03. 010 **ALT2** — Select signal ECSPI1_MOSI. - Configure register IOMUXC_ECSPI1_MOSI_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW5. - Configure register IOMUXC_KEY_ROW5_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD3_TXD. 101 **ALT5** — Select signal GPIO5_IO23. 111 **ALT7** — Select signal ARM_TRACE02.

36.4.152 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06)

Address: 20E_0000h base + 270h offset = 20E_0270h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA06 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT6. NOTE: Pad CSI0_DAT6 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA06. 001 **ALT1** — Select signal EIM_DATA04. 010 **ALT2** — Select signal ECSPI1_MISO. - Configure register IOMUXC_ECSPI1_MISO_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL6. - Configure register IOMUXC_KEY_COL6_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD3_TXFS. 101 **ALT5** — Select signal GPIO5_IO24. 111 **ALT7** — Select signal ARM_TRACE03.

36.4.153 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07)

Address: 20E_0000h base + 274h offset = 20E_0274h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA07 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT7. NOTE: Pad CSI0_DAT7 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA07. 001 **ALT1** — Select signal EIM_DATA05. 010 **ALT2** — Select signal ECSPI1_SS0. - Configure register IOMUXC_ECSPI1_SS0_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW6. - Configure register IOMUXC_KEY_ROW6_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal AUD3_RXD. 101 **ALT5** — Select signal GPIO5_IO25. 111 **ALT7** — Select signal ARM_TRACE04.

36.4.154 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08)

Address: 20E_0000h base + 278h offset = 20E_0278h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

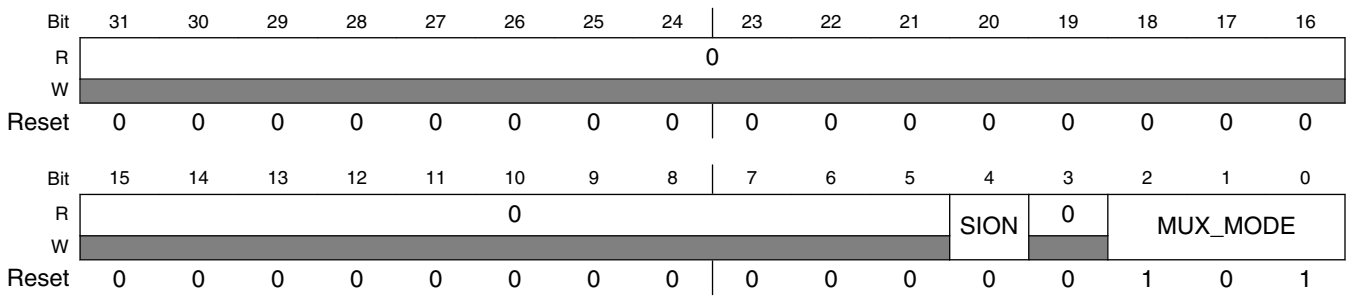
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA08 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT8. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT8. NOTE: Pad CSI0_DAT8 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA08. 001 **ALT1** — Select signal EIM_DATA06. 010 **ALT2** — Select signal ECSPI2_SCLK. - Configure register IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_COL7. - Configure register IOMUXC_KEY_COL7_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal I2C1_SDA. - Configure register IOMUXC_I2C1_SDA_IN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO26. 111 **ALT7** — Select signal ARM_TRACE05.

36.4.155 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09)

Address: 20E_0000h base + 27Ch offset = 20E_027Ch



IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA09 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT9. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: CSI0_DAT9. NOTE: Pad CSI0_DAT9 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA09. 001 **ALT1** — Select signal EIM_DATA07. 010 **ALT2** — Select signal ECSPI2_MOSI. - Configure register IOMUXC_ECSPI2_MOSI_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal KEY_ROW7. - Configure register IOMUXC_KEY_ROW7_SELECT_INPUT for mode ALT3. 100 **ALT4** — Select signal I2C1_SCL. - Configure register IOMUXC_I2C1_SCL_IN_SELECT_INPUT for mode ALT4. 101 **ALT5** — Select signal GPIO5_IO27. 111 **ALT7** — Select signal ARM_TRACE06.

36.4.156 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10)

Address: 20E_0000h base + 280h offset = 20E_0280h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

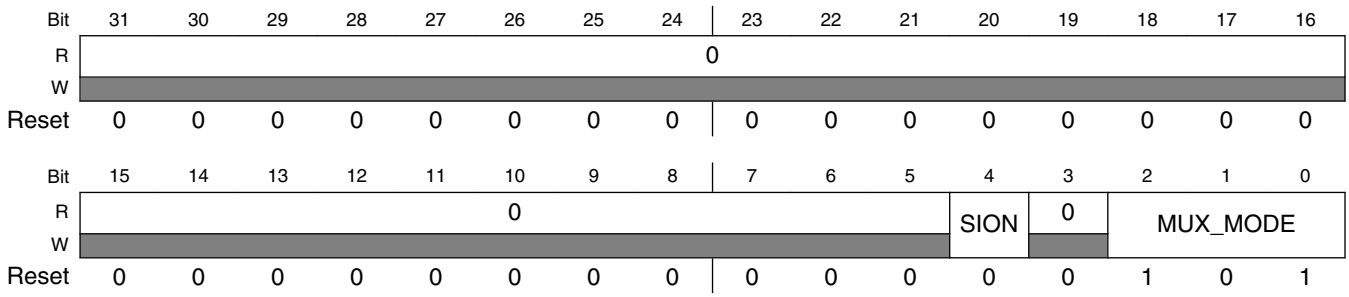
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA10 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT10. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: CSI0_DAT10. NOTE: Pad CSI0_DAT10 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA10. 001 **ALT1** — Select signal AUD3_RXC. 010 **ALT2** — Select signal ECSPI2_MISO. - Configure register IOMUXC_ECSPI2_MISO_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal UART1_TX_DATA. - Configure register IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO28. 111 **ALT7** — Select signal ARM_TRACE07.

36.4.157 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11)

Address: 20E_0000h base + 284h offset = 20E_0284h



IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA11 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT11. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: CSI0_DAT11. NOTE: Pad CSI0_DAT11 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA11. 001 **ALT1** — Select signal AUD3_RXFS. 010 **ALT2** — Select signal ECSPI2_SS0. - Configure register IOMUXC_ECSPI2_SS0_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal UART1_RX_DATA. - Configure register IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO29. 111 **ALT7** — Select signal ARM_TRACE08.

36.4.158 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12)

Address: 20E_0000h base + 288h offset = 20E_0288h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA12 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT12. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT12. NOTE: Pad CSI0_DAT12 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA12. 001 **ALT1** — Select signal EIM_DATA08. 011 **ALT3** — Select signal UART4_TX_DATA. - Configure register IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO30. 111 **ALT7** — Select signal ARM_TRACE09.

36.4.159 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13)

Address: 20E_0000h base + 28Ch offset = 20E_028Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA13 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT13. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT13. NOTE: Pad CSI0_DAT13 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA13. 001 **ALT1** — Select signal EIM_DATA09. 011 **ALT3** — Select signal UART4_RX_DATA. - Configure register IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO5_IO31. 111 **ALT7** — Select signal ARM_TRACE10.

36.4.160 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14)

Address: 20E_0000h base + 290h offset = 20E_0290h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA14 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT14. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT14. NOTE: Pad CSI0_DAT14 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA14. 001 **ALT1** — Select signal EIM_DATA10. 011 **ALT3** — Select signal UART5_TX_DATA. - Configure register IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO00. 111 **ALT7** — Select signal ARM_TRACE11.

36.4.161 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15)

Address: 20E_0000h base + 294h offset = 20E_0294h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA15 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT15. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT15. NOTE: Pad CSI0_DAT15 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA15. 001 **ALT1** — Select signal EIM_DATA11. 011 **ALT3** — Select signal UART5_RX_DATA. - Configure register IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO01. 111 **ALT7** — Select signal ARM_TRACE12.

36.4.162 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16)

Address: 20E_0000h base + 298h offset = 20E_0298h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA16 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT16. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT16. NOTE: Pad CSI0_DAT16 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA16. 001 **ALT1** — Select signal EIM_DATA12. 011 **ALT3** — Select signal UART4_RTS_B. - Configure register IOMUXC_UART4_UART_RTS_B_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO02. 111 **ALT7** — Select signal ARM_TRACE13.

36.4.163 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17)

Address: 20E_0000h base + 29Ch offset = 20E_029Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA17 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT17. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT17. NOTE: Pad CSI0_DAT17 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA17. 001 **ALT1** — Select signal EIM_DATA13. 011 **ALT3** — Select signal UART4_CTS_B. - Configure register IOMUXC_UART4_UART_RTS_B_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO03. 111 **ALT7** — Select signal ARM_TRACE14.

36.4.164 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18)

Address: 20E_0000h base + 2A0h offset = 20E_02A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA18 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT18. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: CSI0_DAT18. NOTE: Pad CSI0_DAT18 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA18. 001 **ALT1** — Select signal EIM_DATA14. 011 **ALT3** — Select signal UART5_RTS_B. - Configure register IOMUXC_UART5_UART_RTS_B_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO04. 111 **ALT7** — Select signal ARM_TRACE15.

36.4.165 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19)

Address: 20E_0000h base + 2A4h offset = 20E_02A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_CSI0_DATA19 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad CSI0_DAT19. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: CSI0_DAT19. NOTE: Pad CSI0_DAT19 is involved in Daisy Chain. 000 **ALT0** — Select signal IPU1_CSI0_DATA19. 001 **ALT1** — Select signal EIM_DATA15. 011 **ALT3** — Select signal UART5_CTS_B. - Configure register IOMUXC_UART5_UART_RTS_B_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO6_IO05.

36.4.166 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7)

Address: 20E_0000h base + 2A8h offset = 20E_02A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

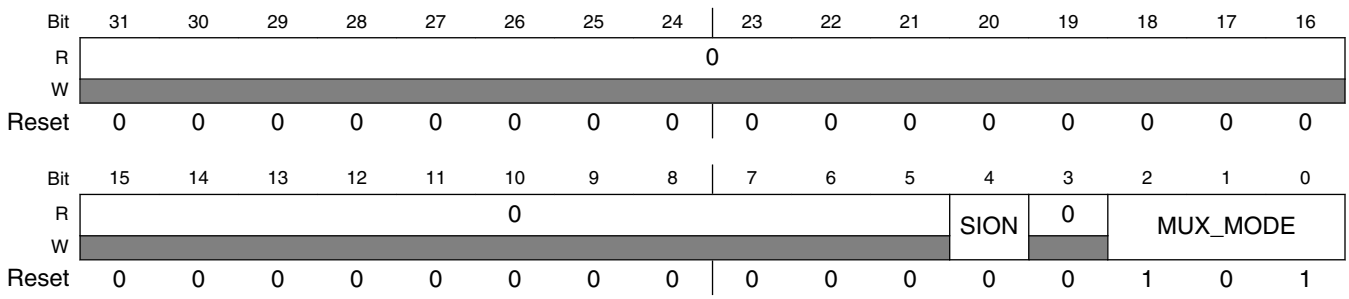
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_DATA7 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_DAT7. NOTE: Pad SD3_DAT7 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA7. 001 **ALT1** — Select signal UART1_TX_DATA. - Configure register IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO17.

36.4.167 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6)

Address: 20E_0000h base + 2ACh offset = 20E_02ACh



IOMUXC_SW_MUX_CTL_PAD_SD3_DATA6 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_DAT6. NOTE: Pad SD3_DAT6 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA6. 001 **ALT1** — Select signal UART1_RX_DATA. - Configure register IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO6_IO18.

36.4.168 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5)

Address: 20E_0000h base + 2B0h offset = 20E_02B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

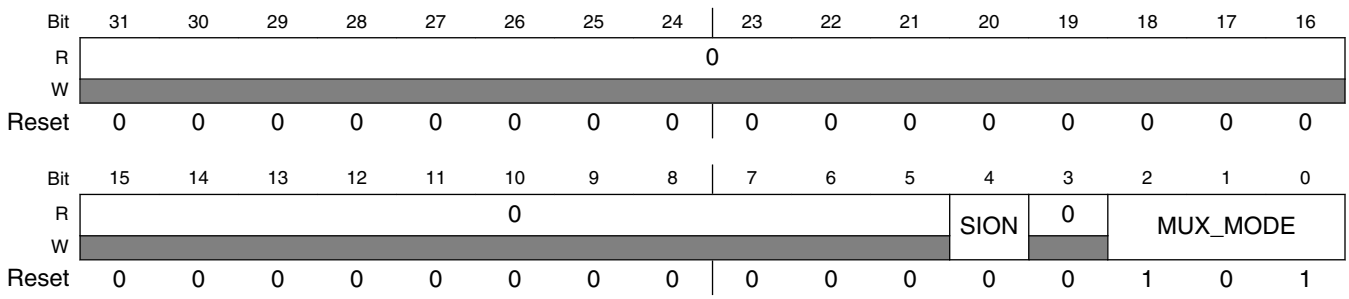
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_DATA5 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_DAT5. NOTE: Pad SD3_DAT5 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA5. 001 **ALT1** — Select signal UART2_TX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO7_IO00.

36.4.169 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4)

Address: 20E_0000h base + 2B4h offset = 20E_02B4h



IOMUXC_SW_MUX_CTL_PAD_SD3_DATA4 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_DAT4. NOTE: Pad SD3_DAT4 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA4. 001 **ALT1** — Select signal UART2_RX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO7_IO01.

36.4.170 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_CMD)

Address: 20E_0000h base + 2B8h offset = 20E_02B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

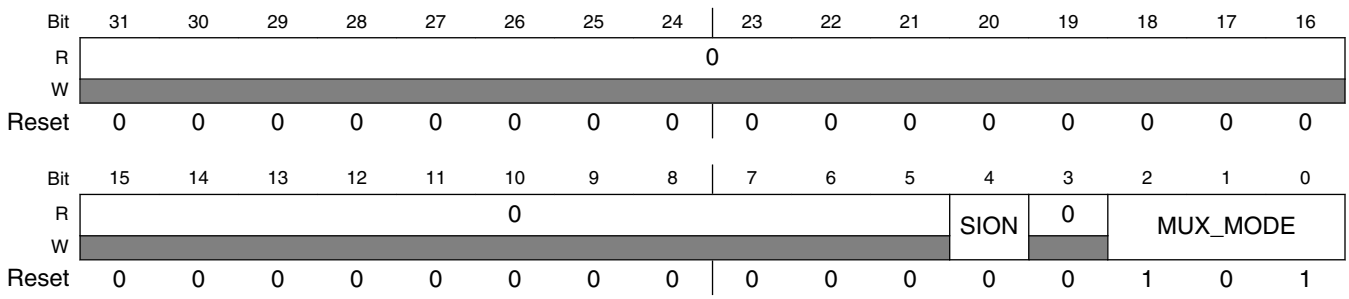
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_CMD field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_CMD. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD3_CMD. NOTE: Pad SD3_CMD is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_CMD. 001 **ALT1** — Select signal UART2_CTS_B. - Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal FLEXCAN1_TX. 101 **ALT5** — Select signal GPIO7_IO02.

36.4.171 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_CLK)

Address: 20E_0000h base + 2BCh offset = 20E_02BCh



IOMUXC_SW_MUX_CTL_PAD_SD3_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD3_CLK. NOTE: Pad SD3_CLK is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_CLK. 001 **ALT1** — Select signal UART2_RTS_B. - Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal FLEXCAN1_RX. - Configure register IOMUXC_FLEXCAN1_RX_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO7_IO03.

36.4.172 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0)

Address: 20E_0000h base + 2C0h offset = 20E_02C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

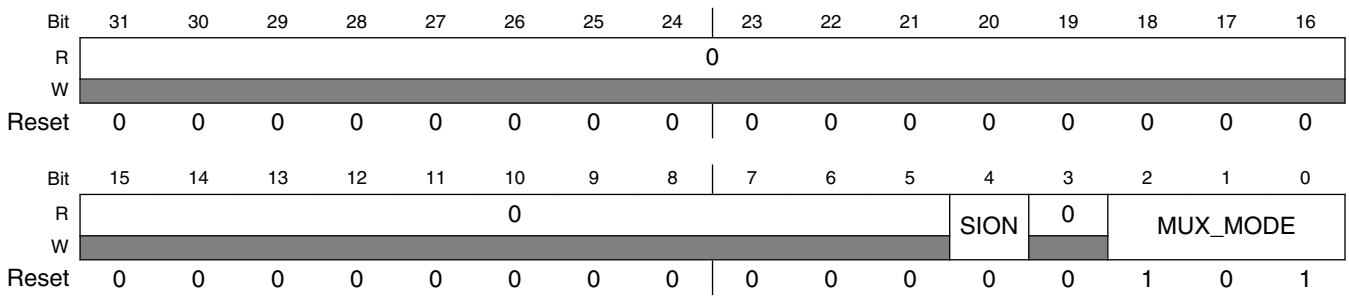
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD3_DAT0. NOTE: Pad SD3_DAT0 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA0. 001 **ALT1** — Select signal UART1_CTS_B. - Configure register IOMUXC_UART1_UART_RTS_B_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal FLEXCAN2_TX. 101 **ALT5** — Select signal GPIO7_IO04.

36.4.173 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1)

Address: 20E_0000h base + 2C4h offset = 20E_02C4h



IOMUXC_SW_MUX_CTL_PAD_SD3_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD3_DAT1. NOTE: Pad SD3_DAT1 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA1. 001 **ALT1** — Select signal UART1_RTS_B. - Configure register IOMUXC_UART1_UART_RTS_B_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal FLEXCAN2_RX. - Configure register IOMUXC_FLEXCAN2_RX_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO7_IO05.

36.4.174 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA2)

Address: 20E_0000h base + 2C8h offset = 20E_02C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_DATA2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 2 iomux modes to be used for pad: SD3_DAT2. 000 **ALT0** — Select signal SD3_DATA2. 101 **ALT5** — Select signal GPIO7_IO06.

36.4.175 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3)

Address: 20E_0000h base + 2CCh offset = 20E_02CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_DATA3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_DAT3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_DAT3. NOTE: Pad SD3_DAT3 is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_DATA3. 001 **ALT1** — Select signal UART3_CTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO7_IO07.

36.4.176 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD3_RESET)

Address: 20E_0000h base + 2D0h offset = 20E_02D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD3_RESET field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD3_RST. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).

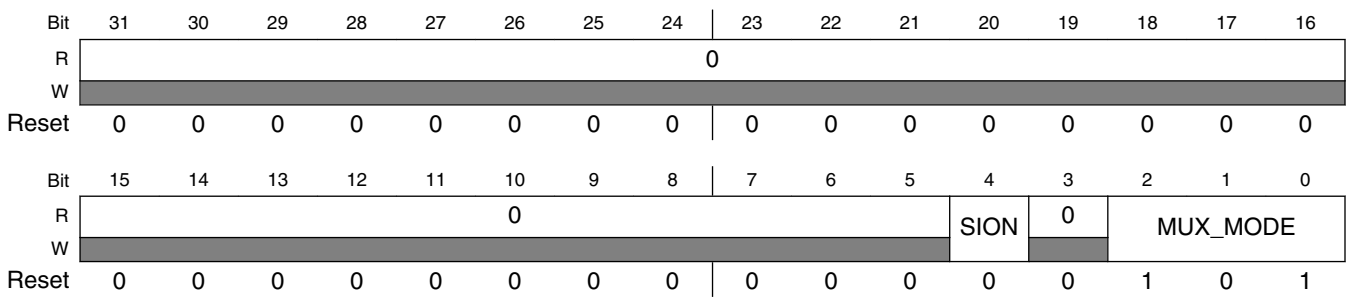
Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD3_RESET field descriptions (continued)

Field	Description
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD3_RST. NOTE: Pad SD3_RST is involved in Daisy Chain. 000 **ALT0** — Select signal SD3_RESET. 001 **ALT1** — Select signal UART3_RTS_B. - Configure register IOMUXC_UART3_UART_RTS_B_SELECT_INPUT for mode ALT1. 101 **ALT5** — Select signal GPIO7_IO08.

36.4.177 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CLE)

Address: 20E_0000h base + 2D4h offset = 20E_02D4h

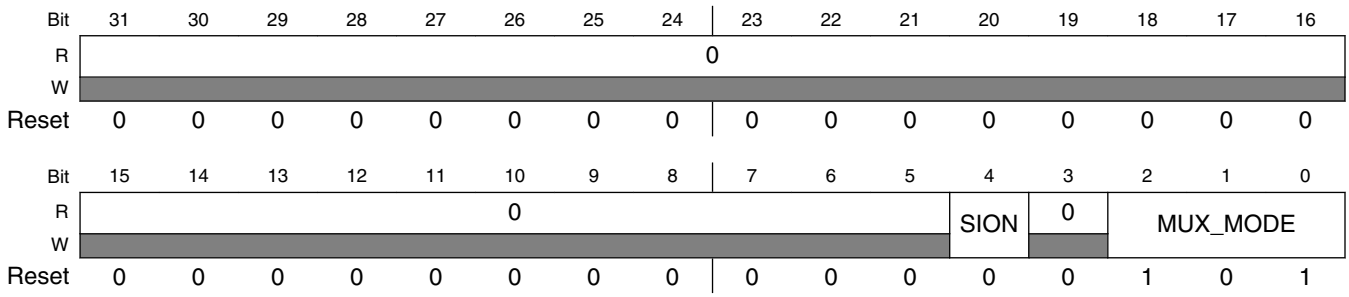


IOMUXC_SW_MUX_CTL_PAD_NAND_CLE field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_CLE. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_CLE. 000 **ALT0** — Select signal NAND_CLE. 001 **ALT1** — Select signal IPU2_SISG4. 101 **ALT5** — Select signal GPIO6_IO07.

36.4.178 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_ALE)

Address: 20E_0000h base + 2D8h offset = 20E_02D8h



IOMUXC_SW_MUX_CTL_PAD_NAND_ALE field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_ALE. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_ALE. 000 **ALT0** — Select signal NAND_ALE. 001 **ALT1** — Select signal SD4_RESET. 101 **ALT5** — Select signal GPIO6_IO08.

36.4.179 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B)

Address: 20E_0000h base + 2DCh offset = 20E_02DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_WP_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_WP_B. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_WP_B. 000 **ALT0** — Select signal NAND_WP_B. 001 **ALT1** — Select signal IPU2_SISG5. 101 **ALT5** — Select signal GPIO6_IO09.

36.4.180 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B)

Address: 20E_0000h base + 2E0h offset = 20E_02E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_READY_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_RB0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_RB0. 000 **ALT0** — Select signal NAND_READY_B. 001 **ALT1** — Select signal IPU2_DI0_PIN01. 101 **ALT5** — Select signal GPIO6_IO10.

36.4.181 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS0_B)

Address: 20E_0000h base + 2E4h offset = 20E_02E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_CS0_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_CS0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 2 iomux modes to be used for pad: NANDF_CS0. 000 **ALT0** — Select signal NAND_CE0_B. 101 **ALT5** — Select signal GPIO6_IO11.

36.4.182 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B)

Address: 20E_0000h base + 2E8h offset = 20E_02E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

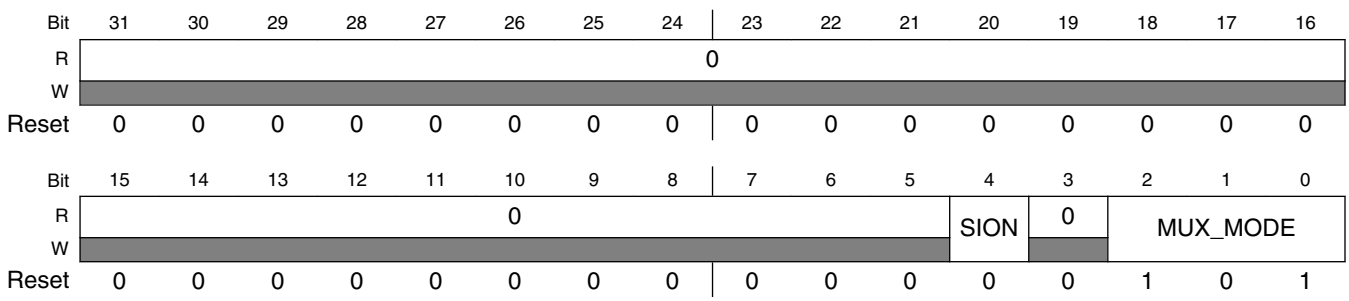
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_CS1_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_CS1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: NANDF_CS1. 000 **ALT0** — Select signal NAND_CE1_B. 001 **ALT1** — Select signal SD4_VSELECT. 010 **ALT2** — Select signal SD3_VSELECT. 101 **ALT5** — Select signal GPIO6_IO14.

36.4.183 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B)

Address: 20E_0000h base + 2ECh offset = 20E_02ECh



IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_CS2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_NAND_CS2_B field descriptions (continued)

Field	Description
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: NANDF_CS2. NOTE: Pad NANDF_CS2 is involved in Daisy Chain. 000 **ALT0** — Select signal NAND_CE2_B. 001 **ALT1** — Select signal IPU1_SISG0. 010 **ALT2** — Select signal ESAI_TX0. - Configure register IOMUXC_ESAI_SDO0_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal EIM_CRE. 100 **ALT4** — Select signal CCM_CLKO2. 101 **ALT5** — Select signal GPIO6_IO15. 110 **ALT6** — Select signal IPU2_SISG0.

36.4.184 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B)

Address: 20E_0000h base + 2F0h offset = 20E_02F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_CS3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 6 iomux modes to be used for pad: NANDF_CS3. NOTE: Pad NANDF_CS3 is involved in Daisy Chain.

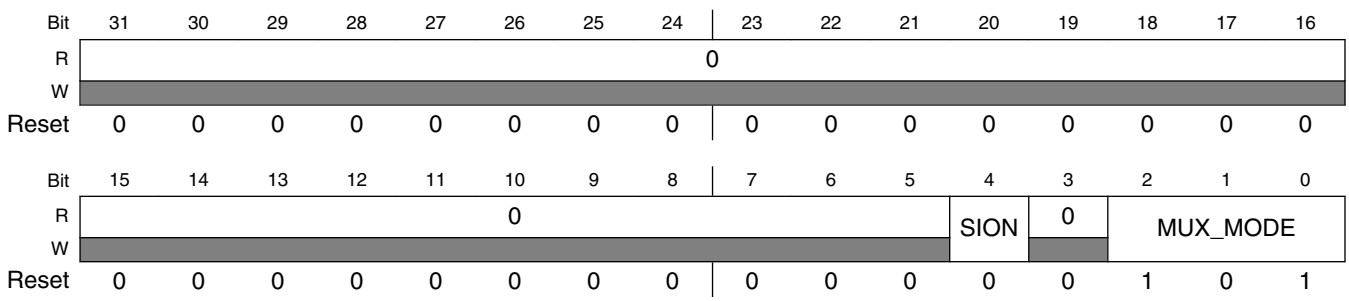
Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_NAND_CS3_B field descriptions (continued)

Field	Description
	000 **ALT0** — Select signal NAND_CE3_B. 001 **ALT1** — Select signal IPU1_SISG1. 010 **ALT2** — Select signal ESAI_TX1. - Configure register IOMUXC_ESAI_SDO1_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal EIM_ADDR26. 101 **ALT5** — Select signal GPIO6_IO16. 110 **ALT6** — Select signal IPU2_SISG1.

36.4.185 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_CMD)

Address: 20E_0000h base + 2F4h offset = 20E_02F4h



IOMUXC_SW_MUX_CTL_PAD_SD4_CMD field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_CMD. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD4_CMD. NOTE: Pad SD4_CMD is involved in Daisy Chain. 000 **ALT0** — Select signal SD4_CMD. 001 **ALT1** — Select signal NAND_RE_B. 010 **ALT2** — Select signal UART3_TX_DATA. - Configure register IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO7_IO09.

36.4.186 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_CLK)

Address: 20E_0000h base + 2F8h offset = 20E_02F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

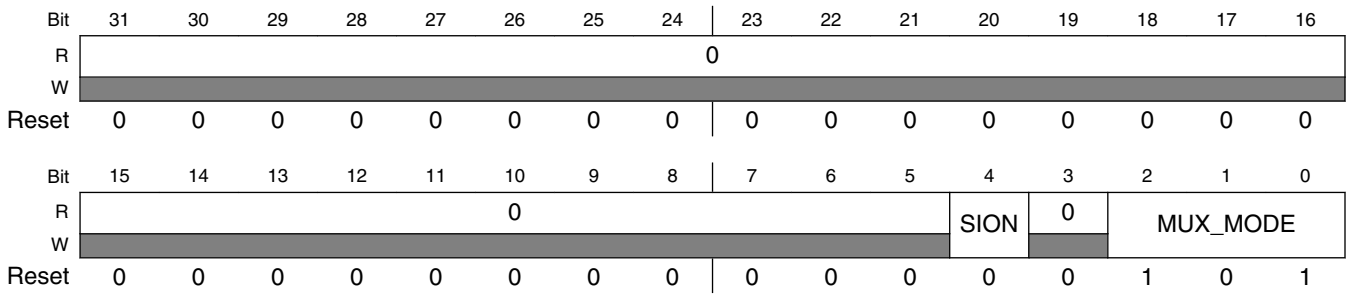
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD4_CLK. NOTE: Pad SD4_CLK is involved in Daisy Chain. 000 **ALT0** — Select signal SD4_CLK. 001 **ALT1** — Select signal NAND_WE_B. 010 **ALT2** — Select signal UART3_RX_DATA. - Configure register IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO7_IO10.

36.4.187 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00)

Address: 20E_0000h base + 2FCh offset = 20E_02FCh



IOMUXC_SW_MUX_CTL_PAD_NAND_DATA00 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D0. 000 **ALT0** — Select signal NAND_DATA00. 001 **ALT1** — Select signal SD1_DATA4. 101 **ALT5** — Select signal GPIO2_IO00.

36.4.188 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01)

Address: 20E_0000h base + 300h offset = 20E_0300h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_DATA01 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D1. 000 **ALT0** — Select signal NAND_DATA01. 001 **ALT1** — Select signal SD1_DATA5. 101 **ALT5** — Select signal GPIO2_IO01.

36.4.189 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02)

Address: 20E_0000h base + 304h offset = 20E_0304h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_DATA02 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D2. 000 **ALT0** — Select signal NAND_DATA02. 001 **ALT1** — Select signal SD1_DATA6. 101 **ALT5** — Select signal GPIO2_IO02.

36.4.190 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03)

Address: 20E_0000h base + 308h offset = 20E_0308h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

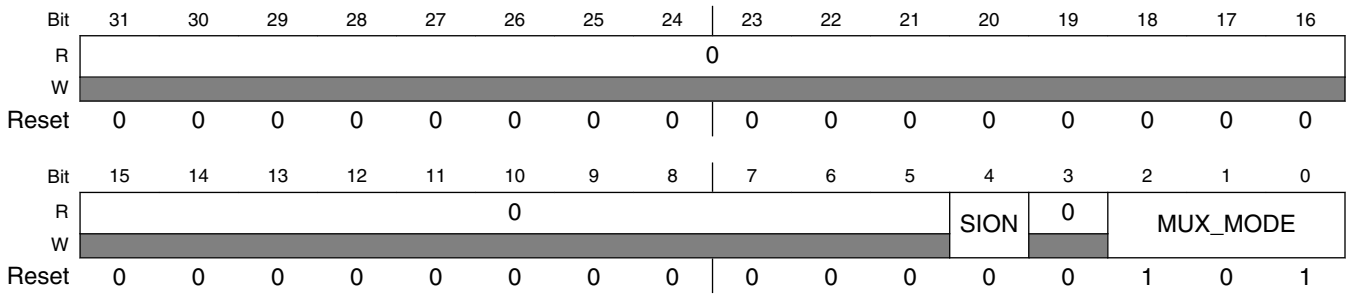
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_DATA03 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D3. 000 **ALT0** — Select signal NAND_DATA03. 001 **ALT1** — Select signal SD1_DATA7. 101 **ALT5** — Select signal GPIO2_IO03.

36.4.191 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04)

Address: 20E_0000h base + 30Ch offset = 20E_030Ch

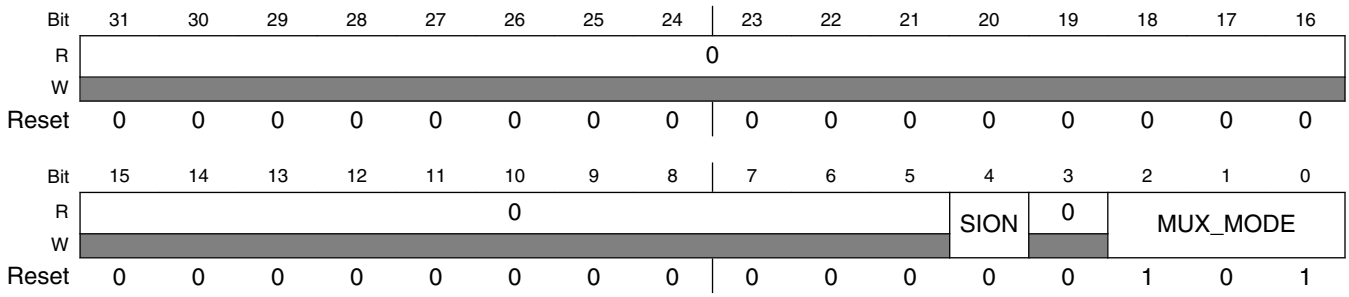


IOMUXC_SW_MUX_CTL_PAD_NAND_DATA04 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D4. 000 **ALT0** — Select signal NAND_DATA04. 001 **ALT1** — Select signal SD2_DATA4. 101 **ALT5** — Select signal GPIO2_IO04.

36.4.192 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05)

Address: 20E_0000h base + 310h offset = 20E_0310h



IOMUXC_SW_MUX_CTL_PAD_NAND_DATA05 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D5. 000 **ALT0** — Select signal NAND_DATA05. 001 **ALT1** — Select signal SD2_DATA5. 101 **ALT5** — Select signal GPIO2_IO05.

36.4.193 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06)

Address: 20E_0000h base + 314h offset = 20E_0314h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_DATA06 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D6. 000 **ALT0** — Select signal NAND_DATA06. 001 **ALT1** — Select signal SD2_DATA6. 101 **ALT5** — Select signal GPIO2_IO06.

36.4.194 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07)

Address: 20E_0000h base + 318h offset = 20E_0318h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_NAND_DATA07 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad NANDF_D7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: NANDF_D7. 000 **ALT0** — Select signal NAND_DATA07. 001 **ALT1** — Select signal SD2_DATA7. 101 **ALT5** — Select signal GPIO2_IO07.

36.4.195 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0)

Address: 20E_0000h base + 31Ch offset = 20E_031Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT0. 001 **ALT1** — Select signal SD4_DATA0. 010 **ALT2** — Select signal NAND_DQS. 101 **ALT5** — Select signal GPIO2_IO08.

36.4.196 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1)

Address: 20E_0000h base + 320h offset = 20E_0320h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT1. 001 **ALT1** — Select signal SD4_DATA1. 010 **ALT2** — Select signal PWM3_OUT. 101 **ALT5** — Select signal GPIO2_IO09.

36.4.197 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2)

Address: 20E_0000h base + 324h offset = 20E_0324h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT2. 001 **ALT1** — Select signal SD4_DATA2. 010 **ALT2** — Select signal PWM4_OUT. 101 **ALT5** — Select signal GPIO2_IO10.

36.4.198 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA3)

Address: 20E_0000h base + 328h offset = 20E_0328h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 2 iomux modes to be used for pad: SD4_DAT3. 001 **ALT1** — Select signal SD4_DATA3. 101 **ALT5** — Select signal GPIO2_IO11.

36.4.199 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4)

Address: 20E_0000h base + 32Ch offset = 20E_032Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA4 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT4. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT4. NOTE: Pad SD4_DAT4 is involved in Daisy Chain. 001 **ALT1** — Select signal SD4_DATA4. 010 **ALT2** — Select signal UART2_RX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO12.

36.4.200 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5)

Address: 20E_0000h base + 330h offset = 20E_0330h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT5. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA5 field descriptions (continued)

Field	Description
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT5. NOTE: Pad SD4_DAT5 is involved in Daisy Chain. 001 **ALT1** — Select signal SD4_DATA5. 010 **ALT2** — Select signal UART2_RTS_B. - Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO13.

36.4.201 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6)

Address: 20E_0000h base + 334h offset = 20E_0334h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT6. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT6. NOTE: Pad SD4_DAT6 is involved in Daisy Chain. 001 **ALT1** — Select signal SD4_DATA6. 010 **ALT2** — Select signal UART2_CTS_B.

Table continues on the next page...

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA6 field descriptions (continued)

Field	Description
	- Configure register IOMUXC_UART2_UART_RTS_B_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO14.

36.4.202 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7)

Address: 20E_0000h base + 338h offset = 20E_0338h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

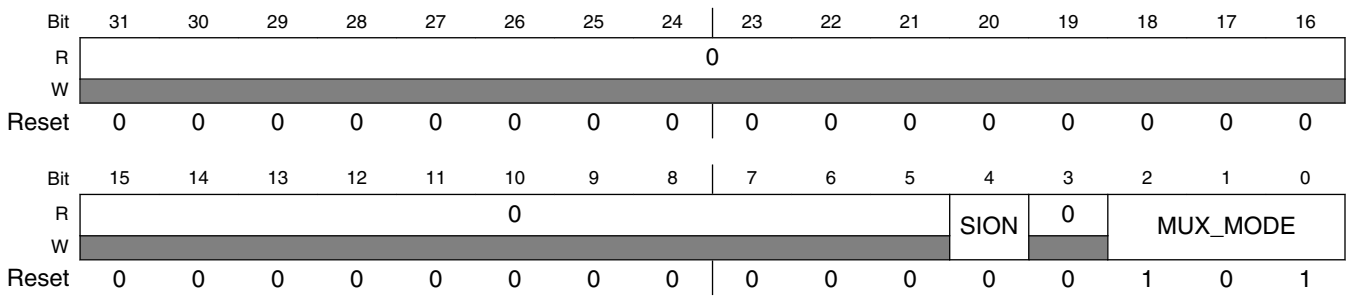
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD4_DATA7 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD4_DAT7. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 3 iomux modes to be used for pad: SD4_DAT7. NOTE: Pad SD4_DAT7 is involved in Daisy Chain. 001 **ALT1** — Select signal SD4_DATA7. 010 **ALT2** — Select signal UART2_TX_DATA. - Configure register IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT for mode ALT2. 101 **ALT5** — Select signal GPIO2_IO15.

36.4.203 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1)

Address: 20E_0000h base + 33Ch offset = 20E_033Ch

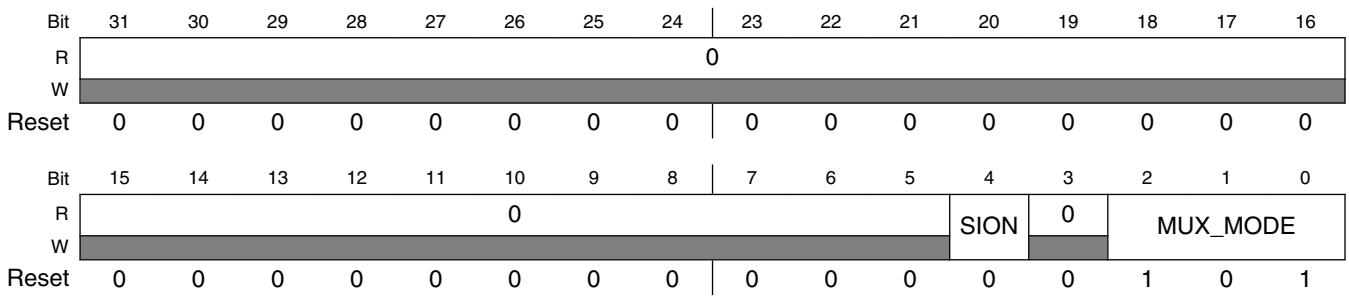


IOMUXC_SW_MUX_CTL_PAD_SD1_DATA1 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_DAT1. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD1_DAT1. NOTE: Pad SD1_DAT1 is involved in Daisy Chain. 000 **ALT0** — Select signal SD1_DATA1. 001 **ALT1** — Select signal ECSPI5_SS0. - Configure register IOMUXC_ECSPI5_SS0_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal PWM3_OUT. 011 **ALT3** — Select signal GPT_CAPTURE2. 101 **ALT5** — Select signal GPIO1_IO17.

36.4.204 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0)

Address: 20E_0000h base + 340h offset = 20E_0340h



IOMUXC_SW_MUX_CTL_PAD_SD1_DATA0 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_DAT0. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 4 iomux modes to be used for pad: SD1_DAT0. NOTE: Pad SD1_DAT0 is involved in Daisy Chain. 000 **ALT0** — Select signal SD1_DATA0. 001 **ALT1** — Select signal ECSPI5_MISO. - Configure register IOMUXC_ECSPI5_MISO_SELECT_INPUT for mode ALT1. 011 **ALT3** — Select signal GPT_CAPTURE1. 101 **ALT5** — Select signal GPIO1_IO16.

36.4.205 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3)

Address: 20E_0000h base + 344h offset = 20E_0344h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

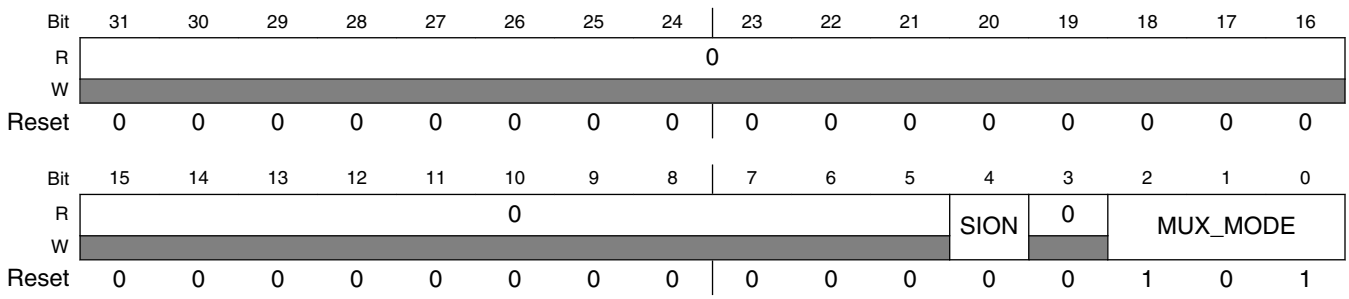
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD1_DATA3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_DAT3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: SD1_DAT3. 000 **ALT0** — Select signal SD1_DATA3. 001 **ALT1** — Select signal ECSPI5_SS2. 010 **ALT2** — Select signal GPT_COMPARE3. 011 **ALT3** — Select signal PWM1_OUT. 100 **ALT4** — Select signal WDOG2_B. 101 **ALT5** — Select signal GPIO1_IO21. 110 **ALT6** — Select signal WDOG2_RESET_B_DEB.

36.4.206 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_CMD)

Address: 20E_0000h base + 348h offset = 20E_0348h



IOMUXC_SW_MUX_CTL_PAD_SD1_CMD field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_CMD. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD1_CMD. NOTE: Pad SD1_CMD is involved in Daisy Chain. 000 **ALT0** — Select signal SD1_CMD. 001 **ALT1** — Select signal ECSPI5_MOSI. - Configure register IOMUXC_ECSPI5_MOSI_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal PWM4_OUT. 011 **ALT3** — Select signal GPT_COMPARE1. 101 **ALT5** — Select signal GPIO1_IO18.

36.4.207 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2)

Address: 20E_0000h base + 34Ch offset = 20E_034Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD1_DATA2 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_DAT2. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 7 iomux modes to be used for pad: SD1_DAT2. NOTE: Pad SD1_DAT2 is involved in Daisy Chain. 000 **ALT0** — Select signal SD1_DATA2. 001 **ALT1** — Select signal ECSPI5_SS1. - Configure register IOMUXC_ECSPI5_SS1_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal GPT_COMPARE2. 011 **ALT3** — Select signal PWM2_OUT. 100 **ALT4** — Select signal WDOG1_B. 101 **ALT5** — Select signal GPIO1_IO19. 110 **ALT6** — Select signal WDOG1_RESET_B_DEB.

36.4.208 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD1_CLK)

Address: 20E_0000h base + 350h offset = 20E_0350h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

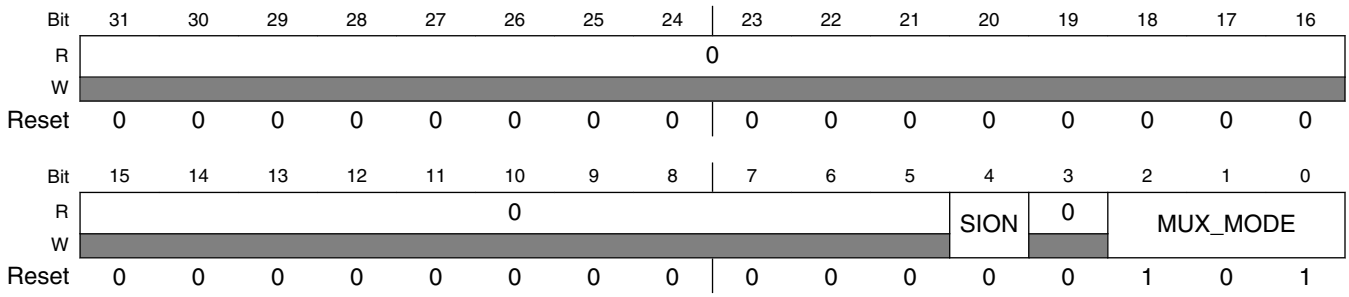
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD1_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD1_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD1_CLK. NOTE: Pad SD1_CLK is involved in Daisy Chain. 000 **ALT0** — Select signal SD1_CLK. 001 **ALT1** — Select signal ECSPI5_SCLK. - Configure register IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal XTALOSC_OSC32K_32K_OUT. 011 **ALT3** — Select signal GPT_CLKIN. 101 **ALT5** — Select signal GPIO1_IO20.

36.4.209 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_CLK)

Address: 20E_0000h base + 354h offset = 20E_0354h

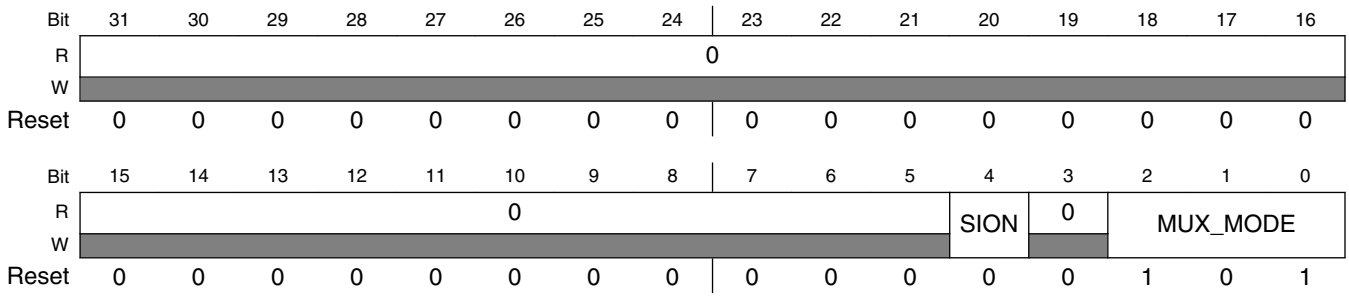


IOMUXC_SW_MUX_CTL_PAD_SD2_CLK field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_CLK. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD2_CLK. NOTE: Pad SD2_CLK is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_CLK. 001 **ALT1** — Select signal ECSPI5_SCLK. - Configure register IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal KEY_COL5. - Configure register IOMUXC_KEY_COL5_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_RXFS. - Configure register IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO1_IO10.

36.4.210 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_CMD)

Address: 20E_0000h base + 358h offset = 20E_0358h



IOMUXC_SW_MUX_CTL_PAD_SD2_CMD field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_CMD. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD2_CMD. NOTE: Pad SD2_CMD is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_CMD. 001 **ALT1** — Select signal ECSPI5_MOSI. - Configure register IOMUXC_ECSPI5_MOSI_SELECT_INPUT for mode ALT1. 010 **ALT2** — Select signal KEY_ROW5. - Configure register IOMUXC_KEY_ROW5_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_RXC. - Configure register IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO1_IO11.

36.4.211 Pad Mux Register (IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3)

Address: 20E_0000h base + 35Ch offset = 20E_035Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											SION	0	MUX_MODE		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IOMUXC_SW_MUX_CTL_PAD_SD2_DATA3 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4 SION	Software Input On Field. Force the selected mux mode input path no matter of MUX_MODE functionality. 1 **ENABLED** — Force input path of pad SD2_DAT3. 0 **DISABLED** — Input Path is determined by functionality of the selected mux mode (regular).
3 Reserved	This read-only field is reserved and always has the value 0.
MUX_MODE	MUX Mode Select Field. Select 1 of 5 iomux modes to be used for pad: SD2_DAT3. NOTE: Pad SD2_DAT3 is involved in Daisy Chain. 000 **ALT0** — Select signal SD2_DATA3. 001 **ALT1** — Select signal ECSPI5_SS3. 010 **ALT2** — Select signal KEY_COL6. - Configure register IOMUXC_KEY_COL6_SELECT_INPUT for mode ALT2. 011 **ALT3** — Select signal AUD4_TXC. - Configure register IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT for mode ALT3. 101 **ALT5** — Select signal GPIO1_IO12.

36.4.212 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA1)

Address: 20E_0000h base + 360h offset = 20E_0360h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_DAT1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_DAT1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_DAT1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_DAT1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA1 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_DAT1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.213 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA2)

Address: 20E_0000h base + 364h offset = 20E_0364h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_DAT2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_DAT2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_DAT2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_DAT2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_DAT2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA2 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.214 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA0)

Address: 20E_0000h base + 368h offset = 20E_0368h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_DAT0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_DAT0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA0 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_DAT0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_DAT0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_DAT0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA0 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.215 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TXC)

Address: 20E_0000h base + 36Ch offset = 20E_036Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_TXC field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TXC. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TXC. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up

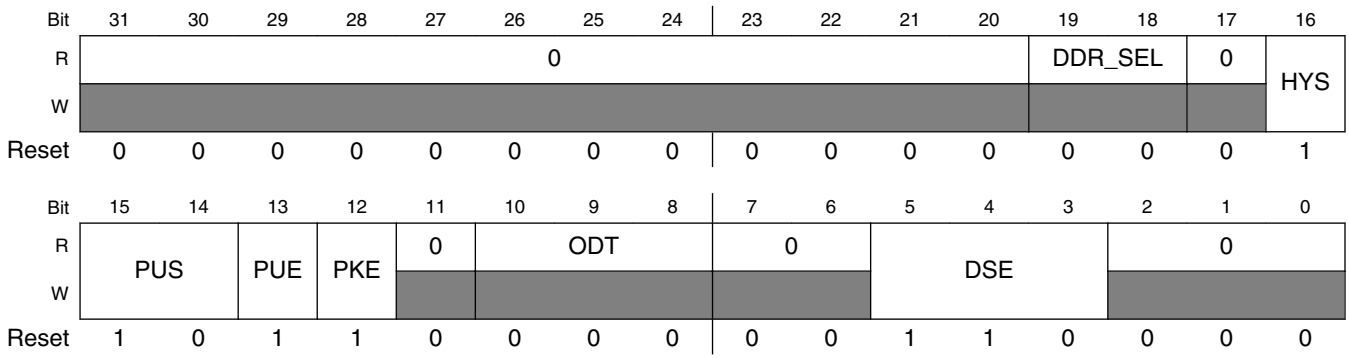
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TXC field descriptions (continued)

Field	Description
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TXC. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TXC. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TXC. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.216 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD0)

Address: 20E_0000h base + 370h offset = 20E_0370h



IOMUXC_SW_PAD_CTL_PAD_RGMII_TD0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TD0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TD0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TD0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TD0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD0 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TD0. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.217 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD1)

Address: 20E_0000h base + 374h offset = 20E_0374h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TD1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TD1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TD1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TD1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TD1. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V

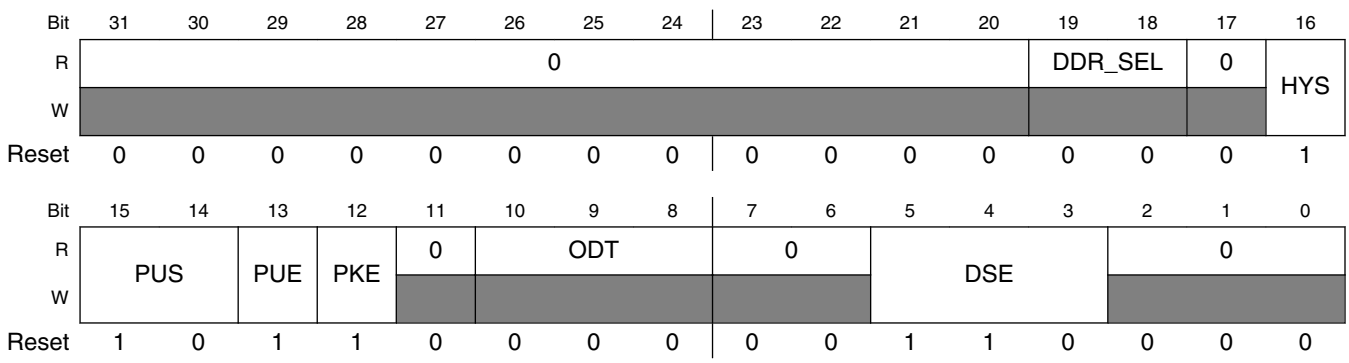
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD1 field descriptions (continued)

Field	Description
	010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.218 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD2)

Address: 20E_0000h base + 378h offset = 20E_0378h



IOMUXC_SW_PAD_CTL_PAD_RGMII_TD2 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TD2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TD2.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD2 field descriptions (continued)

Field	Description
	00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TD2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TD2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TD2. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.219 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TD3)

Address: 20E_0000h base + 37Ch offset = 20E_037Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD3 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TD3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TD3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TD3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TD3.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TD3 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TD3. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.220 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RX_CTL)

Address: 20E_0000h base + 380h offset = 20E_0380h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RX_CTL field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RX_CTL. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RX_CTL. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RX_CTL. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RX_CTL. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RX_CTL. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RX_CTL field descriptions (continued)

Field	Description
	011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.221 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD0)

Address: 20E_0000h base + 384h offset = 20E_0384h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RD0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RD0. 00 **100K_OHM_PD** — 100K Ohm Pull Down

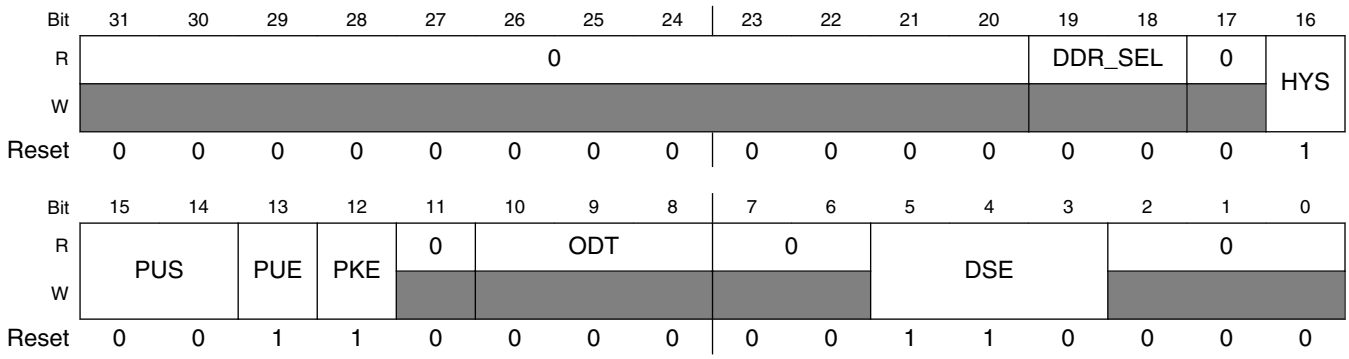
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD0 field descriptions (continued)

Field	Description
	01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RD0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RD0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RD0. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.222 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_TX_CTL)

Address: 20E_0000h base + 388h offset = 20E_0388h



IOMUXC_SW_PAD_CTL_PAD_RGMII_TX_CTL field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_TX_CTL. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_TX_CTL. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_TX_CTL. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_TX_CTL.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_TX_CTL field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Read Only Field The value of this field is fixed and cannot be changed. 000 **DISABLED** — Disabled
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_TX_CTL. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.223 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD1)

Address: 20E_0000h base + 38Ch offset = 20E_038Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RD1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RD1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RD1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RD1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RD1. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD1 field descriptions (continued)

Field	Description
	011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.224 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD2)

Address: 20E_0000h base + 390h offset = 20E_0390h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD2 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RD2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RD2. 00 **100K_OHM_PD** — 100K Ohm Pull Down

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD2 field descriptions (continued)

Field	Description
	01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RD2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RD2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RD2. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.225 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RD3)

Address: 20E_0000h base + 394h offset = 20E_0394h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	1	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD3 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RD3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RD3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RD3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RD3.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RD3 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RD3. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V 011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.226 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_RGMII_RXC)

Address: 20E_0000h base + 398h offset = 20E_0398h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	0	ODT			0		DSE			0		
W																
Reset	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_RGMII_RXC field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII Note: The value of this field does not reflect the value of the Group Control Register.
17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: RGMII_RXC. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: RGMII_RXC. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: RGMII_RXC. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: RGMII_RXC. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM Note: The value of this field does not reflect the value of the Group Control Register.
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: RGMII_RXC. 000 **HIZ** — HI-Z 001 **287_OHM** — 287/174 Ohm @ 2.5V, 247/201 Ohm pd/pu @ 1.8V 010 **121_OHM** — 121/85 Ohm @ 2.5V, 113/96 Ohm pd/pu @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_RGMII_RXC field descriptions (continued)

Field	Description
	011 **76_OHM** — 76/56 Ohm @ 2.5V, 73/63 Ohm pd/pu @ 1.8V 100 **57_OHM** — 57/43 Ohm @ 2.5V, 55/48 Ohm pd/pu @ 1.8V 101 **45_OHM** — 45/34 Ohm @ 2.5V, 43/38 Ohm pd/pu @ 1.8V 110 **37_OHM** — 37/28 Ohm @ 2.5V, 36/32 Ohm pd/pu @ 1.8V 111 **31_OHM** — 31/24 Ohm @ 2.5V, 30/27 Ohm pd/pu @ 1.8V
Reserved	This read-only field is reserved and always has the value 0.

36.4.227 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR25)

Address: 20E_0000h base + 39Ch offset = 20E_039Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR25 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A25. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A25. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A25. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR25 field descriptions (continued)

Field	Description
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A25. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A25. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.228 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB2_B)

Address: 20E_0000h base + 3A0h offset = 20E_03A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_EB2_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_EB2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_EB2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_EB2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_EB2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_EB2_B field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_EB2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.229 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA16)

Address: 20E_0000h base + 3A4h offset = 20E_03A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA16 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D16. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D16. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D16. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D16. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D16. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA16 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.230 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA17)

Address: 20E_0000h base + 3A8h offset = 20E_03A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA17 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D17. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D17. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA17 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D17. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D17. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D17. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

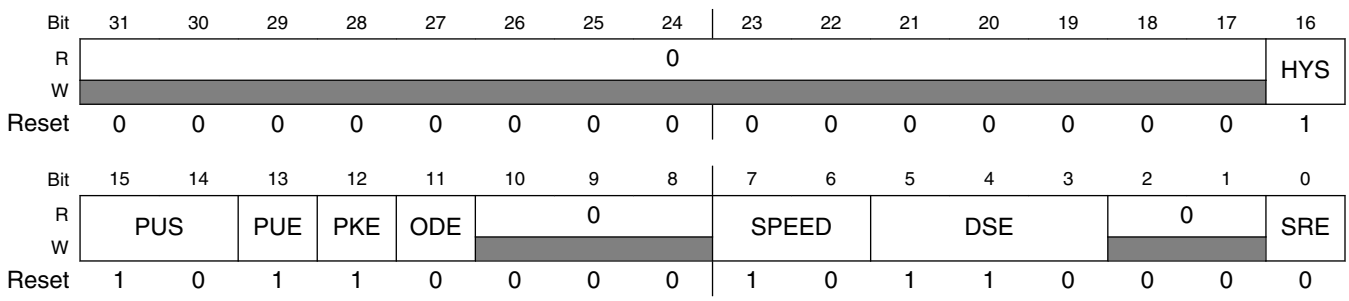
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA17 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.231 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA18)

Address: 20E_0000h base + 3ACh offset = 20E_03ACh



IOMUXC_SW_PAD_CTL_PAD_EIM_DATA18 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D18. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D18. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D18. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D18.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA18 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D18. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.232 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA19)

Address: 20E_0000h base + 3B0h offset = 20E_03B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA19 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D19. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D19. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D19. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D19. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA19 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D19. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.233 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA20)

Address: 20E_0000h base + 3B4h offset = 20E_03B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA20 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D20. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D20. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D20. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D20. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D20. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA20 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.234 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA21)

Address: 20E_0000h base + 3B8h offset = 20E_03B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA21 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D21. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D21. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA21 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D21. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D21. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D21. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA21 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.235 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA22)

Address: 20E_0000h base + 3BCh offset = 20E_03BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA22 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D22. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D22. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D22. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D22.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA22 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D22. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.236 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA23)

Address: 20E_0000h base + 3C0h offset = 20E_03C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA23 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D23. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D23. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D23. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D23. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA23 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D23. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.237 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB3_B)

Address: 20E_0000h base + 3C4h offset = 20E_03C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_EB3_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_EB3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_EB3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_EB3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_EB3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_EB3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_EB3_B field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.238 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA24)

Address: 20E_0000h base + 3C8h offset = 20E_03C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA24 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D24. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D24. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA24 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D24. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D24. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D24. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA24 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.239 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA25)

Address: 20E_0000h base + 3CCh offset = 20E_03CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA25 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D25. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D25. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D25. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D25.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA25 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D25. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.240 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA26)

Address: 20E_0000h base + 3D0h offset = 20E_03D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA26 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D26. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D26. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D26. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D26. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA26 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D26. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.241 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA27)

Address: 20E_0000h base + 3D4h offset = 20E_03D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA27 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D27. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D27. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D27. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D27. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D27. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA27 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.242 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA28)

Address: 20E_0000h base + 3D8h offset = 20E_03D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA28 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D28. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D28. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA28 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D28. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D28. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D28. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA28 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.243 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA29)

Address: 20E_0000h base + 3DCh offset = 20E_03DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA29 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D29. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D29. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D29. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D29.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA29 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D29. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.244 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA30)

Address: 20E_0000h base + 3E0h offset = 20E_03E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA30 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D30. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D30. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D30. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D30. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA30 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D30. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.245 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_DATA31)

Address: 20E_0000h base + 3E4h offset = 20E_03E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA31 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_D31. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_D31. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_D31. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_D31. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_D31. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_DATA31 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.246 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR24)

Address: 20E_0000h base + 3E8h offset = 20E_03E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR24 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A24. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A24. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR24 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A24. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A24. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A24. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

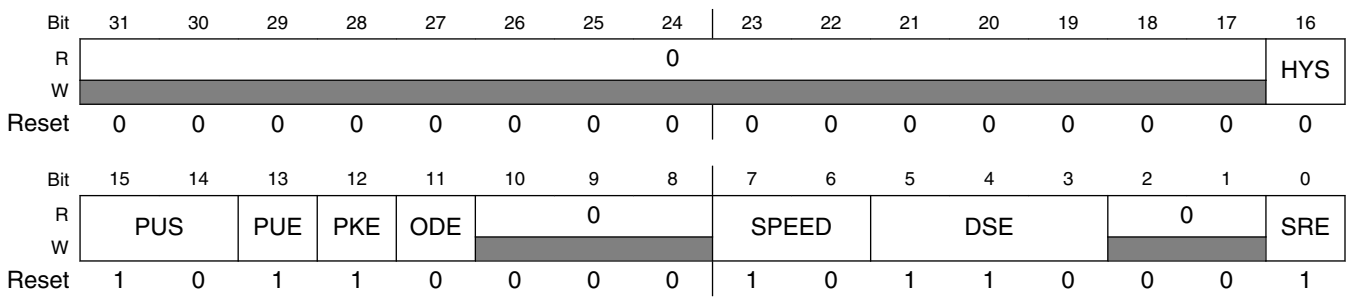
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR24 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.247 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR23)

Address: 20E_0000h base + 3ECh offset = 20E_03ECh



IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR23 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A23. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A23. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A23. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A23.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR23 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A23. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.248 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR22)

Address: 20E_0000h base + 3F0h offset = 20E_03F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR22 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A22. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A22. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A22. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A22. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR22 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A22. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.249 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR21)

Address: 20E_0000h base + 3F4h offset = 20E_03F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR21 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A21. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A21. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A21. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A21. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A21. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR21 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.250 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR20)

Address: 20E_0000h base + 3F8h offset = 20E_03F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR20 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A20. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A20. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR20 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A20. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A20. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A20. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR20 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.251 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR19)

Address: 20E_0000h base + 3FCh offset = 20E_03FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR19 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A19. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A19. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A19. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A19.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR19 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A19. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.252 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR18)

Address: 20E_0000h base + 400h offset = 20E_0400h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR18 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A18. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A18. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A18. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A18. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR18 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A18. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.253 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR17)

Address: 20E_0000h base + 404h offset = 20E_0404h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR17 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A17. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A17. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A17. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A17. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A17. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR17 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.254 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR16)

Address: 20E_0000h base + 408h offset = 20E_0408h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR16 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_A16. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_A16. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR16 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_A16. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_A16. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_A16. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

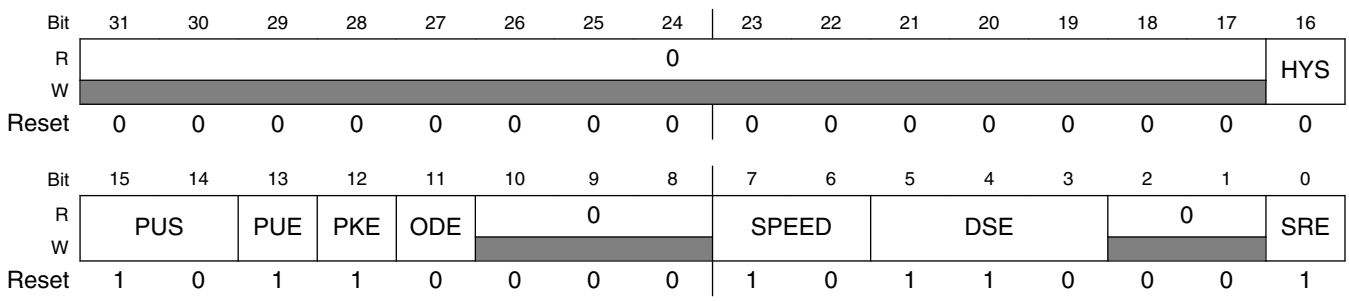
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_ADDR16 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.255 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_CS0_B)

Address: 20E_0000h base + 40Ch offset = 20E_040Ch



IOMUXC_SW_PAD_CTL_PAD_EIM_CS0_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_CS0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_CS0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_CS0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_CS0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_CS0_B field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_CS0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.256 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_CS1_B)

Address: 20E_0000h base + 410h offset = 20E_0410h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_CS1_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_CS1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_CS1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_CS1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_CS1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_CS1_B field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_CS1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.257 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_OE_B)

Address: 20E_0000h base + 414h offset = 20E_0414h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_OE_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_OE. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_OE. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_OE. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_OE. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_OE. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_OE_B field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.258 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_RW)

Address: 20E_0000h base + 418h offset = 20E_0418h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_RW field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_RW. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_RW. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_RW field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_RW. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_RW. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_RW. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

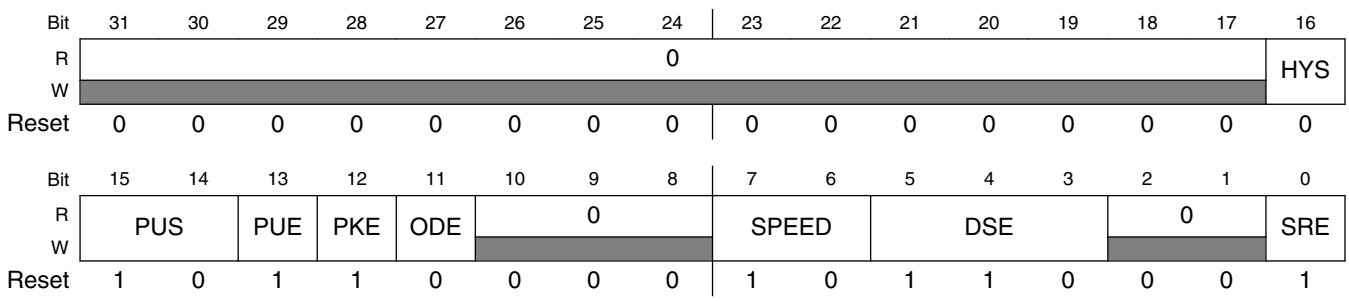
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_RW field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.259 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_LBA_B)

Address: 20E_0000h base + 41Ch offset = 20E_041Ch



IOMUXC_SW_PAD_CTL_PAD_EIM_LBA_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_LBA. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_LBA. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_LBA. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_LBA.

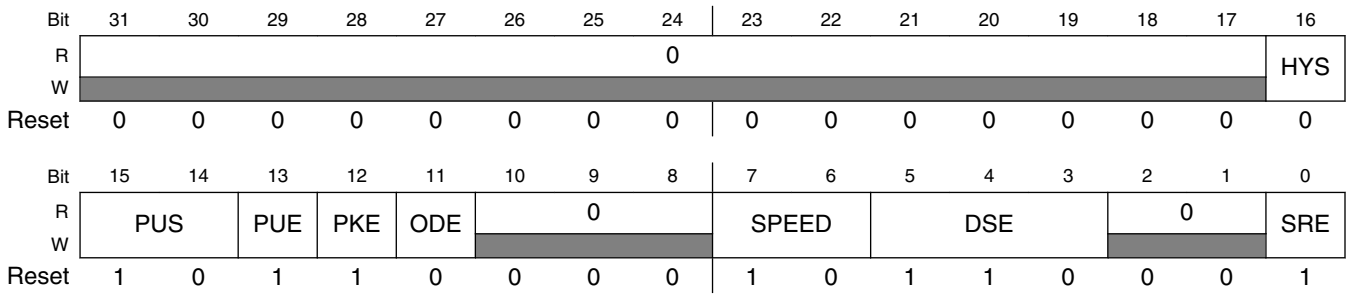
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_LBA_B field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_LBA. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.260 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB0_B)

Address: 20E_0000h base + 420h offset = 20E_0420h



IOMUXC_SW_PAD_CTL_PAD_EIM_EB0_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_EB0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_EB0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_EB0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_EB0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_EB0_B field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_EB0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.261 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_EB1_B)

Address: 20E_0000h base + 424h offset = 20E_0424h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_EB1_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_EB1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_EB1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_EB1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_EB1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_EB1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_EB1_B field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.262 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD00)

Address: 20E_0000h base + 428h offset = 20E_0428h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD00 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD00 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD00 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.263 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD01)

Address: 20E_0000h base + 42Ch offset = 20E_042Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD01 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA1.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD01 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.264 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD02)

Address: 20E_0000h base + 430h offset = 20E_0430h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD02 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD02 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.265 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD03)

Address: 20E_0000h base + 434h offset = 20E_0434h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD03 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD03 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.266 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD04)

Address: 20E_0000h base + 438h offset = 20E_0438h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD04 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

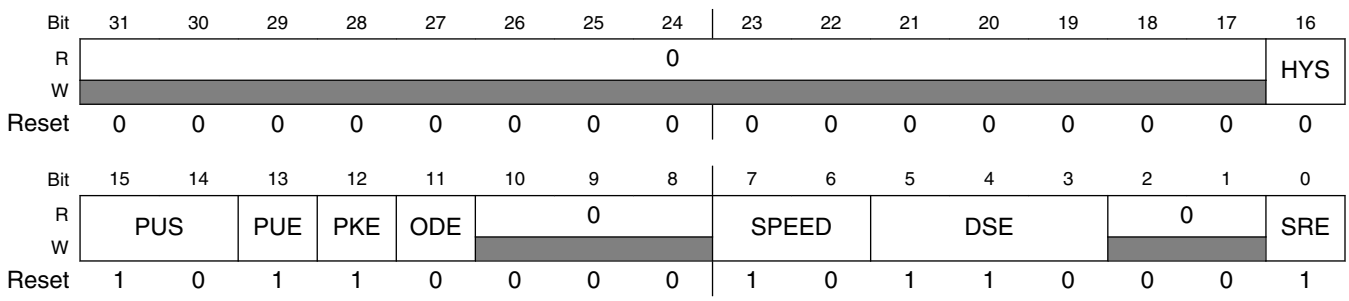
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD04 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.267 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD05)

Address: 20E_0000h base + 43Ch offset = 20E_043Ch



IOMUXC_SW_PAD_CTL_PAD_EIM_AD05 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA5.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD05 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.268 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD06)

Address: 20E_0000h base + 440h offset = 20E_0440h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD06 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD06 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.269 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD07)

Address: 20E_0000h base + 444h offset = 20E_0444h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD07 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD07 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.270 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD08)

Address: 20E_0000h base + 448h offset = 20E_0448h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD08 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA8. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA8. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD08 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA8. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA8. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA8. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD08 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.271 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD09)

Address: 20E_0000h base + 44Ch offset = 20E_044Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD09 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA9. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA9. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA9. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA9.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD09 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA9. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.272 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD10)

Address: 20E_0000h base + 450h offset = 20E_0450h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD10 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA10. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA10. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA10. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA10. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD10 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA10. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.273 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD11)

Address: 20E_0000h base + 454h offset = 20E_0454h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD11 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA11. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA11. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA11. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA11. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA11. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD11 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.274 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD12)

Address: 20E_0000h base + 458h offset = 20E_0458h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD12 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA12. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA12. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD12 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA12. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA12. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA12. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD12 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.275 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD13)

Address: 20E_0000h base + 45Ch offset = 20E_045Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD13 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA13. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA13. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA13. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA13.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD13 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA13. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.276 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD14)

Address: 20E_0000h base + 460h offset = 20E_0460h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD14 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA14. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA14. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA14. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA14. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD14 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA14. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.277 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_AD15)

Address: 20E_0000h base + 464h offset = 20E_0464h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_EIM_AD15 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_DA15. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_DA15. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_DA15. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_DA15. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_DA15. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_AD15 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.278 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_WAIT_B)

Address: 20E_0000h base + 468h offset = 20E_0468h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_EIM_WAIT_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_WAIT. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_WAIT. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_WAIT_B field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_WAIT. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_WAIT. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_WAIT. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

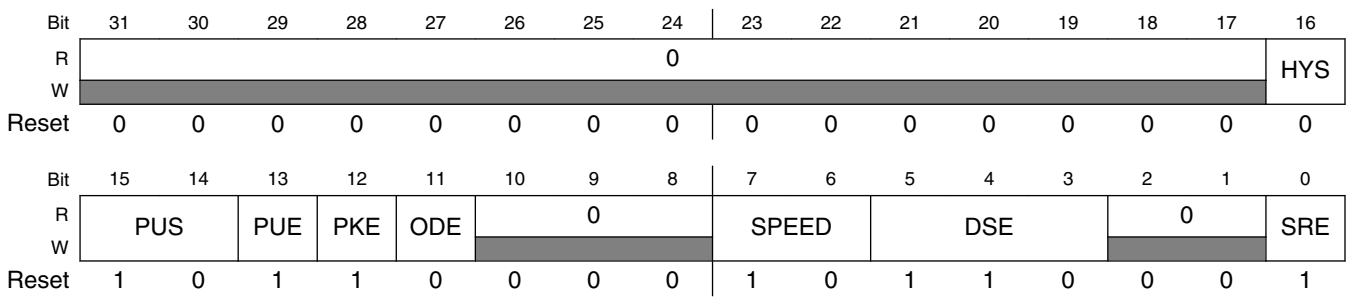
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_WAIT_B field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.279 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_EIM_BCLK)

Address: 20E_0000h base + 46Ch offset = 20E_046Ch



IOMUXC_SW_PAD_CTL_PAD_EIM_BCLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: EIM_BCLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: EIM_BCLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: EIM_BCLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: EIM_BCLK.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_EIM_BCLK field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: EIM_BCLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.280 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_DISP_CLK)

Address: 20E_0000h base + 470h offset = 20E_0470h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DI0_DISP_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DI0_DISP_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DI0_DISP_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DI0_DISP_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DI0_DISP_CLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_DISP_CLK field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DI0_DISP_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.281 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN15)

Address: 20E_0000h base + 474h offset = 20E_0474h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN15 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DI0_PIN15. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DI0_PIN15. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DI0_PIN15. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DI0_PIN15. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DI0_PIN15. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN15 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.282 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN02)

Address: 20E_0000h base + 478h offset = 20E_0478h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN02 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DI0_PIN2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DI0_PIN2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN02 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DI0_PIN2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DI0_PIN2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DI0_PIN2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN02 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.283 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN03)

Address: 20E_0000h base + 47Ch offset = 20E_047Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN03 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DI0_PIN3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DI0_PIN3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DI0_PIN3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DI0_PIN3.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN03 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DI0_PIN3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.284 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DI0_PIN04)

Address: 20E_0000h base + 480h offset = 20E_0480h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DI0_PIN4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DI0_PIN4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DI0_PIN4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DI0_PIN4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DI0_PIN04 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DI0_PIN4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.285 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA00)

Address: 20E_0000h base + 484h offset = 20E_0484h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA00 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA00 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.286 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA01)

Address: 20E_0000h base + 488h offset = 20E_0488h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA01 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA01 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

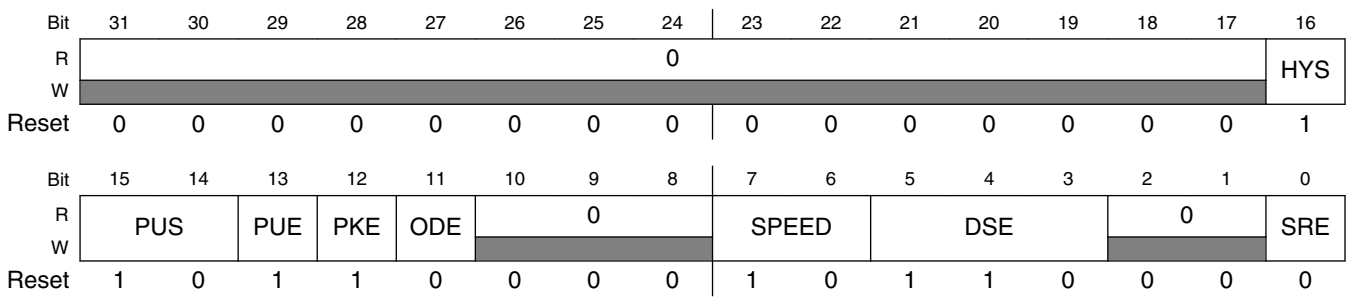
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA01 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.287 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA02)

Address: 20E_0000h base + 48Ch offset = 20E_048Ch



IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA02 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT2.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA02 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.288 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA03)

Address: 20E_0000h base + 490h offset = 20E_0490h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA03 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA03 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.289 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA04)

Address: 20E_0000h base + 494h offset = 20E_0494h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA04 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.290 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA05)

Address: 20E_0000h base + 498h offset = 20E_0498h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA05 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA05 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA05 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.291 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA06)

Address: 20E_0000h base + 49Ch offset = 20E_049Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA06 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT6.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA06 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.292 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA07)

Address: 20E_0000h base + 4A0h offset = 20E_04A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA07 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA07 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.293 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA08)

Address: 20E_0000h base + 4A4h offset = 20E_04A4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA08 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT8. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT8. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT8. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT8. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT8. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA08 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.294 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA09)

Address: 20E_0000h base + 4A8h offset = 20E_04A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA09 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT9. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT9. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA09 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT9. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT9. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT9. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

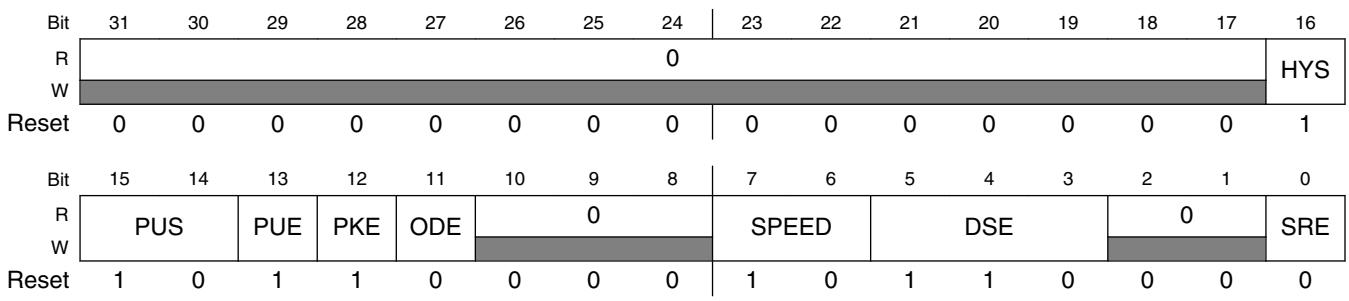
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA09 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.295 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA10)

Address: 20E_0000h base + 4ACh offset = 20E_04ACh



IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA10 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT10. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT10. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT10. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT10.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA10 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT10. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.296 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA11)

Address: 20E_0000h base + 4B0h offset = 20E_04B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA11 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT11. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT11. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT11. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT11. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA11 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT11. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.297 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA12)

Address: 20E_0000h base + 4B4h offset = 20E_04B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA12 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT12. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT12. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT12. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT12. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT12. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA12 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.298 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA13)

Address: 20E_0000h base + 4B8h offset = 20E_04B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA13 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT13. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT13. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA13 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT13. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT13. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT13. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA13 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.299 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA14)

Address: 20E_0000h base + 4BCh offset = 20E_04BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA14 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT14. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT14. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT14. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT14.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA14 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT14. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.300 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA15)

Address: 20E_0000h base + 4C0h offset = 20E_04C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA15 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT15. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT15. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT15. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT15. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA15 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT15. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.301 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA16)

Address: 20E_0000h base + 4C4h offset = 20E_04C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA16 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT16. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT16. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT16. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT16. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT16. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA16 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.302 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA17)

Address: 20E_0000h base + 4C8h offset = 20E_04C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA17 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT17. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT17. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA17 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT17. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT17. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT17. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

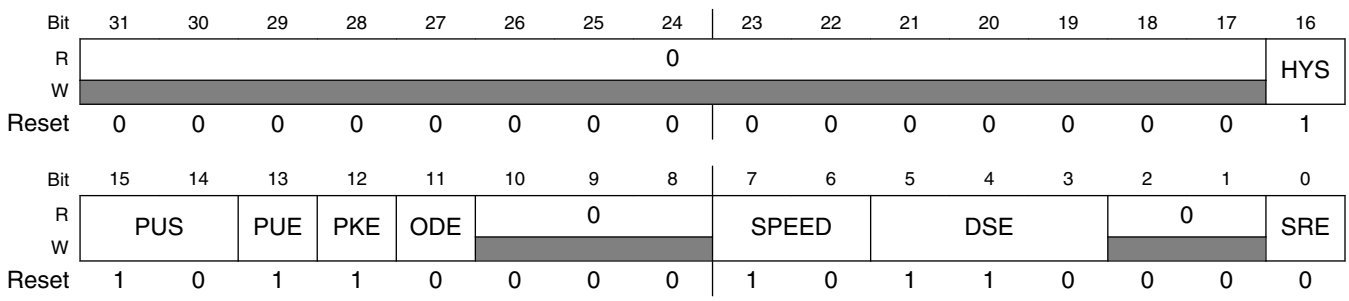
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA17 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.303 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA18)

Address: 20E_0000h base + 4CCh offset = 20E_04CCh



IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA18 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT18. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT18. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT18. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT18.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA18 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT18. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.304 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA19)

Address: 20E_0000h base + 4D0h offset = 20E_04D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA19 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT19. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT19. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT19. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT19. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA19 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT19. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.305 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA20)

Address: 20E_0000h base + 4D4h offset = 20E_04D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA20 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT20. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT20. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT20. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT20. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT20. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA20 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.306 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA21)

Address: 20E_0000h base + 4D8h offset = 20E_04D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA21 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT21. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT21. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA21 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT21. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT21. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT21. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA21 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.307 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA22)

Address: 20E_0000h base + 4DCh offset = 20E_04DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA22 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT22. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT22. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT22. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT22.

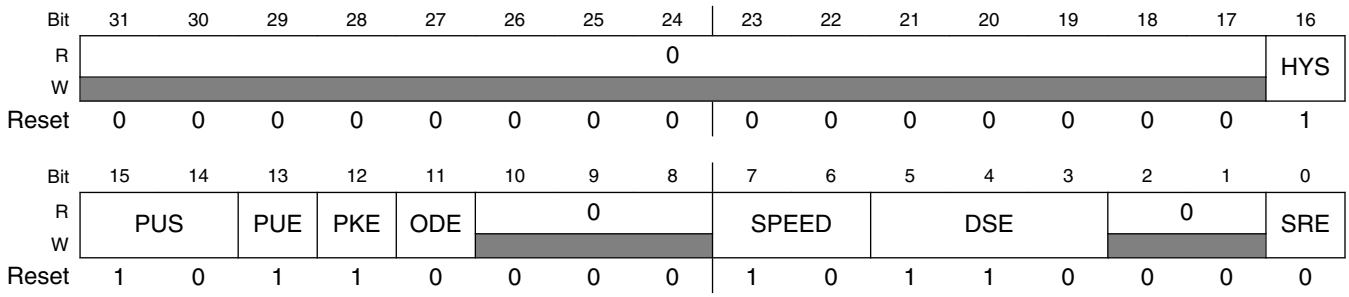
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA22 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT22. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.308 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA23)

Address: 20E_0000h base + 4E0h offset = 20E_04E0h



IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA23 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DISP0_DAT23. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DISP0_DAT23. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DISP0_DAT23. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DISP0_DAT23. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DISP0_DATA23 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: DISP0_DAT23. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.309 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_MDIO)

Address: 20E_0000h base + 4E4h offset = 20E_04E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_MDIO field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_MDIO. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_MDIO. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_MDIO. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_MDIO. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_MDIO. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_MDIO field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.310 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_REF_CLK)

Address: 20E_0000h base + 4E8h offset = 20E_04E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_REF_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_REF_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_REF_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_REF_CLK field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_REF_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_REF_CLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_REF_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_REF_CLK field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.311 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_ER)

Address: 20E_0000h base + 4ECh offset = 20E_04ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_ER field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_RX_ER. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_RX_ER. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_RX_ER. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_RX_ER.

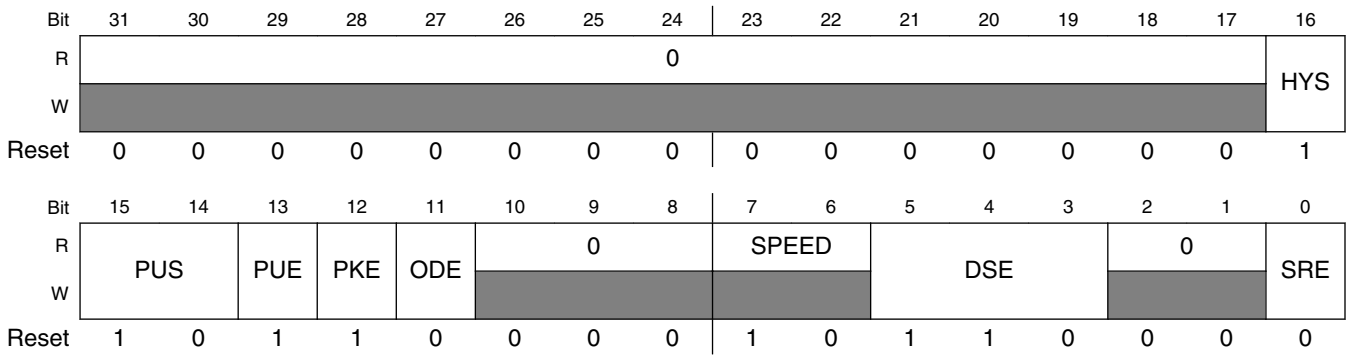
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_ER field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_RX_ER. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.312 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_CRS_DV)

Address: 20E_0000h base + 4F0h offset = 20E_04F0h



IOMUXC_SW_PAD_CTL_PAD_ENET_CRS_DV field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_CRS_DV. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_CRS_DV. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_CRS_DV. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_CRS_DV. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_CRS_DV field descriptions (continued)

Field	Description
	0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. Read Only Field The value of this field is fixed and cannot be changed. 10 **MEDIUM** — Medium frequency (100, 150 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_CRS_DV. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.313 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA1)

Address: 20E_0000h base + 4F4h offset = 20E_04F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_RXD1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_RXD1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_RXD1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_RXD1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_RXD1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA1 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.314 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA0)

Address: 20E_0000h base + 4F8h offset = 20E_04F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_RXD0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_RXD0. 00 **100K_OHM_PD** — 100K Ohm Pull Down

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA0 field descriptions (continued)

Field	Description
	01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_RXD0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_RXD0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. Read Only Field The value of this field is fixed and cannot be changed. 10 **MEDIUM** — Medium frequency (100, 150 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_RXD0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_RX_DATA0 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.315 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_EN)

Address: 20E_0000h base + 4FCh offset = 20E_04FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_TX_EN field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_TX_EN. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_TX_EN. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_TX_EN. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_TX_EN.

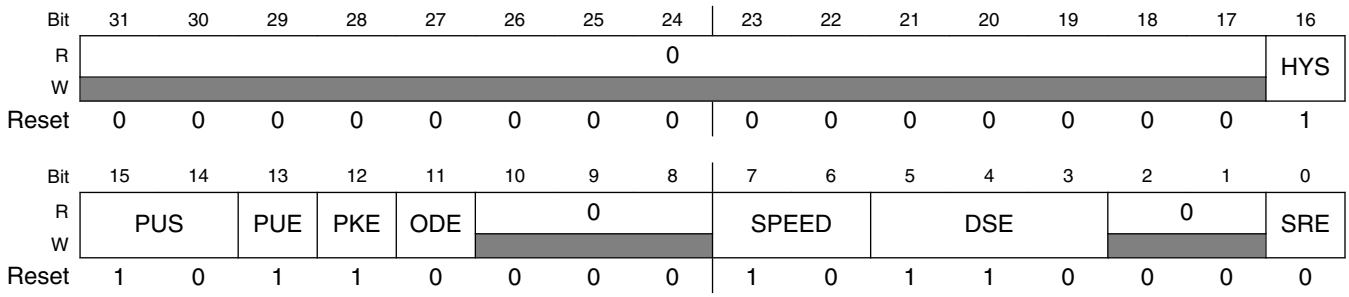
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_TX_EN field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_TX_EN. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.316 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA1)

Address: 20E_0000h base + 500h offset = 20E_0500h



IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_TXD1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_TXD1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_TXD1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_TXD1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA1 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_TXD1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.317 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA0)

Address: 20E_0000h base + 504h offset = 20E_0504h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_TXD0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_TXD0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_TXD0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_TXD0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_TXD0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_TX_DATA0 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.318 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_ENET_MDC)

Address: 20E_0000h base + 508h offset = 20E_0508h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_ENET_MDC field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: ENET_MDC. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: ENET_MDC. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_MDC field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: ENET_MDC. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: ENET_MDC. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: ENET_MDC. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

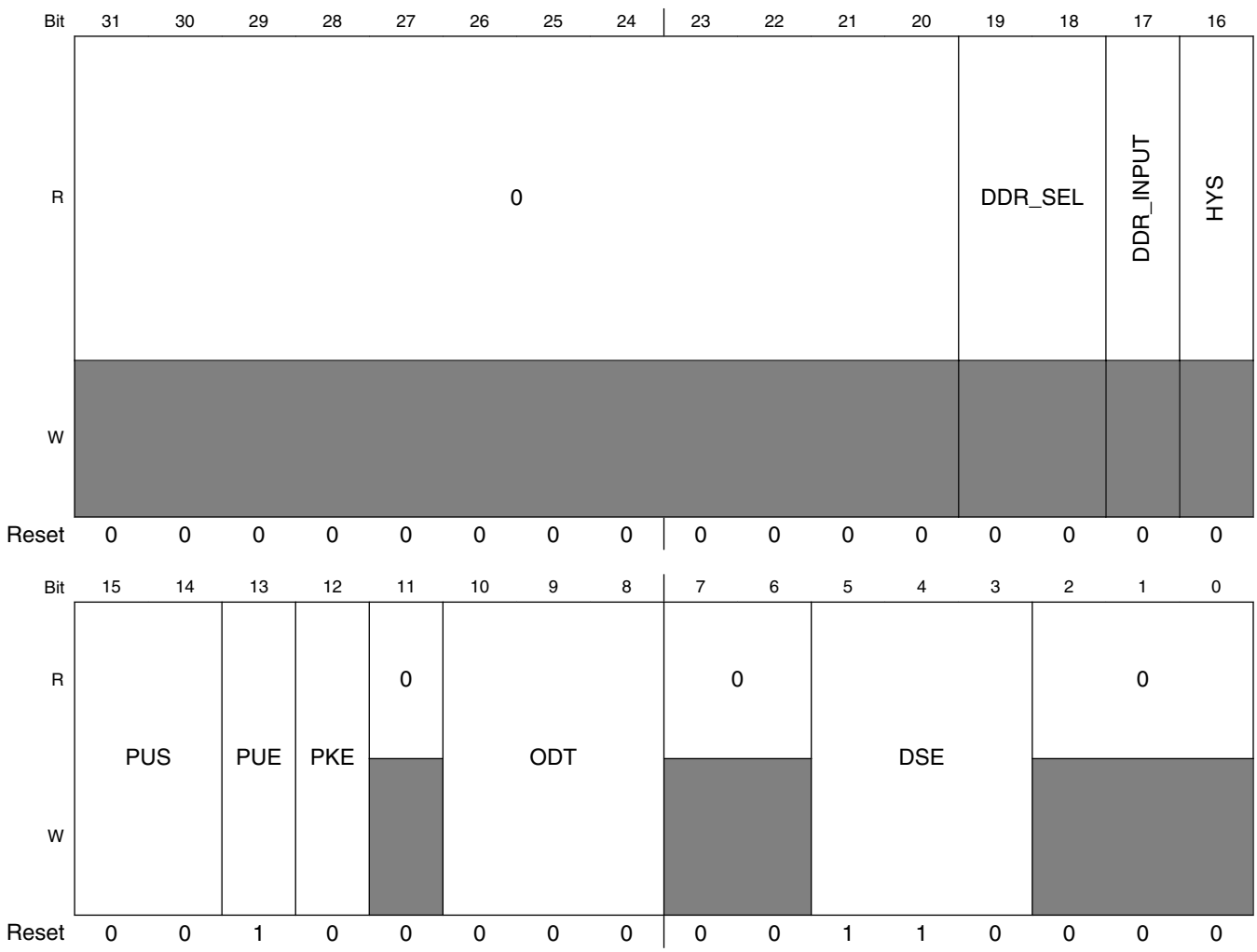
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_ENET_MDC field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.319 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS5_P)

Address: 20E_0000h base + 50Ch offset = 20E_050Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS5_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS5_P field descriptions (continued)

Field	Description
	This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS5. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field

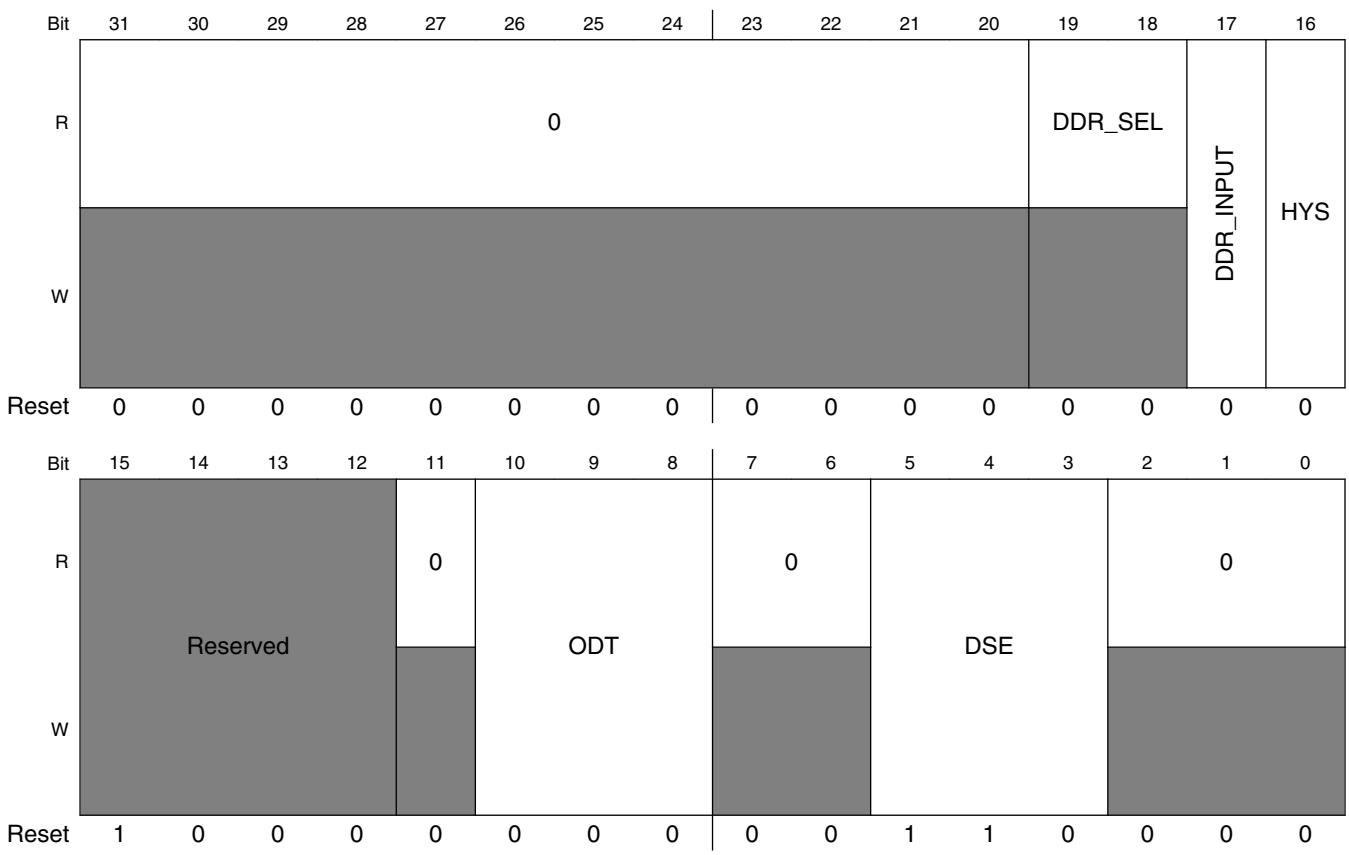
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS5_P field descriptions (continued)

Field	Description
	Select one of next values for pad: DRAM_SDQS5. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.320 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM5)

Address: 20E_0000h base + 510h offset = 20E_0510h

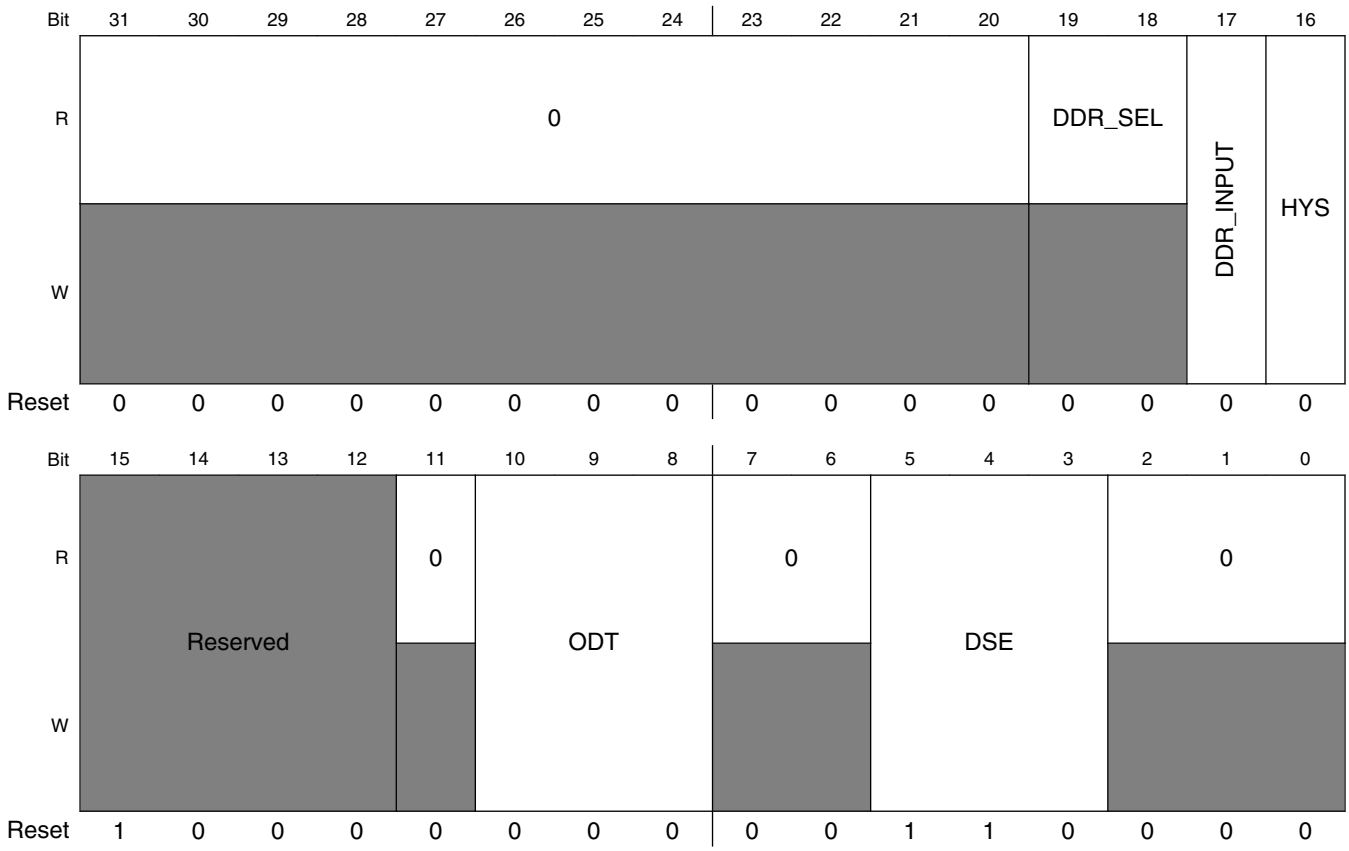


IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM5 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM5. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM5. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM5. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.321 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM4)

Address: 20E_0000h base + 514h offset = 20E_0514h



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM4 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM4. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM4.

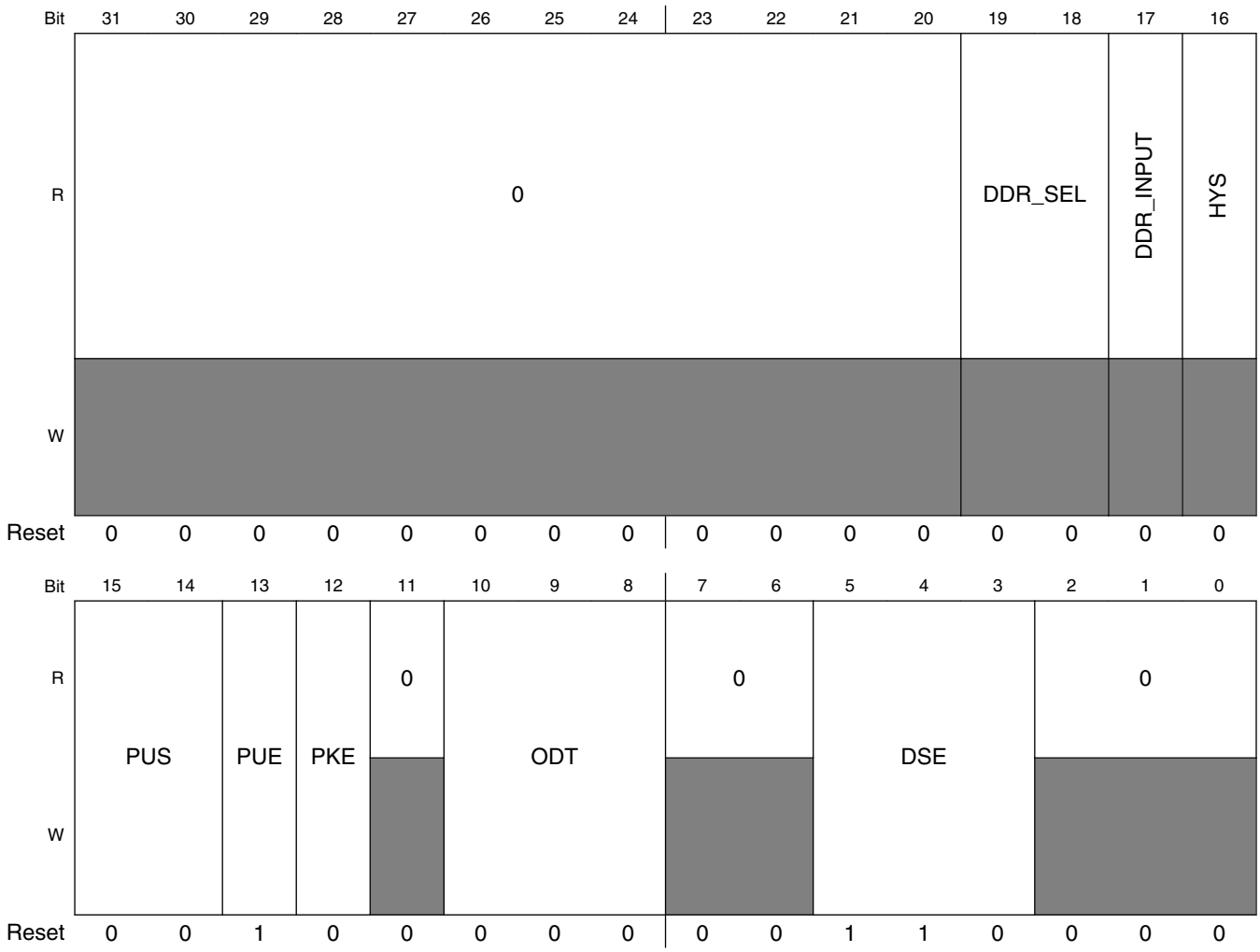
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM4 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM4. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM4. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.322 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS4_P)

Address: 20E_0000h base + 518h offset = 20E_0518h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS4_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS4_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS4. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS4. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

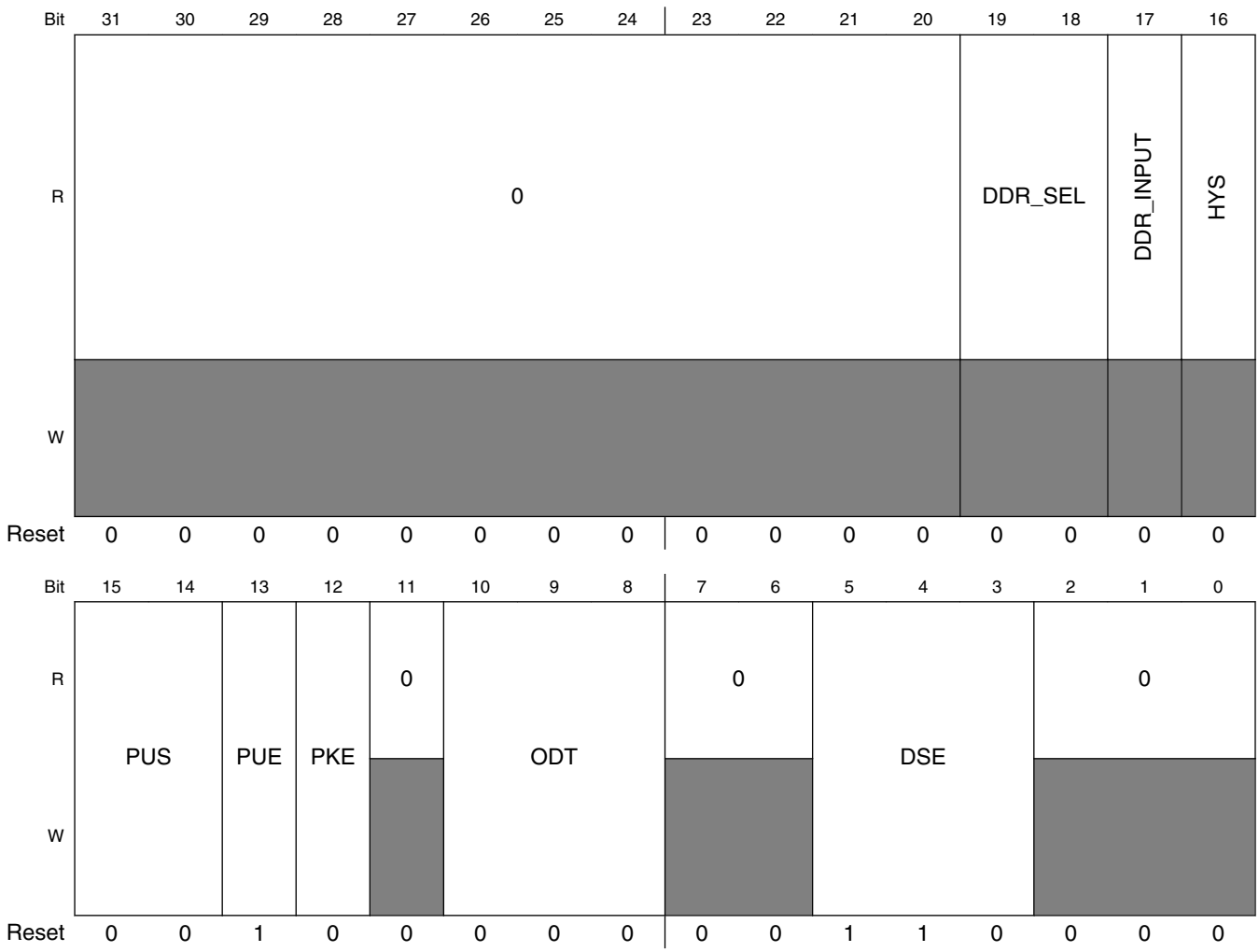
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS4_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.323 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS3_P)

Address: 20E_0000h base + 51Ch offset = 20E_051Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS3_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS3. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT

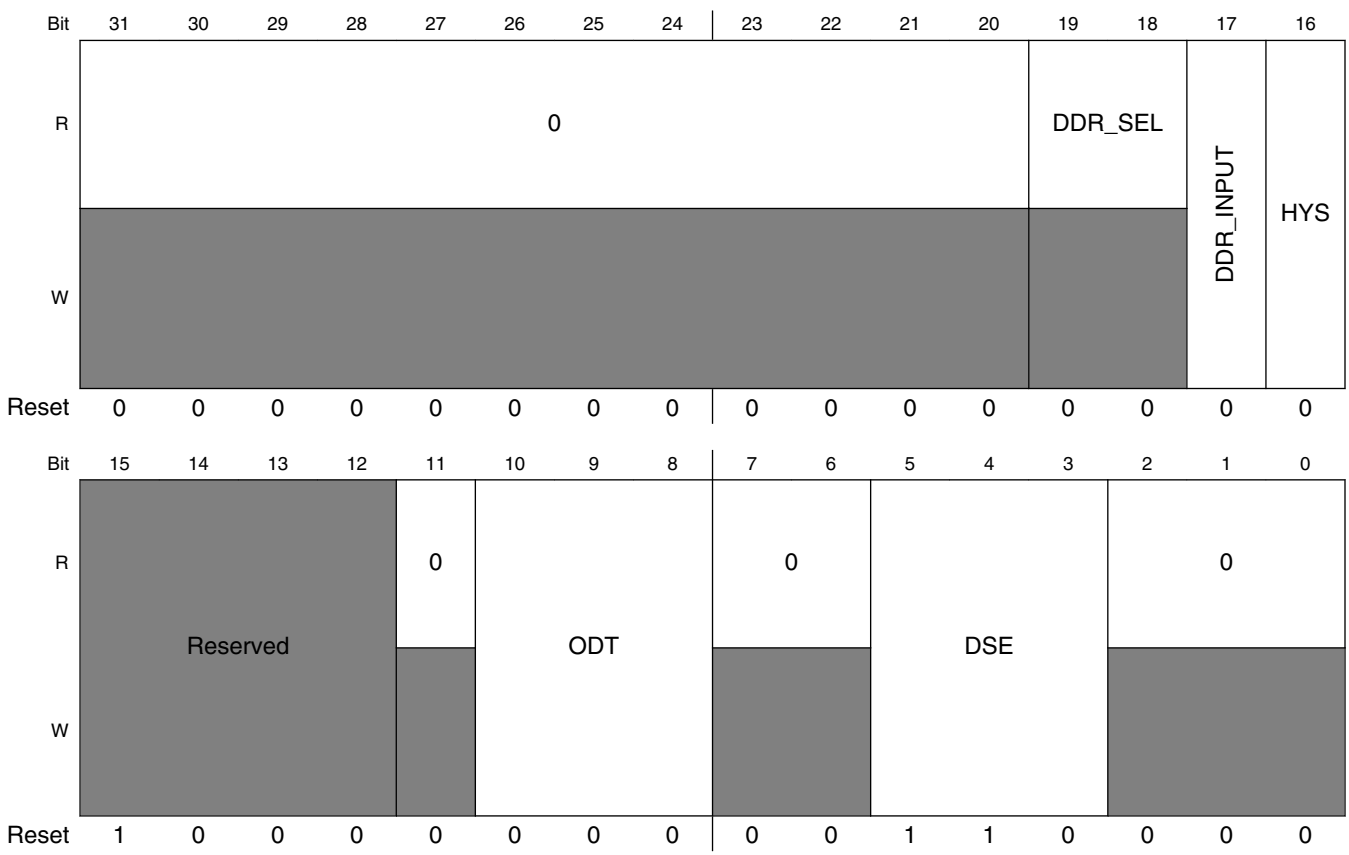
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS3_P field descriptions (continued)

Field	Description
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS3. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.324 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM3)

Address: 20E_0000h base + 520h offset = 20E_0520h

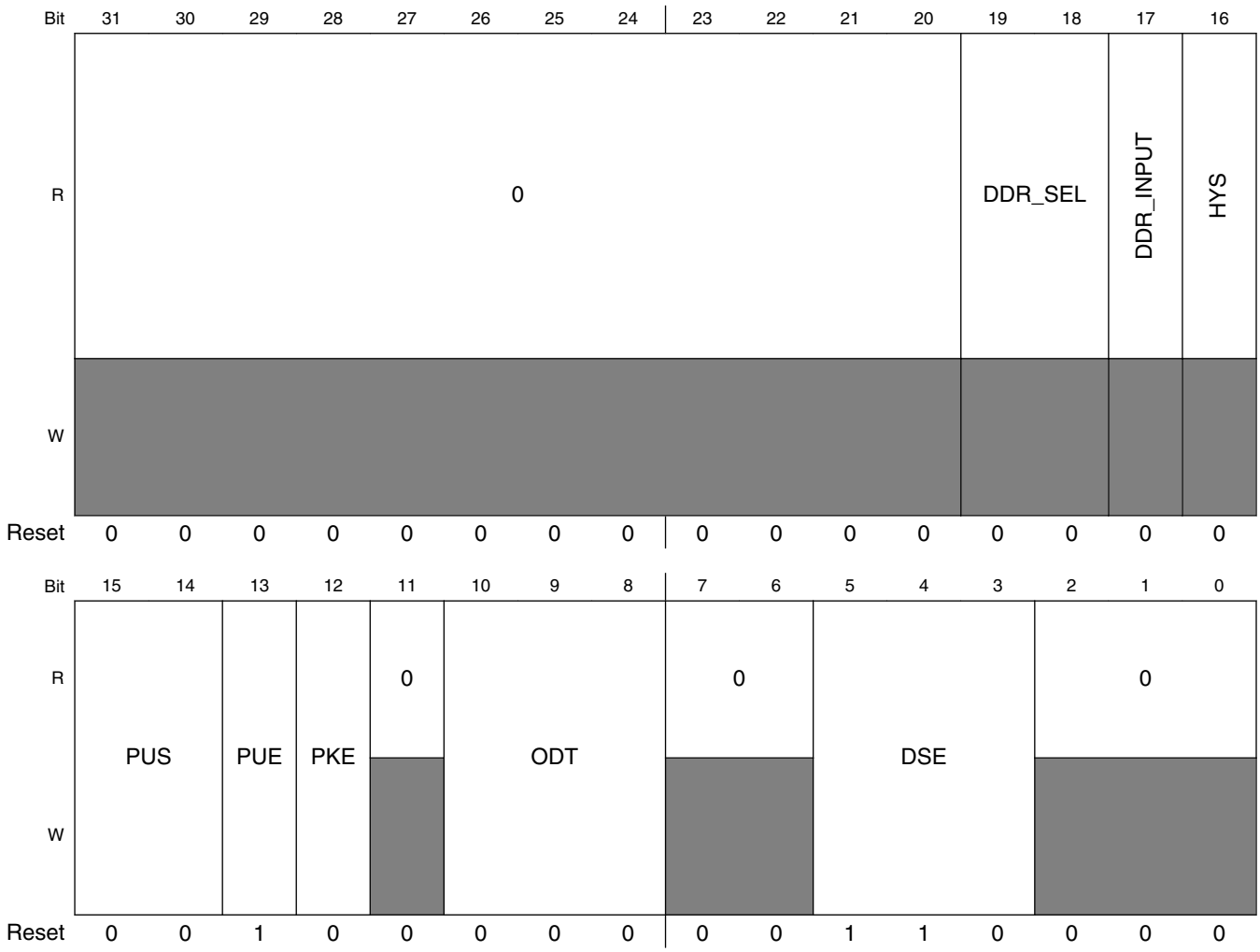


IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM3 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM3. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM3. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM3. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.325 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2_P)

Address: 20E_0000h base + 524h offset = 20E_0524h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS2. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS2. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

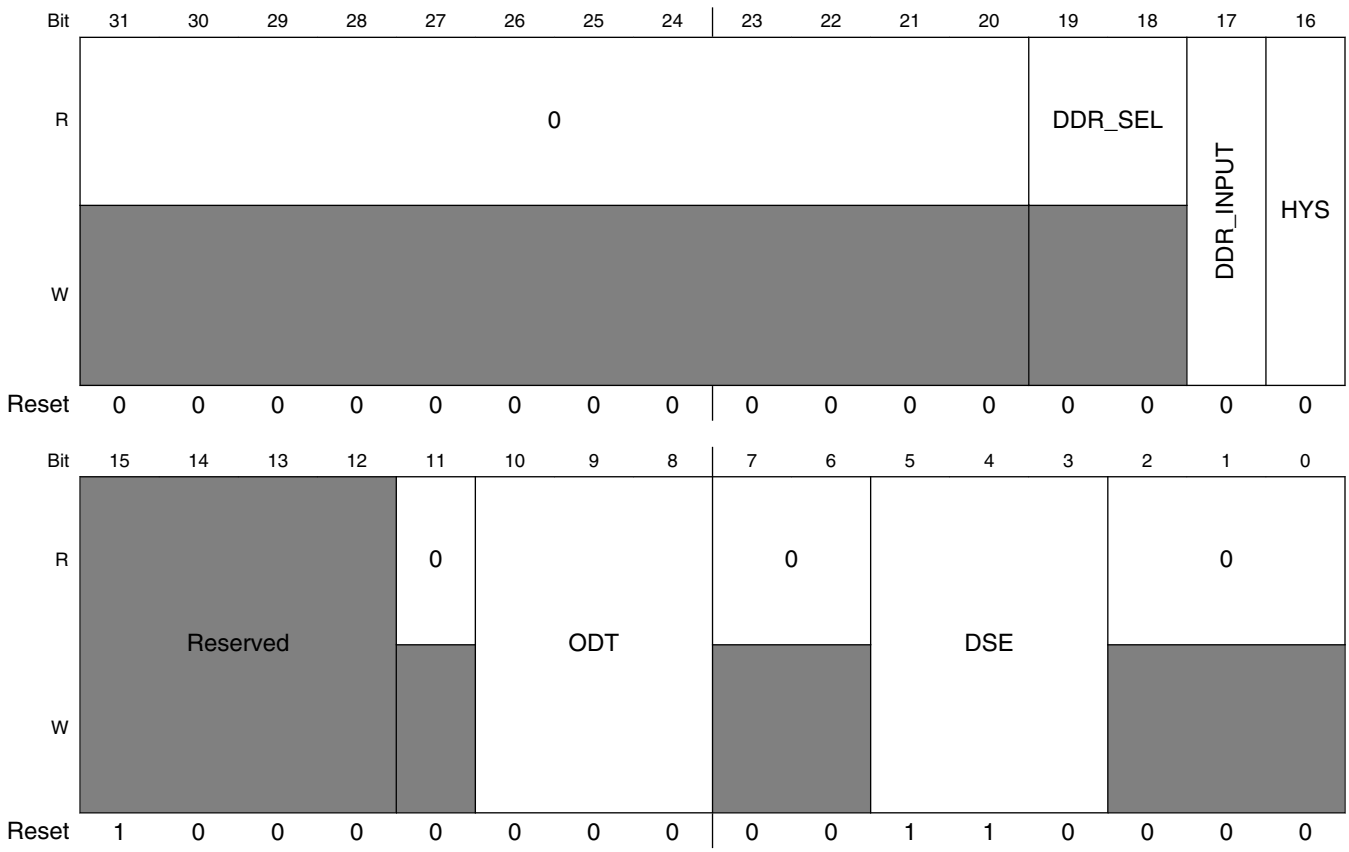
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS2_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.326 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM2)

Address: 20E_0000h base + 528h offset = 20E_0528h



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM2 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE

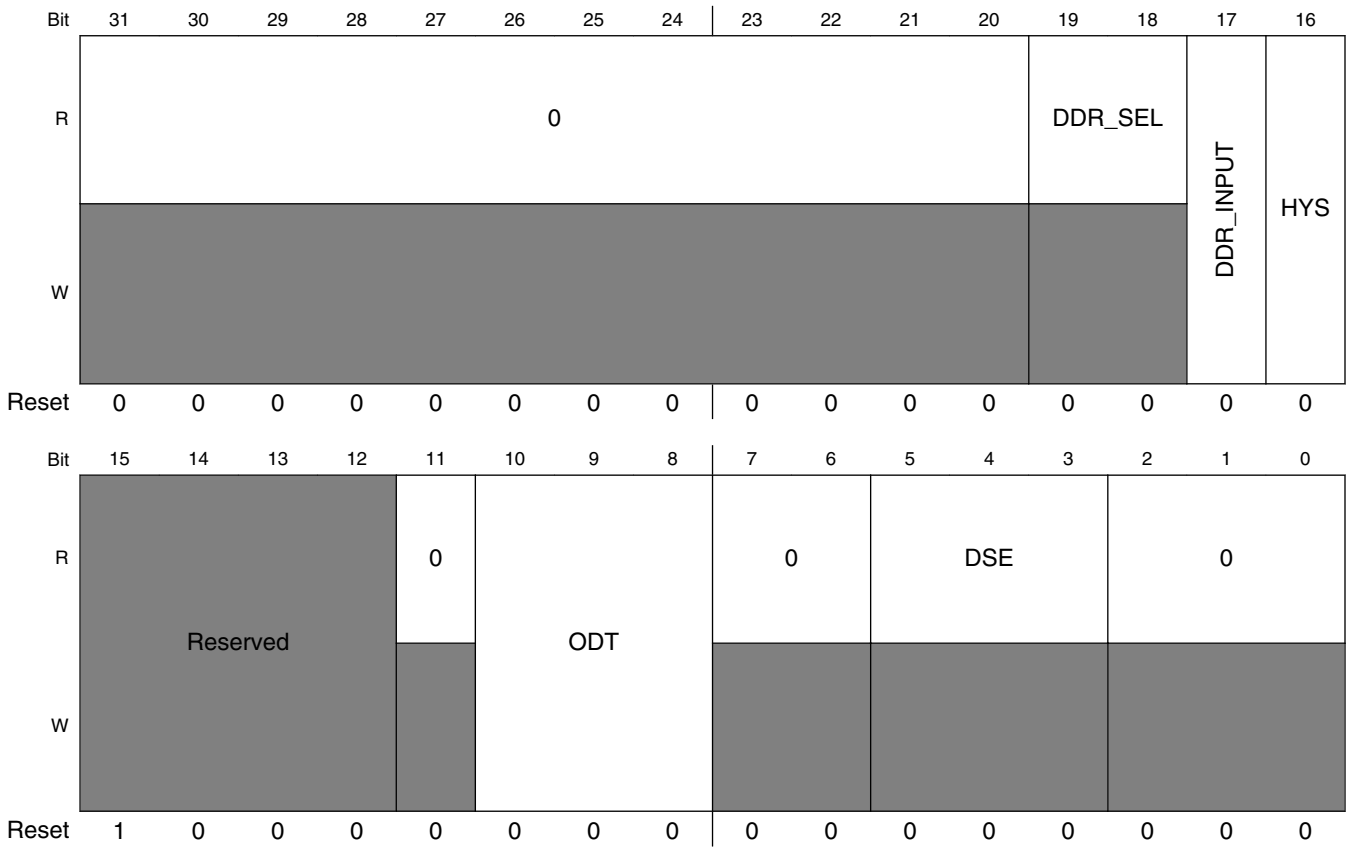
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM2 field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM2. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM2. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM2. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.327 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR00)

Address: 20E_0000h base + 52Ch offset = 20E_052Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR00 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A0.

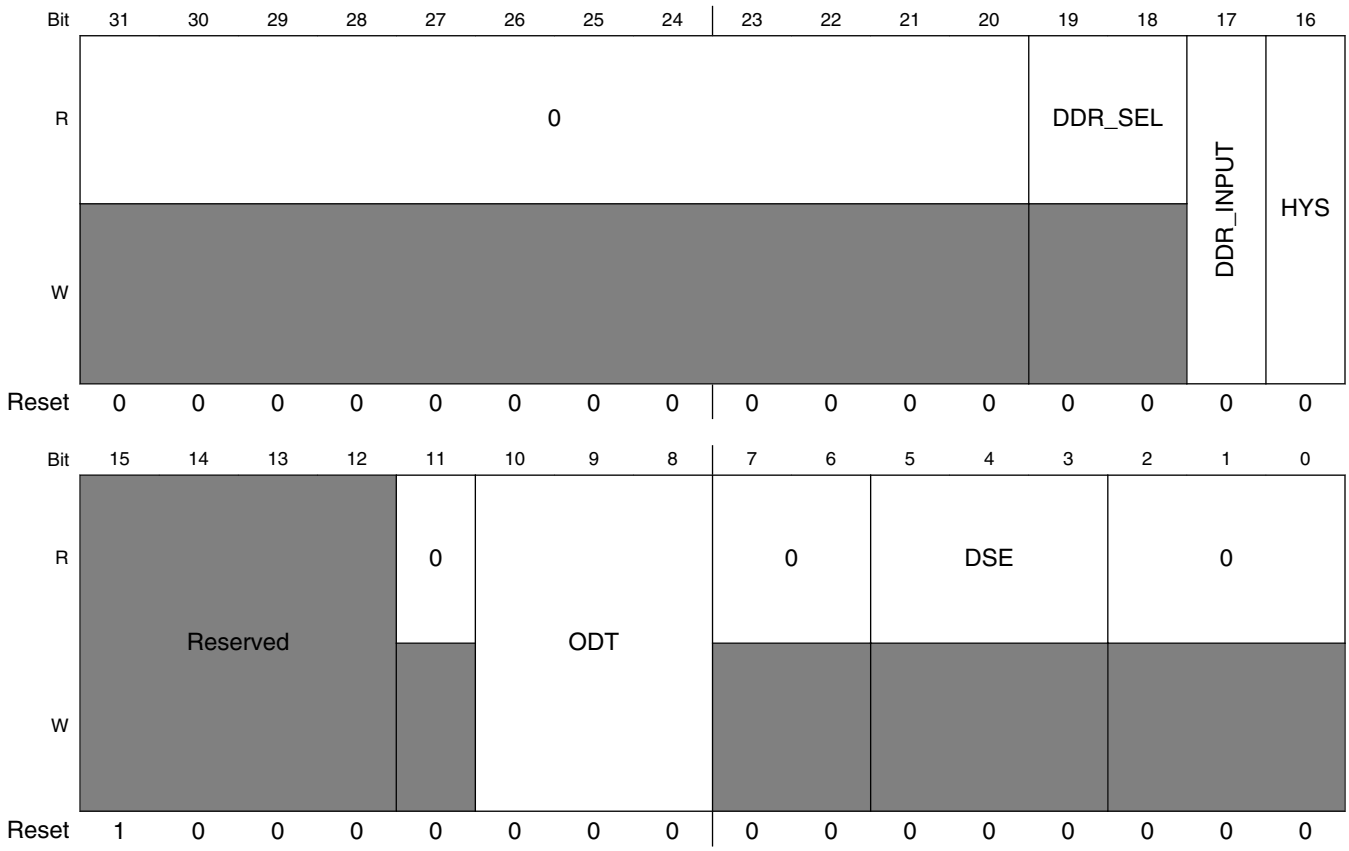
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR00 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.328 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR01)

Address: 20E_0000h base + 530h offset = 20E_0530h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR01 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A1.

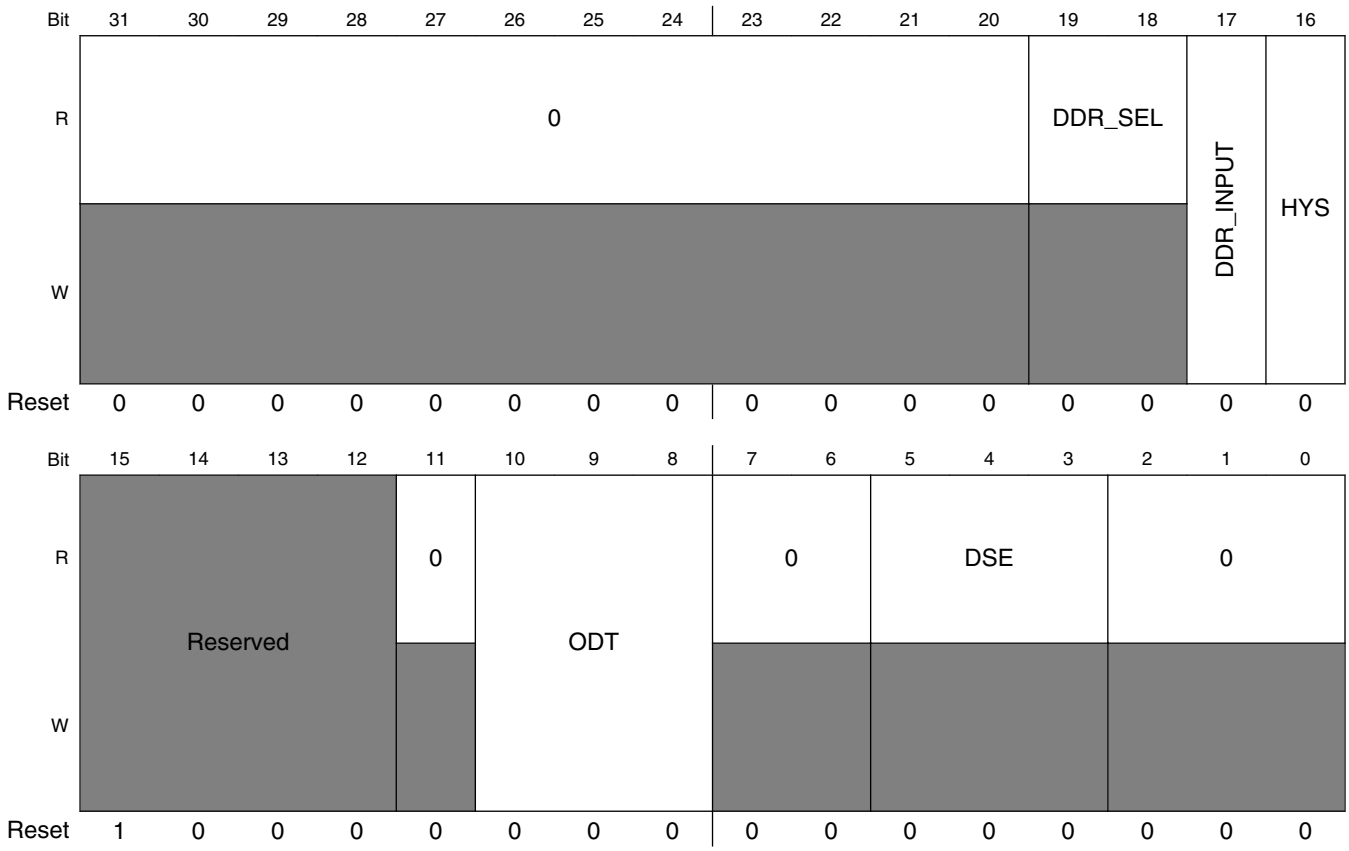
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR01 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.329 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR02)

Address: 20E_0000h base + 534h offset = 20E_0534h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR02 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A2. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A2.

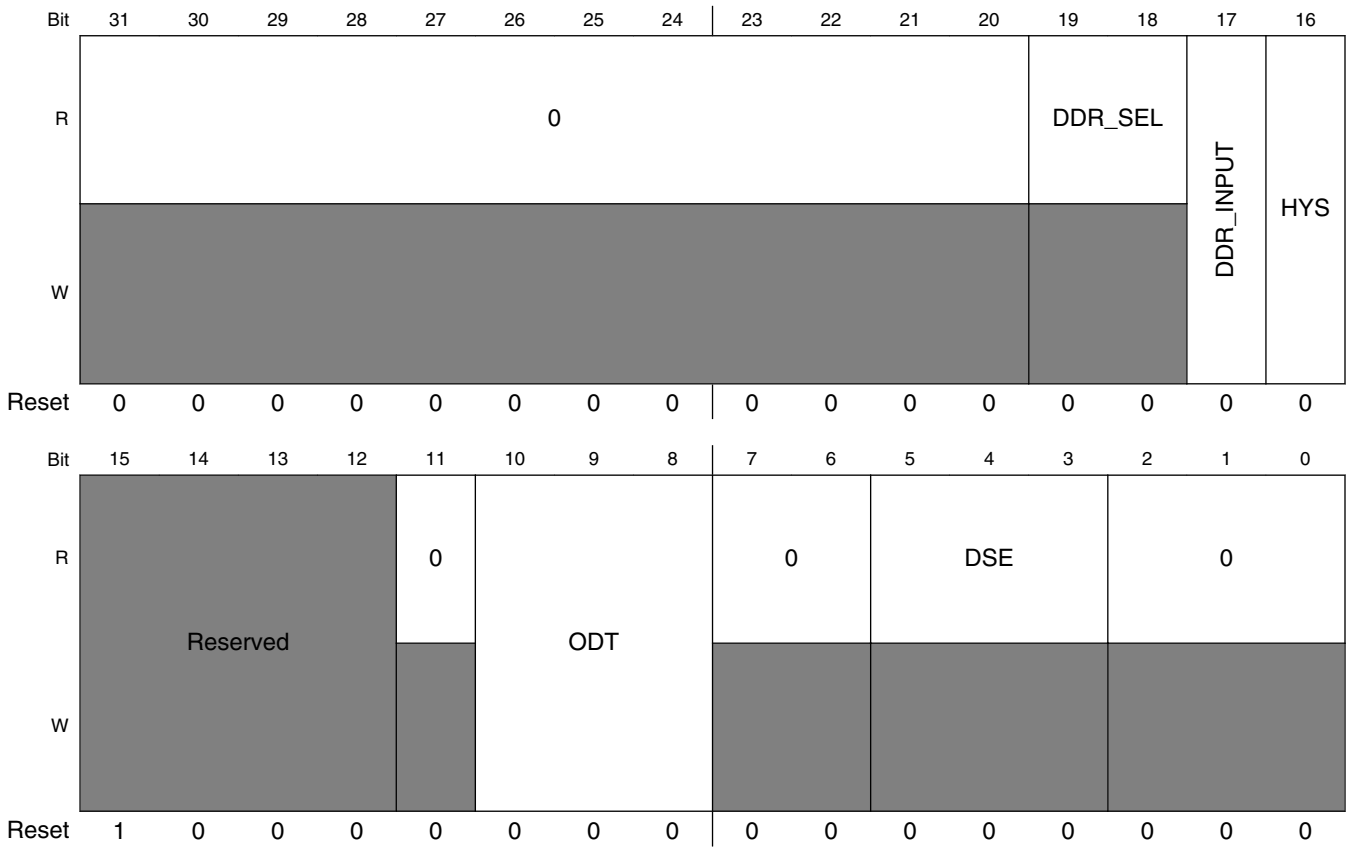
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR02 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A2. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.330 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR03)

Address: 20E_0000h base + 538h offset = 20E_0538h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR03 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A3. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A3.

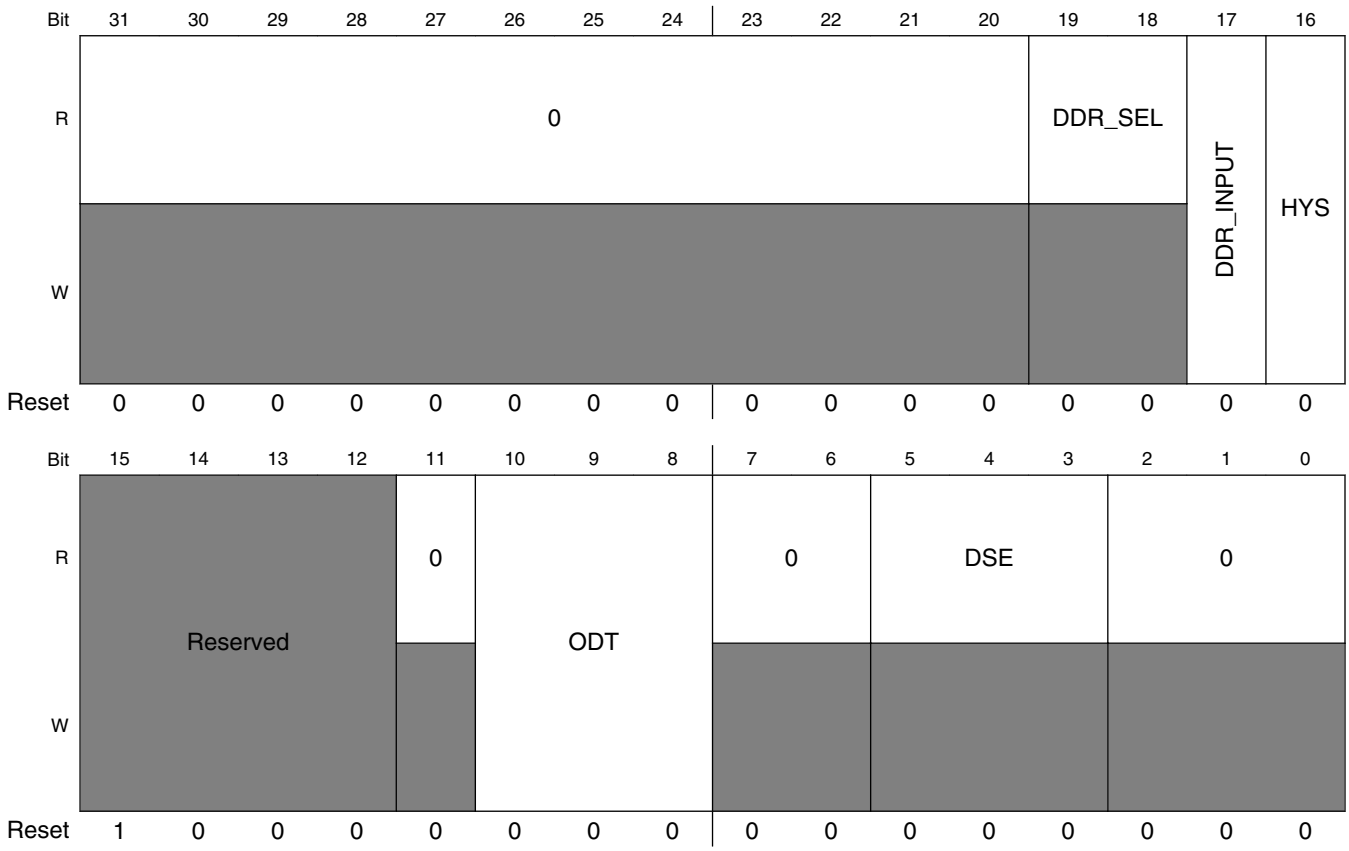
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR03 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A3. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.331 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR04)

Address: 20E_0000h base + 53Ch offset = 20E_053Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR04 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A4. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A4.

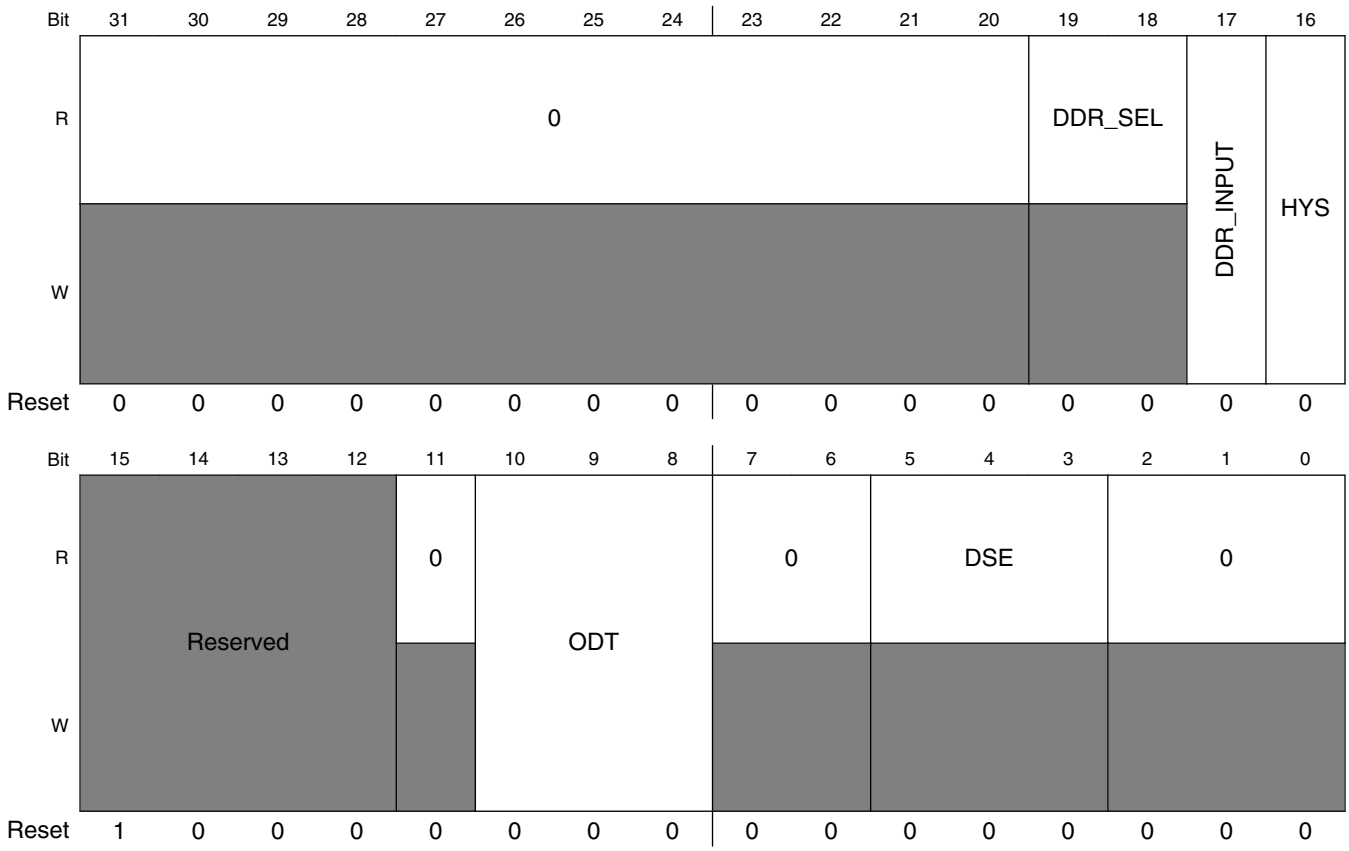
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR04 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A4. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.332 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR05)

Address: 20E_0000h base + 540h offset = 20E_0540h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR05 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A5. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A5.

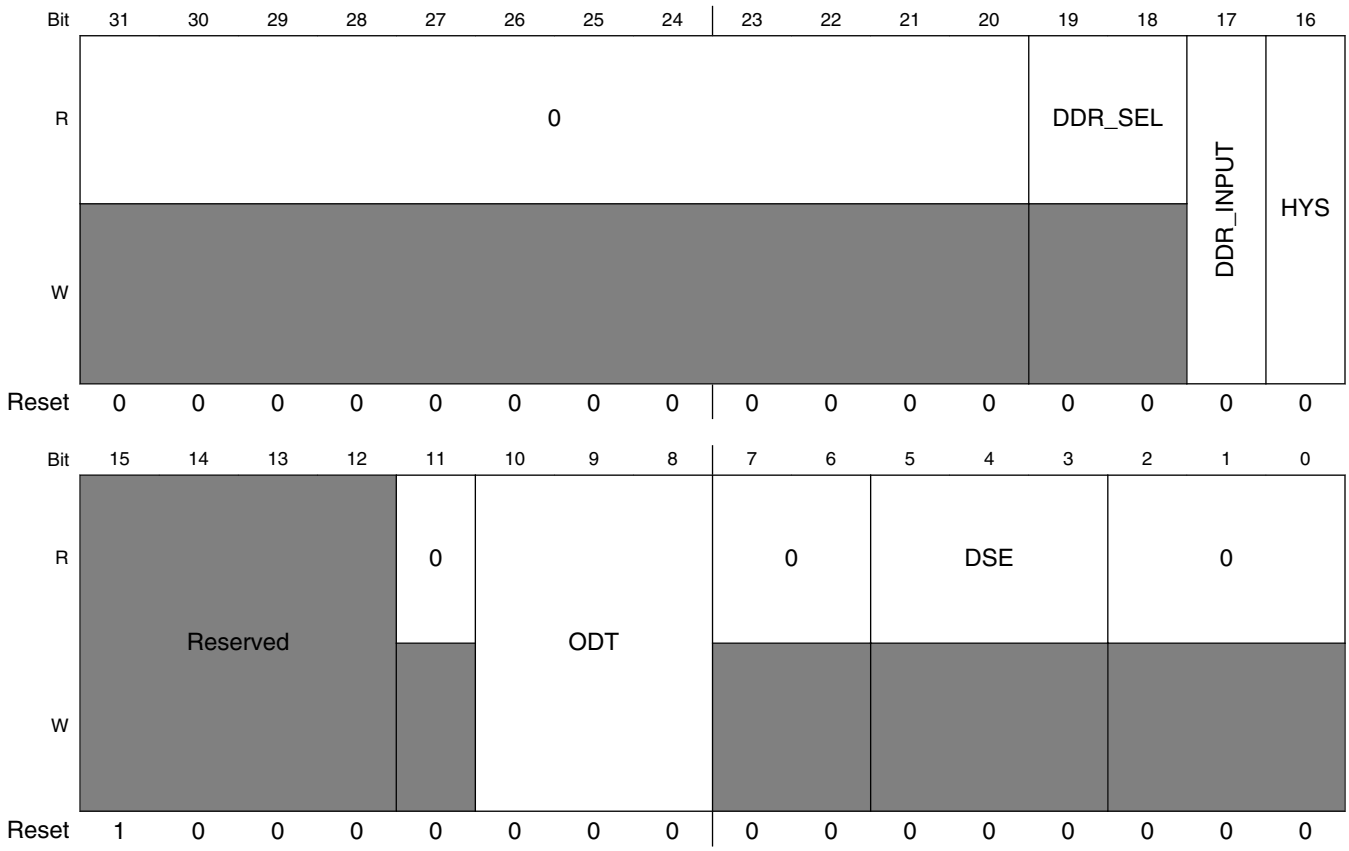
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR05 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A5. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.333 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR06)

Address: 20E_0000h base + 544h offset = 20E_0544h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR06 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A6. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A6.

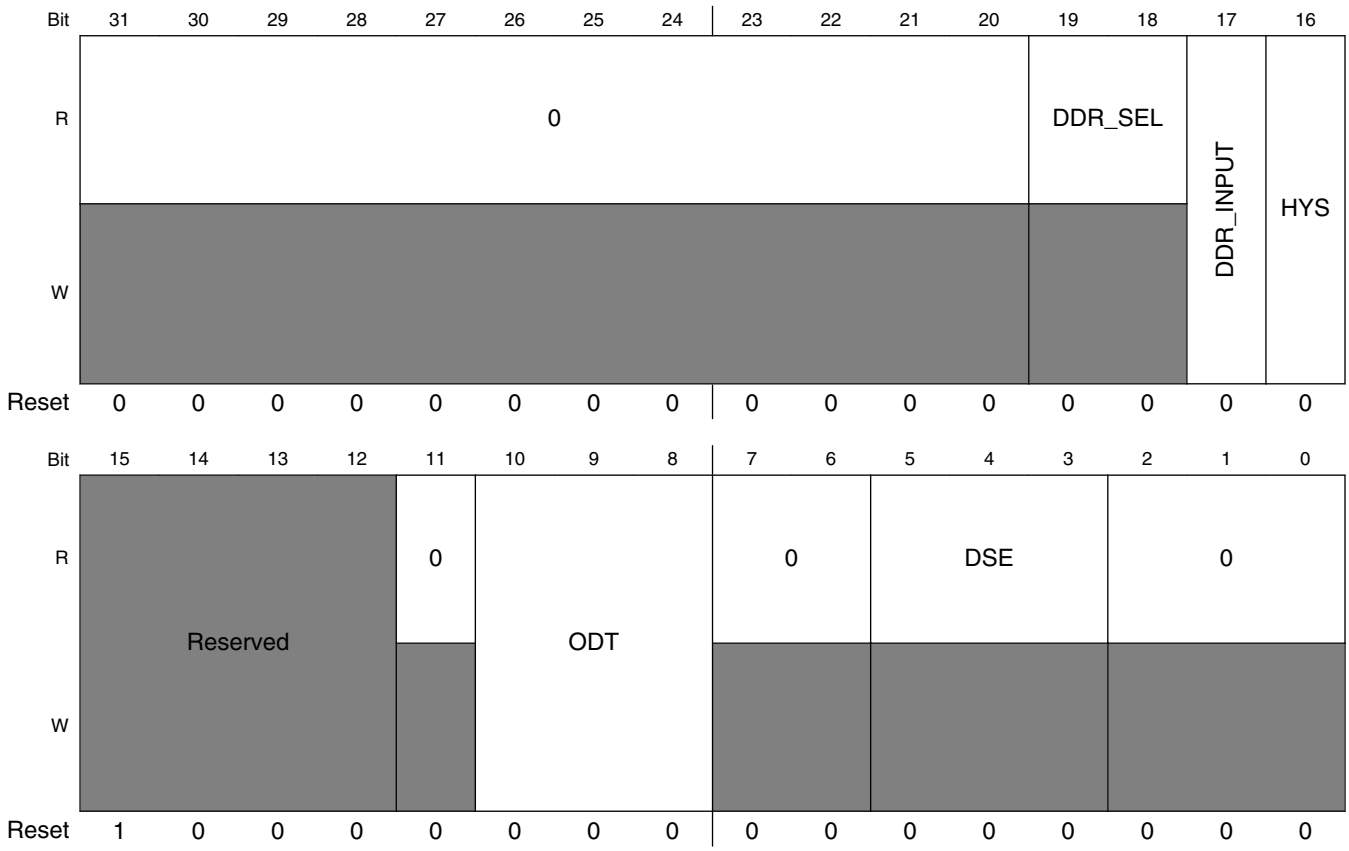
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR06 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A6. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.334 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR07)

Address: 20E_0000h base + 548h offset = 20E_0548h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR07 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A7. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A7.

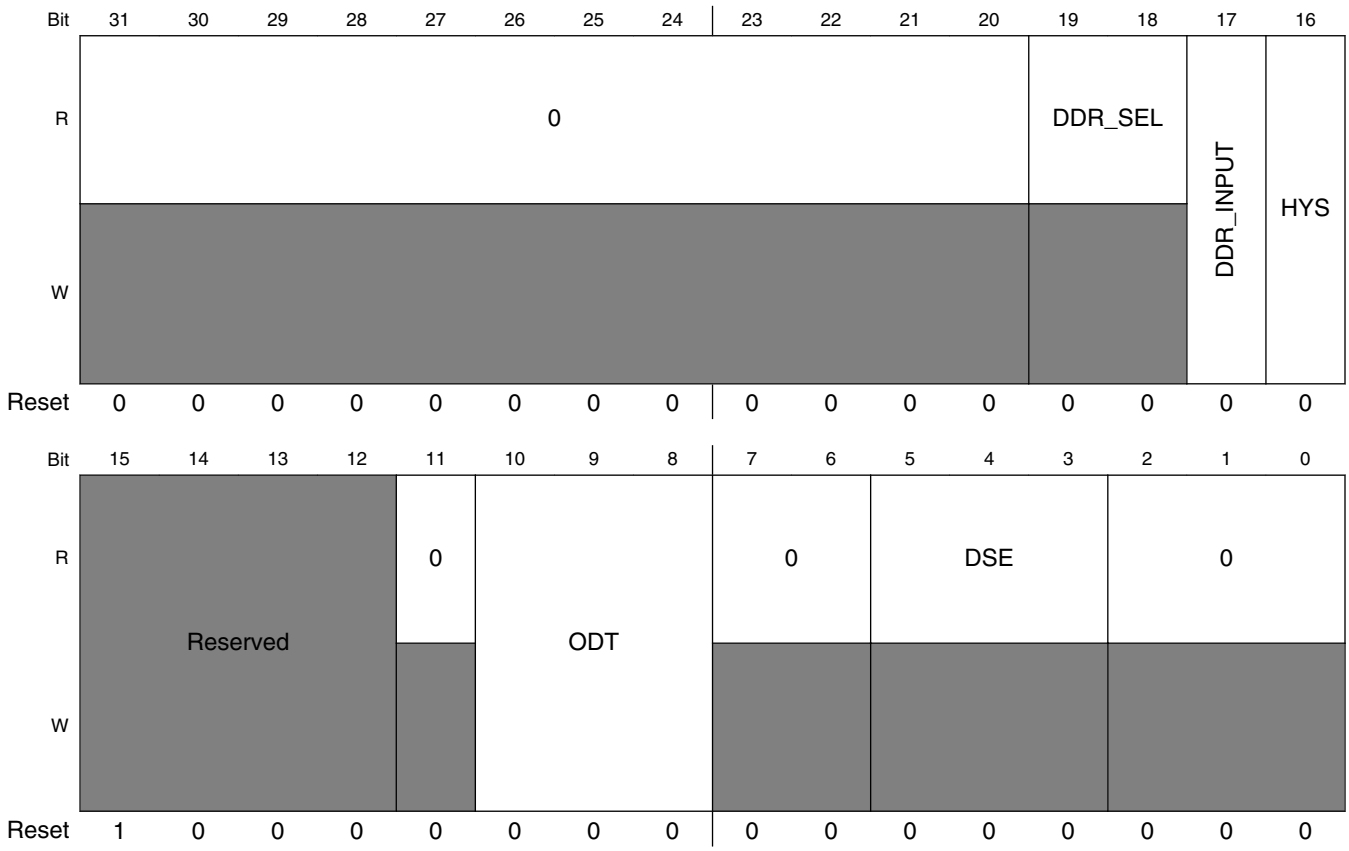
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR07 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A7. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.335 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR08)

Address: 20E_0000h base + 54Ch offset = 20E_054Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR08 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A8. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A8.

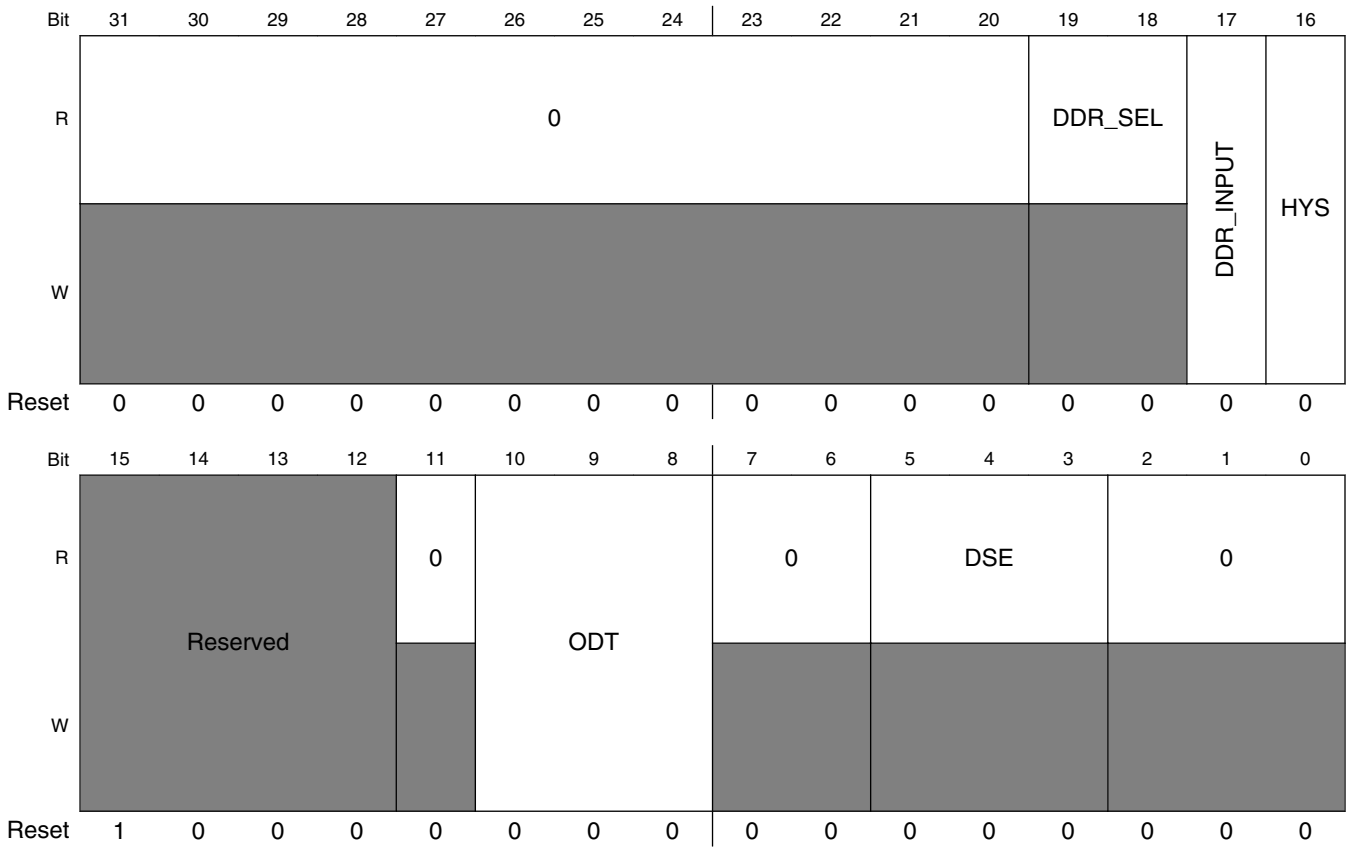
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR08 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A8. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.336 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR09)

Address: 20E_0000h base + 550h offset = 20E_0550h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR09 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A9. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A9.

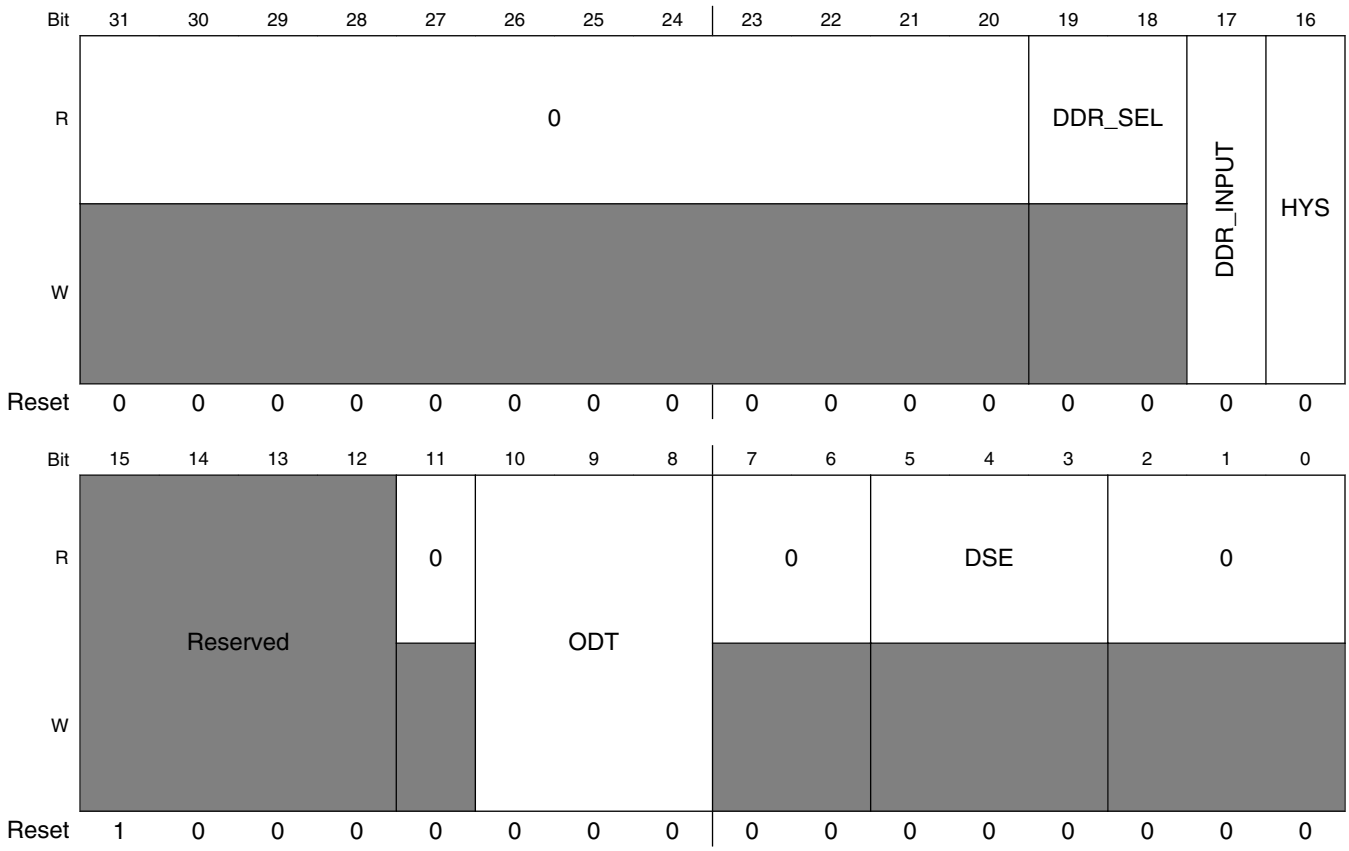
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR09 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A9. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.337 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR10)

Address: 20E_0000h base + 554h offset = 20E_0554h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR10 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A10. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A10.

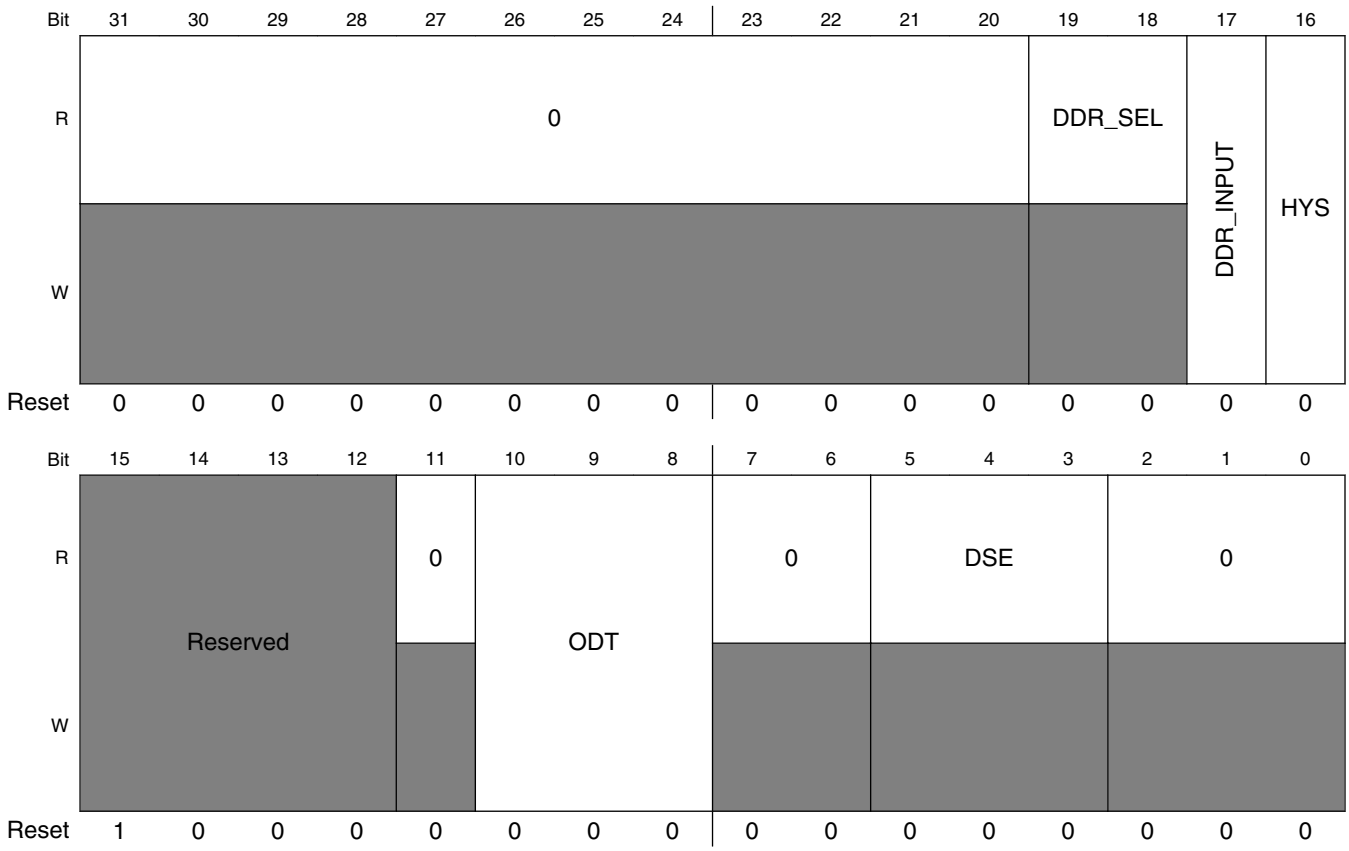
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR10 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A10. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.338 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR11)

Address: 20E_0000h base + 558h offset = 20E_0558h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR11 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A11. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A11.

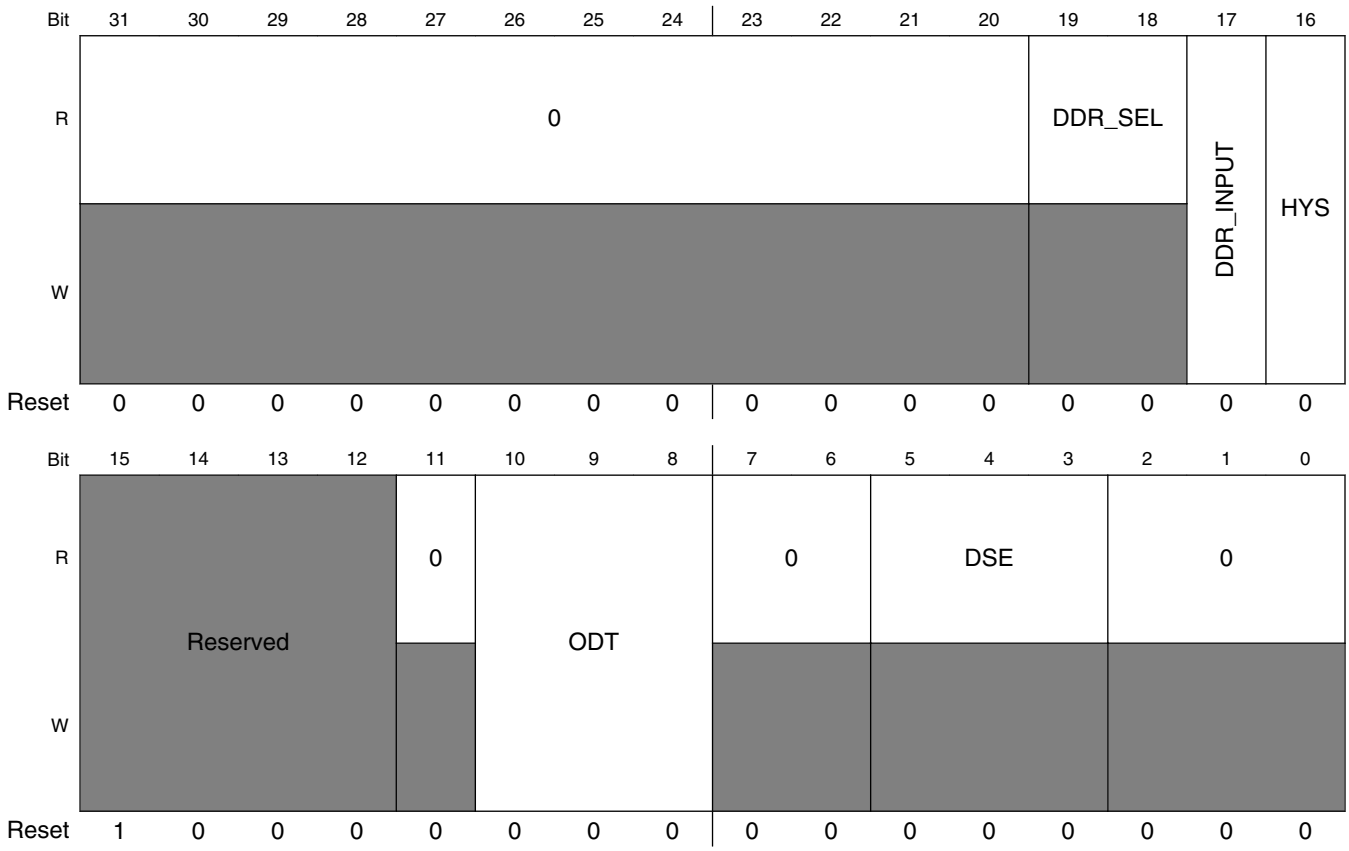
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR11 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A11. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.339 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR12)

Address: 20E_0000h base + 55Ch offset = 20E_055Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR12 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A12. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A12.

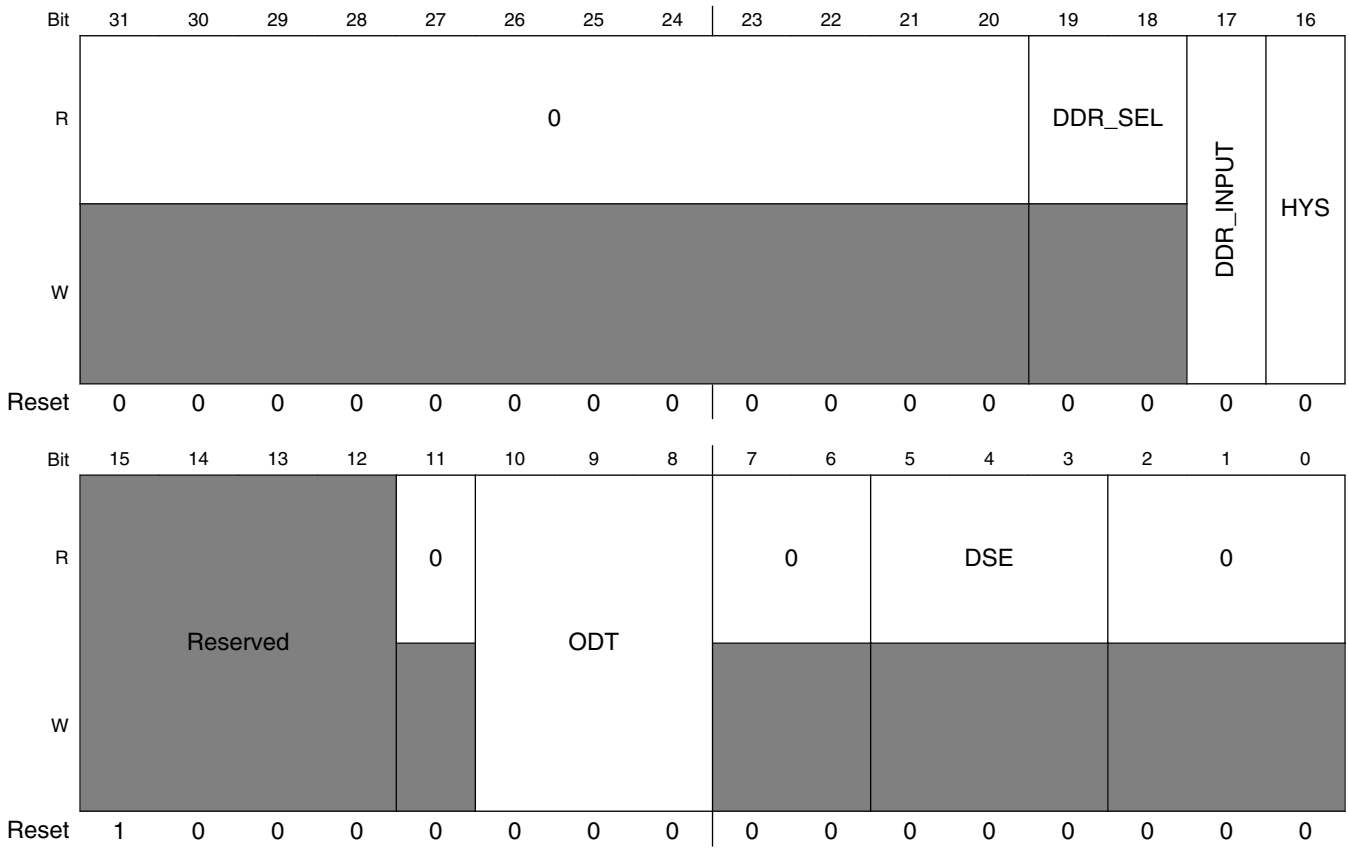
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR12 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A12. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.340 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR13)

Address: 20E_0000h base + 560h offset = 20E_0560h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR13 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A13. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A13.

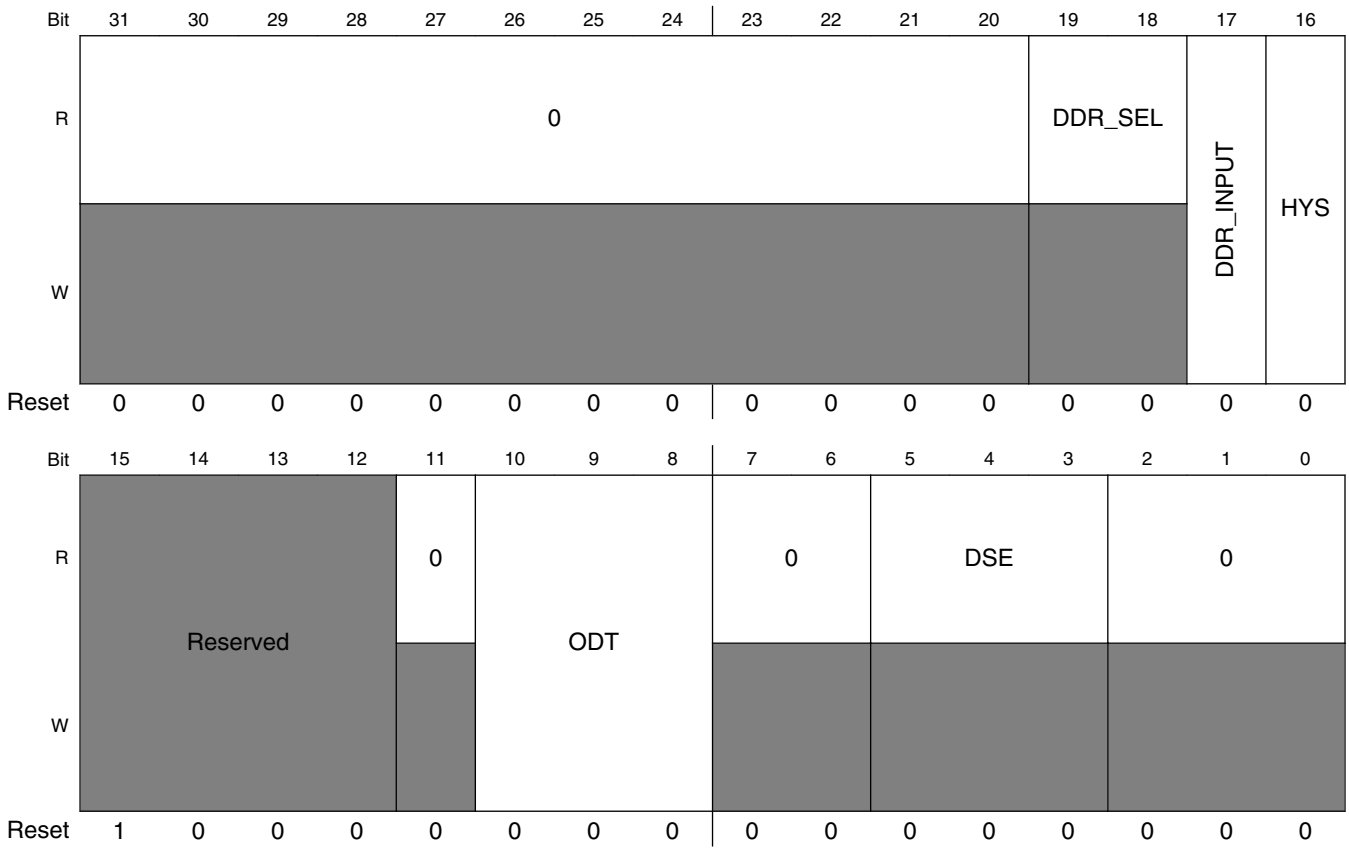
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR13 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A13. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.341 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR14)

Address: 20E_0000h base + 564h offset = 20E_0564h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR14 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A14. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A14.

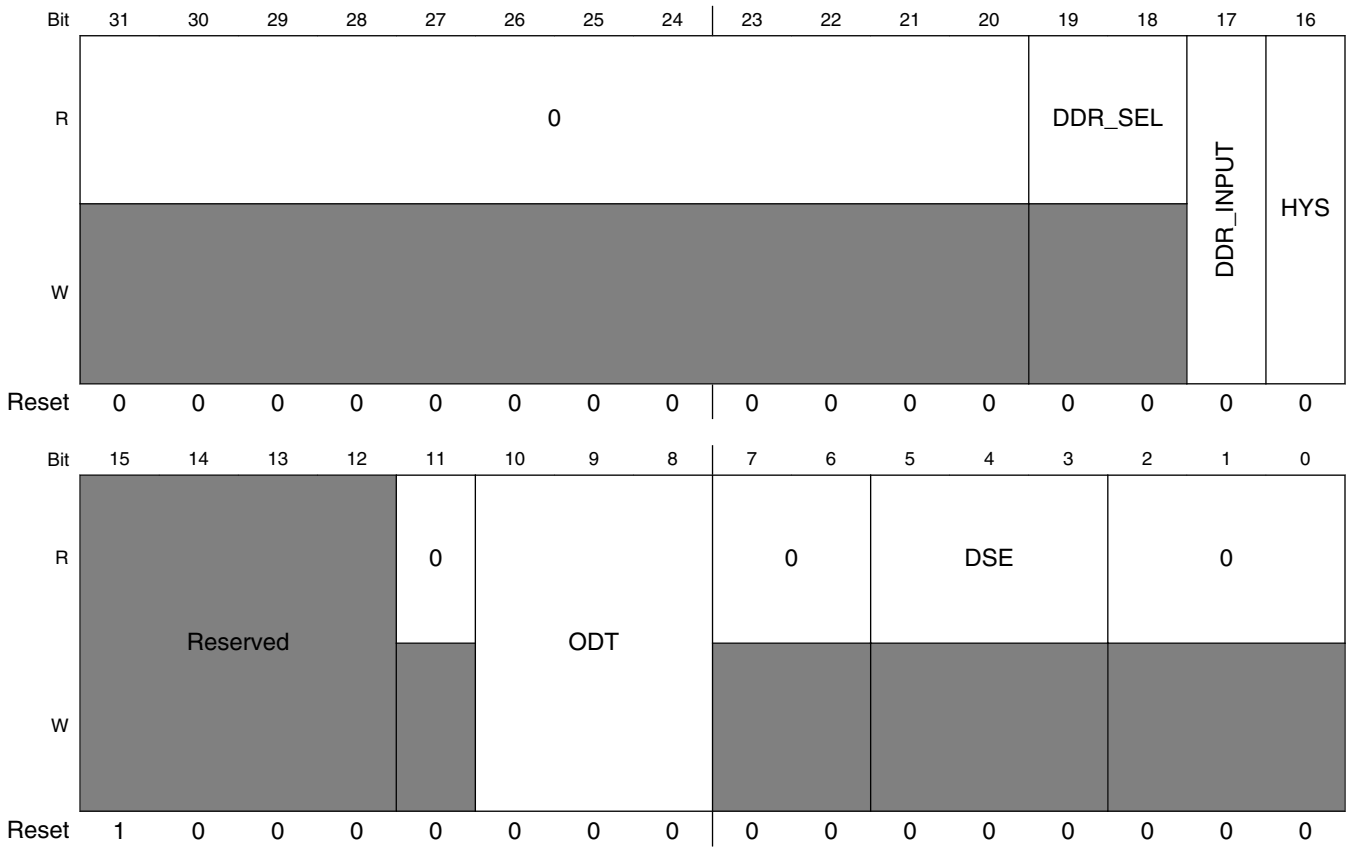
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR14 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A14. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.342 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR15)

Address: 20E_0000h base + 568h offset = 20E_0568h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR15 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_A15. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_A15.

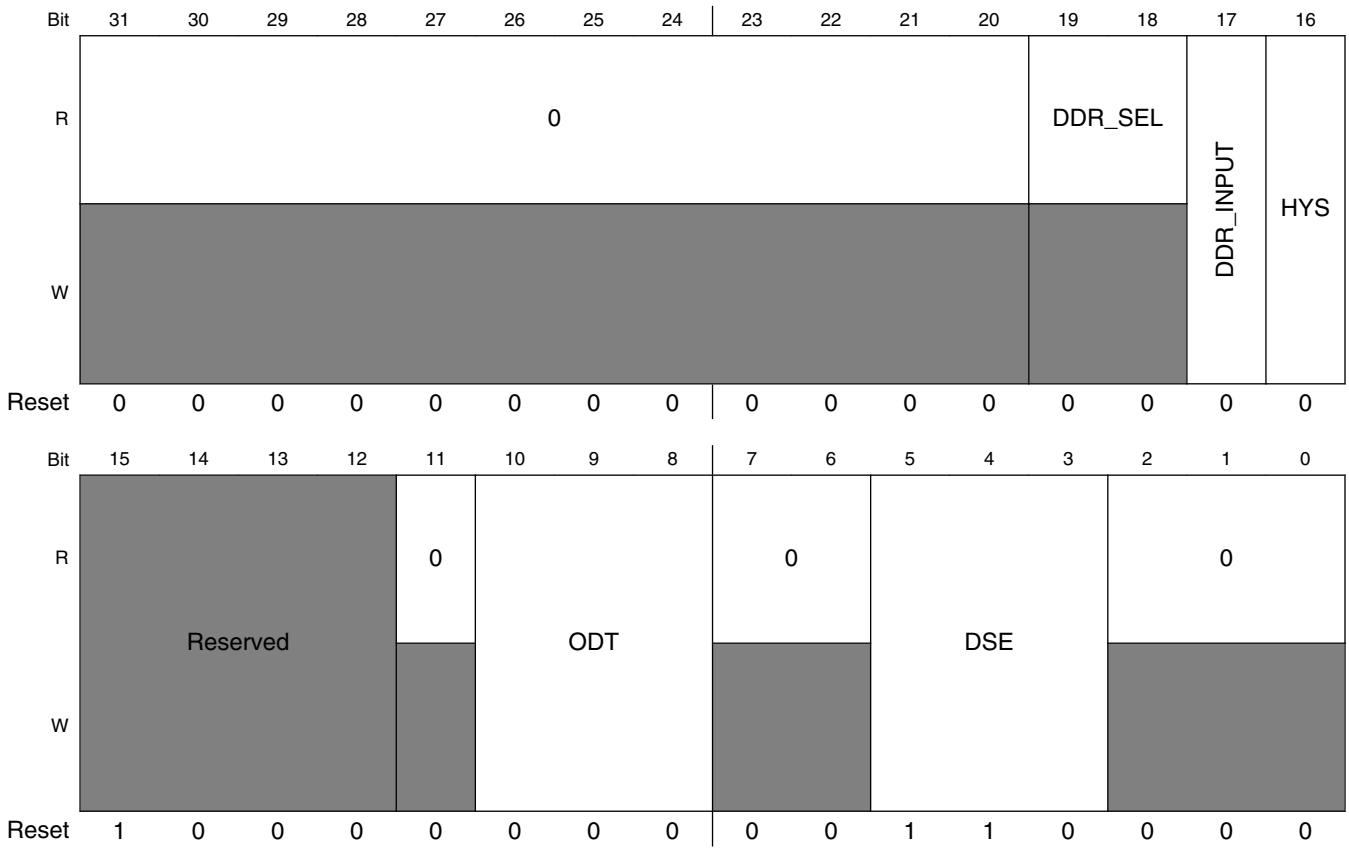
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ADDR15 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_A15. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.343 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CAS_B)

Address: 20E_0000h base + 56Ch offset = 20E_056Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_CAS_B field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_CAS. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_CAS.

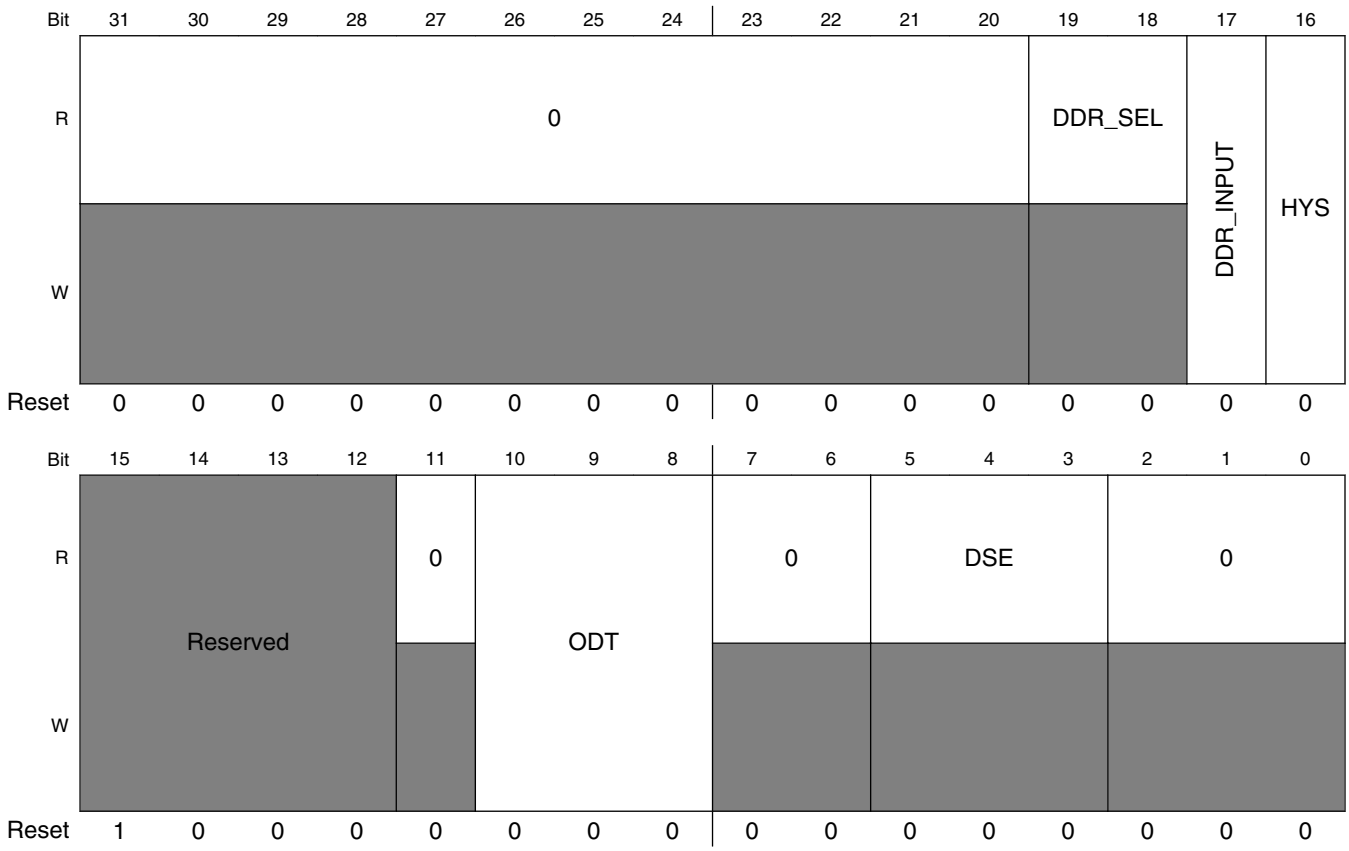
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_CAS_B field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_CAS. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_CAS. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.344 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CS0_B)

Address: 20E_0000h base + 570h offset = 20E_0570h



IOMUXC_SW_PAD_CTL_PAD_DRAM_CS0_B field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_CS0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_CS0.

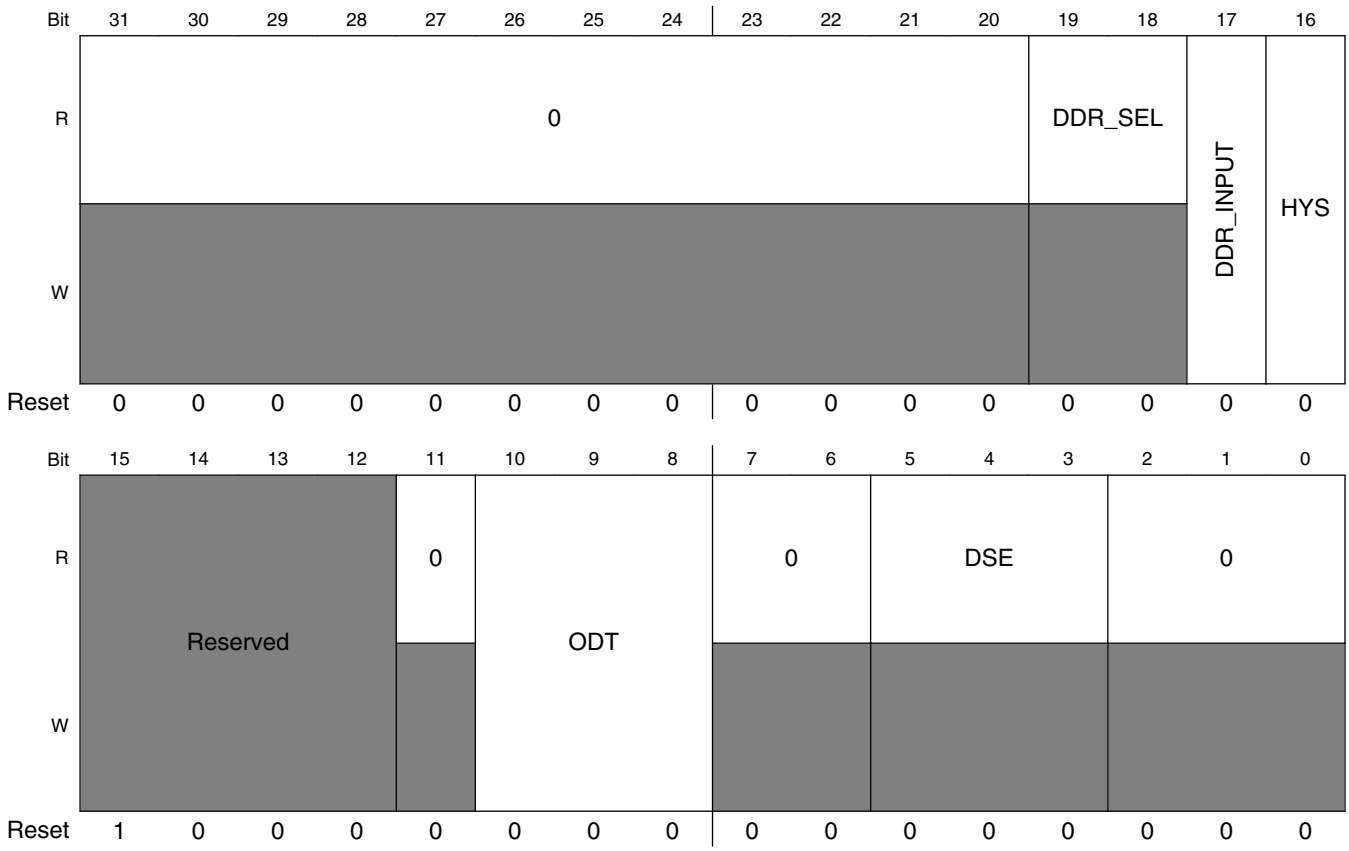
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_CS0_B field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_CS0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.345 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_CS1_B)

Address: 20E_0000h base + 574h offset = 20E_0574h



IOMUXC_SW_PAD_CTL_PAD_DRAM_CS1_B field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_CS1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_CS1.

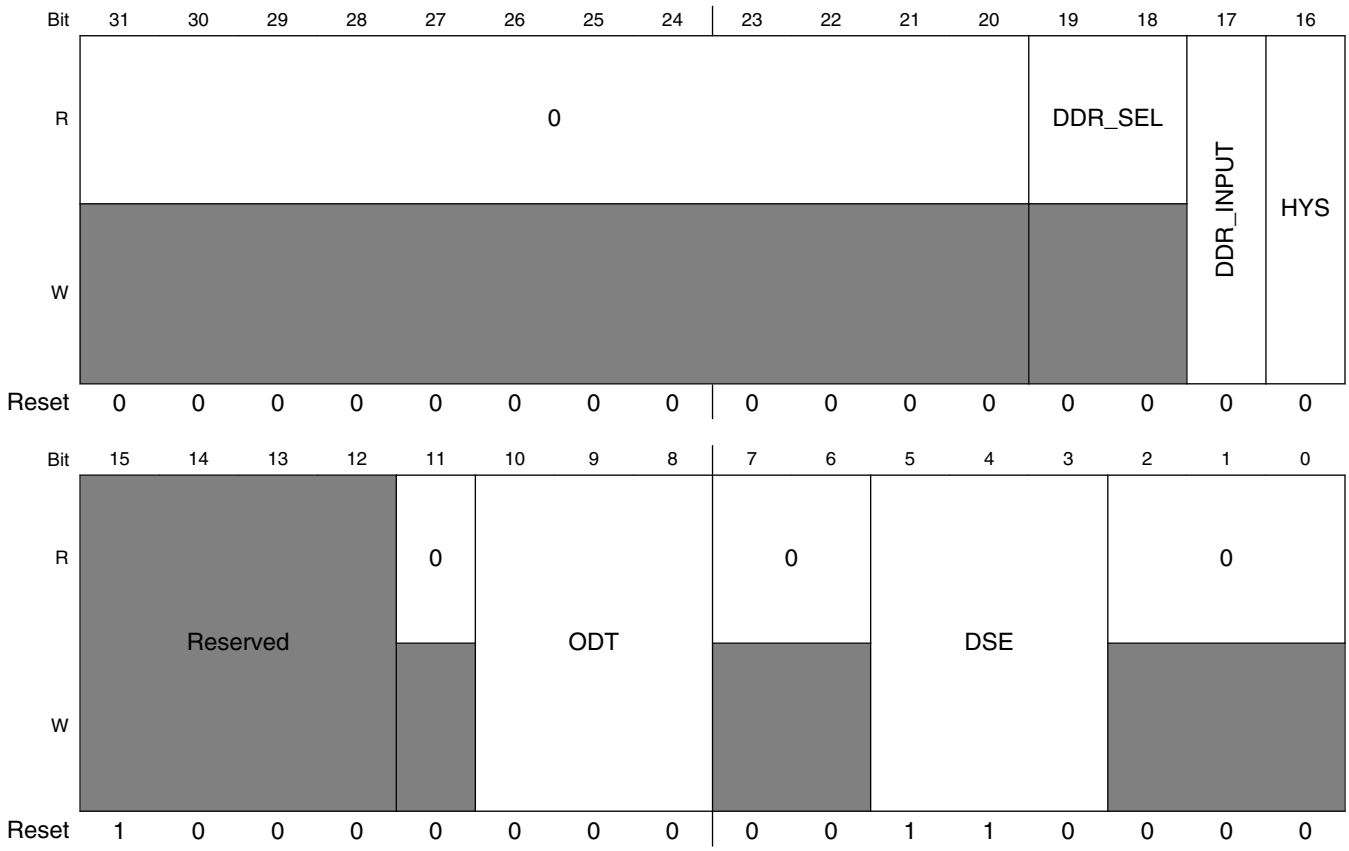
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_CS1_B field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_CS1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.346 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_RAS_B)

Address: 20E_0000h base + 578h offset = 20E_0578h



IOMUXC_SW_PAD_CTL_PAD_DRAM_RAS_B field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_RAS. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_RAS.

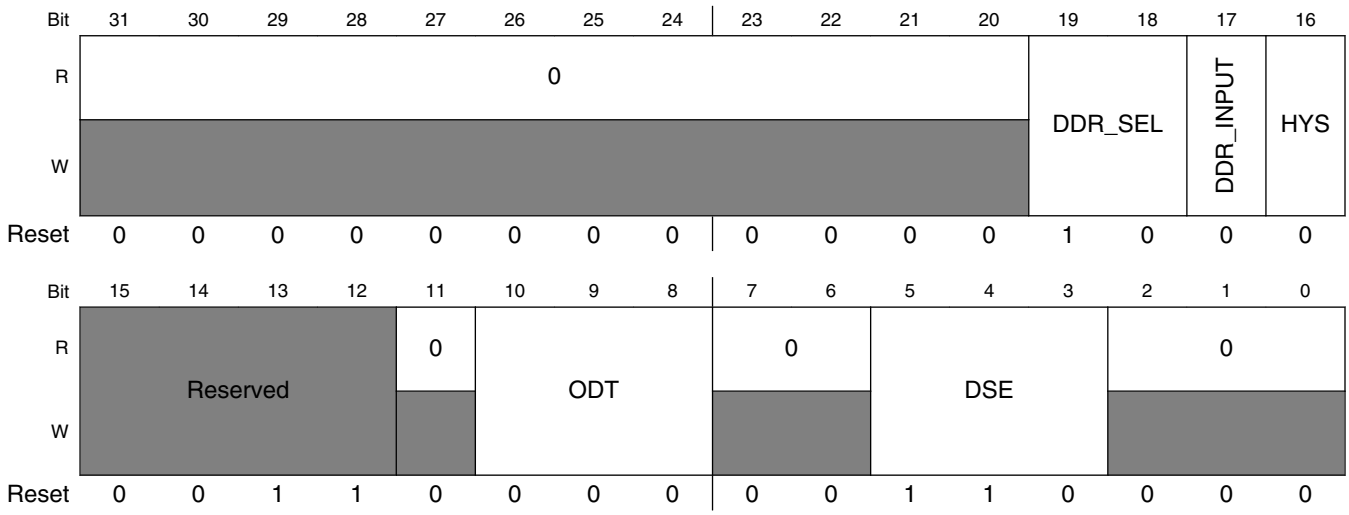
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_RAS_B field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_RAS. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_RAS. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.347 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_RESET)

Address: 20E_0000h base + 57Ch offset = 20E_057Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_RESET field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field Select one of next values for pad: DRAM_RESET. 00 **DDR3_LPDDR2** — DDR3 and LPDDR2 mode. 01 **RESERVED1** — Reserved 10 **RESERVED2** — Reserved 11 **RESERVED3** — Reserved
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_RESET. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_RESET. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.

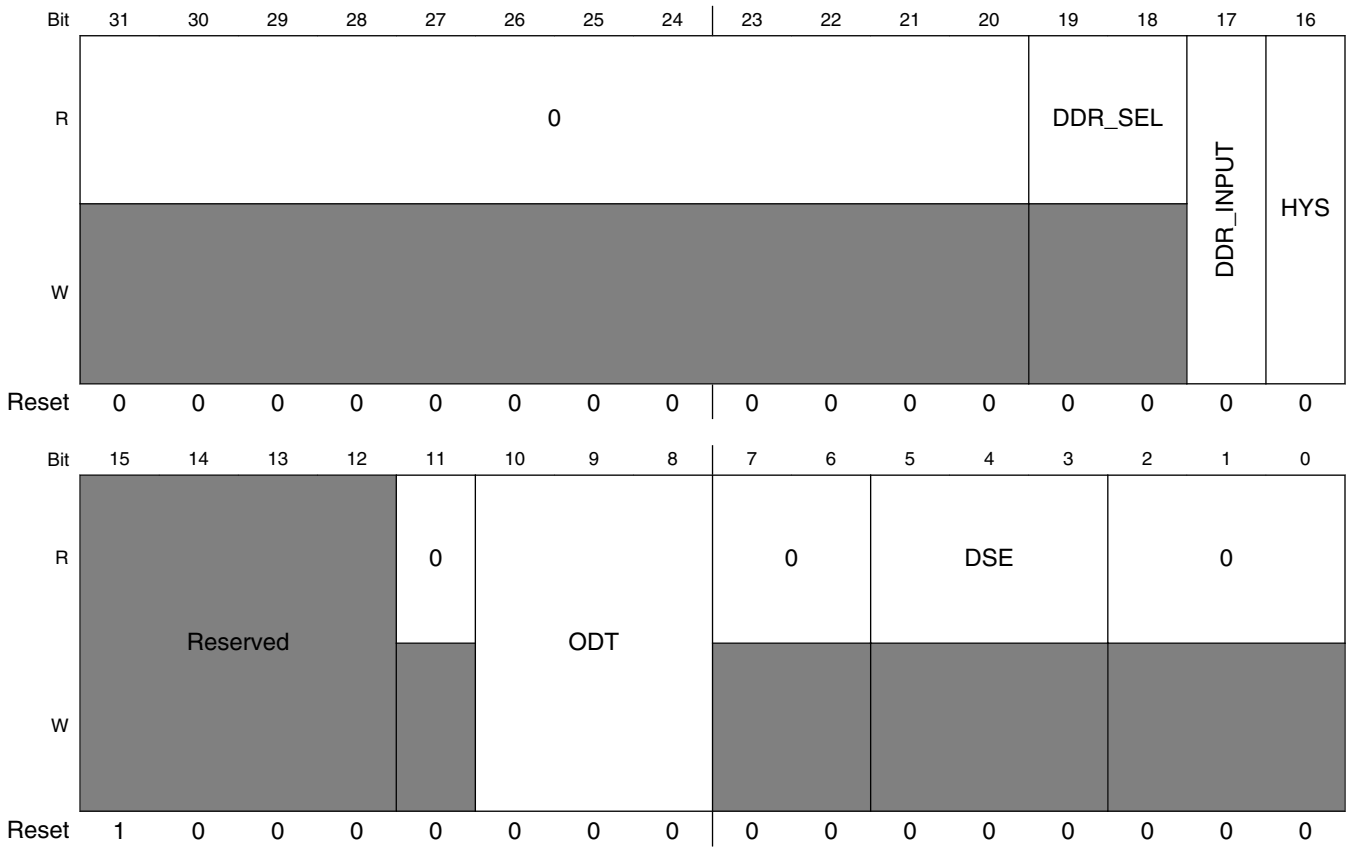
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_RESET field descriptions (continued)

Field	Description
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_RESET. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_RESET. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.348 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA0)

Address: 20E_0000h base + 580h offset = 20E_0580h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDBA0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDBA0.

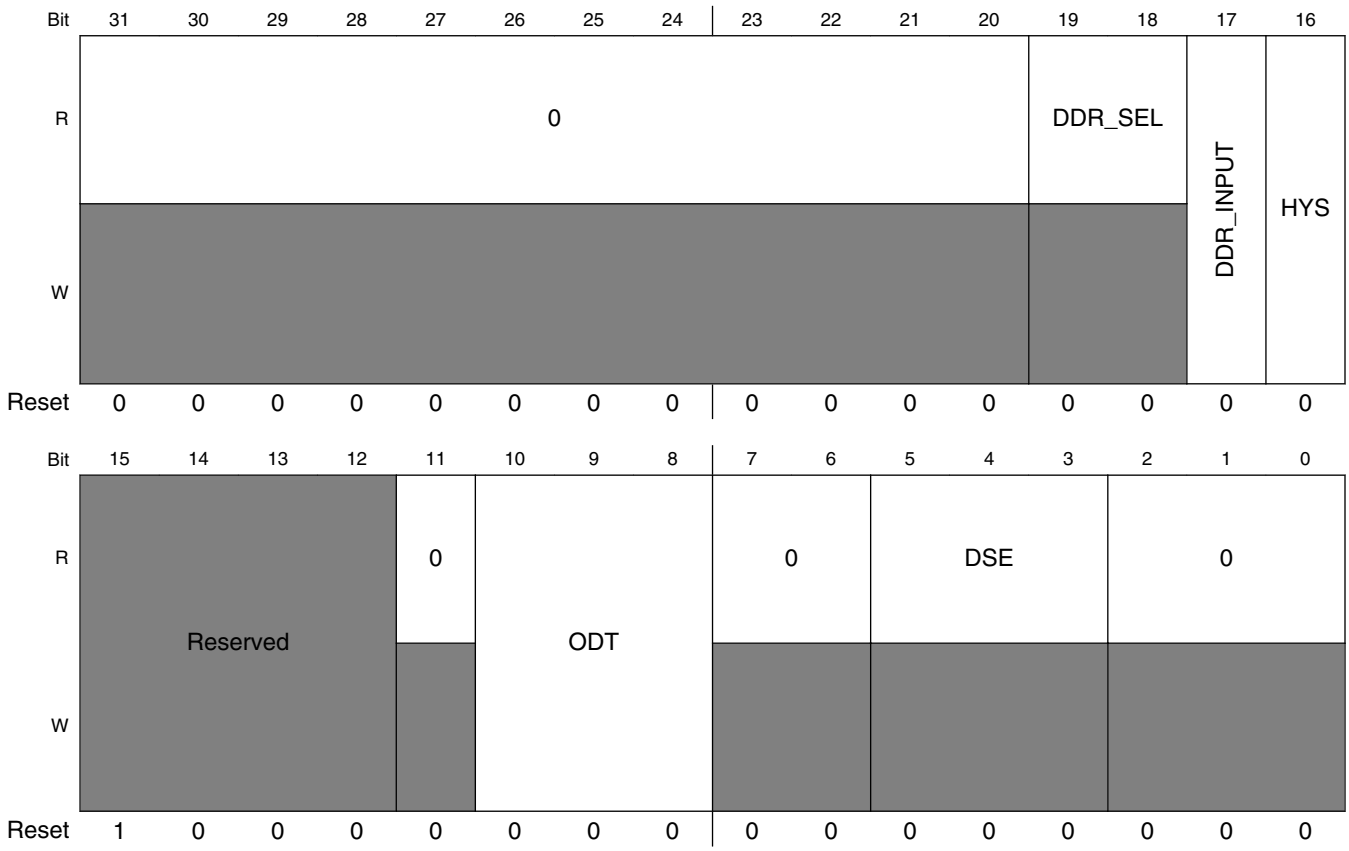
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA0 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDBA0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.349 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA1)

Address: 20E_0000h base + 584h offset = 20E_0584h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDBA1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDBA1.

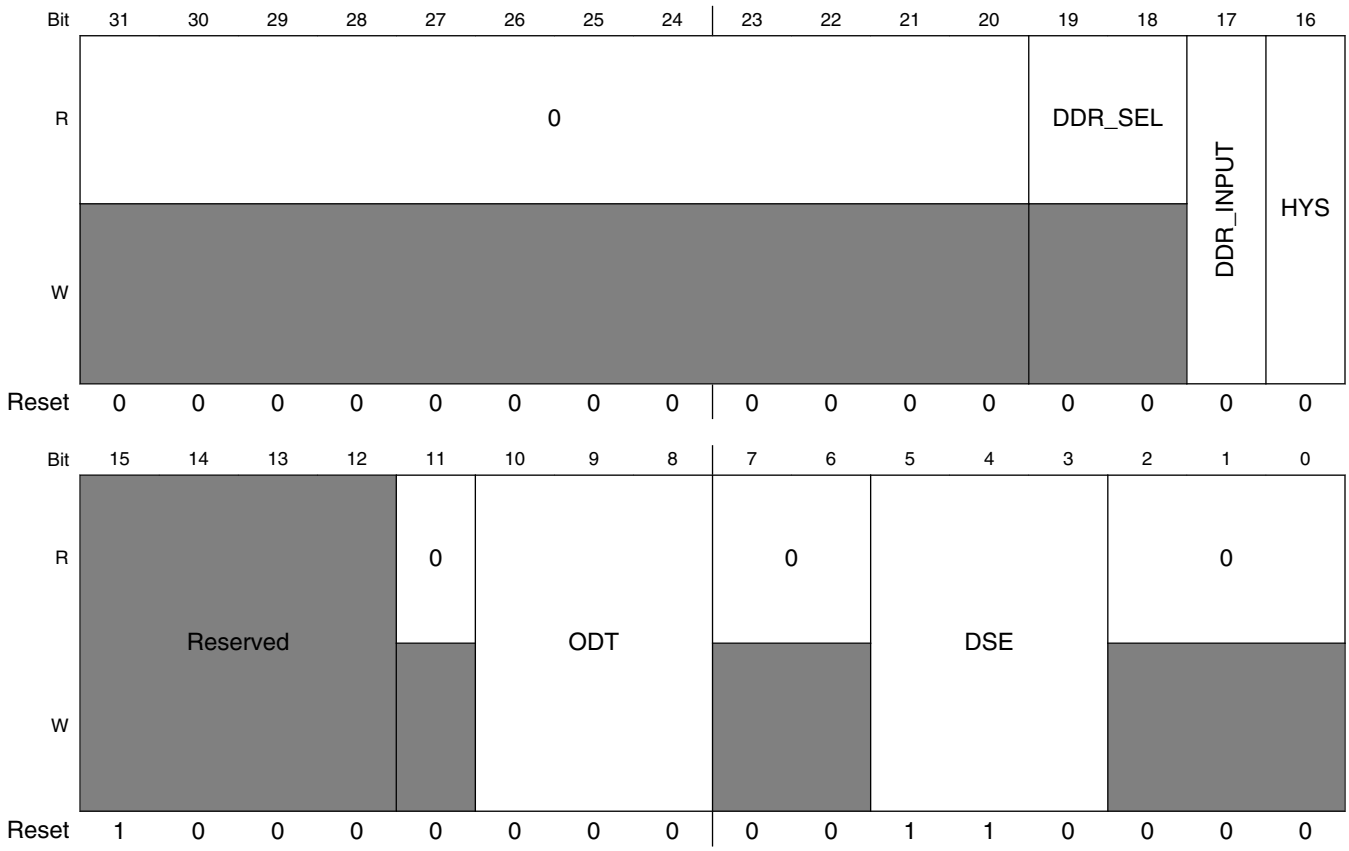
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA1 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDBA1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_ADDDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.350 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK0_P)

Address: 20E_0000h base + 588h offset = 20E_0588h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK0_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDCLK_0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDCLK_0.

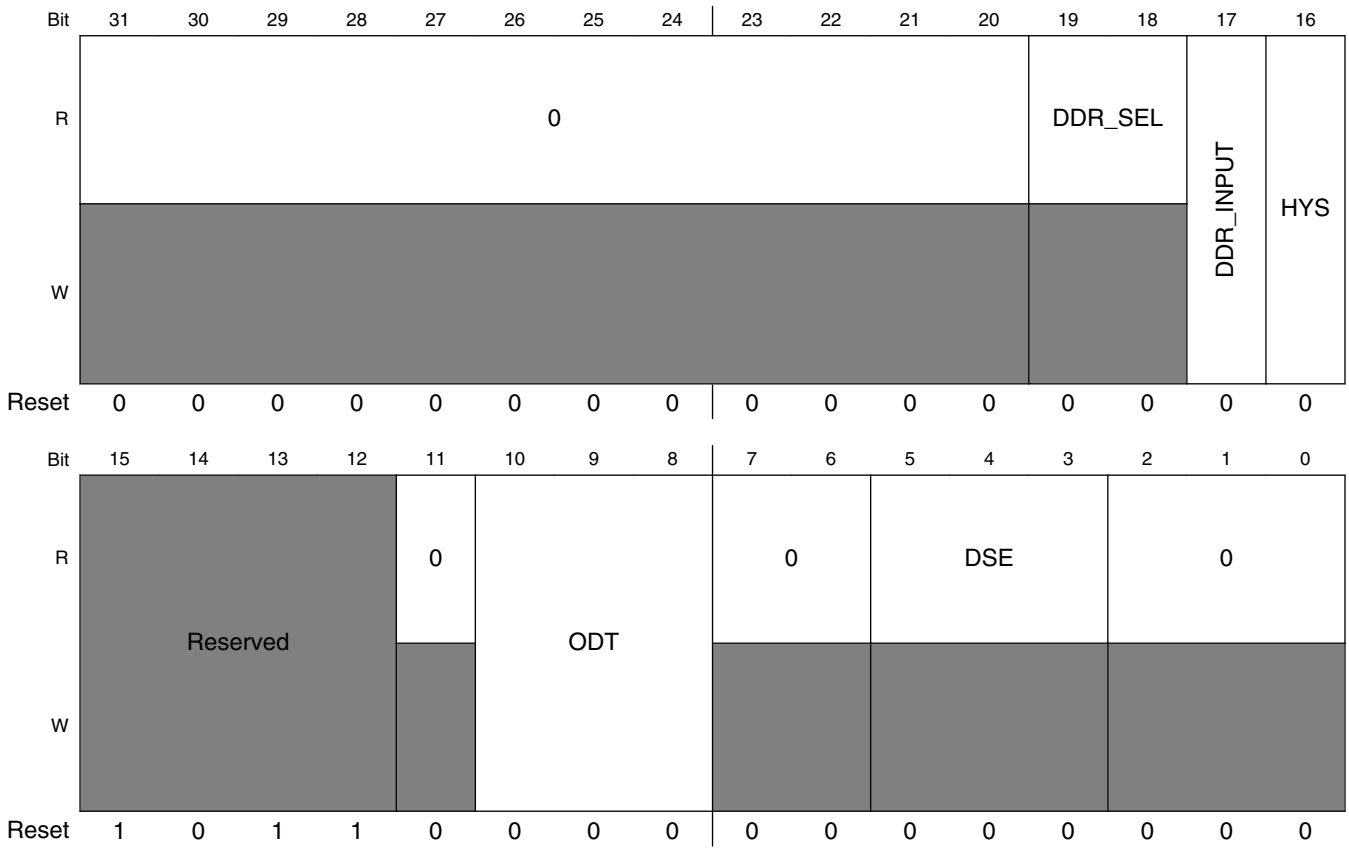
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK0_P field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDCLK_0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDCLK_0. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.351 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA2)

Address: 20E_0000h base + 58Ch offset = 20E_058Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA2 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDBA2. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDBA2.

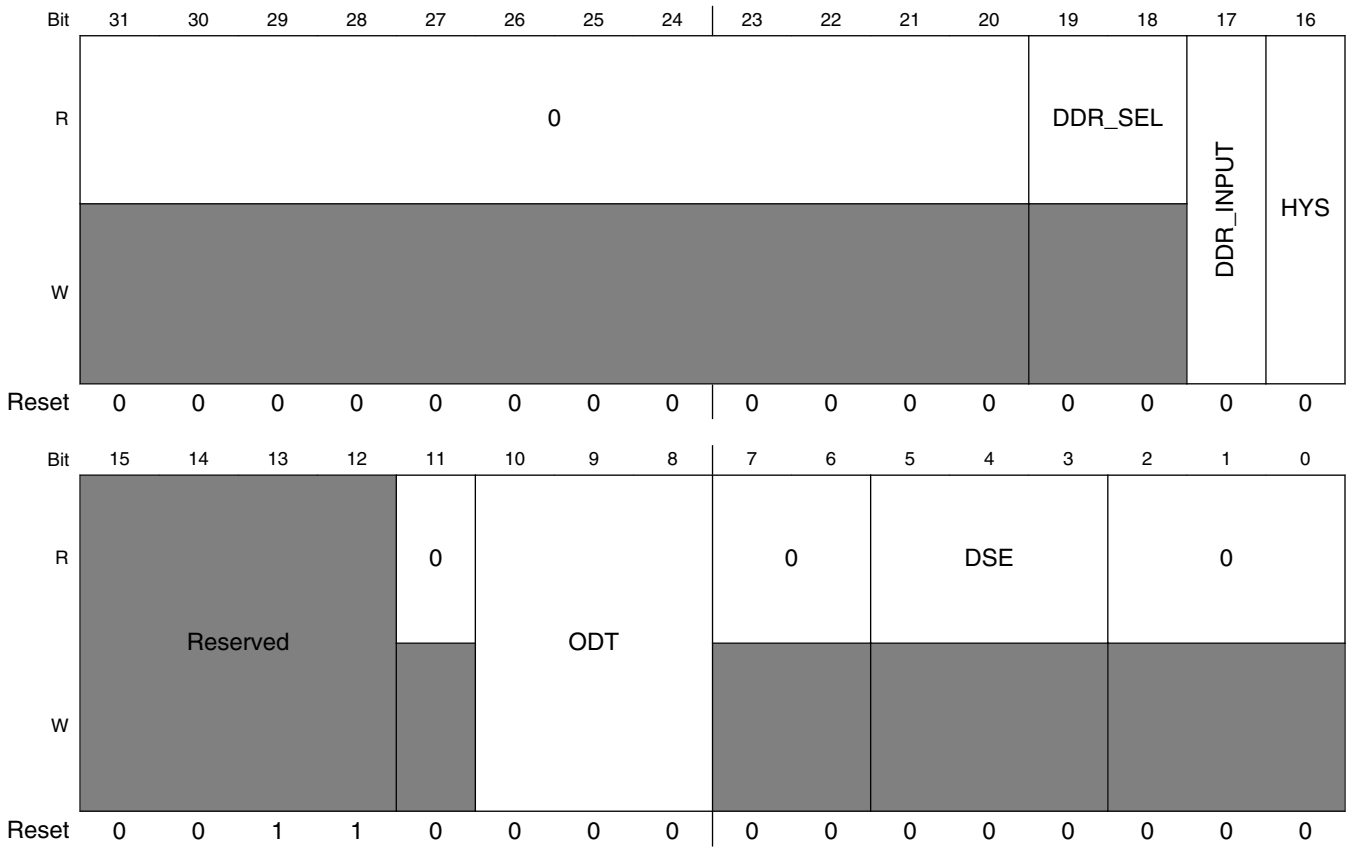
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDBA2 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDBA2. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.352 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE0)

Address: 20E_0000h base + 590h offset = 20E_0590h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDCKE0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDCKE0.

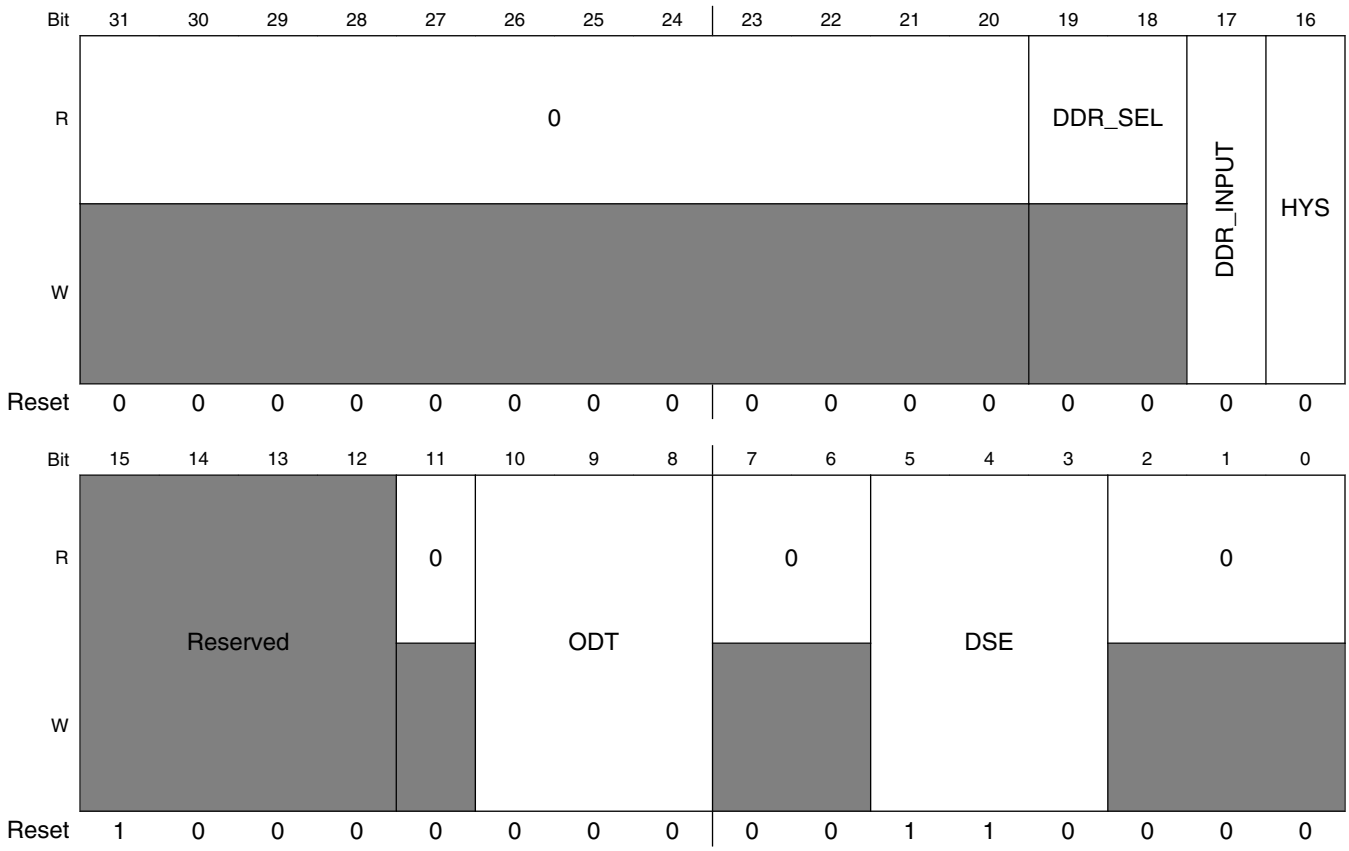
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE0 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDCKE0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.353 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK1_P)

Address: 20E_0000h base + 594h offset = 20E_0594h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK1_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDCLK_1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDCLK_1.

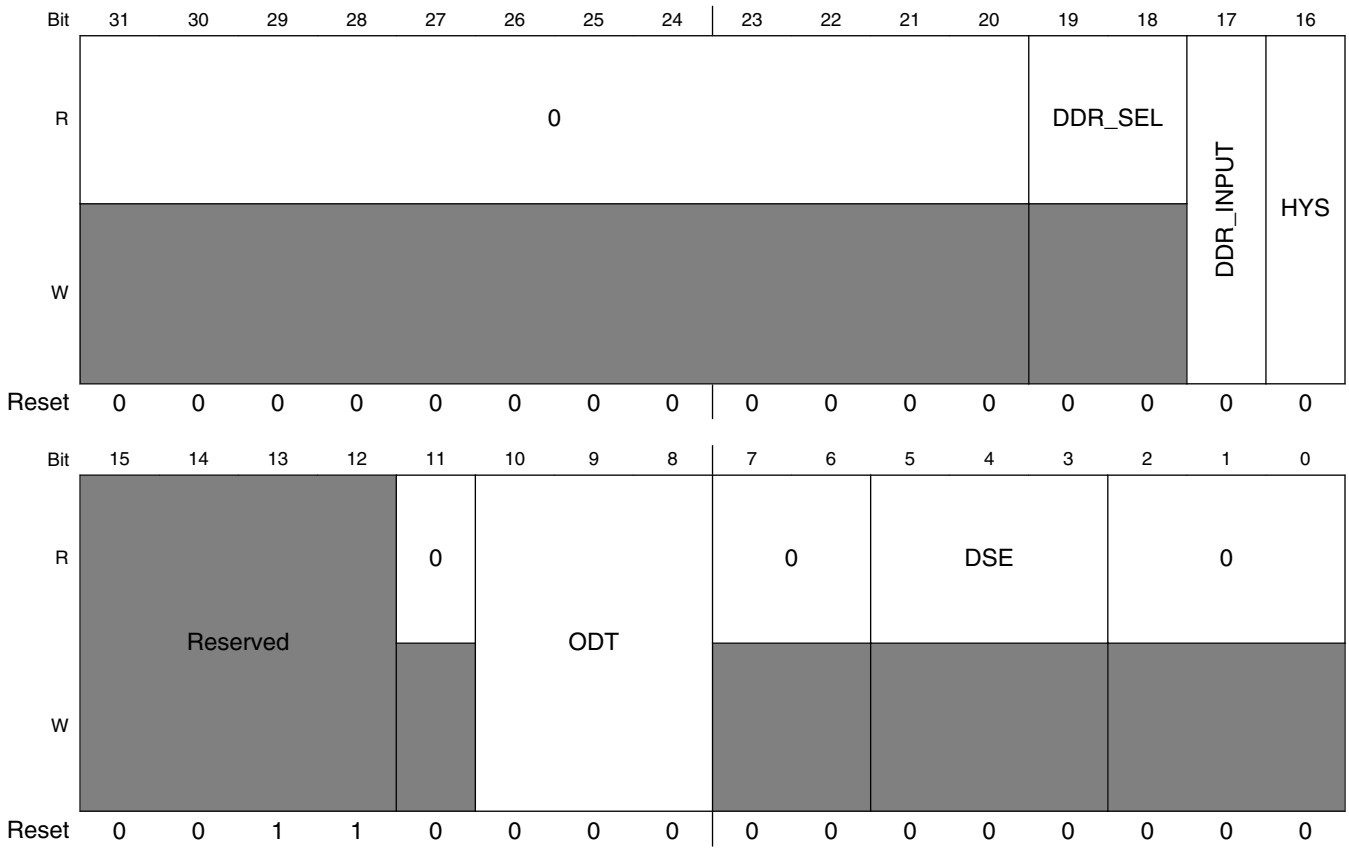
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCLK1_P field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDCLK_1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDCLK_1. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.354 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE1)

Address: 20E_0000h base + 598h offset = 20E_0598h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDCKE1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDCKE1.

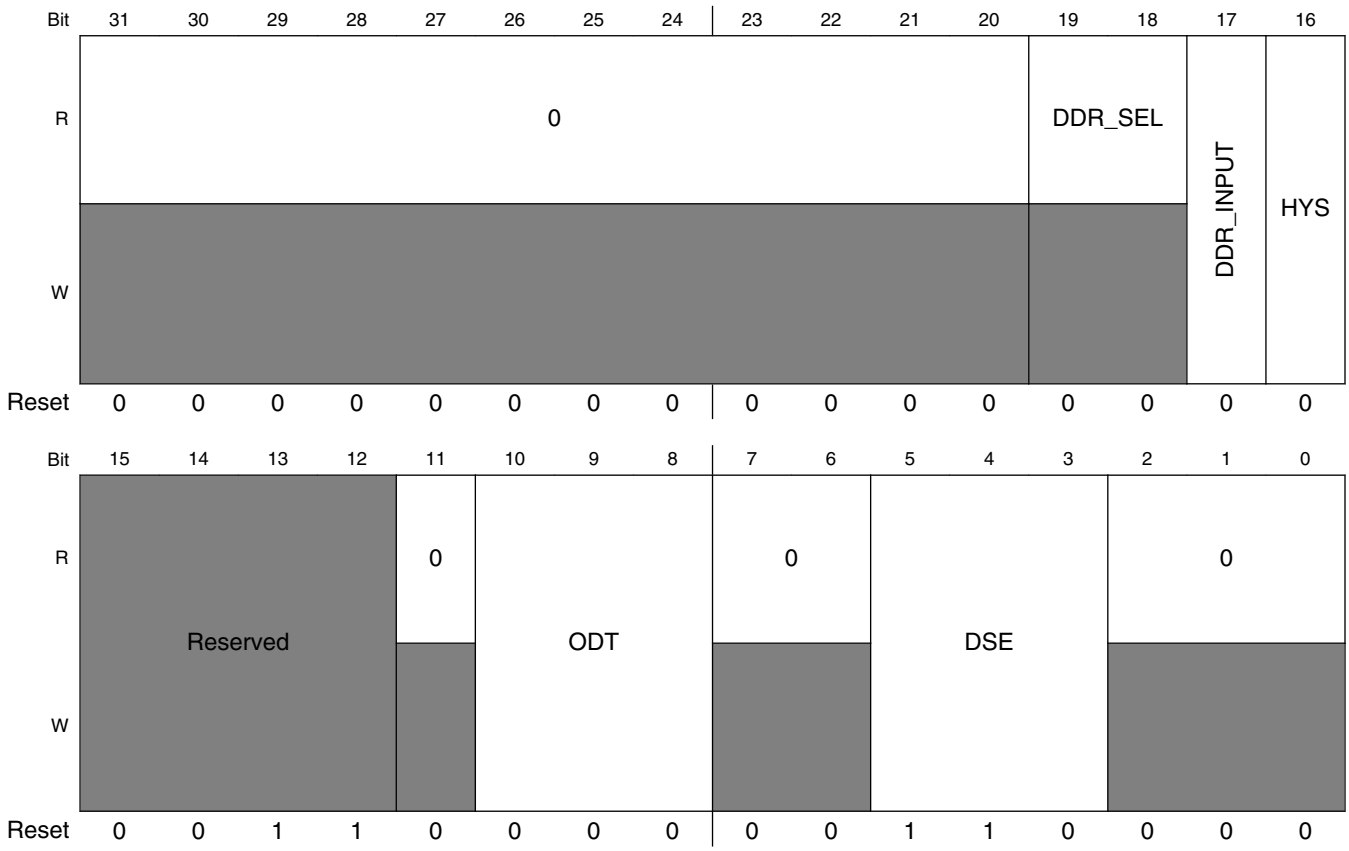
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDCKE1 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDCKE1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.355 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT0)

Address: 20E_0000h base + 59Ch offset = 20E_059Ch



IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDODT0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDODT0.

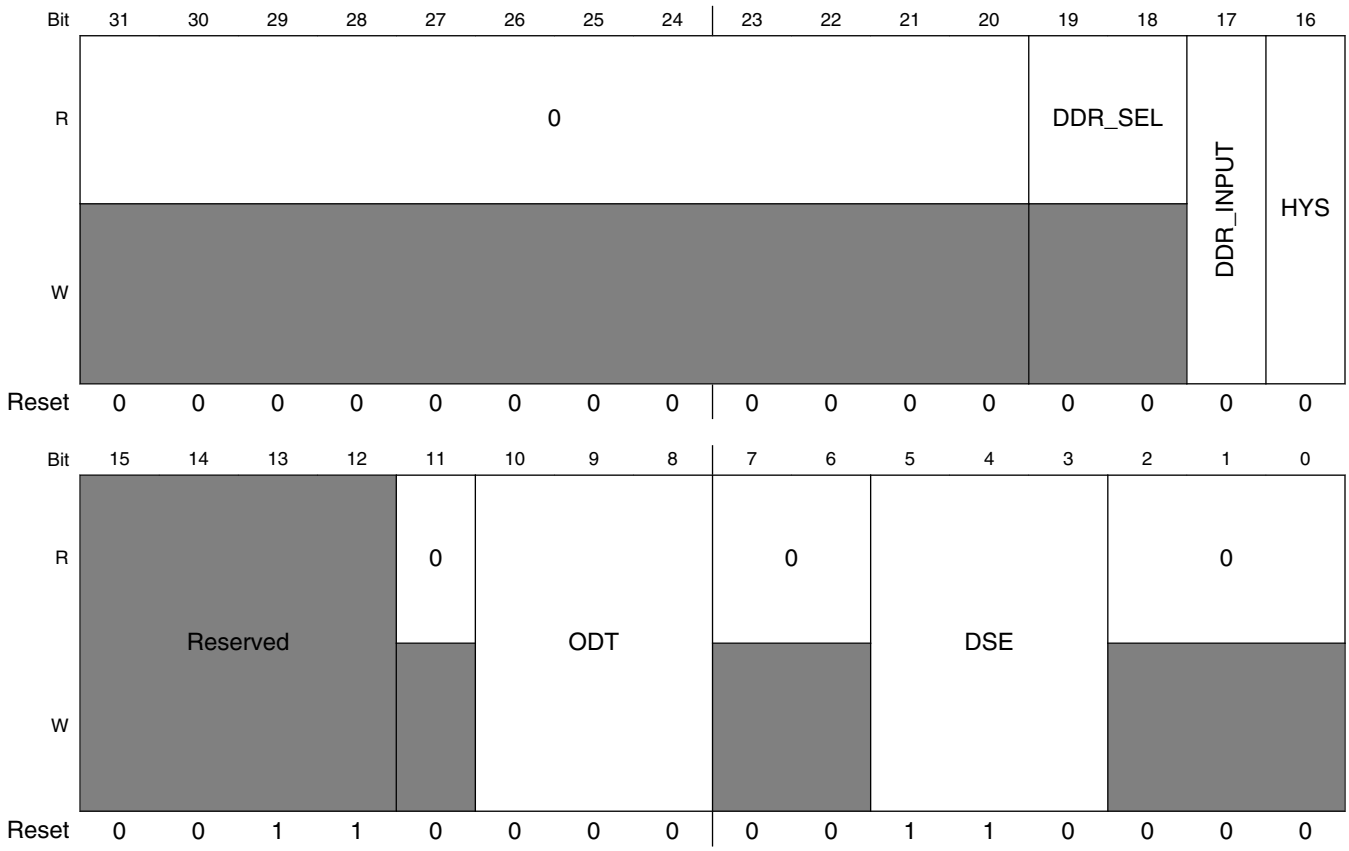
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT0 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDODT0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDODT0. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.356 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT1)

Address: 20E_0000h base + 5A0h offset = 20E_05A0h



IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDODT1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDODT1.

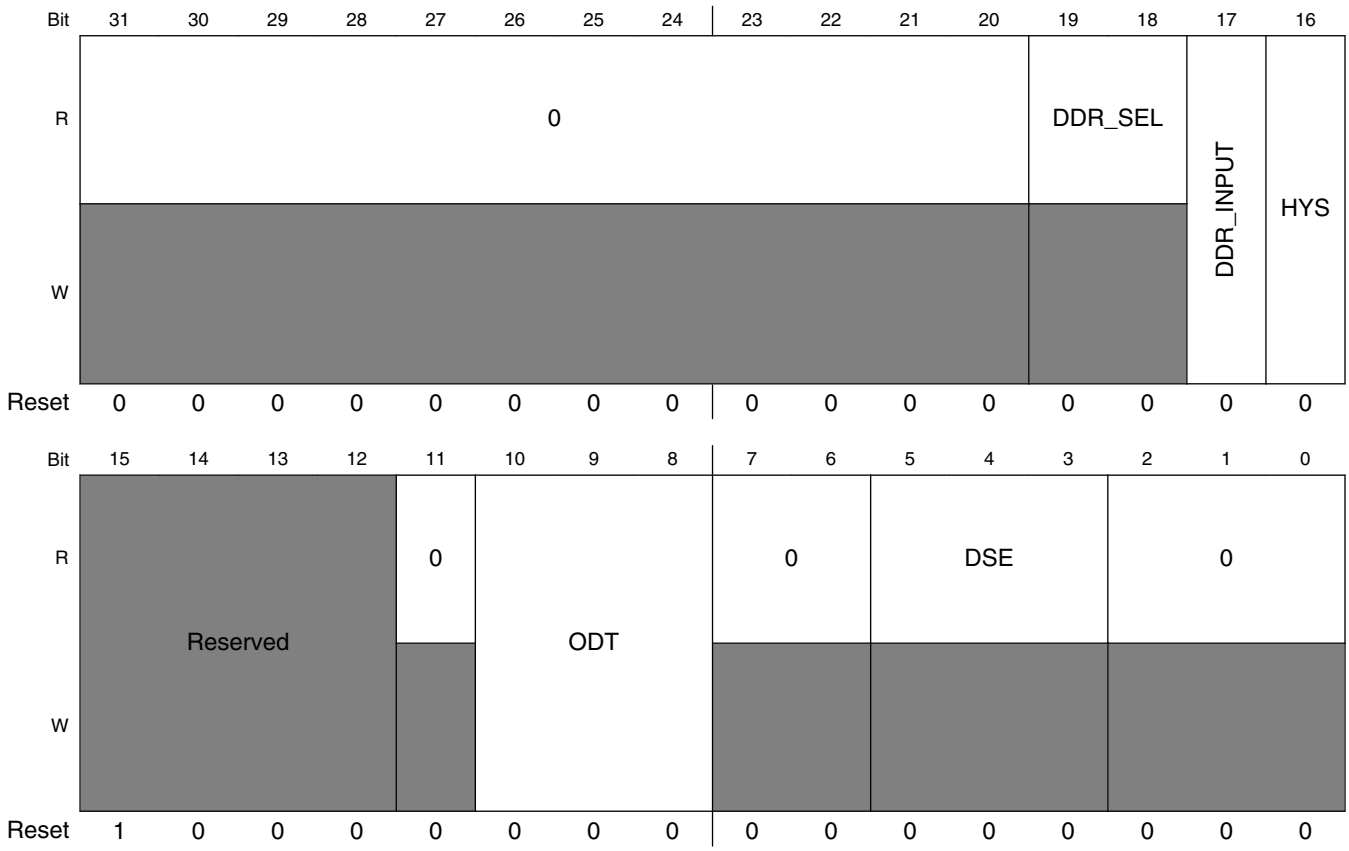
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_ODT1 field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDODT1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDODT1. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.357 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDWE_B)

Address: 20E_0000h base + 5A4h offset = 20E_05A4h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDWE_B field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_SDWE. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_SDWE.

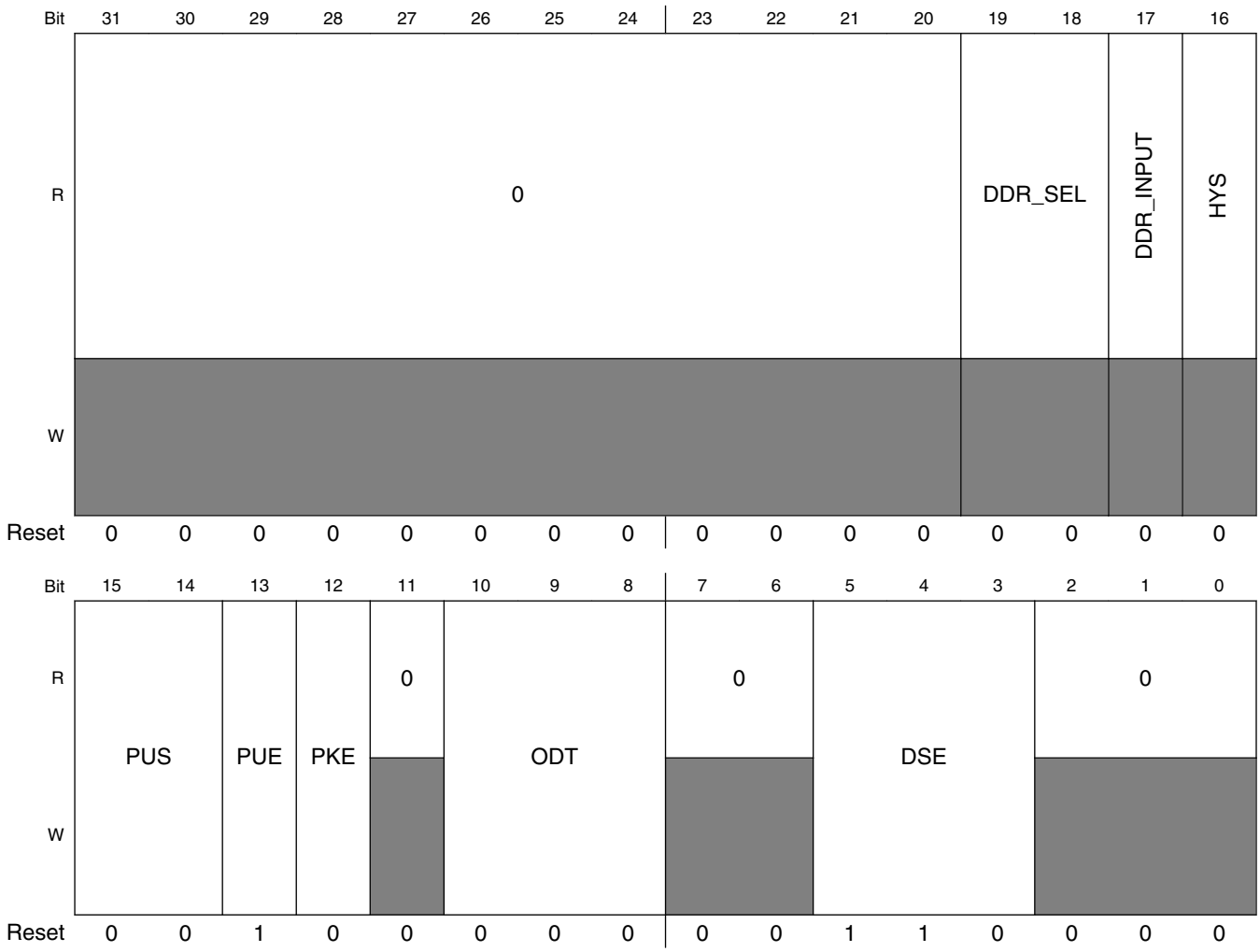
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDWE_B field descriptions (continued)

Field	Description
	0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDWE. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_CTLDS Note: The value of this field does not reflect the value of the Group Control Register.
Reserved	This read-only field is reserved and always has the value 0.

36.4.358 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0_P)

Address: 20E_0000h base + 5A8h offset = 20E_05A8h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS0. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

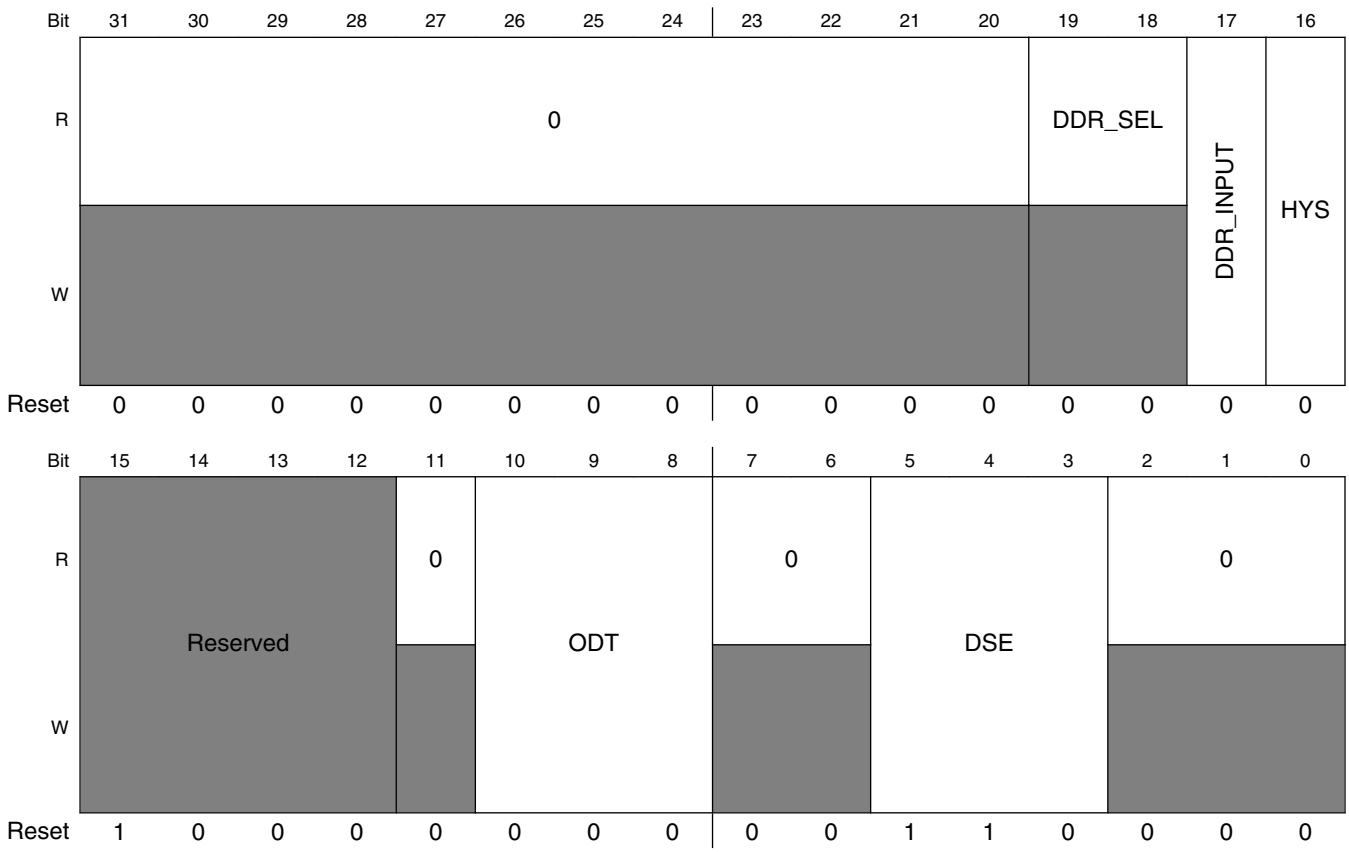
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS0_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.359 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM0)

Address: 20E_0000h base + 5ACh offset = 20E_05ACh



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM0 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE

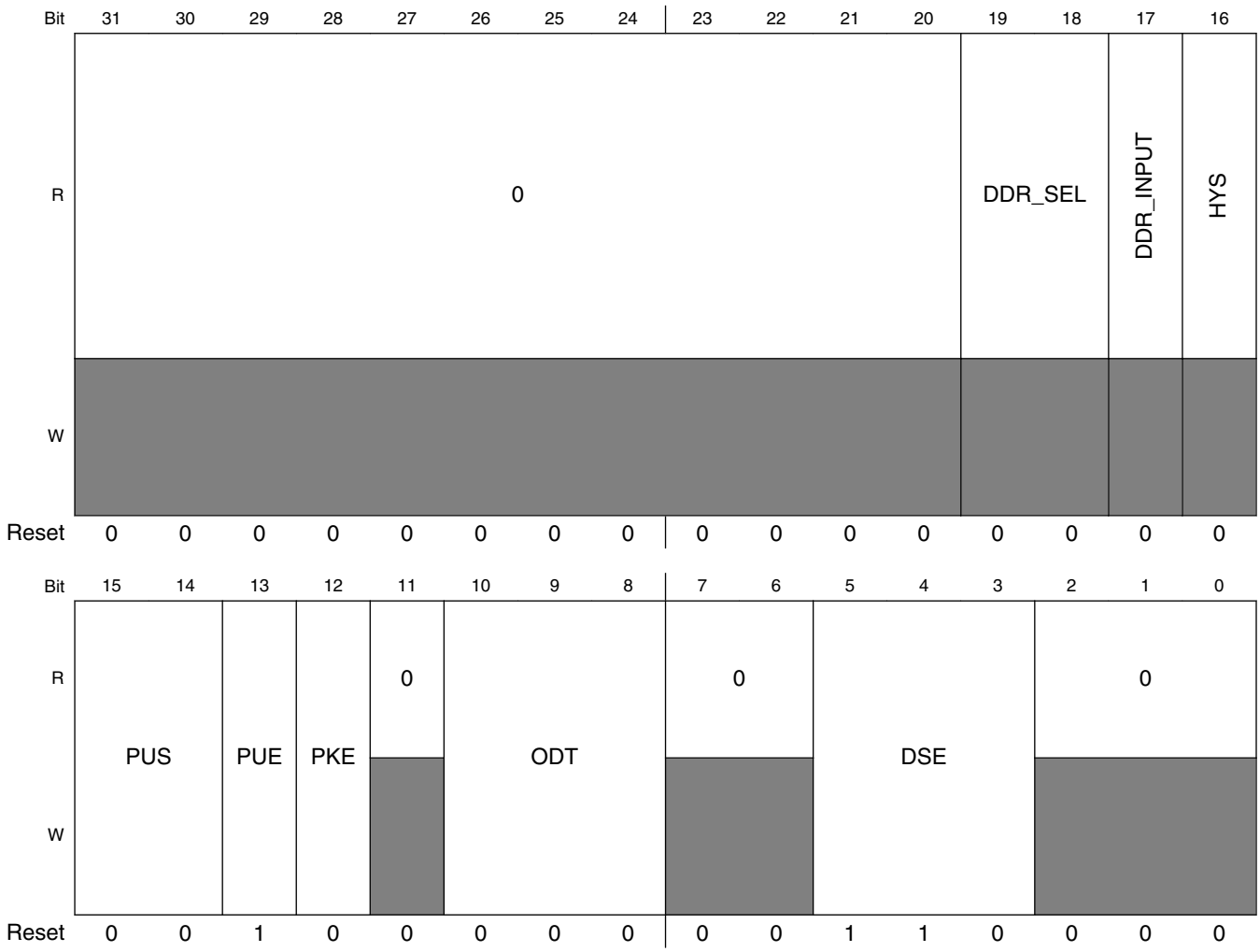
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM0 field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM0. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM0. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM0. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.360 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1_P)

Address: 20E_0000h base + 5B0h offset = 20E_05B0h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS1. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

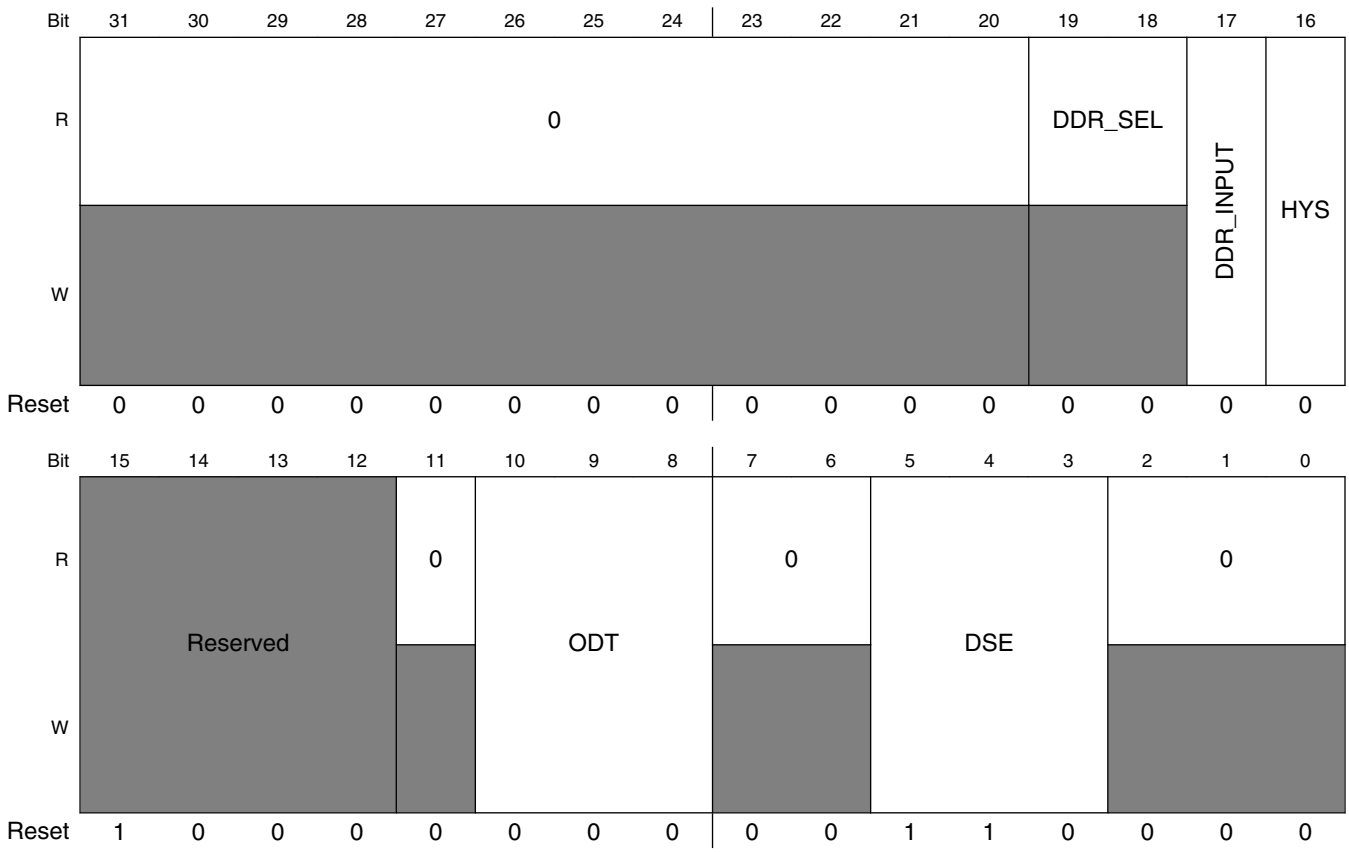
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS1_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.361 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM1)

Address: 20E_0000h base + 5B4h offset = 20E_05B4h



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM1 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE

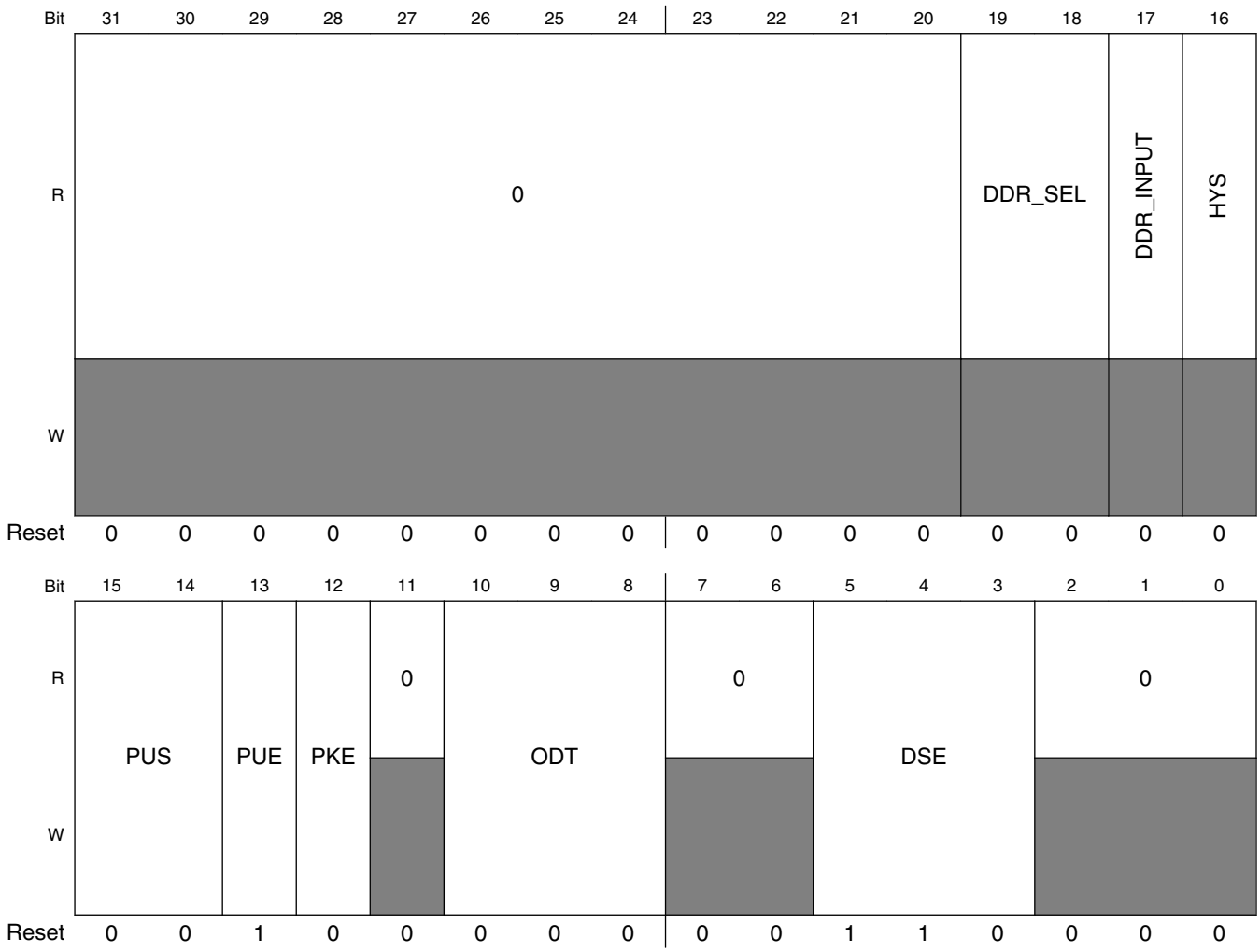
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM1 field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM1. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM1. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM1. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.362 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS6_P)

Address: 20E_0000h base + 5B8h offset = 20E_05B8h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS6_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS6_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS6. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS6. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

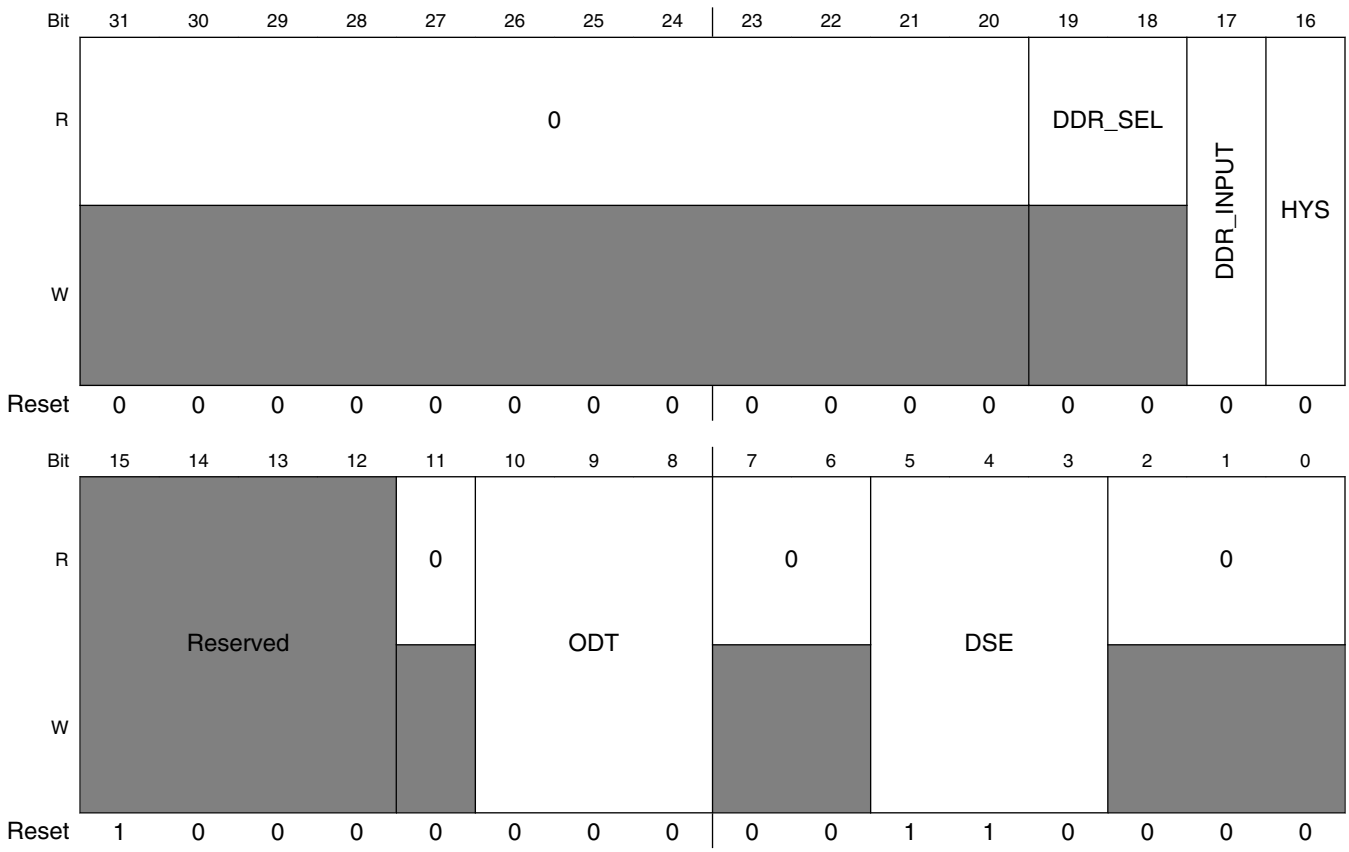
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS6_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.363 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM6)

Address: 20E_0000h base + 5BCh offset = 20E_05BCh



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM6 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE

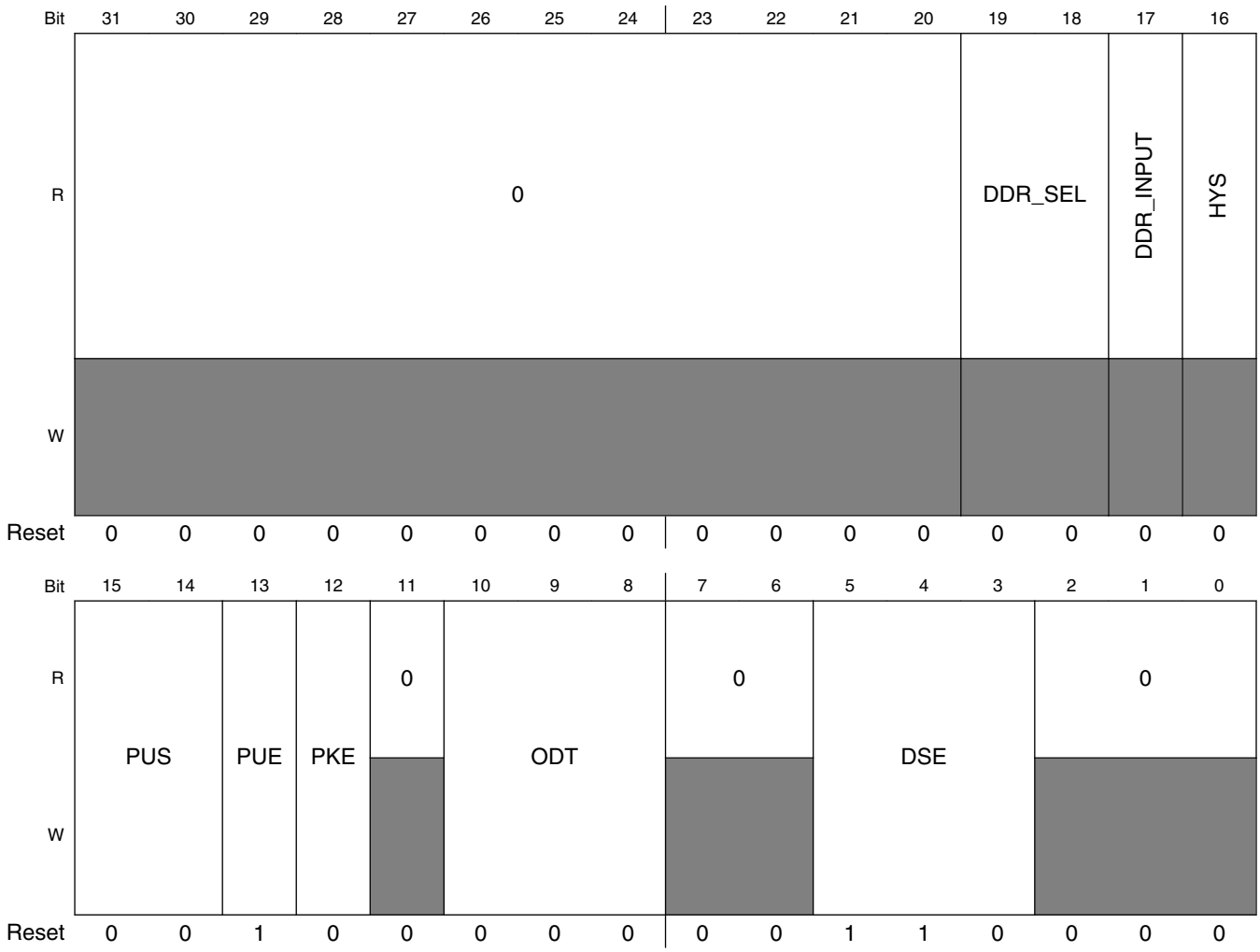
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM6 field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM6. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM6. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM6. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.364 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS7_P)

Address: 20E_0000h base + 5C0h offset = 20E_05C0h



IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS7_P field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS7_P field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
16 HYS	Hysteresis Enable Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDRHYS Note: The value of this field does not reflect the value of the Group Control Register.
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: DRAM_SDQS7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: DRAM_SDQS7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: DRAM_SDQS7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_SDQS7. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_SDQS7. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm

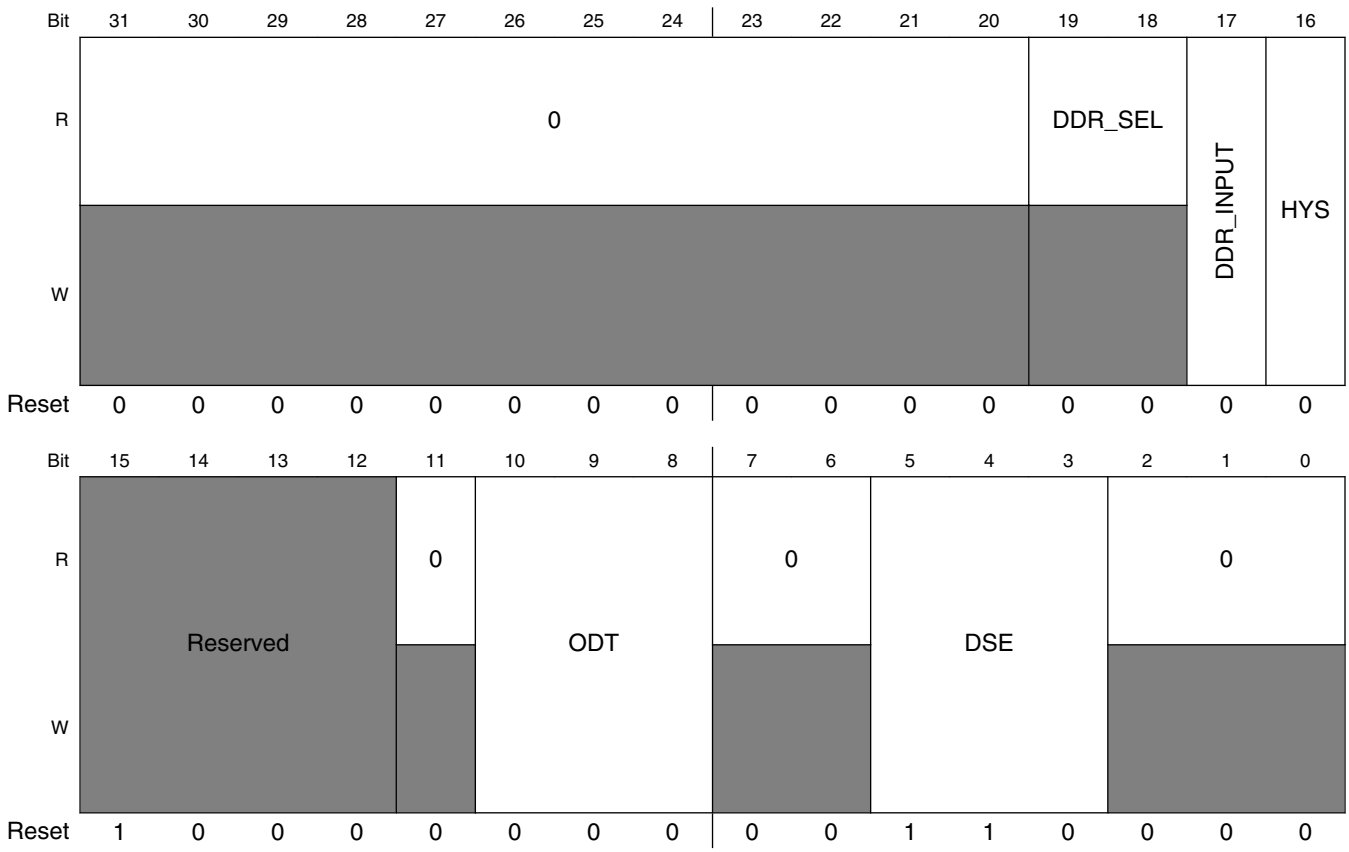
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_SDQS7_P field descriptions (continued)

Field	Description
	101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.365 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM7)

Address: 20E_0000h base + 5C4h offset = 20E_05C4h



IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM7 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field This property can be configured using Group Control Register: IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_DRAM_DQM7 field descriptions (continued)

Field	Description
	Note: The value of this field does not reflect the value of the Group Control Register.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for pad: DRAM_DQM7. 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
16 HYS	Hysteresis Enable Field Select one of next values for pad: DRAM_DQM7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–12 -	This field is reserved. Reserved
11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for pad: DRAM_DQM7. 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
7–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for pad: DRAM_DQM7. 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.366 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL0)

Address: 20E_0000h base + 5C8h offset = 20E_05C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_COL0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_COL0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_COL0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_COL0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_COL0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL0 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_COL0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.367 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW0)

Address: 20E_0000h base + 5CCh offset = 20E_05CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_ROW0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_ROW0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_ROW0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_ROW0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_ROW0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW0 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.368 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL1)

Address: 20E_0000h base + 5D0h offset = 20E_05D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_COL1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_COL1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_COL1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL1 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_COL1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_COL1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_COL1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

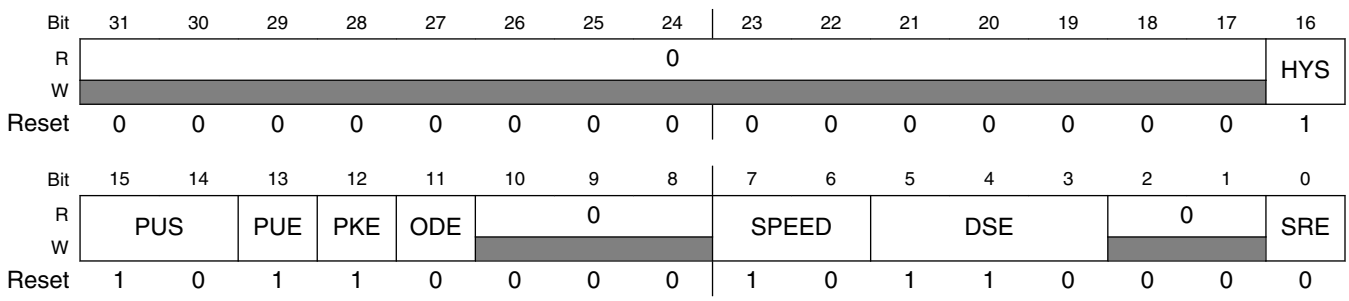
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL1 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.369 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW1)

Address: 20E_0000h base + 5D4h offset = 20E_05D4h



IOMUXC_SW_PAD_CTL_PAD_KEY_ROW1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_ROW1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_ROW1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_ROW1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_ROW1.

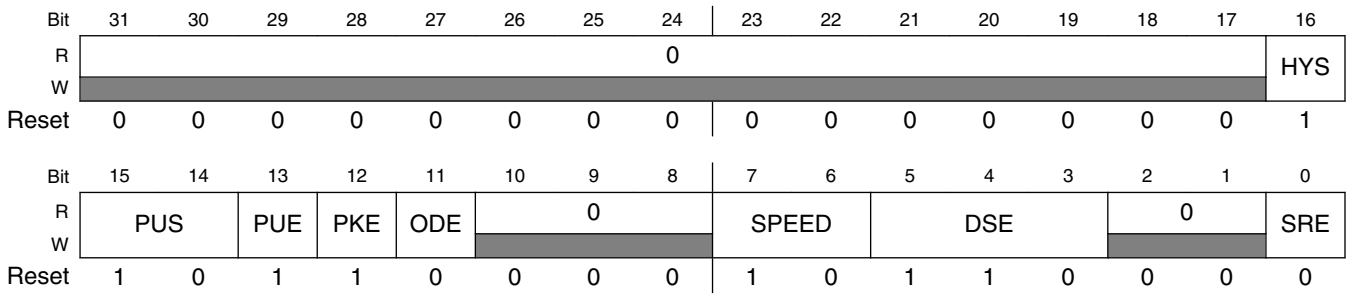
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW1 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_ROW1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.370 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL2)

Address: 20E_0000h base + 5D8h offset = 20E_05D8h



IOMUXC_SW_PAD_CTL_PAD_KEY_COL2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_COL2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_COL2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_COL2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_COL2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL2 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_COL2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.371 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW2)

Address: 20E_0000h base + 5DCh offset = 20E_05DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_ROW2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_ROW2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_ROW2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_ROW2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_ROW2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW2 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.372 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL3)

Address: 20E_0000h base + 5E0h offset = 20E_05E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_COL3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_COL3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_COL3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL3 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_COL3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_COL3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_COL3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

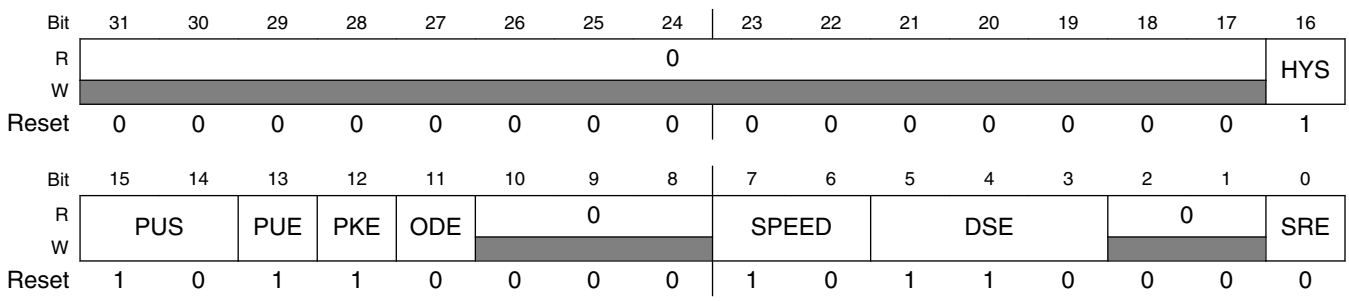
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL3 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.373 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW3)

Address: 20E_0000h base + 5E4h offset = 20E_05E4h



IOMUXC_SW_PAD_CTL_PAD_KEY_ROW3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_ROW3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_ROW3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_ROW3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_ROW3.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW3 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_ROW3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.374 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_COL4)

Address: 20E_0000h base + 5E8h offset = 20E_05E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_COL4 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_COL4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_COL4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_COL4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_COL4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_COL4 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_COL4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.375 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_KEY_ROW4)

Address: 20E_0000h base + 5ECh offset = 20E_05ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW4 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: KEY_ROW4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: KEY_ROW4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: KEY_ROW4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: KEY_ROW4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: KEY_ROW4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

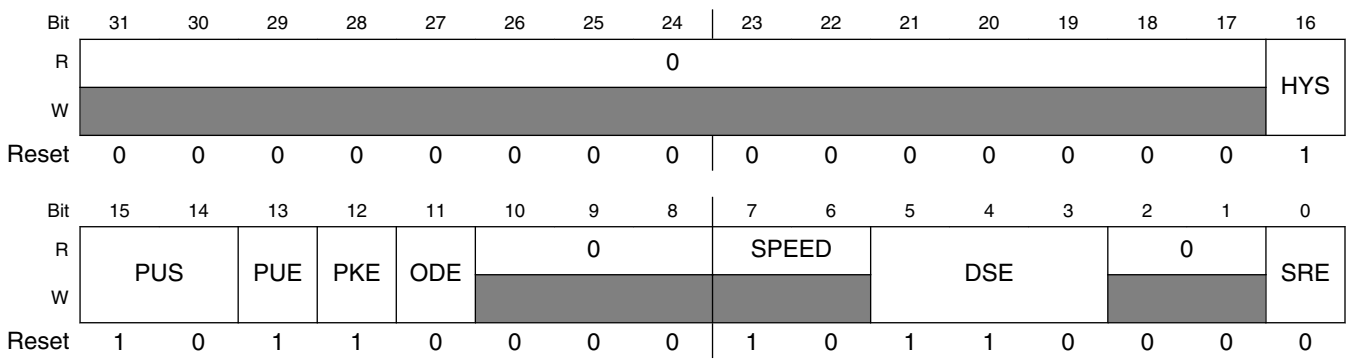
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_KEY_ROW4 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.376 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO00)

Address: 20E_0000h base + 5F0h offset = 20E_05F0h



IOMUXC_SW_PAD_CTL_PAD_GPIO00 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_0. 00 **100K_OHM_PD** — 100K Ohm Pull Down

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO00 field descriptions (continued)

Field	Description
	01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. Read Only Field The value of this field is fixed and cannot be changed. 10 **MEDIUM** — Medium frequency (100, 150 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO00 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.377 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO01)

Address: 20E_0000h base + 5F4h offset = 20E_05F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO01 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_1.

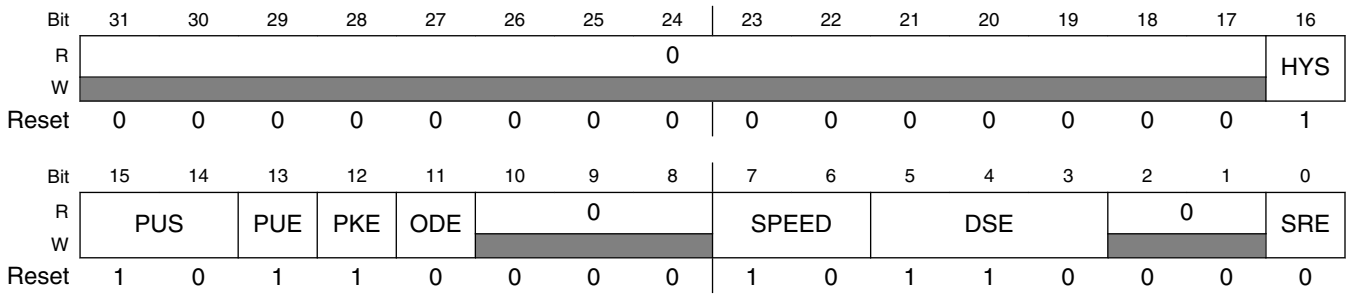
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO01 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.378 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO09)

Address: 20E_0000h base + 5F8h offset = 20E_05F8h



IOMUXC_SW_PAD_CTL_PAD_GPIO09 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_9. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_9. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_9. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_9. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO09 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_9. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.379 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO03)

Address: 20E_0000h base + 5FCh offset = 20E_05FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO03 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO03 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.380 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO06)

Address: 20E_0000h base + 600h offset = 20E_0600h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO06 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO06 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO06 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.381 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO02)

Address: 20E_0000h base + 604h offset = 20E_0604h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO02 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_2.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO02 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.382 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO04)

Address: 20E_0000h base + 608h offset = 20E_0608h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO04 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.383 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO05)

Address: 20E_0000h base + 60Ch offset = 20E_060Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO05 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO05 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.384 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO07)

Address: 20E_0000h base + 610h offset = 20E_0610h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO07 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO07 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO07 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.385 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO08)

Address: 20E_0000h base + 614h offset = 20E_0614h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO08 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_8. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_8. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_8. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_8.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO08 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_8. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.386 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO16)

Address: 20E_0000h base + 618h offset = 20E_0618h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO16 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_16. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_16. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_16. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_16. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO16 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_16. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.387 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO17)

Address: 20E_0000h base + 61Ch offset = 20E_061Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO17 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_17. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_17. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_17. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_17. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. Read Only Field The value of this field is fixed and cannot be changed. 10 **MEDIUM** — Medium frequency (100, 150 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_17. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO17 field descriptions (continued)

Field	Description
	100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.388 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO18)

Address: 20E_0000h base + 620h offset = 20E_0620h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO18 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_18. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_18. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO18 field descriptions (continued)

Field	Description
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_18. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_18. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_18. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.389 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_GPIO19)

Address: 20E_0000h base + 624h offset = 20E_0624h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_GPIO19 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: GPIO_19. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: GPIO_19. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: GPIO_19. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: GPIO_19. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_GPIO19 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: GPIO_19. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.390 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_PIXCLK)

Address: 20E_0000h base + 628h offset = 20E_0628h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_PIXCLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_PIXCLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_PIXCLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_PIXCLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_PIXCLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_PIXCLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_PIXCLK field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.391 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_HSYNC)

Address: 20E_0000h base + 62Ch offset = 20E_062Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_HSYNC field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_MCLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_MCLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_HSYNC field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_MCLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_MCLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_MCLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_HSYNC field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.392 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA_EN)

Address: 20E_0000h base + 630h offset = 20E_0630h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA_EN field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DATA_EN. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DATA_EN. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DATA_EN. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DATA_EN.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA_EN field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DATA_EN. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.393 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_VSYNC)

Address: 20E_0000h base + 634h offset = 20E_0634h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_VSYNC field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_VSYNC. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_VSYNC. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_VSYNC. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_VSYNC. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_VSYNC field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_VSYNC. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.394 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA04)

Address: 20E_0000h base + 638h offset = 20E_0638h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA04 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.395 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA05)

Address: 20E_0000h base + 63Ch offset = 20E_063Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA05 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA05 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

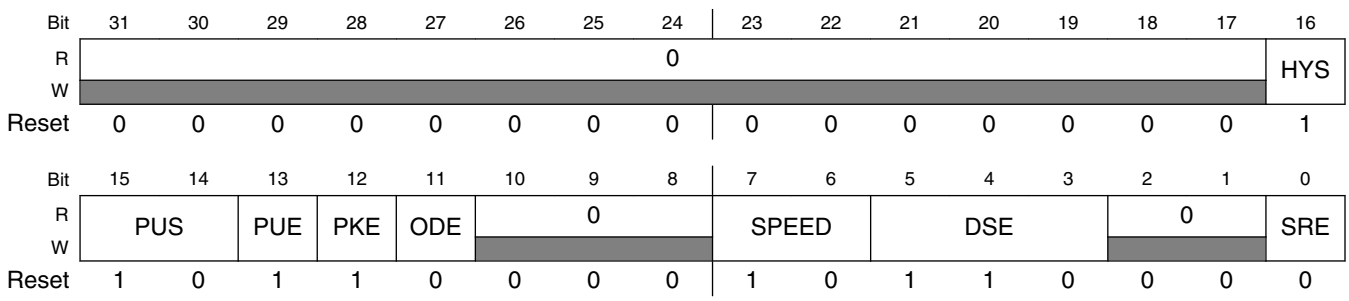
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA05 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.396 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA06)

Address: 20E_0000h base + 640h offset = 20E_0640h



IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA06 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT6.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA06 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.397 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA07)

Address: 20E_0000h base + 644h offset = 20E_0644h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA07 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA07 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.398 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA08)

Address: 20E_0000h base + 648h offset = 20E_0648h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA08 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT8. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT8. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT8. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT8. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT8. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA08 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.399 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA09)

Address: 20E_0000h base + 64Ch offset = 20E_064Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA09 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT9. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT9. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA09 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT9. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT9. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT9. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA09 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.400 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA10)

Address: 20E_0000h base + 650h offset = 20E_0650h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA10 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT10. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT10. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT10. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT10.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA10 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT10. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.401 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA11)

Address: 20E_0000h base + 654h offset = 20E_0654h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA11 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT11. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT11. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT11. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT11. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA11 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT11. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.402 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA12)

Address: 20E_0000h base + 658h offset = 20E_0658h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA12 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT12. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT12. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT12. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT12. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT12. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

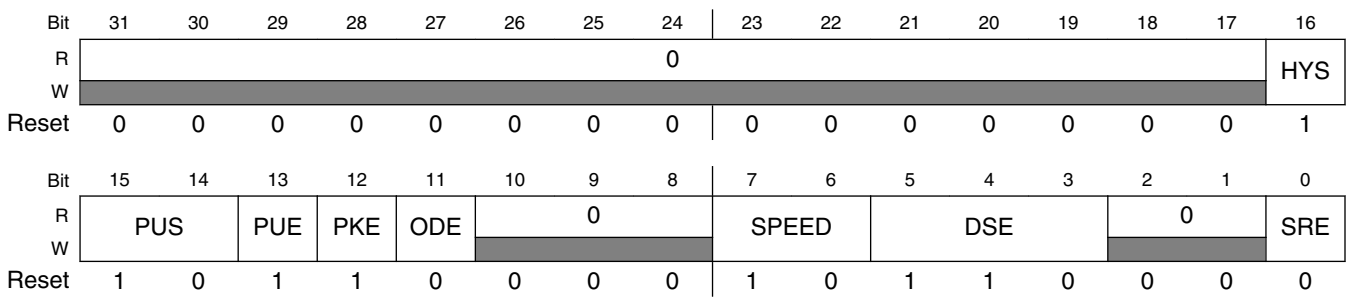
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA12 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.403 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA13)

Address: 20E_0000h base + 65Ch offset = 20E_065Ch



IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA13 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT13. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT13. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA13 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT13. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT13. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT13. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

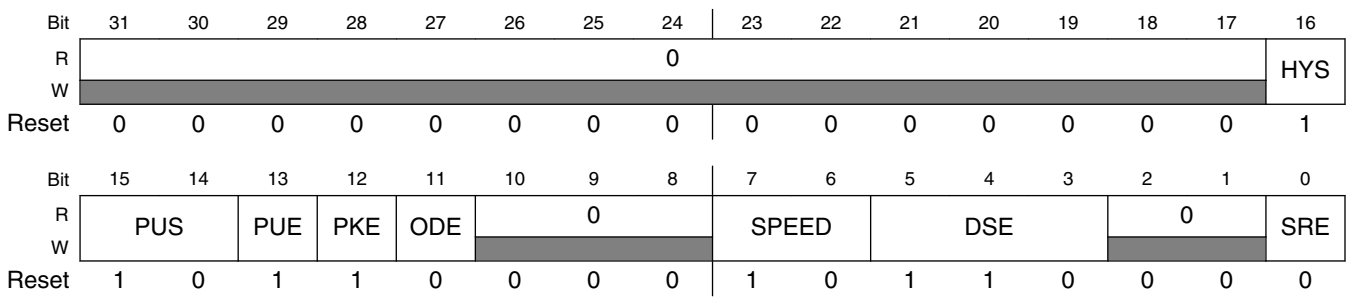
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA13 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.404 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA14)

Address: 20E_0000h base + 660h offset = 20E_0660h



IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA14 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT14. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT14. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT14. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT14.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA14 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT14. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.405 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA15)

Address: 20E_0000h base + 664h offset = 20E_0664h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA15 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT15. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT15. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT15. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT15. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA15 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT15. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.406 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA16)

Address: 20E_0000h base + 668h offset = 20E_0668h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA16 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT16. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT16. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT16. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT16. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT16. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA16 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.407 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA17)

Address: 20E_0000h base + 66Ch offset = 20E_066Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA17 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT17. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT17. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA17 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT17. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT17. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT17. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA17 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.408 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA18)

Address: 20E_0000h base + 670h offset = 20E_0670h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA18 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT18. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT18. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT18. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT18.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA18 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT18. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.409 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA19)

Address: 20E_0000h base + 674h offset = 20E_0674h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA19 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: CSI0_DAT19. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: CSI0_DAT19. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: CSI0_DAT19. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: CSI0_DAT19. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_CSI0_DATA19 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: CSI0_DAT19. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.410 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TMS)

Address: 20E_0000h base + 678h offset = 20E_0678h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_JTAG_TMS field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: JTAG_TMS. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: JTAG_TMS. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field The value of this field is fixed and cannot be changed. 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_TMS. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field The value of this field is fixed and cannot be changed. 0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 01 **50MHZ** — Low (50 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 100 **60_OHM** — 60 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.

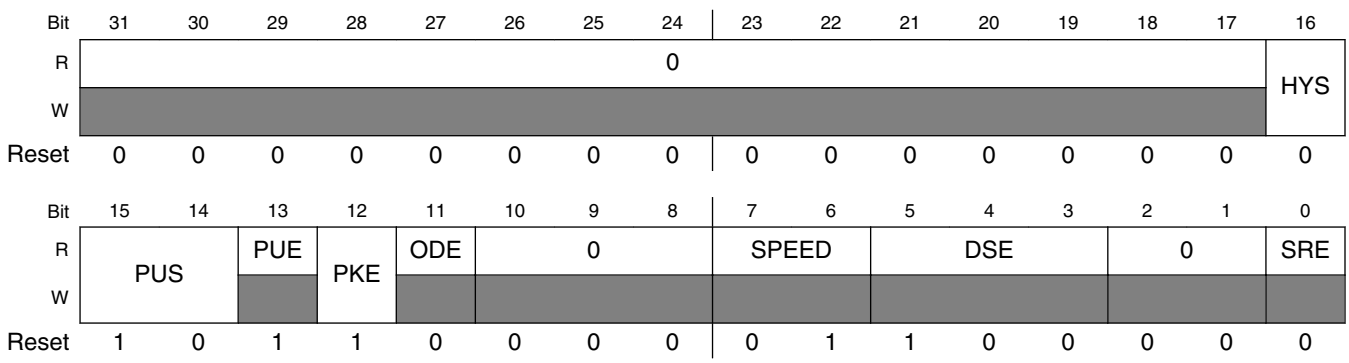
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_TMS field descriptions (continued)

Field	Description
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 0 **SLOW** — Slow Slew Rate

36.4.411 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_MOD)

Address: 20E_0000h base + 67Ch offset = 20E_067Ch



IOMUXC_SW_PAD_CTL_PAD_JTAG_MOD field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: JTAG_MOD. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: JTAG_MOD. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_MOD field descriptions (continued)

Field	Description
	The value of this field is fixed and cannot be changed. 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_MOD. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field The value of this field is fixed and cannot be changed. 0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 01 **50MHZ** — Low (50 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 100 **60_OHM** — 60 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 0 **SLOW** — Slow Slew Rate

36.4.412 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TRSTB)

Address: 20E_0000h base + 680h offset = 20E_0680h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_JTAG_TRSTB field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: JTAG_TRSTB. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: JTAG_TRSTB. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field The value of this field is fixed and cannot be changed. 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_TRSTB. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_TRSTB field descriptions (continued)

Field	Description
	The value of this field is fixed and cannot be changed. 0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 01 **50MHZ** — Low (50 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 100 **60_OHM** — 60 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 0 **SLOW** — Slow Slew Rate

36.4.413 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TDI)

Address: 20E_0000h base + 684h offset = 20E_0684h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_JTAG_TDI field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: JTAG_TDI. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: JTAG_TDI. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field The value of this field is fixed and cannot be changed. 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_TDI. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field The value of this field is fixed and cannot be changed. 0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 01 **50MHZ** — Low (50 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 100 **60_OHM** — 60 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.

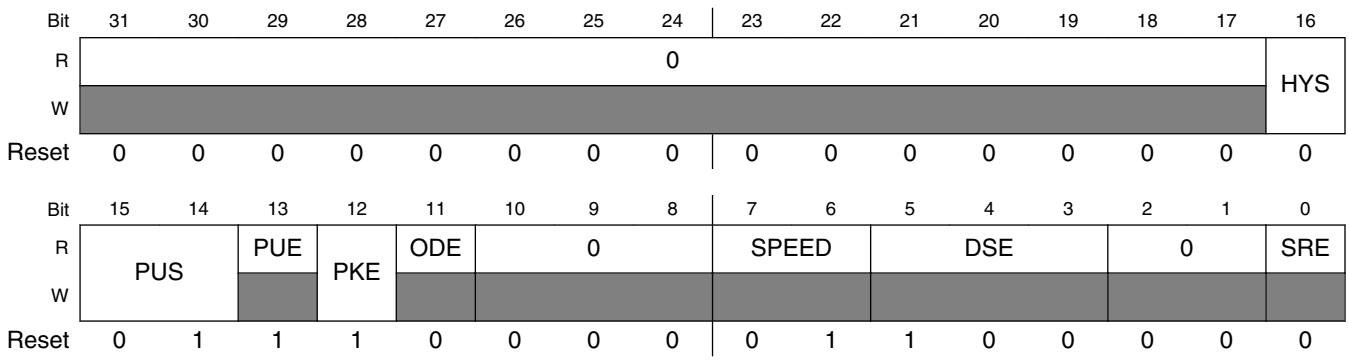
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_TDI field descriptions (continued)

Field	Description
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 0 **SLOW** — Slow Slew Rate

36.4.414 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TCK)

Address: 20E_0000h base + 688h offset = 20E_0688h



IOMUXC_SW_PAD_CTL_PAD_JTAG_TCK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: JTAG_TCK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: JTAG_TCK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_TCK field descriptions (continued)

Field	Description
	The value of this field is fixed and cannot be changed. 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_TCK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field The value of this field is fixed and cannot be changed. 0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 01 **50MHZ** — Low (50 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 100 **60_OHM** — 60 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 0 **SLOW** — Slow Slew Rate

36.4.415 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_JTAG_TDO)

Address: 20E_0000h base + 68Ch offset = 20E_068Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	0	1	0	0	0	0	1	0	1	1	0	0	0	1

IOMUXC_SW_PAD_CTL_PAD_JTAG_TDO field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Read Only Field The value of this field is fixed and cannot be changed. 0 **DISABLED** — CMOS input
15–14 PUS	Pull Up / Down Config. Field Read Only Field The value of this field is fixed and cannot be changed. 10 **100K_OHM_PU** — 100K Ohm Pull Up
13 PUE	Pull / Keep Select Field Read Only Field The value of this field is fixed and cannot be changed. 0 **KEEP** — Keeper Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: JTAG_TDO. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. Read Only Field The value of this field is fixed and cannot be changed.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_JTAG_TDO field descriptions (continued)

Field	Description
	0 **DISABLED** — Output is CMOS.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field Read Only Field The value of this field is fixed and cannot be changed. 10 **100MHZ** — Medium (100 MHz)
5–3 DSE	Drive Strength Field Read Only Field The value of this field is fixed and cannot be changed. 110 **40_OHM** — 40 Ohm
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. Read Only Field The value of this field is fixed and cannot be changed. 1 **FAST** — Fast Slew Rate

36.4.416 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA7)

Address: 20E_0000h base + 690h offset = 20E_0690h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA7 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA7 field descriptions (continued)

Field	Description
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA7 field descriptions (continued)

Field	Description
	101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.417 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA6)

Address: 20E_0000h base + 694h offset = 20E_0694h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA6 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA6 field descriptions (continued)

Field	Description
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.418 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA5)

Address: 20E_0000h base + 698h offset = 20E_0698h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA5 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA5 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.419 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA4)

Address: 20E_0000h base + 69Ch offset = 20E_069Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA4 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT4. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA4 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.420 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_CMD)

Address: 20E_0000h base + 6A0h offset = 20E_06A0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_CMD field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_CMD. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_CMD. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_CMD field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_CMD. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_CMD. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_CMD. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

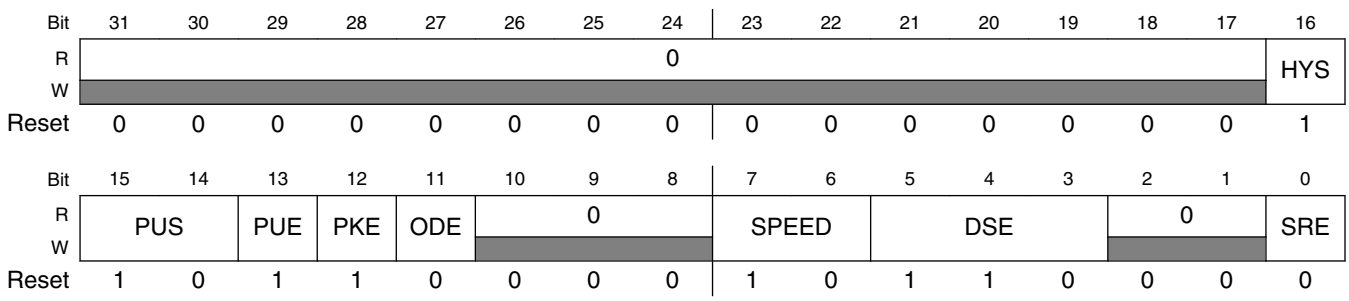
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_CMD field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.421 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_CLK)

Address: 20E_0000h base + 6A4h offset = 20E_06A4h



IOMUXC_SW_PAD_CTL_PAD_SD3_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_CLK.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_CLK field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.422 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA0)

Address: 20E_0000h base + 6A8h offset = 20E_06A8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA0 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.423 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA1)

Address: 20E_0000h base + 6ACh offset = 20E_06ACh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA1 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.424 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA2)

Address: 20E_0000h base + 6B0h offset = 20E_06B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA2 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA2 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.425 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_DATA3)

Address: 20E_0000h base + 6B4h offset = 20E_06B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_DAT3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_DAT3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_DAT3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_DAT3.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_DATA3 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_DAT3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.426 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD3_RESET)

Address: 20E_0000h base + 6B8h offset = 20E_06B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD3_RESET field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD3_RST. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD3_RST. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD3_RST. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD3_RST. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD3_RESET field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD3_RST. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.427 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CLE)

Address: 20E_0000h base + 6BCh offset = 20E_06BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_CLE field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_CLE. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_CLE. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_CLE. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_CLE. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_CLE. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CLE field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.428 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_ALE)

Address: 20E_0000h base + 6C0h offset = 20E_06C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_ALE field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_ALE. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_ALE. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_ALE field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_ALE. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_ALE. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_ALE. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

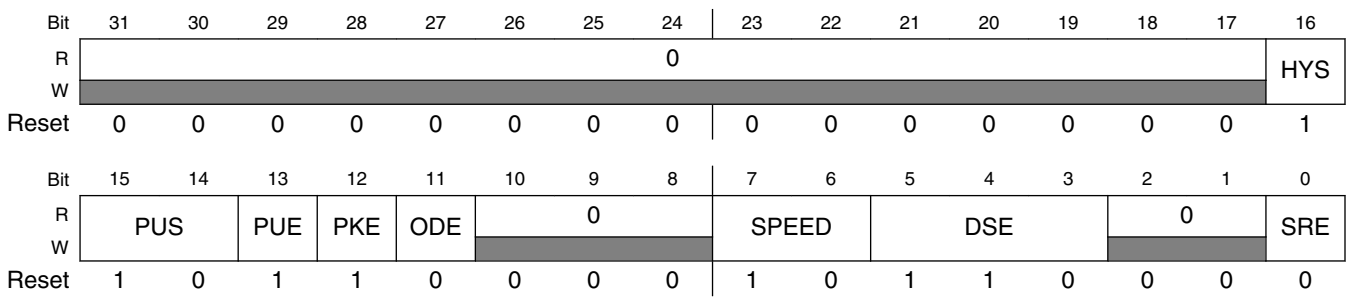
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_ALE field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.429 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_WP_B)

Address: 20E_0000h base + 6C4h offset = 20E_06C4h



IOMUXC_SW_PAD_CTL_PAD_NAND_WP_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_WP_B. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_WP_B. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_WP_B. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_WP_B.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_WP_B field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_WP_B. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.430 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_READY_B)

Address: 20E_0000h base + 6C8h offset = 20E_06C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_READY_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_RB0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_RB0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_RB0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_RB0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_READY_B field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_RB0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.431 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS0_B)

Address: 20E_0000h base + 6CCh offset = 20E_06CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_CS0_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_CS0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_CS0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_CS0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_CS0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_CS0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CS0_B field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.432 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS1_B)

Address: 20E_0000h base + 6D0h offset = 20E_06D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_CS1_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_CS1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_CS1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CS1_B field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_CS1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_CS1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_CS1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CS1_B field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.433 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS2_B)

Address: 20E_0000h base + 6D4h offset = 20E_06D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_CS2_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_CS2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_CS2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_CS2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_CS2.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CS2_B field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_CS2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.434 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_CS3_B)

Address: 20E_0000h base + 6D8h offset = 20E_06D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_CS3_B field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_CS3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_CS3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_CS3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_CS3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_CS3_B field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_CS3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.435 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_CMD)

Address: 20E_0000h base + 6DCh offset = 20E_06DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_CMD field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_CMD. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_CMD. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_CMD. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_CMD. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_CMD. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_CMD field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.436 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_CLK)

Address: 20E_0000h base + 6E0h offset = 20E_06E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_CLK field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_CLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

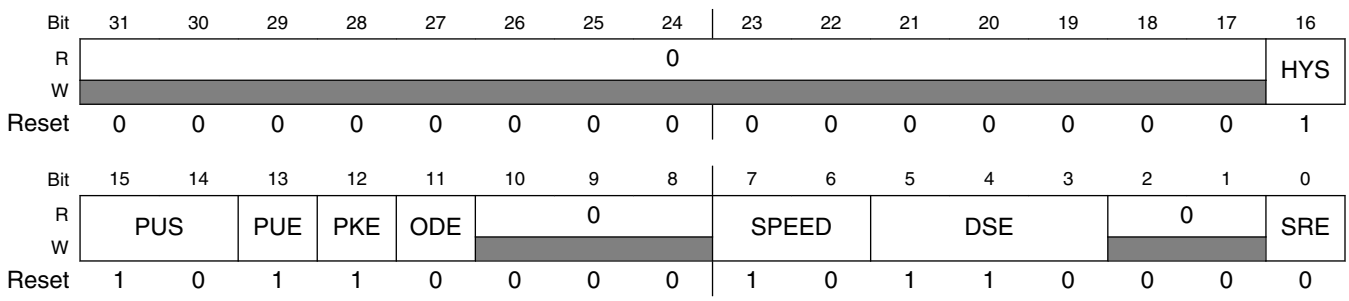
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_CLK field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.437 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA00)

Address: 20E_0000h base + 6E4h offset = 20E_06E4h



IOMUXC_SW_PAD_CTL_PAD_NAND_DATA00 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA00 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.438 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA01)

Address: 20E_0000h base + 6E8h offset = 20E_06E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA01 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA01 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.439 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA02)

Address: 20E_0000h base + 6ECh offset = 20E_06ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA02 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA02 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.440 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA03)

Address: 20E_0000h base + 6F0h offset = 20E_06F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA03 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA03 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA03 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.441 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA04)

Address: 20E_0000h base + 6F4h offset = 20E_06F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA04 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D4.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA04 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.442 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA05)

Address: 20E_0000h base + 6F8h offset = 20E_06F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA05 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA05 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.443 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA06)

Address: 20E_0000h base + 6FCh offset = 20E_06FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA06 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA06 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.444 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_NAND_DATA07)

Address: 20E_0000h base + 700h offset = 20E_0700h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA07 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: NANDF_D7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: NANDF_D7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA07 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: NANDF_D7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: NANDF_D7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: NANDF_D7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_NAND_DATA07 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.445 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA0)

Address: 20E_0000h base + 704h offset = 20E_0704h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA0 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.446 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA1)

Address: 20E_0000h base + 708h offset = 20E_0708h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT1. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA1 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.447 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA2)

Address: 20E_0000h base + 70Ch offset = 20E_070Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT2. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA2 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.448 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA3)

Address: 20E_0000h base + 710h offset = 20E_0710h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA3 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA3 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.449 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA4)

Address: 20E_0000h base + 714h offset = 20E_0714h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA4 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT4. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT4. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT4. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT4.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA4 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT4. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.450 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA5)

Address: 20E_0000h base + 718h offset = 20E_0718h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA5 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT5. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT5. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT5. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT5. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA5 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT5. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.451 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA6)

Address: 20E_0000h base + 71Ch offset = 20E_071Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA6 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT6. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT6. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT6. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT6. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT6. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA6 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.452 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD4_DATA7)

Address: 20E_0000h base + 720h offset = 20E_0720h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA7 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD4_DAT7. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD4_DAT7. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA7 field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD4_DAT7. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD4_DAT7. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD4_DAT7. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD4_DATA7 field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.453 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA1)

Address: 20E_0000h base + 724h offset = 20E_0724h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA1 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_DAT1. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_DAT1. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_DAT1. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_DAT1.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA1 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_DAT1. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.454 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA0)

Address: 20E_0000h base + 728h offset = 20E_0728h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA0 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_DAT0. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_DAT0. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_DAT0. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_DAT0. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA0 field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_DAT0. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.455 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA3)

Address: 20E_0000h base + 72Ch offset = 20E_072Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_DAT3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_DAT3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_DAT3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_DAT3. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_DAT3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA3 field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.456 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_CMD)

Address: 20E_0000h base + 730h offset = 20E_0730h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_CMD field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_CMD. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_CMD. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_CMD field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_CMD. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_CMD. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_CMD. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_CMD field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.457 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_DATA2)

Address: 20E_0000h base + 734h offset = 20E_0734h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA2 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_DAT2. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_DAT2. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_DAT2. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_DAT2.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_DATA2 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_DAT2. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.458 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD1_CLK)

Address: 20E_0000h base + 738h offset = 20E_0738h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD1_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD1_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD1_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD1_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD1_CLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD1_CLK field descriptions (continued)

Field	Description
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD1_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.459 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_CLK)

Address: 20E_0000h base + 73Ch offset = 20E_073Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD2_CLK field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_CLK. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_CLK. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_CLK. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_CLK. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_CLK. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_CLK field descriptions (continued)

Field	Description
	011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.460 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_CMD)

Address: 20E_0000h base + 740h offset = 20E_0740h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	PUS		PUE	PKE	ODE	0			SPEED		DSE			0		SRE
W																
Reset	1	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_PAD_SD2_CMD field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_CMD. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_CMD. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up

Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_CMD field descriptions (continued)

Field	Description
	10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_CMD. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_CMD. 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_CMD. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details.

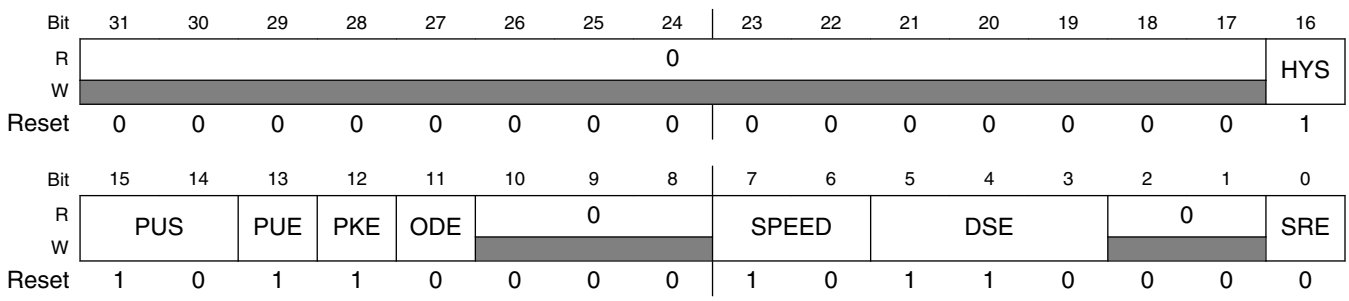
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_CMD field descriptions (continued)

Field	Description
	0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.461 Pad Control Register (IOMUXC_SW_PAD_CTL_PAD_SD2_DATA3)

Address: 20E_0000h base + 744h offset = 20E_0744h



IOMUXC_SW_PAD_CTL_PAD_SD2_DATA3 field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for pad: SD2_DAT3. 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
15–14 PUS	Pull Up / Down Config. Field Select one of next values for pad: SD2_DAT3. 00 **100K_OHM_PD** — 100K Ohm Pull Down 01 **47K_OHM_PU** — 47K Ohm Pull Up 10 **100K_OHM_PU** — 100K Ohm Pull Up 11 **22K_OHM_PU** — 22K Ohm Pull Up
13 PUE	Pull / Keep Select Field Select one of next values for pad: SD2_DAT3. 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
12 PKE	Pull / Keep Enable Field Select one of next values for pad: SD2_DAT3.

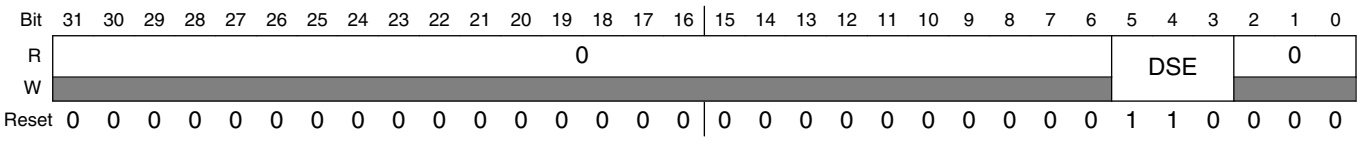
Table continues on the next page...

IOMUXC_SW_PAD_CTL_PAD_SD2_DATA3 field descriptions (continued)

Field	Description
	0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
11 ODE	Open Drain Enable Field Enables open drain of the pin. 0 **DISABLED** — Output is CMOS. 1 **ENABLED** — Output is Open Drain.
10–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 SPEED	Speed Field The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 00 **LOW** — Low frequency (50 MHz) 01 **MEDIUM** — Medium frequency (100, 150 MHz) 10 **MEDIUM** — Medium frequency (100, 150 MHz) 11 **MAXIMUM** — Maximum frequency (100, 150, 200 MHz)
5–3 DSE	Drive Strength Field Select one of next values for pad: SD2_DAT3. 000 **HIZ** — HI-Z 001 **260_OHM** — 150 Ohm @ 3.3V, 260 Ohm @ 1.8V 010 **130_OHM** — 75 Ohm @ 3.3V, 130 Ohm @ 1.8V 011 **90_OHM** — 50 Ohm @ 3.3V, 90 Ohm @ 1.8V 100 **60_OHM** — 37 Ohm @ 3.3V, 60 Ohm @ 1.8V 101 **50_OHM** — 30 Ohm @ 3.3V, 50 Ohm @ 1.8V 110 **40_OHM** — 25 Ohm @ 3.3V, 40 Ohm @ 1.8V 111 **33_OHM** — 20 Ohm @ 3.3V, 33 Ohm @ 1.8V
2–1 Reserved	This read-only field is reserved and always has the value 0.
0 SRE	Slew Rate Field Slew rate control. The operational frequency on GPIO pads is dependent on slew rate (SRE), speed (SPEED), and supply voltage (OVDD). Please see for more details. 0 **SLOW** — Slow Slew Rate 1 **FAST** — Fast Slew Rate

36.4.462 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B7DS)

Address: 20E_0000h base + 748h offset = 20E_0748h

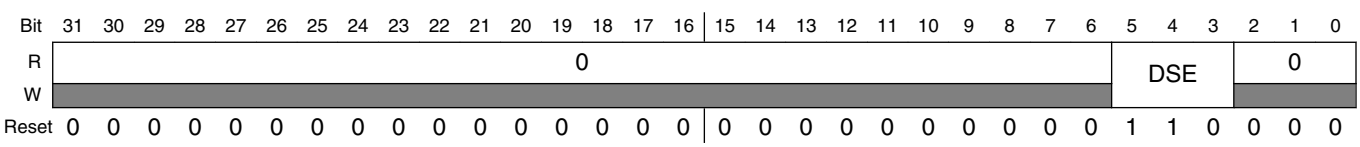


IOMUXC_SW_PAD_CTL_GRP_B7DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.463 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_ADDDS)

Address: 20E_0000h base + 74Ch offset = 20E_074Ch



IOMUXC_SW_PAD_CTL_GRP_ADDDS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.

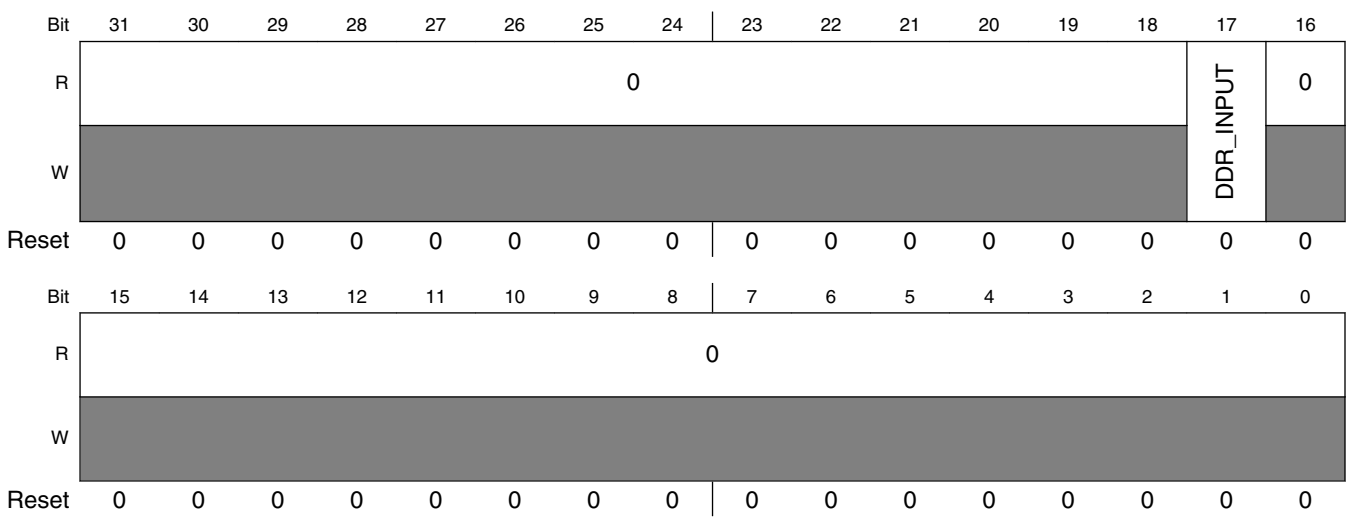
Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_ADDDS field descriptions (continued)

Field	Description
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_ADDR00, DRAM_ADDR01, DRAM_ADDR02, DRAM_ADDR03, DRAM_ADDR04, DRAM_ADDR05, DRAM_ADDR06, DRAM_ADDR07, DRAM_ADDR08, DRAM_ADDR09, DRAM_ADDR10, DRAM_ADDR11, DRAM_ADDR12, DRAM_ADDR13, DRAM_ADDR14, DRAM_ADDR15, DRAM_SDBA0, DRAM_SDBA1 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.464 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL)

Address: 20E_0000h base + 750h offset = 20E_0750h



IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17 DDR_INPUT	DDR / CMOS Input Mode Field

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_DDRMODE_CTL field descriptions (continued)

Field	Description
	Select one of next values for group: . Affected pads: DRAM_SDQS0_P, DRAM_SDQS1_P, DRAM_SDQS2_P, DRAM_SDQS3_P, DRAM_SDQS4_P, DRAM_SDQS5_P, DRAM_SDQS6_P, DRAM_SDQS7_P 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
Reserved	This read-only field is reserved and always has the value 0.

36.4.465 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL0)

Address: 20E_0000h base + 754h offset = 20E_0754h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																					ODT			0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_TERM_CTL0 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.466 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRPKE)

Address: 20E_0000h base + 758h offset = 20E_0758h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			PKE	0											
W																
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_DDRPKE field descriptions

Field	Description
31–13 Reserved	This read-only field is reserved and always has the value 0.
12 PKE	Pull / Keep Enable Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07, DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15, DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23, DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31, DRAM_DATA32, DRAM_DATA33, DRAM_DATA34, DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39, DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47, DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55, DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63 0 **DISABLED** — Pull/Keeper Disabled 1 **ENABLED** — Pull/Keeper Enabled
Reserved	This read-only field is reserved and always has the value 0.

36.4.467 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL1)

Address: 20E_0000h base + 75Ch offset = 20E_075Ch

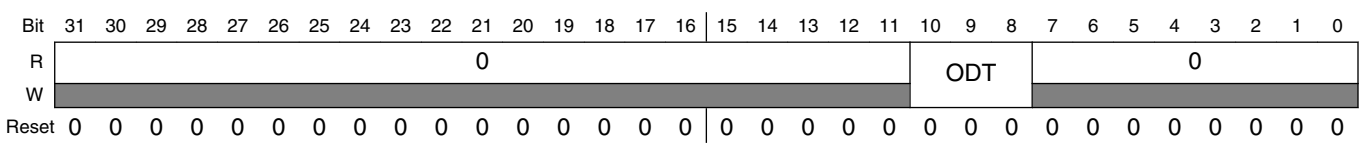
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																					ODT			0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_TERM_CTL1 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.468 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL2)

Address: 20E_0000h base + 760h offset = 20E_0760h



IOMUXC_SW_PAD_CTL_GRP_TERM_CTL2 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT

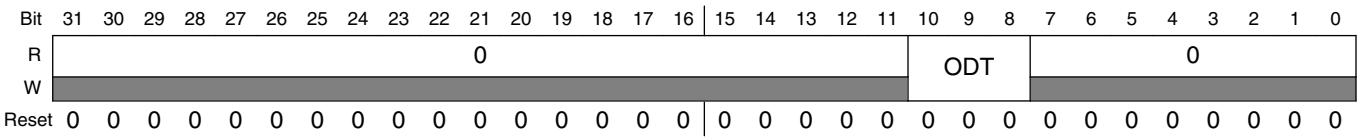
Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_TERM_CTL2 field descriptions (continued)

Field	Description
	110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.469 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL3)

Address: 20E_0000h base + 764h offset = 20E_0764h

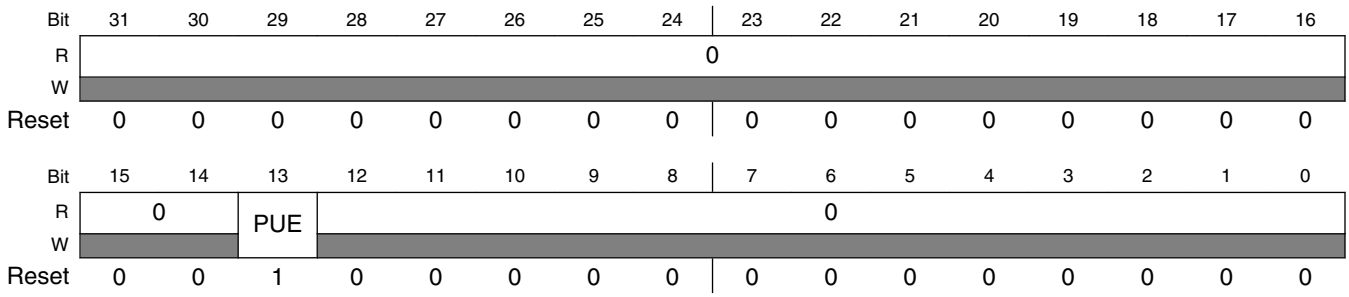


IOMUXC_SW_PAD_CTL_GRP_TERM_CTL3 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.470 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRPK)

Address: 20E_0000h base + 768h offset = 20E_0768h

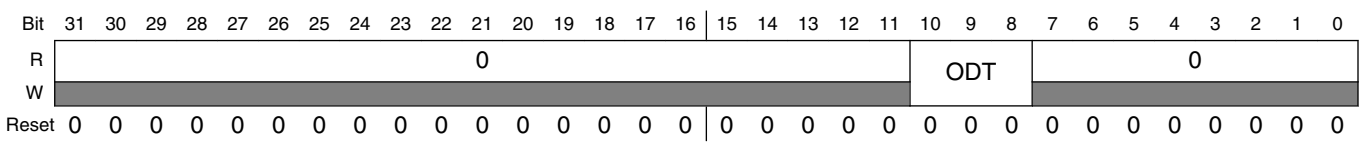


IOMUXC_SW_PAD_CTL_GRP_DDRPK field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13 PUE	Pull / Keep Select Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07, DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15, DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23, DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31, DRAM_DATA32, DRAM_DATA33, DRAM_DATA34, DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39, DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47, DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55, DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63 0 **KEEP** — Keeper Enabled 1 **PULL** — Pull Enabled
Reserved	This read-only field is reserved and always has the value 0.

36.4.471 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL4)

Address: 20E_0000h base + 76Ch offset = 20E_076Ch



IOMUXC_SW_PAD_CTL_GRP_TERM_CTL4 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA32, DRAM_DATA33, DRAM_DATA34, DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.472 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRHYS)

Address: 20E_0000h base + 770h offset = 20E_0770h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															HYS
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_DDRHYS field descriptions

Field	Description
31–17 Reserved	This read-only field is reserved and always has the value 0.
16 HYS	Hysteresis Enable Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07, DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15, DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23, DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31, DRAM_DATA32, DRAM_DATA33, DRAM_DATA34,

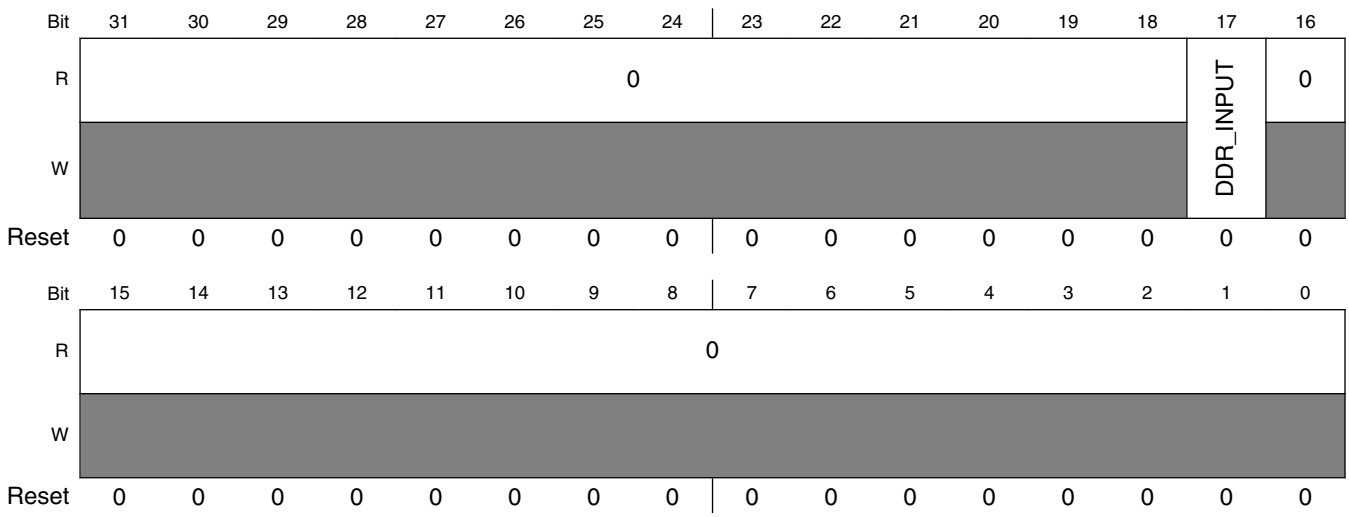
Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_DDRHYS field descriptions (continued)

Field	Description
	DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39, DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47, DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55, DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63, DRAM_SDQS0_P, DRAM_SDQS1_P, DRAM_SDQS2_P, DRAM_SDQS3_P, DRAM_SDQS4_P, DRAM_SDQS5_P, DRAM_SDQS6_P, DRAM_SDQS7_P 0 **DISABLED** — CMOS input 1 **ENABLED** — Schmitt trigger input
Reserved	This read-only field is reserved and always has the value 0.

36.4.473 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDRMODE)

Address: 20E_0000h base + 774h offset = 20E_0774h



IOMUXC_SW_PAD_CTL_GRP_DDRMODE field descriptions

Field	Description
31–18 Reserved	This read-only field is reserved and always has the value 0.
17 DDR_INPUT	DDR / CMOS Input Mode Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07, DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15, DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23, DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31, DRAM_DATA32, DRAM_DATA33, DRAM_DATA34,

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_DDRMODE field descriptions (continued)

Field	Description
	DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39, DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47, DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55, DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63 0 **CMOS** — CMOS input mode. 1 **DIFFERENTIAL** — Differential input mode.
Reserved	This read-only field is reserved and always has the value 0.

36.4.474 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL5)

Address: 20E_0000h base + 778h offset = 20E_0778h

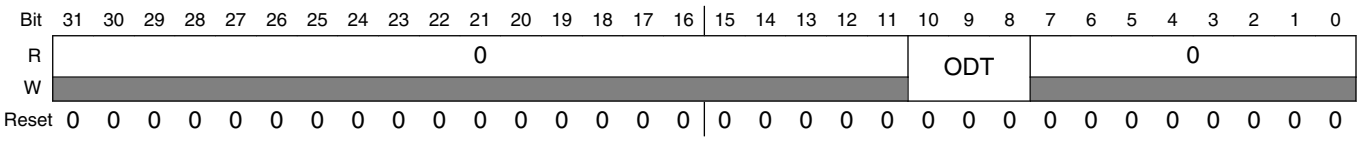
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																					ODT			0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_TERM_CTL5 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.475 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL6)

Address: 20E_0000h base + 77Ch offset = 20E_077Ch

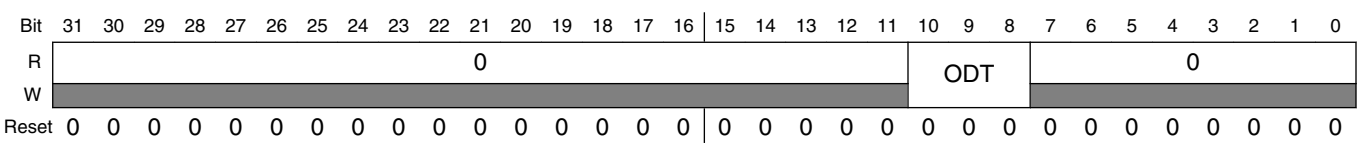


IOMUXC_SW_PAD_CTL_GRP_TERM_CTL6 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.476 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_TERM_CTL7)

Address: 20E_0000h base + 780h offset = 20E_0780h



IOMUXC_SW_PAD_CTL_GRP_TERM_CTL7 field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_TERM_CTL7 field descriptions (continued)

Field	Description
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.477 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B0DS)

Address: 20E_0000h base + 784h offset = 20E_0784h

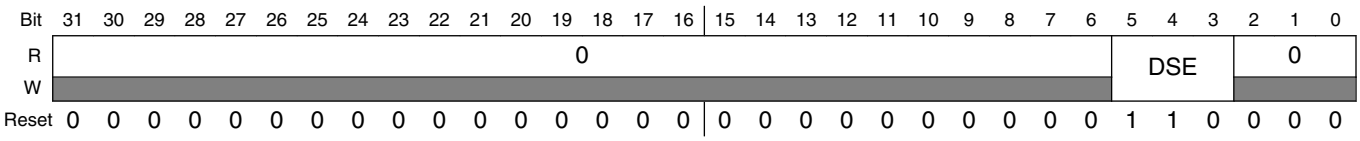
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																										DSE			0		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_B0DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.478 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B1DS)

Address: 20E_0000h base + 788h offset = 20E_0788h

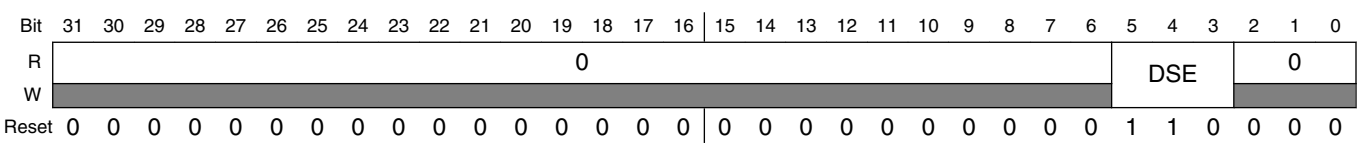


IOMUXC_SW_PAD_CTL_GRP_B1DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.479 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_CTLDS)

Address: 20E_0000h base + 78Ch offset = 20E_078Ch



IOMUXC_SW_PAD_CTL_GRP_CTLDS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_CTLDS field descriptions (continued)

Field	Description
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_CS0_B, DRAM_CS1_B, DRAM_SDBA2, DRAM_SDCKE0, DRAM_SDCKE1, DRAM_SDWE_B 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.480 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII)

Address: 20E_0000h base + 790h offset = 20E_0790h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0												DDR_SEL		0	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field Select one of next values for group: . Affected pads: RGMII_RD0, RGMII_RD1, RGMII_RD2, RGMII_RD3, RGMII_RXC, RGMII_RX_CTL, RGMII_TD0, RGMII_TD1, RGMII_TD2, RGMII_TD3, RGMII_TXC, RGMII_TX_CTL 00 **RESERVED0** — Reserved 01 **RESERVED1** — Reserved

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE_RGMII field descriptions (continued)

Field	Description
	10 **1P2V_IO** — 1.2V I/O interfaces including USB HSIC and MIPI_HSI. Provides calibrated drive strengths for signals ranging from 1.0V up to 1.3V. 11 **1P5V_IO** — 1.5V I/O interfaces including ENET. Provides calibrated drive strengths for signals ranging from 1.3V to 2.5V.
Reserved	This read-only field is reserved and always has the value 0.

36.4.481 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B2DS)

Address: 20E_0000h base + 794h offset = 20E_0794h

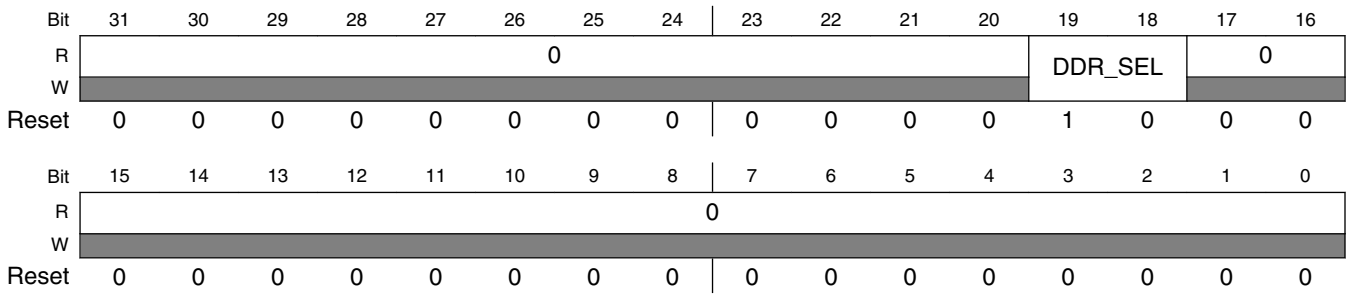
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																										DSE			0		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_B2DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.482 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE)

Address: 20E_0000h base + 798h offset = 20E_0798h

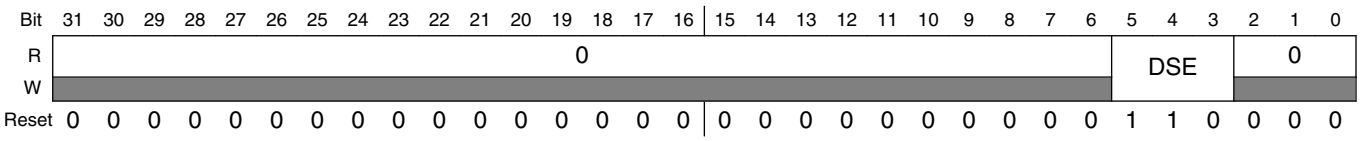


IOMUXC_SW_PAD_CTL_GRP_DDR_TYPE field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–18 DDR_SEL	DDR Select Field Select one of next values for group: . Affected pads: DRAM_ADDR00, DRAM_ADDR01, DRAM_ADDR02, DRAM_ADDR03, DRAM_ADDR04, DRAM_ADDR05, DRAM_ADDR06, DRAM_ADDR07, DRAM_ADDR08, DRAM_ADDR09, DRAM_ADDR10, DRAM_ADDR11, DRAM_ADDR12, DRAM_ADDR13, DRAM_ADDR14, DRAM_ADDR15, DRAM_CAS_B, DRAM_CS0_B, DRAM_CS1_B, DRAM_DATA00, DRAM_DATA01, DRAM_DATA02, DRAM_DATA03, DRAM_DATA04, DRAM_DATA05, DRAM_DATA06, DRAM_DATA07, DRAM_DATA08, DRAM_DATA09, DRAM_DATA10, DRAM_DATA11, DRAM_DATA12, DRAM_DATA13, DRAM_DATA14, DRAM_DATA15, DRAM_DATA16, DRAM_DATA17, DRAM_DATA18, DRAM_DATA19, DRAM_DATA20, DRAM_DATA21, DRAM_DATA22, DRAM_DATA23, DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31, DRAM_DATA32, DRAM_DATA33, DRAM_DATA34, DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39, DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47, DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55, DRAM_DATA56, DRAM_DATA57, DRAM_DATA58, DRAM_DATA59, DRAM_DATA60, DRAM_DATA61, DRAM_DATA62, DRAM_DATA63, DRAM_DQM0, DRAM_DQM1, DRAM_DQM2, DRAM_DQM3, DRAM_DQM4, DRAM_DQM5, DRAM_DQM6, DRAM_DQM7, DRAM_ODT0, DRAM_ODT1, DRAM_RAS_B, DRAM_SDBA0, DRAM_SDBA1, DRAM_SDBA2, DRAM_SDCKE0, DRAM_SDCKE1, DRAM_SDCLK0_P, DRAM_SDCLK1_P, DRAM_SDQS0_P, DRAM_SDQS1_P, DRAM_SDQS2_P, DRAM_SDQS3_P, DRAM_SDQS4_P, DRAM_SDQS5_P, DRAM_SDQS6_P, DRAM_SDQS7_P, DRAM_SDWE_B 00 **RESERVED0** — Reserved 01 **RESERVED1** — Reserved 10 **LPDDR2** — LPDDR2 mode (240 Ohm driver unit calibration, 240, 120, 80, 60, 48, 40, 32 Ohm drive strengths at 1.2V) 11 **DDR3** — DDR3 mode (240 Ohm driver unit calibration, 240, 120, 80, 60, 48, 40, 32 Ohm drive strengths at 1.5V)
Reserved	This read-only field is reserved and always has the value 0.

36.4.483 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B3DS)

Address: 20E_0000h base + 79Ch offset = 20E_079Ch

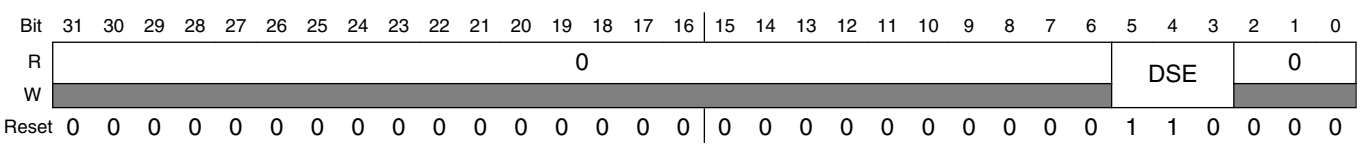


IOMUXC_SW_PAD_CTL_GRP_B3DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA24, DRAM_DATA25, DRAM_DATA26, DRAM_DATA27, DRAM_DATA28, DRAM_DATA29, DRAM_DATA30, DRAM_DATA31 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.484 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B4DS)

Address: 20E_0000h base + 7A0h offset = 20E_07A0h



IOMUXC_SW_PAD_CTL_GRP_B4DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_B4DS field descriptions (continued)

Field	Description
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA32, DRAM_DATA33, DRAM_DATA34, DRAM_DATA35, DRAM_DATA36, DRAM_DATA37, DRAM_DATA38, DRAM_DATA39 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.485 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B5DS)

Address: 20E_0000h base + 7A4h offset = 20E_07A4h

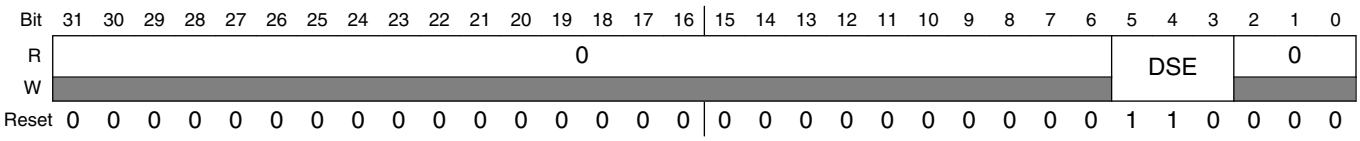
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																										DSE			0		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

IOMUXC_SW_PAD_CTL_GRP_B5DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA40, DRAM_DATA41, DRAM_DATA42, DRAM_DATA43, DRAM_DATA44, DRAM_DATA45, DRAM_DATA46, DRAM_DATA47 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.486 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_B6DS)

Address: 20E_0000h base + 7A8h offset = 20E_07A8h

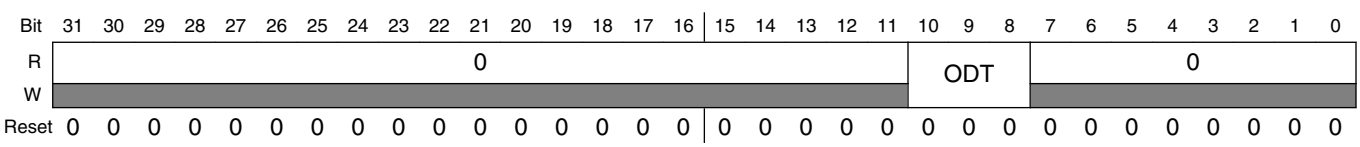


IOMUXC_SW_PAD_CTL_GRP_B6DS field descriptions

Field	Description
31–6 Reserved	This read-only field is reserved and always has the value 0.
5–3 DSE	Drive Strength Field Select one of next values for group: . Affected pads: DRAM_DATA48, DRAM_DATA49, DRAM_DATA50, DRAM_DATA51, DRAM_DATA52, DRAM_DATA53, DRAM_DATA54, DRAM_DATA55 000 **HIZ** — HI-Z 001 **240_OHM** — 240 Ohm 010 **120_OHM** — 120 Ohm 011 **80_OHM** — 80 Ohm 100 **60_OHM** — 60 Ohm 101 **48_OHM** — 48 Ohm 110 **40_OHM** — 40 Ohm 111 **34_OHM** — 34 Ohm
Reserved	This read-only field is reserved and always has the value 0.

36.4.487 Pad Group Control Register (IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM)

Address: 20E_0000h base + 7ACh offset = 20E_07ACh



IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM field descriptions

Field	Description
31–11 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_SW_PAD_CTL_GRP_RGMII_TERM field descriptions (continued)

Field	Description
10–8 ODT	On Die Termination Field Select one of next values for group: . Affected pads: RGMII_RD0, RGMII_RD1, RGMII_RD2, RGMII_RD3, RGMII_RXC, RGMII_RX_CTL 000 **DISABLED** — Disabled 001 **120_OHM** — 120 Ohm ODT 010 **60_OHM** — 60 Ohm ODT 011 **40_OHM** — 40 Ohm ODT 100 **30_OHM** — 30 Ohm ODT 101 **24_OHM** — 24 Ohm ODT 110 **20_OHM** — 20 Ohm ODT 111 **17_OHM** — 17 Ohm ODT
Reserved	This read-only field is reserved and always has the value 0.

36.4.488 Select Input Register (IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT)

Address: 20E_0000h base + 7B0h offset = 20E_07B0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

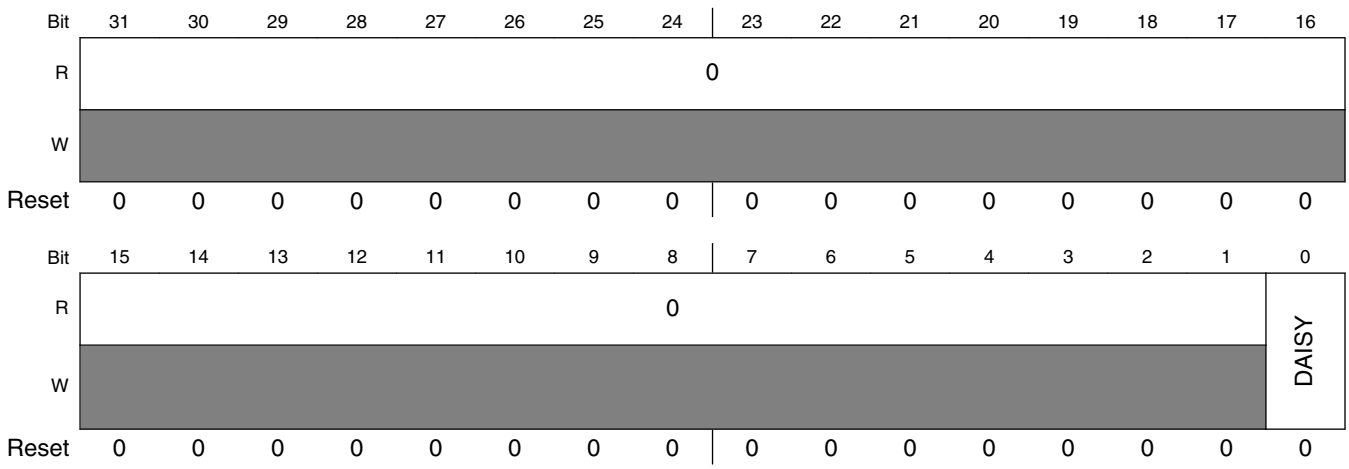
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ASRC_ASRCK_CLOCK_6_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **KEY_ROW3_ALT1** — Selecting ALT1 mode of pad KEY_ROW3 for ASRC_EXT_CLK. 01 **GPIO00_ALT3** — Selecting ALT3 mode of pad GPIO_0 for ASRC_EXT_CLK. 10 **GPIO18_ALT4** — Selecting ALT4 mode of pad GPIO_18 for ASRC_EXT_CLK.

36.4.489 Select Input Register (IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7B4h offset = 20E_07B4h



IOMUXC_AUD4_INPUT_DA_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA0_ALT3** — Selecting ALT3 mode of pad SD2_DAT0 for AUD4_RXD. 1 **DISP0_DATA23_ALT3** — Selecting ALT3 mode of pad DISP0_DAT23 for AUD4_RXD.

36.4.490 Select Input Register (IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7B8h offset = 20E_07B8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

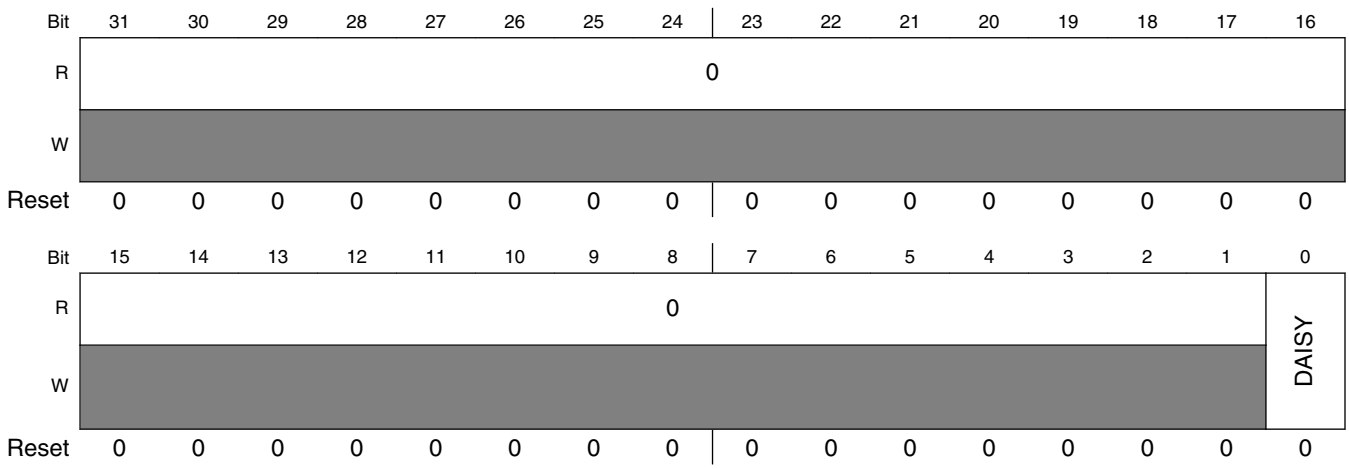
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_AUD4_INPUT_DB_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA2_ALT3** — Selecting ALT3 mode of pad SD2_DAT2 for AUD4_TXD. 1 **DISP0_DATA21_ALT3** — Selecting ALT3 mode of pad DISP0_DAT21 for AUD4_TXD.

36.4.491 Select Input Register (IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7BCh offset = 20E_07BCh



IOMUXC_AUD4_INPUT_RXCLK_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA19_ALT4** — Selecting ALT4 mode of pad DISP0_DAT19 for AUD4_RXC. 1 **SD2_CMD_ALT3** — Selecting ALT3 mode of pad SD2_CMD for AUD4_RXC.

36.4.492 Select Input Register (IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7C0h offset = 20E_07C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

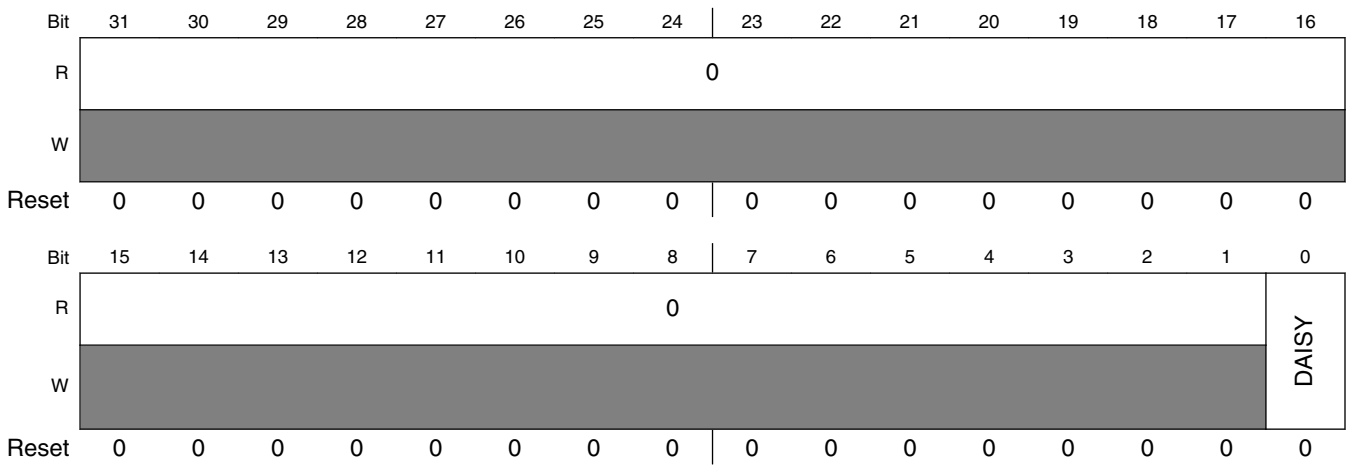
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_AUD4_INPUT_RXFS_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA18_ALT4** — Selecting ALT4 mode of pad DISP0_DAT18 for AUD4_RXFS. 1 **SD2_CLK_ALT3** — Selecting ALT3 mode of pad SD2_CLK for AUD4_RXFS.

36.4.493 Select Input Register (IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7C4h offset = 20E_07C4h

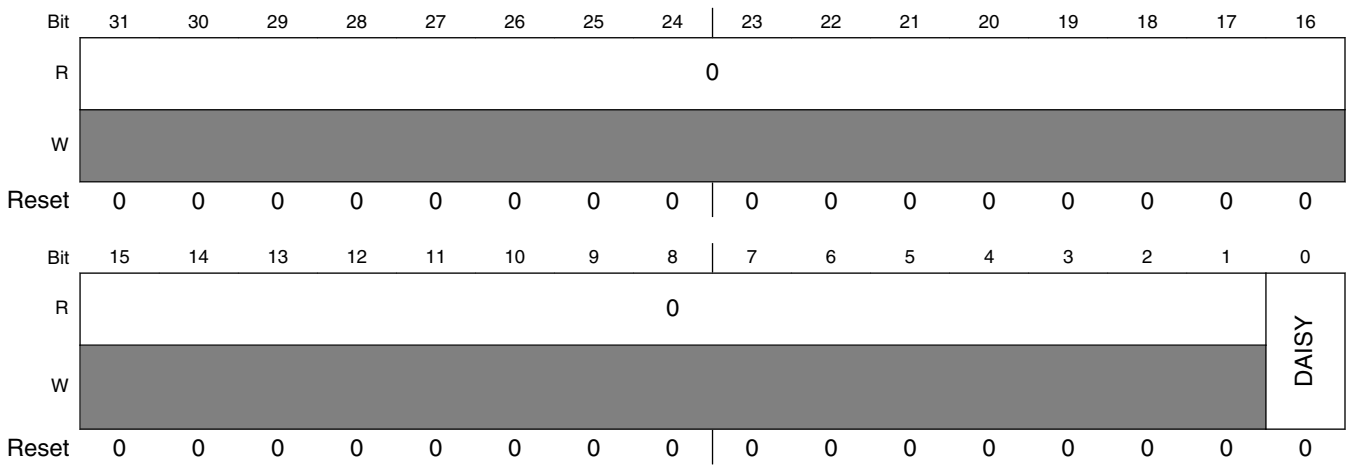


IOMUXC_AUD4_INPUT_TXCLK_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA20_ALT3** — Selecting ALT3 mode of pad DISP0_DAT20 for AUD4_TXC. 1 **SD2_DATA3_ALT3** — Selecting ALT3 mode of pad SD2_DAT3 for AUD4_TXC.

36.4.494 Select Input Register (IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7C8h offset = 20E_07C8h

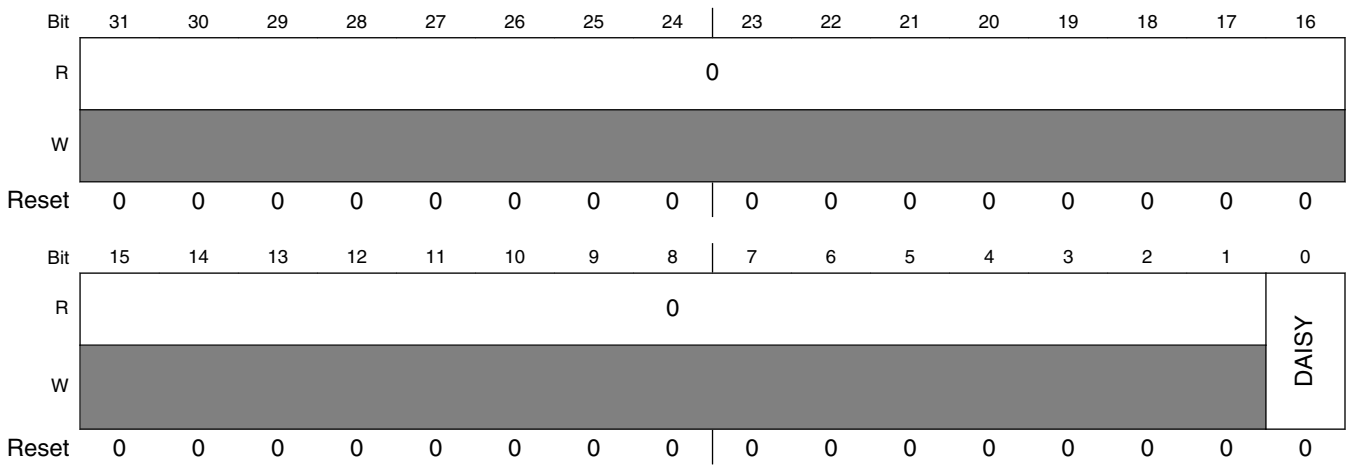


IOMUXC_AUD4_INPUT_TXFS_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA1_ALT3** — Selecting ALT3 mode of pad SD2_DAT1 for AUD4_TXFS. 1 **DISP0_DATA22_ALT3** — Selecting ALT3 mode of pad DISP0_DAT22 for AUD4_TXFS.

36.4.495 Select Input Register (IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7CCh offset = 20E_07CCh



IOMUXC_AUD5_INPUT_DA_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA19_ALT3** — Selecting ALT3 mode of pad DISP0_DAT19 for AUD5_RXD. 1 **KEY_ROW1_ALT2** — Selecting ALT2 mode of pad KEY_ROW1 for AUD5_RXD.

36.4.496 Select Input Register (IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7D0h offset = 20E_07D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

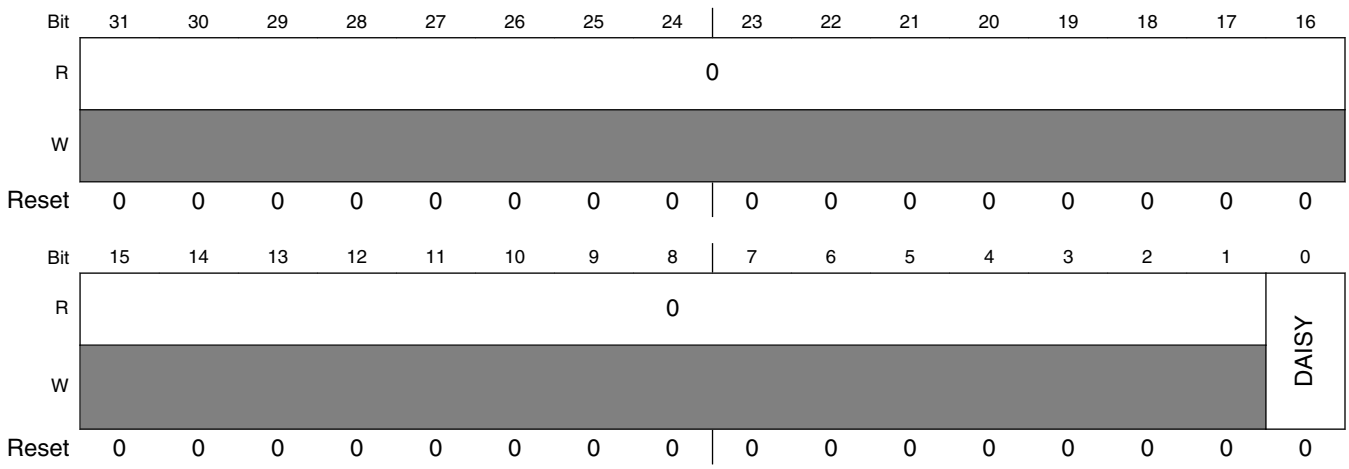
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_AUD5_INPUT_DB_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA17_ALT3** — Selecting ALT3 mode of pad DISP0_DAT17 for AUD5_TXD. 1 **KEY_ROW0_ALT2** — Selecting ALT2 mode of pad KEY_ROW0 for AUD5_TXD.

36.4.497 Select Input Register (IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7D4h offset = 20E_07D4h



IOMUXC_AUD5_INPUT_RXCLK_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA25_ALT6** — Selecting ALT6 mode of pad EIM_D25 for AUD5_RXC. 1 **DISP0_DATA14_ALT3** — Selecting ALT3 mode of pad DISP0_DAT14 for AUD5_RXC.

36.4.498 Select Input Register (IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7D8h offset = 20E_07D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

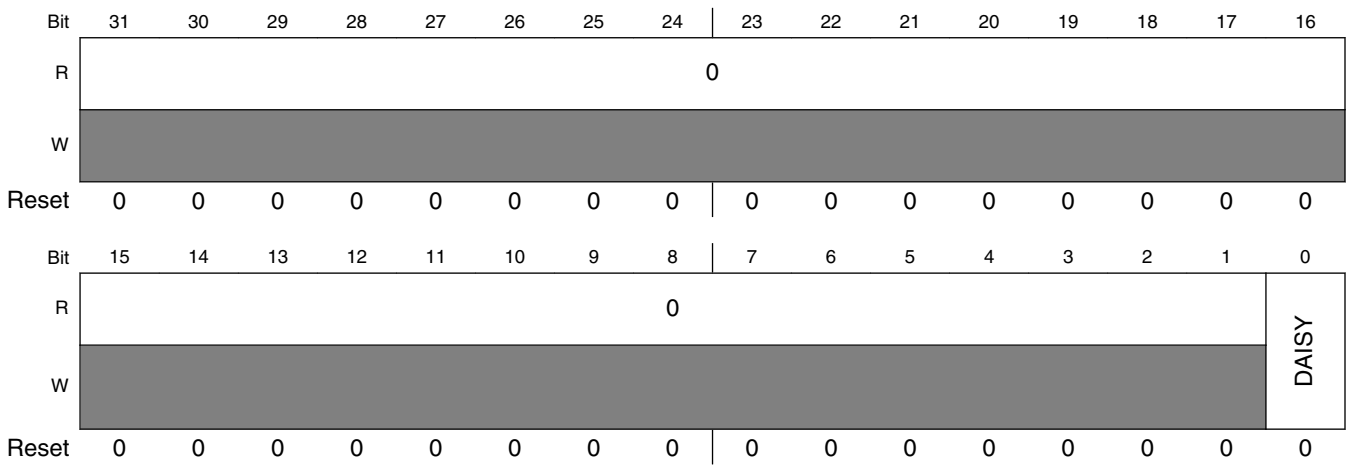
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_AUD5_INPUT_RXFS_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA24_ALT6** — Selecting ALT6 mode of pad EIM_D24 for AUD5_RXFS. 1 **DISP0_DATA13_ALT3** — Selecting ALT3 mode of pad DISP0_DAT13 for AUD5_RXFS.

36.4.499 Select Input Register (IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7DCh offset = 20E_07DCh



IOMUXC_AUD5_INPUT_TXCLK_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA16_ALT3** — Selecting ALT3 mode of pad DISP0_DAT16 for AUD5_TXC. 1 **KEY_COL0_ALT2** — Selecting ALT2 mode of pad KEY_COL0 for AUD5_TXC.

36.4.500 Select Input Register (IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT)

Address: 20E_0000h base + 7E0h offset = 20E_07E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_AUD5_INPUT_TXFS_AMX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA18_ALT3** — Selecting ALT3 mode of pad DISP0_DAT18 for AUD5_TXFS. 1 **KEY_COL1_ALT2** — Selecting ALT2 mode of pad KEY_COL1 for AUD5_TXFS.

36.4.501 Select Input Register (IOMUXC_FLEXCAN1_RX_SELECT_INPUT)

Address: 20E_0000h base + 7E4h offset = 20E_07E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

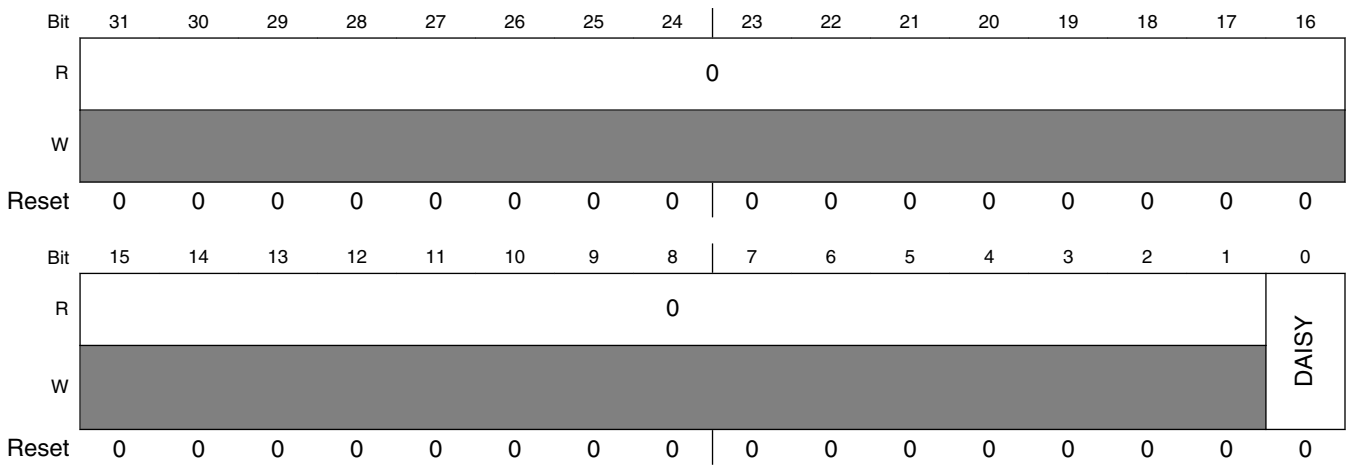
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_FLEXCAN1_RX_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **KEY_ROW2_ALT2** — Selecting ALT2 mode of pad KEY_ROW2 for FLEXCAN1_RX. 01 **GPIO08_ALT3** — Selecting ALT3 mode of pad GPIO_8 for FLEXCAN1_RX. 10 **SD3_CLK_ALT2** — Selecting ALT2 mode of pad SD3_CLK for FLEXCAN1_RX.

36.4.502 Select Input Register (IOMUXC_FLEXCAN2_RX_SELECT_INPUT)

Address: 20E_0000h base + 7E8h offset = 20E_07E8h



IOMUXC_FLEXCAN2_RX_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **KEY_ROW4_ALT0** — Selecting ALT0 mode of pad KEY_ROW4 for FLEXCAN2_RX. 1 **SD3_DATA1_ALT2** — Selecting ALT2 mode of pad SD3_DAT1 for FLEXCAN2_RX.

36.4.503 Select Input Register (IOMUXC_CCM_PMIC_READY_SELECT_INPUT)

Address: 20E_0000h base + 7F0h offset = 20E_07F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_CCM_PMIC_READY_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_EB0_B_ALT4** — Selecting ALT4 mode of pad EIM_EB0 for CCM_PMIC_READY. 1 **GPIO17_ALT2** — Selecting ALT2 mode of pad GPIO_17 for CCM_PMIC_READY.

36.4.504 Select Input Register (IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT)

Address: 20E_0000h base + 7F4h offset = 20E_07F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI1_CSPI_CLK_IN_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA16_ALT1** — Selecting ALT1 mode of pad EIM_D16 for ECSPI1_SCLK. 01 **DISP0_DATA20_ALT2** — Selecting ALT2 mode of pad DISP0_DAT20 for ECSPI1_SCLK. 10 **KEY_COL0_ALT0** — Selecting ALT0 mode of pad KEY_COL0 for ECSPI1_SCLK. 11 **CSI0_DATA04_ALT2** — Selecting ALT2 mode of pad CSI0_DAT4 for ECSPI1_SCLK.

36.4.505 Select Input Register (IOMUXC_ECSPI1_MISO_SELECT_INPUT)

Address: 20E_0000h base + 7F8h offset = 20E_07F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI1_MISO_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA17_ALT1** — Selecting ALT1 mode of pad EIM_D17 for ECSPI1_MISO. 01 **DISP0_DATA22_ALT2** — Selecting ALT2 mode of pad DISP0_DAT22 for ECSPI1_MISO. 10 **KEY_COL1_ALT0** — Selecting ALT0 mode of pad KEY_COL1 for ECSPI1_MISO. 11 **CSI0_DATA06_ALT2** — Selecting ALT2 mode of pad CSI0_DAT6 for ECSPI1_MISO.

36.4.506 Select Input Register (IOMUXC_ECSPI1_MOSI_SELECT_INPUT)

Address: 20E_0000h base + 7FCh offset = 20E_07FCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI1_MOSI_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA18_ALT1** — Selecting ALT1 mode of pad EIM_D18 for ECSPI1_MOSI. 01 **DISP0_DATA21_ALT2** — Selecting ALT2 mode of pad DISP0_DAT21 for ECSPI1_MOSI. 10 **KEY_ROW0_ALT0** — Selecting ALT0 mode of pad KEY_ROW0 for ECSPI1_MOSI. 11 **CSI0_DATA05_ALT2** — Selecting ALT2 mode of pad CSI0_DAT5 for ECSPI1_MOSI.

36.4.507 Select Input Register (IOMUXC_ECSPI1_SS0_SELECT_INPUT)

Address: 20E_0000h base + 800h offset = 20E_0800h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

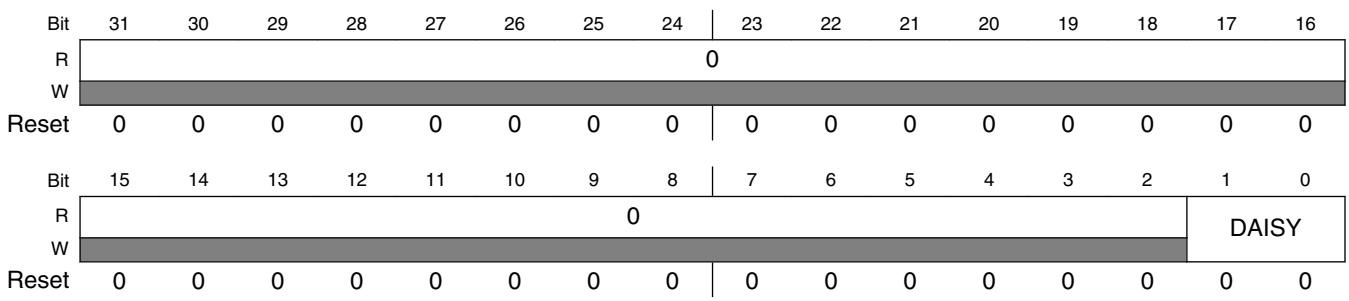
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI1_SS0_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_EB2_B_ALT1** — Selecting ALT1 mode of pad EIM_EB2 for ECSPI1_SS0. 01 **DISP0_DATA23_ALT2** — Selecting ALT2 mode of pad DISP0_DAT23 for ECSPI1_SS0. 10 **KEY_ROW1_ALT0** — Selecting ALT0 mode of pad KEY_ROW1 for ECSPI1_SS0. 11 **CSI0_DATA07_ALT2** — Selecting ALT2 mode of pad CSI0_DAT7 for ECSPI1_SS0.

36.4.508 Select Input Register (IOMUXC_ECSPI1_SS1_SELECT_INPUT)

Address: 20E_0000h base + 804h offset = 20E_0804h



IOMUXC_ECSPI1_SS1_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA19_ALT1** — Selecting ALT1 mode of pad EIM_D19 for ECSPI1_SS1. 01 **DISP0_DATA15_ALT2** — Selecting ALT2 mode of pad DISP0_DAT15 for ECSPI1_SS1. 10 **KEY_COL2_ALT0** — Selecting ALT0 mode of pad KEY_COL2 for ECSPI1_SS1.

36.4.509 Select Input Register (IOMUXC_ECSPI1_SS2_SELECT_INPUT)

Address: 20E_0000h base + 808h offset = 20E_0808h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

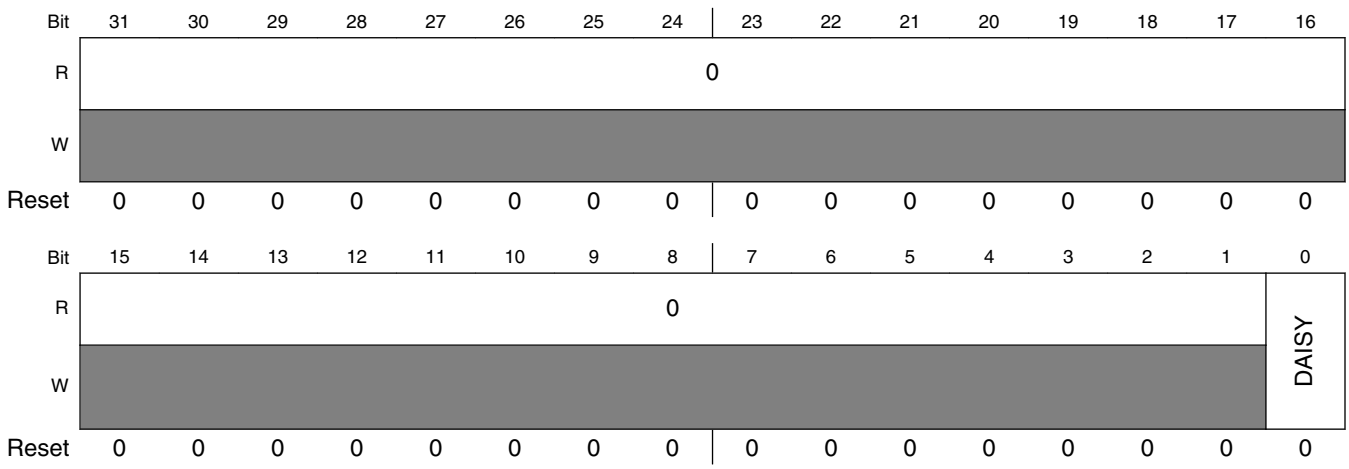
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI1_SS2_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA24_ALT3** — Selecting ALT3 mode of pad EIM_D24 for ECSPI1_SS2. 1 **KEY_ROW2_ALT0** — Selecting ALT0 mode of pad KEY_ROW2 for ECSPI1_SS2.

36.4.510 Select Input Register (IOMUXC_ECSPI1_SS3_SELECT_INPUT)

Address: 20E_0000h base + 80Ch offset = 20E_080Ch

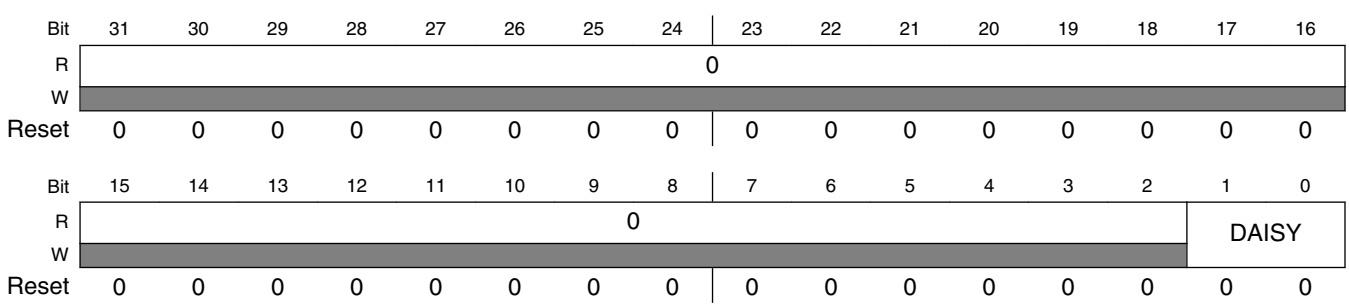


IOMUXC_ECSPI1_SS3_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA25_ALT3** — Selecting ALT3 mode of pad EIM_D25 for ECSPI1_SS3. 1 **KEY_COL3_ALT0** — Selecting ALT0 mode of pad KEY_COL3 for ECSPI1_SS3.

36.4.511 Select Input Register (IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT)

Address: 20E_0000h base + 810h offset = 20E_0810h

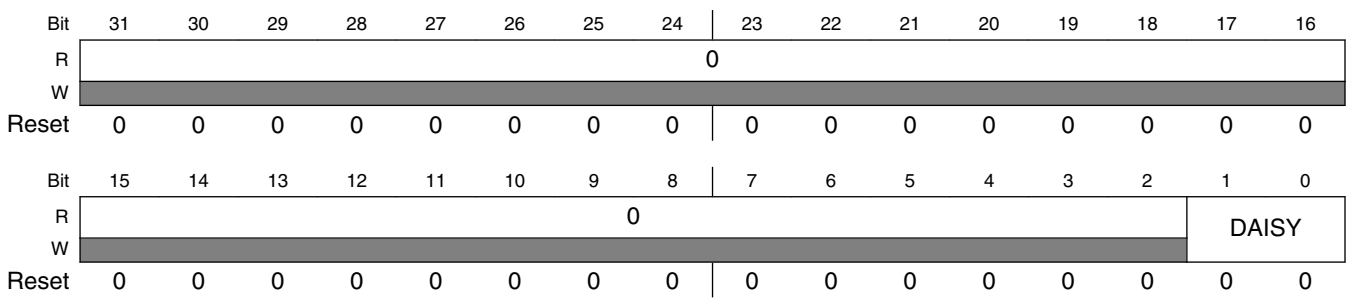


IOMUXC_ECSPI2_CSPI_CLK_IN_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_CS0_B_ALT2** — Selecting ALT2 mode of pad EIM_CS0 for ECSPI2_SCLK. 01 **DISP0_DATA19_ALT2** — Selecting ALT2 mode of pad DISP0_DAT19 for ECSPI2_SCLK. 10 **CSI0_DATA08_ALT2** — Selecting ALT2 mode of pad CSI0_DAT8 for ECSPI2_SCLK.

36.4.512 Select Input Register (IOMUXC_ECSPI2_MISO_SELECT_INPUT)

Address: 20E_0000h base + 814h offset = 20E_0814h

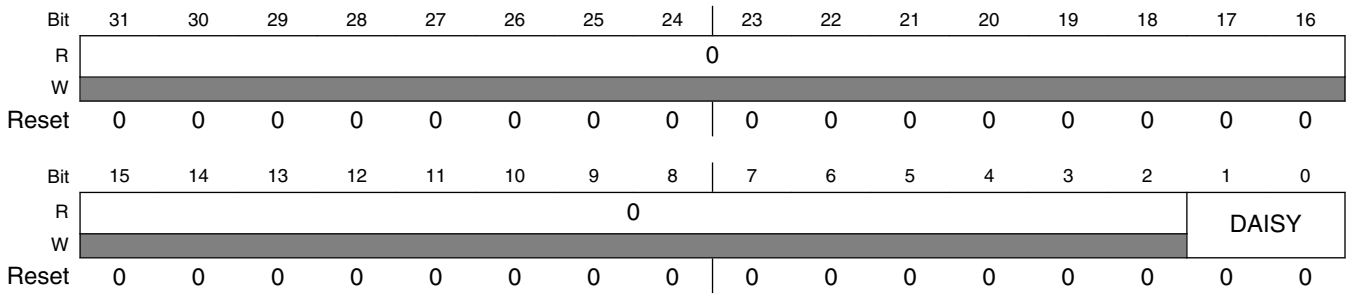


IOMUXC_ECSPI2_MISO_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_OE_B_ALT2** — Selecting ALT2 mode of pad EIM_OE for ECSPI2_MISO. 01 **DISP0_DATA17_ALT2** — Selecting ALT2 mode of pad DISP0_DAT17 for ECSPI2_MISO. 10 **CSI0_DATA10_ALT2** — Selecting ALT2 mode of pad CSI0_DAT10 for ECSPI2_MISO.

36.4.513 Select Input Register (IOMUXC_ECSPI2_MOSI_SELECT_INPUT)

Address: 20E_0000h base + 818h offset = 20E_0818h

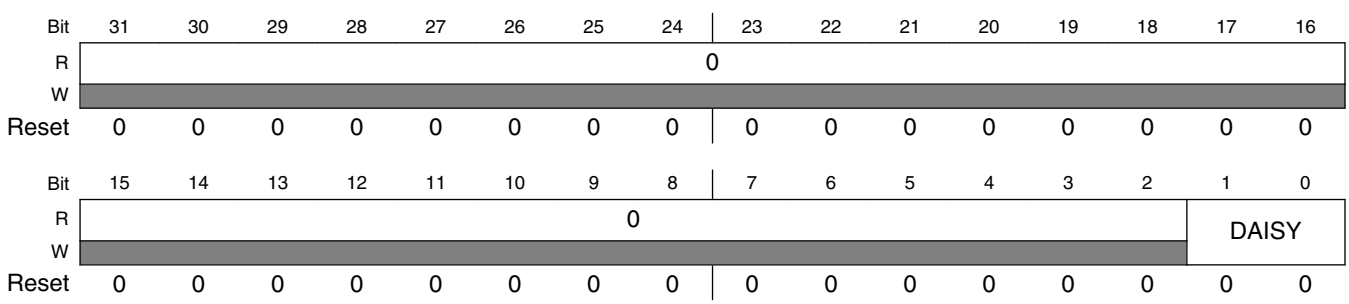


IOMUXC_ECSPI2_MOSI_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_CS1_B_ALT2** — Selecting ALT2 mode of pad EIM_CS1 for ECSPI2_MOSI. 01 **DISP0_DATA16_ALT2** — Selecting ALT2 mode of pad DISP0_DAT16 for ECSPI2_MOSI. 10 **CSI0_DATA09_ALT2** — Selecting ALT2 mode of pad CSI0_DAT9 for ECSPI2_MOSI.

36.4.514 Select Input Register (IOMUXC_ECSPI2_SS0_SELECT_INPUT)

Address: 20E_0000h base + 81Ch offset = 20E_081Ch



IOMUXC_ECSPI2_SS0_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.

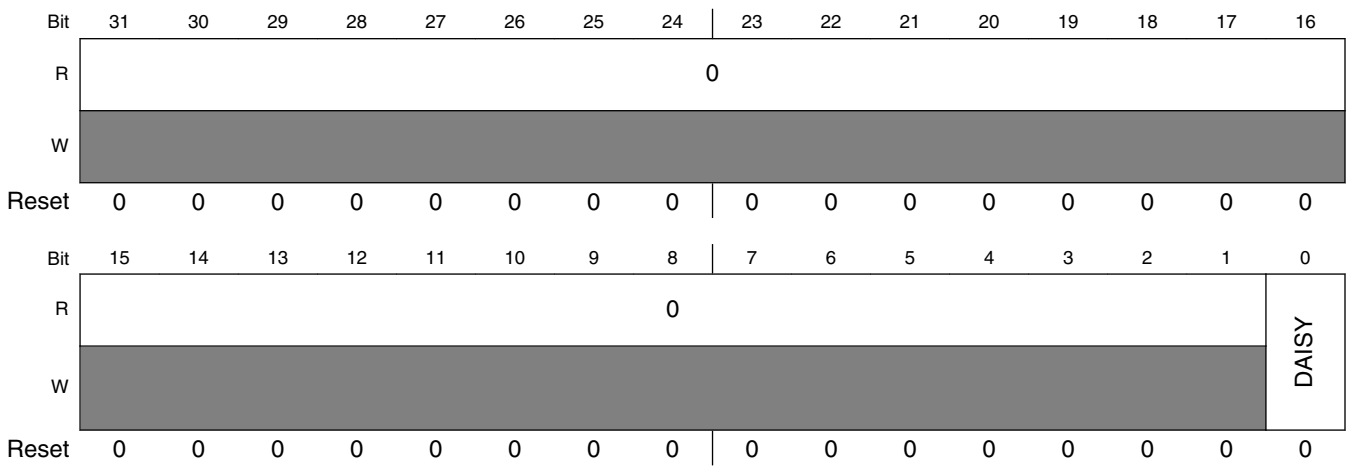
Table continues on the next page...

IOMUXC_ECSPI2_SS0_SELECT_INPUT field descriptions (continued)

Field	Description
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_RW_ALT2** — Selecting ALT2 mode of pad EIM_RW for ECSPI2_SS0. 01 **DISP0_DATA18_ALT2** — Selecting ALT2 mode of pad DISP0_DAT18 for ECSPI2_SS0. 10 **CSI0_DATA11_ALT2** — Selecting ALT2 mode of pad CSI0_DAT11 for ECSPI2_SS0.

36.4.515 Select Input Register (IOMUXC_ECSPI2_SS1_SELECT_INPUT)

Address: 20E_0000h base + 820h offset = 20E_0820h

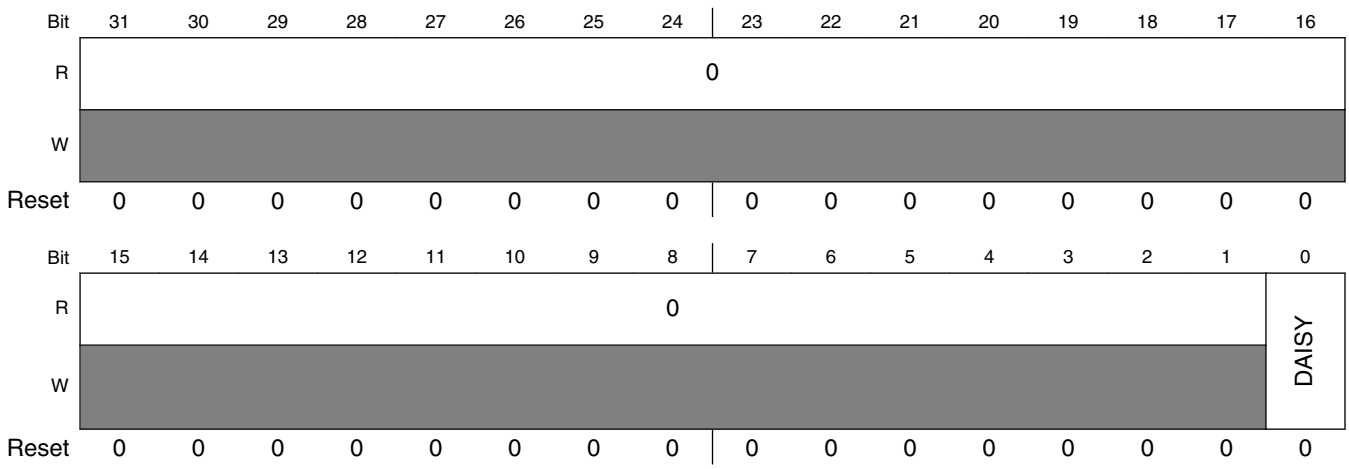


IOMUXC_ECSPI2_SS1_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_LBA_B_ALT2** — Selecting ALT2 mode of pad EIM_LBA for ECSPI2_SS1. 1 **DISP0_DATA15_ALT3** — Selecting ALT3 mode of pad DISP0_DAT15 for ECSPI2_SS1.

36.4.516 Select Input Register (IOMUXC_ECSPI4_SS0_SELECT_INPUT)

Address: 20E_0000h base + 824h offset = 20E_0824h



IOMUXC_ECSPI4_SS0_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA20_ALT1** — Selecting ALT1 mode of pad EIM_D20 for ECSPI4_SS0. 1 **EIM_DATA29_ALT2** — Selecting ALT2 mode of pad EIM_D29 for ECSPI4_SS0.

36.4.517 Select Input Register (IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT)

Address: 20E_0000h base + 828h offset = 20E_0828h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

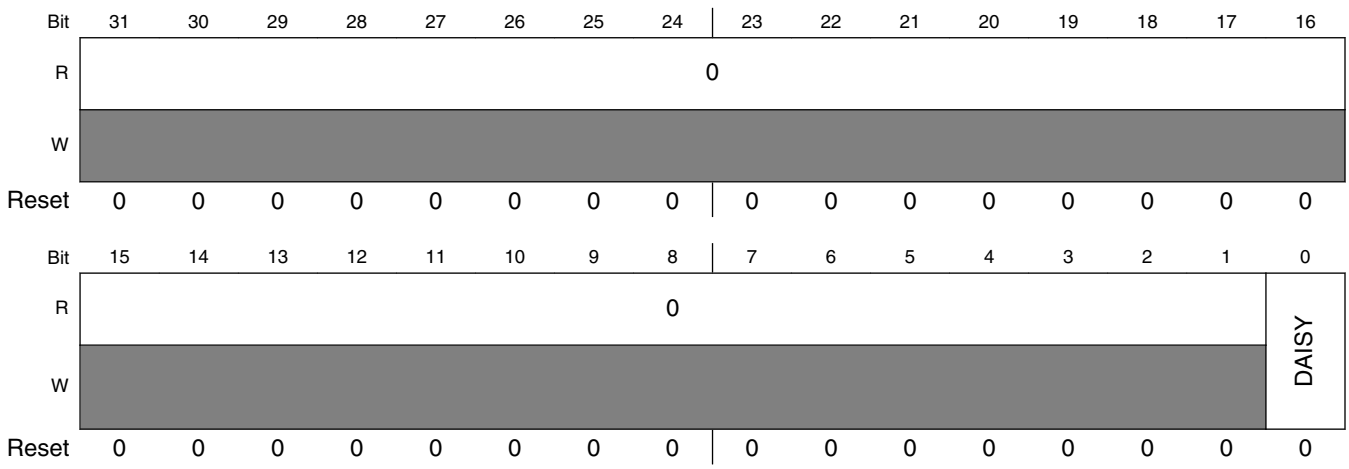
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI5_CSPI_CLK_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD1_CLK_ALT1** — Selecting ALT1 mode of pad SD1_CLK for ECSPI5_SCLK. 1 **SD2_CLK_ALT1** — Selecting ALT1 mode of pad SD2_CLK for ECSPI5_SCLK.

36.4.518 Select Input Register (IOMUXC_ECSPI5_MISO_SELECT_INPUT)

Address: 20E_0000h base + 82Ch offset = 20E_082Ch



IOMUXC_ECSPI5_MISO_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA0_ALT1** — Selecting ALT1 mode of pad SD2_DAT0 for ECSPI5_MISO. 1 **SD1_DATA0_ALT1** — Selecting ALT1 mode of pad SD1_DAT0 for ECSPI5_MISO.

36.4.519 Select Input Register (IOMUXC_ECSPI5_MOSI_SELECT_INPUT)

Address: 20E_0000h base + 830h offset = 20E_0830h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

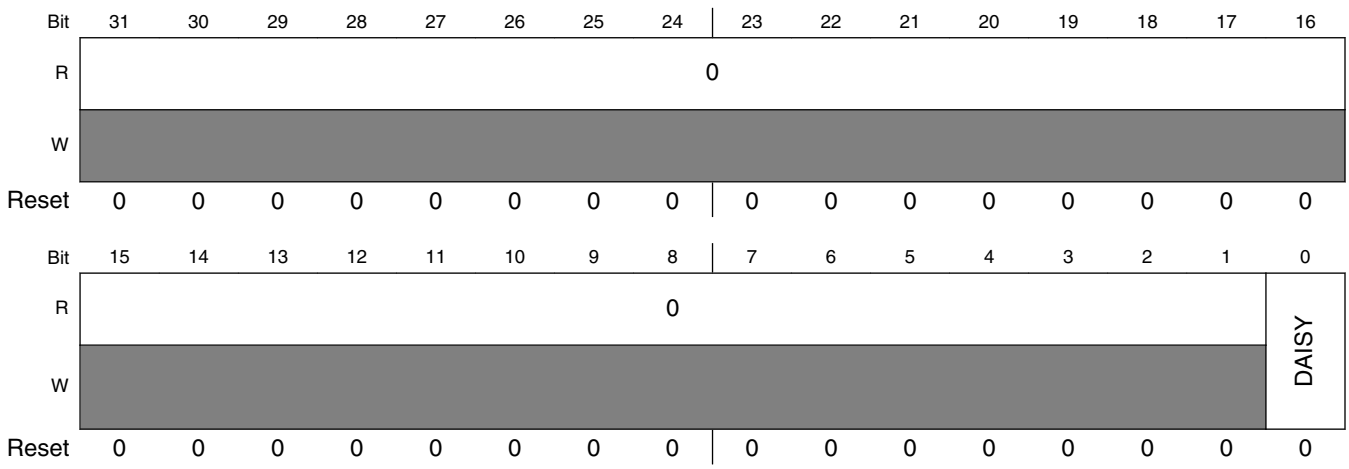
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ECSPI5_MOSI_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD1_CMD_ALT1** — Selecting ALT1 mode of pad SD1_CMD for ECSPI5_MOSI. 1 **SD2_CMD_ALT1** — Selecting ALT1 mode of pad SD2_CMD for ECSPI5_MOSI.

36.4.520 Select Input Register (IOMUXC_ECSPI5_SS0_SELECT_INPUT)

Address: 20E_0000h base + 834h offset = 20E_0834h

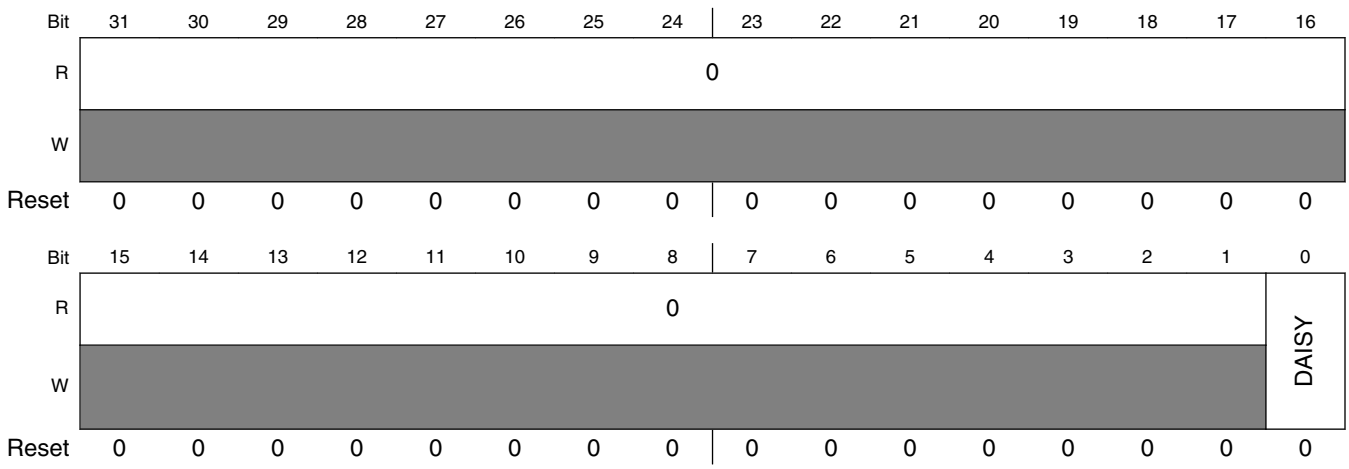


IOMUXC_ECSPI5_SS0_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA1_ALT1** — Selecting ALT1 mode of pad SD2_DAT1 for ECSPI5_SS0. 1 **SD1_DATA1_ALT1** — Selecting ALT1 mode of pad SD1_DAT1 for ECSPI5_SS0.

36.4.521 Select Input Register (IOMUXC_ECSPI5_SS1_SELECT_INPUT)

Address: 20E_0000h base + 838h offset = 20E_0838h

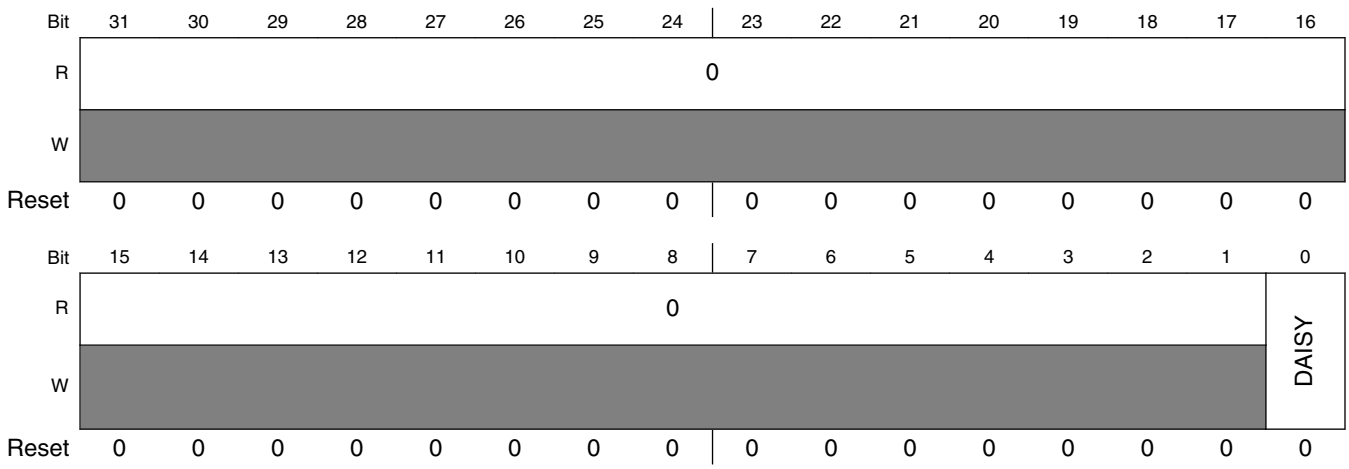


IOMUXC_ECSPI5_SS1_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **SD2_DATA2_ALT1** — Selecting ALT1 mode of pad SD2_DAT2 for ECSPI5_SS1. 1 **SD1_DATA2_ALT1** — Selecting ALT1 mode of pad SD1_DAT2 for ECSPI5_SS1.

36.4.522 Select Input Register (IOMUXC_ENET_REF_CLK_SELECT_INPUT)

Address: 20E_0000h base + 83Ch offset = 20E_083Ch



IOMUXC_ENET_REF_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_TX_CTL_ALT7** — Selecting ALT7 mode of pad RGMII_TX_CTL for ENET_REF_CLK. 1 **GPIO16_ALT2** — Selecting ALT2 mode of pad GPIO_16 for ENET_REF_CLK.

36.4.523 Select Input Register (IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT)

Address: 20E_0000h base + 840h offset = 20E_0840h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

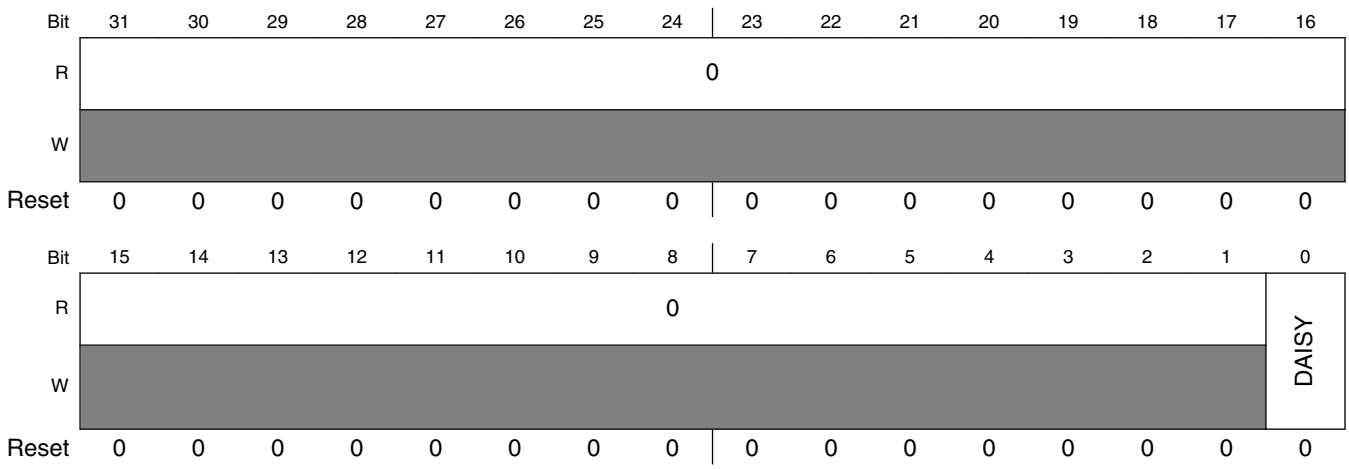
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ENET_MAC0_MDIO_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_MDIO_ALT1** — Selecting ALT1 mode of pad ENET_MDIO for ENET_MDIO. 1 **KEY_COL1_ALT1** — Selecting ALT1 mode of pad KEY_COL1 for ENET_MDIO.

36.4.524 Select Input Register (IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT)

Address: 20E_0000h base + 844h offset = 20E_0844h



IOMUXC_ENET_MAC0_RX_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RXC_ALT1** — Selecting ALT1 mode of pad RGMII_RXC for RGMII_RXC. 1 **GPIO18_ALT1** — Selecting ALT1 mode of pad GPIO_18 for ENET_RX_CLK.

36.4.525 Select Input Register (IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT)

Address: 20E_0000h base + 848h offset = 20E_0848h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

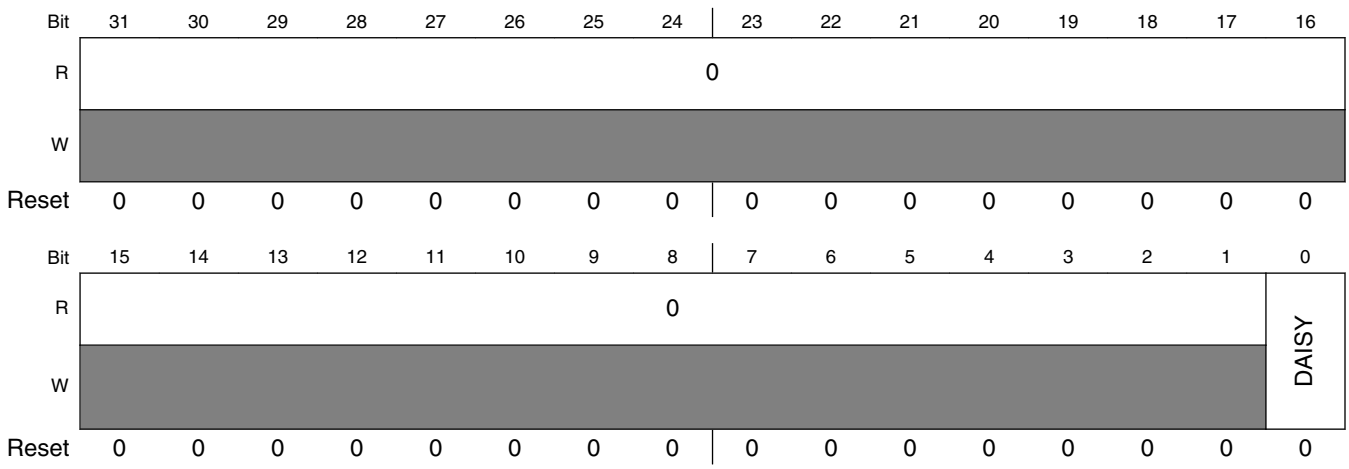
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ENET_MAC0_RX_DATA0_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RD0_ALT1** — Selecting ALT1 mode of pad RGMII_RD0 for RGMII_RD0. 1 **ENET_RX_DATA0_ALT1** — Selecting ALT1 mode of pad ENET_RXD0 for ENET_RX_DATA0.

36.4.526 Select Input Register (IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT)

Address: 20E_0000h base + 84Ch offset = 20E_084Ch



IOMUXC_ENET_MAC0_RX_DATA1_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RD1_ALT1** — Selecting ALT1 mode of pad RGMII_RD1 for RGMII_RD1. 1 **ENET_RX_DATA1_ALT1** — Selecting ALT1 mode of pad ENET_RXD1 for ENET_RX_DATA1.

36.4.527 Select Input Register (IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT)

Address: 20E_0000h base + 850h offset = 20E_0850h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

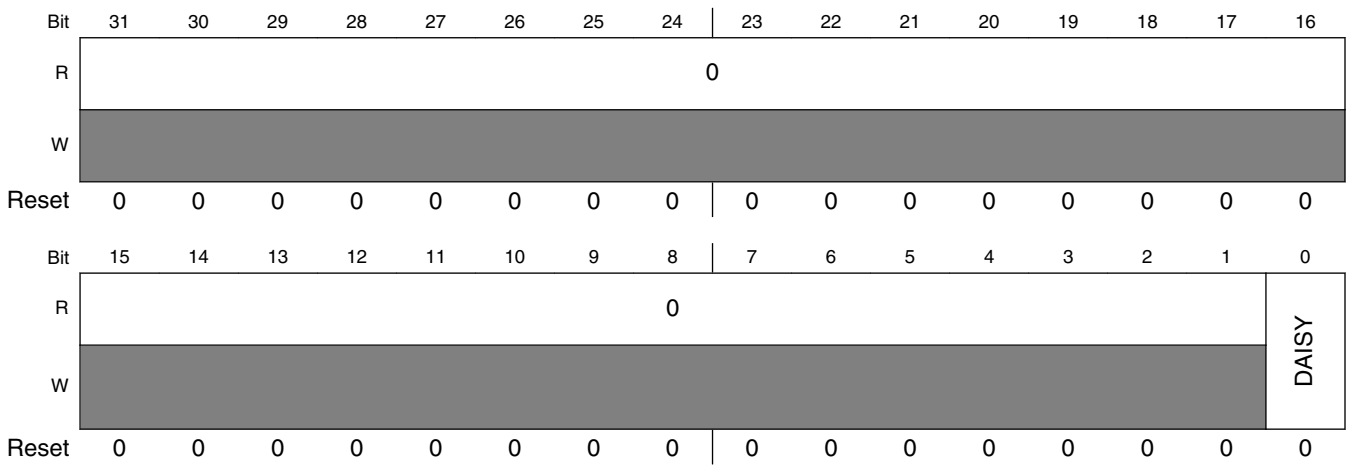
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ENET_MAC0_RX_DATA2_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RD2_ALT1** — Selecting ALT1 mode of pad RGMII_RD2 for RGMII_RD2. 1 **KEY_COL2_ALT1** — Selecting ALT1 mode of pad KEY_COL2 for ENET_RX_DATA2.

36.4.528 Select Input Register (IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT)

Address: 20E_0000h base + 854h offset = 20E_0854h



IOMUXC_ENET_MAC0_RX_DATA3_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RD3_ALT1** — Selecting ALT1 mode of pad RGMII_RD3 for RGMII_RD3. 1 **KEY_COL0_ALT1** — Selecting ALT1 mode of pad KEY_COL0 for ENET_RX_DATA3.

36.4.529 Select Input Register (IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT)

Address: 20E_0000h base + 858h offset = 20E_0858h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

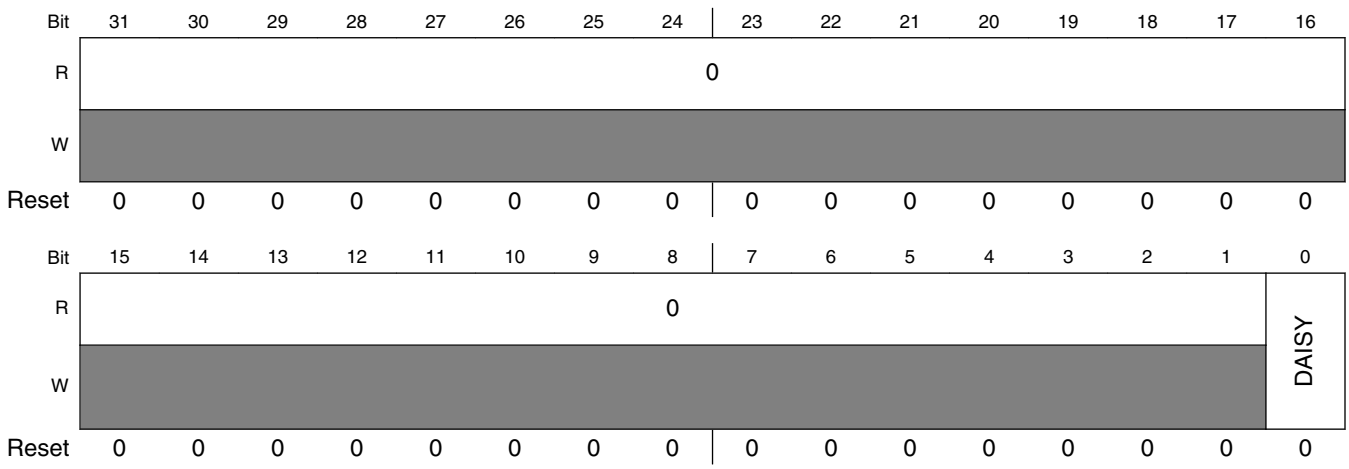
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ENET_MAC0_RX_EN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_RX_CTL_ALT1** — Selecting ALT1 mode of pad RGMII_RX_CTL for RGMII_RX_CTL. 1 **ENET_CRS_DV_ALT1** — Selecting ALT1 mode of pad ENET_CRS_DV for ENET_RX_EN.

36.4.530 Select Input Register (IOMUXC_ESAI_RX_FS_SELECT_INPUT)

Address: 20E_0000h base + 85Ch offset = 20E_085Ch

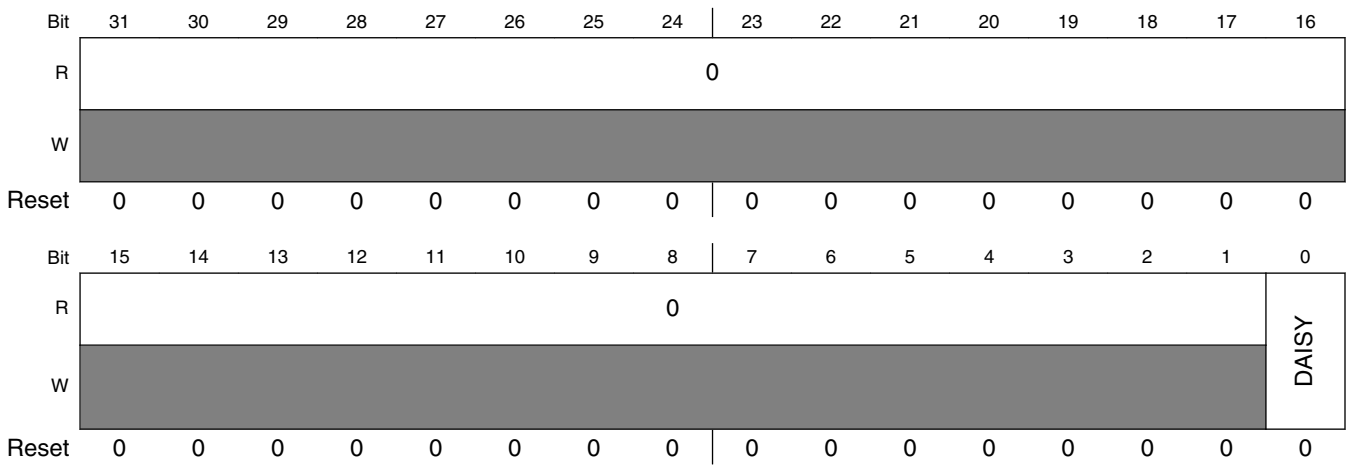


IOMUXC_ESAI_RX_FS_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_REF_CLK_ALT2** — Selecting ALT2 mode of pad ENET_REF_CLK for ESAI_RX_FS. 1 **GPIO09_ALT0** — Selecting ALT0 mode of pad GPIO_9 for ESAI_RX_FS.

36.4.531 Select Input Register (IOMUXC_ESAI_TX_FS_SELECT_INPUT)

Address: 20E_0000h base + 860h offset = 20E_0860h

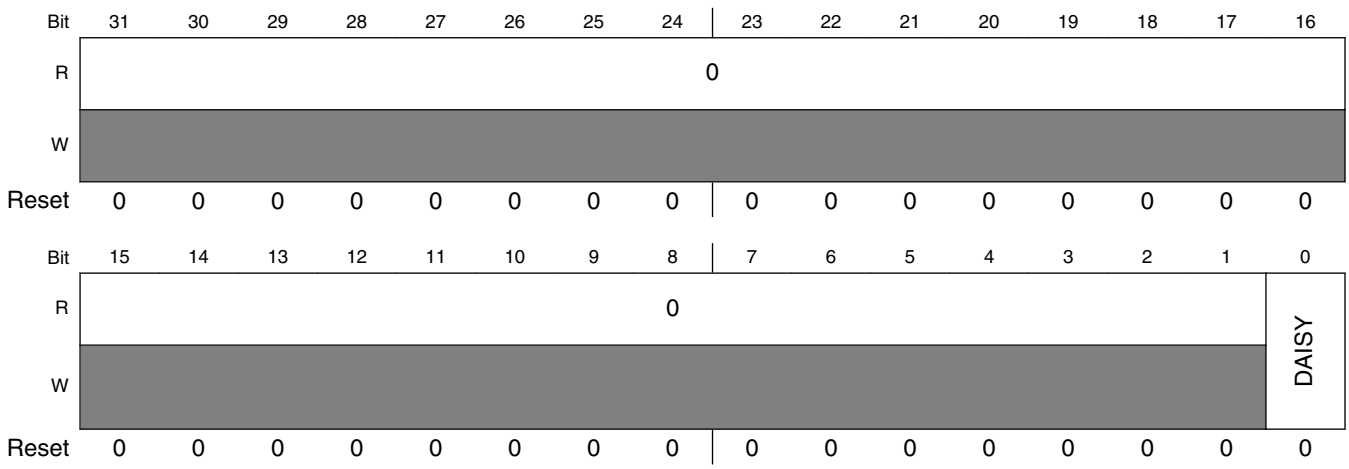


IOMUXC_ESAI_TX_FS_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_RX_DATA1_ALT2** — Selecting ALT2 mode of pad ENET_RXD1 for ESAI_TX_FS. 1 **GPIO02_ALT0** — Selecting ALT0 mode of pad GPIO_2 for ESAI_TX_FS.

36.4.532 Select Input Register (IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT)

Address: 20E_0000h base + 864h offset = 20E_0864h



IOMUXC_ESAI_RX_HF_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_RX_ER_ALT2** — Selecting ALT2 mode of pad ENET_RX_ER for ESAI_RX_HF_CLK. 1 **GPIO03_ALT0** — Selecting ALT0 mode of pad GPIO_3 for ESAI_RX_HF_CLK.

36.4.533 Select Input Register (IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT)

Address: 20E_0000h base + 868h offset = 20E_0868h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

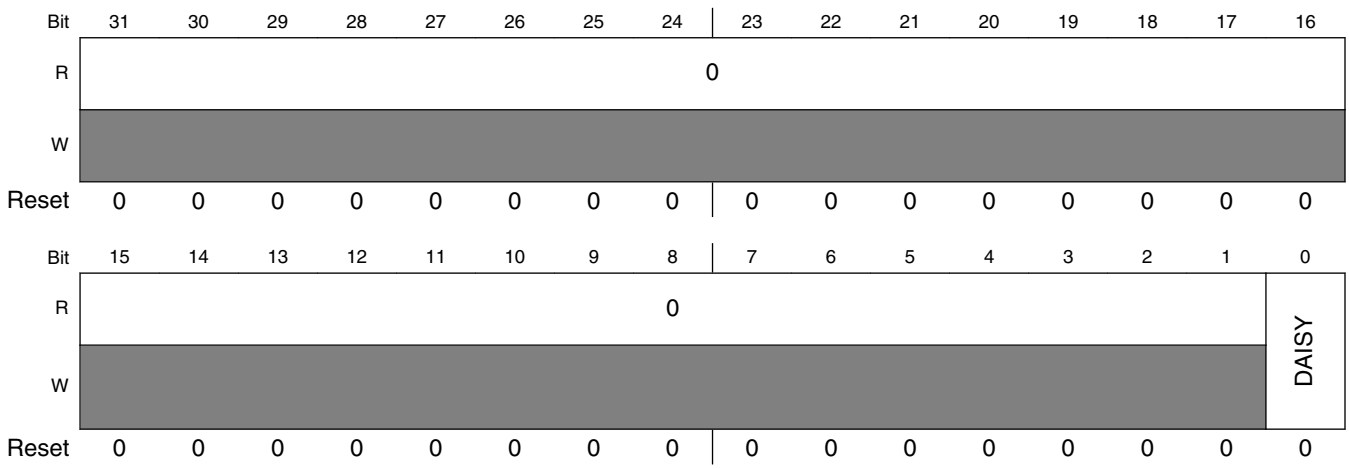
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ESAI_TX_HF_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_RX_DATA0_ALT2** — Selecting ALT2 mode of pad ENET_RXD0 for ESAI_TX_HF_CLK. 1 **GPIO04_ALT0** — Selecting ALT0 mode of pad GPIO_4 for ESAI_TX_HF_CLK.

36.4.534 Select Input Register (IOMUXC_ESAI_RX_CLK_SELECT_INPUT)

Address: 20E_0000h base + 86Ch offset = 20E_086Ch

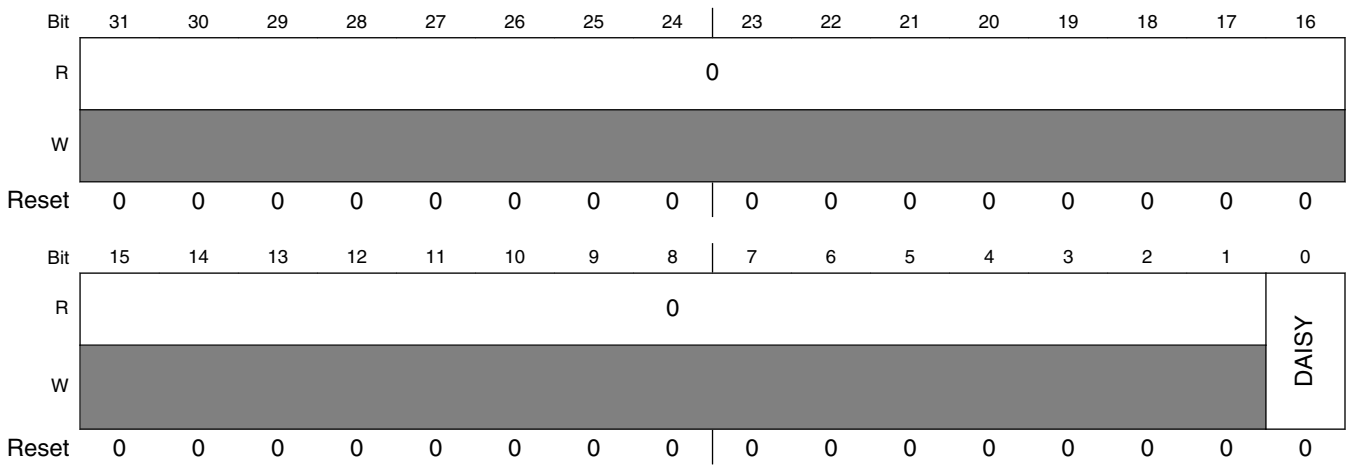


IOMUXC_ESAI_RX_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_MDIO_ALT2** — Selecting ALT2 mode of pad ENET_MDIO for ESAI_RX_CLK. 1 **GPIO01_ALT0** — Selecting ALT0 mode of pad GPIO_1 for ESAI_RX_CLK.

36.4.535 Select Input Register (IOMUXC_ESAI_TX_CLK_SELECT_INPUT)

Address: 20E_0000h base + 870h offset = 20E_0870h

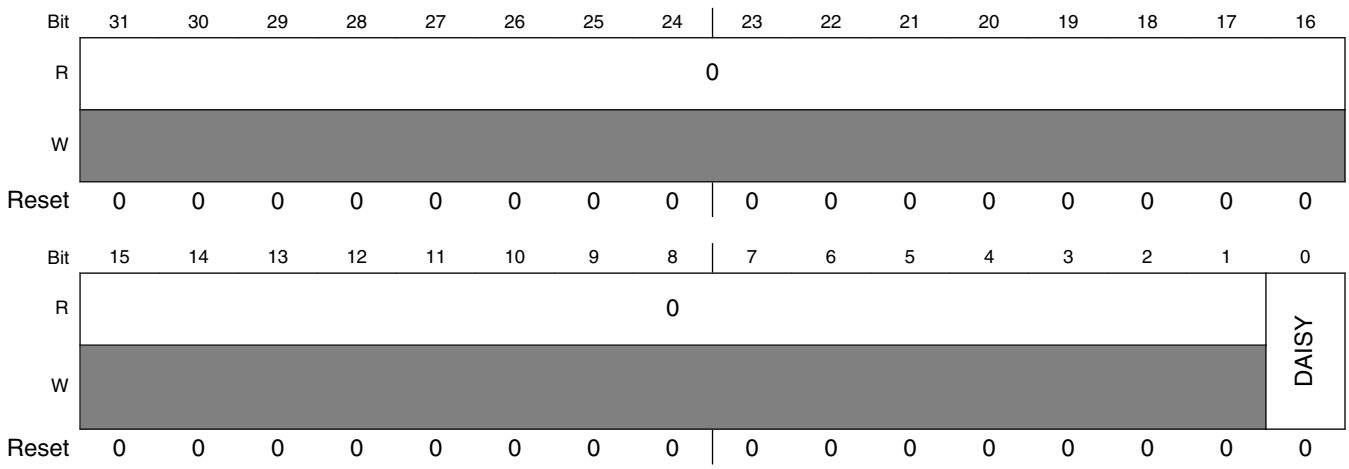


IOMUXC_ESAI_TX_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_CRS_DV_ALT2** — Selecting ALT2 mode of pad ENET_CRS_DV for ESAI_TX_CLK. 1 **GPIO06_ALT0** — Selecting ALT0 mode of pad GPIO_6 for ESAI_TX_CLK.

36.4.536 Select Input Register (IOMUXC_ESAI_SDO0_SELECT_INPUT)

Address: 20E_0000h base + 874h offset = 20E_0874h



IOMUXC_ESAI_SDO0_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **GPIO17_ALT0** — Selecting ALT0 mode of pad GPIO_17 for ESAI_TX0. 1 **NAND_CS2_B_ALT2** — Selecting ALT2 mode of pad NANDF_CS2 for ESAI_TX0.

36.4.537 Select Input Register (IOMUXC_ESAI_SDO1_SELECT_INPUT)

Address: 20E_0000h base + 878h offset = 20E_0878h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

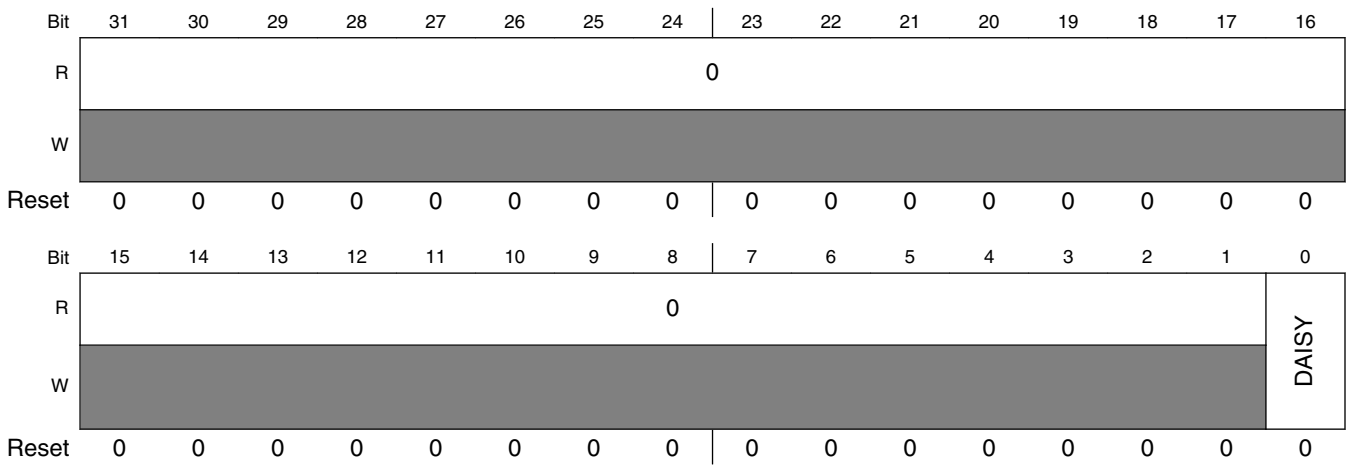
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ESAI_SDO1_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **GPIO18_ALT0** — Selecting ALT0 mode of pad GPIO_18 for ESAI_TX1. 1 **NAND_CS3_B_ALT2** — Selecting ALT2 mode of pad NANDF_CS3 for ESAI_TX1.

36.4.538 Select Input Register (IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT)

Address: 20E_0000h base + 87Ch offset = 20E_087Ch



IOMUXC_ESAI_SDO2_SDI3_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_TX_DATA1_ALT2** — Selecting ALT2 mode of pad ENET_TXD1 for ESAI_TX2_RX3. 1 **GPIO05_ALT0** — Selecting ALT0 mode of pad GPIO_5 for ESAI_TX2_RX3.

36.4.539 Select Input Register (IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT)

Address: 20E_0000h base + 880h offset = 20E_0880h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

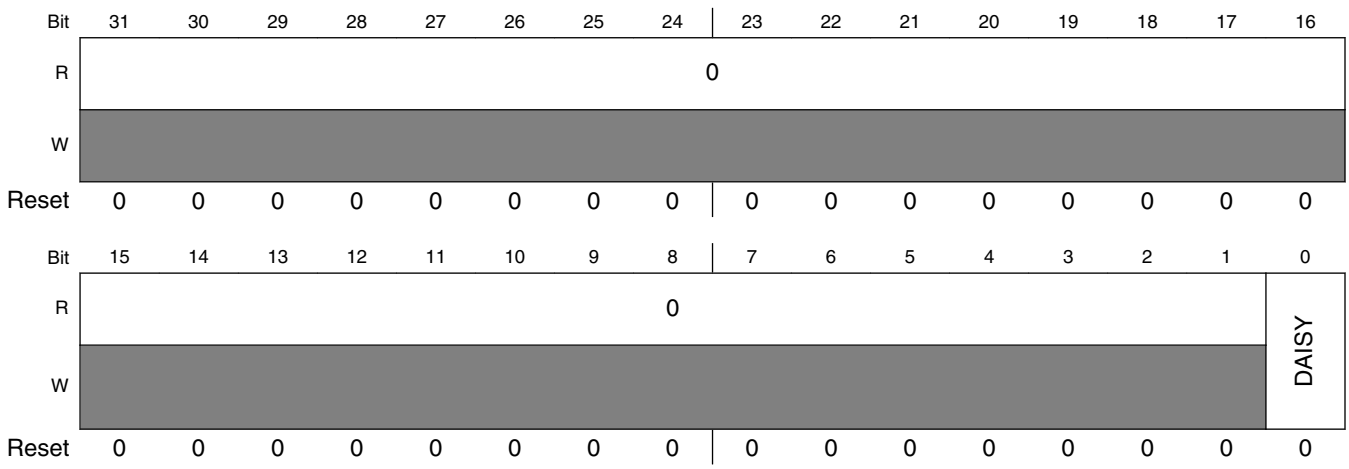
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ESAI_SDO3_SDI2_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_TX_EN_ALT2** — Selecting ALT2 mode of pad ENET_TX_EN for ESAI_TX3_RX2. 1 **GPIO16_ALT0** — Selecting ALT0 mode of pad GPIO_16 for ESAI_TX3_RX2.

36.4.540 Select Input Register (IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT)

Address: 20E_0000h base + 884h offset = 20E_0884h



IOMUXC_ESAI_SDO4_SDI1_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_TX_DATA0_ALT2** — Selecting ALT2 mode of pad ENET_TXD0 for ESAI_TX4_RX1. 1 **GPIO07_ALT0** — Selecting ALT0 mode of pad GPIO_7 for ESAI_TX4_RX1.

36.4.541 Select Input Register (IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT)

Address: 20E_0000h base + 888h offset = 20E_0888h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

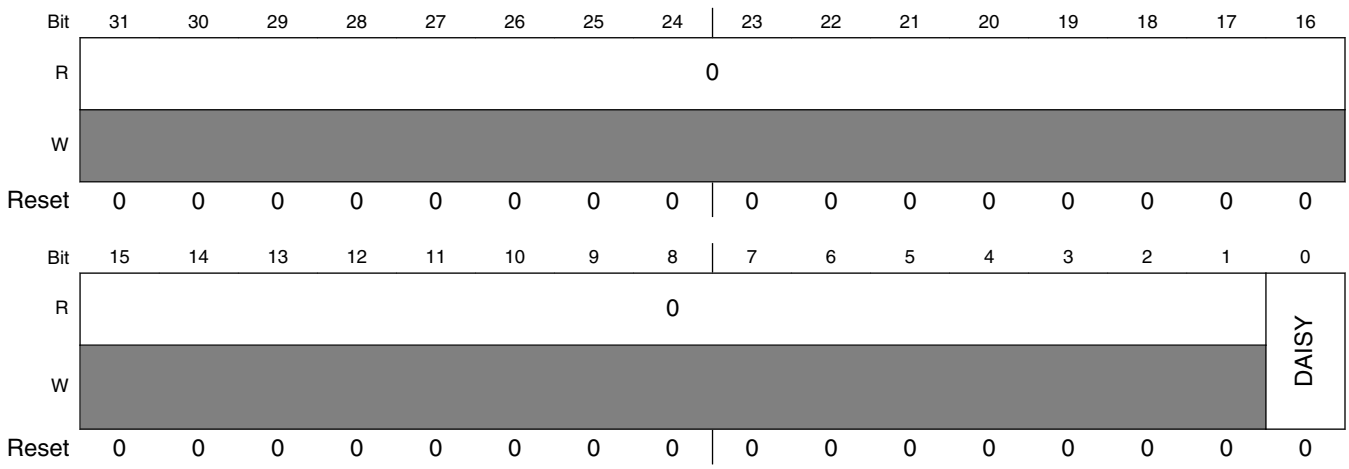
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_ESAI_SDO5_SDI0_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_MDC_ALT2** — Selecting ALT2 mode of pad ENET_MDC for ESAI_TX5_RX0. 1 **GPIO08_ALT0** — Selecting ALT0 mode of pad GPIO_8 for ESAI_TX5_RX0.

36.4.542 Select Input Register (IOMUXC_HDMI_ICECIN_SELECT_INPUT)

Address: 20E_0000h base + 88Ch offset = 20E_088Ch

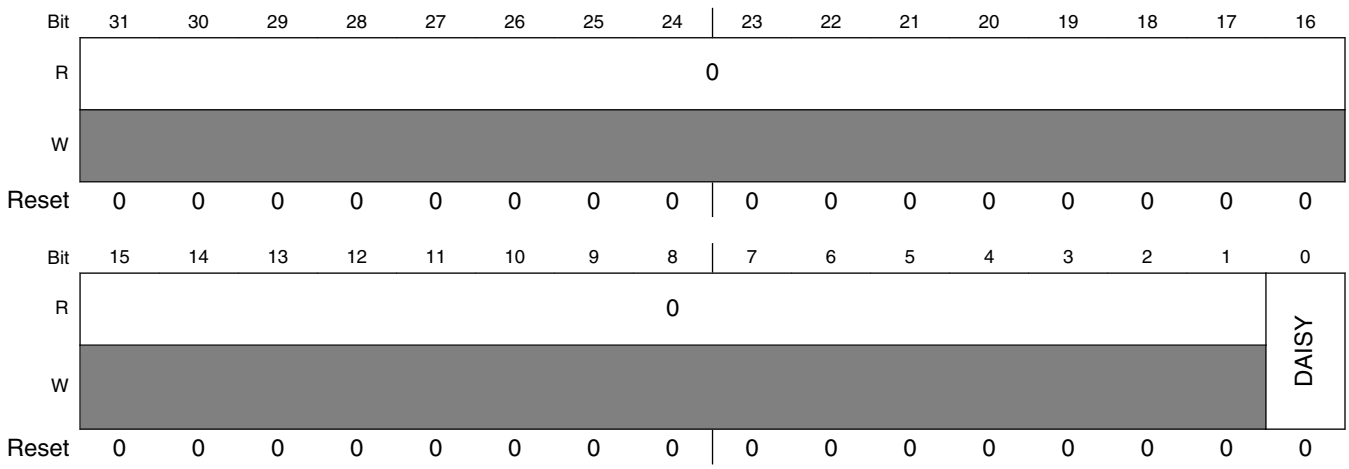


IOMUXC_HDMI_ICECIN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_ADDR25_ALT6** — Selecting ALT6 mode of pad EIM_A25 for HDMI_TX_CEC_LINE. 1 **KEY_ROW2_ALT6** — Selecting ALT6 mode of pad KEY_ROW2 for HDMI_TX_CEC_LINE.

36.4.543 Select Input Register (IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT)

Address: 20E_0000h base + 890h offset = 20E_0890h

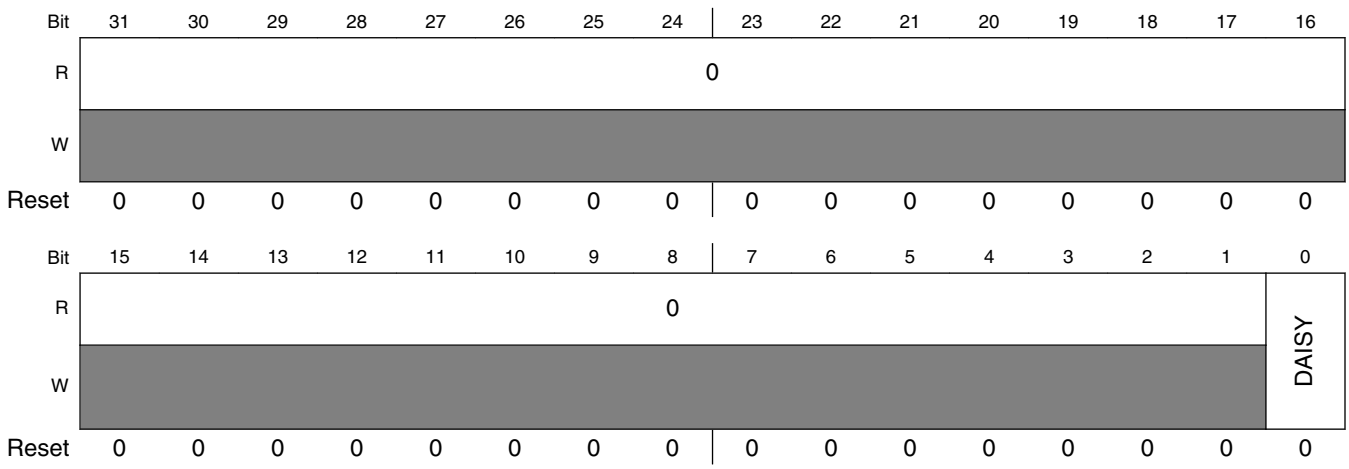


IOMUXC_HDMI_II2C_CLKIN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_EB2_B_ALT4** — Selecting ALT4 mode of pad EIM_EB2 for HDMI_TX_DDC_SCL. 1 **KEY_COL3_ALT2** — Selecting ALT2 mode of pad KEY_COL3 for HDMI_TX_DDC_SCL.

36.4.544 Select Input Register (IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT)

Address: 20E_0000h base + 894h offset = 20E_0894h

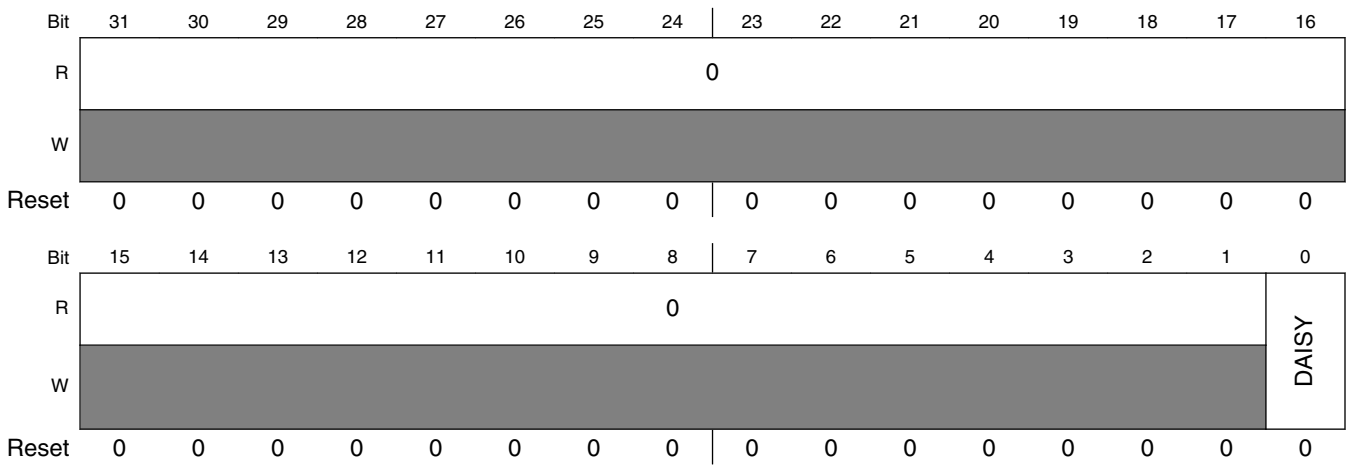


IOMUXC_HDMI_II2C_DATAIN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA16_ALT4** — Selecting ALT4 mode of pad EIM_D16 for HDMI_TX_DDC_SDA. 1 **KEY_ROW3_ALT2** — Selecting ALT2 mode of pad KEY_ROW3 for HDMI_TX_DDC_SDA.

36.4.545 Select Input Register (IOMUXC_I2C1_SCL_IN_SELECT_INPUT)

Address: 20E_0000h base + 898h offset = 20E_0898h

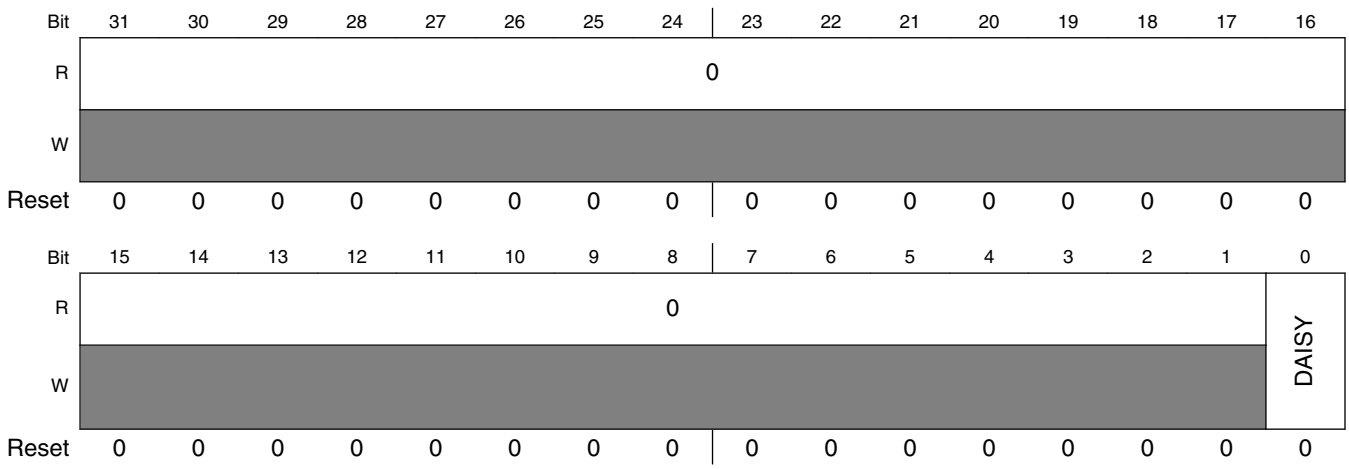


IOMUXC_I2C1_SCL_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA21_ALT6** — Selecting ALT6 mode of pad EIM_D21 for I2C1_SCL. 1 **CSI0_DATA09_ALT4** — Selecting ALT4 mode of pad CSI0_DAT9 for I2C1_SCL.

36.4.546 Select Input Register (IOMUXC_I2C1_SDA_IN_SELECT_INPUT)

Address: 20E_0000h base + 89Ch offset = 20E_089Ch

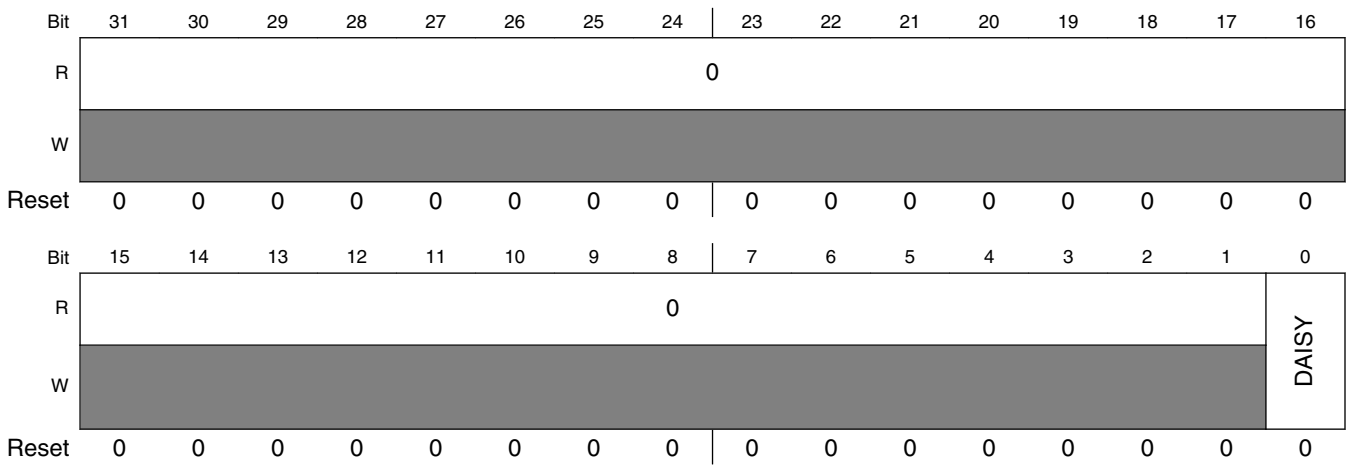


IOMUXC_I2C1_SDA_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA28_ALT1** — Selecting ALT1 mode of pad EIM_D28 for I2C1_SDA. 1 **CSI0_DATA08_ALT4** — Selecting ALT4 mode of pad CSI0_DAT8 for I2C1_SDA.

36.4.547 Select Input Register (IOMUXC_I2C2_SCL_IN_SELECT_INPUT)

Address: 20E_0000h base + 8A0h offset = 20E_08A0h

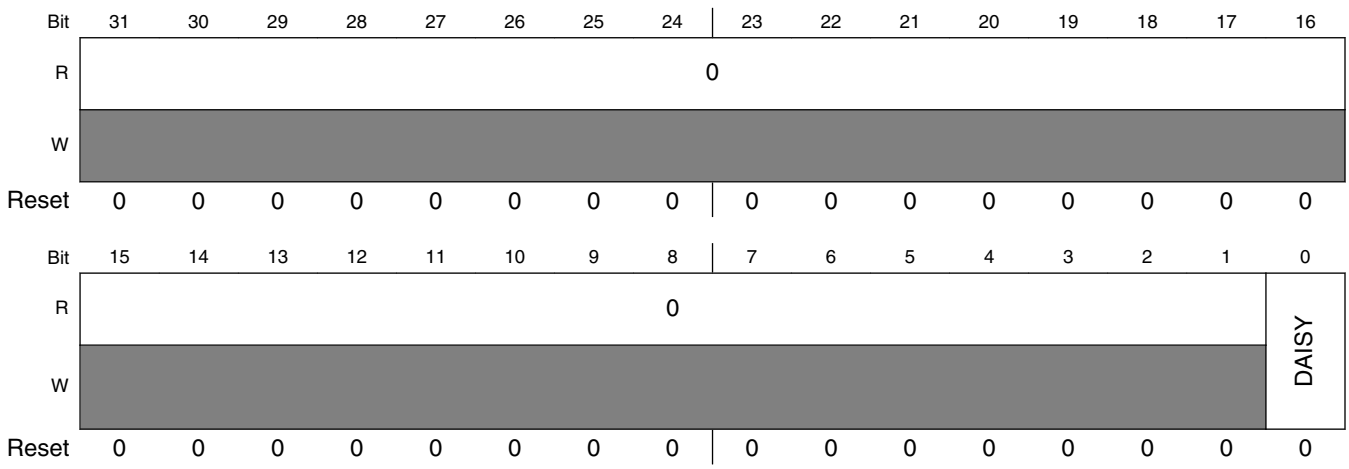


IOMUXC_I2C2_SCL_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_EB2_B_ALT6** — Selecting ALT6 mode of pad EIM_EB2 for I2C2_SCL. 1 **KEY_COL3_ALT4** — Selecting ALT4 mode of pad KEY_COL3 for I2C2_SCL.

36.4.548 Select Input Register (IOMUXC_I2C2_SDA_IN_SELECT_INPUT)

Address: 20E_0000h base + 8A4h offset = 20E_08A4h

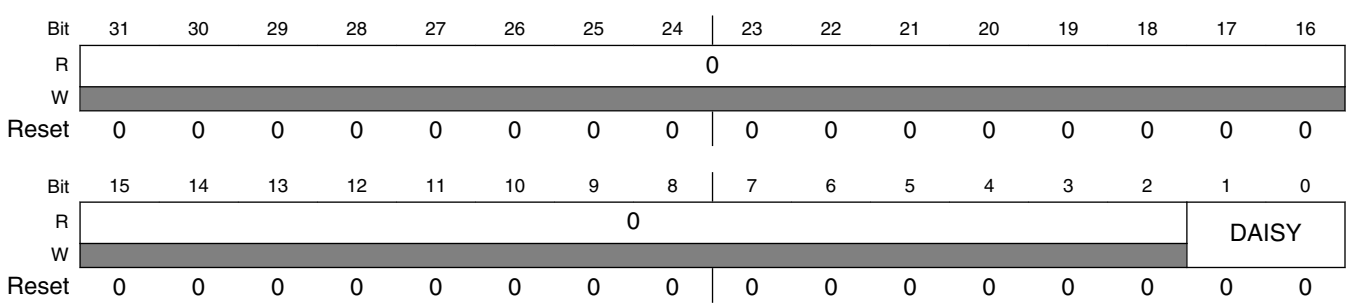


IOMUXC_I2C2_SDA_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA16_ALT6** — Selecting ALT6 mode of pad EIM_D16 for I2C2_SDA. 1 **KEY_ROW3_ALT4** — Selecting ALT4 mode of pad KEY_ROW3 for I2C2_SDA.

36.4.549 Select Input Register (IOMUXC_I2C3_SCL_IN_SELECT_INPUT)

Address: 20E_0000h base + 8A8h offset = 20E_08A8h



IOMUXC_I2C3_SCL_IN_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA17_ALT6** — Selecting ALT6 mode of pad EIM_D17 for I2C3_SCL. 01 **GPIO03_ALT2** — Selecting ALT2 mode of pad GPIO_3 for I2C3_SCL. 10 **GPIO05_ALT6** — Selecting ALT6 mode of pad GPIO_5 for I2C3_SCL.

36.4.550 Select Input Register (IOMUXC_I2C3_SDA_IN_SELECT_INPUT)

Address: 20E_0000h base + 8ACh offset = 20E_08ACh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

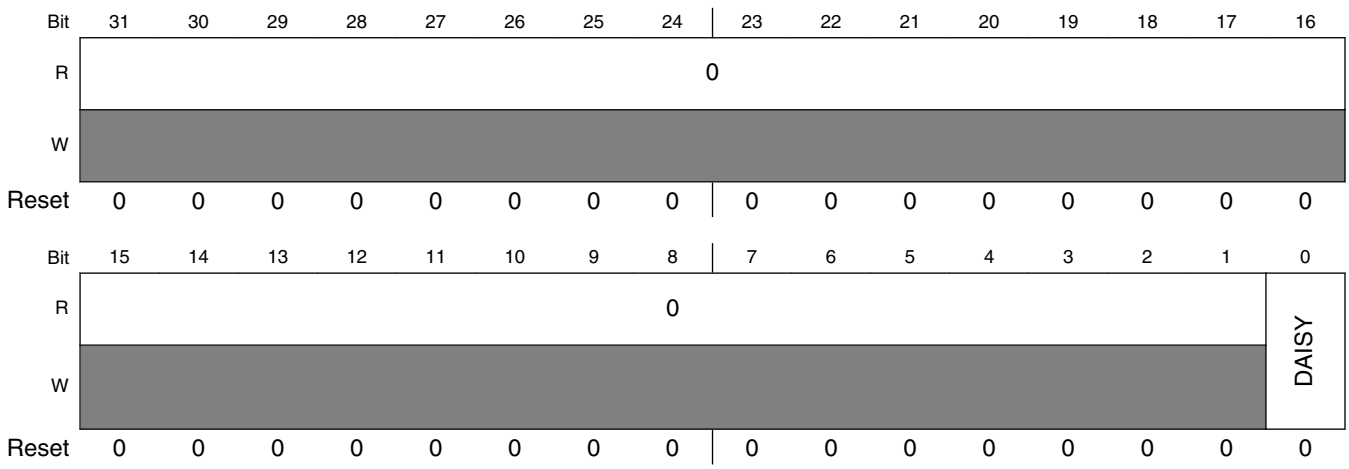
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_I2C3_SDA_IN_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA18_ALT6** — Selecting ALT6 mode of pad EIM_D18 for I2C3_SDA. 01 **GPIO06_ALT2** — Selecting ALT2 mode of pad GPIO_6 for I2C3_SDA. 10 **GPIO16_ALT6** — Selecting ALT6 mode of pad GPIO_16 for I2C3_SDA.

36.4.551 Select Input Register (IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT)

Address: 20E_0000h base + 8B0h offset = 20E_08B0h



IOMUXC_IPU2_SENS1_DATA10_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA22_ALT3** — Selecting ALT3 mode of pad EIM_D22 for IPU2_CSI1_DATA10. 1 **EIM_EB1_B_ALT2** — Selecting ALT2 mode of pad EIM_EB1 for IPU2_CSI1_DATA10.

36.4.552 Select Input Register (IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT)

Address: 20E_0000h base + 8B4h offset = 20E_08B4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

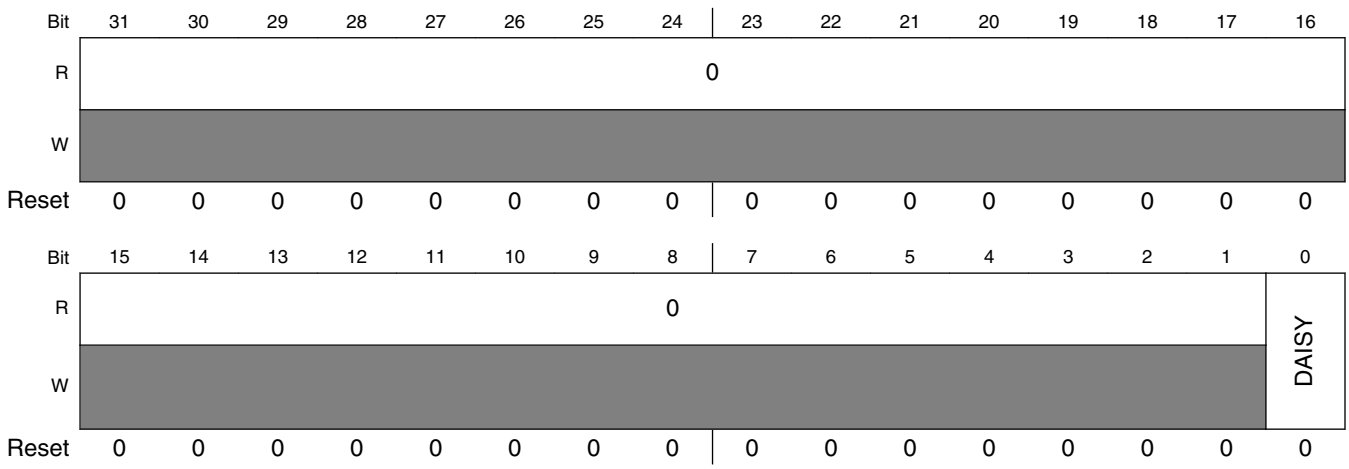
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_DATA11_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA21_ALT3** — Selecting ALT3 mode of pad EIM_D21 for IPU2_CSI1_DATA11. 1 **EIM_EB0_B_ALT2** — Selecting ALT2 mode of pad EIM_EB0 for IPU2_CSI1_DATA11.

36.4.553 Select Input Register (IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT)

Address: 20E_0000h base + 8B8h offset = 20E_08B8h



IOMUXC_IPU2_SENS1_DATA12_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA28_ALT3** — Selecting ALT3 mode of pad EIM_D28 for IPU2_CSI1_DATA12. 1 **EIM_ADDR17_ALT2** — Selecting ALT2 mode of pad EIM_A17 for IPU2_CSI1_DATA12.

36.4.554 Select Input Register (IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT)

Address: 20E_0000h base + 8BCh offset = 20E_08BCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

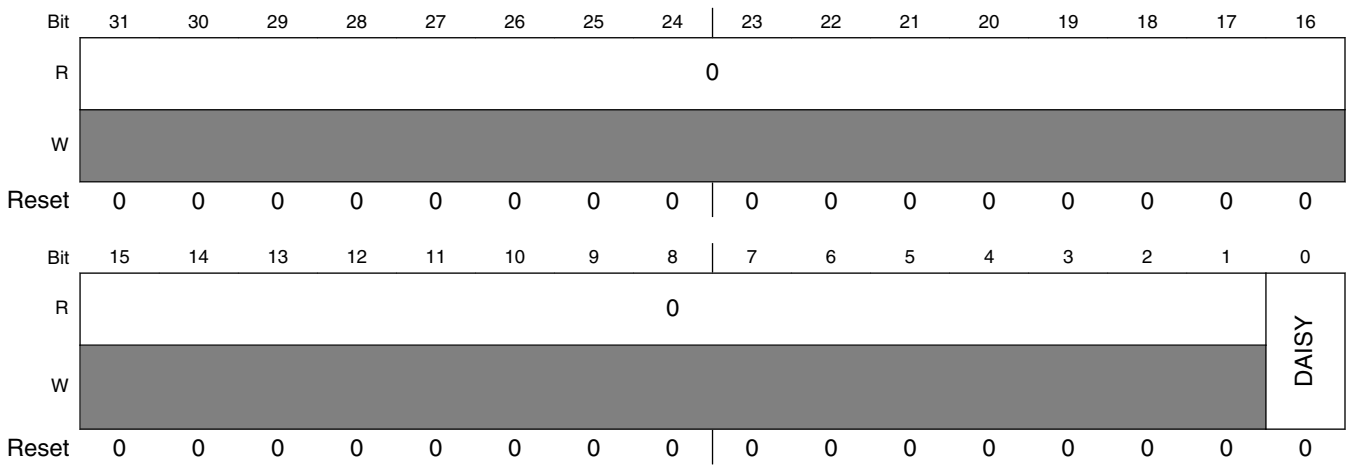
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_DATA13_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA27_ALT3** — Selecting ALT3 mode of pad EIM_D27 for IPU2_CSI1_DATA13. 1 **EIM_ADDR18_ALT2** — Selecting ALT2 mode of pad EIM_A18 for IPU2_CSI1_DATA13.

36.4.555 Select Input Register (IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT)

Address: 20E_0000h base + 8C0h offset = 20E_08C0h



IOMUXC_IPU2_SENS1_DATA14_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA26_ALT3** — Selecting ALT3 mode of pad EIM_D26 for IPU2_CSI1_DATA14. 1 **EIM_ADDR19_ALT2** — Selecting ALT2 mode of pad EIM_A19 for IPU2_CSI1_DATA14.

36.4.556 Select Input Register (IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT)

Address: 20E_0000h base + 8C4h offset = 20E_08C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

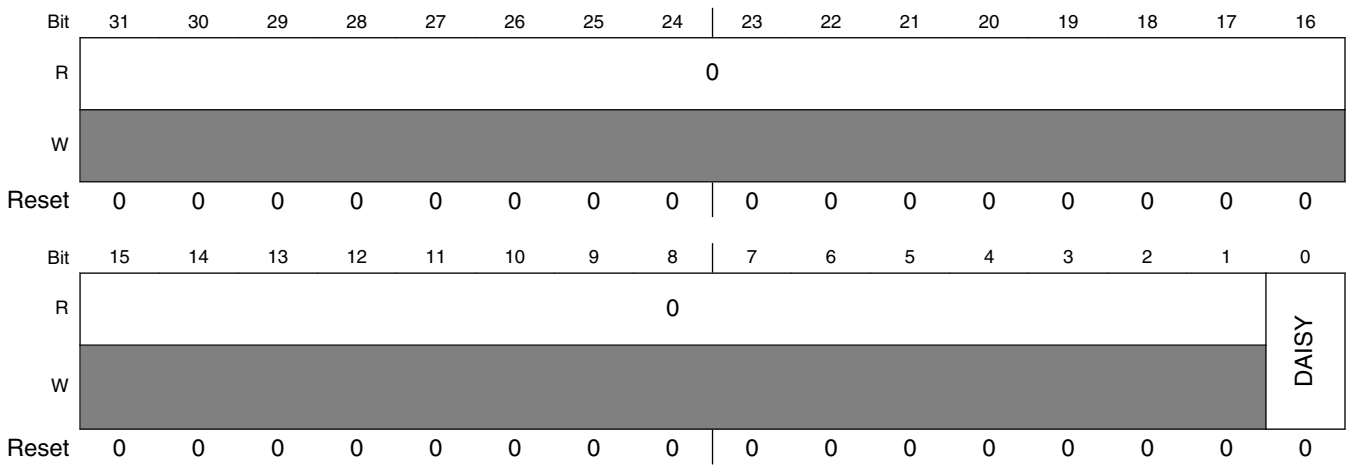
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_DATA15_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA20_ALT3** — Selecting ALT3 mode of pad EIM_D20 for IPU2_CSI1_DATA15. 1 **EIM_ADDR20_ALT2** — Selecting ALT2 mode of pad EIM_A20 for IPU2_CSI1_DATA15.

36.4.557 Select Input Register (IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT)

Address: 20E_0000h base + 8C8h offset = 20E_08C8h



IOMUXC_IPU2_SENS1_DATA16_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA19_ALT3** — Selecting ALT3 mode of pad EIM_D19 for IPU2_CSI1_DATA16. 1 **EIM_ADDR21_ALT2** — Selecting ALT2 mode of pad EIM_A21 for IPU2_CSI1_DATA16.

36.4.558 Select Input Register (IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT)

Address: 20E_0000h base + 8CCh offset = 20E_08CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

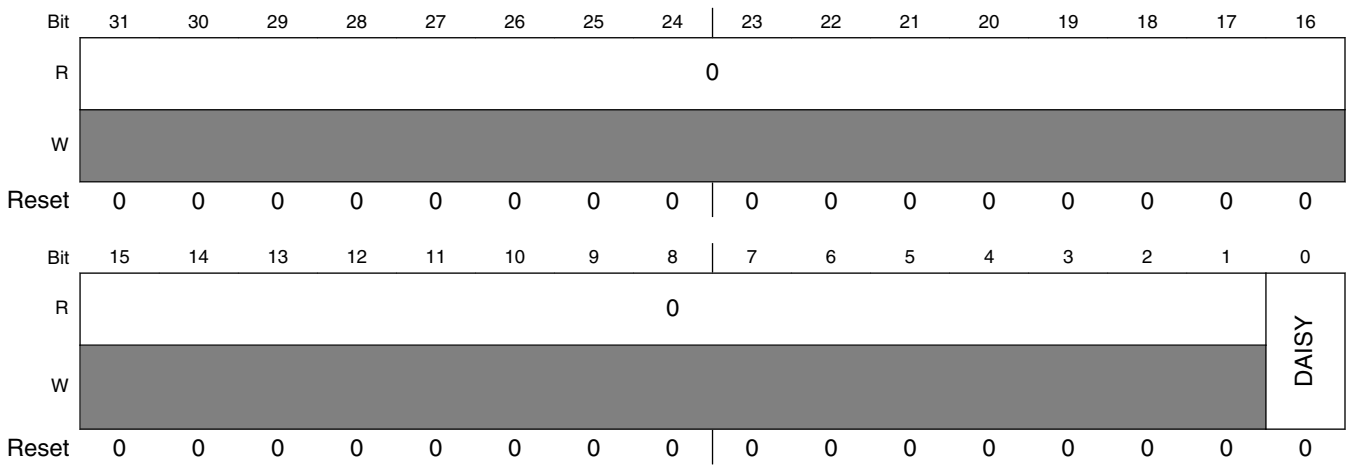
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_DATA17_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA18_ALT3** — Selecting ALT3 mode of pad EIM_D18 for IPU2_CSI1_DATA17. 1 **EIM_ADDR22_ALT2** — Selecting ALT2 mode of pad EIM_A22 for IPU2_CSI1_DATA17.

36.4.559 Select Input Register (IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT)

Address: 20E_0000h base + 8D0h offset = 20E_08D0h



IOMUXC_IPU2_SENS1_DATA18_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA16_ALT3** — Selecting ALT3 mode of pad EIM_D16 for IPU2_CSI1_DATA18. 1 **EIM_ADDR23_ALT2** — Selecting ALT2 mode of pad EIM_A23 for IPU2_CSI1_DATA18.

36.4.560 Select Input Register (IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT)

Address: 20E_0000h base + 8D4h offset = 20E_08D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

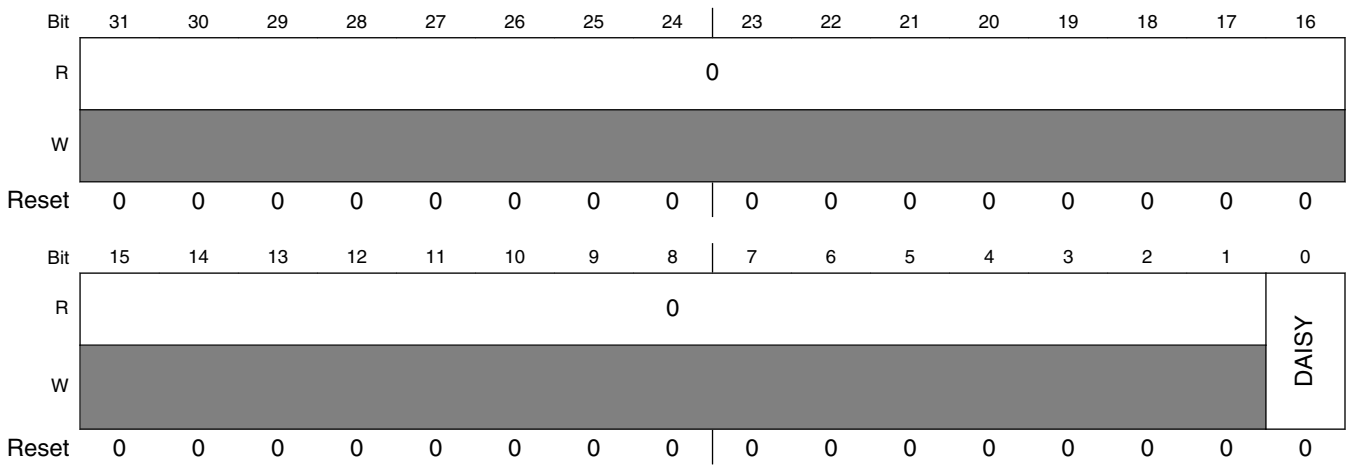
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_DATA19_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_EB2_B_ALT3** — Selecting ALT3 mode of pad EIM_EB2 for IPU2_CSI1_DATA19. 1 **EIM_ADDR24_ALT2** — Selecting ALT2 mode of pad EIM_A24 for IPU2_CSI1_DATA19.

36.4.561 Select Input Register (IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT)

Address: 20E_0000h base + 8D8h offset = 20E_08D8h

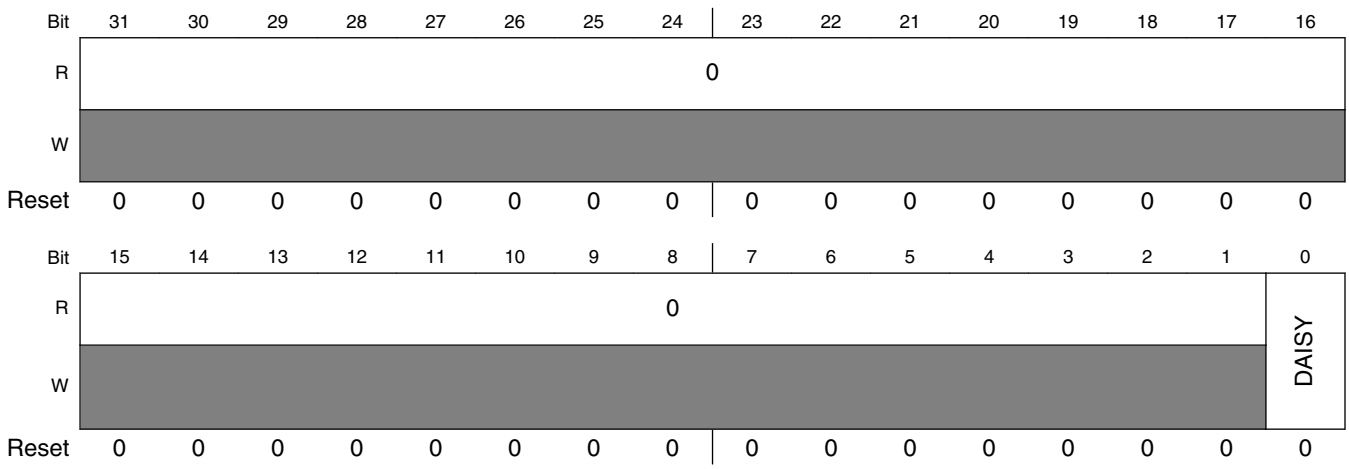


IOMUXC_IPU2_SENS1_DATA_EN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA23_ALT4** — Selecting ALT4 mode of pad EIM_D23 for IPU2_CSI1_DATA_EN. 1 **EIM_AD10_ALT2** — Selecting ALT2 mode of pad EIM_DA10 for IPU2_CSI1_DATA_EN.

36.4.562 Select Input Register (IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT)

Address: 20E_0000h base + 8DCh offset = 20E_08DCh

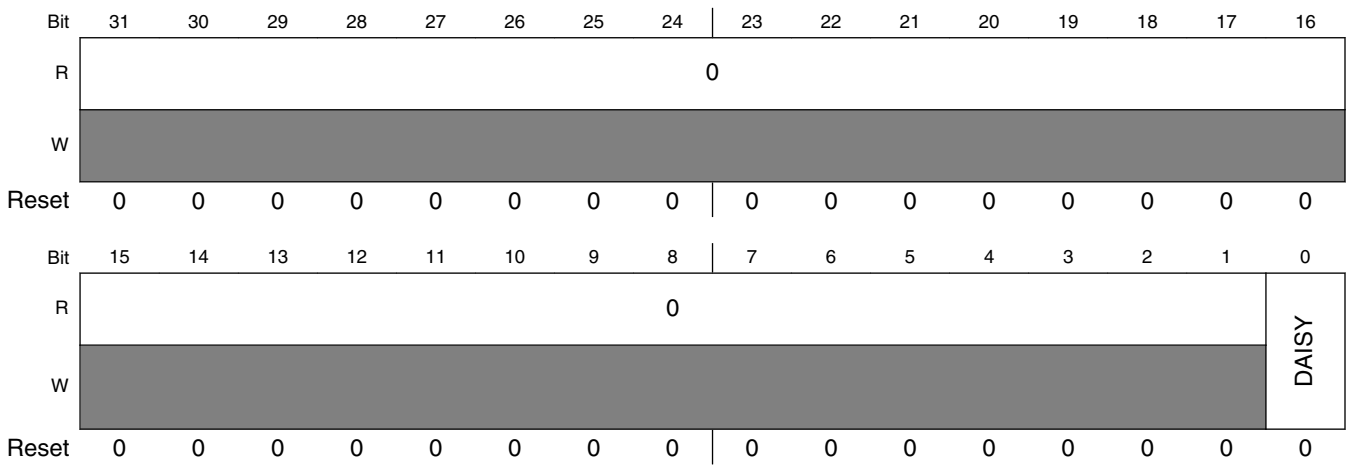


IOMUXC_IPU2_SENS1_HSYNC_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_EB3_B_ALT4** — Selecting ALT4 mode of pad EIM_EB3 for IPU2_CSI1_HSYNC. 1 **EIM_AD11_ALT2** — Selecting ALT2 mode of pad EIM_DA11 for IPU2_CSI1_HSYNC.

36.4.563 Select Input Register (IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT)

Address: 20E_0000h base + 8E0h offset = 20E_08E0h



IOMUXC_IPU2_SENS1_PIX_CLK_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA17_ALT3** — Selecting ALT3 mode of pad EIM_D17 for IPU2_CSI1_PIXCLK. 1 **EIM_ADDR16_ALT2** — Selecting ALT2 mode of pad EIM_A16 for IPU2_CSI1_PIXCLK.

36.4.564 Select Input Register (IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT)

Address: 20E_0000h base + 8E4h offset = 20E_08E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_IPU2_SENS1_VSYNC_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA29_ALT6** — Selecting ALT6 mode of pad EIM_D29 for IPU2_CSI1_VSYNC. 1 **EIM_AD12_ALT2** — Selecting ALT2 mode of pad EIM_DA12 for IPU2_CSI1_VSYNC.

36.4.565 Select Input Register (IOMUXC_KEY_COL5_SELECT_INPUT)

Address: 20E_0000h base + 8E8h offset = 20E_08E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_KEY_COL5_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **GPIO00_ALT2** — Selecting ALT2 mode of pad GPIO_0 for KEY_COL5. 01 **GPIO19_ALT0** — Selecting ALT0 mode of pad GPIO_19 for KEY_COL5. 10 **CSI0_DATA04_ALT3** — Selecting ALT3 mode of pad CSI0_DAT4 for KEY_COL5. 11 **SD2_CLK_ALT2** — Selecting ALT2 mode of pad SD2_CLK for KEY_COL5.

36.4.566 Select Input Register (IOMUXC_KEY_COL6_SELECT_INPUT)

Address: 20E_0000h base + 8ECh offset = 20E_08ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_KEY_COL6_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **GPIO09_ALT2** — Selecting ALT2 mode of pad GPIO_9 for KEY_COL6. 01 **CSI0_DATA06_ALT3** — Selecting ALT3 mode of pad CSI0_DAT6 for KEY_COL6. 10 **SD2_DATA3_ALT2** — Selecting ALT2 mode of pad SD2_DAT3 for KEY_COL6.

36.4.567 Select Input Register (IOMUXC_KEY_COL7_SELECT_INPUT)

Address: 20E_0000h base + 8F0h offset = 20E_08F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_KEY_COL7_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **SD2_DATA1_ALT4** — Selecting ALT4 mode of pad SD2_DAT1 for KEY_COL7. 01 **GPIO04_ALT2** — Selecting ALT2 mode of pad GPIO_4 for KEY_COL7. 10 **CSI0_DATA08_ALT3** — Selecting ALT3 mode of pad CSI0_DAT8 for KEY_COL7.

36.4.568 Select Input Register (IOMUXC_KEY_ROW5_SELECT_INPUT)

Address: 20E_0000h base + 8F4h offset = 20E_08F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_KEY_ROW5_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IOMUXC_KEY_ROW5_SELECT_INPUT field descriptions (continued)

Field	Description
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **GPIO01_ALT2** — Selecting ALT2 mode of pad GPIO_1 for KEY_ROW5. 01 **CSI0_DATA05_ALT3** — Selecting ALT3 mode of pad CSI0_DAT5 for KEY_ROW5. 10 **SD2_CMD_ALT2** — Selecting ALT2 mode of pad SD2_CMD for KEY_ROW5.

36.4.569 Select Input Register (IOMUXC_KEY_ROW6_SELECT_INPUT)

Address: 20E_0000h base + 8F8h offset = 20E_08F8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

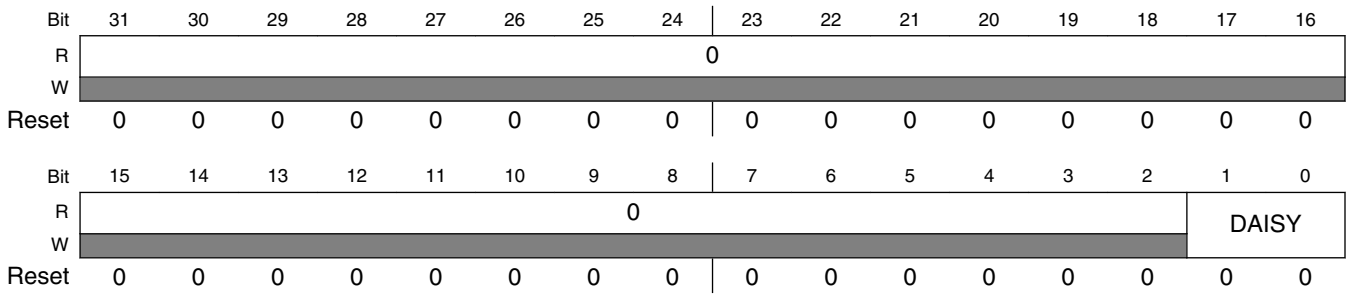
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_KEY_ROW6_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **SD2_DATA2_ALT4** — Selecting ALT4 mode of pad SD2_DAT2 for KEY_ROW6. 01 **GPIO02_ALT2** — Selecting ALT2 mode of pad GPIO_2 for KEY_ROW6. 10 **CSI0_DATA07_ALT3** — Selecting ALT3 mode of pad CSI0_DAT7 for KEY_ROW6.

36.4.570 Select Input Register (IOMUXC_KEY_ROW7_SELECT_INPUT)

Address: 20E_0000h base + 8FCh offset = 20E_08FCh

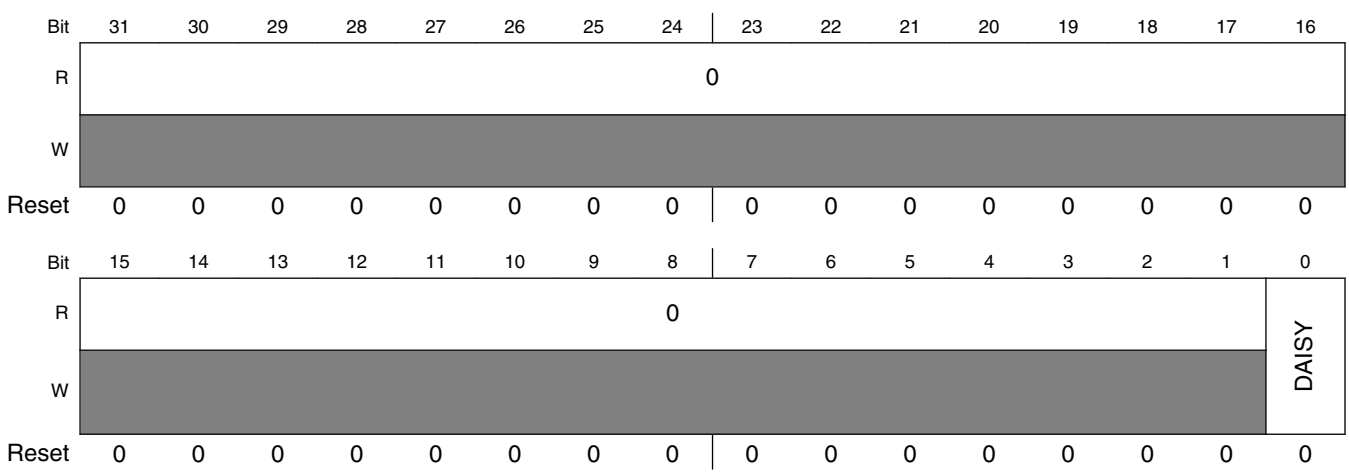


IOMUXC_KEY_ROW7_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **SD2_DATA0_ALT4** — Selecting ALT4 mode of pad SD2_DAT0 for KEY_ROW7. 01 **GPIO05_ALT2** — Selecting ALT2 mode of pad GPIO_5 for KEY_ROW7. 10 **CSI0_DATA09_ALT3** — Selecting ALT3 mode of pad CSI0_DAT9 for KEY_ROW7.

36.4.571 Select Input Register (IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT)

Address: 20E_0000h base + 900h offset = 20E_0900h

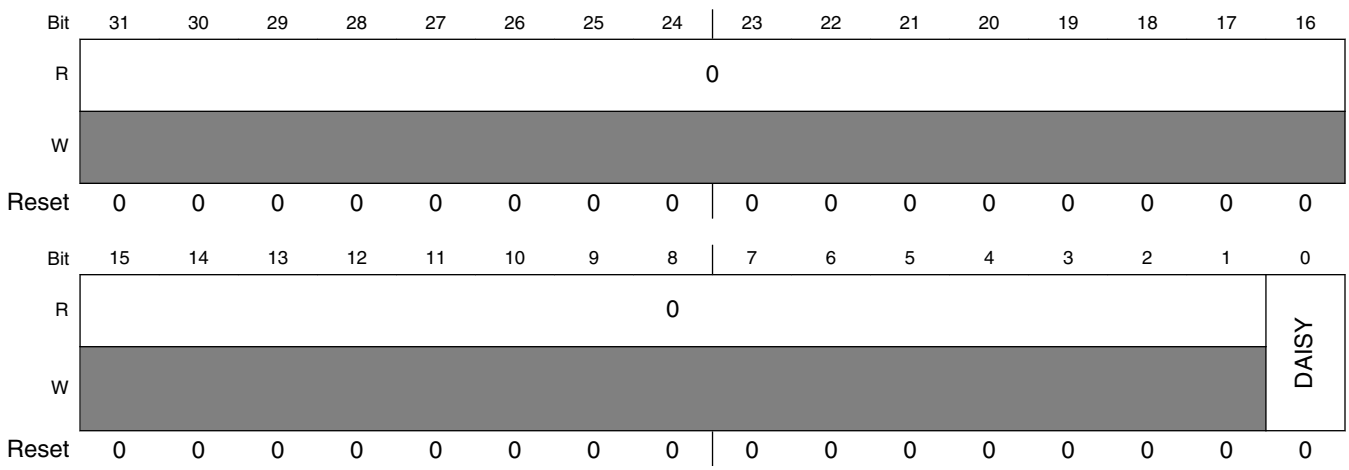


IOMUXC_MLB_MLB_CLK_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_TX_DATA1_ALT0** — Selecting ALT0 mode of pad ENET_TXD1 for MLB_CLK. 1 **GPIO03_ALT7** — Selecting ALT7 mode of pad GPIO_3 for MLB_CLK.

36.4.572 Select Input Register (IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT)

Address: 20E_0000h base + 904h offset = 20E_0904h



IOMUXC_MLB_MLB_DATA_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_MDC_ALT0** — Selecting ALT0 mode of pad ENET_MDC for MLB_DATA. 1 **GPIO02_ALT7** — Selecting ALT7 mode of pad GPIO_2 for MLB_DATA.

36.4.573 Select Input Register (IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT)

Address: 20E_0000h base + 908h offset = 20E_0908h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

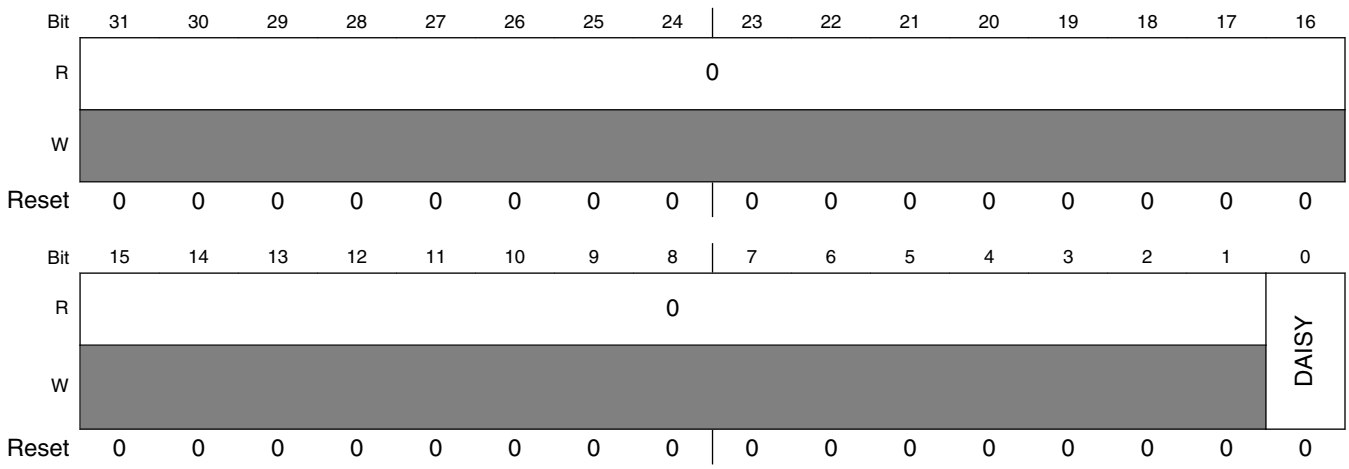
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_MLB_MLB_SIG_IN_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **ENET_RX_DATA1_ALT0** — Selecting ALT0 mode of pad ENET_RXD1 for MLB_SIG. 1 **GPIO06_ALT7** — Selecting ALT7 mode of pad GPIO_6 for MLB_SIG.

36.4.574 Select Input Register (IOMUXC_SDMA_EVENTS14_SELECT_INPUT)

Address: 20E_0000h base + 90Ch offset = 20E_090Ch

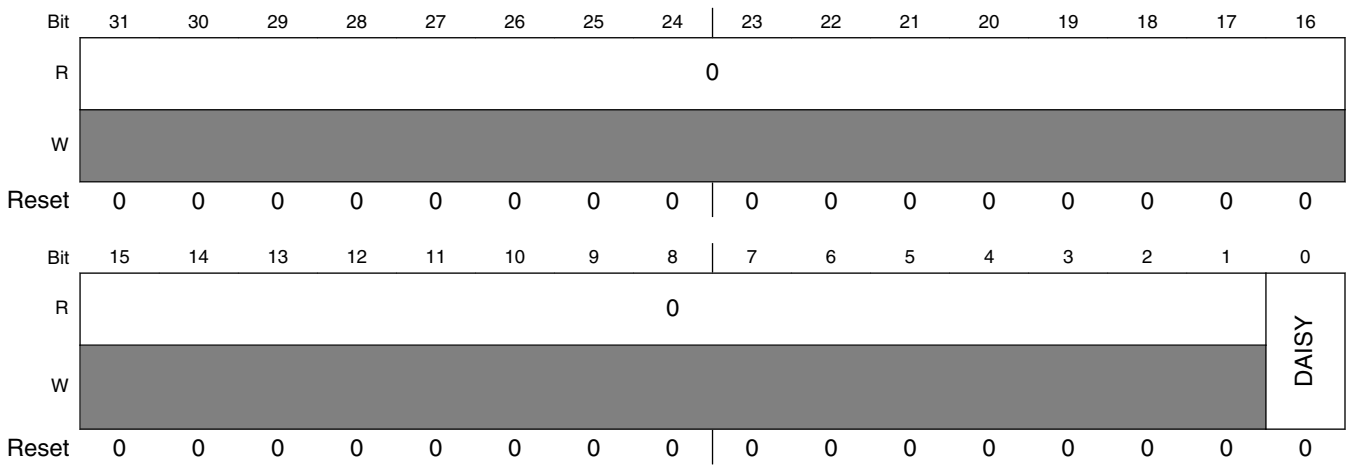


IOMUXC_SDMA_EVENTS14_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA16_ALT4** — Selecting ALT4 mode of pad DISP0_DAT16 for SDMA_EXT_EVENT0. 1 **GPIO17_ALT3** — Selecting ALT3 mode of pad GPIO_17 for SDMA_EXT_EVENT0.

36.4.575 Select Input Register (IOMUXC_SDMA_EVENTS15_SELECT_INPUT)

Address: 20E_0000h base + 910h offset = 20E_0910h

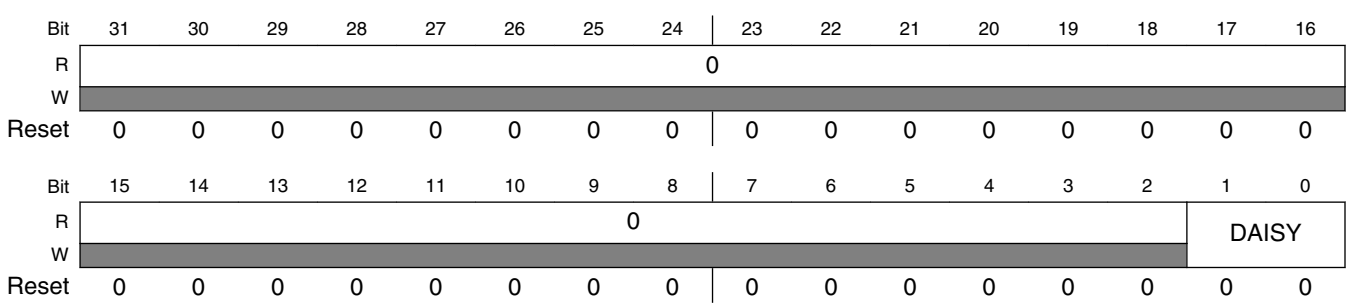


IOMUXC_SDMA_EVENTS15_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DISP0_DATA17_ALT4** — Selecting ALT4 mode of pad DISP0_DAT17 for SDMA_EXT_EVENT1. 1 **GPIO18_ALT3** — Selecting ALT3 mode of pad GPIO_18 for SDMA_EXT_EVENT1.

36.4.576 Select Input Register (IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT)

Address: 20E_0000h base + 914h offset = 20E_0914h

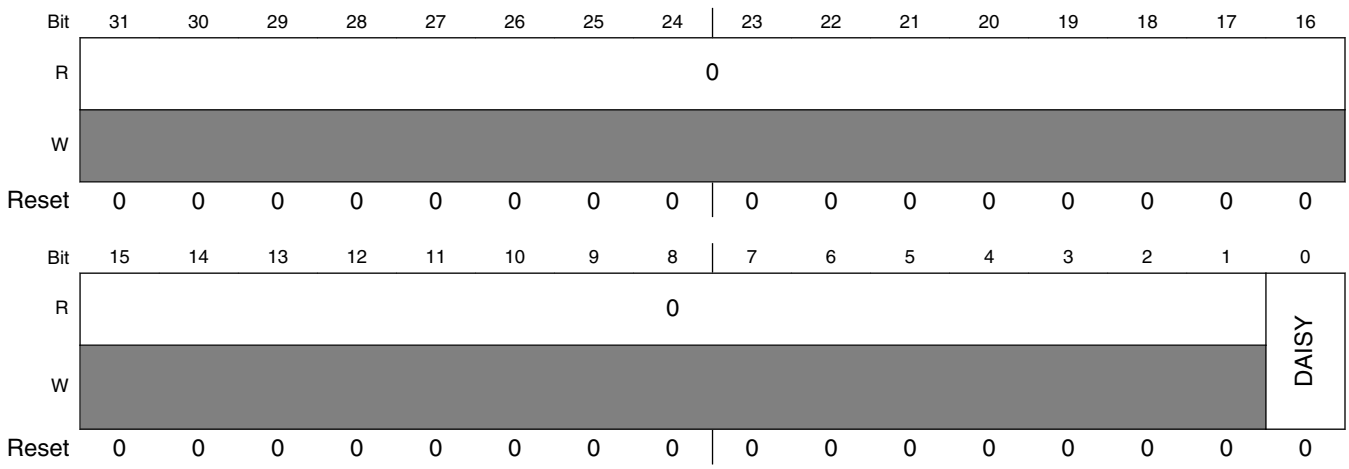


IOMUXC_SPDIF_SPDIF_IN1_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 00 **EIM_DATA21_ALT7** — Selecting ALT7 mode of pad EIM_D21 for SPDIF_IN. 01 **ENET_RX_ER_ALT3** — Selecting ALT3 mode of pad ENET_RX_ER for SPDIF_IN. 10 **KEY_COL3_ALT6** — Selecting ALT6 mode of pad KEY_COL3 for SPDIF_IN. 11 **GPIO16_ALT4** — Selecting ALT4 mode of pad GPIO_16 for SPDIF_IN.

36.4.577 Select Input Register (IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT)

Address: 20E_0000h base + 918h offset = 20E_0918h



IOMUXC_SPDIF_TX_CLK2_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **RGMII_TXC_ALT2** — Selecting ALT2 mode of pad RGMII_TXC for SPDIF_EXT_CLK. 1 **ENET_CRS_DV_ALT3** — Selecting ALT3 mode of pad ENET_CRS_DV for SPDIF_EXT_CLK.

36.4.578 Select Input Register (IOMUXC_UART1_UART_RTS_B_SELECT_INPUT)

Address: 20E_0000h base + 91Ch offset = 20E_091Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART1_UART_RTS_B_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RTS_B signal to pad when UART is in DCE mode. Route CTS_B signal to pad when UART is in DTE mode. 00 **EIM_DATA19_ALT4** — Selecting ALT4 mode of pad EIM_D19 for UART1_CTS_B. 01 **EIM_DATA20_ALT4** — Selecting ALT4 mode of pad EIM_D20 for UART1_RTS_B. 10 **SD3_DATA0_ALT1** — Selecting ALT1 mode of pad SD3_DAT0 for UART1_CTS_B. 11 **SD3_DATA1_ALT1** — Selecting ALT1 mode of pad SD3_DAT1 for UART1_RTS_B.

36.4.579 Select Input Register (IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT)

Address: 20E_0000h base + 920h offset = 20E_0920h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART1_UART_RX_DATA_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RX_DATA signal to pad when UART is in DCE mode. Route TX_DATA signal to pad when UART is in DTE mode. 00 **CSI0_DATA10_ALT3** — Selecting ALT3 mode of pad CSI0_DAT10 for UART1_TX_DATA. 01 **CSI0_DATA11_ALT3** — Selecting ALT3 mode of pad CSI0_DAT11 for UART1_RX_DATA. 10 **SD3_DATA7_ALT1** — Selecting ALT1 mode of pad SD3_DAT7 for UART1_TX_DATA. 11 **SD3_DATA6_ALT1** — Selecting ALT1 mode of pad SD3_DAT6 for UART1_RX_DATA.

36.4.580 Select Input Register (IOMUXC_UART2_UART_RTS_B_SELECT_INPUT)

Address: 20E_0000h base + 924h offset = 20E_0924h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																													DAISY		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART2_UART_RTS_B_SELECT_INPUT field descriptions

Field	Description
31–3 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RTS_B signal to pad when UART is in DCE mode. Route CTS_B signal to pad when UART is in DTE mode. 000 **EIM_DATA28_ALT4** — Selecting ALT4 mode of pad EIM_D28 for UART2_CTS_B. 001 **EIM_DATA29_ALT4** — Selecting ALT4 mode of pad EIM_D29 for UART2_RTS_B. 010 **SD3_CMD_ALT1** — Selecting ALT1 mode of pad SD3_CMD for UART2_CTS_B. 011 **SD3_CLK_ALT1** — Selecting ALT1 mode of pad SD3_CLK for UART2_RTS_B. 100 **SD4_DATA5_ALT2** — Selecting ALT2 mode of pad SD4_DAT5 for UART2_RTS_B. 101 **SD4_DATA6_ALT2** — Selecting ALT2 mode of pad SD4_DAT6 for UART2_CTS_B.

36.4.581 Select Input Register (IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT)

Address: 20E_0000h base + 928h offset = 20E_0928h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																													DAISY		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART2_UART_RX_DATA_SELECT_INPUT field descriptions

Field	Description
31–3 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RX_DATA signal to pad when UART is in DCE mode. Route TX_DATA signal to pad when UART is in DTE mode. 000 **EIM_DATA26_ALT4** — Selecting ALT4 mode of pad EIM_D26 for UART2_TX_DATA. 001 **EIM_DATA27_ALT4** — Selecting ALT4 mode of pad EIM_D27 for UART2_RX_DATA. 010 **GPIO07_ALT4** — Selecting ALT4 mode of pad GPIO_7 for UART2_TX_DATA. 011 **GPIO08_ALT4** — Selecting ALT4 mode of pad GPIO_8 for UART2_RX_DATA. 100 **SD3_DATA5_ALT1** — Selecting ALT1 mode of pad SD3_DAT5 for UART2_TX_DATA. 101 **SD3_DATA4_ALT1** — Selecting ALT1 mode of pad SD3_DAT4 for UART2_RX_DATA. 110 **SD4_DATA4_ALT2** — Selecting ALT2 mode of pad SD4_DAT4 for UART2_RX_DATA. 111 **SD4_DATA7_ALT2** — Selecting ALT2 mode of pad SD4_DAT7 for UART2_TX_DATA.

36.4.582 Select Input Register (IOMUXC_UART3_UART_RTS_B_SELECT_INPUT)

Address: 20E_0000h base + 92Ch offset = 20E_092Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																													DAISY		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART3_UART_RTS_B_SELECT_INPUT field descriptions

Field	Description
31–3 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field

Table continues on the next page...

IOMUXC_UART3_UART_RTS_B_SELECT_INPUT field descriptions (continued)

Field	Description
	Selecting Pads Involved in Daisy Chain. Route RTS_B signal to pad when UART is in DCE mode. Route CTS_B signal to pad when UART is in DTE mode. 000 **EIM_DATA23_ALT2** — Selecting ALT2 mode of pad EIM_D23 for UART3_CTS_B. 001 **EIM_EB3_B_ALT2** — Selecting ALT2 mode of pad EIM_EB3 for UART3_RTS_B. 010 **EIM_DATA30_ALT4** — Selecting ALT4 mode of pad EIM_D30 for UART3_CTS_B. 011 **EIM_DATA31_ALT4** — Selecting ALT4 mode of pad EIM_D31 for UART3_RTS_B. 100 **SD3_DATA3_ALT1** — Selecting ALT1 mode of pad SD3_DAT3 for UART3_CTS_B. 101 **SD3_RESET_ALT1** — Selecting ALT1 mode of pad SD3_RST for UART3_RTS_B.

36.4.583 Select Input Register (IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT)

Address: 20E_0000h base + 930h offset = 20E_0930h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART3_UART_RX_DATA_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RX_DATA signal to pad when UART is in DCE mode. Route TX_DATA signal to pad when UART is in DTE mode. 00 **EIM_DATA24_ALT2** — Selecting ALT2 mode of pad EIM_D24 for UART3_TX_DATA. 01 **EIM_DATA25_ALT2** — Selecting ALT2 mode of pad EIM_D25 for UART3_RX_DATA. 10 **SD4_CMD_ALT2** — Selecting ALT2 mode of pad SD4_CMD for UART3_TX_DATA. 11 **SD4_CLK_ALT2** — Selecting ALT2 mode of pad SD4_CLK for UART3_RX_DATA.

36.4.584 Select Input Register (IOMUXC_UART4_UART_RTS_B_SELECT_INPUT)

Address: 20E_0000h base + 934h offset = 20E_0934h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															DAISY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART4_UART_RTS_B_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RTS_B signal to pad when UART is in DCE mode. Route CTS_B signal to pad when UART is in DTE mode. 0 **CSI0_DATA16_ALT3** — Selecting ALT3 mode of pad CSI0_DAT16 for UART4_RTS_B. 1 **CSI0_DATA17_ALT3** — Selecting ALT3 mode of pad CSI0_DAT17 for UART4_CTS_B.

36.4.585 Select Input Register (IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT)

Address: 20E_0000h base + 938h offset = 20E_0938h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART4_UART_RX_DATA_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RX_DATA signal to pad when UART is in DCE mode. Route TX_DATA signal to pad when UART is in DTE mode. 00 **KEY_COL0_ALT4** — Selecting ALT4 mode of pad KEY_COL0 for UART4_TX_DATA. 01 **KEY_ROW0_ALT4** — Selecting ALT4 mode of pad KEY_ROW0 for UART4_RX_DATA. 10 **CSI0_DATA12_ALT3** — Selecting ALT3 mode of pad CSI0_DAT12 for UART4_TX_DATA. 11 **CSI0_DATA13_ALT3** — Selecting ALT3 mode of pad CSI0_DAT13 for UART4_RX_DATA.

36.4.586 Select Input Register (IOMUXC_UART5_UART_RTS_B_SELECT_INPUT)

Address: 20E_0000h base + 93Ch offset = 20E_093Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART5_UART_RTS_B_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RTS_B signal to pad when UART is in DCE mode. Route CTS_B signal to pad when UART is in DTE mode. 00 **KEY_COL4_ALT4** — Selecting ALT4 mode of pad KEY_COL4 for UART5_RTS_B. 01 **KEY_ROW4_ALT4** — Selecting ALT4 mode of pad KEY_ROW4 for UART5_CTS_B. 10 **CSI0_DATA18_ALT3** — Selecting ALT3 mode of pad CSI0_DAT18 for UART5_RTS_B. 11 **CSI0_DATA19_ALT3** — Selecting ALT3 mode of pad CSI0_DAT19 for UART5_CTS_B.

36.4.587 Select Input Register (IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT)

Address: 20E_0000h base + 940h offset = 20E_0940h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

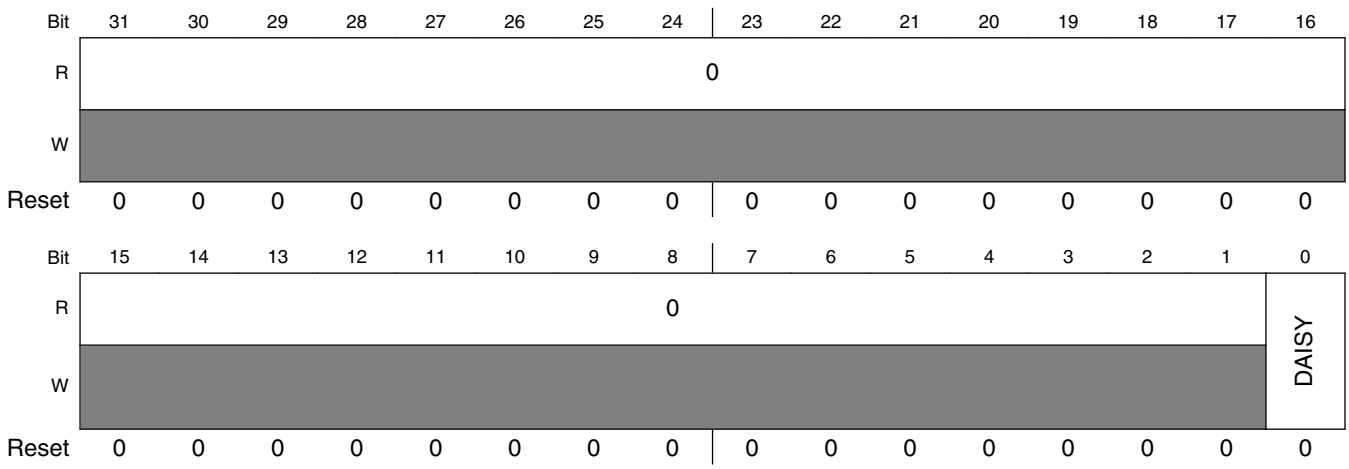
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0														DAISY	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IOMUXC_UART5_UART_RX_DATA_SELECT_INPUT field descriptions

Field	Description
31–2 Reserved	This read-only field is reserved and always has the value 0.
DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. Route RX_DATA signal to pad when UART is in DCE mode. Route TX_DATA signal to pad when UART is in DTE mode. 00 **KEY_COL1_ALT4** — Selecting ALT4 mode of pad KEY_COL1 for UART5_TX_DATA. 01 **KEY_ROW1_ALT4** — Selecting ALT4 mode of pad KEY_ROW1 for UART5_RX_DATA. 10 **CSI0_DATA14_ALT3** — Selecting ALT3 mode of pad CSI0_DAT14 for UART5_TX_DATA. 11 **CSI0_DATA15_ALT3** — Selecting ALT3 mode of pad CSI0_DAT15 for UART5_RX_DATA.

36.4.588 Select Input Register (IOMUXC_USB_OTG_OC_SELECT_INPUT)

Address: 20E_0000h base + 944h offset = 20E_0944h

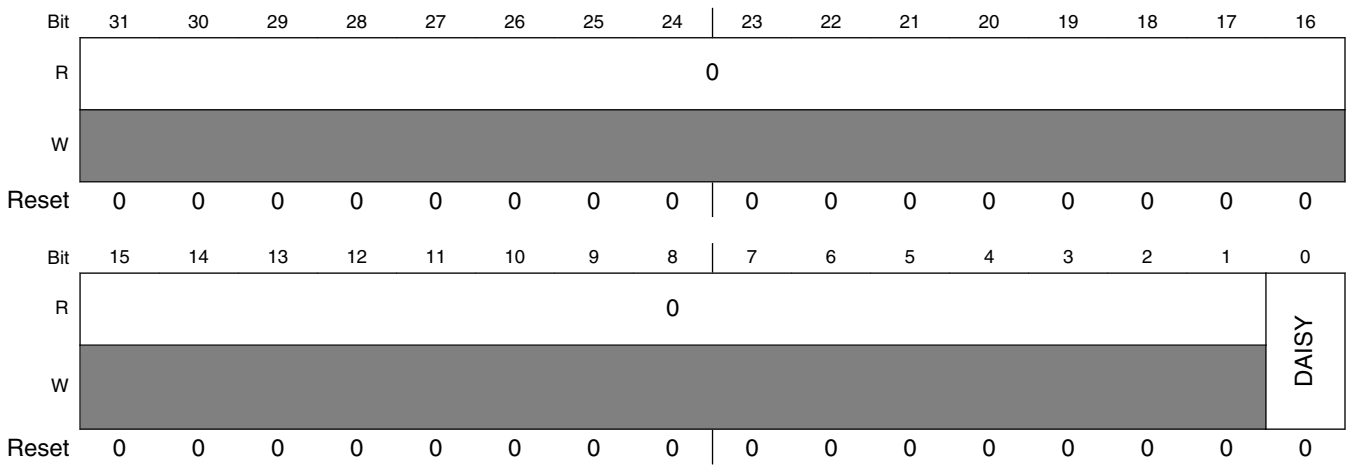


IOMUXC_USB_OTG_OC_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA21_ALT4** — Selecting ALT4 mode of pad EIM_D21 for USB_OTG_OC. 1 **KEY_COL4_ALT2** — Selecting ALT2 mode of pad KEY_COL4 for USB_OTG_OC.

36.4.589 Select Input Register (IOMUXC_USB_H1_OC_SELECT_INPUT)

Address: 20E_0000h base + 948h offset = 20E_0948h

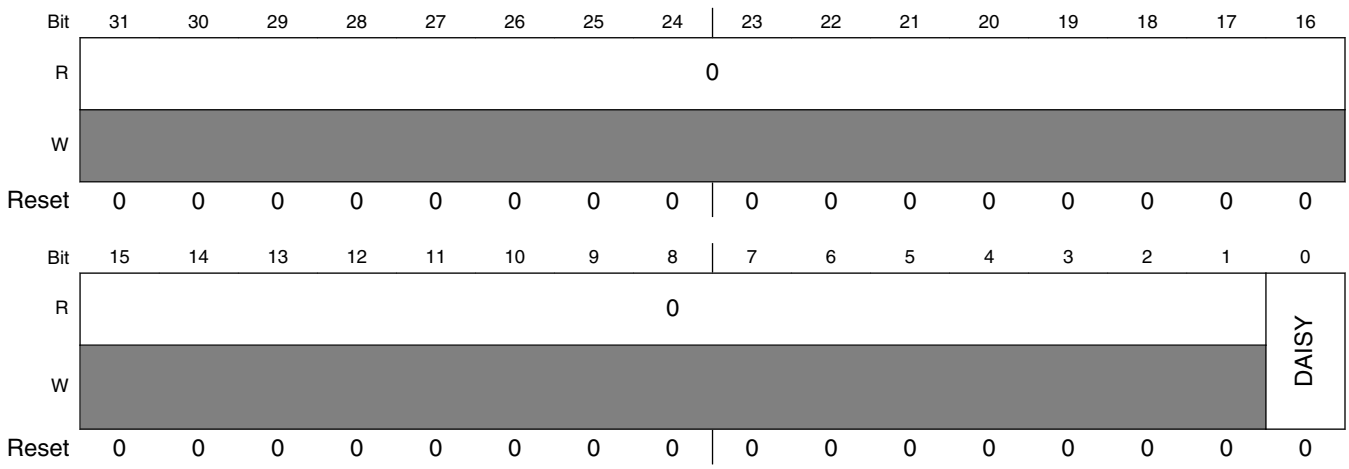


IOMUXC_USB_H1_OC_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **EIM_DATA30_ALT6** — Selecting ALT6 mode of pad EIM_D30 for USB_H1_OC. 1 **GPIO03_ALT6** — Selecting ALT6 mode of pad GPIO_3 for USB_H1_OC.

36.4.590 Select Input Register (IOMUXC_USDHC1_WP_ON_SELECT_INPUT)

Address: 20E_0000h base + 94Ch offset = 20E_094Ch



IOMUXC_USDHC1_WP_ON_SELECT_INPUT field descriptions

Field	Description
31–1 Reserved	This read-only field is reserved and always has the value 0.
0 DAISY	Input Select (DAISY) Field Selecting Pads Involved in Daisy Chain. 0 **DI0_PIN04_ALT3** — Selecting ALT3 mode of pad DI0_PIN4 for SD1_WP. 1 **GPIO09_ALT6** — Selecting ALT6 mode of pad GPIO_9 for SD1_WP.

Chapter 37
Image Processing Unit (IPU)

37.1 Overview

The IPU is planned to be a part of the video and graphics subsystem in an application processor.

The goal of the IPU is to provide comprehensive support for the flow of data from an image sensor and/or to a display device. This support covers all aspects of these activities:

- Connectivity to relevant devices - cameras, displays, graphics accelerators, TV encoders and decoders.
- Related image processing and manipulation: sensor image signal processing, display processing, image conversions, etc.
- Synchronization and control capabilities (to avoid tearing artifacts).

This integrative approach leads to several significant advantages:

- Automation: The involvement of the ARM platform in image management is minimized. In particular, display refresh/update and a camera preview (displaying the input from an image sensor) can be performed completely autonomously. The resulting benefits are reducing the overhead due to SW-HW synchronization, freeing the ARM platform to perform other tasks and reduced power consumption (when the ARM core is idle and can be powered down).
- Optimal data path: Access to system memory is minimized. In particular, significant processing can be performed on-the-fly while receiving data from an image sensor and/or sending data to a display. System memory is used essentially only when a change in pixel order or frame rate is needed. The resulting benefits are reduced load on the system bus and further reduction of power consumption.
- Resource sharing: Maximal HW reuse for different applications, resulting with the support of a wide range of requirements with minimal HW.

The HW reuse mentioned above is enabled by a sophisticated configurability of each HW block. This configurability also allows the support of a wide range of external devices, data formats and operation modes. The resulting flexibility is important also because the support requirements are evolving significantly, so expected future changes need to be anticipated and accounted for.

The following further principles guided the choice of support provided by the IPU:

- For key applications that deserve and need HW support (for acceleration or low power), provide the best support (leading to an optimal implementation).
- For additional applications that can benefit from the HW, consider cost vs. benefit of making minor modifications/extensions to support them.
- For all other relevant applications (to be supported by SW), verify that their support is not degraded.
- Whenever possible, let the operating system (and its windowing system) act as it would without the IPU.

37.1.1 Architecture

A simplified block diagram of the IPU can be found here. The role of each block is described in IPU.

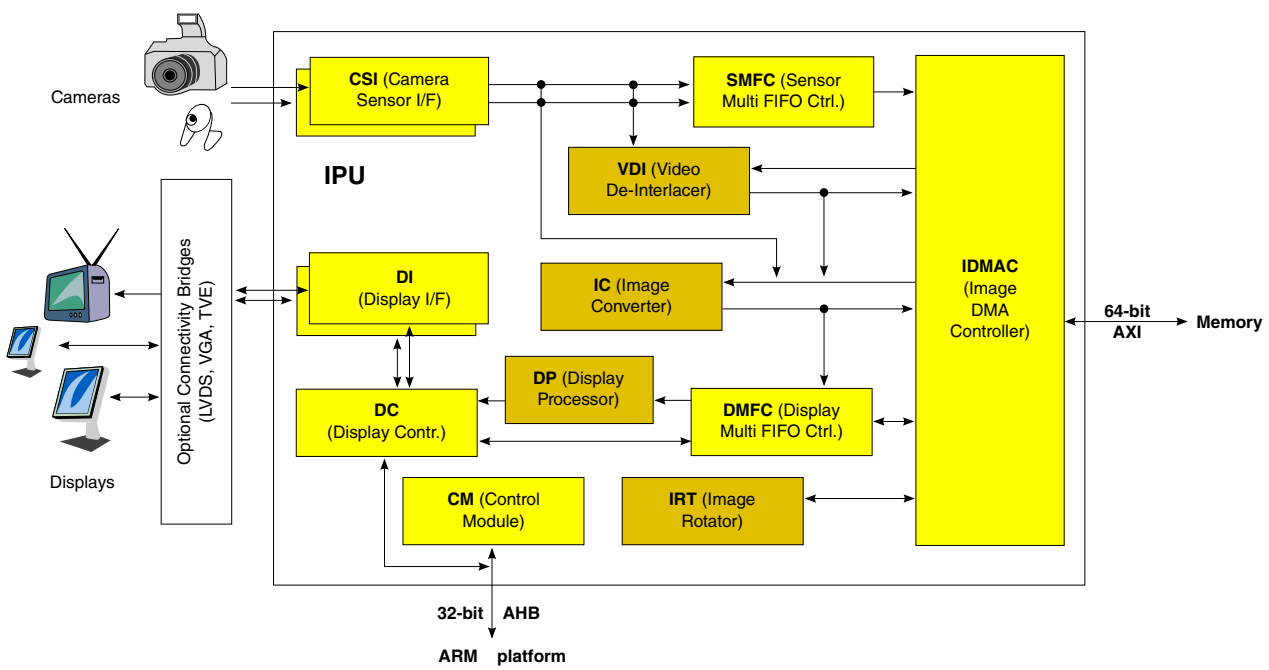


Figure 37-1. IPU Block Diagram

Table 37-1. IPU - Block Description

Block	Description
CSI - Camera Sensor Interface	Controls a camera port; provides interface to an image sensor or a related device. IPU includes 2 such blocks.
DI - Display Interface	Provides interface to displays, display controllers and related devices. IPU includes 2 such blocks.
DC - Display Controller	Controls the display ports.
DP - Display Processor	Performs the processing required for data sent to display.
IC - Image Converter	Performs resizing, color conversion/correction, combining with graphics, and horizontal inversion.
VDIC - Video De Interlacer	Performs video de interlacing (Interlaced -> progressive) or combining.
IRT - Image Rotator	Performs rotation (90 or 180 degrees) and inversion (vertical/horizontal).
IDMAC - Image DMA Controller	Controls the memory port; transfers data to/from system memory.
DMFC - Display Multi FIFO Controller	Controls FIFOs for IDMAC channels related to the display system.
CM - Control Module	Provides control and synchronization.

37.1.2 Features And Functionality

37.1.2.1 External Ports

IPU has the following ports:

- Two camera ports - each controlled by a CSI sub-block, providing a connection to image sensors and related devices.
- Two display ports - each controlled by a DI sub-block, providing a connection to displays and related devices.
- Memory port - AXI (AHB V3.0) master, controlled by the IDMAC - providing connection to the system memory.
- AHB-lite slave port, providing connection to the ARM Platform (and to any other master connected to the ARM's cross-bar switch).
- Additional ports for control and debug.

37.1.2.1.1 Camera Ports

The role of these ports is to receive input from image sensors (or TV decoders) and to provide support for time-sensitive control signals to the camera.

(Non-time-sensitive controls; configuration, reset are performed by the ARM platform through I2C I/F or GPIO signals).

Each of the camera ports includes the following features:

- Direct connectivity to most relevant image sensors and to TV decoders.
- Interface types
  - Parallel interface
    - Up to 20-bit input data bus.
    - A single value in each cycle, except for special cases listed in the table below (comments column).
    - Programmable polarity.
  - High-speed serial interface - MIPI (Mobile Industry Processor Interface) CSI-2 (Camera Serial Interface) (implemented partly in the IPU and partly in the HSC).
    - Up to four data lanes; up to 800 Mbps per lane
    - Class 1 compliancy (supporting all primary formats)
- The data formats
  - Interleaved color components, up to 16 bits per value (component).
  - The supported formats are listed in the table below.

Table 37-2. Data Formats Supported By The Camera Port

Format	Resolution	On-The-Fly Processing	Direct path to memory	Comments
Bayer RGB	8 bits/value	No	8- or 16-bit values	MIPI mandatory format
	9-10 bits/value	No	Written to the MSB of a 16-bit word	10 bits/value is a MIPI mandatory format
	16 bits/value	No		
Full RGB or YUV 4:4:4	444/555 mode	Yes, starting with color extension to 8 bits/ sample	Yes	MIPI optional formats In parallel I/F: through an 8-bit or 16-bit bus
	565 mode			MIPI mandatory format In parallel I/F: through an 8-bit or 16-bit bus
	8 bits/value (888 mode)	Yes	Yes	MIPI mandatory format
	8-16 bits/value	No	8- or 16-bit components are written to the MSB of a 16-bit word 10 bits/ value can also be packed in a 32-bit word	
YUV 4:2:2 Component order: UY1VY2... or Y1UY2V...	8 bits/value	Yes	Yes	MIPI mandatory format (UY1VY2...) In parallel I/F: through an 8-bit bus (such as BT.656) or 16-bit bus (such as BT.1120)

Table continues on the next page...

Table 37-2. Data Formats Supported By The Camera Port (continued)

Format	Resolution	On-The-Fly Processing	Direct path to memory	Comments
	9-10 bits/value	No	Written to the MSB of a 16-bit word	In parallel I/F: through a 10-bit bus (such as BT.656) or 20-bit bus (such as BT.1120)
	16 bits/value	No	Written to the MSB of a 16-bit word	
Gray scale	8 bits/value	No	Yes	
	16 bits/value	No	Written to the MSB of a 16-bit word	
Generic data		No	Yes In a parallel I/F, if wider than 8 bits, each bus word is written to the MSB of a 16-bit word	MIPI mandatory format May be used for any other formant, such as JPEG/MPEG4

- Scan order: progressive or interlaced data (expected only for YUV 4:2:2) is sent directly to system memory, where it can be read back for further processing.
- Frame size: up to 8192 x 4096 pixels
- Synchronization: video mode
  - The sensor is the master of the pixel clock (PIXCLK) & synchronization signals
  - Synchronization signals are received using either of the following methods:
    - Dedicated control signals -VSYNC, HSYNC - with programmable pulse width & polarity
    - Controls embedded in the data stream, following loosely the BT.656 protocol, with flexibility in code values and location.
- Synchronization : still image capture
  - The image capture is triggered by the ARM platform or by an external signal (such as a mechanical shutter).
  - Synchronized strobes are generated for up to 6 outputs - the sensor and camera peripherals (such as flash, mechanical shutter).
- Additional features
  - Frame rate reduction, by the periodic skipping of frames
    - The supported reduction ratios are: m:n, where m,n<=5
    - This is supported independently for the different destinations - IC, SMFC.
  - Window-of-interest selection
  - Pre-flash - for red-eye reduction and for measurements (such as focus) in low-light conditions

Several sensors can be connected to each of the CSIs. Simultaneous functionality (sending data) is supported as follows:

- Two sensors can send data independently, each through a different port, each using either parallel or fast serial interface.
- Several sensors can send data to the same port, using the MIPI interface (through a HUB), each sensor being identified by different ID's.
- Unpacking and companding capabilities are provided for up to two streams (either through same or different interfaces), while the other ones are treated as generic data.
- Only one of the (non-generic) streams can be transferred to theVDI C or IC for on-the-fly processing, while the others are sent directly to system memory.

The input rate supported by the camera port is as follows:

For parallel interface, the maximum speed of the interface is 240Mhz. The required operating frequency of the interface is calculated in the following way:

F = FH * FW * FPS * BI * DF

Where

- **FH** = frame height (in pixels)
- **FW** = Frame width (in pixels)
- **fps** = frame rate (frames per second)
- **BI** = typically 35% overhead, should be assumed as 1.35. The actual blanking intervals are a parameter of the attached device.
- **DF** = data format, defines the number of cycles needed to send a single pixel.

The number of cycles needed to send a single pixel depends on the interface and the data format.

Data format examples:

- YUV422 over 16 bit = 1 cycles/pixel
- RGB888 over 8 bit = 3 cycles/pixel
- RGB565 over 16 bit = 2 cycles/pixel
- Bayer/Generic data = 1 cycle/pixel
- YUV422 over 8 bit = 2 cycles/pixel
- BT.656, YUV422 format = 2 cycles/pixel
- BT.1120, YUV422 format = 1 cycle/pixel

Examples of supported interfaces:

- 3.2MP camera, 15fps, yuv422 format, 8 bit interface
- 1080P30, yuv422, 8 bit interface

Fast serial interface (MIPI-CSI2):

- IPU receives 2 components per cycle from the MIPI-CSI2 interface
- The maximum speed of the interface is:
  - 200Mhz for 4 data lanes configuration
  - 250Mhz for 2 data lanes configuration

The maximum bandwidth of the interface is as follows:

- 200Mhz for 4 data lanes configuration (800Mbps/lane, 400MByte/sec)
- 187.5Mhz for 3 data lanes configuration (1000Mbps/lane, 375MByte/sec)
- 125Mhz for 2 data lanes configuration (1000Mbps/lane, 250MByte/sec)
- 62.5Mhz for 1 data lane configuration (1000Mbps/lane, 125Mbyte/sec)

The required operating frequency of the interface is calculated in the same way as for parallel interface above. The DF parameter is different.

- YUV422 = 1 cycle/pixel
- RGB888 = 1.5 cycles/pixel
- Generic data = 2 bytes/pixel

Examples of supported interfaces:

- 3.2MP camera, 2 lanes configuration, 15fps, yuv422 format (~65Mhz)
- 6MP camera, 4 lanes configuration, 15fps, RGB888 format (~182Mhz)

37.1.2.1.2 Display Ports

The role of these ports is to communicate with display devices, either directly or through a controller (such as a graphics accelerator) or a bridge (such as a TV encoder or an LVDS interface bridge).

37.1.2.1.2.1 Access Modes

Two access modes are supported.

37.1.2.1.2.1.1 Synchronous Access

In this mode, the IPU transfers a two-dimensional block of pixels to the display device, in synchronization with the screen refresh cycle.

It is called "video mode" in the MIPI standards.

This mode has a dual role:

- For a RAM-less display or a TV screen, this mode is used to perform the screen refresh process from a display buffer in system memory.
- For a "smart" display, this mode is used to transfer a rectangular block of pixels to the display's screen and, in some cases, also to the display buffer
  - The transferred block may be only part of the screen (the rest of the screen being refreshed by the integrated controller, from the internal buffer). Moreover, a mask can be used to transfer to the display only parts of the block, such as a window partly hidden by other windows.
  - If the block is transferred only to the screen, the transfer rate must be equal to the refresh rate. If, however, the transfer is also to the display's memory, the rate can be reduced to the rate at which the input buffer is updated.

In all cases (including the last one), the IPU sends to the display all the synchronization signals controlling the screen refresh and the block transfer is synchronized with these signals. This synchronization means that tearing effects are avoided when using this mode.

37.1.2.1.2.1.2 Asynchronous Access

This is the main mode used for communicating with an external display controller (possibly in a smart display or a graphics accelerator).

It is called "command mode" in the MIPI standards. In this mode, the IPU performs random access - read/write - to the memory and registers of the controller.

Two types of addressing methods are supported

- Generic linear addressing of pixels and generic data
- 2-dimensional (X/Y) addressing of pixels

The following access types are provided:

- Data transfer to the external device, after on-the-fly processing in the IPU.
- Data transfer (DMA) - read/write - between the host's system memory and the external device, through the IPU's memory port (controlled by the IDMAC), such as the transfer of a rectangular block of pixels (possibly full screen).
- Host access - read/write - to an external device, through the AHB-slave port
  - Access types

    - Direct access - emulating a directly-addressed access (see below) This includes burst access (incremental; up to 8 words/burst)
    - Low-level access - leaving to the host the explicit generation of the access protocol
  - The possible accessing modules include the ARM platform and the system DMA controller (as well as any other AHB master connected to the ARM's cross-bar switch).

Transfer of video/graphics data stream to controller's display buffer is performed using one of the first two modes above. Unlike in the synchronous mode, this process is not tightly-synchronized with the screen refresh cycle. However, a loose synchronization - to avoid tearing - is still possible: the appropriate timing for the transfer can be derived from the VSYNC signal of the screen refresh - either generated by the IPU's display controller or received from the external controller.

The asynchronous access requires the specification of an address. The display interface uses "indirect addressing", namely, there is no address bus, and the address, as well as control and configuration commands, are embedded in the data stream. The access procedure - including writing addresses and commands - is managed autonomously by the interface, in one of two ways:

- Automatic emulation of transparent access, following microcode ("access template") generated by the ARM Platform. This mechanism is very flexible, supporting a wide variety of devices.
- Streaming commands/addresses from a buffer stored (by the ARM Platform) in system memory.

Note that direct access requires the use of the first method - automatic emulation.

37.1.2.1.2.2 Display Interface

The display interface is very flexible and supports a wide variety of devices from major manufacturers. The following interface types are provided (in each of the two display ports)

- Parallel video interface (for synchronous access) - up to 24-bit data bus.
  - Compatible with MIPI-DPI standard .
  - Control protocol - follows Sharp HR and generic TFT definitions
  - Supports BT.656 (8-bit) and BT.1120 (16-bit) protocols
  - Supports HDTV standards SMPTE274 (1080i/p) and SMPTE296 (720p)
- A parallel bidirectional bus interface (for asynchronous access) - up to 32-bit data bus.
  - Compatible with MIPI-DBI standard.

    - Control protocol - either system-80 or system-68K The timing and polarity of the signals are programmable.
    - Byte-enable - optional, for a 16-bit interface
    - Burst access for direct access, the burst is determined by the corresponding signal in the AHB interface.
- A serial interface - 3-wire, 4-wire and 5-wire (two flavors) (for asynchronous access)
- High-speed serial interface: MIPI (Mobile Industry Processor Interface) - DSI (Display Serial Interface) full support, with up to 2 data lanes, up to 4 virtual channels (implemented party in the IPU and partly in the HSC).

The supported formats for pixel data are

- RGB - color depth fully configurable; up to 8 bits/value (color component)
- YUV 4:2:2, 8 bits/value (for TV encoder)
- All mandatory formats in MIPI's DBI, DPI and DSI.

In the parallel interfaces, the data bus has up to 32 bits. Non-trivial mapping of pixels to the bus is restricted to the 24 LSB's. This mapping is fully configurable and very flexible. In the serial interfaces, the data is mapped in the same way as in the parallel interfaces and then serialized.

The interface also supports "generic data". Such data is transferred - byte-by-byte, without modification - between the system memory and the display device (through a serial interface or 8/16-bit parallel interface). Non-conventional pixel formats can be supported by considering them as "generic data".

For the interface clock, there are the following options (independently for each port)

- Derived from the IPU internal clock (master mode)
- Provided by an external source (slave mode)

The transfer rate supported:

- For single port (for on-chip interfaces):
    - 240 Mega-Accesses/Sec if the DI clock is derived from an external to IPU source (like another PLL)
    - 264 Mega-Accesses/Sec if the DI clock is derived from the IPU clock (HSP_CLK)
    - When off-chip interfaces are involved the rate may be limited by IO capabilities. Please refer to the device's data-sheet for exact numbers.

For synchronous access with one cycle/pixel, this enables, e.g (including 35% blanking intervals)

- 1080p (1920x1080) @ 60 fps

- WSXGA+ (1680x1050) @ 60 fps
- The combined rate for the two ports is up to 240 MP/sec

The interface includes the following additional features:

- Screen size: up to 4096 x 2048 pixels, programmable by software.
- Scan Order: progressive or interlaced
- Synchronization
  - Programmable horizontal and vertical synchronization output signals (for synchronous access)
  - Data enabling output signal
- Software contrast control using 8-bit programmable pulse-width modulation (PWM) Two dedicated PWM outputs are provided

Connecting To Display Devices

IPU allows the connectivity to multiple display devices. In particular, it supports the following setup:

- Primary LCD display; can be smart, dumb (RAM-less) or dual-port; may use fast serial, or the parallel interface or (through an integrated bridge) LVDS interface.
- Second LCD display; can be smart or dumb (RAM-less); may use fast serial, parallel or serial interface or (through an integrated bridge) LVDS interface.

Each of the above connections has independent settings - interface timing, access template, chip-select, etc.

Simultaneous functionality of the above devices is possible in each of the following ways:

- Two devices can be accessed (synchronously or asynchronously) independently, each through a different port: each using any of the available interfaces.
- Two devices can time-share asynchronous accesses through the legacy serial & parallel interfaces, using the CS signals.
- Two devices can be accessed - synchronously or asynchronously - through the same port, using the MIPI interface (through a HUB), each device being identified by different ID's.
- An asynchronous access can be performed during vertical blanking intervals of a synchronous access (screen refresh; to the same or other device).

The possibilities for simultaneous functionality by time-sharing the legacy interfaces in a single port are summarized in the following table.

Table 37-3. Simultaneous Functionality of Display Port By Time-Sharing Legacy Interfaces

	Second/Third Display Type		
Primary Display Type	**Smart Display Serial Interface**	**Smart Display Parallel Interface Asynchronous access**	**RAM-less display or TV screen**
Smart Display Parallel Interface Asynchronous access	Yes	Yes	Yes; access to the smart display is restricted to blanking intervals
Dual Port Smart Display Synchronous access	Yes	Yes, access to the secondary display is restricted to blanking intervals	Not Available
TFT RAM-less Display Or TV screen	Yes	Yes, access to the smart display is restricted to blanking intervals	Not Available
Graphics Accelerator	Yes	Yes, if the accelerator supports a chip select functionality	Not Available

37.1.2.1.3 Memory Port

The memory port is an AXI (AHB V3.0) master port, used to read/write data - typically two-dimensional blocks from/to system memory.

The interface supports the following features

- Clock rate up to 264 MHz (equal to the internal clock)
- 64-bit data bus
- The supported data formats are listed in the table below.

Table 37-4. Data Formats Supported By The Memory Interface

Format	Resolution	Input/Output	Comments
Non-interleaved YUV (in three separate buffers)	8 bits/value	both	4:4:4, 4:2:2, 4:2:0 formats
Partially-interleaved YUV (in two separate buffers)	8 bits/value	both	4:2:2, 4:2:0 formats Y buffer and UV buffer
Interleaved YUV (all color components in a single buffer)	8 bits/value	both	4:4:4 format (YUV...) 4:2:2 format (UY1VY2... or UY2VY1... or Y1UY2V...or Y2UY1V...)
Interleaved true color	8, 12,16, 18, 24, 32 bits/ pixel	both	Flexible component location A: translucency value

Table continues on the next page...

Table 37-4. Data Formats Supported By The Memory Interface (continued)

Format	Resolution	Input/Output	Comments
	0 - 8 bits per R/G/B/A value		(only in input)
Coded color (using a palette)	4,8 bits/pixel	input only	
Gray scale	8 bits/pixel	both	
	4 bits/pixel	input only	Transferred to display port
Generic data (Transparent M)	8 bits/unit	both	E.g.: From CSI to DI Compressed data to/from DP Translucency for combining

- The pixel formats are translated to/from a uniform internal format: RGBA/YUVA 8:8:8:8
- The supported ordering of bytes and pixels is little endian. For 4 bits/pixel, big endian is also supported.
- Addressing modes include:
  - Sequential access (to a contiguous memory buffer) - for generic data.
  - Raster-scan within a two-dimensional window of a video/display buffer - for both pixel and generic data.
  - Raster-scan of two-dimensional blocks within a two-dimensional window (for rotation of pixel data)
- Additional features
  - Scan order: progressive or interlaced Interlaced access is supported for fields which are stored either in separate memory buffers or with rows interleaved in a single buffer.
  - Reordered scan, implementing inversion and rotation.
    - Rotation and horizontal inversion - only when transferring two-dimensional blocks (to/from the IRT)
    - Vertical inversion - also in row-by-row raster-scan
  - Scrolling
    - Applications Panning within a frame Frame scrolling
    - Not supported for non-interleaved and partially-interleaved formats
    - Resolution Vertical: single pixel Horizontal: 18 BPP - 4 pixels; 12, 4 BPP or YUV 422 - 2 pixels; other formats - one pixel
  - Conditional read (for combining): fully-transparent or hidden pixels are not read.
    - This is supported, for graphics. by reading the transparency (alpha) from a separate buffer
  - Input/output FIFOs (in the SMFC, DMFC and in the processing sub-blocks) - size adjusted to provide resilience for latency of up to 1500 cycles.

37.1.2.1.4 Processing

The IPU processes rectangular blocks of pixels. The processing is performed in these sub-blocks - VDIC, DP, IC and IRT.

(see the IPU block diagram and Table 37-1).

37.1.2.1.4.1 Processing flows

Several time-shared data flows are supported, as described in the following table.

Table 37-5. Time-Shared Data Flows through the IPU

Name	Number	Type	Flow	Target	Restrictions
Display Refresh/ Update	5 flows (at most two of them of type DS1)	DS1	Fmem -> DP -> Display	Synchronous Access (e.g. display refresh; controlled by the DI)	
		DS2	Fmem -> DP -> Display	Asynchronous Access (e.g. display update)	
		DS3	Fmem <-> Display	Generic Data Transfer	
	1 flow	DS4	ARM Platform<-> Display	Direct Access	
Video Playback	flows	PL1	Bmem -> VDIC -> IC -> Bmem -> IRT -> Fmem + DSx	Main option	
		PL2	Fmem -> IRT -> Bmem -> IC -> DP	Low power (branching to DSx, as a video plane)	Large enough window No other video flows
		PL3	Fmem -> VDIC -> IC -> DP	Low power (branching to DSx, as a video plane)	Interlaced source Large enough window No other video flows
Camera Preview	1 flow (VF2 may be used also as a playback flow)	VF1	Sensor -> IC -> Bmem -> IRT -> Fmem+DSx	main option	Single progressive input
		VF2	Sensor -> Fmem -> VDIC -> IC -> Bmem -> IRT -> Fmem+ DSx	two inputs and/or interlaced input	When the VDIC is used, one of the three input fields can go directly from the sensor to the VDIC. In that case the sensor output goes to the memory via the VDIC and not the SMFC
		VF4	Sensor -> IC -> Fmem + DS1	Low power RAM-less Display Single Display Buffer (in internal memory) Tearing-less	Single progressive input Refresh rate = 2x sensor frame rate Large enough window No other video flows
Video Record	1 flow	RCx	IC -> Bmem -> IRT -> Fmem	(branching from VFx)	

Table continues on the next page...

Table 37-5. Time-Shared Data Flows through the IPU (continued)

Name	Number	Type	Flow	Target	Restrictions
Graphic Overlays	2 flows	GF1	Fmem -> IC	(combining with the main flow	
	2 flow	GF2	Fmem -> DP		

Comments

- System memory usage - legend
  - Fmem: frame double-buffer (page-flip) in system memory (typically external memory: DDR DRAM [address 0x1000_0000 - 0xFFFF_FFFF])
  - Bmem: two possibilities
    - A frame double buffer, as above
    - A band (4-256 rows) double-buffer (page-flip) in system memory (could be internal memory: OCRAM [address 0x0900_0000 - 0x093F_FFFF])
  - Direct arrow between two processing stages represents an internal pipeline
- Time-sharing
  - IC can time-share tightly three flows: one VFx, one RCx and one PLx (with independent processing parameters)
  - DP can time-share one DS1 flow and a one DS2 flow (each with different destinations and independent processing parameters)
  - Direct access to display (DS4) time-shares tightly the display port with other active DSx flows.
  - Other time-sharing (between PLx flow and DS2 and DS3 flows in IRT) is frame-by-frame
- Any of the processing stages in the above flows can be skipped.
- Triggering and synchronization
  - Flow segments starting from a sensor are triggered and synchronized by input from the sensor
  - Flow segments ending with display refresh are triggered and synchronized by the refresh control mechanism in the DI.
  - Flow segments starting and ending in system memory, are triggered either by the double buffering mechanism or by explicit configuration and are processed continuously without delay, at a rate determined by the available resources. These flow segments have a lower priority than the sensor/display-driven flows.

The functionality of each of the processing blocks is described below.

37.1.2.1.4.2 Display Processor (DP)

The Display Processor performs all the processing required for data sent to a display.

- Input: from the IC and/or from system memory

    - Order: rows, progressive or interlaced
    - Format: YUVA/RGBA, non-decimated, 8 bits/value
- Processing chain
    - Combining 2 video/graphics planes
    - Overlaying a simple HW cursor 32 x 32 pixels, uniform color; may be combined logically with the full plane.
    - Color conversion/correction - linear (multiplicative & additive) programmable including:
        - YUV <-> RGB, YUV<->YUV conversions where YUV stands for any one of the color formats defined in the MPEG-4 standard
        - Adjustments: brightness, contrast, color saturation, etc.
        - Special effects: gray-scale, color inversion, sephia, blue-tone, etc.
        - Color-preserving clipping, for gamut mapping
        - Hue-preserving gamut mapping - for minimal color distortion
        - Applied to the output of combining or to one of the inputs
    - Gamma correction and contrast stretching - programmable piecewise-linear map
- Output: to display (through the DC)
    - Rate: up to 240M pixels/sec
    - Format: YUV/RGB, non-decimated, 8 bits/value

The DP processes a single data flow at any given time, but supports up to three data flows by time sharing.

- A Primary flow:
    - The input is loaded periodically using a timer (e.g. for a synchronous access)
    - Optionally, frames are skipped if the content has not changed (as appropriate for a smart display)
- Two secondary flows:
    - Asynchronous; processed when the DP is not needed for the primary flow (during blanking intervals or when a primary frame is skipped).
    - The two secondary flows are switched frame-by-frame.

37.1.2.1.5 Video De-Interlacer or Combiner (VDIC)

The Video De-Interlacer as well as the Combiner have two operation modes

- De-interlacing: converts an interlaced video stream to progressive order.
- Combining: combines two video/graphics planes and a background color

37.1.2.1.5.1 De-interlacing in the VDIC

The Video De-Interlacer converts an interlaced video stream to progressive order, using a high-quality 3-field motion-adaptive filter.

- Video source - SDTV: 480i30 (720x480 @ 30 fps) or 576i25 (720x576 @ 25 fps) and HDTV: 1080i/30 (1920x1080 @ 30 fps)
- Input: -three consecutive fields
  - Source
    - The most recent field may come from the CSI or from system memory
    - The other two fields are read from memory
  - Field size: up to 968x1024 pixels (may be a vertical stripe of a wider field; e.g. 1920 pixels)
  - Pixel format: YUV 4:2:2/4:2:0, 8 bits/value
- Output: progressive frame
  - Destination: to system memory or to the Image Converter.
  - Frame size: up to 968x2048 pixels
  - Rate: up to 240 MP/sec (e.g., 1920x1080 @ 85 fps)
  - Format: same as input format

The de-interlacing is performed using a high-quality 3-field filter which is motion adaptive:

- For slow motion - retains the full resolution (of both top and bottom fields)
- For fast motion - prevents motion artifacts

The VDIC supports a single video stream at any given time.

37.1.2.1.5.2 Combining in the VDIC

- Input for combining: two progressive video/graphics planes
  - Source: system memory
  - Plane size: up to 1920x1200 pixels.
  - Pixel format: RGB/YUV 4:2:2, 8 bits/value

In this mode:

- The two input planes are read from system memory, using the previous field and next field input FIFOs.
- Their relative height, up/down is configurable
- Each of them may cover only part of the output frame. For the remaining part of the frame, the chip uses the following values:
  - Down plane: a 24-bit background color, stored in an internal register
  - Up plane: a transparent pixel
- The combining method is identical to the one in the DP, IC.

37.1.2.1.5.3 Image Converter (IC)

The Image Converter performs various operations on a video stream.

- Input: from sensor or from system memory
  - Frame size: up to 4096x4096 pixels
  - Rate: up to 200M pixels/sec (e.g. 5 MP @ 30 fps + 35% blanking intervals)
  - Order: rows, progressive. If resizing is not used, interlaced order is also acceptable.
  - Pixel format: YUV/RGB, 8 bits/value
- Processing chain:
  - Resizing
    - Fully flexible resizing ratio Maximal downsizing ratio: 8:1. Subject to this limitation, any N->M resizing can be performed.
    - Independent horizontal and vertical resizing ratios.
  - Color conversion/correction - linear (multiplicative & additive) programmable, including:
    - YUV <-> RGB, YUV<->YUV conversions where YUV stands for any one of the color formats defined in the MPEG-4 standard
    - Adjustments: brightness, contrast, color saturation...
    - Special effects: gray-scale, color inversion, sephia, blue-tone...
  - Combining with a graphics plane (e.g. application-specific overlay)
  - Horizontal inversion
- Output: to system memory or (for a single active flow) to a display device (through the DP).
  - Frame size: up to 1024x1024 pixels
  - Rate: up to 100Mpixels/sec (e.g. 1920x1080 @ 30 fps)
  - Order: rows, progressive. If resizing is not used, interlaced order is also acceptable.
  - Format: YUV/RGB, 8 bits/value

The IC supports three time-shared data flows: record, camera preview and playback (the first two share a common input).

37.1.2.1.5.4 Image Rotator (IRT)

- Input/output: from/to system memory
  - Rate
    - Up to 120M pixels/sec (when a single task is active).
    - Up to 100M pixels/sec (when more than one task is active).
  - Order: raster scan of 8x8-pixel blocks
  - Format: YUV/RGB, non-decimated, 8 bits/value
- Transformation: combination of the following
  - 90-degree rotation
  - Horizontal inversion
  - Vertical inversion

37.1.2.1.6 Automatic Procedures

The IPU is equipped with powerful control and synchronization capabilities to perform its tasks with minimal involvement of the ARM Core and minimal use of memory.

In particular, it includes:

- An integrated DMA controller with an AXI master port, allowing autonomous access to the system memory.
- An integrated display controller, performing screen refresh of a RAM-less display.
- A page-flip double buffering mechanism, synchronizing read and write access to system memory, to prevent tearing effects.
- A double/triple buffer synchronization mechanism with a video/graphics source.
- Internal synchronization, e.g., between input from sensor and output to display

As a result, in most cases, the ARM platform is involved only when it also performs part of the processing (e.g. video coding). In particular, the following procedures are performed by the IPU completely autonomously:

- Screen refresh for RAM-less displays
- Update of the ("partial plane") display buffer used for screen refresh (located either in system memory or in an external display controller, e.g. of a smart display or graphics accelerator), when the content is generated in a different ("full plane") buffer.

Typically, there are extended periods of times in which there is no other activity in the system. The ARM platform, being idle, can be put to a low-power mode, reducing the power consumption and extending significantly the battery life.

The IPU supports several techniques to reduce further the power consumption of the display system:

- Dynamically optimized screen refresh rate (see Screen Refresh below)
- Optimized update of the display buffer (see Update Of The Display Buffer below)
- Dynamic backlight control, with low-light compensation by image enhancement

Further features and capabilities of the automatic procedures are outlined below

37.1.2.1.6.1 Screen Refresh

- The refresh rate may vary within a predefined range. Within this range, the rate is dynamically adjusted to the content update rate.
- An indication about the availability of new content is obtained as follows:

    - If the page-flip double buffering is used, the mechanism provides this indication
    - If only a single buffer is used (and incrementally updated), the IPU can receive an indication of a modification from the ARM platform (by setting an internal flag).
- The IPU counts the refresh cycles: the total and those with new content. The ARM platform can use these counters to optimize display management (e.g. switching display buffer compression on/off). The counters are reset by the ARM platform.
- The transferred data may be processed on the way, using the IC and DP.

37.1.2.1.6.2 Update Of The Display Buffer

- Conditional update The IPU can receive an external "snooping" signal indicating a modification of the full plane buffer (as during screen refresh above). It monitors the signal and, upon detection, it performs one of the following:
    - Performs an update, without any SW intervention
    - Interrupts the ARM core, that can initiate some more involved procedure (e.g. selective update)
- Automatic display of a changing image (animation) or moving image (scrolling) This is implemented by reading frames (from a full plane buffer) with incremental offset. When the IPU reaches the last programmed frame, it can perform one of the following:
    - Return to the first frame, without any SW intervention
    - Interrupt the ARM platform, to generate the next content.
- The timing of the update can be adjusted to avoid tearing.
- The transferred data may be processed on the way, using the IC and DP

37.1.2.1.6.3 Camera Preview

- Tearing artifacts can be prevented by (automatic) page-flip double buffering in system memory
- Alternatively, the video stream from an image sensor can be sent directly to the display buffer used for screen refresh. The significance of this option is that only a single frame buffer is needed (and not two). This buffer may be located either in system memory or in an external display controller.
    - This option is useful, e.g., in a low frame rate, when tearing is not visible.
    - When tearing must be prevented, the refresh cycle in the display can be synchronized with the timing signals from the sensor (two refresh cycles for each input frame): the IPU receives a VSYNC signal from the sensor and generates from it synchronization signals for the display.

37.2 External Signals

The table found here describes the external signals of IPU1.

Table 37-6. IPU1 External Signals

Signal	Description	Pad	Mode	Direction
IPU1_CSI0_DATA00	-	EIM_D27	ALT2	I
IPU1_CSI0_DATA01	-	EIM_D26	ALT2	I
IPU1_CSI0_DATA02	-	EIM_D31	ALT3	I
IPU1_CSI0_DATA03	-	EIM_D30	ALT3	I
IPU1_CSI0_DATA04	-	CSI0_DAT4	ALT0	I
IPU1_CSI0_DATA05	-	CSI0_DAT5	ALT0	I
IPU1_CSI0_DATA06	-	CSI0_DAT6	ALT0	I
IPU1_CSI0_DATA07	-	CSI0_DAT7	ALT0	I
IPU1_CSI0_DATA08	-	CSI0_DAT8	ALT0	I
IPU1_CSI0_DATA09	-	CSI0_DAT9	ALT0	I
IPU1_CSI0_DATA10	-	CSI0_DAT10	ALT0	I
IPU1_CSI0_DATA11	-	CSI0_DAT11	ALT0	I
IPU1_CSI0_DATA12	-	CSI0_DAT12	ALT0	I
IPU1_CSI0_DATA13	-	CSI0_DAT13	ALT0	I
IPU1_CSI0_DATA14	-	CSI0_DAT14	ALT0	I
IPU1_CSI0_DATA15	-	CSI0_DAT15	ALT0	I
IPU1_CSI0_DATA16	-	CSI0_DAT16	ALT0	I
IPU1_CSI0_DATA17	-	CSI0_DAT17	ALT0	I
IPU1_CSI0_DATA18	-	CSI0_DAT18	ALT0	I
IPU1_CSI0_DATA19	-	CSI0_DAT19	ALT0	I
IPU1_CSI0_DATA_EN	-	CSI0_DATA_EN	ALT0	I
IPU1_CSI0_HSYNC	-	CSI0_MCLK	ALT0	I
IPU1_CSI0_PIXCLK	-	CSI0_PIXCLK	ALT0	I
IPU1_CSI0_VSYNC	-	CSI0_VSYNC	ALT0	I
IPU1_DI0_D0_CS	-	EIM_D23	ALT1	O
IPU1_DI0_D1_CS	-	EIM_A25	ALT4	O
IPU1_DI0_DISP_CLK	-	DI0_DISP_CLK	ALT0	O
IPU1_DI0_PIN01	-	EIM_D22	ALT2	IO
IPU1_DI0_PIN02	-	DI0_PIN2	ALT0	O
IPU1_DI0_PIN03	-	DI0_PIN3	ALT0	O
IPU1_DI0_PIN04	-	DI0_PIN4	ALT0	O
IPU1_DI0_PIN05	-	EIM_D16	ALT2	O
IPU1_DI0_PIN06	-	EIM_D17	ALT2	O
IPU1_DI0_PIN07	-	EIM_D18	ALT2	O
IPU1_DI0_PIN08	-	EIM_D19	ALT2	O
IPU1_DI0_PIN11	-	EIM_D30	ALT2	O

Table continues on the next page...

Table 37-6. IPU1 External Signals (continued)

Signal	Description	Pad	Mode	Direction
IPU1_DI0_PIN12	-	EIM_D31	ALT2	O
IPU1_DI0_PIN13	-	EIM_D28	ALT7	O
IPU1_DI0_PIN14	-	EIM_D29	ALT7	O
IPU1_DI0_PIN15	-	DI0_PIN15	ALT0	O
IPU1_DI0_PIN16	-	EIM_D20	ALT2	O
IPU1_DI0_PIN17	-	EIM_D21	ALT2	O
IPU1_DI1_D0_CS		EIM_DA13	ALT1	O
		EIM_D18	ALT4	
IPU1_DI1_D1_CS	-	EIM_DA14	ALT1	O
IPU1_DI1_DISP_CLK	-	EIM_A16	ALT1	O
IPU1_DI1_PIN01	-	EIM_DA15	ALT1	IO
IPU1_DI1_PIN02		EIM_DA11	ALT1	O
		EIM_D23	ALT6	
IPU1_DI1_PIN03		EIM_DA12	ALT1	O
		EIM_EB3	ALT6	
IPU1_DI1_PIN04	-	EIM_DA15	ALT2	O
IPU1_DI1_PIN05	-	EIM_CS0	ALT1	O
IPU1_DI1_PIN06	-	EIM_CS1	ALT1	O
IPU1_DI1_PIN07	-	EIM_OE	ALT1	O
IPU1_DI1_PIN08	-	EIM_RW	ALT1	O
IPU1_DI1_PIN11	-	EIM_D26	ALT1	O
IPU1_DI1_PIN12	-	EIM_A25	ALT3	O
IPU1_DI1_PIN13	-	EIM_D27	ALT1	O
IPU1_DI1_PIN14	-	EIM_D23	ALT7	O
IPU1_DI1_PIN15		EIM_DA10	ALT1	O
		EIM_D29	ALT1	
IPU1_DI1_PIN16	-	EIM_BCLK	ALT1	O
IPU1_DI1_PIN17	-	EIM_LBA	ALT1	O
IPU1_DISP0_DATA00	-	DISP0_DAT0	ALT0	IO
IPU1_DISP0_DATA01	-	DISP0_DAT1	ALT0	IO
IPU1_DISP0_DATA02	-	DISP0_DAT2	ALT0	IO
IPU1_DISP0_DATA03	-	DISP0_DAT3	ALT0	IO
IPU1_DISP0_DATA04	-	DISP0_DAT4	ALT0	IO
IPU1_DISP0_DATA05	-	DISP0_DAT5	ALT0	IO
IPU1_DISP0_DATA06	-	DISP0_DAT6	ALT0	IO
IPU1_DISP0_DATA07	-	DISP0_DAT7	ALT0	IO
IPU1_DISP0_DATA08	-	DISP0_DAT8	ALT0	IO
IPU1_DISP0_DATA09	-	DISP0_DAT9	ALT0	IO
IPU1_DISP0_DATA10	-	DISP0_DAT10	ALT0	IO

Table continues on the next page...

Table 37-6. IPU1 External Signals (continued)

Signal	Description	Pad	Mode	Direction
IPU1_DISP0_DATA11	-	DISP0_DAT11	ALT0	IO
IPU1_DISP0_DATA12	-	DISP0_DAT12	ALT0	IO
IPU1_DISP0_DATA13	-	DISP0_DAT13	ALT0	IO
IPU1_DISP0_DATA14	-	DISP0_DAT14	ALT0	IO
IPU1_DISP0_DATA15	-	DISP0_DAT15	ALT0	IO
IPU1_DISP0_DATA16	-	DISP0_DAT16	ALT0	IO
IPU1_DISP0_DATA17	-	DISP0_DAT17	ALT0	IO
IPU1_DISP0_DATA18	-	DISP0_DAT18	ALT0	IO
IPU1_DISP0_DATA19	-	DISP0_DAT19	ALT0	IO
IPU1_DISP0_DATA20	-	DISP0_DAT20	ALT0	IO
IPU1_DISP0_DATA21	-	DISP0_DAT21	ALT0	IO
IPU1_DISP0_DATA22	-	DISP0_DAT22	ALT0	IO
IPU1_DISP0_DATA23	-	DISP0_DAT23	ALT0	IO
IPU1_DISP1_DATA00	-	EIM_DA9	ALT1	IO
IPU1_DISP1_DATA01	-	EIM_DA8	ALT1	IO
IPU1_DISP1_DATA02	-	EIM_DA7	ALT1	IO
IPU1_DISP1_DATA03	-	EIM_DA6	ALT1	IO
IPU1_DISP1_DATA04	-	EIM_DA5	ALT1	IO
IPU1_DISP1_DATA05	-	EIM_DA4	ALT1	IO
IPU1_DISP1_DATA06	-	EIM_DA3	ALT1	IO
IPU1_DISP1_DATA07	-	EIM_DA2	ALT1	IO
IPU1_DISP1_DATA08	-	EIM_DA1	ALT1	IO
IPU1_DISP1_DATA09	-	EIM_DA0	ALT1	IO
IPU1_DISP1_DATA10	-	EIM_EB1	ALT1	IO
IPU1_DISP1_DATA11	-	EIM_EB0	ALT1	IO
IPU1_DISP1_DATA12	-	EIM_A17	ALT1	IO
IPU1_DISP1_DATA13	-	EIM_A18	ALT1	IO
IPU1_DISP1_DATA14	-	EIM_A19	ALT1	IO
IPU1_DISP1_DATA15	-	EIM_A20	ALT1	IO
IPU1_DISP1_DATA16	-	EIM_A21	ALT1	IO
IPU1_DISP1_DATA17	-	EIM_A22	ALT1	IO
IPU1_DISP1_DATA18	-	EIM_A23	ALT1	IO
IPU1_DISP1_DATA19	-	EIM_A24	ALT1	IO
IPU1_DISP1_DATA20	-	EIM_D31	ALT1	IO
IPU1_DISP1_DATA21	-	EIM_D30	ALT1	IO
IPU1_DISP1_DATA22	-	EIM_D26	ALT7	IO
IPU1_DISP1_DATA23	-	EIM_D27	ALT7	IO
IPU1_EXT_TRIG	-	EIM_D28	ALT6	I
IPU1_SISG0	-	NANDF_CS2	ALT1	O

Table continues on the next page...

Table 37-6. IPU1 External Signals (continued)

Signal	Description	Pad	Mode	Direction
IPU1_SISG1	-	NANDF_CS3	ALT1	O
IPU1_SISG2		EIM_A24	ALT4	O
		EIM_D26	ALT6	
IPU1_SISG3		EIM_A23	ALT4	O
		EIM_D27	ALT6	
IPU1_SISG4	-	KEY_COL4	ALT1	O
IPU1_SISG5	-	KEY_ROW4	ALT1	O

Table 37-7. IPU2 External Signals

Signal	Description	Pad	Mode	Direction
IPU2_CSI1_DATA00	-	EIM_DA9	ALT2	I
IPU2_CSI1_DATA01	-	EIM_DA8	ALT2	I
IPU2_CSI1_DATA02	-	EIM_DA7	ALT2	I
IPU2_CSI1_DATA03	-	EIM_DA6	ALT2	I
IPU2_CSI1_DATA04	-	EIM_DA5	ALT2	I
IPU2_CSI1_DATA05	-	EIM_DA4	ALT2	I
IPU2_CSI1_DATA06	-	EIM_DA3	ALT2	I
IPU2_CSI1_DATA07	-	EIM_DA2	ALT2	I
IPU2_CSI1_DATA08	-	EIM_DA1	ALT2	I
IPU2_CSI1_DATA09	-	EIM_DA0	ALT2	I
IPU2_CSI1_DATA10		EIM_D22	ALT3	I
		EIM_EB1	ALT2	
IPU2_CSI1_DATA11		EIM_D21	ALT3	I
		EIM_EB0	ALT2	
IPU2_CSI1_DATA12		EIM_A17	ALT2	I
		EIM_D28	ALT3	
IPU2_CSI1_DATA13		EIM_A18	ALT2	I
		EIM_D27	ALT3	
IPU2_CSI1_DATA14		EIM_A19	ALT2	I
		EIM_D26	ALT3	
IPU2_CSI1_DATA15		EIM_A20	ALT2	I
		EIM_D20	ALT3	
IPU2_CSI1_DATA16		EIM_A21	ALT2	I
		EIM_D19	ALT3	
IPU2_CSI1_DATA17		EIM_A22	ALT2	I
		EIM_D18	ALT3	
IPU2_CSI1_DATA18		EIM_A23	ALT2	I
		EIM_D16	ALT3	

Table continues on the next page...

Table 37-7. IPU2 External Signals (continued)

Signal	Description	Pad	Mode	Direction
IPU2_CSI1_DATA19		EIM_A24	ALT2	I
		EIM_EB2	ALT3	
IPU2_CSI1_DATA_EN		EIM_DA10	ALT2	I
		EIM_D23	ALT4	
IPU2_CSI1_HSYNC		EIM_DA11	ALT2	I
		EIM_EB3	ALT4	
IPU2_CSI1_PIXCLK		EIM_A16	ALT2	I
		EIM_D17	ALT3	
IPU2_CSI1_VSYNC		EIM_DA12	ALT2	I
		EIM_D29	ALT6	
IPU2_DI0_DISP_CLK	-	DI0_DISP_CLK	ALT1	O
IPU2_DI0_PIN01	-	NANDF_RB0	ALT1	IO
IPU2_DI0_PIN02	-	DI0_PIN2	ALT1	O
IPU2_DI0_PIN03	-	DI0_PIN3	ALT1	O
IPU2_DI0_PIN04	-	DI0_PIN4	ALT1	O
IPU2_DI0_PIN15	-	DI0_PIN15	ALT1	O
IPU2_DISP0_DATA00	-	DISP0_DAT0	ALT1	IO
IPU2_DISP0_DATA01	-	DISP0_DAT1	ALT1	IO
IPU2_DISP0_DATA02	-	DISP0_DAT2	ALT1	IO
IPU2_DISP0_DATA03	-	DISP0_DAT3	ALT1	IO
IPU2_DISP0_DATA04	-	DISP0_DAT4	ALT1	IO
IPU2_DISP0_DATA05	-	DISP0_DAT5	ALT1	IO
IPU2_DISP0_DATA06	-	DISP0_DAT6	ALT1	IO
IPU2_DISP0_DATA07	-	DISP0_DAT7	ALT1	IO
IPU2_DISP0_DATA08	-	DISP0_DAT8	ALT1	IO
IPU2_DISP0_DATA09	-	DISP0_DAT9	ALT1	IO
IPU2_DISP0_DATA10	-	DISP0_DAT10	ALT1	IO
IPU2_DISP0_DATA11	-	DISP0_DAT11	ALT1	IO
IPU2_DISP0_DATA12	-	DISP0_DAT12	ALT1	IO
IPU2_DISP0_DATA13	-	DISP0_DAT13	ALT1	IO
IPU2_DISP0_DATA14	-	DISP0_DAT14	ALT1	IO
IPU2_DISP0_DATA15	-	DISP0_DAT15	ALT1	IO
IPU2_DISP0_DATA16	-	DISP0_DAT16	ALT1	IO
IPU2_DISP0_DATA17	-	DISP0_DAT17	ALT1	IO
IPU2_DISP0_DATA18	-	DISP0_DAT18	ALT1	IO
IPU2_DISP0_DATA19	-	DISP0_DAT19	ALT1	IO
IPU2_DISP0_DATA20	-	DISP0_DAT20	ALT1	IO
IPU2_DISP0_DATA21	-	DISP0_DAT21	ALT1	IO
IPU2_DISP0_DATA22	-	DISP0_DAT22	ALT1	IO

Table continues on the next page...

Table 37-7. IPU2 External Signals (continued)

Signal	Description	Pad	Mode	Direction
IPU2_DISP0_DATA23	-	DISP0_DAT23	ALT1	IO
IPU2_SISG0	-	NANDF_CS2	ALT6	O
IPU2_SISG1	-	NANDF_CS3	ALT6	O
IPU2_SISG2	-	EIM_A24	ALT3	O
IPU2_SISG3	-	EIM_A23	ALT3	O
IPU2_SISG4	-	NANDF_CLE	ALT1	O
IPU2_SISG5	-	NANDF_WP_B	ALT1	O

37.3 Clocks

The table found here describes the clock sources for IPU.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 37-8. IPU Clocks

Clock name	Clock Root	Description
hsp_clk	ipu1_ipu_hsp_clk_root	HSP clock
ipp_di_0_ext_clk	ipu1_di0_clk_root	IPU DI0 interface pixel clock
ipp_di_1_ext_clk	ipu1_di1_clk_root	IPU DI1 interface pixel clock
ipu_master_hclk	ahb_clk_root	IPU master clock

37.4 Functional Description

This section provides a complete functional description of the block.

37.4.1 IPU detailed block diagram

The following figure is the IPU top level block diagram.

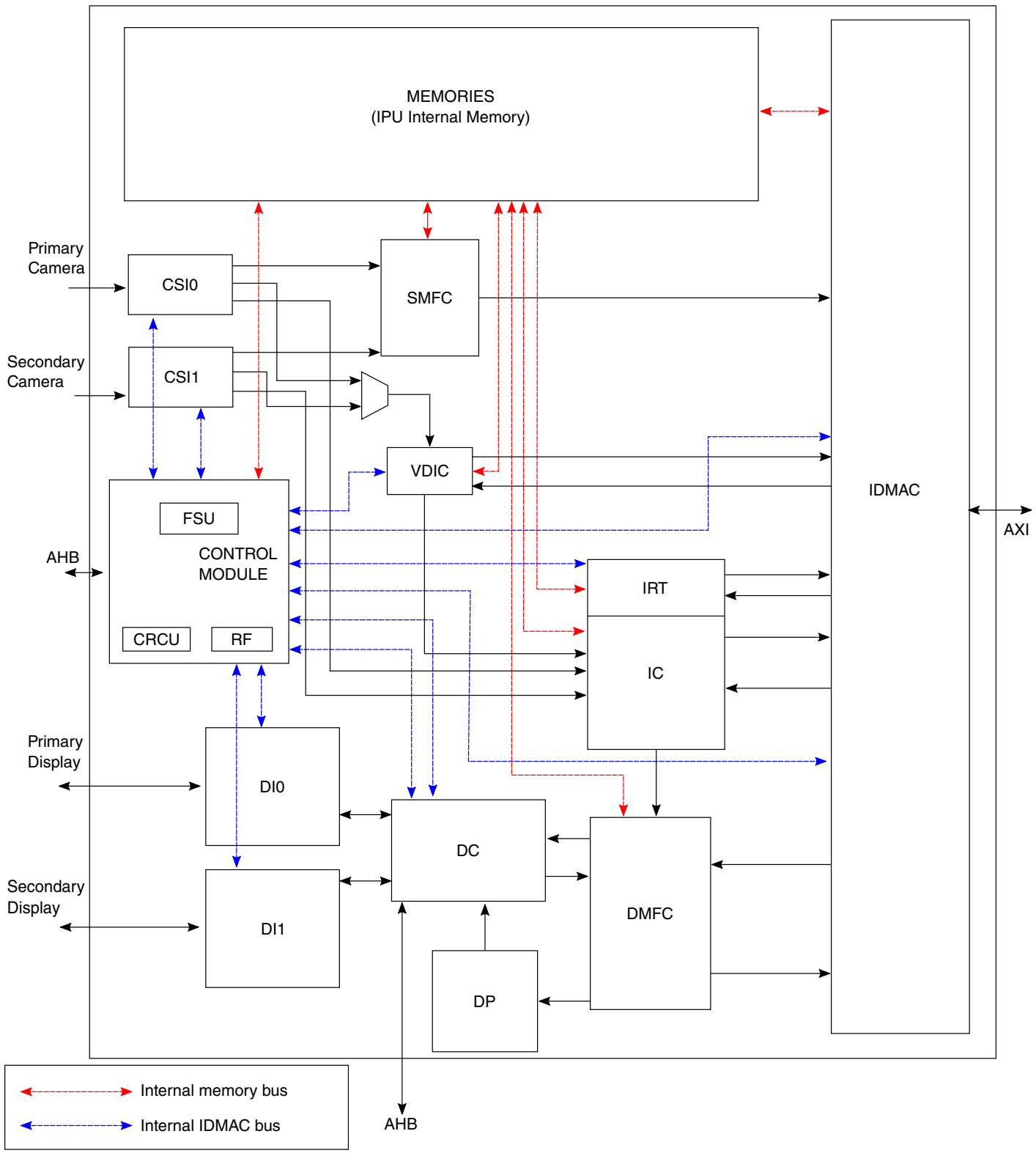


Figure 37-2. IPU Detailed Block Diagram

37.4.2 Image DMA Controller (IDMAC)

The following diagram is the IDMAC's block diagram.

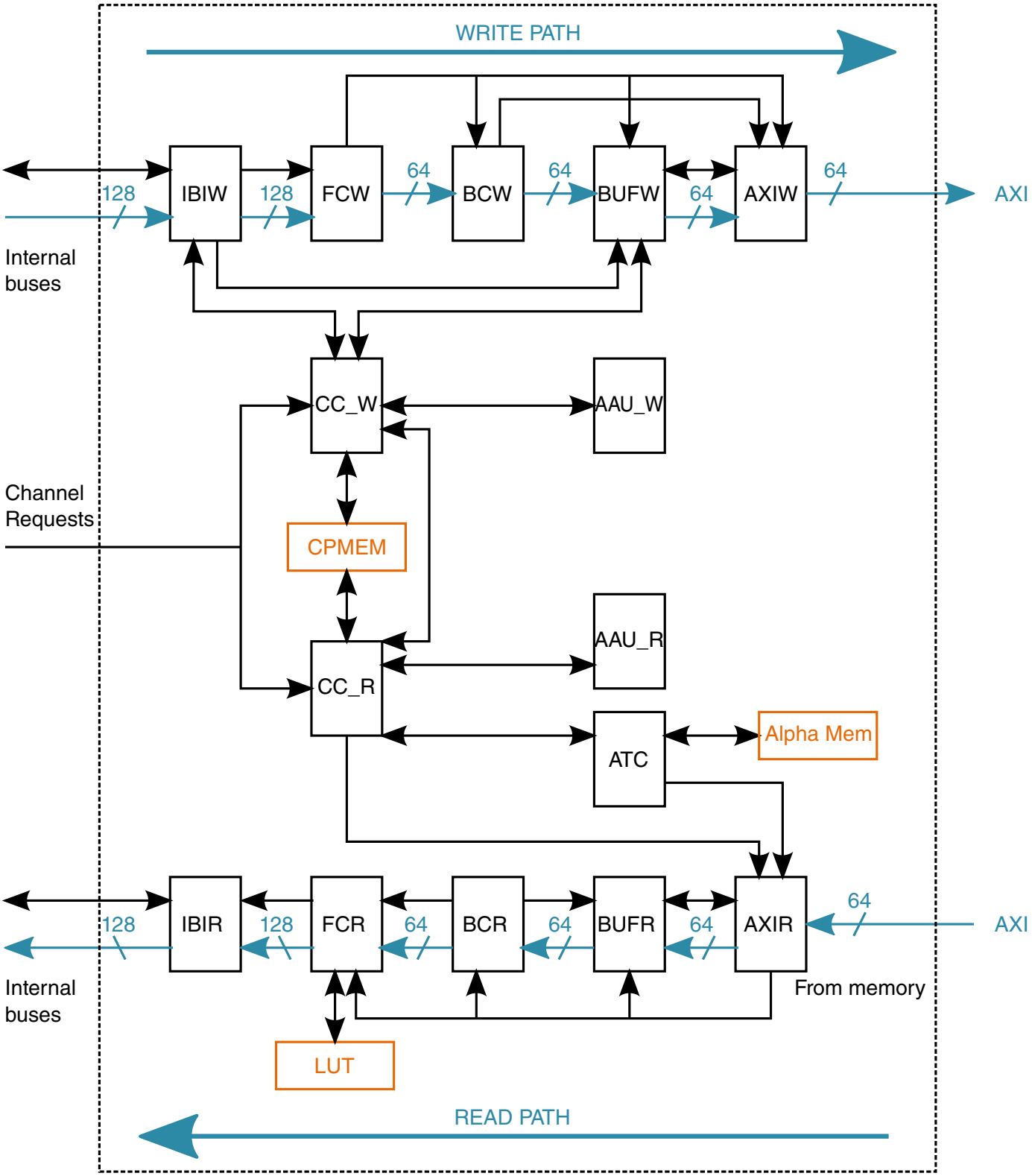


Figure 37-3. IDMAC Block Diagram

The following table describes the IDMAC's sub-block glossary

Table 37-9. IDMAC's sub modules glossary

Sub Module	Description
IBIW	Internal Bus Interface Write
IBIR	Internal Bus Interface Read
FCW	Format Converter Write
FCR	Format Converter Read
BCW	Buffer Controller Write
BCR	Buffer Controller Read
BUFW	Buffer Write
BUFR	Buffer Read
AXIW	AXI Write
AXIR	AXI Read
CC_W	Channel Control Write
CC_R	Channel Control Read
AAU_W	Address Arithmetic Unit Write
AAU_R	Address Arithmetic Unit Read
ATC	Alpha Transparency Controller
LUT	Look up table
CPMEM	Channel Parameter Memory

37.4.2.1 IDMAC's channels

The table below summarizes the IDMAC's channels.

Enabling a channel is done via the channel's corresponding IDMAC_CH_EN bit.

Table 37-10. IDMAC DMA channels list

Channel #	Source	Destination	Alternate Destination	Purpose	Data type
0	CSI (via SMFC)	Fmem		VF2 - Bayer; BPP>8; JPEG; MIPI additional channels	Generic or Pixel
1	CSI (via SMFC)	Fmem		VF2 - Bayer; BPP>8; JPEG; MIPI additional channels	Generic or Pixel
2	CSI (via SMFC)	Fmem		VF2 - Bayer; BPP>8; JPEG; MIPI additional channels	Generic or Pixel
3	CSI (via SMFC)	Fmem		VF2 - Bayer; BPP>8; JPEG; MIPI additional channels	Generic or Pixel
5	VDIC	Bmem	IC	VF1/VF2	Pixel
8	Fmem	VDIC		Previous field	Pixel
9	Fmem	VDIC		Current field	Pixel
10	Fmem	VDIC		Next field	Pixel

Table continues on the next page...

Table 37-10. IDMAC DMA channels list (continued)

Channel #	Source	Destination	Alternate Destination	Purpose	Data type
11	Bmem	IC		video plane for post processing task	Pixel
12	Bmem	IC		video plane for PrP tasks (view finder or encoding)	Pixel
13	VDIC	Fmem		Recent field from CSI	Pixel
14	Fmem	IC		graphics plane for PrP task (view finder or encoding)	Pixel
15	Fmem	IC		graphics plane for post processing task	Pixel
16	Reserved				
17	Fmem	IC		Transparency (alpha for channel 14)	Generic
18	Fmem	IC		Transparency (alpha for channel 15)	Generic
19	Fmem	VDIC		Transparency (alpha for channel 25)	Generic
20	IC	Bmem		Preprocessing data from IC (encoding task) to memory	Pixel
21	IC	Bmem	DMFC	Preprocessing data from IC (viewfinder task) to memory; This channel can be configured to send the data directly to the DMFC. This is done by programming the IC_DMFC_SEL bit.	Pixel
22	IC	Bmem		Postprocessing data from IC to memory	Pixel
23	Fmem	DP		DP primary flow - main plane	Pixel
24	Fmem	DP		DP secondary flow - main plane	Pixel
25	Fmem	VDIC		Plane #1 of the VDIC for combining	pixel
26	Fmem	VDIC		Plane #3 of the VDIC for combining	pixel
27	Fmem	DP		DP primary flow - auxiliary plane	Pixel
28	Fmem	DC		DC channel for both sync and async flows	Pixel
29	Fmem	DP		DP secondary flow - auxiliary plane	Pixel
30	Reserved				
31	Fmem	DP		Transparency (alpha for channel 27)	Generic
32	Reserved				
33	Fmem	DP		Transparency (alpha for channel 29)	Generic
34	Reserved				
35	Reserved				
36	Reserved				
37	Reserved				
38	Reserved				
39	Reserved				
40	DC	Fmem		DC read channel	Generic
41	Fmem	DC		DC async flow	Generic
42	Fmem	DC		DC command stream	Generic
43	Fmem	DC		DC command stream	Generic
44	Fmem	DC		DC output mask	Generic
45	Bmem	IRT		Rotation for post Encoding task	Pixel
46	Bmem	IRT		Rotation for viewfinder task	Pixel

Table continues on the next page...

Table 37-10. IDMAC DMA channels list (continued)

Channel #	Source	Destination	Alternate Destination	Purpose	Data type
47	Bmem	IRT		Rotation for post processing task	Pixel
48	IRT	Bmem		Rotation for Encoding task	Pixel
49	IRT	Bmem		Rotation for viewfinder task	Pixel
50	IRT	Bmem		Rotation for post processing task	Pixel
51	Fmem	DP		Transparency (alpha for channel 23)	Generic
52	Fmem	DP		Transparency (alpha for channel 24)	Generic
53-63	Reserved				

37.4.2.2 IBIW & IBIR - Internal bus interface for write and read

The Internal Bus Interface handles the internal IPU protocol communicating between the IDMAC and the IPU's sub modules.

The IBIR handles channels that perform read from external memory. The IBIW handles channels that perform write accesses to external memory.

37.4.2.3 FCW & FCR - Format converter write and read

The format converter performs packing ("write direction") / unpacking ("read" direction) of pixels with programmable position and width of color components, decoding 4- or 8-bits coded pixels according to a loaded look-up table, panning of an image read from the system memory according to a panning offset (start pixel address).

The format converter supports formats with a pixel width of 4, 8, 12, 16, 18, 24 or 32 bits. The format converter unit handles two pixels simultaneously.

The IPU sub modules can handle only the formats, presented below. Each component is 8 bit:

Figure 37-4. IPU internal pixel formats

The pixel can be stored in the memory in the formats presented below. (R/G/B means that this could component can be R or G or B; A is the location of the alpha component).

For Read:

R/G/B/A	R/G/B/A	R/G/B/A	R/G/B/A

For Write:

R/G/B	R/G/B	R/G/B	A

A	R/G/B	R/G/B	R/G/B

Figure 37-5. IPU external pixel formats

Formatting parameters are written in the channel parameter memory (CPMEM - Channel Parameter Memory). The following parameters are used:

- Offset OFS0 between MSB position of the color component 0 and MSB position of packed pixel. The color component 0 occupies the most significant bits of the unpacked pixel (mostly this is the R component). The OFS0 range is from 0 to 31.
- Color component 0 width (WID0 minus 1).
- Offset OFS1 between MSB position of the color component 1 and MSB position of packed pixel. The color component 1 occupies the middle left bits of the unpacked pixel (mostly this is the G component). The OFS1 range is from 0 to 31.
- Color component 1 width (WID1 minus 1).
- Offset OFS2 between MSB position of the color component 2 and MSB position of packed pixel. The color component 2 occupies the middle right bits of the unpacked pixel (mostly this is the B component). The OFS2 range is from 0 to 31.
- Color component 2 width (WID2 minus 1).
- Offset OFS3 between MSB position of the color component 3 and MSB position of packed pixel. The color component 3 occupies the least significant bits of the unpacked pixel (mostly this is the A component). The OFS3 range is from 0 to 31. For write specified DMA channels, the OFS3 value is set to 24 or 0 bits.
- In cases of read with separate alpha (alpha is located in a separate buffer in the system's memory than the pixel data), the alpha component size is defined according to WID3.

The figures below show examples of data packing and unpacking.

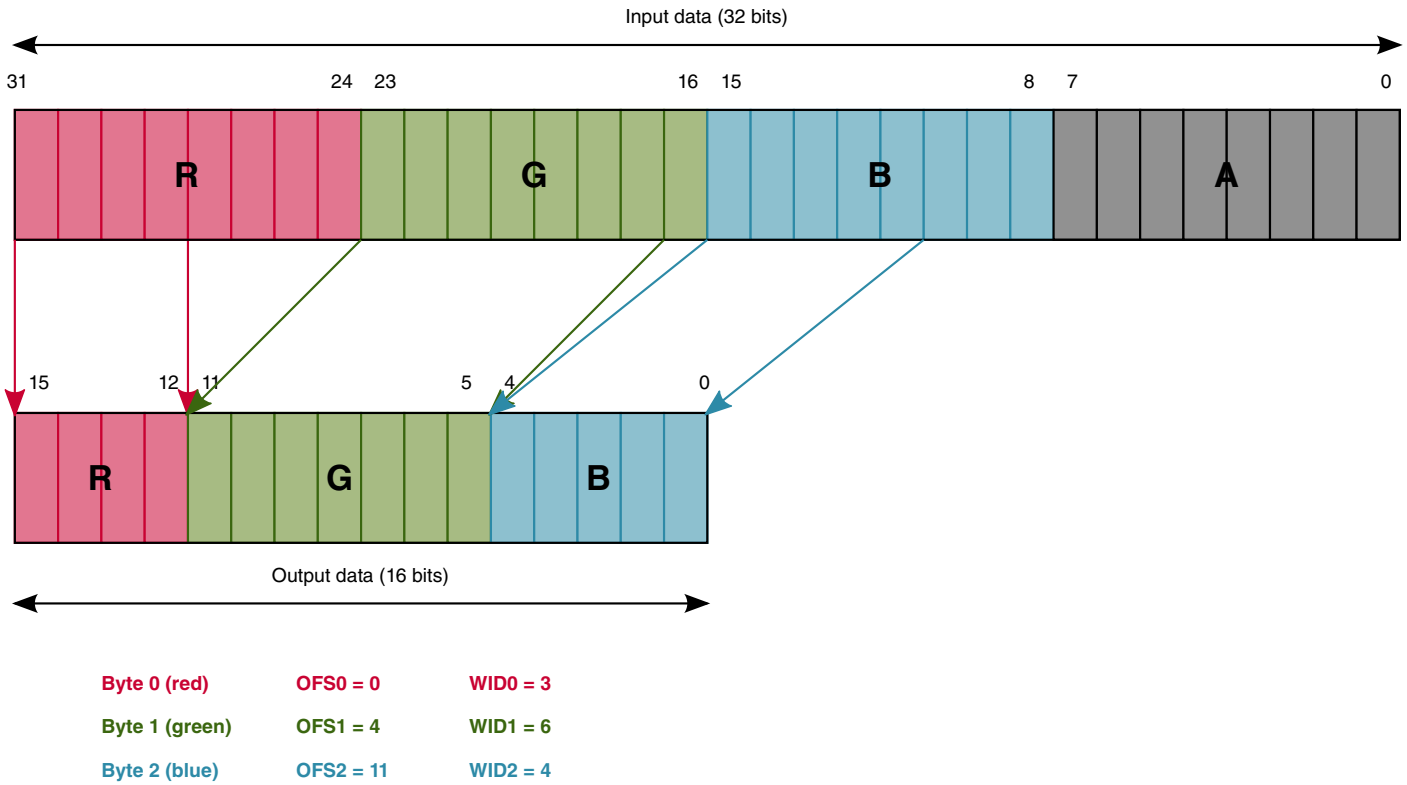


Figure 37-6. Data Packing Example

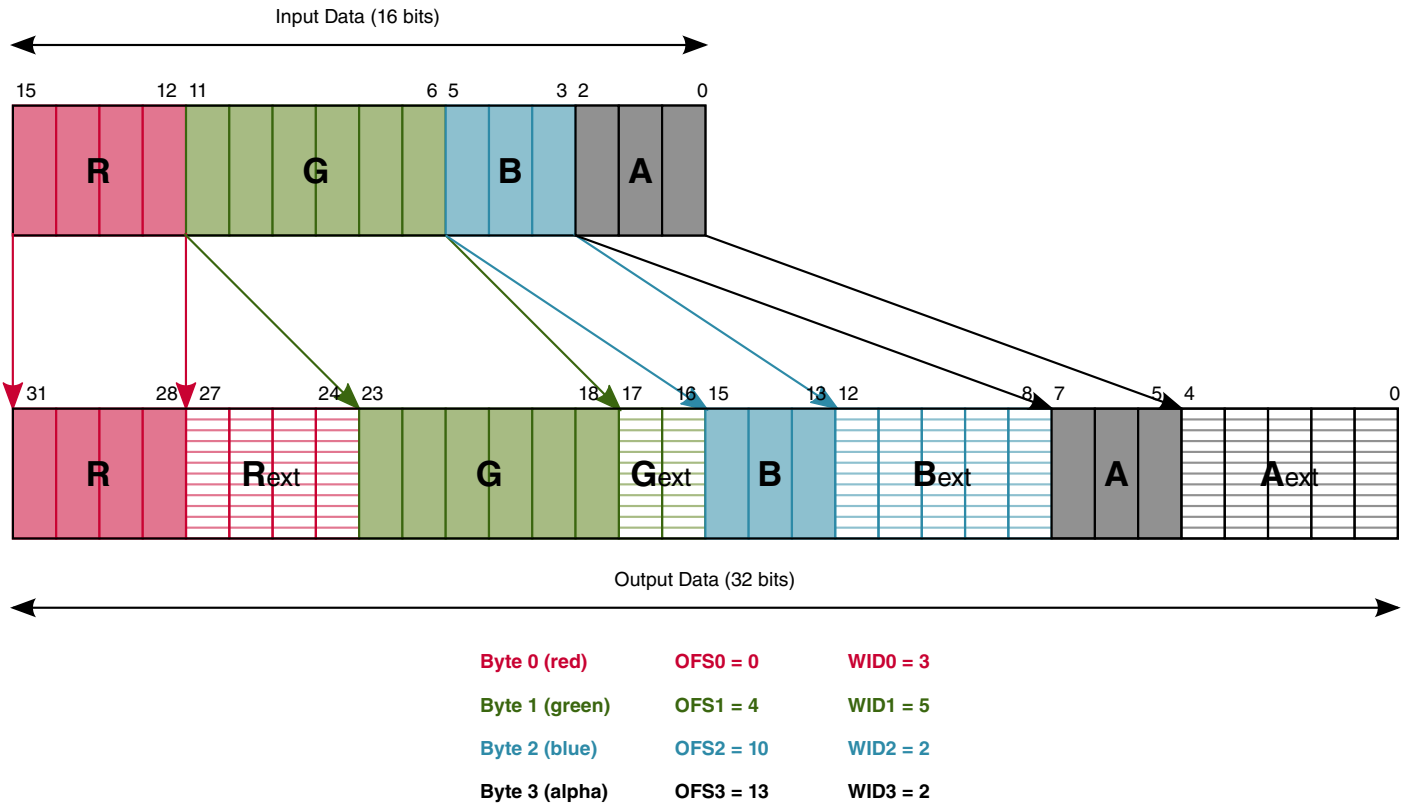


Figure 37-7. Data Unpacking Example

If read data has the coded format, it is decoded via the look-up table. The Look-Up Table Memory (LUT- Look Up Table) must be loaded at the IDMAC initialization step. The LUT output format must match the IPU internal format RGBA 8888 where R is placed in MSB and A is placed in LSB. The A field is used only for graphics data.

37.4.2.4 Buffering units

The buffering units (BUFW, BUFR) are used to store the data consisting of different coded color components.

- On write transactions (BUFW), before writing to memory & after the format has been coded.
- On read transactions (BUFR), after reading from memory & before the format has been decoded.

Each buffering unit includes 4 X 64 bytes buffers. Each buffer, for each direction, can handle any kind of color component (interleaved - Y / partial interleaved - Y, UV / non interleaved - Y, U, V).

The Write buffers are controlled by the buffer controller for write (BCW) and AXIW units.

The Read buffers are controlled by the buffer controller for read (BCR) and AXIR units.

37.4.2.4.1 Handling real time channels

The memory controller connected to the AXI bus of the IPU can use the AXI ID associated with each burst in order to distinguish between real time and non real time channels.

In order to do that, the user has to set the channel's ID according to the settings in the memory controller and set the priority of the channel according to its nature. The buffer controller (BCW/BCR) holds all the pending requests that won the arbitration. However, as the memory controller can distinguish between the real time channels and non real time channels within the IPU, there could be a situation where the real time requests are blocked as the IPU's queue is filled with non real time requests. To avoid that, the user can limit the number of non-real time requests in the queue.

The queue for read requests can handle up to 8 requests. The queue for write requests can handle up to 6 requests. The user can limit the number of non real time requests by setting the USED_BUFS_MAX_W for write requests and USED_BUFS_MAX_R for read requests. The feature that limits the number of requests is enabled by setting the USED_BUFS_EN_R bit for the read requests and USED_BUFS_EN_W for the write requests.

37.4.2.5 AXIW - AXI Write and AXIR - AXI Read

The AXI Master Interfaces are responsible for data transfer from/to the system memory. The Interface supports only 64-bits burst accesses of 1-8 words, with nonalignment of a byte resolution.

2 separate & independent AXI masters are used for "read" (AXIR) & "write" (AXIW), each can be programmed (via CPMEM) with 4 different IDs to support out-of-order accesses within bursts.

37.4.2.6 CC_W & CC_R - Channel Control Write and Read

The Channel Control unit is the main control unit of the IDMAC.

- It arbitrates the channels according to the priority.
- Controls the address arithmetic unit.

- Functions as a memory interface to the CPMEM. It reads the parameters from the CPMEM, prepares the controls accordingly and writes back updated parameters to the CPMEM.
- The read unit provides the parameters to the IBIR unit
- The write unit provides the parameters to the AXIW unit

The CC calculates all the parameters related to the access except the address and BS (burst size), which are calculated at the AAU.

The priority is set according to:

- The channel's corresponding bit in the IDMAC_CH_PRI_1 & IDMAC_CH_PRI_2 registers.
- The watermark signal generated from the sub module. The watermark signal is ignored unless the channel's corresponding IDMAC_WM_EN bit is set.
- Special priority for alpha channels

A priority value is calculated for each of the enabled channels according to the above conditions. Then, the CC unit selects between the channels with the same priority value in a round-robin fashion.

Table 37-11. Calculated priority value

alpha channel	Channel's priority bit	watermark signal	Priority Value
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	1
1	0	1	2
1	1	0	3
1	1	1	4

37.4.2.6.1 Locking the arbitration and reordering the AXI bursts

The performance of the overall system can be improved by sending AXI bursts of consecutive addresses. This can be done by reordering the AXI bursts in a way that accesses that belong to the same channel will be sent one after the other. This can be done by controlling the IDMAC_LOCK_# bits of the corresponding channel.

An IDMAC request that won the arbitration will be served for the next bursts according to the settings of the IDMAC_LOCK_# bits. The block that issued the request (DMFC on IPU) will assert the request only if it has enough room in its FIFO to accept the number

of bursts defined by the IDMAC_LOCK_# bits. IPU provides this capability to channels that serve real time screen refresh to synchronous display (23,27,28). In addition, it provides this capability to channels that may generate very short AXI bursts (IC and IRT)

37.4.2.7 AAU_W & AAU_R- Address Arithmetic Unit for Write and Read

The AAU_R & AAU_W units calculate the address in the system memory to be accessed by the IPU. These units also calculate the burst size (BS). The address calculation is done according to parameters stored in the CPMEM.

The following main addressing parameters are used:

- XB-Horizontal pixel position in frame
- YB-Vertical pixel position in frame
- SL-Stride line minus 1 (gap in bytes between two pixels in the same column in two consecutive rows).
- SX-Horizontal pixel scrolling offset
- SY-Vertical pixel scrolling offset
- EBA-Frame buffer base address in bytes (there are two such parameters to support double buffering)
- BPP-Bits per pixel
- FW-Frame width minus 1
- FH-Frame height minus 1

Relations between the addressing parameters and image frame are shown in the table below.

The system memory address in bytes is calculated as:

ADDR = EBA + (XB + SX) * BPP + (YB + SY) * (SL + 1)

with 0 < XB <= FW and 0 < YB <= FH.

For non-interleaved formats the 4 LSB bits of SX are defined according to the IOX parameter.

When double buffering is used, the EBA0 is the base address of the buffer 0 and the EBA1 is the base address of the buffer 1. The IPU_CHA_CUR_BUF Register is a status register. It contains 1-bit pointers to the current working buffers for all IPU DMA channels. The IPU automatically toggles a pointer after completion of the current buffer processing. If the ARM platform is a data source for specific double-buffered channel, it should check this status bit in order to know what is the IPU current buffer. The ARM platform is allowed to write to the buffer only when a working DMA channel does not

use it. After the ARM platform has been fill the buffer, it has to set the corresponding bit in the IPU_CHA_BUF0_RDY and IPU_CHA_BUF1_RDY Registers. If needed, the ARM platform can only clear the pointer by writing 1 but not set it.

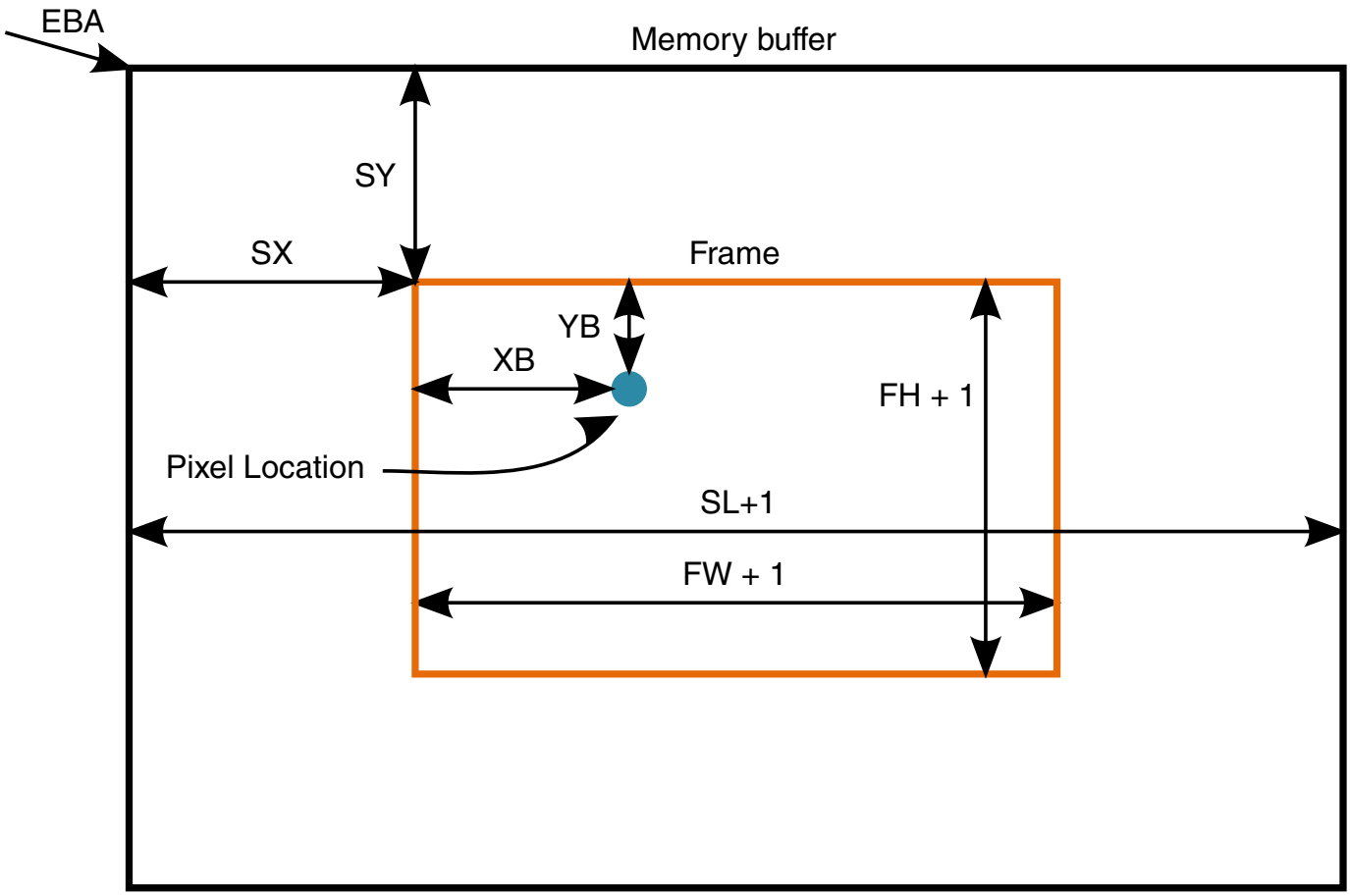


Figure 37-8. Addressing Parameters and Image Frame

The XB and YB coordinates are calculated according to addressing mode. There are two addressing modes:

- 2D mode
- Block mode

In 2D mode the pixel data is transferred to the memory row-by-row. There are two ways to use 2D mode: start from YB = 0 and finish at YB <= FH (YB is incremented) or start from YB = FH and finish at YB <= 0 (YB is decremented). The second option provides vertical flip of the image.

In block mode the frame is divided into blocks. This is needed for rotation or post-filtering, where the order used for data transfers is block-by-block. The order of the block transfer is according to the VF,HF and ROT bits in the IDMAC Channel Parameter

Memory. The order within the block is row-by-row where the block size is limited by the block width (BW) and block height (BH) parameters. The BW and BH parameters are set by IC rotation section and cannot be configured through the Channel Parameter Memory.

The Channel Control is responsible for the address calculation flow. It takes channel parameters from the Channel Parameter Memory, updates them and controls the Address Arithmetic Unit.

37.4.2.7.1 Scrolling support

Automatic display of a changing image (animation) or moving image (scrolling) is implemented by reading frames (from a background buffer) with incremental offset. Enabling the scrolling feature is done by setting the channel's corresponding SCE bit.

The scrolling step is controlled by channel's corresponding SDX and SDY parameters, and the scrolling direction is defined by the channel's corresponding SDRX and SDRY parameters. The maximum number of scrolled frames to be read is defined by the channel's corresponding SM parameter.

When the last programmed frame is reached (IDMAC's internal counter reached SM), IDMAC can perform one of the following (controlled by the SCC bit):

- Return to the first frame, without any SW intervention. The return point is defined by SX0 and SY0 parameters.
- Interrupt the ARM platform, to generate the next content.

37.4.2.8 ATC - Alpha Transparency Controller

The Alpha transparency controller (ATC) handles the alpha buffers on the external memory for cases where the pixel data and the alpha data are located on separate buffers (separate alpha mode).

In that case the IDMAC reads the alpha data and the pixel data, merge them together and provides a pixel that includes the alpha information to the relevant sub module. The ATC's main functions are:

- Generates requests for alpha channels, following a request to pixel data from the module.
- Maintain an internal alpha memory buffer for each channel.
- When there isn't enough alphas in the memory the ATC blocks the corresponding pixel channels.
- When there is a request of pixel channel the ATC load and accumulate its alphas in a register.

- ATC memory controller can manage 8 channels of alphas.
- ATC supports synchronous new frame before end of frame errors.

In order to configure the channel to use separate alpha the channel's corresponding IDMAC_SEP_AL bit should be set in addition to the ALU bit in the CPMEM of the corresponding channel.

The following figure is the ATC's block diagram.

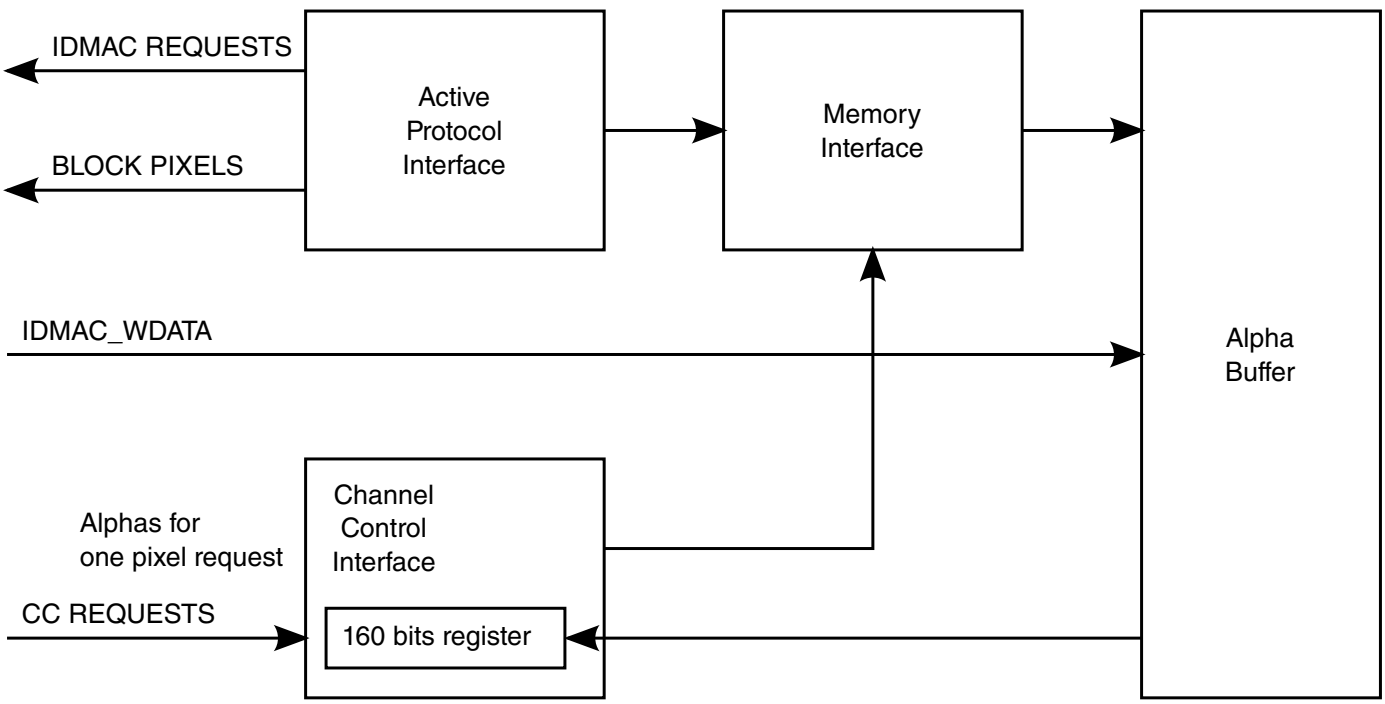


Figure 37-9. ATC block diagram

The ATC alpha buffer memory can hold up to 8 buffers of alphas. A pointer to a buffer in the ATC memory is defined according to the ALBM parameter in the CPMEM. The table below describes the relations between a data channel, an alpha channel and the pointer in the alpha buffer memory.

Table 37-12. Alpha channels mapping

data channel number	associated alpha channel number	Alpha buffer memory (ALBM)
14	17	0
15	18	1
27	31	2
29	33	3
23	51	4
24	52	5
25	19	6

37.4.2.8.1 Conditional read

The alpha data can be used to reduce reads from the memory of pixels that are going to be transparent (alpha = 0). The conditional read feature is enabled by the CRE bit in the CPMEM.

If all of the corresponding alpha values for a single burst of pixels are equal to zero, the IDMAC will block the access to the external memory and provide a data of all zeros to the corresponding channel. This way some of the accesses to the memory can be prevented, thus reducing the load on the memory.

37.4.2.9 LUT- Look Up Table

When working in coded pixel format, the data read from the memory is the decoded value of pixel according to address given. In case of 8 bit code, the data read from the memory is the decoded value of the pixel according to the address given.

In case of 4 bit code configuration, The address of the 4 bit decoded values is set according to DEC_SEL parameter in the CPMEM

00 = addresses 0 to 15

01 = addresses 64 to 79

10 = addresses 128 to 143

11 = addresses 192 to 207

Table 37-13. Look-Up Table Memory Structure

Address	Word	DEC_SEL	Description
0	Word0	4 BPP = 00 8 BPP = don't care	Decoded Pixels [15:0] for both 8 and 4 bit coded configuration
...	...		
15	Word15		
16	Word16	don't care	Decoded Pixels [63:16] for 8 bit coded configuration only
...	...		
63	Word63		
64	Word64	4 BPP = 01 8 BPP = don't care	Decoded Pixels [79:64] for both 8 and 4 bit coded configuration
...	...		
79	Word79		
80	Word80	don't care	Decoded Pixels [127:80] for 8 bit coded configuration only
...	...		
127	Word127		

Table continues on the next page...

Table 37-13. Look-Up Table Memory Structure (continued)

Address	Word	DEC_SEL	Description
128	Word128	4 BPP = 10 8 BPP = don't care	Decoded Pixels [143:128] for both 8 and 4 bit coded configuration
...	...		
143	Word143		
144	Word144	don't care	Decoded Pixels [191:144] for 8 bit coded configuration only
...	...		
191	Word191		
192	Word192	4 BPP = 11 8 BPP = don't care	Decoded Pixels [207:192] for both 8 and 4 bit coded configuration
...	...		
207	Word207		
208	Word208	don't care	Decoded Pixels [255:208] for 8 bit coded configuration only
...	...		
255	Word255		

37.4.2.10 CPMEM - Channel Parameter Memory

The CPMEM holds the configuration parameters for each IDMAC channel. The CPMEM can holds the settings of 80 channels.

Each channel's settings are defined by a two mega-words. Each mega-word is 160 bits wide. The following diagram illustrates the CPMEM's structure.

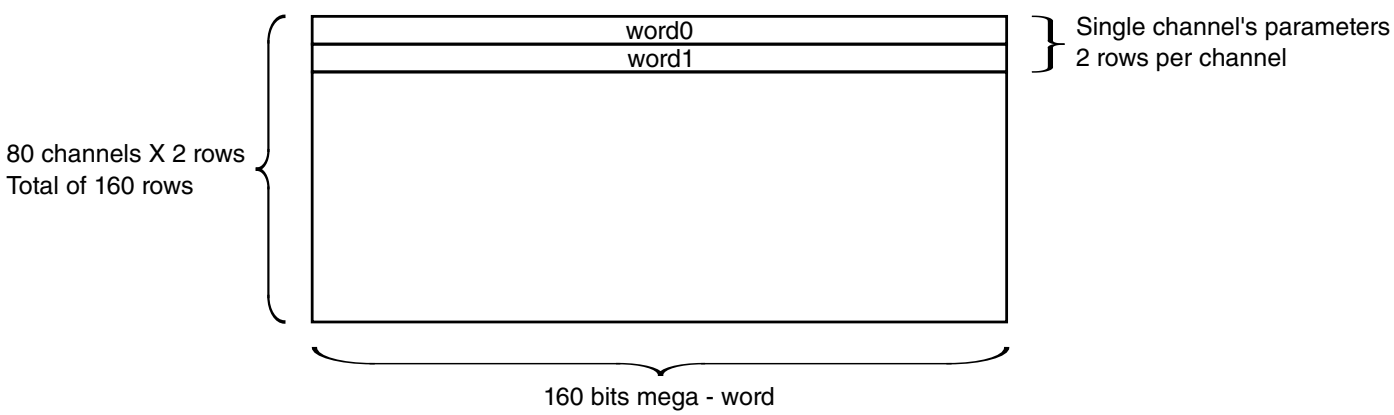


Figure 37-10. CPMEM structure

Each IDMAC channel can be configured to work in one of two different modes:

- Non Interleaved mode where the Y:U:V data is organized in 3 separate buffers in the system's memory
- Interleaved mode where the Y:U:V data is organized in a single buffer in the system's memory

The parameters and the way they are organized are different for each mode. The Pixel Format Select (PFS) value of the CPMEM words are used by the IPU to determine if the words should be interpreted as interleaved or non-interleaved format.

NOTE

CPMEM is "Memory Mapped" and can be accessed by the AHB bus through the Control Module (see Memory Access Unit).

The following tables describe the IDMAC parameters and their organization in each mode.

37.4.2.10.1 CPMEM's words' structure for non interleaved mode

The table below describes the CPMEM's words' structure for non interleaved mode. Each CPMEM word consist of 160 bits. The bits that are not listed in the table below are reserved bits.

Table 37-14. Channel Parameters Memory for non-interleaved

Name	Mnemonic	Size	Location	Description
Word 0				
XV Virtual Coordinate	XV	10 bits	W0[9:0]	Variable coordinates for determining next block address. {X1,Y1} and {X2,Y2} coordinates will be determined according to {XV,YV} upon restart of channel. These coordinates are used for Y:U:V (Y pointer) and RGB formats.
YV Virtual Coordinate	YV	9 bits	W0[18:10]	
XB inner Block Coordinate	XB	13 bits	W0[31:19]	Variable coordinates for determining address within the block. These coordinates are used for Y:U:V (Y pointer) and RGB formats. Need 24 bits for 2D transfer support.
YB inner Block Coordinate	YB	12 bits	W0[43:32]	
New Sub Block	NSB_B	1 bit	W0[44]	This bit determines if the next value for {XB,YB} should be taken from {XB,YB} saved in channel parameter memory or from new {x1,y1}/{x2,y2}.
Current Field	CF	1 bit	W0[45]	CF = 0 Current field is even CF = 1 Current field is odd
Mem U Buffer Offset	UBO	22 bits	W0[67:46]	Double buffer destination address offset for Y:U:V (U pointer) formats.The actual physical address value is divided by 8 (i.e. this parameter includes bits [24:3] of the actual address)

Table continues on the next page...

Table 37-14. Channel Parameters Memory for non-interleaved (continued)

Mem V Buffer Offset	VBO	22 bits	W0[89:68]	Double buffer destination address offset for Y:U:V (V pointer) format. The actual physical address value is divided by 8 (i.e. this parameter includes bits [24:3] of the actual address)
Initial Offset X	IOX	4 bits	W0[93:90]	The IOX parameter, is the offset in pixels for a frame that starts at a non aligned address. for 42x formats must be even.
Reduce Double Read or Writes	RDRW	1 bits	W0[94:94]	This bit is relevant for YUV4:2:0 formats. For read channels: U and V components are not read from odd rows. (read - supported only for the VDIC) For write channels: U and V components are not written to odd rows. (write - supported for all write channels)
Scan Order	SO	1 bit	W0[113]	SO = 0 Scan order is progressive SO = 1 Scan order is interlaced
Band Mode	BNDM	3 bits	W0[116:114]	BNDM = 000 bands disable. BNDM = 001 bands enable. Band height = 4 lines. BNDM = 010 bands enable. Band height = 8 lines. BNDM = 011 bands enable. Band height = 16 lines. BNDM = 100 bands enable. Band height = 32 lines. BNDM = 101 bands enable. Band height = 64 lines. BNDM = 110 bands enable. Band height = 128 lines. BNDM = 111 bands enable. Band height = 256 When working in band mode, the channel's corresponding IDMAC_BNDM_EN bit has to be set.
Block Mode	BM	2 bits	W0[118:117]	BM = 00 block mode disable. BW = FW, BH = FH BM = 01 block mode enable. BW = 8, BH = 8 BM = 10 block mode enable. BW = 16, BH = 16 (this mode is reserved for future use) BM = 11 not used
Rotation	ROT	1 bit	W0[119]	ROT = 0 -> No rotation ROT = 1 -> 90 degree rotation clockwise
Horizontal Flip	HF	1 bit	W0[120]	HF = 0 -> No flip HF = 1 -> Horizontal flip enable
Vertical Flip	VF	1 bit	W0[121]	VF = 0 -> No flip VF = 1 -> Vertical flip enable

Table continues on the next page...

Table 37-14. Channel Parameters Memory for non-interleaved (continued)

Threshold Enable	THE	1 bit	W0[122]	THE = 0 -> Threshold disable THE = 1 -> Threshold enable
Conditional Access Polarity	CAP	1 bit	W0[123]	CAP = 0 -> If conditional bit in CM register is low skip the access CAP = 1 -> If conditional bit in CM register is high skip the access. This mode is reserved for future use.
Conditional Access Enable	CAE	1 bit	W0[124]	CAE = 0 -> Conditional access disable CAE = 1 -> Conditional access enable This mode is reserved for future use.
Frame Width	FW	13 bits	W0[137:125]	Number of pixels in one row, of the channel frame. FW 000000000000 = 0001 pixels 000000000001 = 0002 pixels ..... 111111111111 = 8192 pixels
Frame Height	FH	12 bits	W0[149:138]	Number of pixels in one column, of the channel frame. FH 000000000000 = 0001 line 000000000001 = 0002 lines ..... 111111111111 = 4096 lines For progressive YUV 4:2:0 (non interleaved and partial interleaved formats) the FH value should be a multiple of 2. For interlaced YUV 4:2:0 (non interleaved and partial interleaved formats) the FH value should be a multiple of 4.
End of Line interrupt	EOLI	1 bit	W0[150]	End of line interrupt enable. The end of line indication is asserted once the last data of the line has been written. 1 - generate an end of line interrupt 0 - no affect
Word 1				
Ext Mem Buffer 0 Address	EBA0	29 bits	W1[28:0]	1st double buffer destination address for RGB and Y:U:V (Y pointer) formats. This parameter must not be changed if the corresponding channel is enabled. The actual physical address value is divided by 8 (i.e. this parameter includes bits [31:3] of the actual address)

Table continues on the next page...

Table 37-14. Channel Parameters Memory for non-interleaved (continued)

Ext Mem Buffer 1 Address	EBA1	29 bits	W1[57:29]	2nd double buffer destination address for RGB and Y:U:V (Y pointer) formats. This parameter must not be changed if the corresponding channel is enabled. The actual physical address value is divided by 8 (i.e. this parameter includes bits [31:3] of the actual address)
Interlace Offset	ILO	20 bits	W1[77:58]	2nd double buffer destination address for RGB and Y:U:V (Y pointer) formats. An interlaced data stored in the memory can be read as a consecutive progressive only if FW*BPP is a multiplication of 8. The actual physical address value is divided by 8 (i.e. this parameter includes bits [22:3] of the actual address). For YUV420 formats, the ILO is relevant only to the Y component as the U and V components do not exist for the even lines. This value is signed
Number of Pixels in Whole Burst Access	NPB	7 bits	W1[84:78]	Number of pixels per burst access. The following are valid numbers of pixels in a memory burst access according to the BPP parameter: NPB 0000111 = 08 pixels in each burst 0001111 = 16 pixels in each burst ..... 0111111 = 64 pixels in each burst Range: 16BPP => 1 -> 32 pixels (for YUV444 NI) 08BPP => 1 -> 64 pixels (For YUV420 NI/PI and YUV422 NI/PI) In NI/PI formats the NPB has to be a multiplication of 8
Pixel Format Select	PFS	4 bits	W1[88:85]	4'h0 = non-interleaved 4:4:4 4'h1 = non-interleaved 4:2:2 4'h2 = non-interleaved 4:2:0 4'h3 = partial interleaved 4:2:2 4'h4 = partial interleaved 4:2:0 4'h5 to 4'hF = NA
Alpha Used	ALU	1 bit	W1[89]	1 = the alpha associated with the data of this channel resides on another channel (separate buffer) 0 = the alpha associated with the data of this channel resides along with the pixel data (same buffer) The corresponding alpha channel must be enabled to assure correct behavior.

Table continues on the next page...

Table 37-14. Channel Parameters Memory for non-interleaved (continued)

Alpha Channel Mapping	ALBM	2 bits	W1[92:90]	Alpha channel mapping - This parameter is a pointer to a buffer in the ATC memory. This parameter is relevant only to data channels that are associated with a separate alpha buffer (like graphic plane channels). The parameter should be programmed on the data channels' ALBM. Setting this parameter to any other channel has no meaning. See Table 37-12 for exact ALBM mapping.
AXI Id	ID	2 bits	W1[94:93]	AXI protocol id
Threshold	TH	7 bits	W1[101:95]	0000000 = 32 lines 0000001 = 64 lines ..... 1111111 = 4096 lines
Stride Line	SLY	14 bits	W1[115:102]	Address vertical scaling factor in bytes for memory access. Also number of maximum bytes in the "Y" component row according to memory limitations. SLY 00000000000000 = 00001 bytes 00000000000001 = 00002 bytes ..... 11111111111111 = 16384 bytes
Width3	WID3	3 bits	W1[127:125]	Fourth color component size of the input-unpacking/ output-packing pixel. WID3 000 = 1 bits 001 = 2 bits ..... 111 = 8 bits As this is a non-interleaved format, this field is relevant only to the alpha associated with this pixel channel.
Stride Line	SLUV	14 bits	W1[141:128]	Address vertical scaling factor in bytes for memory access. Also number of maximum bytes in the "U" or "V" component row according to memory limitations. SLUV 00000000000000 = 00001 bytes 00000000000001 = 00002 bytes ..... 11111111111111 = 16384 bytes
Conditional Read Enable	CRE	1 bit	W1[149:149]	This bit enables the conditional read feature.

37.4.2.10.2 CPMEM's words' structure for interleaved mode

The table below describes the CPMEM's words' structure for interleaved mode. Each CPMEM word consist of 160 bits. The bits that are not listed in the table below are reserved bits

Table 37-15. Channel Parameters Memory for interleaved

Name	Mnemonic	Size	Location	Description
Word 0				
XV Virtual Coordinate	XV	10 bits	W0[9:0]	Variable coordinates for determining next block address. {X1,Y1} and {X2,Y2} coordinates will be determined according to {XV,YV} upon restart of channel. These coordinates are used for Y:U:V (Y pointer) and RGB formats.
YV Virtual Coordinate	YV	9 bits	W0[18:10]	
YB inner Block Coordinate	XB	13 bits	W0[31:19]	Variable coordinates for determining address within the block. These coordinates are used for Y:U:V (Y pointer) and RGB formats. Need 24 bits for 2D transfer support.
XB inner Block Coordinate	YB	12 bits	W0[43:32]	
New Sub Block	NSB_B	1 bit	W0[44]	This bit determines if the next value for {XB,YB} should be taken from {XB,YB} saved in channel parameter memory or from new {x1,y1}/{x2,y2}.
Current Field	CF	1 bit	W0[45]	CF = 0 Current field is even CF = 1 Current field is odd
Scroll X counter	SX	12 bits	W0[57:46]	Holds the temporary count for the Scroll X in between frame For interleaved YUV4:2:2 formats the SX should be a multiple of 2.
Scroll Y counter	SY	11 bits	W0[68:58]	Holds the temporary count for the Scroll Y in between frame
Number of Scroll	NS	10 bits	W0[78:69]	This variable holds the total number of Scrolls
Scroll Delta X	SDX	7 bits	W0[85:79]	Frame start row offset, compared to last frame. SDX 0000000 = 00 pixels 0000001 = 01 pixels 0000010 = 02 pixels ..... 1111110 = 126 pixels 1111111 = 127 pixels For interleaved YUV4:2:2 formats the SDX should be a multiple of 2.

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

Scroll Max	SM	10 bits	W0[95:86]	Frame maximum row and column increment offset in frame. SM 0000000000 = 0001 0000000001 = 0002 ..... 1111111111 = 1024
Scrolling Configuration	SCC	1 bit	W0[96]	Determines if scrolling will continue from zero when NS counter has reached SM or stop at the current value for SX and SY for the next frames to come. SCC 0 => Scrolling will stop at NS = SM 1 => Scrolling will start from "0" at NS = SM
Scrolling Enable	SCE	1 bit	W0[97]	SCE = 0 Scrolling disable SCE = 1 Scrolling enable
Scroll Delta Y	SDY	7 bits	W0[104:98]	Frame start column offset, compared to last frame. SDY 0000000 = 00 pixels 0000001 = 01 pixels 0000010 = 02 pixels ..... 1111110 = 30 pixels 1111111 = 127 pixels
Scroll Horizontal Direction	SDRX	1 bit	W0[105]	Determines if the next frame will move right or left compared to the current frame. SDRX 0 => Next frame will be right of current 1 => Next frame will be left of current
Scroll Vertical Direction	SDRY	1 bit	W0[106]	Determines if the next frame will move down or up compared to the current frame. SDRY 0 => Next frame will be down of current 1 => Next frame will be up of current
Bits Per Pixel	BPP	3 bits	W0[109:107]	3'h0 = 32 Bits per pixel 3'h1 = 24 Bits per pixel 3'h2 = 18 Bits per pixel

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

				3'h3 = 16 Bits per pixel 3'h4 = 12 Bits per pixel 3'h5 = 08 Bits per pixel 3'h6 = 04 Bits per pixel
Decode Address Select	DEC_SEL	2 bits	W0[111:110]	Upon 4BPP, selects between two look-up tables DEC_SEL 00 = addresses 0 to 15 01 = addresses 64 to 79 10 = addresses 128 to 143 11 = addresses 192 to 207
Access Dimension	DIM	1 bit	W0[112]	DIM = 0 Access Dimension is 2d DIM = 1 Access Dimension is 1d
Scan Order	SO	1 bit	W0[113]	SO = 0 Scan order is progressive SO = 1 Scan order is interlaced
Band Mode	BNDM	3 bits	W0[116:114]	BNDM = 000 bands disable. BNDM = 001 bands enable. Band height = 4 lines. BNDM = 010 bands enable. Band height = 8 lines. BNDM = 011 bands enable. Band height = 16 lines. BNDM = 100 bands enable. Band height = 32 lines. BNDM = 101 bands enable. Band height = 64 lines. BNDM = 110 bands enable. Band height = 128 lines. BNDM = 111 bands enable. Band height = 256 When working in band mode, the channel's corresponding IDMAC_BNDM_EN bit has to be set.
Block Mode	BM	2 bits	W0[118:117]	BM = 00 block mode disable. BW = FW, BH = FH BM = 01 block mode enable. BW = 8, BH = 8 BM = 10 block mode enable. BW = 16, BH = 16 (this mode is reserved for future use) BM = 11 not used
Rotation	ROT	1 bit	W0[119]	ROT = 0 -> No rotation ROT = 1 -> 90 degree rotation clockwise

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

Horizontal Flip	HF	1 bit	W0[120]	HF = 0 -> No flip HF = 1 -> Horizontal flip enable
Vertical Flip	VF	1 bit	W0[121]	VF = 0 -> No flip VF = 1 -> Vertical flip enable
Threshold Enable	THE	1 bit	W0[122]	THE = 0 -> Threshold disable THE = 1 -> Threshold flip enable
Conditional Access Polarity	CAP	1 bit	W0[123]	CAP = 0 -> If conditional bit in CM register is low skip the access CAP = 1 -> If conditional bit in CM register is high skip the access
Conditional Access Enable	CAE	1 bit	W0[124]	CAE = 0 -> Conditional access disable CAE = 1 -> Conditional access enable
Frame Width	FW	13 bits	W0[137:125]	Number of pixels in one row, of the channel frame. FW 0000000000000 = 0001 pixels 0000000000001 = 0002 pixels ..... 1111111111111 = 8192 pixels For interleaved YUV4:2:2 formats the FW should be a multiple of 2.
Frame Height	FH	12 bits	W0[149:138]	Number of pixels in one column, of the channel frame. FH 000000000000 = 0001 line 000000000001 = 0002 lines ..... 111111111111 = 4096 lines
End of Line interrupt	EOLI	1 bit	W0[150]	End of line interrupt enable. The end of line indication is asserted once the last data of the line has been written. 1 - generate an end of line interrupt 0 - no affect
Word 1				
Ext Mem Buffer 0 Address	EBA0	29 bits	W1[28:0]	1st double buffer destination address for RGB and Y:U:V (Y pointer) formats. This parameter must not be changed if the corresponding channel is enabled.The actual physical address value is divided by 8 (i.e. this parameter includes bits [31:3] of the actual address)

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

Ext Mem Buffer 1 Address	EBA1	29 bits	W1[57:29]	2nd double buffer destination address for RGB and Y:U:V (Y pointer) formats. This parameter must not be changed if the corresponding channel is enabled.The actual physical address value is divided by 8 (i.e. this parameter includes bits [31:3] of the actual address)
Interlace Offset	ILO	20 bits	W1[77:58]	2nd double buffer destination address for RGB and Y:U:V (Y pointer) formats. An interlaced data stored in the memory can be read as a consecutive progressive only if FW*BPP is a multiplication of 8.The actual physical address value is divided by 8 (i.e. this parameter includes bits [22:3] of the actual address). This value is signed
Number of Pixels in Whole Burst Access	NPB	7 bits	W1[84:78]	Number of pixels per burst access. The following are valid numbers of pixels in a memory burst access according to the BPP parameter: NPB 0000000 = 01 pixels in each burst 0000001 = 02 pixels in each burst ..... 1111111 = 128 pixels in each burst Range: 32BPP => 1 -> 16 pixels 24BPP => 1 -> 20 pixels 16BPP => 1 -> 32 pixels 12BPP => 1 -> 40 pixels 08BPP => 1 -> 64 pixels 04BPP => 1 -> 128 pixels
Pixel Format Select	PFS	4 bits	W1[88:85]	4'h0 to 4'h4 = NA 4'h5 = Code (LUT) 4'h6 = Generic data 4'h7 = RGB (& also YUV interleaved 4:4:4) 4'h8 = interleaved 4:2:2 Y1U1Y2V1[1] 4'h9 = interleaved 4:2:2 Y2U1Y1V1[2] 4'hA = interleaved 4:2:2 U1Y1V1Y2[3] 4'hB = interleaved 4:2:2 U1Y2V1Y1[4] 4'hC to 4'hF = NA

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

Alpha Used	ALU	1 bit	W1[89]	1 = the alpha associated with the data of this channel resides on another channel (separate buffer) 0 = the alpha associated with the data of this channel resides along with the pixel data (same buffer) The corresponding alpha channel must be enabled to assure correct behavior.
Alpha Channel Mapping	ALBM	3 bits	W1[92:90]	Alpha channel mapping - This parameter is a pointer to a buffer in the ATC memory. This parameter is relevant only to data channels that are associated with a separate alpha buffer (like graphic plane channels). The parameter should be programmed on the data channels' ALBM. Setting this parameter to any other channel has no meaning. See Table 37-12 for exact ALBM mapping.
AXI Id	ID	2 bits	W1[94:93]	AXI protocol id; IPU is targeted to an AXI slave that can handle up to 2 requests with 2 different IDs + one request with a third ID. In case that IPU is going to be used on a system that can handle more than 2 requests with different IDs, the number of different IDs programmed in the CPMEM for different channels is limited for 2. This limitation is relevant for read channels only. For write channels there's no such limitation
Threshold	TH	7 bits	W1[101:95]	0000000 = 32 lines 0000001 = 64 lines ..... 1111111 = 4096 lines
Stride Line	SL	14 bits	W1[115:102]	Address vertical scaling factor in bytes for memory access. Also number of maximum bytes in row according to memory limitations. SL 00000000000000 = 00001 bytes 00000000000001 = 00002 bytes ..... 11111111111111 = 16384 bytes
Width0	WID0	3 bits	W1[118:116]	First color component size of the input-unpacking/ output-packing pixel. WID0 000 = 1 bits

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

				001 = 2 bits ..... 111 = 8 bits This field is relevant only for interleaved RGB format (PFS = 4'h7)
Width1	WID1	3 bits	W1[121:119]	Second color component size of the input-unpacking/ output-packing pixel. WID1 000 = 1 bits 001 = 2 bits ..... 111 = 8 bits This field is relevant only for interleaved RGB format (PFS = 4'h7)
Width2	WID2	3 bits	W1[124:122]	Third color component size of the input-unpacking/ output-packing pixel. WID2 000 = 1 bits 001 = 2 bits ..... 111 = 8 bits This field is relevant only for interleaved RGB format (PFS = 4'h7)
Width3	WID3	3 bits	W1[127:125]	Fourth color component size of the input-unpacking/ output-packing pixel. WID3 000 = 1 bits 001 = 2 bits ..... 111 = 8 bits This field is relevant only for interleaved RGB format (PFS = 4'h7)
Offset0	OFS0	5 bits	W1[132:128]	Number of bits between MSB of pixel and MSB of color component, on input. 1 states that the color component will be the first color component aligned to MSB of output pixel. OFS0 00000 = No offset 00001 = u => 1 bit sleft, p => 1 bit sright .....

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

				11111 = u => 31 bit sleft, p => 31 bit sright * u = unpacking * p = packing * sleft = shift left * sright = shift right This field is relevant only for interleaved RGB format (PFS = 4'h7)
Offset1	OFS1	5 bits	W1[137:133]	Number of bits between MSB of pixel and MSB of color component on input. 2 states that the color component will be the second color component aligned to MSB output pixel. OFS1 00000 = No offset 00001 = u => 1 bit sleft, p => 1 bit sright ..... 11111 = u => 31 bit sleft, p => 31 bit sright * u = unpacking * p = packing * sleft = shift left * sright = shift right This field is relevant only for interleaved RGB format (PFS = 4'h7)
Offset2	OFS2	5 bits	W1[142:138]	Number of bits between MSB of pixel and MSB of color component on input. 3 states that the color component will be the third color component aligned to MSB output pixel. OFS2 00000 = No offset 00001 = u => 1 bit sleft, p => 1 bit sright ..... 11111 = u => 31 bit sleft, p => 31 bit sright * u = unpacking * p = packing * sleft = shift left * sright = shift right

Table continues on the next page...

Table 37-15. Channel Parameters Memory for interleaved (continued)

				This field is relevant only for interleaved RGB format (PFS = 4'h7)
Offset3	OFS3	5 bits	W1[147:143]	Number of bits between MSB of pixel and MSB of color component on input. 4 states that the color component will be the fourth color component aligned to MSB output pixel. OFS3 00000 = No offset 00001 = u => 1 bit sleft, p => 1 bit sright ..... 11111 = u => 31 bit sleft, p => 31 bit sright * u = unpacking * p = packing * sleft = shift left * sright = shift right This field is relevant only for interleaved RGB format (PFS = 4'h7)
Select SX SY Set	SXYS	1 bit	W1[148:148]	This bit selects between the settings on: SC_CORD and SC_CORD1
Conditional Read Enable	CRE	1 bit	W1[149:149]	This bit enables the conditional read feature.
Decode Address Select bit[2]	DEC_SEL2	1 bit	W1[150:150]	This field is reserved

1. Y1U1Y2V1 means byte0 = bits [7:0] =Y1; byte1 = bits [15:8] =U1; byte2 = bits [23:16] =Y2; byte3 = bits [31:24] = V1
2. Y2U1Y1V1 means byte0 =Y2; byte1 =U1; byte2 =Y1; byte3 = V1
3. U1Y1V1Y2 means byte0 =U1; byte1 =Y1; byte2 =V1; byte3 = Y2
4. U1Y2V1Y1 means byte0 =U1; byte1 =Y2; byte2 =V1;byte3 = Y1

37.4.2.10.3 Accessing the CPMEM for programming

Each IDMAC's channel's parameters are located on 2 CPMEM entries. Each Entry is 160 bit. The CPMEM is memory mapped and is accessible via the AHB bus. The AHB bus's accesses are 32bit wide. A CPMEM entry is composed of 5x32bit words. The next CPMEM entry starts at the next 8x32bit words (0x0, 0x20,0x40, etc.).

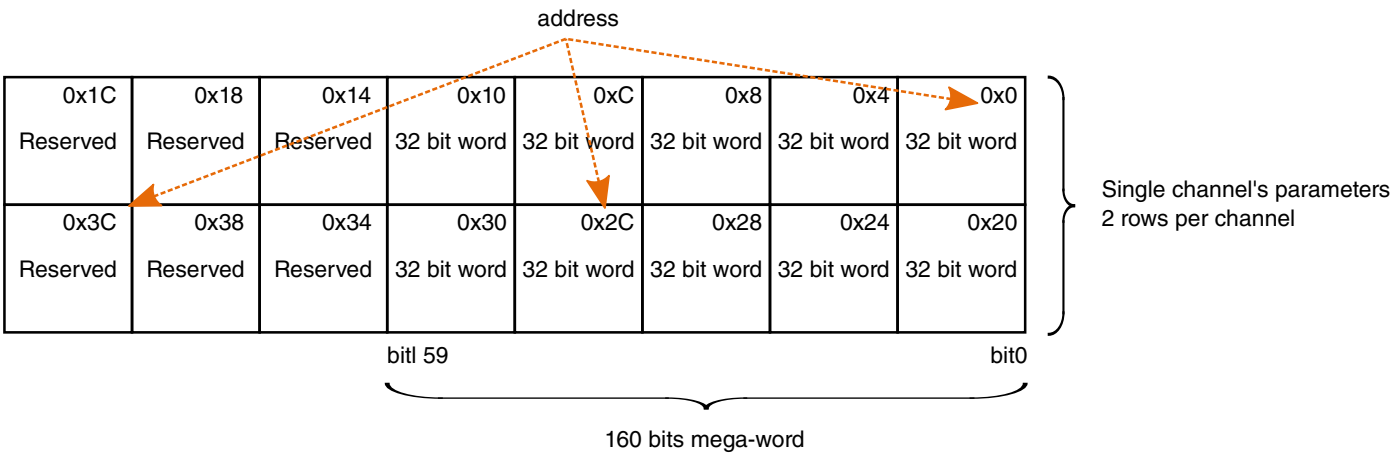


Figure 37-11. CPMEM's word structure

37.4.2.10.4 Alternate IDMAC settings

Some of the IDMAC channels support alternate flow. This means that a physical IDMAC channel can use alternate set of parameters. Switching between the flows is controlled by the CM.

The primary flow the IDMAC's settings are read from the channel's corresponding entry in the CPMEM. The alternate settings can be stored in another entry on the CPMEM. The pointer to the alternate entry on the CPMEM is stored on the physical channel's corresponding IDMAC_SUB_ADDR parameters.

37.4.2.11 IDMAC's modes of operation

37.4.2.11.1 Rotation modes

Rotation is performed by the IDMAC and the Rotation unit inside the IC.

The frame is partitioned into 8X8 pixels blocks. The IC reorders the pixels within a block. The IDMAC reorders the block. The reordering is done according to the ROT, VF & HF parameters in the CPMEM.

The following diagram illustrate various options for reordering of blocks.

- ROTATE means that the ROT bit is set
- HORIZONTAL means that the HF bit is set
- VERTICAL means that the VF bit is set

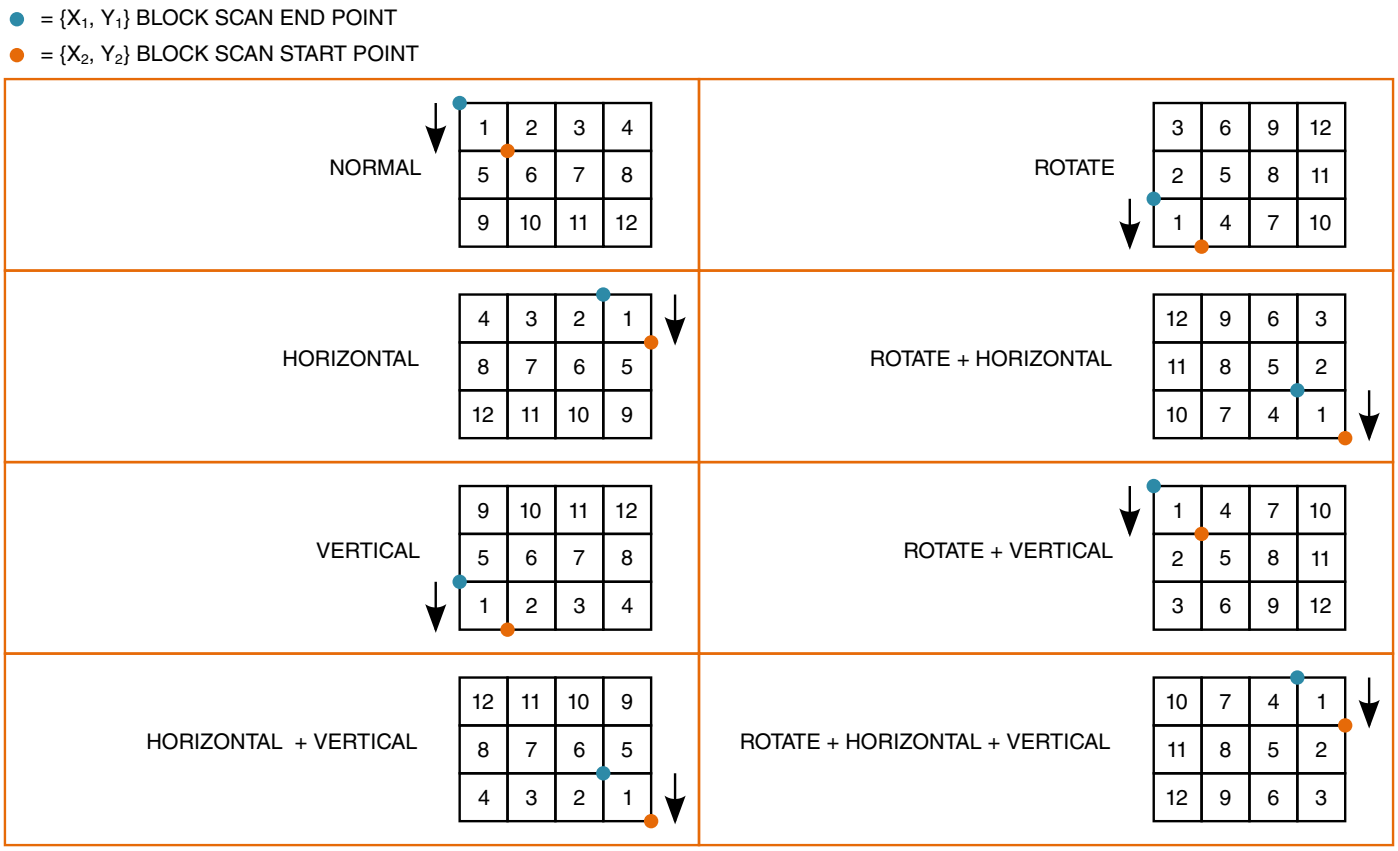


Figure 37-12. Rotation options

37.4.2.11.2 Frame size

The IPU supports various non-interleaved modes; the Frame Height (FH) and Frame Width (FW).

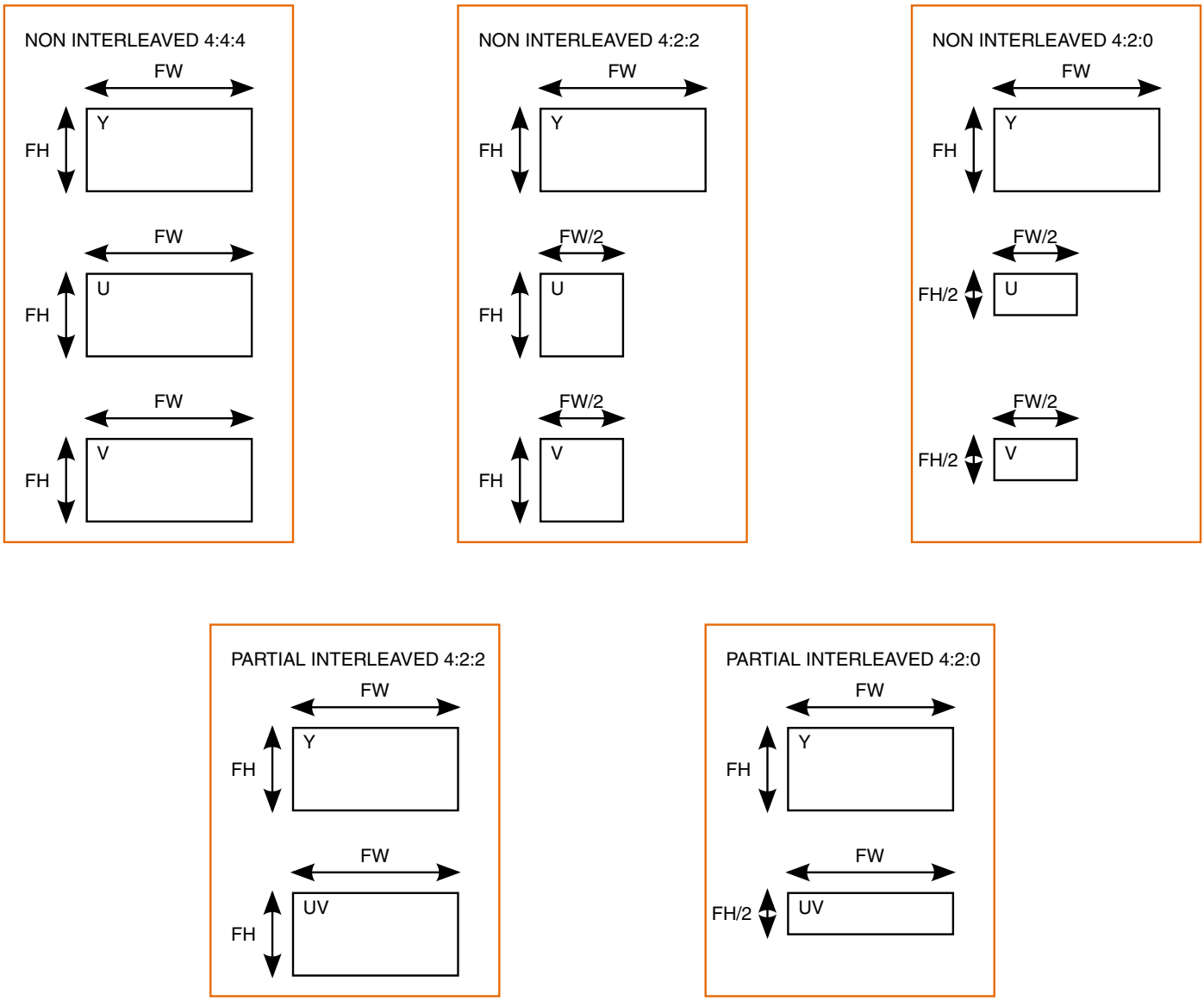


Figure 37-13. Frame size in various modes

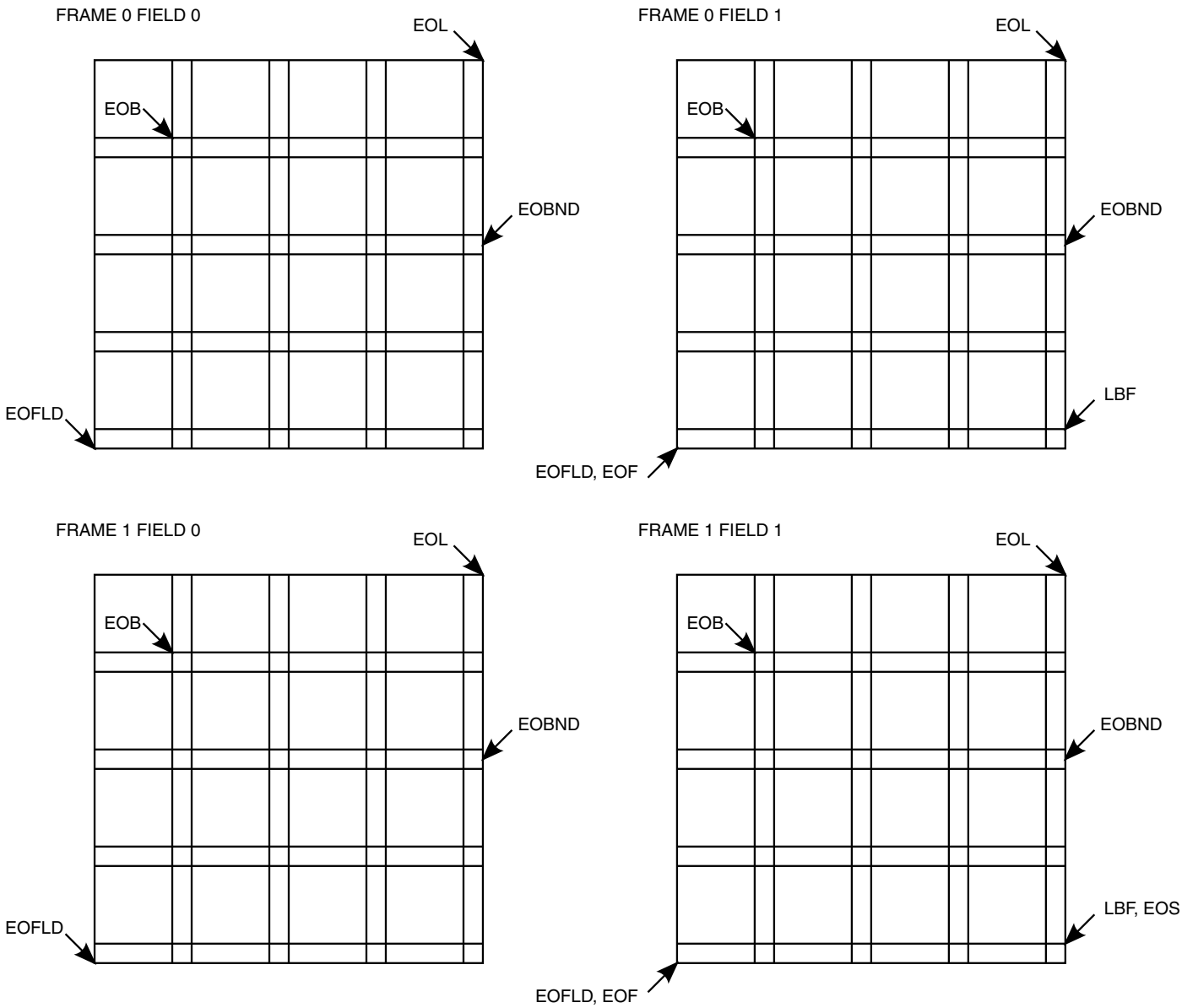


Figure 37-14. Frame's controls

37.4.2.12 IDMAC's restriction

The data must be received from the system's memory through the AXIR interface ("read" direction) "in-order" within a single burst. The entire burst can occur "out-of-order."

37.4.2.13 IDMAC's Endianness support

Byte Endianness - only LE (little endian) is supported

Pixel Endianness - both LE & BE are supported, only for read direction (only 4 BPP case is meaningful, supported only for the "read" direction)

37.4.2.14 IDMAC's internal events

Some of the IDMAC's internal signals can be used for monitor the progress of flows. These bits can be polled by software. Some of these bits can be used to trigger an interrupt or an SDMA event.

The following table describes the available events and their meaning

Table 37-16. IDMAC's internal events

IDMAC event's type	Monitored on	Event's meaning
end of frame	IDMAC_EOF	This is the channel's end of frame indication. This indication is asserted once the entire frame was read/written via the IDMAC. This indication is normally used as an interrupt or SDMA event
new frame acknowledge	IDMAC_NFACK	This indication means that the IDMAC acknowledge the new frame's request from the module. It can be used to track the starting point of a flow or a frame.
new frame before end of frame error	IDMAC_NFB4EOF_ERR	This error indication may indicate on data lost. This indication is asserted when a new frame starts before the completion of the previous frame. For example if a real time input (from camera) was not written properly to the memory due to FIFO full condition.
end of scroll	IDMAC_EOS	end of scroll; This indication is asserted when the scroll counter finished counting its pre defined value
end of band	IDMAC_EOBND	This is the end of band indication. Any time IDMAC complete reading/writing a band it will assert this indication. This is useful to manually control a flow via channels working in band mode.
treshold	IDMAC_TH	Threshold crossing indication. The treshold is defined according to the TH parameter in the IDMAC.
channel busy	IDMAC_CH_BUSY	This signal is asserted when a channel is between NFACK event to EOF event. Negation of these indications is one of the conditions for low power modes handshake.

37.4.3 Camera Sensor Interface (CSI)

The IPU has 2 identical camera sensor interfaces (CSI). The CSI description below refers to a single CSI.

37.4.3.1 CSI Block Diagram

The CSI consists of synchronizer, interface logic, Data packing unit and Sensor Interface Control.

The CSI is controlled via the peripheral bus registers. All programming parameters for the CSI are double buffered with synchronous change at the frame start.

The CSI gets data from the sensor, synchronizes the data and the control signals to the IPU clock (HSP_CLK), and transfer it according to configuration of DATA_DEST register to one or more of the following: IC, SMFC.

This figure shows the CSI Block Diagram.

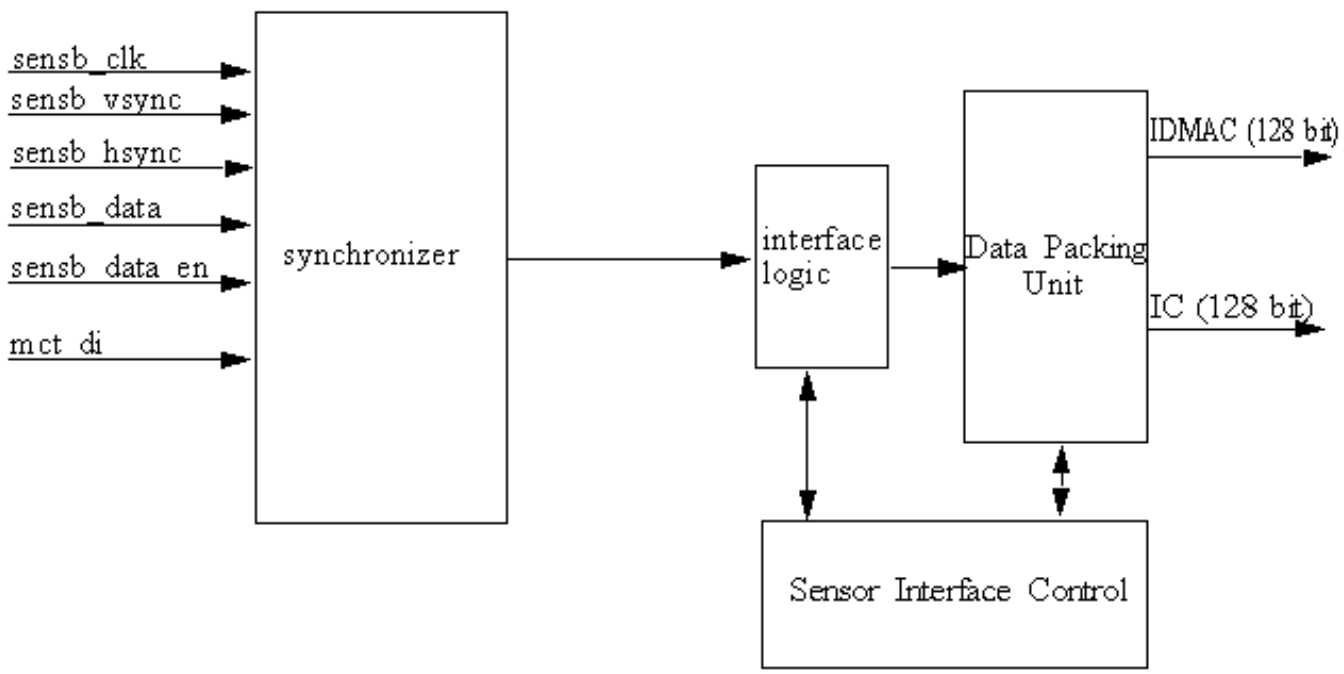


Figure 37-15. CSI Block Diagram

37.4.3.2 CSI Interface

CSI supports two types of interfaces. The interface is determined via the DATA_SOURCE register.

37.4.3.2.1 Parallel Interface

In parallel interface a single value arrives in each clock, except when working in BT.1120 mode, in which two values arrive in each cycle. Each value can be 8-16 bit wide according to configuration of DATA_WIDTH. If DATA_WIDTH is configured to N, then 20-N LSB bits are ignored.

CSI can work with several data formats according to SENS_DATA_FORMAT configuration. In case the data format is YUV, the output of the CSI is always YUV444 (even if the data arrives in YUV422 format).

The polarity of the inputs can be configured using the registers SENS_PIX_CLK_POL, DATA_POL, HSYNC_POL and VSYNC_POL.

37.4.3.2.2 High-speed serial interface - MIPI (Mobile Industry Processor Interface).

In MIPI interface two values arrive in each cycle. Each value is 8 bit wide, which means 16 MSB bits of the data bus input are treated, while 4 LSB bits are ignored.

When working in this mode, the CSI can handle up to 4 streams of data. Each stream is identified with DI (data identifier) that includes the virtual channel and the data type of this stream. Each stream that is handled is defined in registers MIPI_DI0-3. Only the main stream (MIPI_DI0) can be sent to all destination units while the other streams are sent only to the SMFC as generic data.

In this mode SENS_DATA_FORMAT and DATA_WIDTH registers are ignored, since this information is coming to the CSI via the MCT_DI bus.

37.4.3.3 Test Mode

When TEST_GEN_MODE register is configured to 1, the TEST MODE which is a debugging mode, is operated.

The CSI generates the frame by itself and sends it to one of the destination units. The sent frame is a chess board composed of black and configured color squares. The configured color is set with the registers PG_B_VALUE, PG_G_VALUE and PG_R_VALUE. The data can be sent in different frequencies according to the configuration of DIV_RATIO register.

CSI Test Mode requires the following CSIx_SENS_CONF settings:

CSIx_EXT_VSYNC = 0x1 (External VSYNC mode)

CSIx_DATA_WIDTH = 0x1 (8 bits per color)

CSIx_SENS_DATA_FORMAT = 0x0 (Full RGB or YUV444)

CSIx_PACK_TIGHT = 0x0

CSIx_SENS_PRTCL = 0x1 (Non-gated clock sensor timing/data mode)

CSIx_SENS_PIX_CLK_POL = 0x1 Pixel clock is inverted before applied to internal circuitry

CSIx_DATA_POL = 0x0 (Data lines are directly applied to internal circuitry.)

CSIx_HSYNC_POL = 0x0 (HSYNC is directly applied to internal circuitry)

CSIx_VSYNC_POL = 0x0 (VSYNC is directly applied to internal circuitry)

37.4.3.4 Sensor Image Frame Relations

The figure found here illustrates the generalized relations between image frames produced by a sensor and accepted by the CSI.

Generally, four frame definitions exist. The virtual frame A starts with the VSYNC signal. After vertical blanking starts frame B. The HSYNC signal indicates boundaries of the frame B. The frame B includes both the active sensor frame C and horizontal blanking intervals. A size of the blanking intervals depends on sensor type and programming. The size of the frame sent by the sensor (the actual pixels) has to be configured in the registers SENS_FRM_WIDTH and SENS_FRM_HEIGHT. The CSI selects a window (the frame D) inside the frame C by skipping rows and columns according to parameters defined in registers HSC, VSC, ACT_FRM_HEIGHT and ACT_FRM_WIDTH. frame D is sent to the rest of the IPU.

NOTE

The following limitation must exist:

SENS_FRM_HEIGHT >= VSC + ACT_FRM_HEIGHT

SENS_FRM_WIDTH >= HSC + ACT_FRM_WIDTH

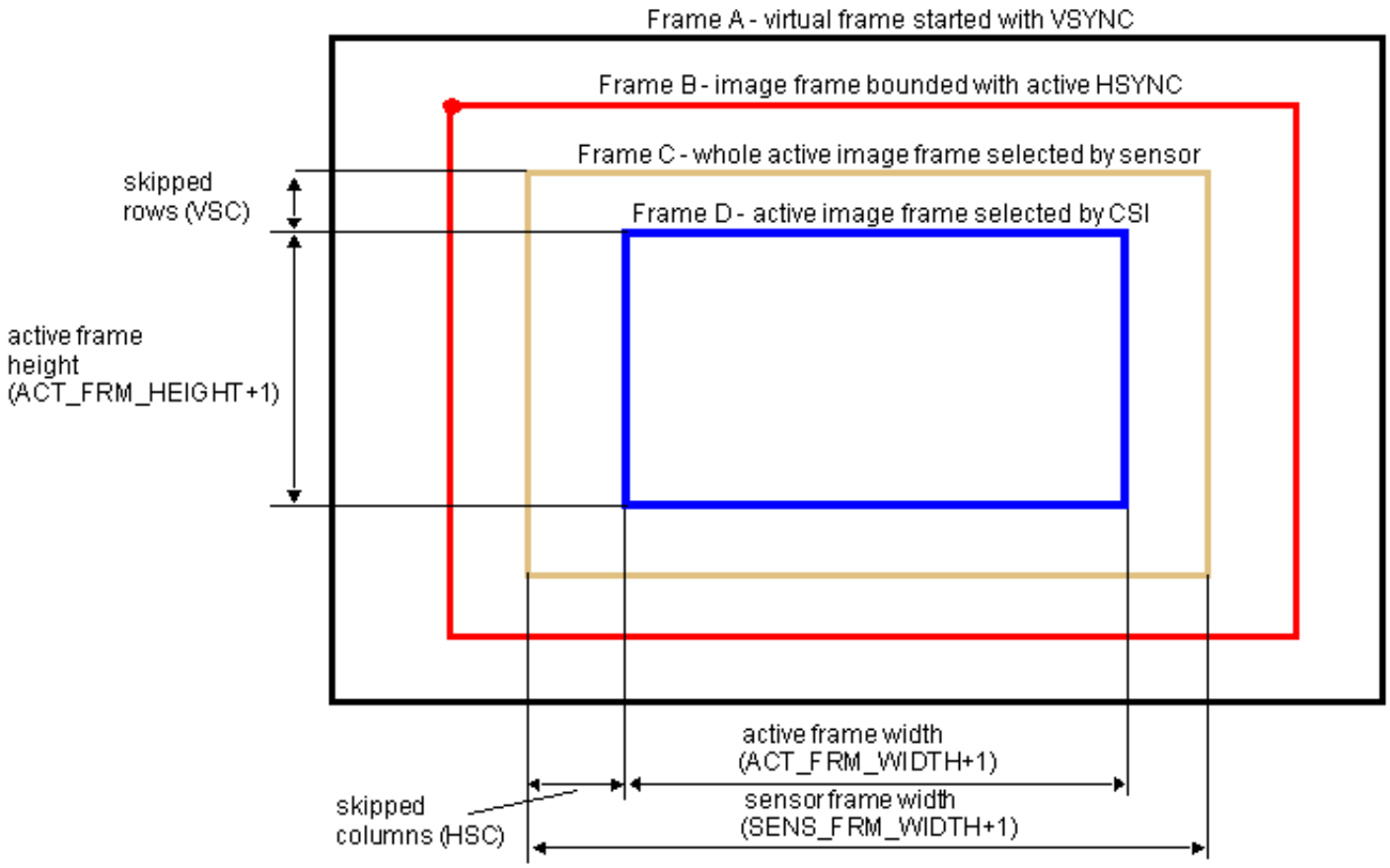


Figure 37-16. Sensor Image Frames

37.4.3.5 Companding

Companding is performed as follows:

input: 10-bit unsigned number.In case that component's size is bigger than 10 bit, 10 MSB bits are taken. In case that component's size is less than 10 bit, color extraction is performed.

First step: shift:

x-> clip(x + offset, 0, 1023)

where the offset is a 10 bit signed number that is configured in CPD_OFFSET1 and CPD_OFFSET2 registers.

second step - piecewise-linear map:

y = Min[255, (y1[k]+(((x-x1[k])*slope[k])>>6 + 1))>>1]

where:

- the input range 0..1023 is partitioned to 16 equal segments: x1[k] .. x1[k+1] - 1 where x1[k] = 0, 64, 128, 192, 256, .. . 960.

- The linear map, in each segment, is characterized by y1[k] (9-bit unsigned number) and slope[k] (8-bit unsigned number).

Each destination unit can get the data after being companded depending on the configuration of CPD register. If this register is configured to 3'h0, the compander units are disable in order to save power. Parameters of the companding are equal for all destinations and can be set in the CPD registers.

2 companding units are located in the CSI. This is because when working in MIPI orBT. 1120 modes 2 components arrive in each cycle and the companding for each 2 components has to be in parallel. When CSI works in other mode, only one compander is needed, so the other one is disabled to save power.

37.4.3.6 Timing/Data mode protocols

CSI can work in several timing/data mode protocols, according to SENS_PRTCL configuration.

37.4.3.6.1 Gated Mode

In this mode VSYNC is used to indicate beginning of a frame, HSYNC is used to indicate beginning of a row. Sensor clock is ticking all the time.

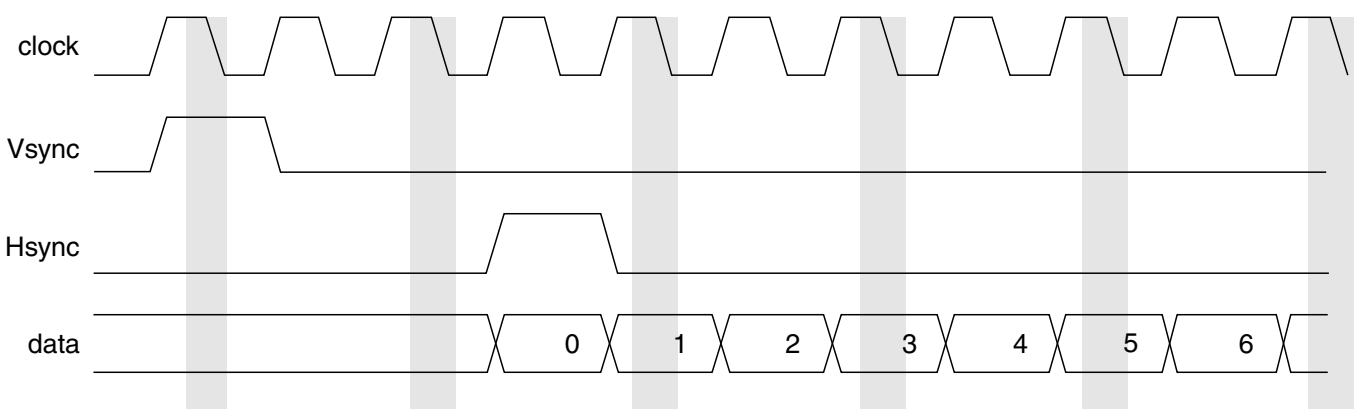


Figure 37-17. gated mode

37.4.3.6.2 Non-Gated Mode

In this mode VSYNC is used to indicate beginning of a frame. Sensor clock is ticking only when data is valid. HSYNC is not used.

When working with MIPI, the non-gated mode should be configured.

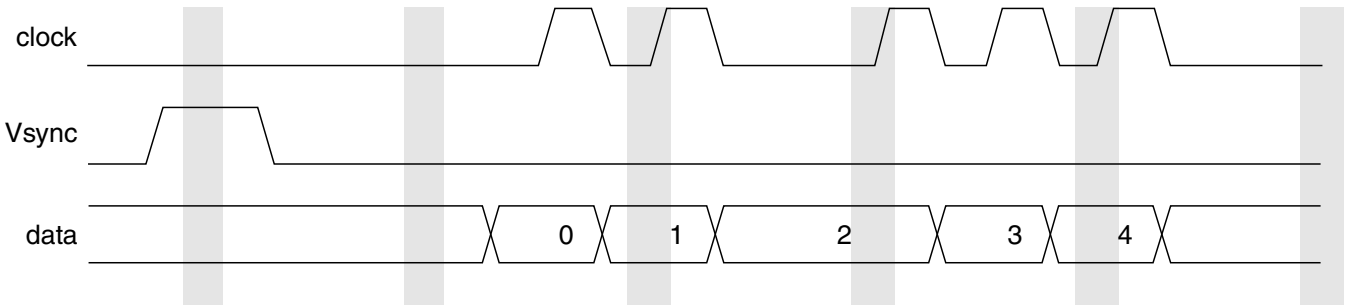


Figure 37-18. non-gated mode

37.4.3.6.3 BT.656 mode

In this mode the CSI works in compliance with recommendation ITU-R BT.656. The timing reference signals (frame start, frame end, line start, line end) are embedded in the data bus input. Each timing reference signal consists of a four word sequence. The first three words are fixed and configured in the CCIR_PRECOM register. The fourth word contains information defining field, the state of field blanking and the state of line blanking. these states are configured in registers CCIR_CODE_1 (for field 0) and CCIR_CODE_2 (for field 1).

In this mode in each cycle one value of data arrives

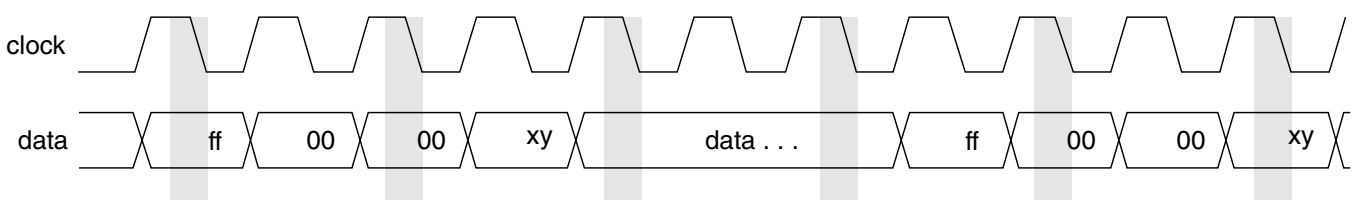


Figure 37-19. BT.656 mode

In this diagram the first three words are 0xff, 0x00, 0x00. The fourth word is XY and it includes the timing reference.

37.4.3.6.4 BT.1120 mode

In this mode the CSI works in compliance with recommendation ITU-R BT.1120. The timing reference signals (frame start, frame end, line start, line end) are embedded in the data bus input. Each timing reference signal consists of a four word sequence. The first three words are fixed and configured in the CCIR_PRECOM register. The fourth word contains information defining field, the state of field blanking and the state of line blanking. these states are configured in registers CCIR_CODE_1 (for field 0) and CCIR_CODE_2 (for field 1).

In this mode, the CSI can also work in DDR mode - data arrives on every edge of the clock. In addition, in each cycle two value of data arrive.

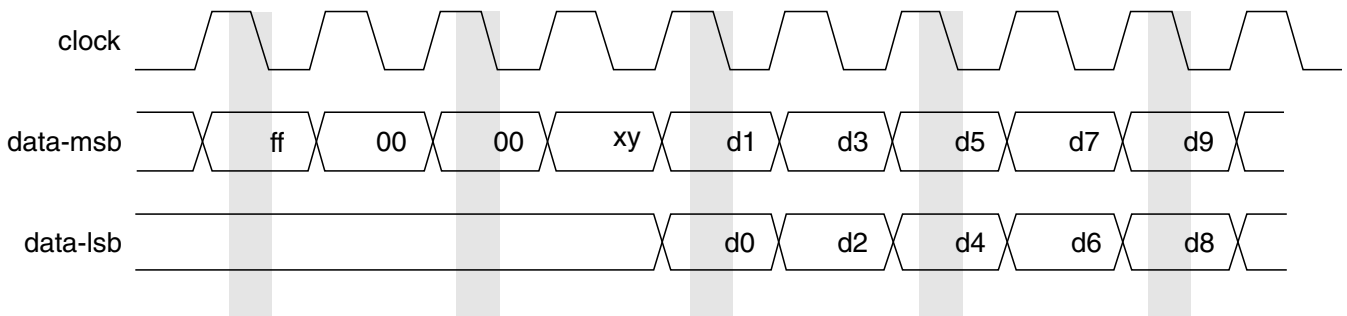


Figure 37-20. BT.1120 mode - SDR mode

In the above diagram each data arrives with the positive edge of the clock.

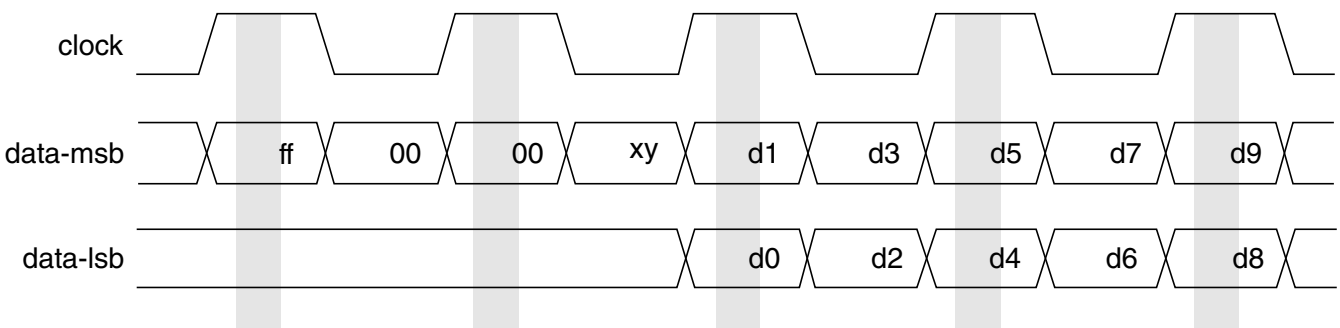


Figure 37-21. BT.1120 mode - DDR mode

In the above diagram, data arrives in the positive edge and the negative edge of the clock.

37.4.3.7 Packing to memory

The data bus output to the SMFC is 128-bit.

The following table shows the how the CSI performs the packing before sending the data to the SMFC.

The data bus output to the SMFC is 128-bit. The following table shows the how the CSI performs the packing before sending the data to the SMFC.

Table 37-17. Packing Unit

data format	component size	companded	regular packing	tight packing
Bayer, Generic data, JPEG	8	{8DC[1] ,8DC,8DC,...,8DC}	{8D,8D,8D,...,8D[2] }	NA
	9-16	{8DC,8DC,8DC,...,8DC}	{16DE[3] ,16DE,...,16DE}	NA
RGB, YUV	8	{8DC,8DC,8DC,8R}	{8D,8D,8D,8R}	NA
	9-10	{8DC,8DC,8DC,8R}	{16DE,16DE,16DE,16R[4] }	{10DE,10DE,10DE,2R}
	11-16	{8DC,8DC,8DC,8R}	{16DE,16DE,16DE,16R}	{10DT[5] ,10DT,10DT,2R}

1. DC - data after being companded
2. D - data arrived from sensor.
3. DE - data after being extended.
4. R- reserved bits
5. DT - data after being truncated.

The tight packing functionality is enabled when PACK_TIGHT register is set. It is only used when data format is RGB or YUV and the data width is bigger than 8.

37.4.3.8 Skipping frames

Some of the frames that are sent to the SMFC can be skipped. Skipped frames are ignore by the CSI and are not sent to the corresponding unit.

Using SKIP_SMFC and MAX_RATIO_SKIP_SMFC registers the user can define the frames for the SMFC that will be skipped.

37.4.3.9 16 bit camera support

Devices that support 16 bit data bus can be connected to the CSI. This can be done in one of the following ways.

16 bit YUV422

In this mode the CSI receives 2 components per cycle. The CSI is programmed to accept the data as 16 bit generic data. The captured data will be stored in the memory through the SMFC. The IDMAC needs to be programmed to store 16bit generic data. When the data is read back from the memory for further processing in the IPU it will be read as YUV422 data.

16 bit RGB as generic data

In this mode the CSI receives 3 components per cycle. If the external device is 24bit - the user can get connect a 16 bit sample of it (such as RGB565) The CSI is programmed to accept the data as 16 bit generic data. The captured data will be stored in the memory through the SMFC. The IDMAC needs to be programmed to store 16bit generic data. When the data is read back from the memory for further processing in the IPU it will be read as 16 bit RGB data. The IDMAC's mapping unit will be used to remap the 16 bit data to the internal 24bpp RGB format In this mode on the fly processing is can't be performed. The data has to be sent to the memory first and then further processed by the IPU.

16 bit RGB565

This is the only mode that allows on the fly processing of 16 bit data. In this mode the CSI is programmed to receive 16 bit generic data. In this mode the interface is restricted to be in "non-gated mode" and the CSI#_DATA_SOURCE bit has to be set If the external device is 24bit - the user can connect a 16 bit sample of it (RGB565 format). The IPU has to be configured in the same way as the case of CSI#_SENS_DATA_FORMAT=RGB565

37.4.3.10 CSI Restrictions

The frequency of the sensor clock must not be greater than the IPU clock (HSP_CLK)

SENS_FRM_HEIGHT >= VSC + ACT_FRM_HEIGHT

SENS_FRM_WIDTH >= HSC + ACT_FRM_WIDTH

37.4.4 Sensor Multi FIFO Controller (SMFC)

The Sensor Multifile Controller used as buffer between CSI and IDMAC. Two masters (CSIs) can be connected to SMFC. Both masters can be active simultaneously.

Each master can send up to 4 frames, distinguished by csi_id bus. The frame can be mapped to one of four IDMAC channels via SMFC mapping registers. Each DMA channel have dedicated FIFO.

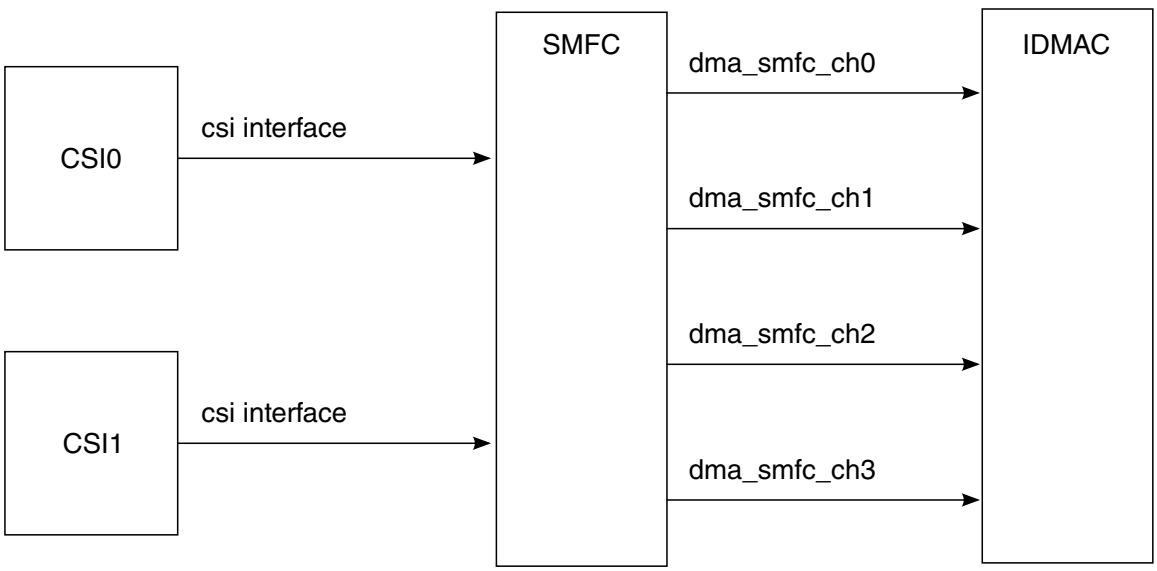


Figure 37-22. SMFC data flow

37.4.4.1 SMFC's Features

- Support two CSI masters and four DMA channels
- Automatic FIFO size calculation

37.4.4.2 SMFC's Functional description

SMFC supports up to four DMA channels. Each channel has a dedicated FIFO controller, as shown in the following figure. Sampled data and frame ID are kept in the buffers until the buffer is selected by Round Robin Priority Mechanism. Then, the content of ID buffer is compared to CH#_MAP bits and the corresponding FIFO controller is activated. As a result, the content of the buffer is copied to the RAM. The wptr and the "base" are used to calculate the location in the RAM. The rptr are following after wptr during dma_active signal, initiated by DMA.

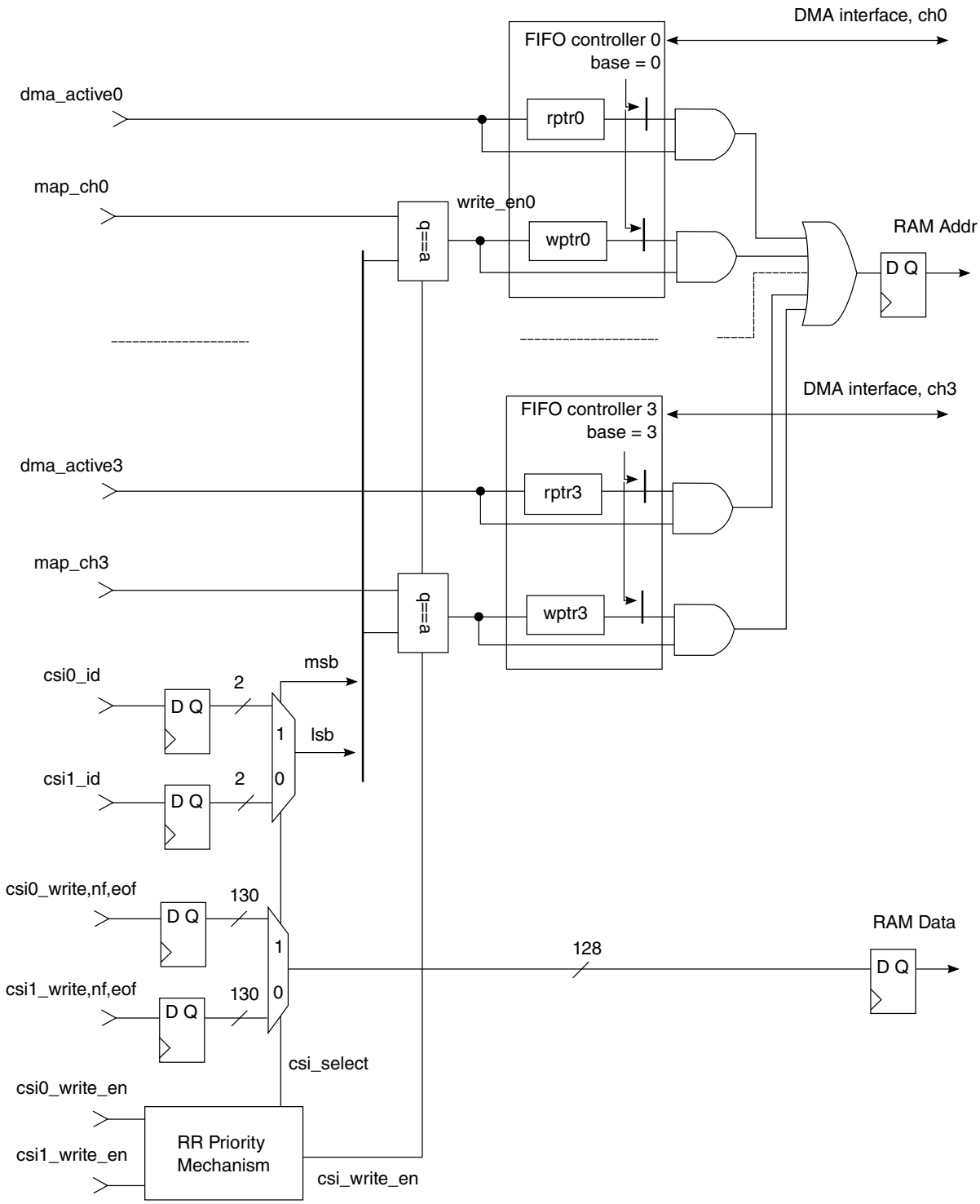


Figure 37-23. Sensor Multi FIFO Controller

Four FIFOs are implemented with single RAM due to the fact that the channels can't be active simultaneously. The memory space of SMFC is divided into four equal sectors. Each FIFO has a fixed base address - "base". The "base" used as MSB for corresponding pointer in order to calculate absolute address of the RAM. FIFO size of channels 1 and 3 are fixed and is equal to size of one sector. FIFO size of channels 0 and 2 depend from other channels as shown in the table below. Other configurations are not allowed.

Table 37-18. FIFO Channels

Number of DMA channels required	enable/disable of channels 3,2,1,0	FIFO size (sectors) per channels 3,2,1,0
1	0 0 0 1	0 0 0 4
2	0 1 0 1	0 2 0 2
3	1 1 0 1	1 1 0 2
4	1 1 1 1	1 1 1 1

NOTE

Channels should not be enabled after activation of channel 0 or/and channel This can cause to overlapping of FIFO areas and other malfunctions.

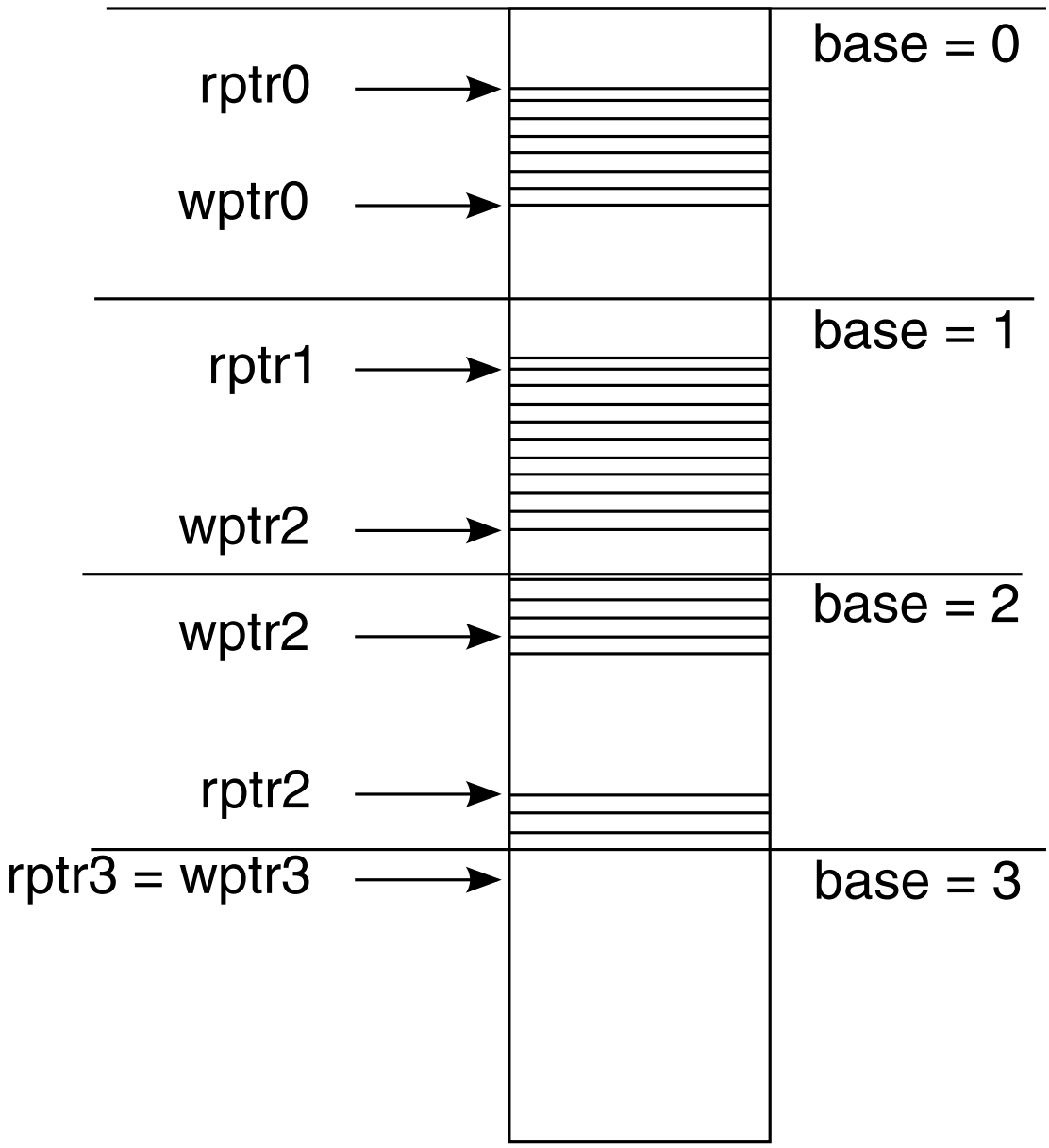


Figure 37-24. SMFC memory map when DMA channels 3,2,1,0 are enabled

37.4.4.2.1 SMFC Master interface.

SMFC Master interface is shown in the following figure.

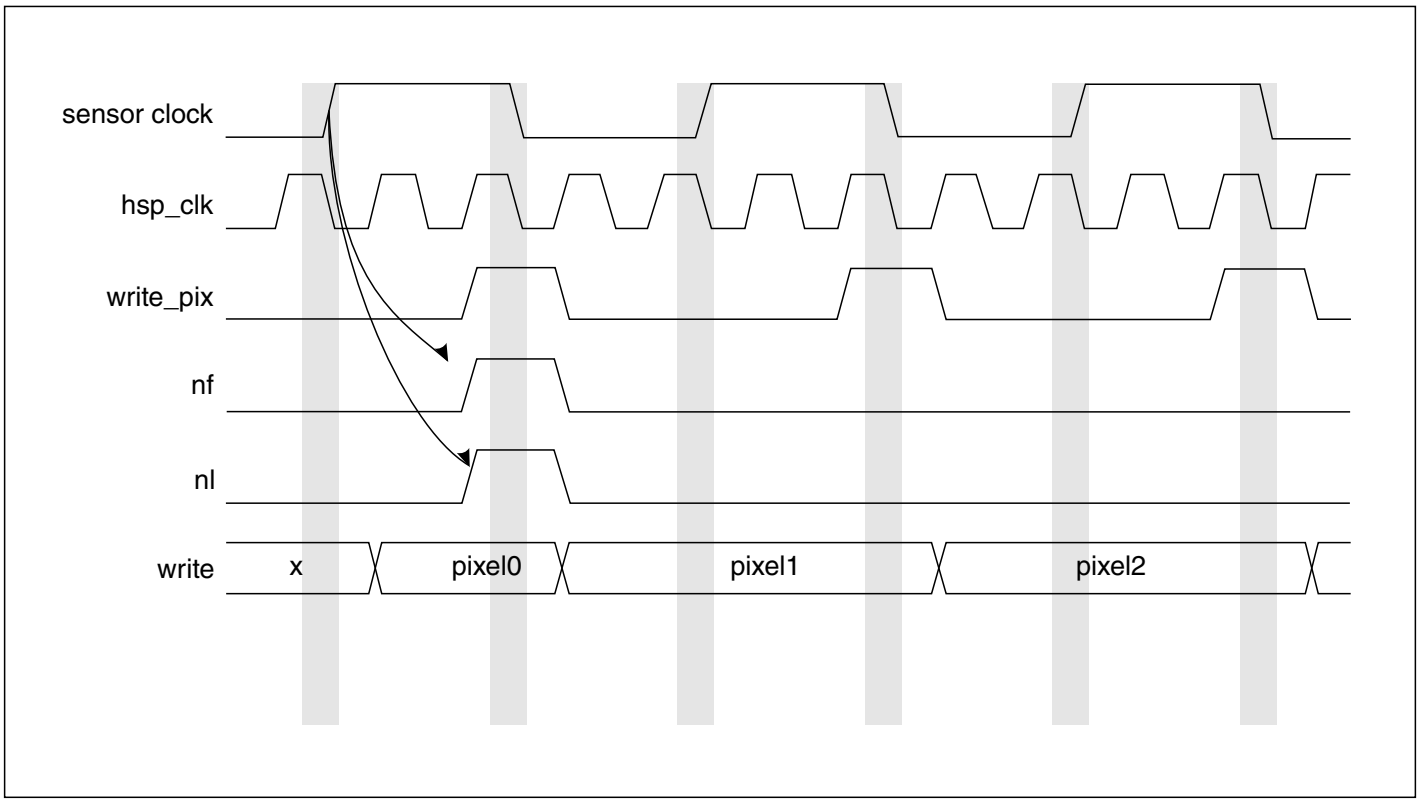


Figure 37-25. Timing diagram of SMFC slave interface

Sensor clock can be asynchronous to IPU clock (hsp_clk). The CSI synchronize data arrived from the sensor to hsp_clk. The csi_write signal indicates when data on csi_pix bus can be sampled by hsp_clk clock.

37.4.4.2.2 Restrictions

1. DMA channels should not be enabled after activation of channel 0 or/and channel 2. This can cause to overlapping of FIFO area and other malfunctions.
2. Watermark set value should be greater than watermark clear value.
3. One frame should not be mapped to few DMA channel.

37.4.5 Image Converter

37.4.5.1 IC Block Diagram

The IC contains three processing sections: downsizing, main processing and rotation.

The block is controlled via the peripheral bus registers. Some processing parameters should be written by the ARM core to the Task Parameter Memory. Writing to the memory is performed via the AHB bus.

The IC Block Diagram is shown in the following figure.

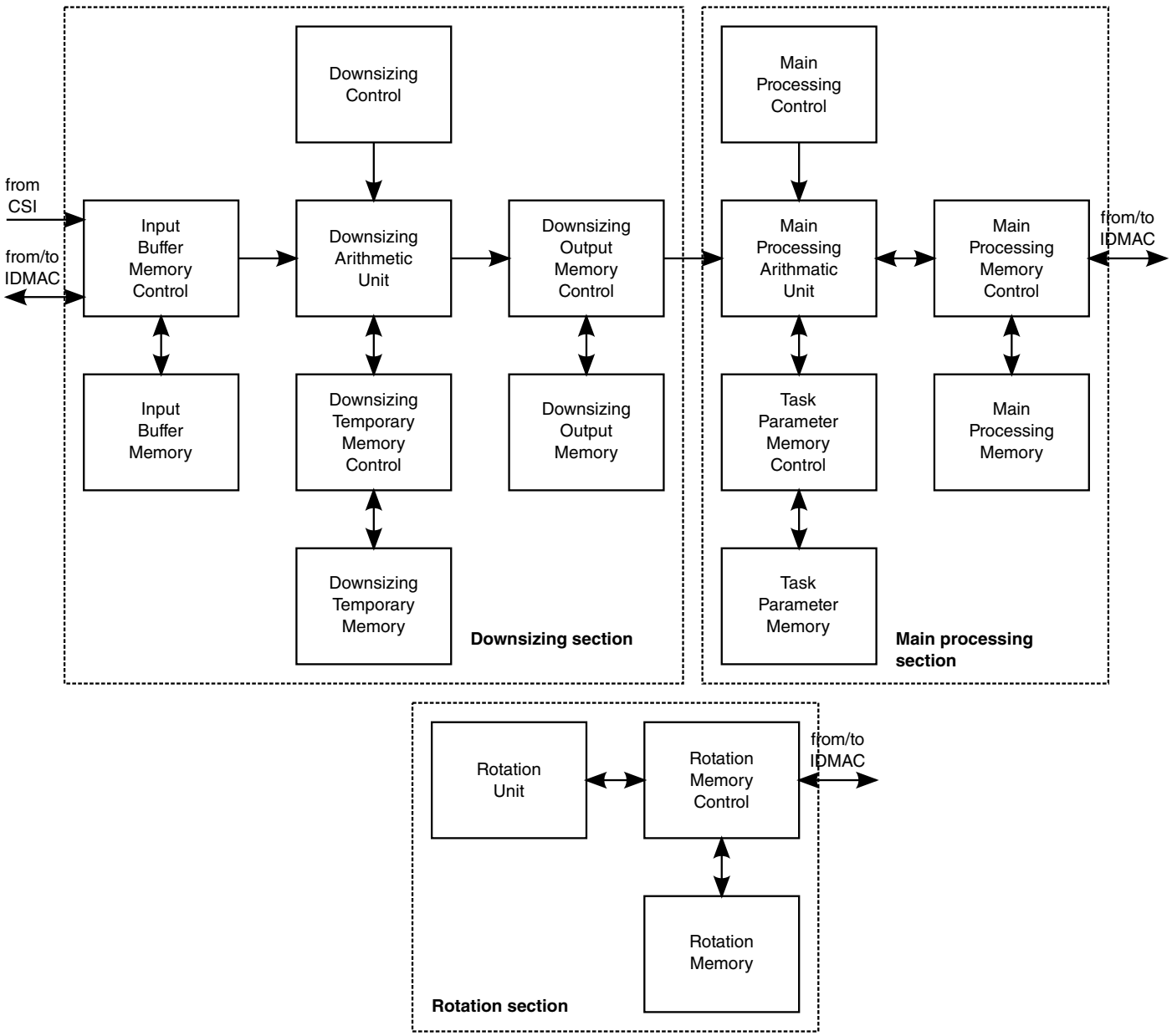


Figure 37-26. IC Block Diagram

37.4.5.2 Processing tasks

Each of the three processing sections performs up to three processing tasks with time sharing:

1. Preprocessing task for encoding.
2. Preprocessing task for displaying image from sensor (viewfinder).

3. Postprocessing task.

The tasks are performed by single hardware. The ARM platform configures each task before enabling it.

Tasks switching is transparent for the ARM platform. The time unit for task switching in the downsizing section corresponds to a processing time of one burst of eight pixels, in the main processing section - to a processing time of one image line, in the rotation section - to a processing time of one image frame.

All three tasks include similar operations controlled by commands. Task configuring consists in definition of commands for each task, as described in the following table.

Table 37-19. Task Commands

Command code	Command	Processing unit	Command parameters	Description
EN	Task Enable	DSU, MPU		Task will enabled from next frame. Task 1 is preprocessing for encoding. Task 2 is preprocessing for viewfinder. Task 3 is postprocessing.
	Downsizing	DSU	Downsizing ratio (GCR)	Downsizing ratio 1:1, 2:1, 4:1
	Resizing	MPU	Resizing ratio (GCR)	Resizing ratio from 2:1 to 1:M Resizing ratio = N:M; M =2^13; N = floor(M*(SI-1)/(SO-1)); SI - input size; SO - output size
CSC1	Color space conversion 1	MPU	Color conversion coefficients and offsets (TPM)	Color conversion matrix 1
GLOB_A	Global alpha	MPU		Used only for Task 2 and Task 3
CMB	Combining	MPU		Combining video with graphics. Used only for Task2 and Task3.

The ARM platform writes the commands to the IC_CONF Register. Because there is no double buffering for all the IC parameters, the ARM platform must disable a task before changing its parameters. After being disabled, the task is still allowed to complete its current frame execution. At frame finish, the IPU sends an interrupt to the ARM platform indicating that the ARM platform can change task parameters. The ARM platform enables the task again and task execution is resumed from start of the next frame.

37.4.5.3 Downsizing Section

The sensor data from the CSI is written into a FIFO located in the Input Buffer Memory. Depending on programmed processing flow, the FIFO data can be sent to the system memory via the IDMAC or straight forward to the Downsizing Unit.

In the first case, the data is processed by the ARM platform and returned by the IDMAC to another FIFO located in the Input Buffer Memory.

For postprocessing, the IDMAC transfers a data from the system memory to the third FIFO. The data is read by the Downsizing Unit when the postprocessing is performed.

Each or three FIFOs has 128 pages. Each page can store one burst of 2 or 4 words which corresponds to 8 or 16 pixels. The size of the burst is defined according to the channel's corresponding CB#_BURST_SIZE bit. The memory word width is 128 bits. Each memory word contains color components of four adjacent pixels or 16 bytes of generic data (e.g. Bayer). Access to the FIFOs is controlled by the Input Buffer Memory Control.

The Downsizing Unit performs averaging and decimation of image pixels both in horizontal and vertical directions according to the following equations:

$$HP_{R,c} = \frac{1}{DS\_R\_H} \sum_{k=0}^{DS\_R\_H-1} IP_{r+k,c}$$

$$VP_{R,C} = \frac{1}{DS\_R\_V} \sum_{l=0}^{DS\_R\_V-1} HP_{R,c+l}$$

where IPr,c - the input pixel, HPR,c - the pixel after horizontal downsizing, VPR,C - the pixel after vertical downsizing, DS_R_H and DS_R_V - the horizontal and vertical downsizing ratios according to the IC_PRP_ENC_RSC, IC_PRP_VF_RSC and IC_PP_RSC Registers. The final calculation result is rounded to 8 bits.

Each of three downsizing tasks processes the data by bursts of 8 pixels. Normally, the current task runs until emptying the corresponding input FIFO. After finishing burst processing, the Downsizing Unit may switch between the current task and another task with higher priority, if the Input Buffer Memory has received a burst for this new task.

Averaging is performed firstly in the horizontal direction. All color components of a pixel are processed in parallel. After horizontal averaging has finished for a single output pixel, the new pixel value is added to the corresponding pixel value of a temporary row derived from previous averaging steps. This provides vertical averaging of the pixels. The temporary row is stored in the Downsizing Temporary Memory. The memory word width matches one accumulated pixel width (36 bits). There are three temporary rows stored in this memory - one per downsizing task.

After vertical averaging has been finished, the output row is written to the Downsizing Output Memory. The memory word of 48 bits includes two output pixels. For each task, the memory has a double buffer of one row. When the Downsizing Unit fills the foreground part of the double buffer, the Main Processing Unit takes pair or pixels from the background part. After the new downsized row has been ready, the foreground and background memory pointers are swapped.

37.4.5.4 Main Processing Section

The Main Processing Unit reads pairs of pixels from the Downsizing Output Memory background part. It processes the complete pixel row for the current task and after that switches to another task if the input data for this new task is ready.

For each task, the Main Processing Unit is able to perform the following sequence of operations:

1. Horizontal flipping the image (optional) performed with reading from the Downsizing Output Memory. Flipping is enabled via the VF, HF & ROT parameters of the corresponding DMA channels (Table 37-14 and Table 37-15) responsible for output of the task results. The preprocessing task for encoding uses the VF, HF & ROT parameters from IDMAC channel #20, the preprocessing task for viewfinder - from the IDMAC channel #21, the postprocessing task - from the IDMAC channel #22.
2. Horizontal resizing by bilinear interpolation between two adjacent pixels received from the Downsizing Output Memory according to the equation:

$$HP_{R,c} = IP_{r,c} + RS\_C\_H \cdot (IP_{r+1,c} - IP_{r,c})$$

where RS_C_H - the current horizontal resizing coefficient. The calculation result is rounded to 8 bits. The resizing coefficient is calculated as

$$RS\_C\_H = \left(\sum_{k=0}^{R-1} RS\_R\_H \right) \bmod(8196)$$

where RS_R_H - the horizontal resizing ratio from the IC_PRP_ENC_RSC, IC_PRP_VF_RSC and IC_PP_RSC Registers. The RS_R_H parameter is equal to a numerator N of the resizing ratio N:M with M =2^13.

The resulting row of the horizontal resizing is stored in the Task Parameter Memory.

3. Vertical resizing by bilinear interpolation between the current and previous results of horizontal resizing. Both current and previous results of horizontal resizing is stored in the Task Parameter Memory. Resizing is accomplished according to the equation:

$$VP_{R,C} = HP_{R,c} + RS\_C\_V \cdot (HP_{R,c+1} - HP_{R,c})$$

where RS_C_V - the current vertical resizing coefficient. The calculation result is rounded to 8 bits. The resizing coefficient is calculated as

$$RS\_C\_V = \left(\sum_{k=0}^{C-1} RS\_R\_V \right) \bmod(8196)$$

where RS_R_V - the horizontal resizing ratio from the IC_PRP_ENC_RSC, IC_PRP_VF_RSC and IC_PP_RSC Registers. The RS_R_V parameter is equal to a numerator N of the resizing ratio N:M with M =2^13.

At completion of vertical resizing, this row is updated - the current result of horizontal resizing replaces the previous one.

4. First color space conversion YUV to RGB or RGB to YUV with the conversion matrix CSC1. The conversion matrix coefficients are programmable. They are stored in the Task Parameter Memory. The conversion equations are:

$$Z_0 = 2^{SCALE-1} \cdot (X_0 \cdot C_{00} + X_1 \cdot C_{01} + X_2 \cdot C_{02} + A_0)$$
$$Z_1 = 2^{SCALE-1} \cdot (X_0 \cdot C_{10} + X_1 \cdot C_{11} + X_2 \cdot C_{12} + A_1)$$
$$Z_2 = 2^{SCALE-1} \cdot (X_0 \cdot C_{20} + X_1 \cdot C_{21} + X_2 \cdot C_{22} + A_2)$$

where for YUV to RGB: X_0=Y, X_1=U, X_2=V, Z_0=R, Z_1=G, Z_2=B, for RGB to YUV: X_0=R, X_1=G, X_2=B, Z_0=Y, Z_1=U, Z_2=V.

All the parameters of the conversion matrix are written by the ARM platform to the Task Parameter Memory (IC Task Parameter Memory). The final calculation result is limited according to the SAT_MODE parameter and rounded to 8 bits.

5. Combining video with graphics. There are the following combining options:
   - local alpha blending,
   - global alpha blending,
   - use of key color.

   If both alpha blending and color keying are enabled, color keying has higher priority (graphic pixels of the key color are fully transparent independently on the alpha value).

   Combining mode is selected via the IC_CONF Register. The combining equation is:

$$OP = IGP \cdot \alpha + IVP \cdot (1 - \alpha)$$

   where IGP - an input graphics pixel, IVP - an input video pixel, α=(A+floor(A/128))/256 - an alpha value, A - a global or local transparency parameter. The global A is written in the IC_CMBP_1 Register, the local A arrives together with the graphics pixel.

   A graphics pixel becomes transparent when color keying is enabled and a pixel color matches a key color (independently on an alpha parameter).

   The graphics data is read from a FIFO located in the Main Processing Memory. The FIFO contains 32 pages of size of 8 pixels. The size of the burst is defined according to the channel's corresponding CB#_BURST_SIZE bit. The graphics pixel format in the FIFO is RGB or RGBA or YUV 4:4:4 or YUVA. The graphics data is loaded by the IDMAC to the FIFO from the system memory.

All the operation are executed by an unified processing unit sequentially. Steps 1 and 2 cannot be interrupted by another task. All other steps can be interrupted by a task with higher priority if an input row is ready for this task. Preprocessing tasks priority is higher than postprocessing task priority.

The processing unit consists of three identical parts for each color component. All three color components are processed in parallel. Each of the processing operations can be enabled or disabled by an appropriate command according to.

The processing results are written to an output FIFO located in the Main Processing Output Memory row-by-row. The FIFO contains 32 pages, each page can include one burst of 8 or 16 pixels. The size of the burst is defined according to the channel's corresponding CB#_BURST_SIZE bit. The IDMAC transfers the output bursts to the system memory or to the display via DMFC (Channel 21only). The Main Processing Memory contains three buffers for each tasks: the temporary row buffer, the graphics FIFO and the output FIFO. Each memory word (128 bits) stores 4 adjacent pixels in formats RGB or RGBA or YUV 4:4:4.

37.4.5.5 Rotation Section

The rotation section includes the Rotation Memory, which stores an input rectangular block of 8x8 pixels and an output FIFO containing four pages of 8 pixels each one. The Rotation Memory word width corresponds to four adjacent pixels - 96 bits. The input block is loaded to the memory by the IDMAC like to a FIFO.

The Rotation Unit rewrites pixels from the input block to the output FIFO with corresponding relocation of a pixel inside the block. Rotation and/or left/right flipping and/or up/down flipping are enabled separately for each of three tasks. Configuring the rotation and flipping options is performed via the VF, HF & ROT parameters of the corresponding DMA channels (Table 37-14 and Table 37-15) responsible for task data input. The preprocessing task for encoding uses the VF, HF & ROT parameters from the IDMAC channel #45, the preprocessing task for viewfinder - from the IDMAC channel #46, the postprocessing task - from the IDMAC channel #47.

Rotation and flip options are shown in the following table.

Table 37-20. Rotation and Flip Options

ROT	FLR	FUD	Image
0	0	0	

Table continues on the next page...

Table 37-20. Rotation and Flip Options (continued)

ROT	FLR	FUD	Image
			
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	

Table continues on the next page...

Table 37-20. Rotation and Flip Options (continued)

ROT	FLR	FUD	Image
			
1	1	0	
1	1	1	

After finishing the rotation task, the IDMAC returns the output FIFO content to the system memory. When writing to the system memory, the IDMAC changes a location of the block relative to an input block location in order to provide proper rotation of the whole frame. Rotation tasks switching is performed after completion of rotation of the whole frame.

37.4.5.6 IC Task Parameter Memory

The following table presents IC task parameter memory details.

Table 37-21. IC Parameters

Address[1]	Word[2]	Parameter	Field	Description
x2008	Encoding CSC1 word0 and word1	C22	8:0	Coefficients of color conversion matrix1 for encoding task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2;
		C11	17:9	
		C00	26:18	

Table continues on the next page...

Table 37-21. IC Parameters (continued)

Address[1]	Word[2]	Parameter	Field	Description
				Coefficients format is s.xxxxxxxx[3];
		A0	39:27	Offset of color conversion matrix1 for encoding task: Offset format is sxxxxxxxxxx.xx
		SCALE	41:40	Scale of coefficients for color conversion matrix1 for encoding task: 0 --> coefficients *0.5 1 --> coefficients*1 2 --> coefficients*2 3 --> coefficients*4
		SAT_MODE	42:42	Saturation mode for color conversion matrix1 for encoding task: 0 --> (min, max) = (0, 255) 1 --> (min, max) = (16, 240) for Z1,Z2 1 --> (min, max) = (16, 235) for Z0
x2010	Encoding CSC1 word2 and word3	C20	8:0	Coefficients of color conversion matrix1 for encoding task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C10	17:9	
		C01	26:18	
		A1	39:27	Offset of color conversion matrix1 for encoding task: Offset format is sxxxxxxxxxx.xx
x2018	Encoding CSC1 word4 and word5	C21	8:0	Coefficients of color conversion matrix1 for encoding task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C12	17:9	
		C02	26:18	
		A2	39:27	Offset of color conversion matrix1 for encoding task: Offset format is sxxxxxxxxxx.xx
x4028	Viewfinder CSC1 word0 and word1	C22	8:0	Coefficients of color conversion matrix1 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C11	17:9	
		C00	26:18	
		A0	39:27	Offset of color conversion matrix1 for viewfinder task: Offset format is sxxxxxxxxxx.xx
		SCALE	41:40	Scale of coefficients for color conversion matrix1 for viewfinder task: 0 -->coefficients *0.5

Table continues on the next page...

Table 37-21. IC Parameters (continued)

Address[1]	Word[2]	Parameter	Field	Description
				1--> coefficients*1 2--> coefficients*2 3-->coefficients*4
		SAT_MODE	42:42	Saturation mode for color conversion matrix1 for viewfinder task: 0 --> (min, max) = (0, 255) 1 --> (min, max) = (16, 240) for Z1,Z2 1 --> (min, max) = (16, 235) for Z0
x4030	Viewfinder CSC1 word2 and word3	C20	8:0	Coefficients of color conversion matrix1 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C10	17:9	
		C01	26:18	
		A1	39:27	Offset of color conversion matrix1 for viewfinder task: Offset format is sxxxxxxxxxx.xx
x4028	Viewfinder CSC1 word4 and word5	C21	8:0	Coefficients of color conversion matrix1 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C12	17:9	
		C02	26:18	
		A2	39:27	Offset of color conversion matrix1 for viewfinder task: Offset format is sxxxxxxxxxx.xx
x6060	Postprocessing CSC1 word0 and word1	C22	8:0	Coefficients of color conversion matrix1 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C11	17:9	
		C00	26:18	
		A0	39:27	Offset of color conversion matrix1 for viewfinder task: Offset format is sxxxxxxxxxx.xx
		SCALE	41:40	Scale of coefficients for color conversion matrix1 for postprocessing task: 0 -->coefficients *0.5 1--> coefficients*1 2--> coefficients*2 3-->coefficients*4
		SAT_MODE	42:42	Saturation mode for color conversion matrix1 for postprocessing task:

Table continues on the next page...

Table 37-21. IC Parameters (continued)

Address[1]	Word[2]	Parameter	Field	Description
				0 --> (min, max) = (0, 255) 1 --> (min, max) = (16, 240) for Z1,Z2 1 --> (min, max) = (16, 235) for Z0
x6068	Postprocessing CSC1 word2 and word3	C20	8:0	Coefficients of color conversion matrix1 for postprocessing task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C10	17:9	
		C01	26:18	
		A1	39:27	Offset of color conversion matrix1 for post-processing task: Offset format is sxxxxxxxxxx.xx
x6070	Postprocessing CSC1 word4 and word5	C21	8:0	Coefficients of color conversion matrix1 for post-processing task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C12	17:9	
		C02	26:18	
		A2	39:27	Offset of color conversion matrix1 for postprocessing task: Offset format is sxxxxxxxxxx.xx
x6078	Postprocessing CSC1 word0 and word1	C22	8:0	Coefficients of color conversion matrix2 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C11	17:9	
		C00	26:18	
		A0	39:27	Offset of color conversion matrix2 for viewfinder task: Offset format is sxx.xxxxxxxxx
		SCALE	41:40	Scale of coefficients for color conversion matrix1 for viewfinder task: 0 -->coefficients *2 1--> coefficients*1 2--> coefficients*0.5 3-->coefficients*0.25
		SAT_MODE	42:42	Saturation mode for color conversion matrix2 for viewfinder task: 0 --> (min, max) = (0, 255) 1 --> (min, max) = (16, 240) for Z1,Z2 1 --> (min, max) = (16, 235) for Z0

Table continues on the next page...

Table 37-21. IC Parameters (continued)

Address[1]	Word[2]	Parameter	Field	Description
x6080	Postprocessing CSC1 word2 and word3	C20	8:0	Coefficients of color conversion matrix2 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C10	17:9	
		C01	26:18	
		A1	39:27	Offset of color conversion matrix2 for viewfinder task: Offset format is sxx.xxxxxxxxx
x6088	Postprocessing CSC1 word4 and word5	C21	8:0	Coefficients of color conversion matrix2 for viewfinder task: Z0 = X0*C00 + X1*C01 +X2*C02+A0; Z1 = X0*C10 + X1*C11 +X2*C12+A1; Z2 = X0*C20 + X1*C21 +X2*C22+A2; Coefficients format is s.xxxxxxxx;
		C12	17:9	
		C02	26:18	
		A2	39:27	Offset of color conversion matrix2 for viewfinder task: Offset format is sxx.xxxxxxxxx

1. The address documented in this table is the relative address within the TPM. This is the address that should be accessed when writing or reading from this memory via the AHB bus.
2. Each word is aligned to 64 bit accessible via the AHB bus in 2 separate 32bit accesses.
3. s - sign position, x - binary digit position

37.4.5.7 IC's DMA channels

The table below mas the IDMAC channels of the IC and the IRT to the corresponding tasks.

The IC's channel name is the name of the channel at IC level. The IC channel name is referred on the IC's programming model.

Table 37-22. IC's DMA Channels

IDMAC's channel number	IC's channel name	Read/ Write	Source	Destination	Processing Flow Purpose
20	CB0	Write	IC ENC	Memory	Preprocessing data from IC (encoding task) to memory
21	CB1	Write	IC VF	Memory/DMFC	Preprocessing data from IC (viewfinder task) to memory; This channel can be configured to send the data directly to the DMFC. This is done by programming the ic_dmfc_sel bit.
22	CB2	Write	IC PP	Memory	Postprocessing data from IC to memory

Table continues on the next page...

Table 37-22. IC's DMA Channels (continued)

IDMAC's channel number	IC's channel name	Read/ Write	Source	Destination	Processing Flow Purpose
14	CB3	Read	Memory	IC VF	Graphics data for combining (viewfinder task)
15	CB4	Read	Memory	IC PP	Graphics data for combining (post-processing task)
11	CB5	Read	Memory	IC PP	Postprocessing data from memory
12	CB6	Read	Memory	IC VF	Preprocessing data from sensor stored in memory (for example Bayer)
5	CB7	Write	IC	Memory	Direct data from IC (sensor data) to memory
48	CB8	Write	ENC ROT	Memory	Preprocessing data after rotation (encoding task)
49	CB9	Write	VF ROT	Memory	Preprocessing data after rotation (viewfinder task)
45	CB10	Read	Memory	ENC ROT	Preprocessing data for rotation (encoding task)
46	CB11	Read	Memory	VF ROT	Preprocessing data for rotation (viewfinder task)
50	CB12	Write	PP ROT	Memory	Postprocessing data after rotation
47	CB13	Read	Memory	PP ROT	Postprocessing data for rotation

37.4.5.8 IC restrictions

- The input's frame width to the IC must be a multiplication of 8 pixels
- When performing resizing the frame width must be multiple of burst size - 8 or 16 pixels as defined by CB#_BURST_16 parameter.

37.4.5.9 IC bridge

The IC sub-block utilizes a single memory to serve read and write channels. These memories are the IBM, RM and MPM. The IDMAC has separate mechanism to handle read and write channels. As a result, a contention between read and write channels may occur on each one of the memories. In order to resolve the contention, an IC bridge sub-block is connected between the channels associated with this memory.

The bridge prioritizes a read channel over a write channel. In order to avoid starvation of the write channels, the user can limit the maximum consecutive requests of the same channel that will be served. The limitation is done by programming the memory's corresponding <>_brdg_max_rq field.

37.4.6 Display port

The display port handles all the IPU features targeted for controlling and sending data to the display. The display port consists of 4 modules.

DC - a display controller,

DP - a display processor,

DMFC - a display multi-FIFO controller

DI - a display interface. The DI is instantiated twice to provide two symmetrical display interfaces.

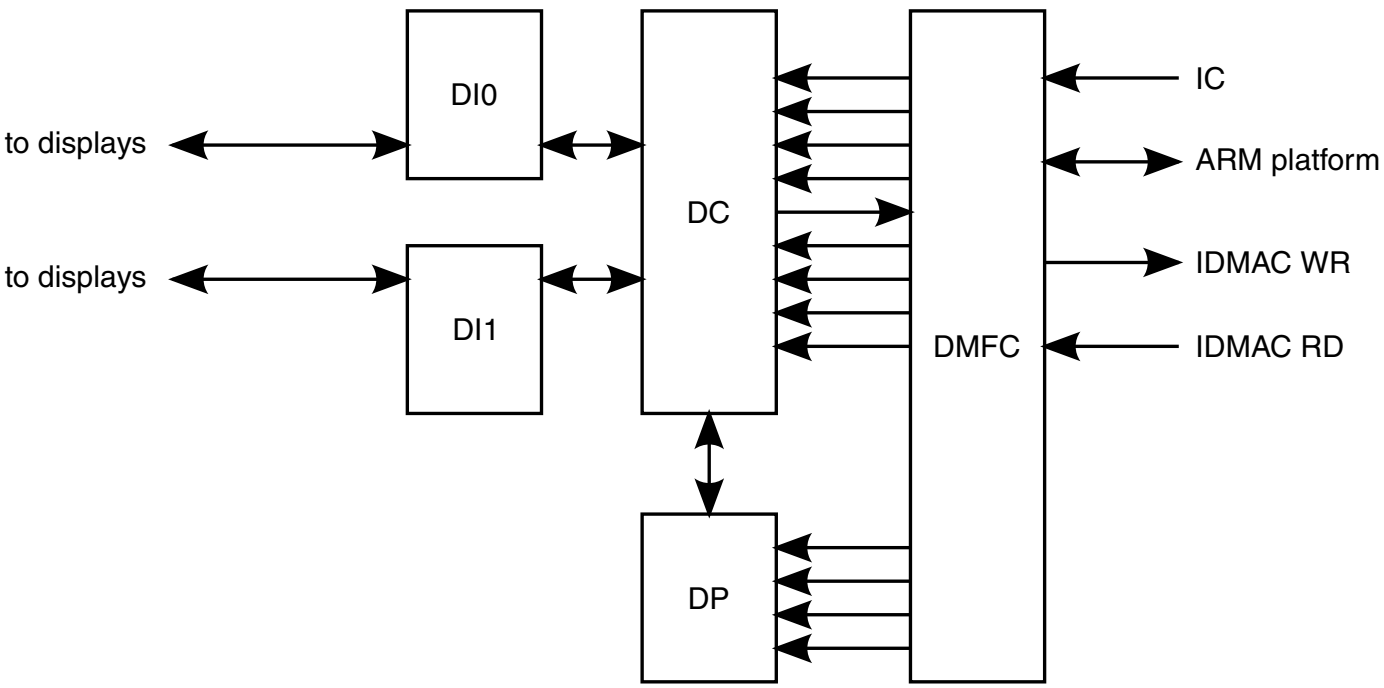


Figure 37-27. The display port

37.4.6.1 Display ports channels

The display port send data to the display over several channels. The data source may be the system's memory (via the IDMAC) the IC block and the ARM platform. The display port can read data from the display and send it to the ARM platform or IDMAC.

The data is routed over channels. The table below maps the IDMAC channels to DMFC, DC, DP channels and describes each channel.

Table 37-23. Display port channels

IDMAC's channel number	Display port's destination	DMFC channel number	DC channel number	Corresponding alpha channel	channel usage
21	DC	Programmable using dmfc_ic_in_port	Programmable using dmfc_ic_in_port	NA	This channel is coming from the IC module. When data is coming from the IC, the IC channel replaces one of the IDMAC's channels connected to the DMFC. When the IC_DMFC_SEL bit is set the output of the IC is routed to the DMFC. This channel can be routed to the DC channels 1,2,5B,5F,6B,6F. Routing this channel to the DC channel is done with the dmfc_ic_in_port bits. The DC's channel allocated to the IC channel can't be used for data coming from other source.
23	DP	5B	5	51	This channel is for the DP's primary flow. When a single plane is used, the data should be sent over this channel. When 2 planes are combined in this flow, the second plane should come on channel 27.
24	DP	6B	6	52	This channel is for the DP's secondary flow. When a single plane is used, the data should be sent over this channel. When 2 planes are combined in this flow, the second plane should come on channel 29.
27	DP	5F	5	31	This channel is for the DP's main flow. When a single plane is used, the data should be sent over channel 23 and this channel should not be used. When 2 planes are combined in this flow, one plane should come on channel 23 and the other plane on this channel.
28	DC	1	1	NA	This channel can serve sync and async flows. When channel 28 is connected to DI0, channel 23 must be connected to DI1 even if ch23 is not used. This is done by programming the PROG_DISP_ID_5 field
29	DP	6F	6	33	This channel is for the DP's secondary flow. When a single plane is used, the data should be sent over channel 24 and this channel should not be used. When 2 planes are combined in this flow, one plane should come on channel 24 and the other plane on this channel.
40	DC	0	0	NA	This is a read channel
41	DC	2	2	NA	This channel can serve only async flow
42	DC	1C		NA	Command stream. See Display port's restrictions
43	DC	2C		NA	Command stream. See Display port's restrictions
44	DC	3		NA	Mask channel. This mask channel can be associated with channel 23 or channel 28

37.4.6.2 Supported display interfaces

- The display port has 2 DI interfaces. Each interface can handle up to 3 displays.

- The total number of supported displays is 4.
- Each DI can handle up to 2 async interface - only one of them can be serial interface.
- Each DI can handle one synchronous interface. Asynchronous displays that are accessed in synchronous mode are considered synchronous interface.

37.4.6.2.1 Synchronous Interfaces

The DI supports the following synchronous display interfaces.

1. Synchronous generic interfaces to TFT dumb displays or RGB interfaces of smart displays.
2. Synchronous interfaces to Sharp displays.
3. Synchronous interfaces to TV encoders:
   - PAL
   - NTSC

   TV interfaces can operate in progressive or interlaced modes.
4. Synchronous interface to a graphic accelerator
5. BT.656
6. BT.1120

   BT.1120 and BT.655 support

   BT.1120 and BT.656 are supported. Only video data is supported, sending data during blanking intervals is not supported. The component size is always 8 bit.

37.4.6.2.2 Asynchronous Parallel Interfaces

The DI has a flexible asynchronous interface. The interface include 2 chip selects (CS) and 7 general purpose control signals.

The user can decide which of the 7 signals will be associated to each one of the asynchronous displays (up to 2 displays per DI). The using can configured some of the control signals to be shared by more than one display.

37.4.6.3 Display port's bandwidth

When the IPU clock (HSP_CLK) is equal to 264Mhz, the peak bandwidth supported by the display port is as follows

For on chip devices (like on chip MIPI-DPI bridge)

- 240 Mega-Accesses/Sec if the DI clock is derived from an external to IPU source (like another PLL)
- 264 Mega-Accesses/Sec if the DI clock is derived from the IPU clock (HSP_CLK)

For off chip devices (Like an external LCD)

- 170 Mega-Accesses/Sec if the DI clock is derived from an external to IPU source (like another PLL)
- 180 Mega-Accesses/Sec if the DI clock is derived from the IPU clock (HSP_CLK)

An access can be a pixel, component or generic data.

37.4.6.4 Display Dual Mode

This mode is useful for smart display with synchronous interface. The data is sent to the display only when the data has changed or when the display processor's settings are changed. The physical interface to this display has synchronous interface's characteristics.

37.4.6.5 Display Errors

There are a few types of errors that may impact the image sent to the display. The IPU provides some automatic mechanisms, which allow the system to overcome these errors.

37.4.6.5.1 Data starvation errors

These types of errors may happen for synchronous displays if the data is ready at the DI to be sent to the display at the time point it is required. This anomaly may happen if the system is very loaded and the IDMAC was not able to read data from the external memory and provide it to the DMFC.

Frame boundary errors

In case that a new frame should be sent to the display but the data of the previous frame was not sent completely, the IPU will reset the display modules and flush the internal buffers, as the IPU expects that the new frame indication will be triggered at least 2 lines (blanking interval) before the actual data is to be sent. The should recover and be ready with the data of the new frame within the blanking interval period.

Error within a frame

In case that a pixel was missed within a frame i.e. the pixel did not arrived to the DI on time. The IPU has 2 ways to handle the situation. The mode is selected by configuring the DI#_ERR_TREATMENT bit.

Redo the last access. The IPU will keep sending the last access to the display. The IPU will drop any pixel that arrive till the end of the line. If the data of the next line arrives correctly the IPU will continue working normally as overcame the problem. In case that the data of the next line is also incorrect the IPU will perform the same procedure as described on frame boundary errors.

Freeze the clock. In this mode, the IPU freezes the clock sent to the display till the correct data arrives. In order to avoid a case where the clock is frozen forever and the system is stuck: when the clock is frozen, a watchdog timer starts counting. When the timer completed counting the IPU will perform the same procedure as described on frame boundary errors. The number of cycles to be counted by the watchdog timer is defined according to the DI#_WATCHDOG_MODE bits.

37.4.6.5.2 Anti tearing errors

In case of anti tearing errors, the IPU indicates about the error by asserting the corresponding DC_TEARING_ERR_# bit.

See also Antitearing control.

37.4.6.6 Display port's restrictions

- In case where 2 synchronous flows are used and additional asynchronous flow via DP is used. The asynchronous flow via DP can be targeted to the same DI that the synchronous flow via DP is targeted.
- There are only 2 command channels (42 and 43). Command channels are associated with data channels (24,28,41).
  - When channel 28 is associated with a command, the command stream will come from channel 42
  - When channel 41 is associated with a command, the command stream will come from channel 43
  - Channel 24 can be associated with a command stream only if channel 28 or channel 41 do not use a command stream. If channel 28 is not associated with a command stream then channel 24 can be associated with channel 42. If channel 41 is not associated with a command stream and channel 28 is associated with a command stream then channel 24 can be associated with channel 43.

- A channel that uses an alternate flow, cannot be associated with a command stream (ch. 24 or ch. 41)
- In case of a synchronous display using external clock, The DI where the synchronous display is connected to can be connected to another async display. But, it can support only the write direction. Read via the asynchronous interface cannot be performed if this DI uses external clock

37.4.7 DC - Display Controller

IPU handles few display flows supporting few displays.

The IPU's flows' data sources can be the ARM platform, the system's memory, a camera or an external device connected on the display's port such as an external graphic accelerator.

The data's destination can be any device connected on one of the DI ports.

The DC controls the flows coming to and from the DI port. The DC manages the flows, decides which flows are currently active and when each flow is activated. The arbitrates between the active flows, gets the data from the predefined source and distribute it to the correct DI.

The DC's core is the microcode. The microcode contains a set of routine. A routine is built of a set of commands stored in the template's (microcode) memory. For each event (like new frame, end of frame etc.) a specific routine is executed. The user writes the routines and map them to a specific events. The routine contains instructions to the DC about the way of handling the data/address/commands associated with the display. The routine may contain information about the data's mapping, about waveform's characteristics, and more.

The figure below shows the micro architecture diagram for the DC block

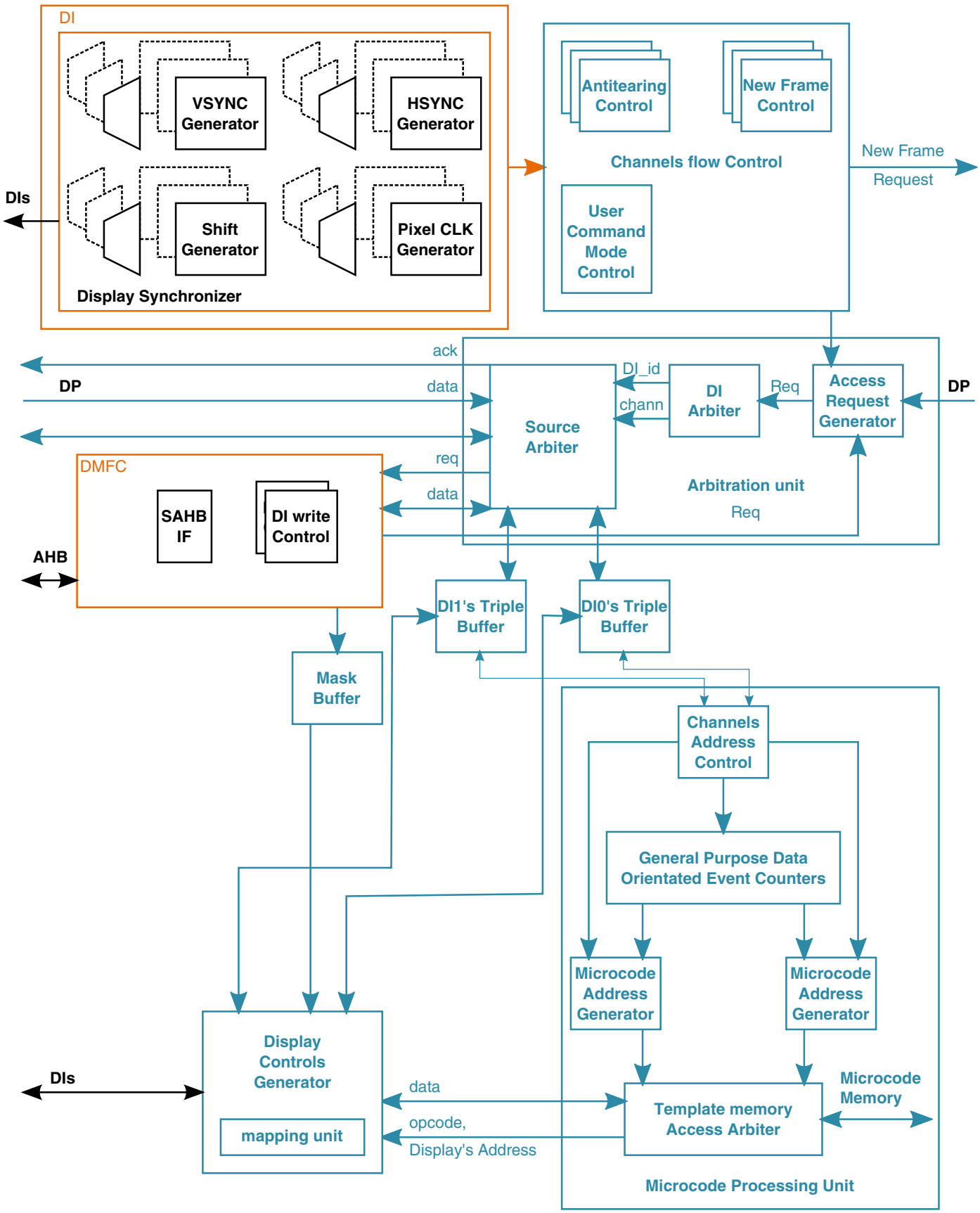


Figure 37-28. DC - Display Controller Block Diagram

37.4.7.1 Channels flow control

37.4.7.1.1 New Frame control

The channels flow control schedules the flows handled by the DC

For asynchronous flows the scheduling is done according to a request coming from the Frame Synchronization Unit (FSU) on the control sub-block (CM).

For synchronous flows the scheduling is done according to a trigger that is generated by a timer located on the one of the display's interface blocks (DI)

37.4.7.1.2 Antitearing control

Anti tearing mechanism uses a signal indicating on a display's refresh of a frame.

The supported tearing elimination triggers can be:

- An internally generated VSYNC signal
- A VSYNC signal generated by the display. The data source is the memory.
- A VSYNC signal coming from the CSI

The DC has the capability to avoid image tearing. For asynchronous flows where the source of the data is the system's memory or postprocessing, the DC monitors the position of a display's refresh pointer. Writing to the display is started only after crossing the window start point by the display refresh pointer. After that, writing to the display is allowed only when a write pointer does not advance beyond the refresh pointer. To provide anti-tearing mode, a window start time (in rows) must be defined in the PROG_START_TIME_1, PROG_START_TIME_2, PROG_START_TIME_5, PROG_START_TIME_6 registers for the corresponding channels.

The antitearing mechanism is limited to a case where only asynchronous flows are handle via the target DI.

In the case when tearing cannot be avoided (when the refresh rate is too high and the refresh pointer overtakes the write pointer after full refresh cycle), an error interrupt is generated. Tearing elimination mode can be disabled via the PROG_CHAN_TYP field for the corresponding channel.

37.4.7.1.3 User command mode control

A user may prepare in the system's memory a buffer that holds commends to be sent to the external device. The command buffer includes the same amount of lines as the data buffer. The line of commands are sent to the display line by line. A line of data is sent following each line of commands. This mode is activated by programming the PROG_CHAN_TYP of the corresponding channel.

The structure of the command is as follows:

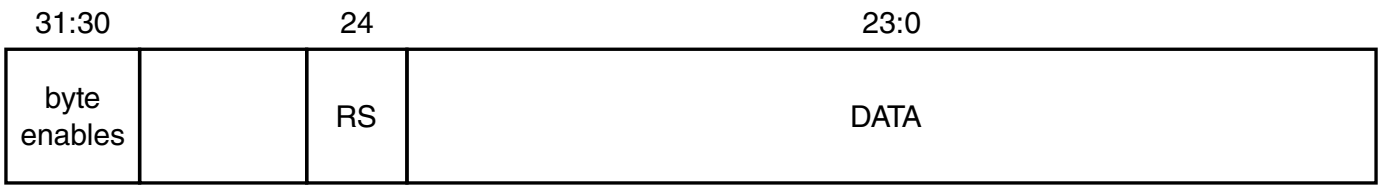


Figure 37-29. Structure of a command word

37.4.7.2 Arbitration Unit

This unit arbitrates the requests coming from the DMFC and from the DP and sends them to corresponding DI.

37.4.7.2.1 Access request generator

The requests coming from the DMFC or DP are sorted according to the target DI and then served according to a hard coded priority. The priority order is Sync flow, ARM platform access, IDMAC's Async flows.

37.4.7.2.2 DI arbiter

This units arbitrates between the DIs. The priority is hard coded. The priority order is Sync flow, ARM platform access, IDMAC's Async flows.

In case of 2 simultaneous requests to different DIs with the same priority the requests are served in a Round-Robin fashion. In case of 2 simultaneous Sync flow requests to different DI, the user can bypass the Round Robin mechanism and prioritize one DI on the other according to the SYNC_PRIORITY_1 & SYNC_PRIORITY_5 bits. Setting low priority to both of these channels is forbidden.

37.4.7.2.3 Source arbiter

Once the source of the request to be served was selected and the target DI was chosen, this unit routes the request and all the signals, associated with it, to the correct triple buffer and to the correct source of the data.

37.4.7.3 Microcode processing unit

The main control unit of the DC is the Microcode processing unit (MPU).

The data coming to the DC may be associated with some additional information like new frame, new line, new address etc. The information is processed in the MPU. The MPU executes the associated routine. The routine includes a set of instructions of the actions to be performed by the DC and DI.

37.4.7.3.1 Channels address control

This unit controls the display's address for each channel. This unit is responsible for defining the next address to be accessed (by incrementing or jumping). Stores special events flags (like EOF, EOL etc.). Based on the display's address and the special events, the type of the routines to be executed is defined.

37.4.7.3.2 General purpose Data oriented events counters

A user may define up to 4 general purpose events. The events are triggered by a standard event (NF, NL). The standard event restarts a counter, when the counter completes counting the user's general purpose event is asserted. This event activates a routine like any other events

37.4.7.3.3 Microcode address generator

This unit calculates the physical address of the template memory where the event associated routine resides. In addition, these unit arbitrates between simultaneous events and select the event to be served.

The arbitration is done according to a user defined priority. The priority of each event is set according to the #_CHAN_PRIORITY_CHAN_# bits of each channel. There are 2 modes of arbitration:

- Serving all the pending events according to the priority
- Serving only the highest priority event while ignoring all the other events

The arbitration mode is defined according to the CHAN_MASK_DEFAULT bit of each channel.

37.4.7.3.4 Template's Memory Access Arbiter

This unit gets memory access requests from the 2 microcode address generator units and arbitrates between them in a Round Robin fashion. In case that only one of the requests belongs to a synchronous flow, this request is selected.

37.4.7.4 DC's Template structure

The template memory contains 256 template words. Each template word is a 42 bits words. Accessing a template word require 2 accesses (32bit each).

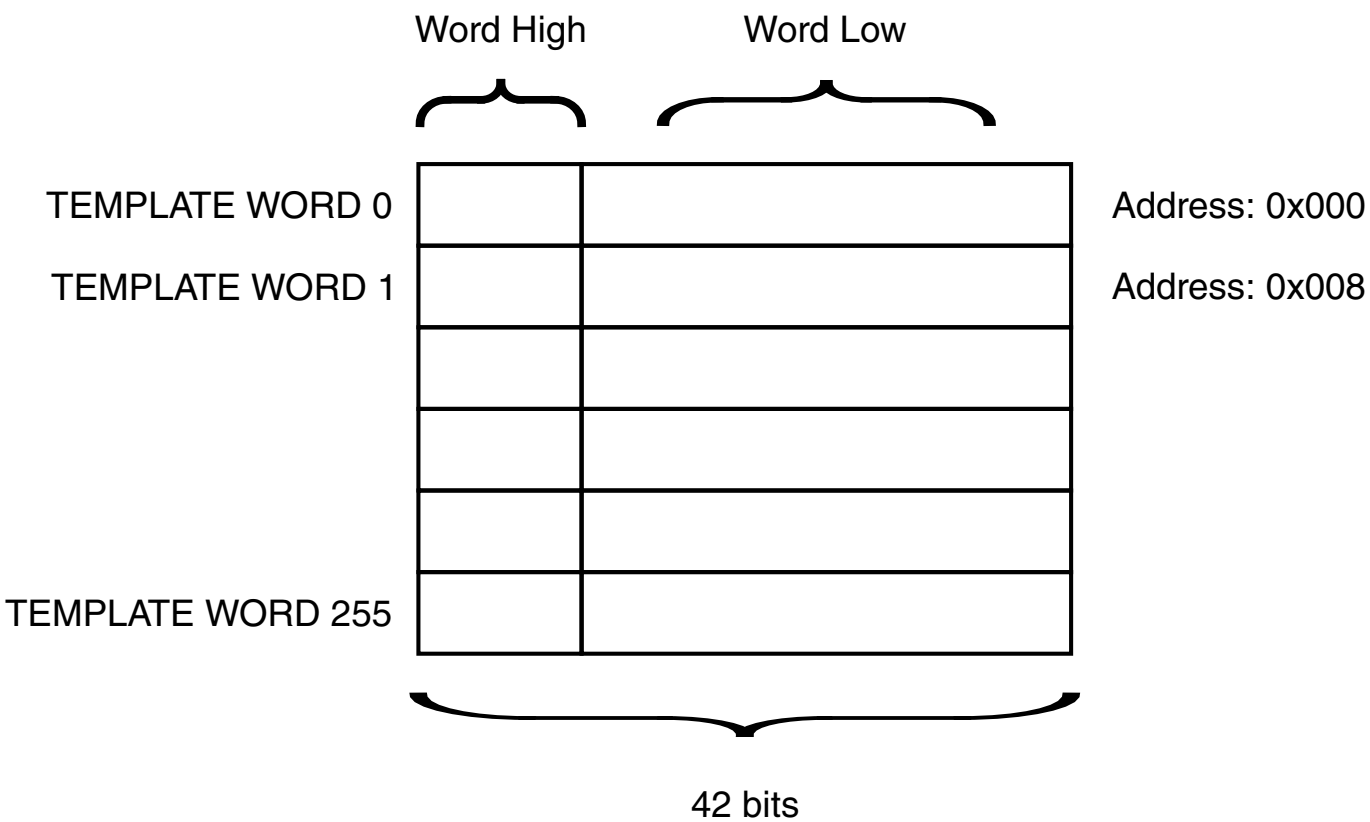


Figure 37-30. Template's structure

37.4.7.4.1 DC template's memory map

Table 37-24. DC template's memory map

0x1F80000 DC_MICROCODE_W0_L																
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	OPERAND												MAPPING			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	MAP PING	WAVEFORM				GLUELOGIC							SYNC			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1F80004 DC_MICROCODE_W0_H																
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	STOP	OPCODE								
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DC template's fields description

Table 37-25. DC template's fields description

Field	Description
STOP	Stop bit - This bit should be set in order to indicate that the current command is the last command of the routine
OPCODE	The command's code
OPERAND	The command's operand - for some of the commands this field can hold a parameter associated with the command
MAPPING	The MAPPING field holds a pointer to a register holding 3 fields: MAPPING_PNTR_BYTE0_X, MAPPING_PNTR_BYTE1_X, MAPPING_PNTR_BYTE2_X. This pointers point to sets of OFFSET and MASK parameters that define the mapping scheme. MAPPING = 0 means that mapping is disabled. The value in this field should be incremented by 1 to get the correct X pointer value In order to point to MAPPING_PNTR_BYTE2_0, MAPPING_PNTR_BYTE1_0, MAPPING_PNTR_BYTE0_0 the user should write 1 to the MAPPING field
WAVEFORM	For data oriented output pins. The IPU has 4 waveform generator units. The IPU holds 12 sets of waveforms' configuration registers called DI0_DW_GEN_<i> and DI1_DW_GEN_<i>

Table continues on the next page...

Table 37-25. DC template's fields description (continued)

Field	Description
	The WAVEFORM field defines which one of the 12 waveforms' configuration registers is used. The DI1_DW_GEN_X register holds a pointer to one of the 4 waveform generators units for each of the data oriented pins. 0 - The waveform of the data oriented output pins is not affected 1 -Points to DI0_DW_GEN_0 or DI1_DW_GEN_0 2 - Points to DI0_DW_GEN_1 or DI1_DW_GEN_1 ... 12 - Points to DI0_DW_GEN_11 or DI1_DW_GEN_11
GLUELOGIC	For signals generated by waveform generator #3; This field provides extra flexibility on the signals waveform
	GLUELOGIC[6] - This bit defines if we are in clock mode (1) or CS mode(0). 1- clock mode When the command is related to the display clock's pin then only if we are in clock mode, GLUELOGIC[5:4] are valid. 0- CS mode When the command is related to the CS pin then only if we are in CS mode, GLUELOGIC[3:0] are valid.
	GLUELOGIC[5:4] - clock mode settings 00 - Freeze the display clock following the execution of the current command 01 - Freeze the display clock before the execution of the next command to be executed 10 - Enable (unfreeze) the display clock following the execution of the current command 11 - Enable (unfreeze) the display clock before the execution of the next command to be executed
	GLUELOGIC[3] - CS mode settings 1 - Once the signal is asserted then it remains asserted (high or low according to the polarity) 0 - No impact on the waveform
	GLUELOGIC[2] - CS mode settings 1 - Once the signal is negated then it remains negated (high or low according to the polarity) 0 - No impact on the waveform
	GLUELOGIC[1] - CS mode settings 1- The current waveform can be attached to the previous waveform. If the previous waveform was asserted and the current waveform start asserted the two waveforms will be attached so the signals' waveforms will be consecutive. This impact the behavior of the previous waveform. This can be done only if GLUELOGIC[0] of the previous waveform is set to 1 0 - No impact on the waveform
	GLUELOGIC[0] - CS mode settings 1 - this bit allows the next waveform to be attached to the current waveform.
SYNC	The data associated with this command should be synchronized to the DI's one of gen_time_sync generators' output. 0000 - No sync. The data is sent without any synchronization to any event 0001 - Sync with unit #1 0010 - Sync with unit #2 ...

Table 37-25. DC template's fields description

Field	Description
	1010 - Sync with unit #10 1011 - 1111 Reserved

DC template's command description

The diagram below illustrates the command processing flow. The command is being fetched from the microcode memory. The Opcode is decoded. According to the decoding several controls are sent to the processing unit. In addition the processing unit gets the pixel's data and the display's address. The processed data/ address is stored in an internal register. There are 2 types of commands

HOLD - In this type of commands the data/address is stored in the register and not sent to the DI. The data/address is held in the register for further processing in the following commands.

WRITE - In this type of commands the data/address is stored in the register and also sent to the DI.

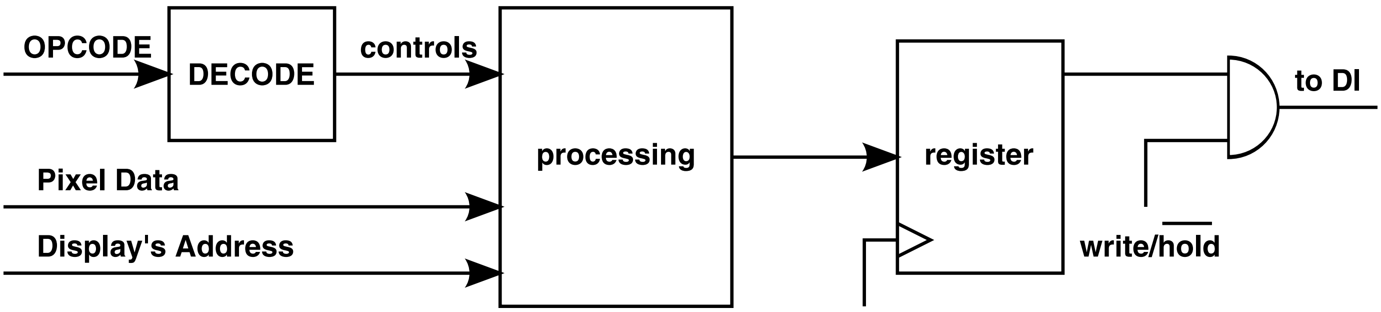


Figure 37-31. Opcode processing

The table below describes the DC's Commands.

Table 37-26. DC template's commands description

Com mand	COMMAND[41:0]																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HLG	S	0	0	0	0	DATA																																0	0	0	0	0
	Hold 32bit word in an internal register for further processing. DATA is a general purpose data to be held.																																									

Table continues on the next page...

Table 37-26. DC template's commands description (continued)

Command	COMMAND[41:0]																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRG	S	0	1	DATA																								WAVEFORM				GLUELOGIC							SYNC			
	Write 24bit word to the DI and Hold the word in register. DATA is a general purpose data to be written.																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HLOA	S	1	0	1	0	af	DATA																MAPPPING					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Hold the display's address in an internal register for further processing. af=0: No shift af=1: 8 bit right shift DATA is a 16bit general purpose data. This data is ORed with the 16 MSB of the mapped address.																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WROA	S	1	1	1	0	af	DATA																MAPPPING					WAVEFORM				GLUELOGIC							SYNC			
	Write address to the display and Hold address in register. af=0: No shift af=1: 8 bit right shift																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HLOD	s	1	0	0	0	0	DATA																MAPPPING					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Hold data in register. DATA is a 16bit general purpose data. This data is ORed with the 16 MSB of the mapped data coming from the data's source IDMAC or MCU.																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WROD	s	1	1	0	0	0	DATA																MAPPPING					WAVEFORM				GLUELOGIC							SYNC			
	Write data to DI and Hold data in register. DATA is a 16bit general purpose data. This data is ORed with the 16 MSB of the mapped data coming from the data's source IDMAC or MCU.																																									
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HLOAR	S	1	0	0	0	1	1	1	af	0	0	0	0	0	0	0	0	0	0	0	0	0	MAPPPING					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Adding Mapped Address to held data and hold in an internal register. af=0: No shift af=1: 8 bit right shift																																									

Table continues on the next page...

Table 37-26. DC template's commands description (continued)

Command	COMMAND[41:0]																																													
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
WROAR	S	1	1	0	0	1	1	1	af	0	0	0	0	0	0	0	0	0	0	0	0	0	MAPPPING					WAVEFORM				GLUELOGIC												SYNC			
	Adding Mapped Address to held data. Write to DI and hold in register af=0: No shift af=1: 8 bit right shift																																													
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
HLODR	S	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	MAPPPING					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	Adding Mapped Data to held data and hold in register.																																													
	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
WRODR	S	1	1	0	0	1	1	0	0	M2	M1	M0	0	0	0	0	0	0	0	0	0	0	MAPPPING					WAVEFORM				GLUELOGIC												SYNC			
	Adding Mapped Data to held data. Write to DI and hold in register. M0: 0: a new data[7:0] before mapping 1: a previous access data [7:0] before mapping M1: 0: a new data[15:8] before mapping 1: a previous access data [15:8] before mapping M2: 0: a new data[31:16] before mapping 1: a previous access data [31:16] before mapping If (M2 = M1 = M0 = 0) than the command performs: Adding a new Mapped Data to a held data in an internal register and write it to display else a display's data will be combined from a new data and previous data according to M-flags. The combined data will be mapped according to MAPPING and sent to a display. Examples: previous_data = 0x89ABCDEF new_data = 0x12345678 held_data (previous_data after mapping) = 0x0000EF Current MAPPING mode: new_data & 0x00ffff00 Output: If M2 = M1 = M0 = 0) than: Output = new_data OR held_data = 0x3456EF If M0 = 0, M1= 1,M2 = 1 than: Output = MAPPING ({previous_data[31:8], new_data[7:0]}) = MAPPING(0x89ABCD78) = 0x00ABCD00)																																													

Table continues on the next page...

Table 37-26. DC template's commands description (continued)

<table>
<tr><th>Command</th><th colspan="42">COMMAND[41:0]</th></tr>
<tr><td></td><td colspan="42">If M0 = 0, M1= 1,M2 = 0 than:
Output = MAPPING ({new_data[31:16], previous_data[15:8], new_data[7:0]}) = MAPPING(0x1234CD78) = 0x0034CD00)
if M0 = 1, M1= 0, M2 = 0 than:
Output = MAPPING ({new_data[31:8], previous_data[7:0]}) = MAPPING(0x123456EF) = 0x00345600)</td></tr>
<tr><td></td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">WRBC</td><td>S</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td colspan="5">MAPPPING</td><td colspan="4">WAVEFORM</td><td colspan="7">GLUELOGIC</td><td colspan="4">SYNC</td></tr>
<tr><td colspan="42">Merge 1 bit mask channel with mapped data. Hold in register and write to DI.
Mask data is coming from the DC's mask channel.</td></tr>
<tr><td></td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">WCLK</td><td>S</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td colspan="13">N_CLK_OPERAND</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="42">Wait N_CLK_OPERAND clocks.
N_CLK_OPERNAD is the number of DI_CLK cycles to wait.</td></tr>
<tr><td></td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">WSTS-I</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td colspan="13">N_CLK_OPERAND</td><td colspan="5">MAPPPING</td><td colspan="4">WAVEFORM</td><td colspan="7">GLUELOGIC</td><td colspan="4">SYNC</td></tr>
<tr><td colspan="42">Wait for Status I
Read from the display and compare to a predefined PASSWORD. If the read data is equal to the PASSWORD, then continue. If not, redo the read & compare cycle.
N_CLK_OPERAND is the number of DI_CLK cycles to wait before latching the data coming from the DI.
DI_READ_DATA_ACK_VALUE_0 DI_READ_DATA_MASK_0</td></tr>
<tr><td></td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">WSTS-II</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td colspan="13">N_CLK_OPERAND</td><td colspan="5">MAPPPING</td><td colspan="4">WAVEFORM</td><td colspan="7">GLUELOGIC</td><td colspan="4">SYNC</td></tr>
<tr><td colspan="42">Wait for Status II
Read from the display and compare to a predefined PASSWORD. If the read data is equal to the PASSWORD, then continue. If not, redo the read & compare cycle.
N_CLK_OPERAND is the number of DI_CLK cycles to wait before latching the data coming from the DI. WSTS-II command must be followed by a WSTS-I command.
This command is useful in case that the PASSWORD is read in 2 accesses.
The comparison will be done only after the execution of the WSTS-I command.</td></tr>
<tr><td></td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">WSTS-III</td><td>S</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td colspan="13">N_CLK_OPERAND</td><td colspan="5">MAPPPING</td><td colspan="4">WAVEFORM</td><td colspan="7">GLUELOGIC</td><td colspan="4">SYNC</td></tr>
<tr><td colspan="42">Wait for Status III</td></tr>
</table>

Table continues on the next page...

Table 37-26. DC template's commands description (continued)

Command	COMMAND[41:0]
	Read from the display and compare to a predefined PASSWORD. If the read data is equal to the PASSWORD, then continue. If not redo the read & compare cycle. N_CLK_OPERAND is the number of DI_CLK cycles to wait before latching the data coming from the DI. WSTS-III command must be followed by a WSTS-II command. This command is useful in case that the PASSWORD is read in 3 accesses. The comparison will be done only after the execution of the WSTS-I command. In case of a PASSWORD mismatch the read is done again. The entire cycle (WSTS-III -> WSTS-II -> WSTS-I) will be performed.
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
RD	S 1 0 0 0 1 0 0 0 N_CLK_OPERAND MAPPPING WAVEFORM GLUELOGIC SYNC
	Read data from DI N_CLK_OPERAND - means delay value in DI_CLK for display's data latching by DI, defined by user For serial display the N_CLK_OPERAND is fixed and should be set to 1 value.
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
WACK	S 1 0 0 0 1 1 0 1 0 N_CLK_OPERAND 0 0 0 0 0 WAVEFORM GLUELOGIC SYNC
	Wait for acknowledge N_CLK_OPERAND - Number of DI_CLK cycles to count before monitoring the ACK received from the display.
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
MSK	S 1 1 0 0 1 0 0 0 0 0 0 0 0 e0m e1m e2m e3m nfm nlm nfldm eofm eolm eofldm nadm ncm dm 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	Mask - Mask a specific event; In case of more than one pending events the user can mask some of the event and serve the others according to the priority. e0m - event 0 mask, defined by user e1m - event 1 mask, defined by user e2m - event 2 mask, defined by user e3m - event 3 mask, defined by user nfm - new frame mask, defined by user nlm - new line mask, defined by user nfldm - new field mask, defined by user eofm - end of frame mask, defined by user eolm - end of line mask, defined by user eofldm - end of field mask, defined by user nadm - new address mask, defined by user

Table continues on the next page...

Table 37-26. DC template's commands description (continued)

Command	COMMAND[41:0]
	ncm - new channel mask, defined by user dm - data mask, defined by user
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
HMA	S 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ADDRESS 0 0 0 0 0
	HOLD_MICROCODE_ADDRESS - hold operand in special Microcode address register 0. This register is used to store a pointer in the microcode to be used by the BMA command ADDRESS is a microcode address in the microcode memory defined by the user.
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
HMA1	S 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ADDRESS 0 0 0 0 0
	HOLD_MICROCODE_ADDRESS - hold operand in special Microcode address register 1. This register is used to store a pointer in the microcode to be used by the BMA command ADDRESS is a microcode address in the microcode memory defined by the user.
	41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
BMA	S 0 0 1 1 LF AF 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 N 0 0 SYNC
	BRANCH_MICROCODE_ADDRESS jump to Microcode address which is stored at an internal Microcode address register. The branch will be done to the ADDRESS stored with the HMA/HMA1 command. The BMA command manages the repeat cycles by one of the auto decrement counters (internal counters), the counter's id is defined by LF-bit (0/1) The N operand defines the N-number of routine's repetitions. In case of N=0. The BMA command functions as a JUMP command. As long as the internal counter didn't reached its predefined value, the microcode will branch to the pre defined address stored by the HMA/HMA1 commands. Selecting between HMA and HMA1 is done by the AF bit (it is calculated according to the "1 - AF" value). - That's the subroutine's microcode start address When the counter reached its predefined value, the microcode will branch to the pre defined address stored by the HMA/HMA1 commands. Selecting between HMA and HMA1 is done by the AF bit (it is calculated according to the "AF" value). - That's the subroutine's microcode return address BMA synchronization (SYNC field): The BMA command can be used for context switching between 2-channels. When switching between flows the user can run a routine for a new channel, which actually affects the previous flow. This feature is useful when a new flow breaks the current flow (post command for example): SYNC is used for display_id selection // 0 0 0 - current display // 0 0 1 - previous display // 1 0 0 - display 0 // 1 0 1 - display 1 // 1 1 0 - display 2 // 1 1 1 - display 3

37.4.7.5 Display controls' generator

This unit generates the control signals associated with the data that are sent to the DI block.

37.4.7.5.1 Bus Mapping Unit

The Bus Mapping Unit is responsible for programmable mapping of the input data and commands to the display interface format and vice versa. Address mapping is done by this unit as well (programmable via micro code).

The internal DI format for data and commands is a 24-bits word divided into three byte components (eight zeroes are added to MSB for 16-bits words from the DC). This word can be output or input in one, two, three or four cycles of the display clock.

The word coming from the memory is a 32bit word. The mapping operation is done on 24 bits only. The 24 bit are selected from the received 32 bit according to the W_SIZE_# field. The 24 bit input word is partitioned to a 3 bytes (3X8bits). Each byte can be mapped to any position at the 24bit output word. The exact position is set according to the MD_MASK & MD_OFFSET fields.

The MAPPING_PNTR_BYTE0_X, MAPPING_PNTR_BYTE1_X, MAPPING_PNTR_BYTE2_X fields holds the pointers for the MD_MASK & MD_OFFSET for each byte.

The MAPPING field holds a pointer to a register holding 3 fields: MAPPING_PNTR_BYTE0_X, MAPPING_PNTR_BYTE1_X, MAPPING_PNTR_BYTE2_X.

0 - no mapping i.e. 32 bits are sent as is.

1 - points to MAPPING_PNTR_BYTE0_0, MAPPING_PNTR_BYTE1_0, MAPPING_PNTR_BYTE2_0

2 - points to MAPPING_PNTR_BYTE0_1, MAPPING_PNTR_BYTE1_1, MAPPING_PNTR_BYTE2_1

...

30 - points to MAPPING_PNTR_BYTE0_29, MAPPING_PNTR_BYTE1_29, MAPPING_PNTR_BYTE2_29

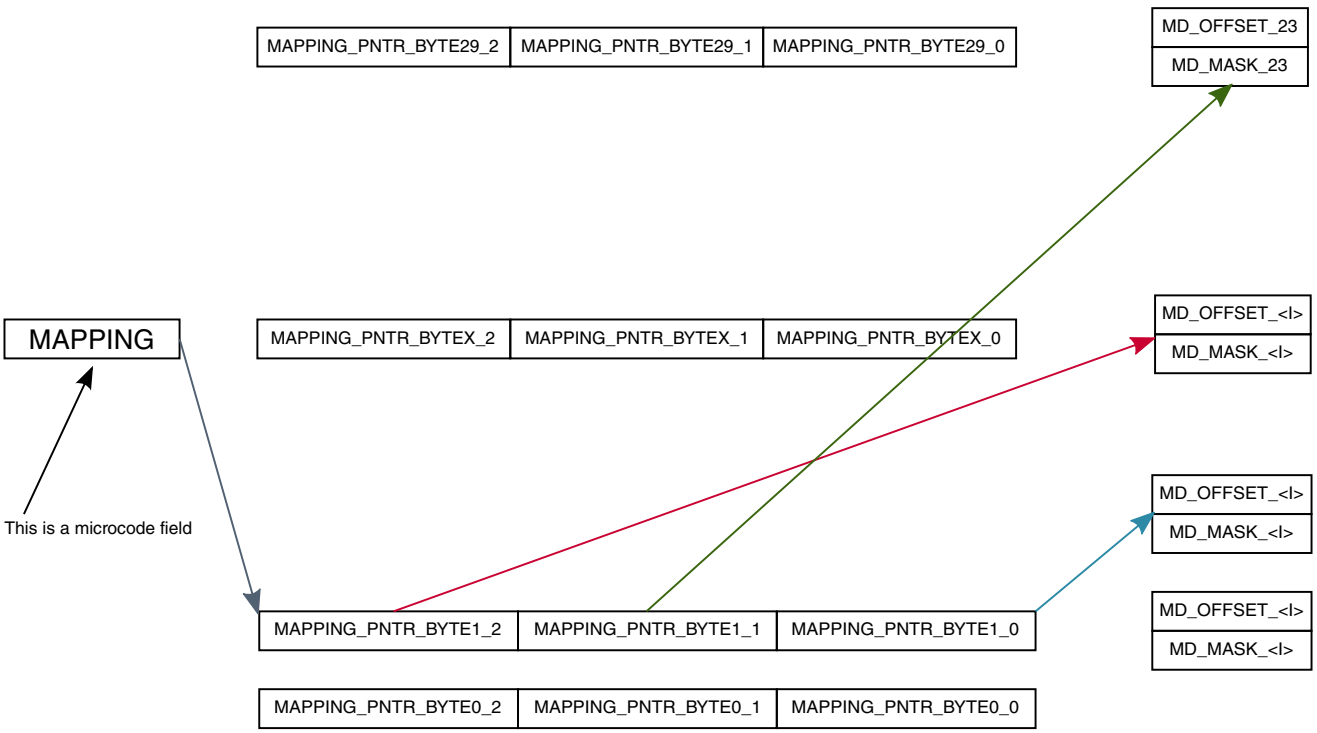
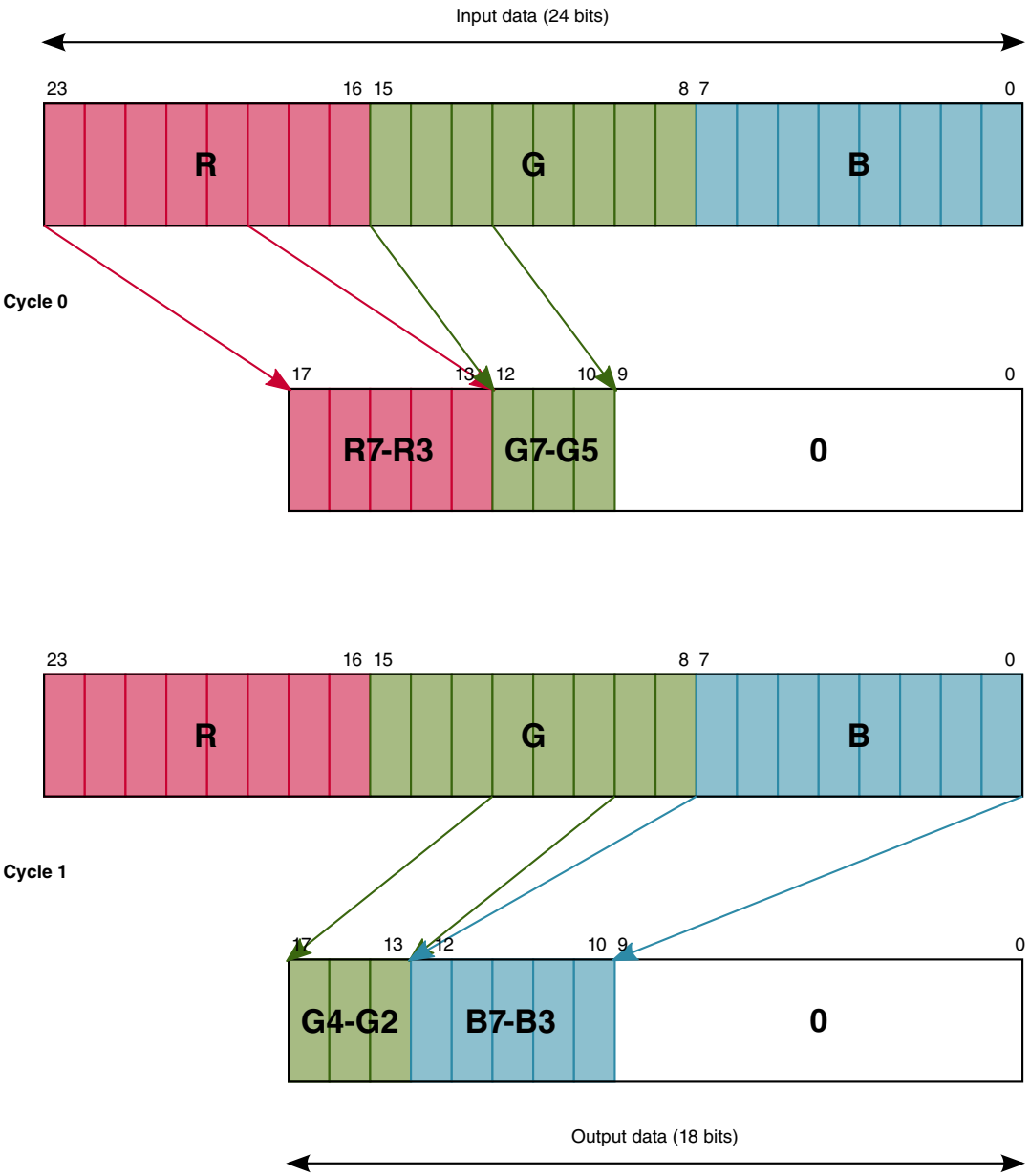


Figure 37-32. Mapping scheme

The mapping rule written in each of the Registers defines two types of parameters for the specific byte component and display:

1. Offsets of the byte component MSB relative to the output word LSB. Because the offsets can change dynamically, they are defined separately for the display clock cycles zero, one and two.
2. Numbers of the display clock cycles at which every bit of the byte component should be valid on the display bus. There are eight such 2-bit numbers in the Register.

Figure 37-33 presents an example of programming data packing for one of displays. Cycle 0 and Cycle1 in the diagram represent two separate atomic operations performed by the microcode template.



Cycle 0

MD_OFFSET_[R] - 0x11; MD_MASK_[R] - 0xF8

MD_OFFSET_[G] - 0xC; MD_MASK_[G] - 0xE0

MD_OFFSET[B] = 0x0; MD_MASK_[B] = 0x0

Cycle 1

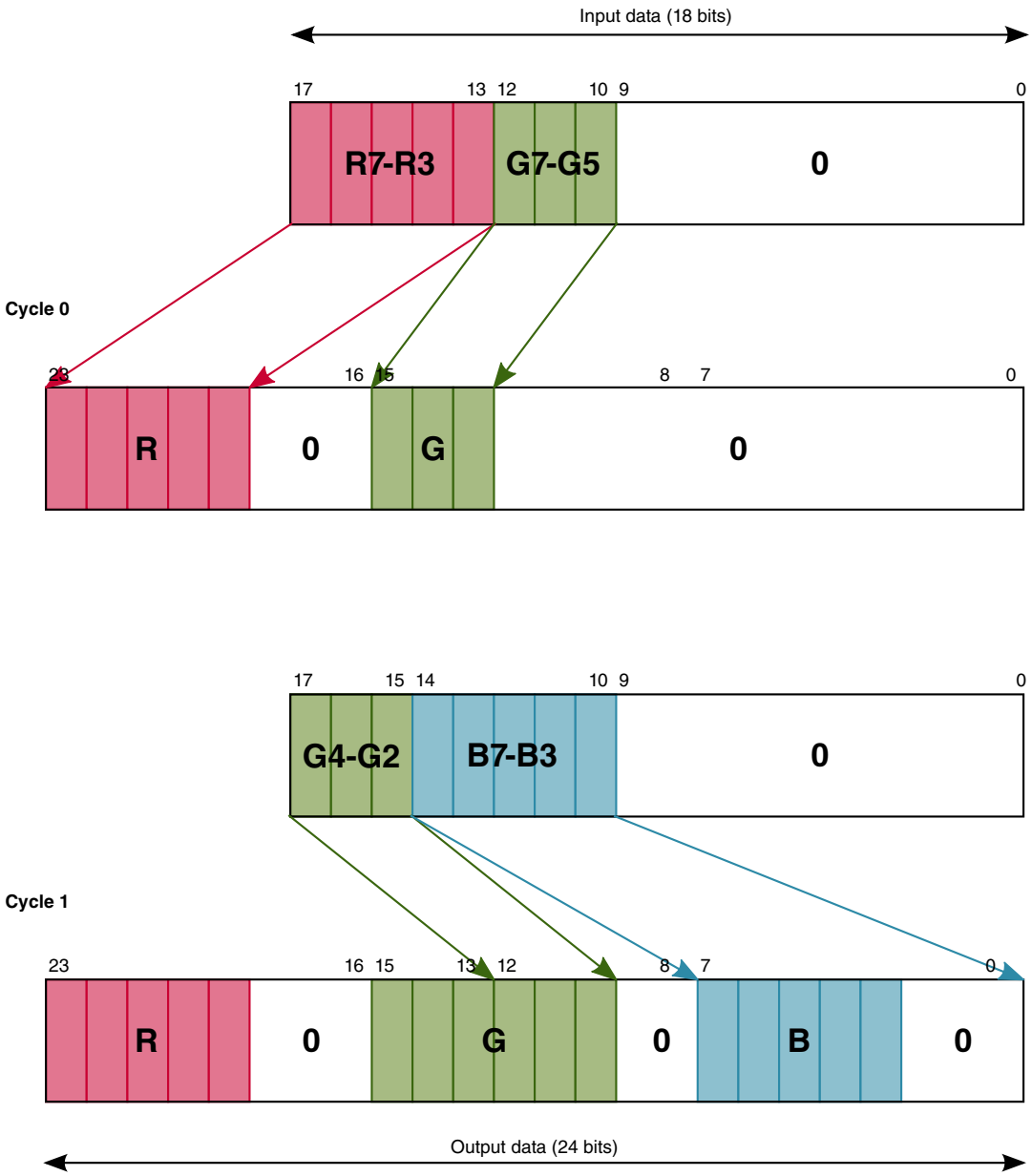
MD_OFFSET_[R] = 0x0; MD_MASK_[R] = 0x0

MD_OFFSET_[G] = 0x14; MD_MASK_[G] = 0x1C

MD_OFFSET_[B] = 0XE; MD_MASK_[B] = 0xF8

Figure 37-33. Example of Data Packing for Writing Data to the Display

The following figure presents an example of programming data packing for one of displays. Cycle 0 and Cycle1 in the diagram represent two separate atomic operations performed by the microcode template.



Cycle 0

MD_OFFSET_[R] - 0x11; MD_MASK_[R] - 0xF8

MD_OFFSET_[G] - 0xC; MD_MASK_[G] - 0xE0

MD_OFFSET[B] = 0x0; MD_MASK_[B] = 0x0

Cycle 1

MD_OFFSET_[R] = 0x0; MD_MASK_[R] = 0x0

MD_OFFSET_[G] = 0x14; MD_MASK_[G] = 0x1C

MD_OFFSET_[B] = 0XE; MD_MASK_[B] = 0xF8

Figure 37-34. Example of Data Unpacking for Reading Data from the Display

The same packing/unpacking registers are used for parallel and serial interface.

37.4.8 DMFC - Display Multi FIFO Controller

The following figure shows the block diagram for the DMFC block.

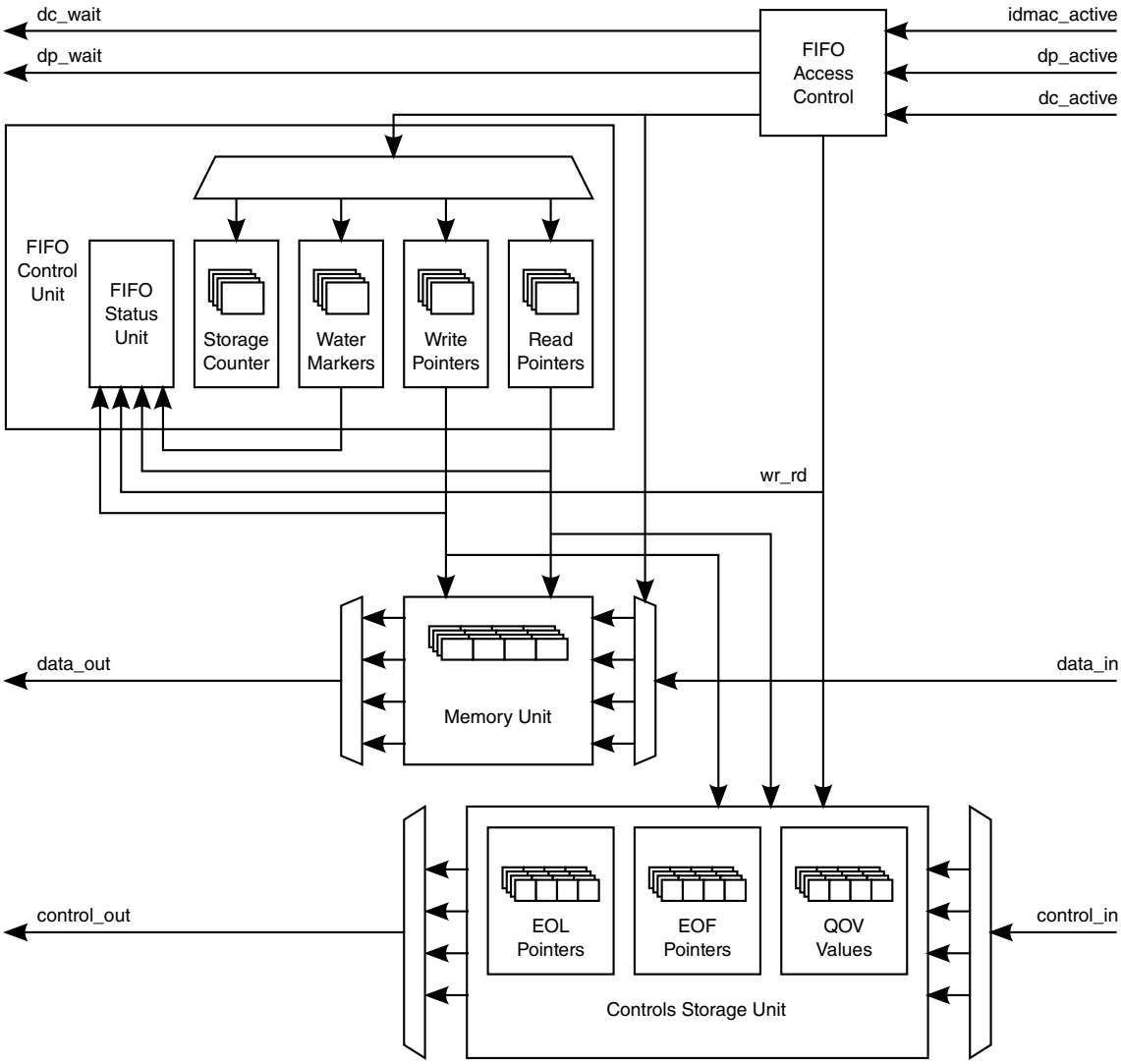


Figure 37-35. Display Multi Fifo Controller Block Diagram

The Display Multi Fifo Control manages Multi channels FIFOs. The DMFC serves the following clients:

- IDMAC - both read and write
- DP - read only
- DC - both read and write

- IC - write only
- AHB - both read and write

The DP and the DC read channels are physically attached to an IDMAC or an IC channel. As the IC has only one output channel connected to the DMFC. When the input is coming from the IC it replaces a channel that was physically attached to the IDMAC. The DMFC uses a single physical memory that serves the DP and DC read channels. The AHB accesses to the DC and the DC's write channel (read from display) use a separate physical memory.

The DMFC's write FIFO is built of 1024 entries of 128-bits each.

37.4.8.1 DP and DC read channels

Each one of the DP and the DC read (read from memory) channels is physically attached to an IDMAC read channel. A portion of the physical memory is allocated for each channel. The DMFC arbitrates between channels according to a predefined priority.

The DMFC controls each of the FIFOs

- Assert a request any time there's available place on the FIFO.
- Make sure that there's available place on the FIFO to accept the coming data.
- Optionally assert a watermark indication to avoid starvation

37.4.8.1.1 FIFO allocation to channels

The physical memory is partitioned to 8 segments. For each channel the user has to define the start address at a segment's boundary using the DMFC_ST_ADDR parameter.

The size of the FIFO allocated to a channel is defined by the DMFC_FIFO_SIZE parameter. The user must allocate the FIFO and avoid overlapping between FIFOs.

The DMFC hold few special indications like EOF, EOL, EOFILD. The most important one is end of line (EOL). If the size of the FIFO is shorter than or equal to the IDMAC line's length (FW) than no special restrictions on the DMFC usage.

If the size of the FIFO is greater than the IDMAC line's length (FW) than the user has to be aware to the following restriction for each channel.

The DMFC has two operation modes which are distinguished by the wait4eot bit. For each channel the DMFC can store a maximum number of EOL indication. The maximum number of EOL indications of EOL is given in the table below.

Table 37-27. DMFC's number of EOL indications

IDMAC's Channel	Maximum lines on the FIFO
23	Up to 3 lines
27	Up to 2 lines
28	Up to 2 lines
Other	Up to 1 line

If the use case is that the number of EOL indications cannot exceed the maximum number of EOL indications than the user should have the wait4eot cleared.

If the use case is that the number of EOL indication can exceed the maximum number of EOL indications than the user should have the wait4eot set.

Having the wait4eot bit set has performance impact as the DMFC analyzes the data prior to sending it to the destination. In addition the DMFC cannot utilize the entire FIFO allocated to this channel.

The user need to specify the burst size of the IDMAC by setting the DMFC_BURST_SIZE field. This field must match the IDMAC settings. In case that the IDMAC's burst size is not a power-of-2 number, the value of this field should be rounded up to the nearest power-of-2 number. The burst size must not be greater than the FIFO's size.

37.4.8.1.2 Arbitration between channels

The arbitration between channels is fully hardware controlled. IDMAC has the highest priority. Then the synchronous channels. Then the asynchronous channels.

37.4.8.1.3 Watermark

The DMFC can generate a water mark signal for each channel. The watermark signal is sent to the IDMAC and dynamically increases the channels priority on the IDMAC's arbitration.

The watermark feature is enabled by the DMFC_WM_EN bit. The FIFO is partitioned to bursts. The user can set the watermark level at a burst boundary.

The watermark signal is set when the number of bursts on the FIFO + the number of already requested burst is smaller than the value specified on DMFC_WM_SET bit.

The watermark signal is cleared when the number of bursts on the FIFO + the number of already requested burst is greater than the value specified on DMFC_WM_CLR bit.

DMFC_WM_SET must be smaller than DMFC_WM_CLR.

37.4.8.2 IC interface

One of the IDMAC channels can be replaced by a flow coming from the IC using the DMFC_IC_IN_PORT. The user has to provide the IC's setting to the DMFC by programming the DMFC_IC_FRAME_WIDTH_RD, DMFC_IC_FRAME_HEIGHT_RD and DMFC_IC_FRAME_PPW_C fields. The burst size of the channel coming from the IC should always be programmed to 4 words.

37.4.8.3 DC write channel and AHB accesses

The second physical memory of the DMFC serves the

- IDMAC write channel (read from display)
- 2 AHB channels that can be read or write

The IDMAC write channel is programmed using the DMFC_RD_CHAN register in a similar way to the DC and DP read channels described above. The user has to provide the frame width and height for this channel and the pixel per word parameter.

The AHB channels are used from accesses via the AHB port to the display. The accesses are distributed between channels according to the MCU_T parameter.

37.4.9 DP - Display Processor

The display processor processes the image prior to sending it to the display. The main task performed by the DP is combining between 2 planes.

The DP has 2 input FIFOs holding the data of full plane and the partial plane. In addition the DP performs some image enhancement functions like gamma correction, Color space conversion including Gamut mapping.

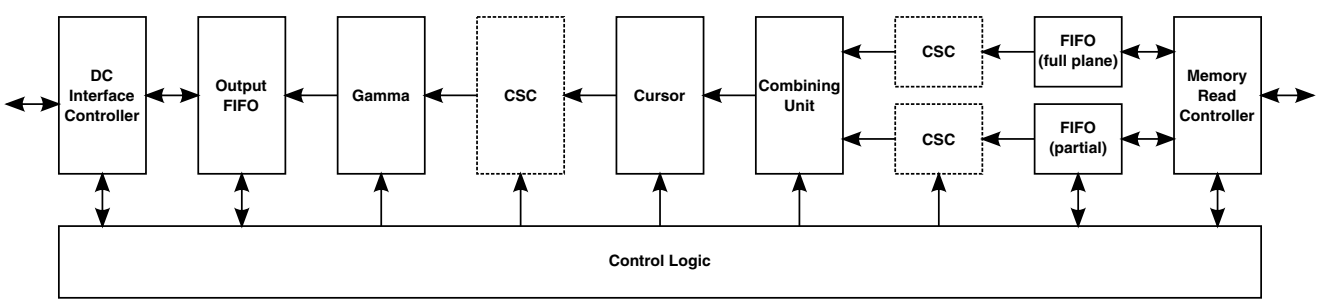


Figure 37-36. DP Micro architecture diagram

37.4.9.1 The DP programming model

The DP supports 3 flows. One sync flow and 2 Async flows. The DP holds 3 sets of registers. one set for each flow. Hence when referring to a register in this section the information is applicable to all the 3 sets. for example when referring to DP_COM_CONF, the information is applicable to DP_COM_CONF_SYNC, DP_COM_CONF_ASYNC0 and DP_COM_CONF_ASYNC1.

37.4.9.2 Displayed Planes

The following figure shows the planes displayed on a display.

There are full and partial planes. The partial plane's position is defined relatively to the upper left corner of the full plane (FGXP and FGYP parameters on the corresponding IPU_DP_FG_POS register). The size of the partial and full planes is defined on the corresponding IDMAC's channels' FW and FH parameters. The cursor position and parameters are set in the DP_CUR_POS register.

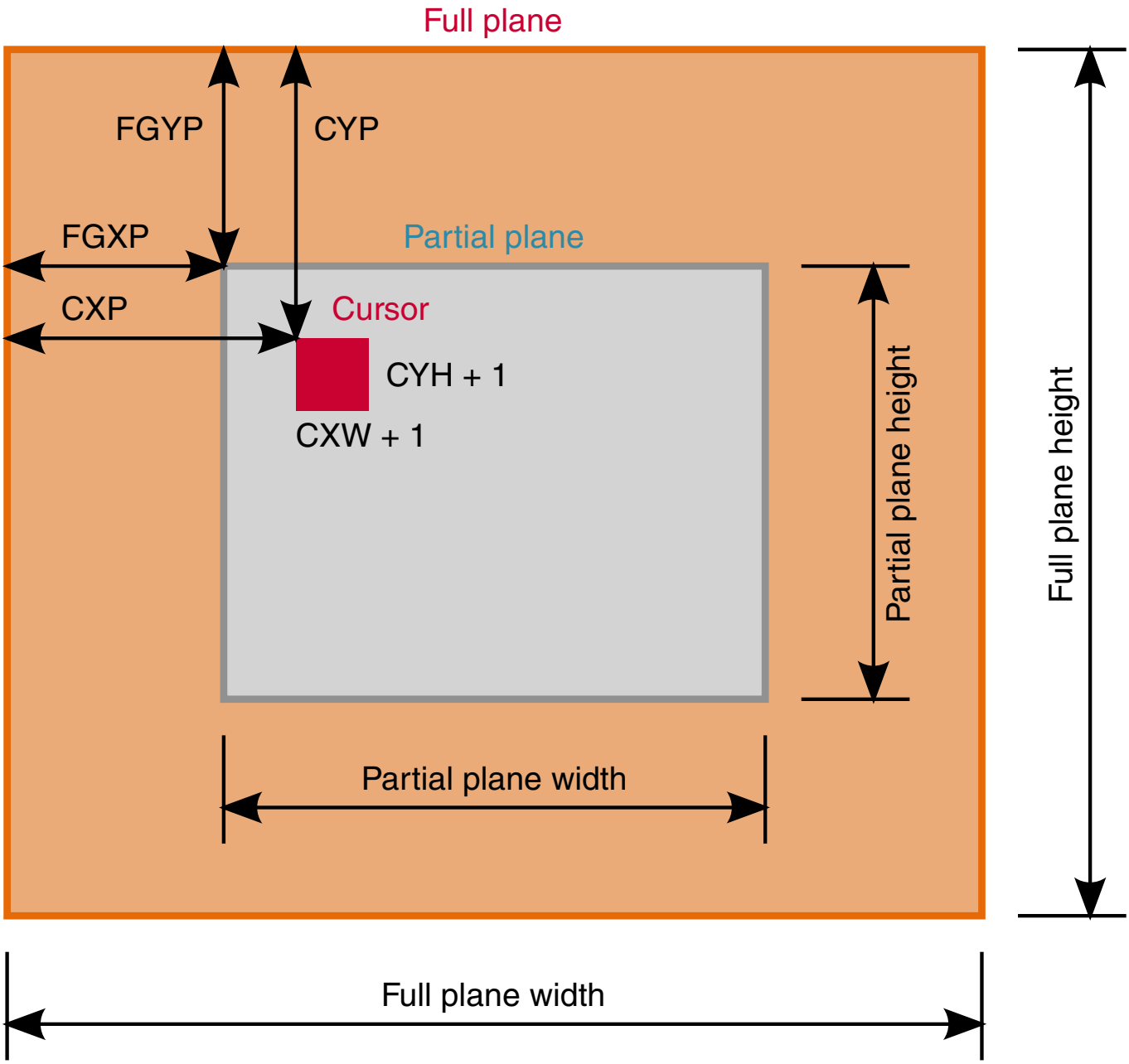


Figure 37-37. Displayed Planes

37.4.9.3 Combining Unit

The Combining Unit performs combining between the full and the partial planes. Each one of the planes may be graphics or video plane.

There are the following combining options:

- local alpha blending,
- global alpha blending,
- use of key color.
- order of the planes (full is presented over the partial plane and vice versa)

Combining mode is selected via the DP_COM_CONF Register. The combining equation is:

OP = BG*(1 - a) + FG*a

Where BG and FG are 2 input pixels; The DP_GWSEL bit defines if the BG is the pixel coming from the full plane or the partial plane.

a= (A+floor(A/128))/256 - an alpha value

A - a global or local transparency parameter.

The global A is written in the DP_GWAV field, the local A arrives together with the pixel.

A pixel becomes transparent when color keying is enabled and a pixel color matches a key color (independently on an alpha parameter). The color keying is defined on:DP_GWCKR, DP_GWCKG, DP_GWCKB

Combining takes 1 cycle per pixel. The Combining Unit outputs 24-bit words in the RGB/YUV format.

37.4.9.4 Cursor Generator

The Combining Unit output is passes through the Cursor Generator. The cursor's size and position are set via the DP_CUR_POS Register, a cursor color - via the DP_CUR_MAP Register. Different logic functions of combining the cursor with the image are supported as defined by the DP_COC field.

The cursor can be blinking. The blinking mechanism resides on the display controller sub-block. The blinking parameters are defined on the DC_BK_EN and DC_BKDVIV fields.

37.4.9.5 Color Space Conversion unit - CSC

The DP can get 2 input pixels from 2 different color spaces (YUV or RGB) and convert one of them to a common color space (YUV or RGB). In addition the 2 inputs can be of the same color space where the result is converted to another color space (YUV or RGB).

The DP has a single CSC unit that can be placed on one of 3 locations:

- At the output of one of the 2 input FIFOs
- At the output of the cursor generator.

Placing is done according to the DP_CSC_DEF field.

The color conversion implements a 3x3 matrix multiplication between the full RGB pixels and the color conversion constants, in order to obtain a YCC format image.

The conversion formula is:

x -> Clip(Round(S* 2^E)), S=Ax+B

where

A is a 3x3-dimensional matrix of weights, each a 10-bit signed number with 8 fractional digits

$$A = \begin{bmatrix} CSC\_A0 & CSC\_A1 & CSC\_A2 \\ CSC\_A3 & CSC\_A4 & CSC\_A5 \\ CSC\_A6 & CSC\_A7 & CSC\_A8 \end{bmatrix}$$

B is a 3-dimensional vector of offsets, each a 14-bit signed number with 2 fractional digits

$$B = \begin{bmatrix} CSC\_B0 & CSC\_B1 & CSC\_B2 \end{bmatrix}$$

S is a 3 dimensional vector of sums, each a 16-bit signed number with 4 fractional digits

S=Ax+B

E is an exponent, assuming one of the following values: -1,0,1,2 (allowing weights up to 8). The CSC_S parameters are encoded by 2 bits, please refer to the CSC_S parameter description.

$$E = [CSC\_S0\ CSC\_S1\ CSC\_S2]$$

A more explicit formula:

S[i] = (sum(A[i][j]*In[j]) >> 4) + (B[i] << 2) + (1 << (3-E[i]))

Out[i] = Clip(S[i] >> 4-E[i])

Where Clip() performs clipping to the range 0..255 (either per-component clipping or more sophisticated clipping that preserves the hue of the pixel)

37.4.9.5.1 Gamut mapping

When the color transformation produces colors outside the allowed range, they must be mapped back. This is called gamut mapping. The DP supports 2 clipping algorithms. (controlled by GAMUT_SAT_EN bit)

Hue preserving clipping algorithm is suitable only for RGB components. For YUV components, the per-component clipping algorithm is used.

Per component Clipping

This mapping is performed by clipping each of the components independently to its allowed range of values (the final value being uint8):

•Y: to 0..255 or 16..235 according to the SAT_MODE bit

•U/V: to 0.255 or 16..240 according to the SAT_MODE bit

Hue Preserving Clipping

Hue Preserving clipping is done in the following way

•First stage - eliminating negative values

```
N = min(R,G,B)
if (N<0) X-> X-N, where X=R,G,B
```

•At this stage, all components are non-negative and the MSB's beyond d=11 bits are ignored (assumed 0).

•Second stage - eliminating large values

```
M = max(R,G,B) (d-bit integer)
if (M>255)
```

```
M' = M >> (m-7), where m=8..d-1 is the index of the most-significant non-zero
bit in M
D' = Ceil(256*255/M') = 256..510 (since M' = 128..255)
Ceil(x) = the smallest integer which is not smaller than x
Implemented by a hard-wired 128x9-bit LUT
X -> min(255, (X*D')>>(m+1)), where X=R,G,B (8x9 multiplier)
else
X -> X
```

37.4.9.6 Gamma correction

The DP includes a gamma correction function. It is approximated by the piece-wise polynomial:

gammar = GAMMA_C<i> + ((R[4:0] * GAMMA_S<i>) >> 4 + 1)>>1

Where R is the input red component, that's a 9 bit input composed pixel_in*2+pixel_in[7].

Single approximation slope is used for Gamma Correction of red, green and blue color components. However the Gamma Correction block instantiated three times since processing for color components should be done in parallel. The gamma transform is also available for changing the contrast of the luminance component only.

The required Gamma correction slope for a specific display should by provided by the display manufacture. This information can be provided in various forms, as graph or formula. The gamma correction input pixel level (Gin) should be normalized to a maximum of 383. The gamma correction output pixel level (Gout) should be normalized to a maximum of 255. Then a following data should be collected:

Table 37-28. Gamma correction values

Gin	Gout
0	Gout0
2	Gout1
4	Gout2
8	Gout3
16	Gout4
32	Gout5
64	Gout6
96	Gout7
128	Gout8
160	Gout9
192	Gout10
224	Gout11
256	Gout12

Table continues on the next page...

Table 37-28. Gamma correction values (continued)

Gin	Gout
288	Gout13
320	Gout14
352	Gout15

Based on the table above the values of DP_GAMMA_S_SYNC<i> and DP_GAMMA_C_SYNC<i> fields for gamma correction control registers are calculated as following:

Table 37-29. Gamma correction values

i	DP_GAMMA_C_SYNC<i>	DP_GAMMA_S_SYNC<i>
0	Gout0	16*(Gout1-Gout0)
1	2*Gout1-Gout2	16*(Gout2-Gout1)
2	2*Gout2-Gout3	8*(Gout3-Gout2)
3	2*Gout3-Gout4	4*(Gout4-Gout3)
4	2*Gout4-Gout5	2*(Gout5-Gout4)
5	Gout5	Gout6-Gout5
6	Gout6	Gout7-Gout6
7	Gout7	Gout8-Gout7
8	Gout8	Gout9-Gout8
9	Gout9	Gout10-Gout9
10	Gout10	Gout11-Gout10
11	Gout11	Gout12-Gout11
12	Gout12	Gout13-Gout12
13	Gout13	Gout14-Gout13
14	Gout14	Gout15-Gout14
15	Gout15	255-Gout15

37.4.9.7 DC interface

The DC interface unit performs 2 tasks.

- Starts the flow via the DP by getting a request from the DC and once the DP is ready send the new frame request to the IDMAC.
- Control the DP's output FIFO and send the data to the DC using an handshake mechanism.

37.4.9.8 DP's flows management

The DP can manage up to 3 flows: one synchronous flow and 2 asynchronous flows. However the DP can handle only one flow simultaneously. The DP has an automatic mechanism to control and switch between flows.

The DP's highest priority flow is the sync flow. An async flow can be executed during the blanking interval of the sync flow. An async flow can be stopped in order to serve the sync flow. The sync flow cannot be broken. The DP holds 3 sets of registers, one set for each flow. The registers are located in the SRM memory. According to the flow that needs to be executed the correct set of registers is loaded to the DP.

The figure below illustrates the flow management done by the DP.

A flow starts when a request from the DC arrives and the internal pipe is empty. For sync flows, the full frame NF (new frame) indication arrives immediately. In case that a partial frame is also used the NF indication will be sent one row before the row that the partial plane is actually positioned.

In case of ASYNC flow the DP first check if the previous async flow was broken. If yes, the DP restores the last settings of the previous flow. If this is a new async flow the DP will first reload the flow's parameters from the SRM. In case that a partial frame is also used the NF indication will be sent one row before the row that the partial plane is actually positioned.

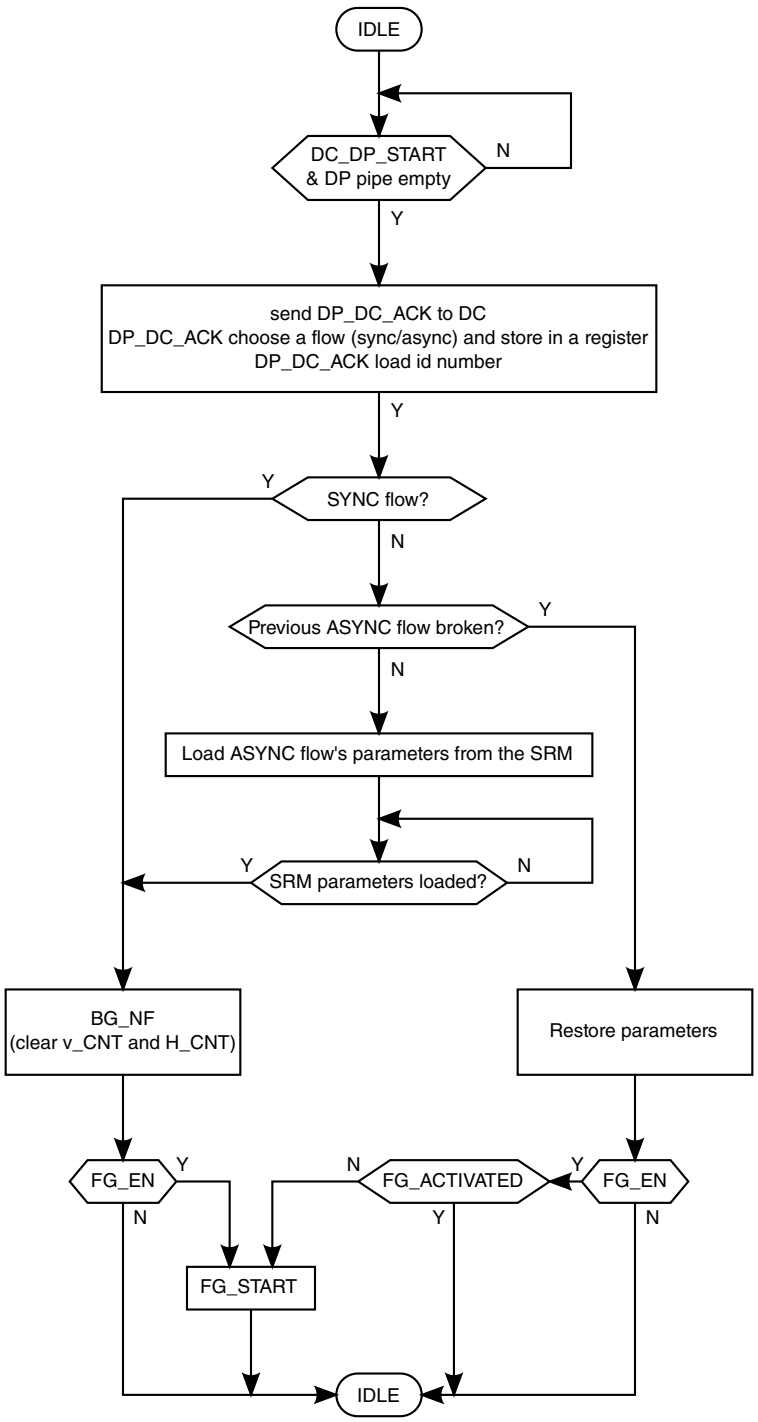


Figure 37-38. DP flow management chart

37.4.9.9 DP debug unit

The DP supports synchronous and asynchronous flows using the same hardware. The asynchronous flow can be broken by the synchronous flow. The DP's debug unit provides the ability to know on which row exactly the async flow has been broken. This is done by providing an interrupt (with DP_DEBUG_CNT register) and providing row status flags (on DP_DEBUG_STAT register).

As the async flow can broken multiple times within a specific frame The user can control the breaking point the issues the debug event by programming BRAKE_CNT field.

37.4.9.10 Restriction

When both full and partial planes are processed, the full plane's minimal frame width is 13 pixels.

The Minimum frame height supported by the DP is 2 lines.

37.4.10 Display Interface (DI)

The DI provides arbitrates access to up to three displays with time multiplexing. It converts a data from the DC or the ARM platform (low level access for serial interface only) to a format suitable for the specific display interface.

The DI generates display clocks and other display control signals with programmable timings. The DI outputs data to or inputs from parallel and/or serial interfaces.

This block generates all the control signals sent to the display. The DC sends to the DI; the data for the display and a set of control signals. The controls coming from the DC are used in order to generate the control signals sent to the display. One exception is serial low level access (LLA), where the DC is bypassed and the data is coming directly from the ARM platform. The figure below is the DI's block diagram.

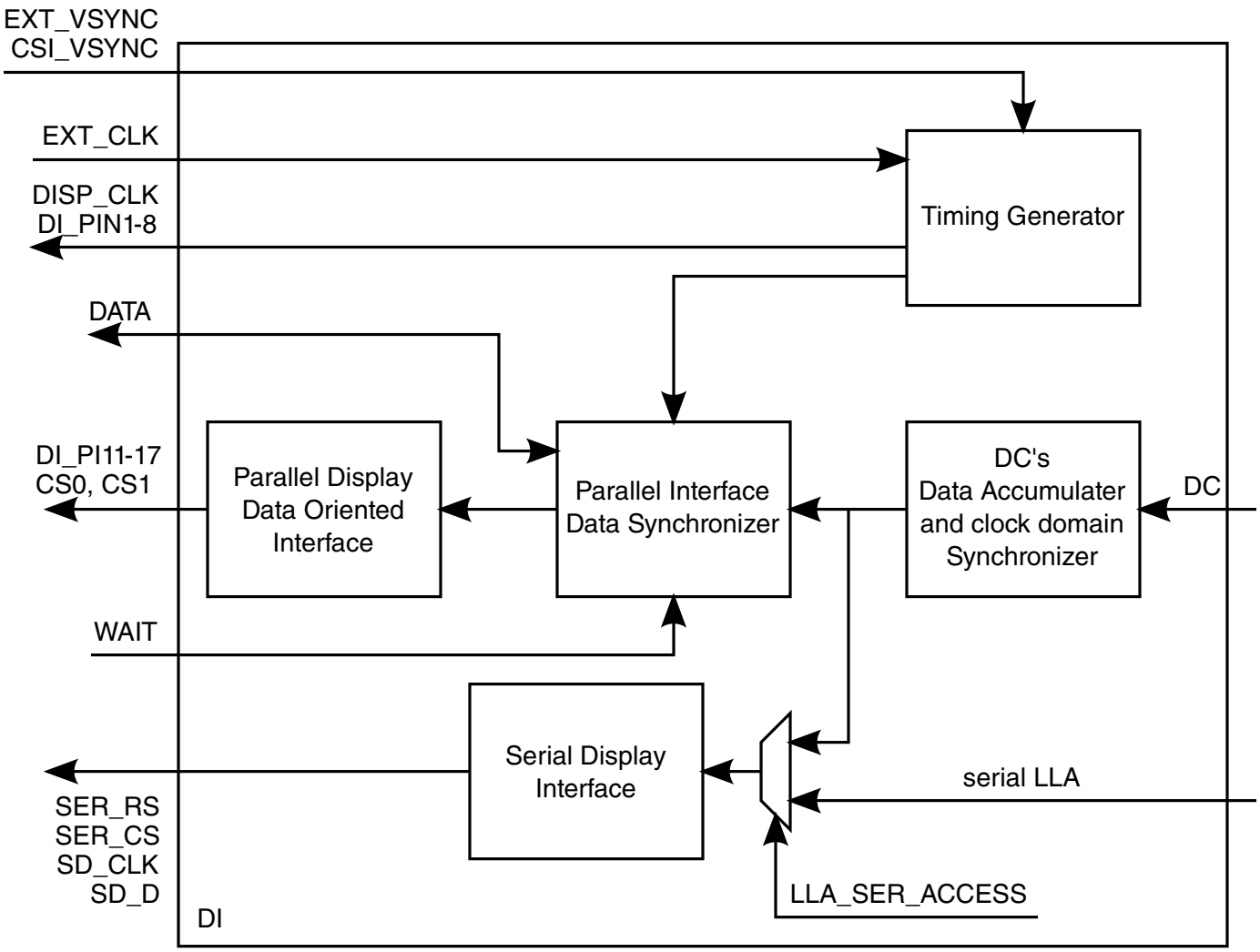


Figure 37-39. DI's block diagram

NOTE

The Serial Display Interface is not supported on this product.

The display interface includes 2 groups of control signals:

- Time oriented signals - These type of signals are generated according to the DI's internal timers. These are free running signals that change their state according to a pre-defined waveform. For example VSYNC, HSYNC, display's clock (pixel clock) etc.
- Data oriented signals - The DC may add markers to data sent to the DI. These markers are used to indicate a specific attribute of the data (for example: end-of-line, end-of-frame, chip-select etc.). The marker coming from the DC triggers a specific waveform of one or more signals on the display's interface. The specific waveform will be seen on the bus along with the associated data. The markers may be synced to

a time oriented signal. For example: attach the end-of-frame signal to the next VSYNC.

37.4.10.1 DC interface, data accumulator and clock domain synchronizer

The data accumulator is the DI's input buffer. It receives the data from the DC along with a set of control signals. The data accumulator receives the data from the DC's clock domain and synchronize it to the DI's clock domain.

37.4.10.2 Parallel interface data synchronizer and data oriented interface

The data accumulated in the DI's input buffer will be sent to the display according to the DI's internal events. Each event is generated by a counter. A tag is attached to each data by the DC's microcode using the SYNC field in the DC's microcode. The tag selects the event that the data will be synced to.

Once the corresponding event is generated, the data will be sent to the display. A data can be a pixel or a component (part of a pixel).The data synchronization occurs separately for each component. For asynchronous displays the tag will be equal to 0 i.e. the data is not synchronized to any event. The data will be sent out of the buffer immediately.

37.4.10.3 Timing generator

The timing generator is used for generating the waveforms' of each pin of the display's interface.

The timing generator is built of 10 counters. One counter functions as a time base This timer is called the BASE TIMER (BS). The other 9 counters are used in order to generate the control signals' waveforms. The last counter (counter #9) is special see Counter number 9.

The DI clock can be derived from IPU's clock (HSP_CLK) or from an external source (via the DIn_DISP_CLK - ipp_di_#_ext_clk pin). The clock's source is statically selected by configuring the DI#_CLK_EXT bit.

Figure 37-41 illustrates the main parameters of a waveform. A waveform's segment is built of 5 parameters. Each segment can be has 2 phase "ACTIVE PHASE" and "OFFSET PHASE".

- TIMEBASE - this is the base timer (fundamental timebase); all the other parameters are derived from this timer. The timebase is generated by counting DI clock cycles. The amount of cycles is defined according to DI#_DISP_CLK_PERIOD. This field defines the Display interface clock period, This parameter contains an integer part (bits 11:4) and a fractional part (bits 3:0). It defines a fractional division ratio of the Di's source clock for generation of the display's interface clock. The timebase is generated from edge aligned pulses of the DI clock. The user can delay the timebase starting point by defining an offset to the timebase. This is done by configuring the DI#_DISP_CLK_OFFSET field. The offset is calculated from the point where the DI is enabled to the point where the timebase starts ticking. The display clock waveform is generated between 2 edges of the timebase. The waveform is defined according to the DI#_DISP_CLK_UP and DI#_DISP_CLK_DOWN. Each pin has a specific timebase that is derived from the fundamental timebase. The following figure illustrates the relations between the TIMEBASE and the DI's clock

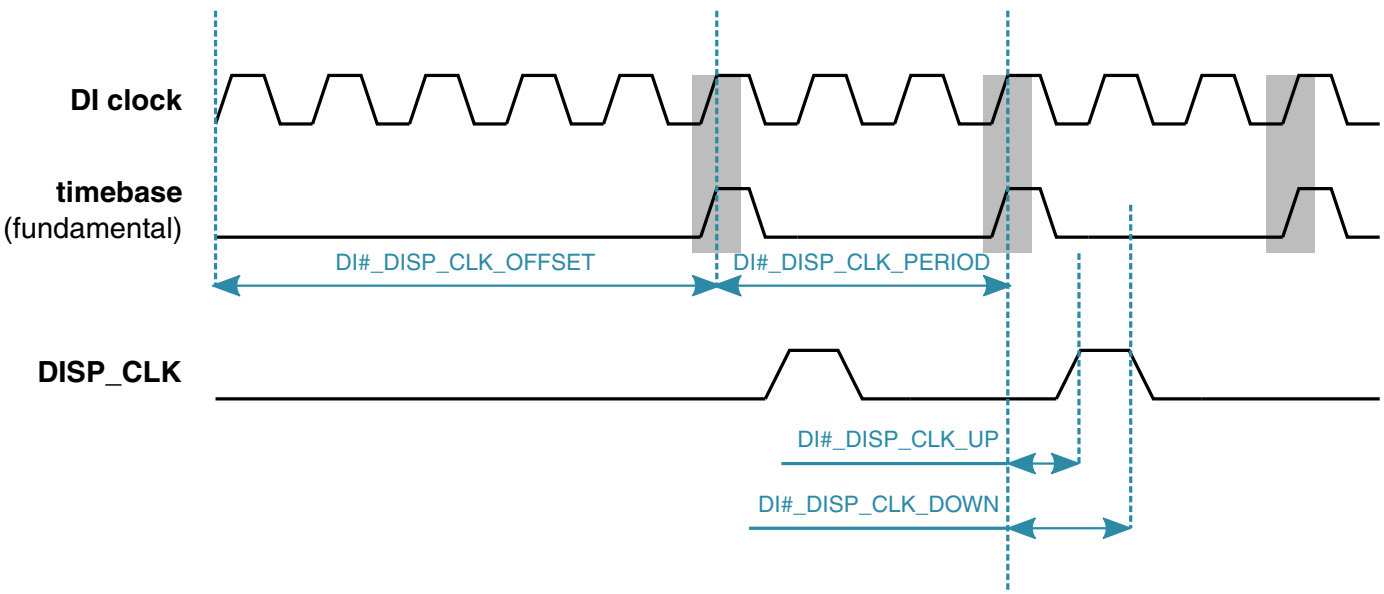


Figure 37-40. Timebase, DI's clock and display's clock relations

- OFFSET - this parameter defines when the length of the "OFFSET PHASE" it is defined by di#_offset_value_<N> field, where N is the counter's index.
- STEP - This parameter defines the length of the "ACTIVE PHASE"; it is defined by the di#_step_repeat_<N> field, where N is the counter's index. If the counter is in auto reload mode (di#_cnt_auto_reload_<N> bit is set) then the counter will be automatically reloaded forever. The value of di#_step_repeat_<N> is ignored in that case.

- RUN - The "ACTIVE PHASE" is partitioned to several "RUN sections"; this parameter defines the length of the "RUN section"; it is defined by the di#_run_value_m1_<N> field, where N is the counter's index.
- UP - Each "RUN section" contains the waveform of a single pulse. This parameter defines the offset from the beginning of the "RUN section" to the assertion of the signal; it is defined by the di#_cnt_up_<N> field, where N is the counter's index.
- DOWN - This parameter defines the offset from the beginning of the "RUN section" to the negation of the signal; it is defined by the di#_cnt_down_<N> field, where N is the counter's index. In case where DOWN < UP the waveform will have a 50% duty cycle.

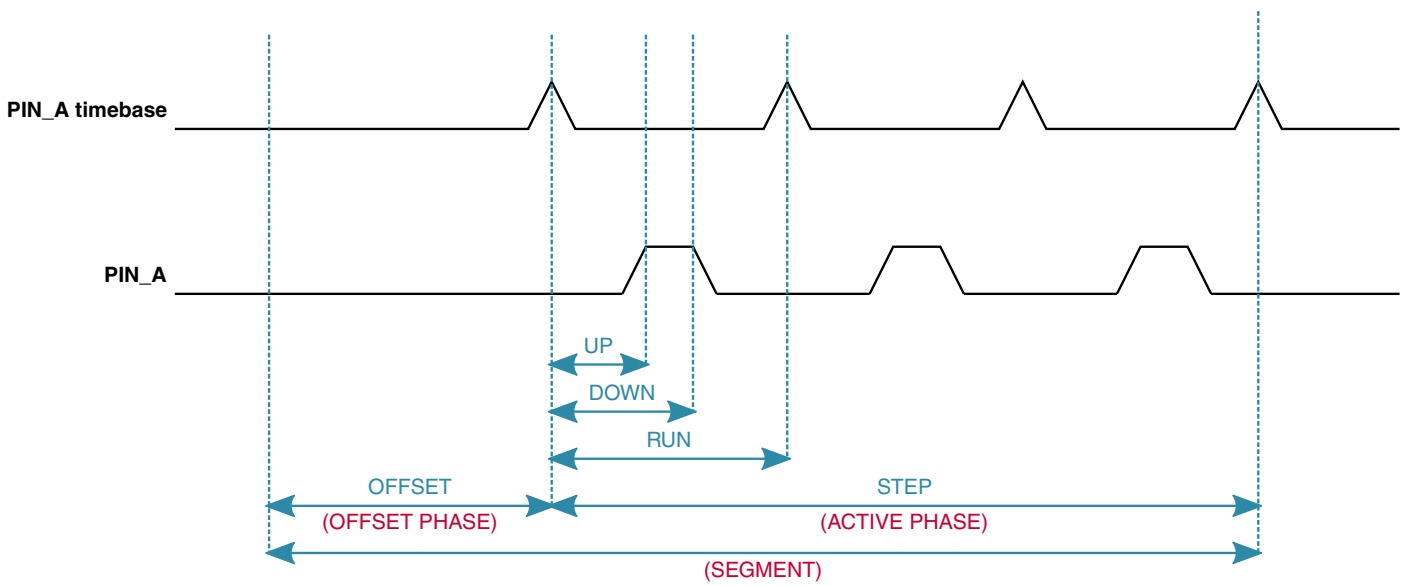


Figure 37-41. DI waveform's main parameters

37.4.10.3.1 Waveform concatenation

The DI provides the ability to derive the waveform from the fundamental timebase or from another PIN. In that case, one pin's waveform is used as another pin's timebase.

The following figure provides an example. PIN_A and PIN_C are derived from the fundamental timebase. However, PIN_B is derived from PIN_A's wavefore. The trigger is selected by DI#_RUN_RESOLUTION_<N>, where <N> is the index of the counter. A counter can be triggered by a counter with lowered index. For example: counter #5 can be triggered by counter #3 but can't be triggered by counter #7.

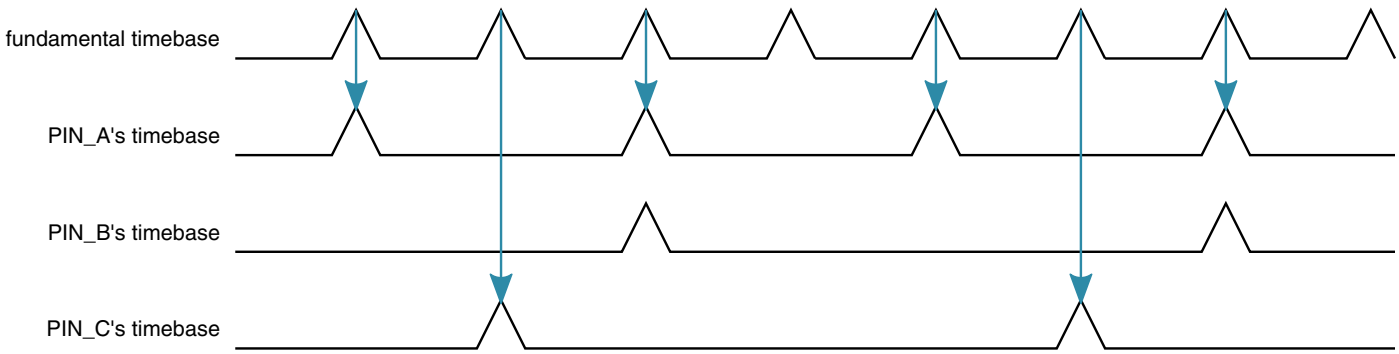


Figure 37-42. DI pins - Waveform's time bases concatenation

37.4.10.3.2 The basic counter

The DI has 9 counters. A counter is built of 3 units: timebase generator, waveform generator and polarity generator.

Figure 37-43 illustrates the counter's structure.

The timebase generator

The timebase generator gets 3 triggers. Clear, offset and run trigger. The Clear is the trigger that resets the counter. It is selected by programming the DI#_CNT_CLR_SEL_<N>.

The Offset trigger is the trigger used for counting the OFFSET_PHASE. The user can select the source of the trigger by programming the DI#_OFFSET_RESOLUTION_<N>. The offset's value is defined by programming the DI#_OFFSET_VALUE_<N>

The RUN trigger is the trigger used for counting the RUN period. The user can select the source of the trigger by programming the DI#_RUN_RESOLUTION_<N>. The RUN's value is defined by programming the DI#_RUN_VALUE_<N>

The timebase generator counts according to the DI_CLK or according to another counter's output. In order to use a source different than DI_CLK, the user should set the POLARITY_GEN_EN bits to 01.

Waveform generator

This unit generates the waveform. It gets the values of RUN, UP, DOWN and STEP and build the waveform accordingly. The waveform is counted according to the signal generated by the timebase generator.

Polarity generator

The waveform's polarity is controlled by 2 units. The static polarity is changed according to the POLARITY_<#> bit of each pin. This bit defines if the waveform of the pin is active high or active low.

The other unit controlling the polarity is the polarity generator. This unit is enabled by setting the POLARITY_GEN_EN[1] to 1. The polarity generator has 2 modes: "toggle mode" and "normal polarity mode". The mode is defined according to POLARITY_GEN_EN[0].

- Normal polarity mode - In this mode the polarity is changed according to 2 other counters. The counter selected by POLARITY_TRIGGER_SEL define the sampling point. The counter selected by POLARITY_CLR_SEL defines the polarity value. At the sampling point the polarity value is defined. The current polarity value defines the current polarity of the waveform.
- Toggle mode - In this mode, the output of the timebase generator's output causes the polarity to toggle. Any tick of the timebase generator inverts the polarity. When this mode is enabled the ticks generated by the counter selected by POLARITY_TRIGGER_SEL initialize the polarity generator and the timebase generator causes the polarity to toggle.

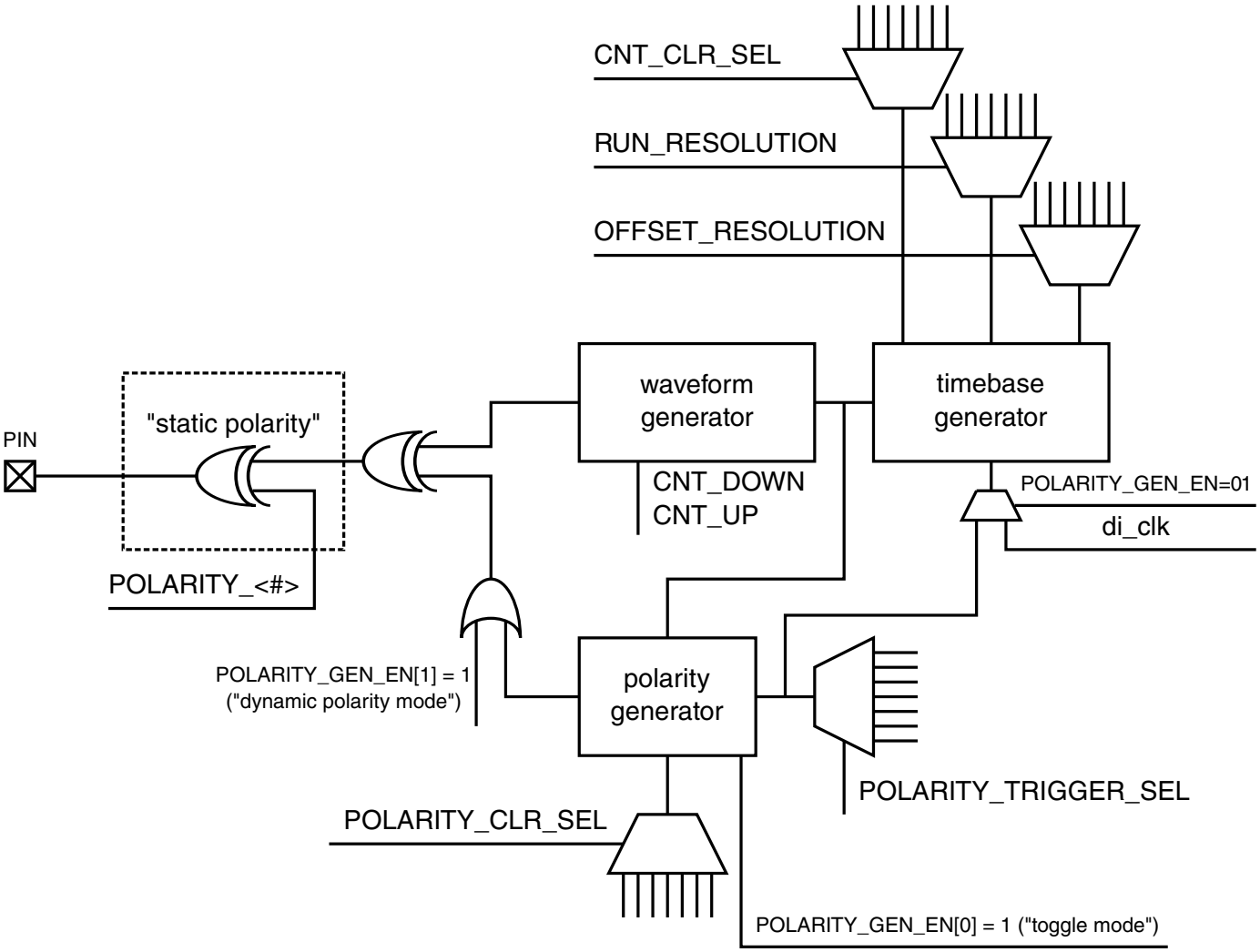


Figure 37-43. DI's counter's structure

The following figure provides an example for waveform generation using different trigger sources. The waveform is generated for PIN_D. PIN_D counter is cleared by the PIN_C's timebase. The offset period is calculated by counting cycles of PIN_B's timebase. The RUN period is calculated by counting cycles of PIN_A's timebase. The UP and DOWN periods of the waveform are calculated by counting cycles of PIN_D's timebase.

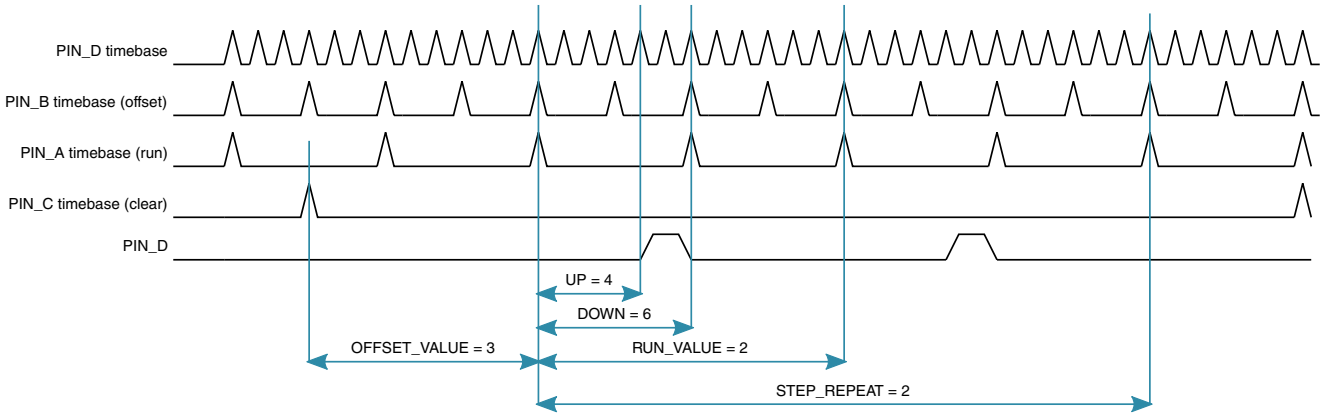


Figure 37-44. Clear, offset and run triggers and values - Example

37.4.10.3.3 Counter number 9

The last counter (counter #9) is an auxiliary counter and it is not used for generating a waveform for a specific pin. It can be used in order to attach another waveform to an existing one.

The user define the waveform for a specific pin (main pin). The user defines the auxiliary waveform using counter #9. The 2 waveforms are logically ORed. The combined waveform will be routed to the main pin. The main waveform that this counter is attached to is defined according to DI#_GENTIME_SEL_9.

The tag of counter #9 can be generated from counter #9 or from the main waveform. This is selected according to the DI#_TAG_SEL_9.

37.4.10.3.4 DI's active window

The DI provides an alternative way to define the synchronous display setting by using an active window and thus, needs to program less counters. The synchronous display's active window is a rectangle on the display where IPU sends data. It is set by programming the parameters defined on DI#_AW0 and DI#_AW1 registers.

The following figure illustrates the different parameters defining the active window. The display's vertical and horizontal position are defined according to counters. The DI#_AW_HCOUNT_SEL selects the counter which the horizontal position is calculated according to. DI#_AW_VCOUNT_SEL selects the counter which the vertical position is calculated according to. The Active data is sent according to a trigger. DI#_AW_TRIG_SEL selects the counter which calculates this trigger. This trigger usually functions as a data enable signal.

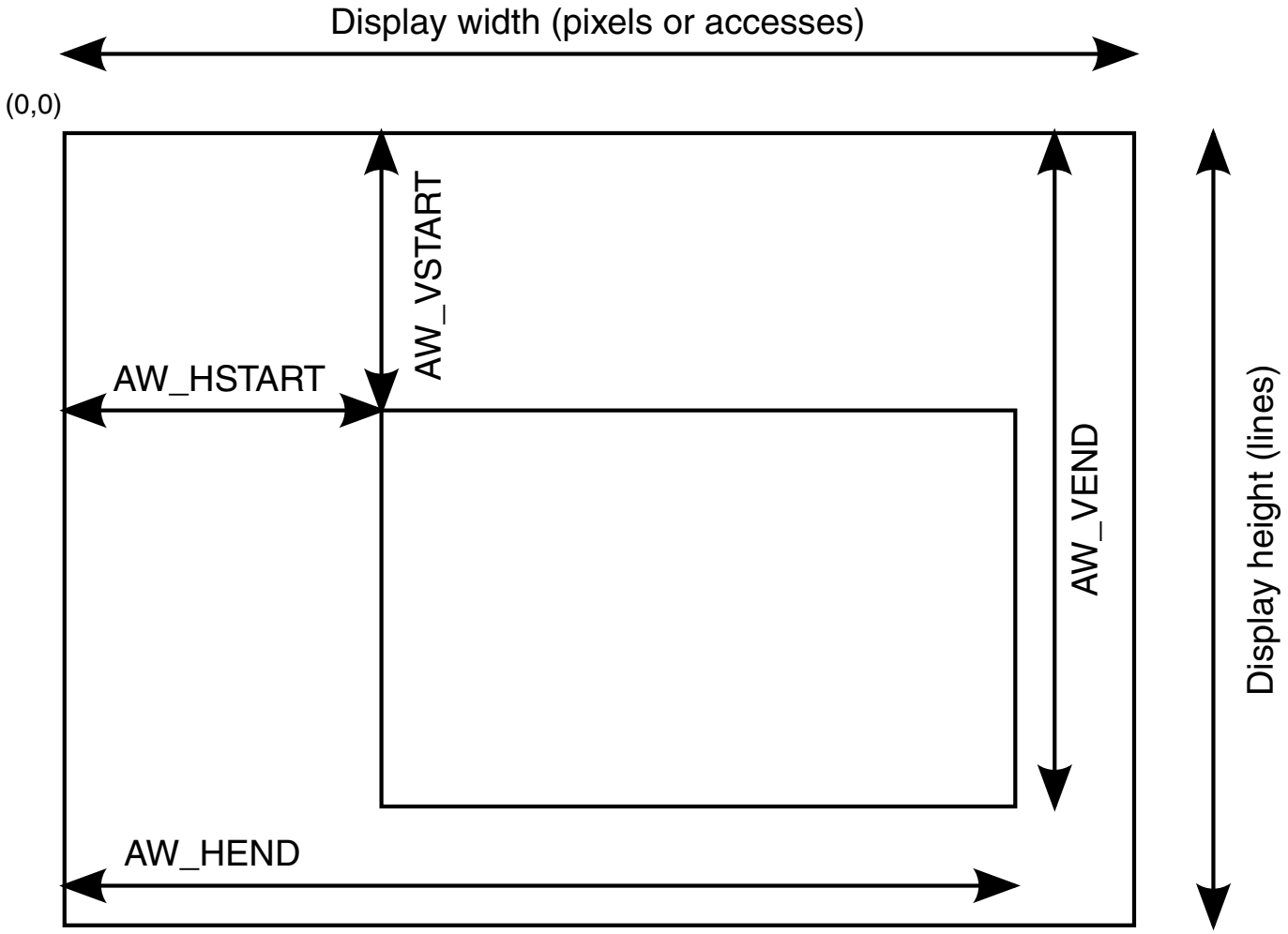


Figure 37-45. DI's Active Window

37.4.10.4 Waveform settings for asynchronous interface pins

The DI provides 8 signals that are used for asynchronous interface. These signals are PIN11 through PIN17 (ipp_di_#_pin_11 through ipp_di_0_pin_17) and the CS (ipp_di_0_do_dispb_d0_cs).

The DI holds 12 wave set quartets. Each quartet includes 4 registers (DI#_DW_SET<j>_<i>). Each DW_SET register holds the UP and DOWN values of the waveform. The DW_SET register is selected in the following way. The WAVEFORM field in the DC template is a pointer that points to one of the 12 quartets. In addition the WAVEFORM field points to one of 12 DI#_DW_GEN_<i> registers. The DI#_DW_GEN_<i> holds 9 pointers. The pointers are 2 bits field that points to one of the registers from the DI#_DW_SET<j>_<i> quartet. Each one of the 8 pointers in the

DI#_DW_GEN_<i> registers is related to a specific pin of the DI's asynchronous interface. The following figure illustrates the relations between the registers controlling the asynchronous interface's signals.

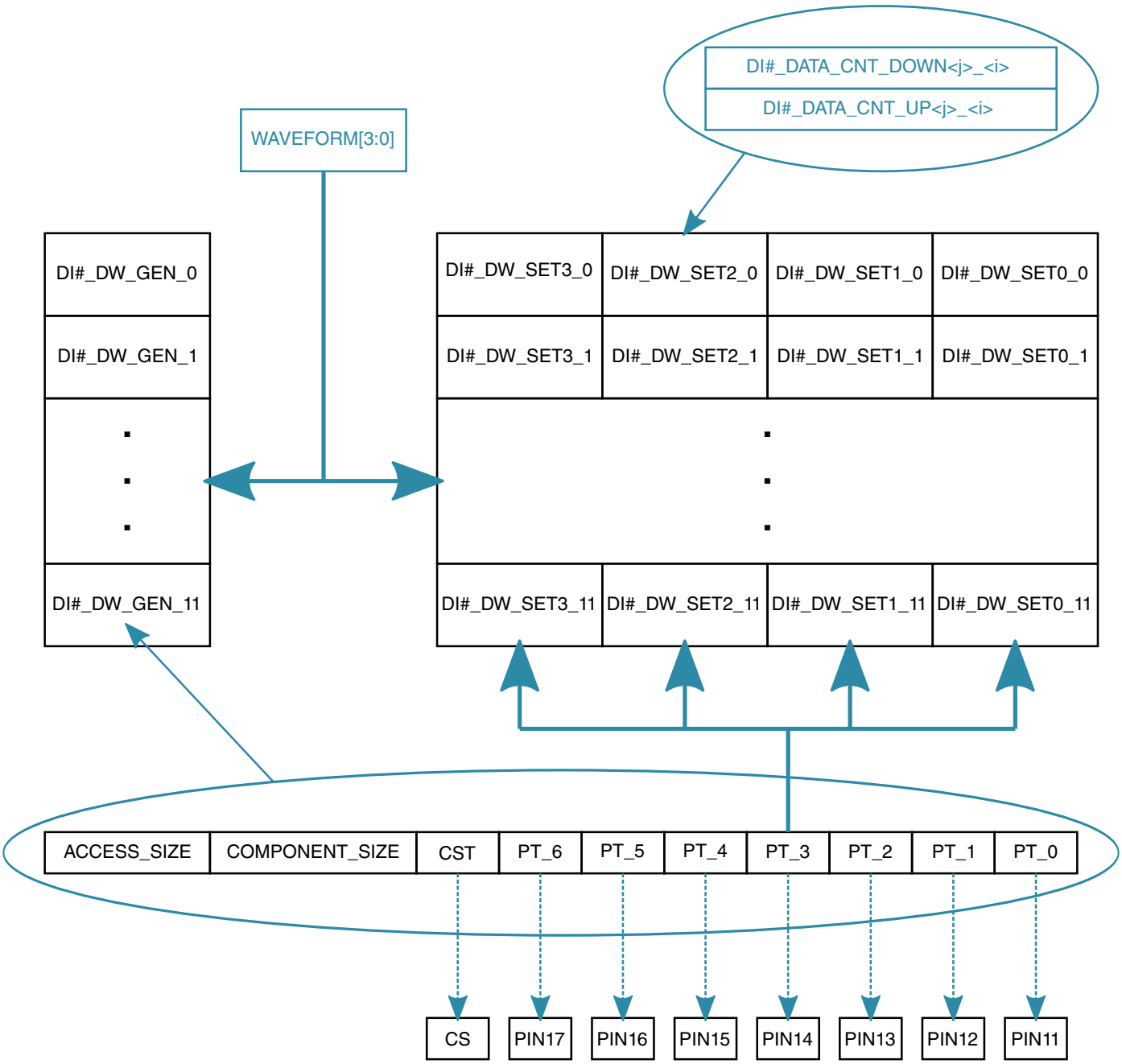


Figure 37-46. Waveform settings for asynchronous interface pins - parallel interface

The DI#_DW_GEN_<i> register includes the data's waveform settings as well. ACCESS_SIZE defines the amount of DI clock cycles that a pixel is valid on the bus. When generic data is sent, this field defines the amount of cycles that the generic data is

valid on the bus. A pixel may be broken into few components. The COMPONNENT_SIZE field defines the amount of cycles that each component is valid on the bus.

The COMPONNENT_SIZE is always smaller or equal to ACCESS_SIZE. For synchronous interface COMPONNENT_SIZE is always equal to ACCESS_SIZE. In case that there's a need for some gap between one type of accesses to another having the COMPONNENT_SIZE smaller than ACCESS_SIZE can be useful (for example read after write accesses that require some gap between them).

37.4.10.5 Low Level Access - LLA

LLA is a ARM platform direct access to the display. For parallel displays the DI's behavior for LLA is the same as any other access to a parallel display. For serial displays the DI allows a DI arbitration bypass. This is done if the DI#_LLA_SER_ACCESS bit is set.

In that case there is a direct access from the DMFC to the DI which bypasses the DC allowing simultaneous access to both serial and parallel interfaces. When DI#_LLA_SER_ACCESS bit is set, the user must not do any other kind of accesses to the serial interface except LLA. When DI#_LLA_SER_ACCESS bit is set the corresponding DI#_WAIT4SERIAL must be clear.

37.4.10.6 Using a mask channel

The IPU is able to provide the windowing function on displays that have data enable control. This is achieved by masking of some screen regions according to a 1-bit/pixel mask read from the memory via IDMAC through channel #44.

When the mask value is zero, the pixel is not displayed. This feature can be used for dual-port smart displays.

37.4.11 Video De Interlacing or Combining Block (VDIC)

The Video De-Interlacer as well as the Combiner have two operation modes:

- De-interlacing: converts an interlaced video stream to progressive order.
- Combining: combines two video/graphics planes and a background color.

The Video De-interlace block (VDIC) deinterlaces standard interlaced video to produce progressive video, that is used for upsizing to HD formats or for display on progressive displays. For VDIC operation three fields are necessary F(n-1),F(n),F(n+1). The F(n-1) field arrived through CSI interface in real time mode or through channel 1 and then stored in FIFO1. The F(n) arrived through channel 2. At least three lines of F(n) are stored in Line Store memory. The F(n+1) arrived through channel 3 and stored in FIFO3. FIFO controllers read data from FIFOs and then data aligned in pixels buffers. From the buffers the data arrived to Line Padding Controller (LPC). The LPC padding missing line at the beginning and end of the frame. The DeInterlacing sub-block (DI) perform the processing. Then data send to IC sub-block for processing or for transferring to external memory.

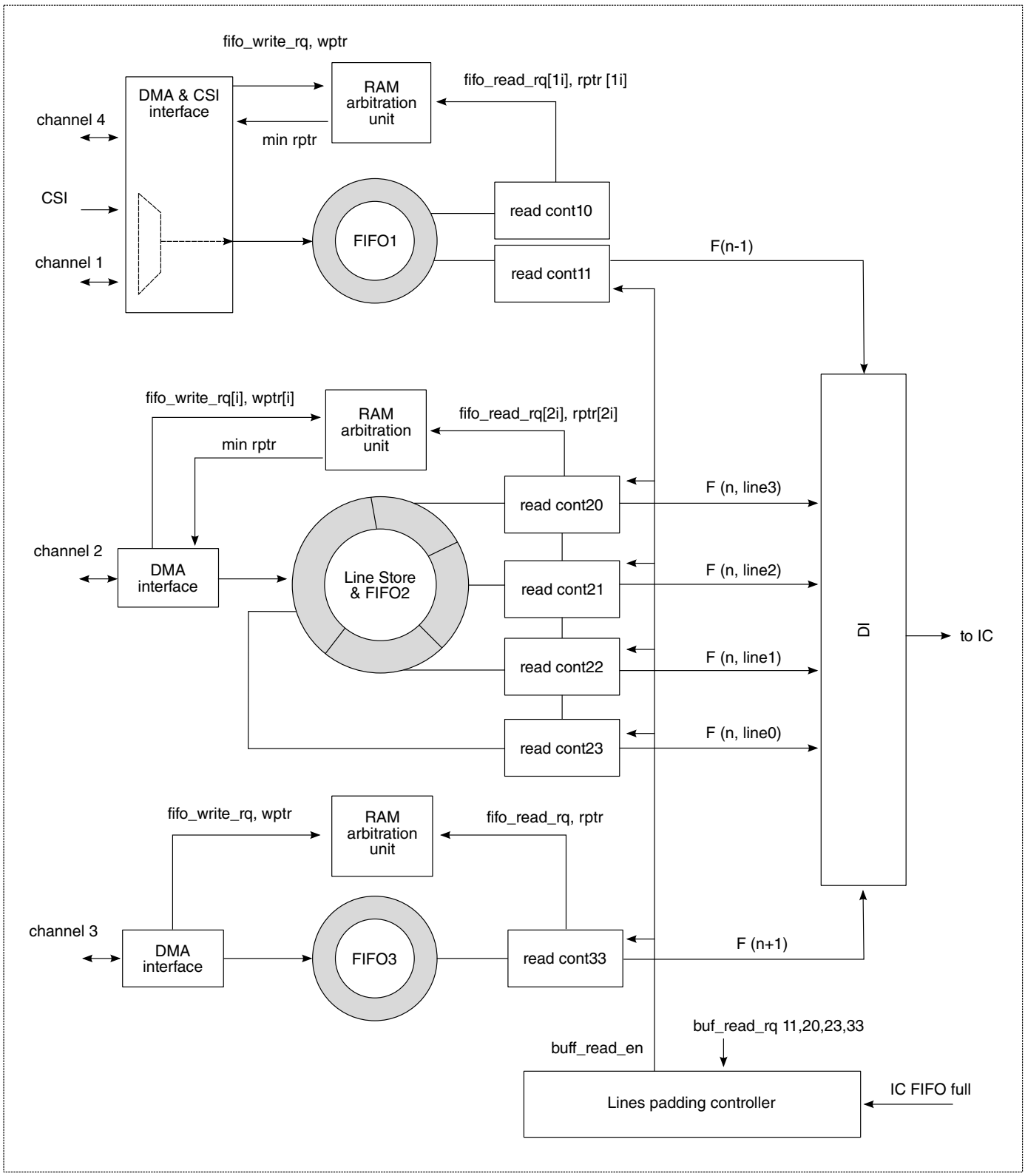


Figure 37-47. VDIC Block Diagram

37.4.11.1 VDIC Features

Key features of the VDIC block include:

- Deintarlacing
  - maximum horizontal resolution 968 pixels
  - maximum pixel rate 75100MP/s
  - Support YUV422 and YUV420 formats
- CSI FIFO mode
- Combining

37.4.11.2 De interlacer (DI) sub-block

The block diagram of the DI block is shown in figure below.

Pipelining is inserted at all stages, so that the design may run at a fast clock speed, if needed.

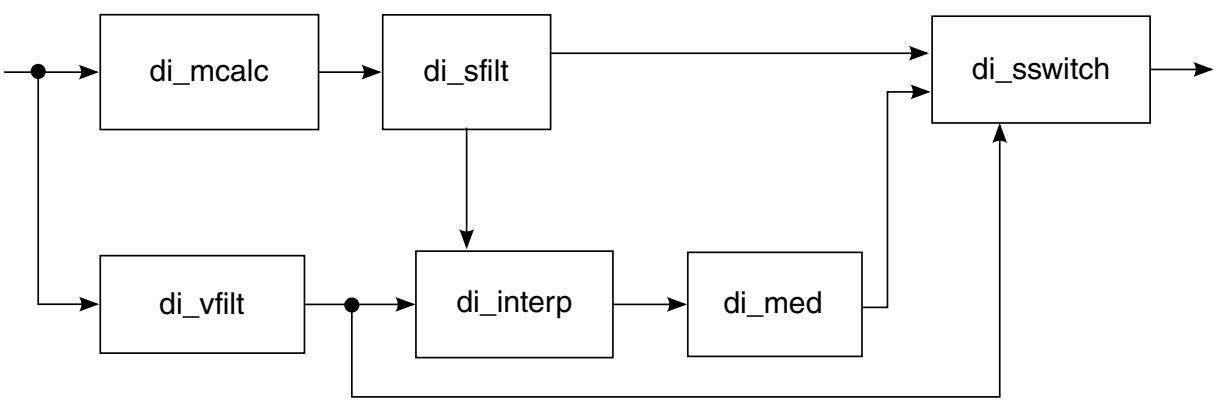


Figure 37-48. DI block diagram

37.4.11.2.1 Vertical Filter Block (di_vfilt)

The di_vfilt block performs spatial vertical filtering of pixels.

It is a four tap vertical filter:

vfilt_out = (-3.0*pix1 + 19.0*pix2 + 19.0*pix3 - 3.0*pix4)/32.0

Where pix1, pix2, pix3, pix4 are four pixels in same horizontal location on four consecutive lines of a field.

vfilt_out is pixel being predicted (between pix2 and pix3)

37.4.11.2.2 Motion Calculator Block (di_mcalc)

The mcalc block estimates the amount of motion for any given pixel by looking at pixel values in current, previous and next fields.

The generic formula used to calculate the estimated motion is:

m = SAT(Ks*(|e-w|/(|e-w|+|n-s|+SPA_DETAIL)))

- Where, n is the pixel above the pixel being predicted
- s is the pixel below the pixel being predicted
- e is the pixel in previous field at same spatial location as the pixel being predicted
- w is the pixel in next field at same spatial location as the pixel being predicted
- m is motion estimation for the pixel being predicted (range from 0 to 1)
- Ks is slope control and decides how quickly the algorithm switches from no motion (m=0) to full motion (m=1)
- SPA_DETAIL is a constant (50) that is added to |n-s|.
- SAT() is saturate at 1 function

The motion calculator block is simplified in a certain respect, by removing the need for a divider, while providing a degree of flexibility. The main motivators for this are the following observations:

1. the above equation defines a set of curves based on the value of |n-s|, but the curves are fairly closely spaced, so that using a granularity of 8 in |n-s| to define specific curves to use gives pretty much the same quality of picture as using all of the curves; and once |n-s| reaches about 120, the effect of an increased |n-s| value is hard to observe
2. once |e-w| gets to about 15, the motion is usually saturated to a value of 1

Based on the above observations, the motion calculator is now implemented with the use of two "ROM"s, using the 4 LSBs of |e-w| and bits 6:3 of |n-s|.

The motion calculator has 3 modes of operation. These modes are defined by the VDI_MOT_SEL field. In case that the user has an information about the motion (For example SW analyzing motion vectors provided by a video decoder) he can select one of the modes listed below. Changing the value of the VDI_MOT_SEL has an affect only on the next frame.

- When VDI_MOT_SEL==2'b01, m_calc is 0 (no motion - use weave)

- When VDI_MOT_SEL==2'b10, We assume high motion and use min(15,delta_t).
- When VDI_MOT_SEL==2'b00, We assume low motion and use sat([0,15],delta_t-8) (delta_t-8 is signed operation).

37.4.11.2.3 Spatial Motion Filter (di_sfilt)

The di_sfilt block spreads motion signal over five pixels:

Mspread = MAX(m3, (0.5*m1 + m2 + m3 + m4 + 0.5*m5)/4.0)

- Where, m3 is motion estimate for current pixel
- m2 is motion estimate for previous pixel
- m4 is motion estimate for next pixel
- m1 is motion estimate for pixel before previous pixel
- m5 is motion estimate for pixel after next pixel

37.4.11.2.4 Interpolated Pixel Calculator Block (di_interp)

The di_interp block uses the motion estimated by the di_sfilt block and computes an interpolated pixel that is weighted sum of the surrounding four pixels (n, s, e, w).

The block performs the following calculations:

```
          if (Mspread <= 0.5) {
i = (1-2*Mspread)*(e+w)/2 + 2*Mspread*vfilt_out
} else {
          i = vfilt_out
}
```

Where, i is the interpolated pixel

n, s, e, w are surrounding pixels as explained earlier

37.4.11.2.5 Median Filter Block (di_med)

The di_med block performs a 5-point median of n, s, e, w and i pixels.

It should be noted that the median required here is not a true 5-point, but can be implemented more efficiently as a 3-point median:

med = MEDIAN(min(max(n,s),max(e,w)), max(min(n,s),min(e,w)), i)

37.4.11.2.6 Soft Switch Block (di_sswitch)

The final output of the deinterlacer is a blend of the median value and the vertical filter, assuming that the pixel data n,s are uncorrelated with the pixel data e,w. By uncorrelated, we mean (max(n,s) < min(e,w)) or (min(n,s) > max(e,w)).

```
          if (Mspread <= 0.5 || not(F)) {
pix_out =    med
} else { /* Mspread > 0.5 */
pix_out = (1-2*(Mspread-0.5))*med + 2*(Mspread-0.5)*vfilt_out
}
```

37.4.11.3 DMA only Mode

In DMA only mode the data is coming from IDMAC only.

37.4.11.4 Real Time Mode

In Real Time Mode the F(n-1) are coming from CSI. The CSI write to FIFO1. The DI sub-block read F(n-1) from processing. In addition IDMAC read the field from FIFO1 and store in external memory. Then stored frames are used as F(n) and F(n+1).

37.4.11.5 CSI only Mode

In CSI only mode VDIC do not perform any processing and used as FIFO only. The data arrived from CSI written to FIFO1. The IDMAC read data from the FIFO1. The FIFO3 and Line store memory are not used. The ID sub-blocks are turned OFF. The CSI only mode can be used as alternative to SMFC, when appropriate.

37.4.11.6 Using Combining in the VDIC

As an alternate function to the de interacing function, the VDIC can perform combining of two planes.

- Both planes have to be at the same color space.
- Overlaying of a single plane over a unified background is supported.
- The planes can be at a different sizes - one plane can be smaller than the other.
- Combining requires a single cycle per output pixel.
- Alpha blending (global or local) is supported.
- Color keying is supported.

The plane's location and size is programmable as shown in the following figure.

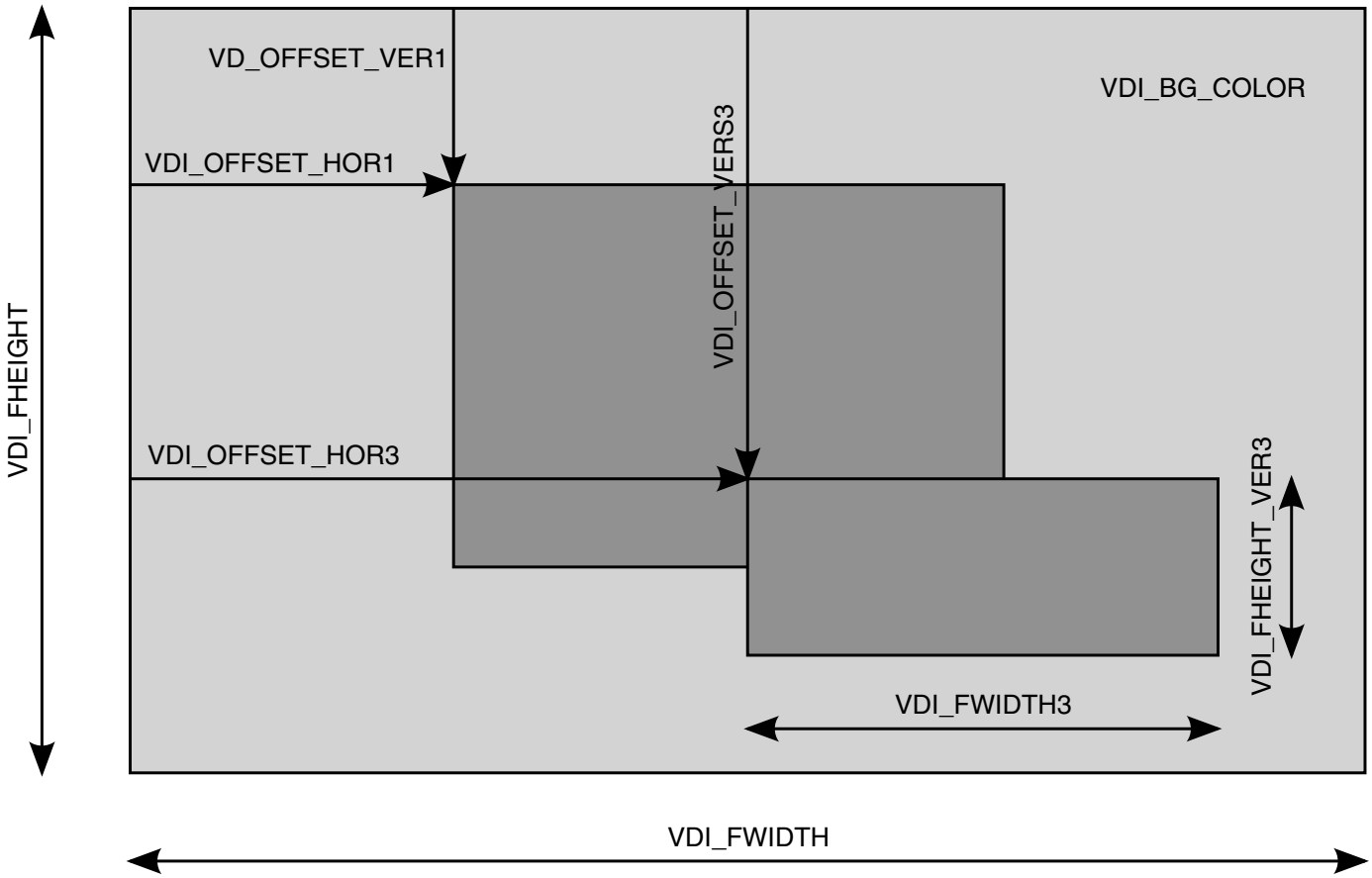


Figure 37-49. The relations between planes of the combining unit

When local alpha is used it should be arrived through channel3 of the VDIC.

The combining equation is:

OP = BG*(1 - alpha) + FG*alpha

Where "OP" is output pixel, "FG" is input pixel of foreground plan arrived thought channel3, BG is input pixel of background plan arrived thought channel1, "alpha" is global or local transparency.

37.4.11.7 VDIC Restrictions

- Maximum output pixel rate is 100 MP/s.
- The output pixel format (YUV422 or YUV420) are equal to input format.
- The VDIC can perform combining or de interlacing. It cannnot perform both functions at the same time.

37.4.12 Control Module (CM)

37.4.12.1 Block Diagram

The CM Block Diagram is shown in the figure below.

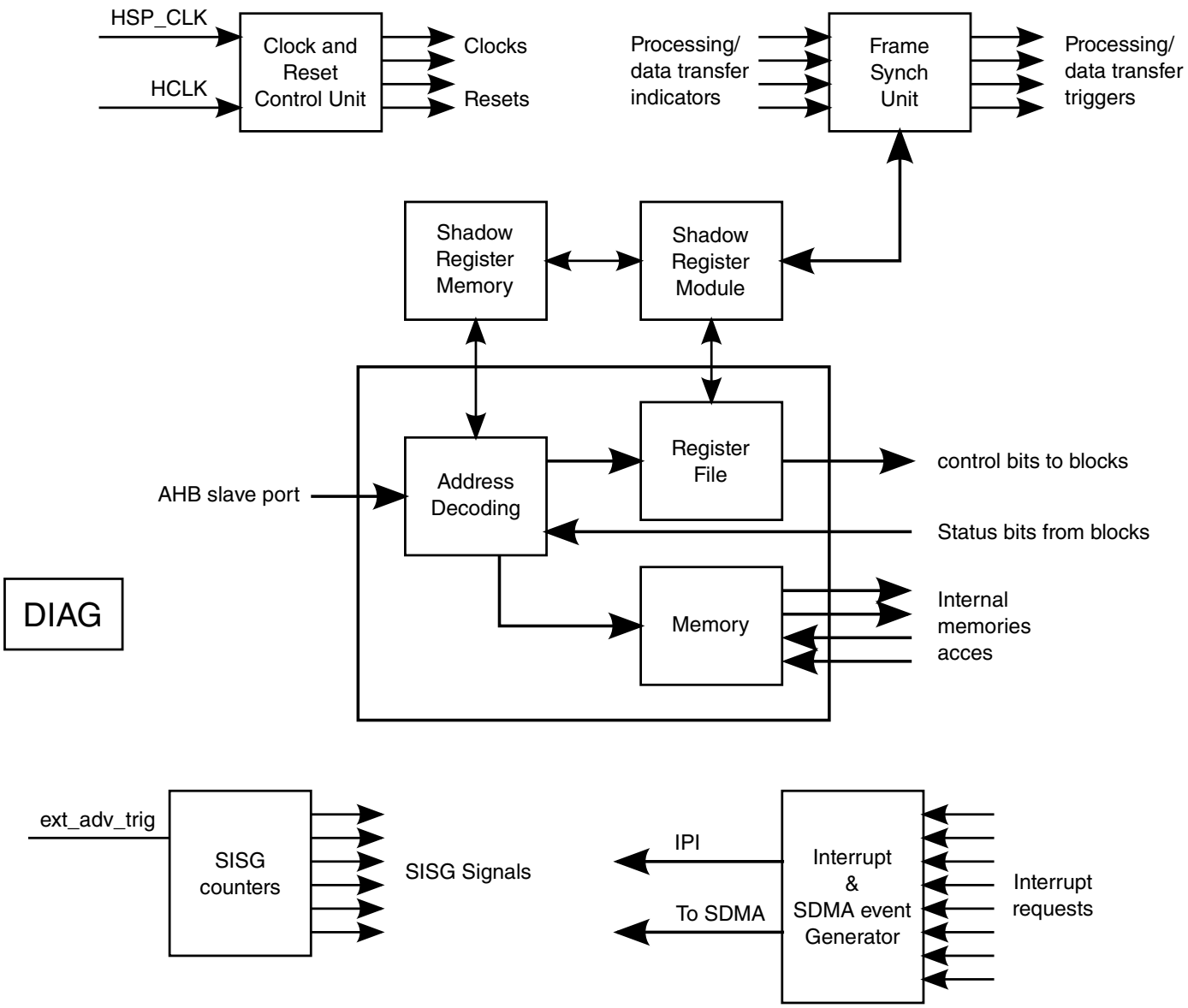


Figure 37-50. CM Block Diagram

The CM consists of the Frame Synchronization Unit (FSU), the Interrupt Generator (IG), the General Configuration Registers (GCR), the Clock and Reset Control Unit (CRCU) and the Shadow Registers Block (SRM).

37.4.12.2 Frame Synchronization Unit

This section details the frame synchronization unit.

37.4.12.2.1 General Description

The FSU provides synchronization of tasks performed by different IPU sub-blocks and ARM platform tasks. This allows to build complex processing flows which are performed automatically (without ARM platform involvement in synchronization of the IPU's tasks).

The FSU supports double buffering of image frames stored in the external memory and allows chaining IPU processing flows in automatic mode.

37.4.12.2.2 Frame Synchronization Flow

1. Initialization

   The ARM platform initializes all the parameters for a task by writing to the GCR and to the parameters memories of each IPU sub-blocks. The initialization must occur before the ARM platform enables the task.

2. Enabling

   After the initialization step has been completed, the ARM platform enables the task by setting its enable bit in an appropriate register.

3. Triggering

   After the task is enabled, the FSU waits for triggering signal. The triggering signals is a combination of the enable bit and the buffer ready signal which can be driven by the ARM platform (DMA_CH_BUF<0/1>_RDY_<#>) and/or by the preceding task (IDMAC_EOF_# or Frame Complete signal from the DC).

   The trigger causes the FSU to invoke the relevant unit to start by assertion of the NEW_FRM_RDY signal. In some cases triggering occurs at the enabling step (when the enable bit is the trigger for the task).

4. Operating

   The triggering step cause the task to move to active mode, this is the operating step. In this step, the FSU monitors the synchronization signals from ARM platform, IDMAC and the corresponding processing units, and controls the units operation. The FSU also controls the IDMAC buffer toggling when double buffer page flipping is used.

The FSU checks at end of each frame if the next frame can be served. If the answer is yes, the FSU stays in active mode with re-sending the <TASK>_NEW_FRM_RDY signal and updating the relevant flags (e.g. DMA<BL>_<#>_CUR_BUF and DMA<BL>_<#>_BUF_RDY). If the answer is no, the FSU moves to pause mode and pauses the task waiting until the next frame can be served.

5. Disabling

   When the task is disabled by the ARM platform (by negating the enable bit), it moves back to non-active mode.

37.4.12.2.3 FSU's fundamentals

Trigger source select

A flow is triggered by a trigger. The trigger may be asserted manually by the ARM platform or may be a result of the completion of the preceding task. Trigger's source select choose the source of the trigger. The trigger means that the data is ready to be processed by the sub-blocks. The trigger source select is defined by the corresponding SRC_SEL bits of the block or task.

Trigger destination select

A block or task that process data needs to know that the following task in the chain is ready to receive the processed data. The user needs to specify what is the destination of the processed data. This is done by setting the corresponding DEST_SEL bits of the block or task.

Double buffering

The IPU supports double buffering in the system's memory. When a flow is processed frame by frame the first frame will be read from one location (BUF0) in the memory, the next frame will be read from another location in the memory (BUF1). The location in the memory of the buffers is defined by the EBA0 and EBA1 parameters of the corresponding channel in the CPMEM. The IDMAC use the correct buffer according to the DMA_CH_CUR_BUF_# signal. The FSU automatically toggles the DMA_CH_CUR_BUF_# to point on the correct buffer to be used by the channel. In order to work in double buffer mode the corresponding DMA_CH_DB_MODE_SEL_# needs to be set.

Alternative flow

Some of the IPU sub-blocks can handle 2 flows via them one is the main flow and the other is the alt flow. In order to support an alternate flow via the same sub-block an alternative configuration should be used by the sub-block. This includes

- Alternate registers including an alternative sub-block's setup - this is handled by the SRM
- Alternate IDMAC settings: parameters in the CPMEM, separate alpha
- Alternate SRC_SEL as the source of the trigger may come from a different function.
- Alternate FSU settings (CUR_BUF, BUF_RDY, DB_MODE_SEL)
- Some of the display sub-blocks alternate settings are handled by programming an alternative set of registers.

The FSU handles the switch to the alternate flow, it controls the update of the alternate configuration and send the appropriate signals to other sub-blocks in the IPU indicating about a need to switch to the alternate configuration.

Once a frame is completed there is a chance that there are 2 buffers ready. One is the next buffer of the current flow and the other is the a buffer in an alternate flow. The FSU will automatically select between the next buffer to handle in a round-robin fashion.

IPU task chaining - Single flow

The diagram below illustrates task chaining in the In this example a single flow is handled

The Frames are coming from the same camera sub-block. The frames are handled frame by Frame. The first frame arriving Frame0 is stored in BUF0 of the "INPUT BUFFER". Once the Frame0 is ready in BUF0, the processing sub-block is triggered and data is read from BUF0 to the processing sub-block. In the meantime, Frame1 coming from the camera is written to BUF1 of the "INPUT BUFFER". The processing sub-block processes the data and stores the first frame on BUF0 of the "OUTPUT BUFFER". Once the processing is done and the frame is ready, the display sub-block is triggered to pick the data and send it to the display. The processing sub-block will start working on the next frame once the data is ready in the "INPUT_BUFFER" and there's a free buffer in the "OUTPUT_BUFFER".

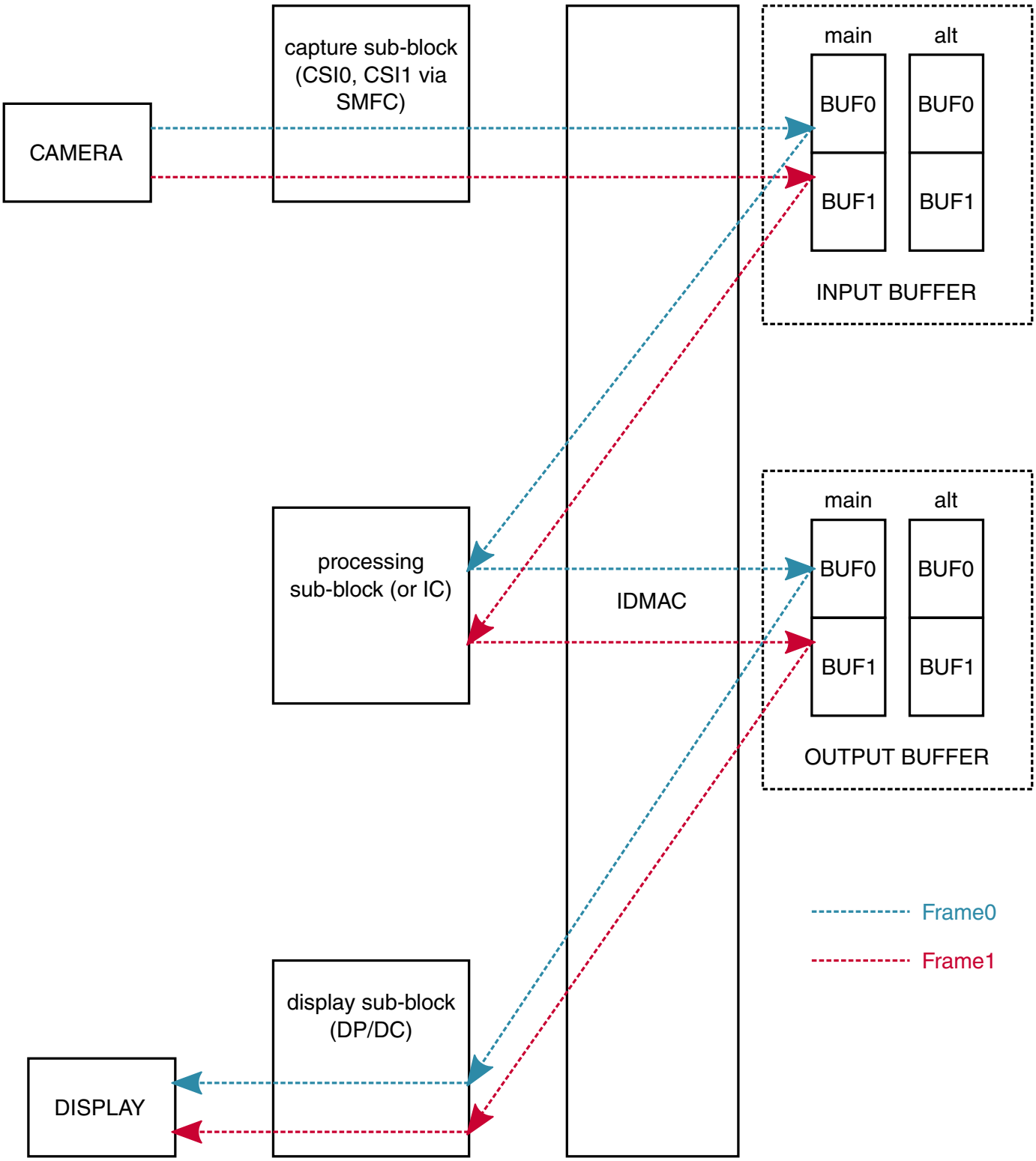


Figure 37-51. IPU tasks' chaining illustration - single flow

IPU task chaining - double flow

The diagram below illustrates task chaining in the In this example double flow is handled. In addition to the flow described on the previous example, another flow is handled via the display sub-blocks. In that case the DC will handle 2 flows. Once sending Frame0 to the display is complete, the FSU will decide in whether to handle Frame1 or Frame0_ALT. The decision is made according the readiness of the other buffers in the memory, in case of 2 ready buffers (main flow and alternate) the FSU will switch between them in a round robin fashion.

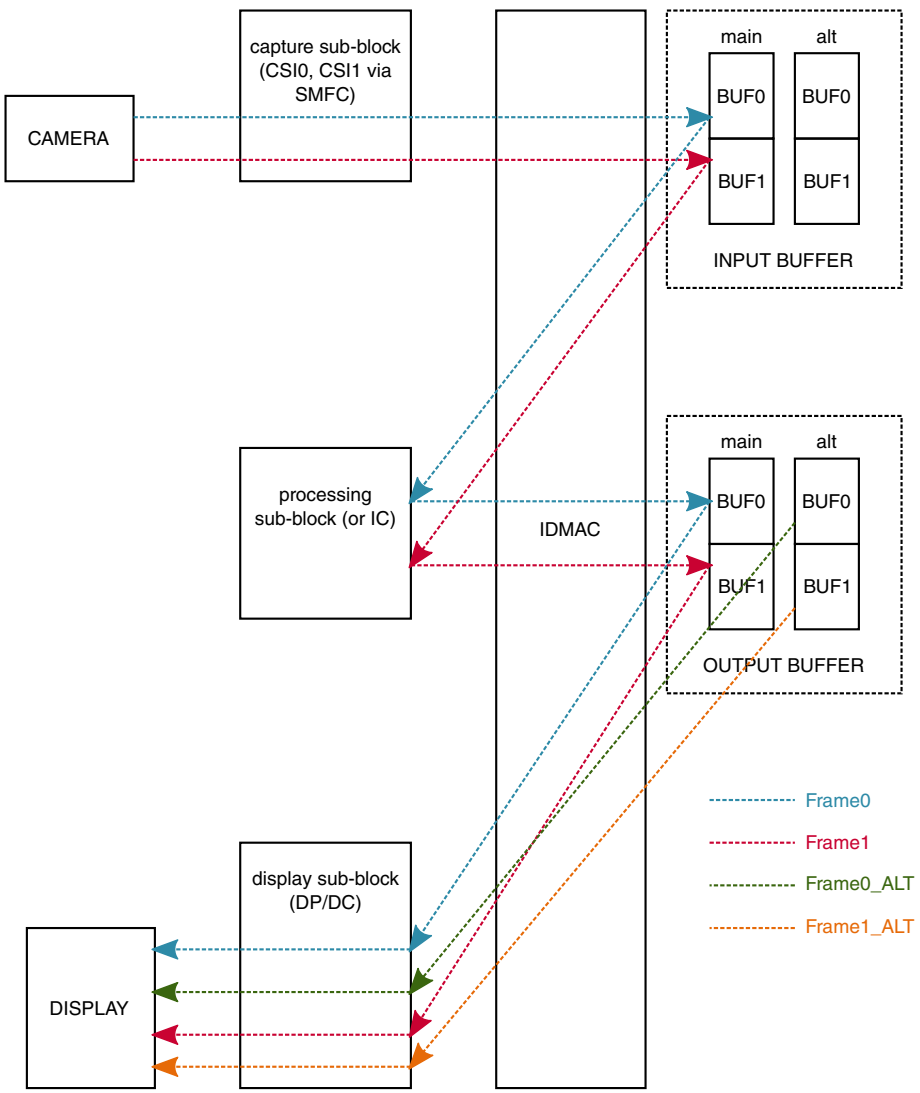


Figure 37-52. IPU tasks' chaining illustration - double flow

37.4.12.2.4 IPU main flows

IPU's flows can be partitioned into 5 groups.

- CF - capture flows

- PF - processing flows
- SF - synchronous display flows
- AF - Asynchronous display flows
- DF - Direct flow from IC to the display

Tasks from different groups can be chained as illustrated below.

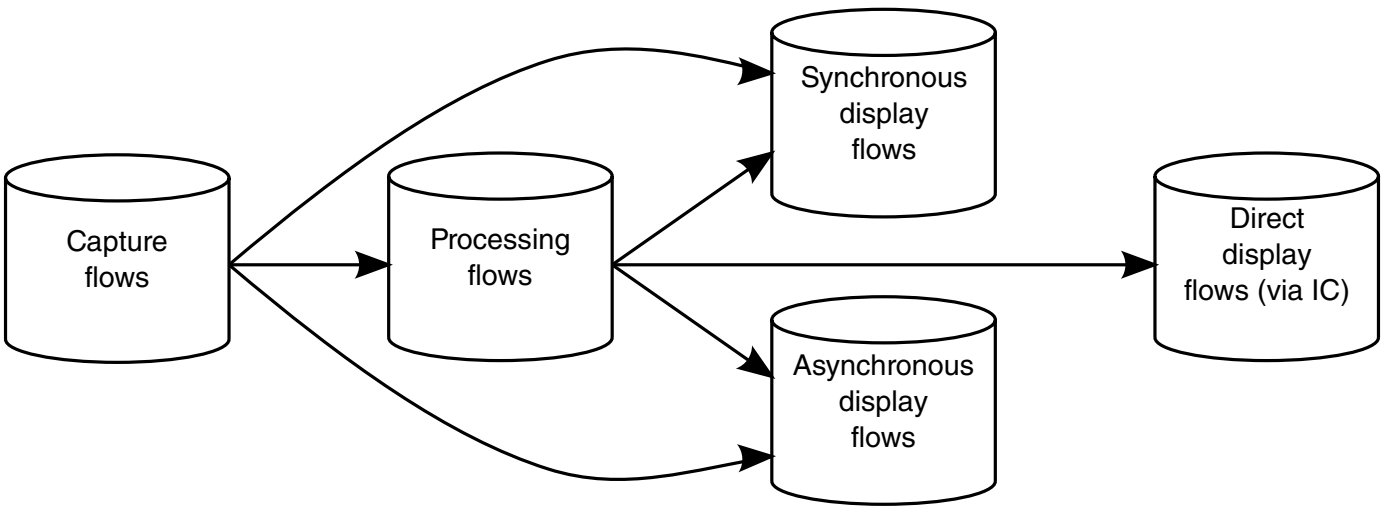


Figure 37-53. IPU task flows chaining

The tables below describe the most important use cases of chaining the IPU tasks. The tables show the main task for each group.

The physical DMA channel is the DMA channel that is used for the main flow - this is the IDMAC channel that is physically connected to the block - the CPMEM parameters should be configured according to the physical channel number. In case of an alternate flow then an alternate entry in the CPMEM should be configured as well

The following table describes capturing flows where data is captured from the sensor and sent to the memory without processing. This flows can be chained to the processing and display flows described on: Table 37-31, Table 37-32, Table 37-33 and on Table 37-34.

Table 37-30. IPU's capture flows

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Capturing image from sensor and storing it in the memory (via SMFC) without processing	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3

Table continues on the next page...

Table 37-30. IPU's capture flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Capturing image from sensor and storing it in the memory (via IC) without processing	CSI0 or CSI1 --> MEM	---	---	IDMAC_CH_5
	Capturing interlaced input and storing it in the memory (via VDIC) while performing video de-interlacing in the VDIC.	CSI0 or CSI1 -> VDIC --> MEM		IDMAC_CH_9 IDMAC_CH_10	IDMAC_CH_5 IDMAC_CH_13
		Interlaced input coming from one of the CSIs is sent to the memory without processing via channel #13. In addition 2 more inputs are read from the memory via channels 9 and 10. The processed image is written to the memory via ch 5 in progressive scan mode.			

NOTE

Register IPUx_SMFC_MAP is used to map CSI0 and CSI1 inputs to one of the four IDMAC channels. See Sensor Multi FIFO Controller (SMFC) for additional information.

The following table describe processing flows via the IC . This flows can start from the memory, can be chained to a capturing flow described on Table 37-30. The target of this flow can be chained to the display flows described on Table 37-32 and on Table 37-33.

Table 37-31. IPU's Processing flows

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Preprocessing image from sensor for encoding	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (PRP ENC) --> MEM	IDMAC_CH_12	---	IDMAC_CH_20
	Preprocessing image from memory for encoding	MEM --> IC (PRP ENC) --> MEM	IDMAC_CH_12	---	IDMAC_CH_20
	Preprocessing image from sensor for encoding	CSI0 or CSI1 --> IC (PRP ENC) --> MEM	---	---	IDMAC_CH_20
	Preprocessing + rotation of image from sensor for encoding	CSI0 or CSI1 -->SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (PRP ENC) --> MEM	IDMAC_CH_12	---	IDMAC_CH_20
		MEM --> IC (ROT ENC) --> MEM	IDMAC_CH_45	---	IDMAC_CH_48

Table continues on the next page...

Table 37-31. IPU's Processing flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Preprocessing + rotation of image from memory for encoding	MEM --> IC (PRP ENC) --> MEM	IDMAC_CH_12	---	IDMAC_CH_20
		MEM --> IC (ROT ENC) --> MEM	IDMAC_CH_45	---	IDMAC_CH_48
	Rotation and preprocessing of image from sensor for encoding	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (ROT ENC) --> MEM	IDMAC_CH_45	---	IDMAC_CH_48
		MEM --> IC (PRP ENC) --> MEM	IDMAC_CH_12	---	IDMAC_CH_20
	Preprocessing image from sensor for viewfinder	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (PRP VF) --> MEM	IDMAC_CH_12	IDMAC_CH_14	IDMAC_CH_21
	Preprocessing image from memory for viewfinder	MEM --> IC (PRP VF) --> MEM	IDMAC_CH_12	IDMAC_CH_14	IDMAC_CH_21
	Preprocessing and rotation of image from sensor for viewfinder	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (PRP VF) --> MEM	IDMAC_CH_12	IDMAC_CH_14	IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
	Rotation and preprocessing of image from sensor for viewfinder	CSI0 or CSI1 -->SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM --> IC (PRP VF) --> MEM	IDMAC_CH_12	IDMAC_CH_14	IDMAC_CH_21
	Preprocessing and rotation of image from sensor for viewfinder	MEM --> IC (PRP VF) --> MEM	IDMAC_CH_12	IDMAC_CH_14	IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
	Preprocessing image from sensor	CSI0 or CSI1 --> IC (PRP VF) --> MEM	---	IDMAC_CH_14	IDMAC_CH_21
	Preprocessing and rotation of image from sensor	CSI0 or CSI1 --> IC (PRP VF) --> MEM	---	IDMAC_CH_14	IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
	Postprocessing image	MEM --> IC (PP) --> MEM	IDMAC_CH_11	IDMAC_CH_15	IDMAC_CH_22
	Postprocessing and rotation of image	MEM --> IC (PP) --> MEM	IDMAC_CH_11	IDMAC_CH_15	IDMAC_CH_22
		MEM --> IC (ROT PP) -->MEM	IDMAC_CH_47	---	IDMAC_CH_50

Table continues on the next page...

Table 37-31. IPU's Processing flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Rotation and postprocessing of image	MEM --> IC (ROT PP) -->MEM	IDMAC_CH_47	---	IDMAC_CH_50
		MEM --> IC (PP) --> MEM	IDMAC_CH_11	IDMAC_CH_15	IDMAC_CH_22
	Postprocessing image from sensor	CSI0 or CSI1 -->SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (PP) --> MEM	IDMAC_CH_11	IDMAC_CH_15	IDMAC_CH_22
	video de-interlacing in the VDIC.	MEM -> VDIC --> MEM	IDMAC_CH_8 IDMAC_CH_9 IDMAC_CH_10	--	IDMAC_CH_5
		Interlaced input coming from the memory via 3 channels 8, 9 and 10. The processed image is written to the memory via ch 5 in progressive scan mode.			
	video de-interlacing in the VDIC. Then processing in the IC.	MEM -> VDIC --> IC (PRP VF) --> MEM	IDMAC_CH_8 IDMAC_CH_9 IDMAC_CH_10	IDMAC_CH_14	IDMAC_CH_21
		Interlaced input coming from the memory via 3 channels 8, 9 and 10. The processed image is sent to the IC for further processing then it is sent to the memory via ch 21			
	video de-interlacing in the VDIC, then preprocessing and rotation of image coming from memory	MEM --> VDIC --> IC (PRP VF) --> MEM	IDMAC_CH_8 IDMAC_CH_9 IDMAC_CH_10	IDMAC_CH_14	IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
	video de-interlacing in the VDIC, then preprocessing and rotation of image coming from CSI	CSI0 or CSI1 --> VDIC--> IC (PRP VF) --> MEM	IDMAC_CH_9 IDMAC_CH_10	IDMAC_CH_14	IDMAC_CH_13 IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
	Combining in the VDIC.	MEM -> VDIC --> MEM	IDMAC_CH_25I DMAC_CH_26	--	IDMAC_CH_5
		Plane3 is coming on IDMAC_CH_25; Additional plane (plane1) may come from IDMAC_CH_26.			
	Combining in the VDIC. Then processing in the IC.	MEM -> VDIC --> IC (PRP VF) --> MEM	IDMAC_CH_25I DMAC_CH_26	IDMAC_CH_14	IDMAC_CH_21
		Plane3 is coming on IDMAC_CH_25; Additional plane (plane1) may come from IDMAC_CH_26.			
	Combining in the VDIC, then preprocessing and rotation	MEM --> VDIC --> IC (PRP VF) --> MEM	IDMAC_CH_25I DMAC_CH_26	IDMAC_CH_14	IDMAC_CH_21
		MEM --> IC (ROT VF) --> MEM		---	IDMAC_CH_49
		Plane3 is coming on IDMAC_CH_25; Additional plane (plane1) may come from IDMAC_CH_26.			
	Rotation of an image and Combining in the VDIC	MEM --> IC (ROT VF) -->MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM -> VDIC --> MEM	IDMAC_CH_25I DMAC_CH_26	--	IDMAC_CH_5

Table continues on the next page...

Table 37-31. IPU's Processing flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
		Plane3 is rotated on the VF task. Then it sent to the VDIC IDMAC_CH_25; Additional plane (plane1) may come from IDMAC_CH_26.			
	Rotation of an image and Combining in the VDIC	MEM --> IC (ROT ENC) -->MEM	IDMAC_CH_45	---	IDMAC_CH_48
		MEM -> VDIC --> MEM	IDMAC_CH_25I DMAC_CH_26	--	IDMAC_CH_5
		Plane1 is rotated on the ENC task. Then it sent to the VDIC IDMAC_CH_26; In additional plane3 is coming from IDMAC_CH_25			
	Rotation of two images and Combining in the VDIC	MEM --> IC (ROT VF) -->MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM --> IC (ROT ENC) -->MEM	IDMAC_CH_45	---	IDMAC_CH_48
		MEM -> VDIC --> MEM	IDMAC_CH_25I DMAC_CH_26	--	IDMAC_CH_5
		Plane3 is rotated on the VF task. Plane1 is rotated on the ENC task.			
	Rotation of two images and Combining in the VDIC. Then processing in the IC.	MEM --> IC (ROT VF) -->MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM --> IC (ROT ENC) -->MEM	IDMAC_CH_45	---	IDMAC_CH_48
		MEM -> VDIC --> IC (PRP VF) --> MEM	IDMAC_CH_25I DMAC_CH_26	IDMAC_CH_14	IDMAC_CH_21
		Plane3 is rotated on the VF task. Plane1 is rotated on the ENC task.			

The following table describes the synchronous display flows, this flows can be chained to the capture, processing and direct flows described on Table 37-30,Table 37-30and on Table 37-34

Table 37-32. IPU synchronous display flows

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Synchronous display refresh via DC	MEM --> DC	IDMAC_CH_28	---	---
	Synchronous display refresh via DP	MEM --> DP	IDMAC_CH_23	---	---
	Synchronous display refresh via DC	MEM --> DC	IDMAC_CH_28	IDMAC_CH_44	---
		Comment: IDMAC_CH_44 is an optional mask Channel			
	Synchronous display refresh via DP	MEM --> DP	IDMAC_CH_23	IDMAC_CH_44	---
		Comment: IDMAC_CH_44 is an optional mask Channel			
	Synchronous display refresh via DP + combining in the DP	MEM --> DP SYNC(BG/FG)	IDMAC_CH_23	IDMAC_CH_27	---
		Comment: IDMAC_CH_23 is the main channel; IDMAC_CH_27 is used for combining of another plane.			

Table continues on the next page...

Table 37-32. IPU synchronous display flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Synchronous display refresh via DP	MEM --> DP SYNC(BG/FG)	IDMAC_CH_23/ IDMAC_CH_27	IDMAC_CH_44 (mask)	---
		Comment: IDMAC_CH_23 is the main channel; IDMAC_CH_27 is optional and can be used for combining of another plane. IDMAC_CH_44 is an optional mask Channel			

The following table describes the asynchronous display flows, this flows can be chained to the capture, processing and direct flows described onTable 37-30,Table 37-30and on Table 37-34

Table 37-33. IPU Asynchronous display flows

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Asynchronous display refresh via DP	MEM --> DC	IDMAC_CH_24	---	---
	Asynchronous display refresh via DP	MEM --> DC	IDMAC_CH_24/ IDMAC_CH_29	---	---
		Comment: IDMAC_CH_29 is another input to the DP to be combined with data coming from IDMAC_CH_24			
	Asynchronous display refresh via DP in command buffer mode	MEM --> DC	IDMAC_CH_24/ IDMAC_CH_29	IDMAC_CH_42 (command)	---
		Comment: IDMAC_CH_29 is another input to the DP to be combined with data coming from IDMAC_CH_24			
	Asynchronous display refresh via DC	MEM --> DC	IDMAC_CH_28	---	---
	Asynchronous display refresh via DC	MEM --> DC	IDMAC_CH_41	---	---
	Asynchronous display refresh via DC	MEM --> DC	IDMAC_CH_28	IDMAC_CH_42(command)	---
		Comment: IDMAC_CH_42 can be optionally used as command channel associated with IDMAC_CH_28			
	Asynchronous display refresh via DC	MEM --> DC	IDMAC_CH_41	IDMAC_CH_43(command)	---
		Comment: IDMAC_CH_43 can be optionally used as command channel associated with IDMAC_CH_41			
	Asynchronous display refresh via DP	MEM --> DP ASYNC	IDMAC_CH_24	IDMAC_CH_42(command)	---
		Comment: IDMAC_CH_42 can be optionally used as command channel associated with IDMAC_CH_24			

Table continues on the next page...

Table 37-33. IPU Asynchronous display flows (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Asynchronous display refresh via DP	MEM --> DC	IDMAC_CH_24	IDMAC_CH_43 (command)	---
		Comment: IDMAC_CH_43 can be optionally used as command channel associated with IDMAC_CH_24			
	Reading data from asynchronous display	DC --> MEM	---	---	IDMAC_CH_40

The table below describes direct flows via the IC. These are processing flows via the IC where the output is sent directly to the DMFC. Direct camera to display flow is possible when the frame rate of the source and the destination is the same (typically the target display will be asynchronous display, where the display is updated at the rate of the source). From that point, any of the display flows described on Table 37-32 and on Table 37-33 can be chained.

Table 37-34. IPU direct flows to the display via the IC

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
	Rotation and preprocessing of image from sensor for viewfinder and displaying it on synchronous display via DP	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM --> IC (PRP VF) --> DMFC	IDMAC_CH_12	IDMAC_CH_14	Direct to the DMFC
		MEM --> DP SYNC (BG/FG)	IDMAC_CH_23/ IDMAC_CH_27	IDMAC_CH_44 (mask)	---
	Rotation and preprocessing of image from sensor for viewfinder and displaying it on asynchronous display via DP	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3
		MEM --> IC (ROT VF) --> MEM	IDMAC_CH_46	---	IDMAC_CH_49
		MEM --> IC (PRP VF) --> DMFC	IDMAC_CH_12	IDMAC_CH_14	Direct to the DMFC
		MEM --> DP SYNC (BG/FG)	IDMAC_CH_24/ IDMAC_CH_29	command channel	---
	Preprocessing image from sensor for viewfinder and direct displaying it on asynchronous display	CSI0 or CSI1 --> SMFC --> MEM	---	---	IDMAC_CH_0 IDMAC_CH_1 IDMAC_CH_2 IDMAC_CH_3

Table continues on the next page...

Table 37-34. IPU direct flows to the display via the IC (continued)

	Flow	Tasks chain	Physical DMA Channels		
			Video Input	Other Input	Output
		MEM --> IC (PRP VF) --> DC	IDMAC_CH_12	IDMAC_CH_14	---
	Preprocessing image from sensor for viewfinder and direct displaying it on asynchronous display	CSI0 or CSI1 --> IC (PRP VF) --> DC	---	IDMAC_CH_14	IDMAC_CH_21
	Preprocessing image from sensor for viewfinder and direct displaying it on asynchronous display via DP	CSI0 or CSI1 --> IC (PRP VF) --> DP ASYNC	---	IDMAC_CH_14	IDMAC_CH_21
	Capturing interlaced input via CSI, then perform video de-interlacing in the VDIC. Then processing in the IC and direct displaying it on asynchronous display via DP	CSI0 or CSI1 -> VDIC --> IC (PRP VF) --> DP ASYNC	--	IDMAC_CH_9 IDMAC_CH_10 IDMAC_CH_14	IDMAC_CH_13 IDMAC_CH_21
		Interlaced input coming from one of the CSIs is sent to the memory without processing via channel #13. In addition 2 more inputs are read from the memory via channels 9 and 10. The processed image is sent to the IC for further processing then it is sent to the display			
	Capturing interlaced input via CSI, then perform video de-interlacing in the VDIC. Then processing in the IC and direct displaying it on asynchronous display	CSI0 or CSI1 -> VDIC --> IC (PRP VF) --> DC	--	IDMAC_CH_9 IDMAC_CH_10 IDMAC_CH_14	IDMAC_CH_13 IDMAC_CH_21
		Interlaced input coming from one of the CSIs is sent to the memory without processing via channel #13. In addition 2 more inputs are read from the memory via channels 9 and 10. The processed image is sent to the IC for further processing then it is sent to the display			

37.4.12.2.5 Sub-Frame Double-Buffering (Band Mode)

Page-flip double buffering is performed using full-frame buffers.

In addition IPU supports also page-flip double buffering using smaller buffers, each containing 4/8/16/32/64/128/256 rows of pixels. This allows the use of internal memory for buffering.

This mode can be supported by the following modifications (relative to full-frame buffers):

The address in system memory is generated by the same formula, but inserting the full row number, only k LSB's (for a band of 2^k rows) are inserted (e.g. [2:0] for 8 rows).

Page flip is triggered at the end of each band (when the k LSB's are all 1) and not only at the end of the frame. This flip may be accompanied by an ARM/SDMA interrupt, if synchronization with other modules (e.g. VPU or ARM) is needed.

The channels that can work in this mode are controlled by the corresponding IDMAC_BNDM_EN_<i>

bit. The BNDM parameter of the corresponding IDMAC channel has to be set as well.

37.4.12.2.6 Automatic Window Refresh

By programming the <>_SRC_SEL bits of the corresponding flow to autoref mode, automatic refresh of a window on the smart display is enabled. This means that the flow is triggered any time the refresh counter completes its counting. The refresh period is defined via the AUTOREF_PER field (the time unit is 2^{17} periods of the HSP_CLK clock).

The actual value of refresh period is equal to:

$T_{HSP}*2^{17}$ *(AUTO_REF_PER+1).

37.4.12.2.7 IPU VDOA synchronization

IPU can interact to a companion block called VDOA.

VDOA (Video Data Order Adapter) is used to re-order video data from the "tiled" order used by the VPU to the conventional raster-scan order needed by the IPU.

The VDOA can transfer its output to the IPU through internal memory, containing a band double buffer.

This tight double-buffering synchronization is performed without MCU involvement, using dedicated signals between the VDOA and IPU and the same protocol as used between two IPU DMA channels.

VDOA notifies the IPU which is the current band to read. IPU notifies the VDOA when the current band is read.

This synchronization is supported for VDI and IC(PP). The target destination of the VDOA data is defined by VDOA_DEST_SEL parameter. The PP_SRC_SEL or VDI_SRC_SEL has to be programmed as well to select VDOA as the source of the task.

For more details about IDMAC's band mode support Sub-Frame Double-Buffering (Band Mode).

The IPU settings and the VDOA settings has to be the same. In particular

- The band height defined at the IPU and at the VDOA must match
- The Number of Frames used by the VDI as defined by the VDI_MOT_SEL has to match the VDOA settings.

37.4.12.3 Interrupt Generator

The IG produces two interrupts to the ARM platform - the functional interrupt and the error interrupt. All of the interrupts are maskable.

The following table describes the functional interrupts.

Table 37-35. Functional Interrupts Summary

Location	Sub-blocks	Interrupt status bit name	Description
IPU_INT_STAT_1[0]	IDMAC	IDMAC_EOF_0	-
IPU_INT_STAT_1[1]	IDMAC	IDMAC_EOF_1	-
IPU_INT_STAT_1[2]	IDMAC	IDMAC_EOF_2	-
IPU_INT_STAT_1[3]	IDMAC	IDMAC_EOF_3	-
IPU_INT_STAT_1[5]	IDMAC	IDMAC_EOF_5	-
IPU_INT_STAT_1[11]	IDMAC	IDMAC_EOF_11	-
IPU_INT_STAT_1[12]	IDMAC	IDMAC_EOF_12	-
IPU_INT_STAT_1[14]	IDMAC	IDMAC_EOF_14	-
IPU_INT_STAT_1[15]	IDMAC	IDMAC_EOF_15	-
IPU_INT_STAT_1[17]	IDMAC	IDMAC_EOF_17	-
IPU_INT_STAT_1[18]	IDMAC	IDMAC_EOF_18	-
IPU_INT_STAT_1[20]	IDMAC	IDMAC_EOF_20	-
IPU_INT_STAT_1[21]	IDMAC	IDMAC_EOF_21	-
IPU_INT_STAT_1[22]	IDMAC	IDMAC_EOF_22	-
IPU_INT_STAT_1[23]	IDMAC	IDMAC_EOF_23	-
IPU_INT_STAT_1[24]	IDMAC	IDMAC_EOF_24	-
IPU_INT_STAT_1[27]	IDMAC	IDMAC_EOF_27	-
IPU_INT_STAT_1[28]	IDMAC	IDMAC_EOF_28	-
IPU_INT_STAT_1[29]	IDMAC	IDMAC_EOF_29	-
IPU_INT_STAT_1[31]	IDMAC	IDMAC_EOF_31	-
IPU_INT_STAT_2[1]	IDMAC	IDMAC_EOF_33	-
IPU_INT_STAT_2[8]	IDMAC	IDMAC_EOF_40	-
IPU_INT_STAT_2[9]	IDMAC	IDMAC_EOF_41	-
IPU_INT_STAT_2[10]	IDMAC	IDMAC_EOF_42	-
IPU_INT_STAT_2[11]	IDMAC	IDMAC_EOF_43	-
IPU_INT_STAT_2[12]	IDMAC	IDMAC_EOF_44	-
IPU_INT_STAT_2[13]	IDMAC	IDMAC_EOF_45	-
IPU_INT_STAT_2[14]	IDMAC	IDMAC_EOF_46	-
IPU_INT_STAT_2[15]	IDMAC	IDMAC_EOF_47	-
IPU_INT_STAT_2[16]	IDMAC	IDMAC_EOF_48	-
IPU_INT_STAT_2[17]	IDMAC	IDMAC_EOF_49	-

Table continues on the next page...

Table 37-35. Functional Interrupts Summary (continued)

Location	Sub-blocks	Interrupt status bit name	Description
IPU_INT_STAT_2[18]	IDMAC	IDMAC_EOF_50	-
IPU_INT_STAT_2[19]	IDMAC	IDMAC_EOF_51	-
IPU_INT_STAT_2[20]	IDMAC	IDMAC_EOF_52	-
IPU_INT_STAT_3[0]	IDMAC	IDMAC_NFACK_0	-
IPU_INT_STAT_3[1]	IDMAC	IDMAC_NFACK_1	-
IPU_INT_STAT_3[2]	IDMAC	IDMAC_NFACK_2	-
IPU_INT_STAT_3[3]	IDMAC	IDMAC_NFACK_3	-
IPU_INT_STAT_3[5]	IDMAC	IDMAC_NFACK_5	-
IPU_INT_STAT_3[8]	IDMAC	IDMAC_NFACK_8	-
IPU_INT_STAT_3[9]	IDMAC	IDMAC_NFACK_9	-
IPU_INT_STAT_3[10]	IDMAC	IDMAC_NFACK_10	-
IPU_INT_STAT_3[11]	IDMAC	IDMAC_NFACK_11	-
IPU_INT_STAT_3[12]	IDMAC	IDMAC_NFACK_12	-
IPU_INT_STAT_3[13]	IDMAC	IDMAC_NFACK_13	-
IPU_INT_STAT_3[14]	IDMAC	IDMAC_NFACK_14	-
IPU_INT_STAT_3[15]	IDMAC	IDMAC_NFACK_15	-
IPU_INT_STAT_3[17]	IDMAC	IDMAC_NFACK_17	-
IPU_INT_STAT_3[18]	IDMAC	IDMAC_NFACK_18	-
IPU_INT_STAT_3[20]	IDMAC	IDMAC_NFACK_20	-
IPU_INT_STAT_3[21]	IDMAC	IDMAC_NFACK_21	-
IPU_INT_STAT_3[22]	IDMAC	IDMAC_NFACK_22	-
IPU_INT_STAT_3[23]	IDMAC	IDMAC_NFACK_23	-
IPU_INT_STAT_3[24]	IDMAC	IDMAC_NFACK_24	-
IPU_INT_STAT_3[27]	IDMAC	IDMAC_NFACK_27	-
IPU_INT_STAT_3[28]	IDMAC	IDMAC_NFACK_28	-
IPU_INT_STAT_3[29]	IDMAC	IDMAC_NFACK_29	-
IPU_INT_STAT_3[31]	IDMAC	IDMAC_NFACK_31	-
IPU_INT_STAT_4[1]	IDMAC	IDMAC_NFACK_33	-
IPU_INT_STAT_4[8]	IDMAC	IDMAC_NFACK_40	-
IPU_INT_STAT_4[9]	IDMAC	IDMAC_NFACK_41	-
IPU_INT_STAT_4[10]	IDMAC	IDMAC_NFACK_42	-
IPU_INT_STAT_4[11]	IDMAC	IDMAC_NFACK_43	-
IPU_INT_STAT_4[12]	IDMAC	IDMAC_NFACK_44	-
IPU_INT_STAT_4[13]	IDMAC	IDMAC_NFACK_45	-
IPU_INT_STAT_4[14]	IDMAC	IDMAC_NFACK_46	-
IPU_INT_STAT_4[15]	IDMAC	IDMAC_NFACK_47	-
IPU_INT_STAT_4[16]	IDMAC	IDMAC_NFACK_48	-
IPU_INT_STAT_4[17]	IDMAC	IDMAC_NFACK_49	-
IPU_INT_STAT_4[18]	IDMAC	IDMAC_NFACK_50	-

Table continues on the next page...

Table 37-35. Functional Interrupts Summary (continued)

Location	Sub-blocks	Interrupt status bit name	Description
IPU_INT_STAT_4[19]	IDMAC	IDMAC_NFACK_51	-
IPU_INT_STAT_4[20]	IDMAC	IDMAC_NFACK_52	-
IPU_INT_STAT_7[23]	IDMAC	IDMAC_EOS_23	-
IPU_INT_STAT_7[24]	IDMAC	IDMAC_EOS_24	-
IPU_INT_STAT_7[27]	IDMAC	IDMAC_EOS_27	-
IPU_INT_STAT_7[28]	IDMAC	IDMAC_EOS_28	-
IPU_INT_STAT_7[29]	IDMAC	IDMAC_EOS_29	-
IPU_INT_STAT_7[31]	IDMAC	IDMAC_EOS_31	-
IPU_INT_STAT_8[2]	IDMAC	IDMAC_EOS_33	-
IPU_INT_STAT_8[9]	IDMAC	IDMAC_EOS_41	-
IPU_INT_STAT_8[10]	IDMAC	IDMAC_EOS_42	-
IPU_INT_STAT_8[11]	IDMAC	IDMAC_EOS_43	-
IPU_INT_STAT_8[12]	IDMAC	IDMAC_EOS_44	-
IPU_INT_STAT_8[19]	IDMAC	IDMAC_EOS_51	-
IPU_INT_STAT_8[20]	IDMAC	IDMAC_EOS_52	-
IPU_INT_STAT_11[0]	IDMAC	IDMAC_EOBND_0	-
IPU_INT_STAT_11[1]	IDMAC	IDMAC_EOBND_1	-
IPU_INT_STAT_11[2]	IDMAC	IDMAC_EOBND_2	-
IPU_INT_STAT_11[3]	IDMAC	IDMAC_EOBND_3	-
IPU_INT_STAT_11[5]	IDMAC	IDMAC_EOBND_5	-
IPU_INT_STAT_11[11]	IDMAC	IDMAC_EOBND_11	-
IPU_INT_STAT_11[12]	IDMAC	IDMAC_EOBND_12	-
IPU_INT_STAT_11[20]	IDMAC	IDMAC_EOBND_20	-
IPU_INT_STAT_11[21]	IDMAC	IDMAC_EOBND_21	-
IPU_INT_STAT_11[22]	IDMAC	IDMAC_EOBND_22	-
IPU_INT_STAT_12[13]	IDMAC	IDMAC_EOBND_45	-
IPU_INT_STAT_12[14]	IDMAC	IDMAC_EOBND_46	-
IPU_INT_STAT_12[15]	IDMAC	IDMAC_EOBND_47	-
IPU_INT_STAT_12[16]	IDMAC	IDMAC_EOBND_48	-
IPU_INT_STAT_12[17]	IDMAC	IDMAC_EOBND_49	-
IPU_INT_STAT_12[18]	IDMAC	IDMAC_EOBND_50	-
IPU_INT_STAT_13[0]	IDMAC	IDMAC_TH_0	-
IPU_INT_STAT_13[1]	IDMAC	IDMAC_TH_1	-
IPU_INT_STAT_13[2]	IDMAC	IDMAC_TH_2	-
IPU_INT_STAT_13[3]	IDMAC	IDMAC_TH_3	-
IPU_INT_STAT_13[5]	IDMAC	IDMAC_TH_5	-
IPU_INT_STAT_13[8]	IDMAC	IDMAC_TH_8	-
IPU_INT_STAT_13[9]	IDMAC	IDMAC_TH_9	-
IPU_INT_STAT_13[10]	IDMAC	IDMAC_TH_10	-

Table continues on the next page...

Table 37-35. Functional Interrupts Summary (continued)

Location	Sub-blocks	Interrupt status bit name	Description
IPU_INT_STAT_13[11]	IDMAC	IDMAC_TH_11	-
IPU_INT_STAT_13[12]	IDMAC	IDMAC_TH_12	-
IPU_INT_STAT_13[13]	IDMAC	IDMAC_TH_13	-
IPU_INT_STAT_13[14]	IDMAC	IDMAC_TH_14	-
IPU_INT_STAT_13[15]	IDMAC	IDMAC_TH_15	-
IPU_INT_STAT_13[17]	IDMAC	IDMAC_TH_17	-
IPU_INT_STAT_13[18]	IDMAC	IDMAC_TH_18	-
IPU_INT_STAT_13[20]	IDMAC	IDMAC_TH_20	-
IPU_INT_STAT_13[21]	IDMAC	IDMAC_TH_21	-
IPU_INT_STAT_13[22]	IDMAC	IDMAC_TH_22	-
IPU_INT_STAT_13[23]	IDMAC	IDMAC_TH_23	-
IPU_INT_STAT_13[24]	IDMAC	IDMAC_TH_24	-
IPU_INT_STAT_13[27]	IDMAC	IDMAC_TH_27	-
IPU_INT_STAT_13[28]	IDMAC	IDMAC_TH_28	-
IPU_INT_STAT_13[29]	IDMAC	IDMAC_TH_29	-
IPU_INT_STAT_13[31]	IDMAC	IDMAC_TH_31	-
IPU_INT_STAT_14[1]	IDMAC	IDMAC_TH_33	-
IPU_INT_STAT_14[8]	IDMAC	IDMAC_TH_40	-
IPU_INT_STAT_14[9]	IDMAC	IDMAC_TH_41	-
IPU_INT_STAT_14[10]	IDMAC	IDMAC_TH_42	-
IPU_INT_STAT_14[11]	IDMAC	IDMAC_TH_43	-
IPU_INT_STAT_14[12]	IDMAC	IDMAC_TH_44	-
IPU_INT_STAT_14[13]	IDMAC	IDMAC_TH_45	-
IPU_INT_STAT_14[14]	IDMAC	IDMAC_TH_46	-
IPU_INT_STAT_14[15]	IDMAC	IDMAC_TH_47	-
IPU_INT_STAT_14[16]	IDMAC	IDMAC_TH_48	-
IPU_INT_STAT_14[17]	IDMAC	IDMAC_TH_49	-
IPU_INT_STAT_14[18]	IDMAC	IDMAC_TH_50	-
IPU_INT_STAT_14[19]	IDMAC	IDMAC_TH_51	-
IPU_INT_STAT_14[20]	IDMAC	IDMAC_TH_52	-
IPU_INT_STAT_15[0]	CM	IPU_SNOOPING1_INT	-
IPU_INT_STAT_15[1]	CM	IPU_SNOOPING2_INT	-
IPU_INT_STAT_15[2]	DP	DP_SF_START	-
IPU_INT_STAT_15[3]	DP	DP_SF_END	-
IPU_INT_STAT_15[4]	DP	DP_ASF_START	-
IPU_INT_STAT_15[5]	DP	DP_ASF_END	-
IPU_INT_STAT_15[6]	DP	DP_SF_BRAKE	-
IPU_INT_STAT_15[7]	DP	DP_ASF_BRAKE	-
IPU_INT_STAT_15[8]	DC	DC_FC_0	-

Table continues on the next page...

Table 37-35. Functional Interrupts Summary (continued)

Location	Sub-blocks	Interrupt status bit name	Description
IPU_INT_STAT_15[9]	DC	DC_FC_1	-
IPU_INT_STAT_15[10]	DC	DC_FC_2	-
IPU_INT_STAT_15[11]	DC	DC_FC_3	-
IPU_INT_STAT_15[12]	DC	DC_FC_4	-
IPU_INT_STAT_15[13]	DC	DC_FC_6	-
IPU_INT_STAT_15[14]	DC	DI_VSYNC_PRE_0	-
IPU_INT_STAT_15[15]	DC	DI_VSYNC_PRE_1	-
IPU_INT_STAT_15[16]	DC	DC_DP_START	-
IPU_INT_STAT_15[17]	DC	DC_ASYNC_STOP	-
IPU_INT_STAT_15[18]	DI0	DI0_DISP_CLK_EN_PRE	-
IPU_INT_STAT_15[19]	DI0	DI0_CNT_EN_PRE_1	-
IPU_INT_STAT_15[20]	DI0	DI0_CNT_EN_PRE_2	-
IPU_INT_STAT_15[21]	DI0	DI0_CNT_EN_PRE_3	-
IPU_INT_STAT_15[22]	DI0	DI0_CNT_EN_PRE_4	-
IPU_INT_STAT_15[23]	DI0	DI0_CNT_EN_PRE_5	-
IPU_INT_STAT_15[24]	DI0	DI0_CNT_EN_PRE_6	-
IPU_INT_STAT_15[25]	DI0	DI0_CNT_EN_PRE_7	-
IPU_INT_STAT_15[26]	DI0	DI0_CNT_EN_PRE_8	-
IPU_INT_STAT_15[27]	DI0	DI0_CNT_EN_PRE_9	-
IPU_INT_STAT_15[28]	DI0	DI0_CNT_EN_PRE_10	-
IPU_INT_STAT_15[29]	DI1	DI1_DISP_CLK_EN_PRE	-
IPU_INT_STAT_15[30]	DI1	DI1_CNT_EN_PRE_3	-
IPU_INT_STAT_15[31]	DI1	DI1_CNT_EN_PRE_8	-

The table below describes the error interrupts. The panic column indicates if this signal is part of the logic generating the ipu_panic signal. The ipu_panic signal can be used for indicating about errors that are result of data rate problems. Such problems may be a result of the IPU running in slower clock then required by the use case. This signal can be used in order to indicate the system that the IPU can't handle the desired data rate. In that case the system may need to increase the clock to the IPU or simplify the use case.

Table 37-36. Error Interrupts Summary (continued)

Location	Sub-blocks	Interrupt Status name	panic	Description
IPU_INT_STAT_5[0]	IDMAC	IDMAC_NFB4EOF_ERR_0	YES	-
IPU_INT_STAT_5[1]	IDMAC	IDMAC_NFB4EOF_ERR_1	YES	-
IPU_INT_STAT_5[2]	IDMAC	IDMAC_NFB4EOF_ERR_2	YES	-
IPU_INT_STAT_5[3]	IDMAC	IDMAC_NFB4EOF_ERR_3	YES	-
IPU_INT_STAT_5[5]	IDMAC	IDMAC_NFB4EOF_ERR_5	YES	-

Table continues on the next page...

Table 37-36. Error Interrupts Summary (continued) (continued)

Location	Sub-blocks	Interrupt Status name	panic	Description
IPU_INT_STAT_5[8]	IDMAC	IDMAC_NFB4EOF_ERR_8	YES	-
IPU_INT_STAT_5[9]	IDMAC	IDMAC_NFB4EOF_ERR_9	YES	-
IPU_INT_STAT_5[10]	IDMAC	IDMAC_NFB4EOF_ERR_10	YES	-
IPU_INT_STAT_5[11]	IDMAC	IDMAC_NFB4EOF_ERR_11	YES	-
IPU_INT_STAT_5[12]	IDMAC	IDMAC_NFB4EOF_ERR_12	YES	-
IPU_INT_STAT_5[13]	IDMAC	IDMAC_NFB4EOF_ERR_13	YES	-
IPU_INT_STAT_5[14]	IDMAC	IDMAC_NFB4EOF_ERR_14	YES	-
IPU_INT_STAT_5[15]	IDMAC	IDMAC_NFB4EOF_ERR_15	YES	-
IPU_INT_STAT_5[17]	IDMAC	IDMAC_NFB4EOF_ERR_17	YES	-
IPU_INT_STAT_5[18]	IDMAC	IDMAC_NFB4EOF_ERR_18	YES	-
IPU_INT_STAT_5[20]	IDMAC	IDMAC_NFB4EOF_ERR_20	YES	-
IPU_INT_STAT_5[21]	IDMAC	IDMAC_NFB4EOF_ERR_21	YES	-
IPU_INT_STAT_5[22]	IDMAC	IDMAC_NFB4EOF_ERR_22	YES	-
IPU_INT_STAT_5[23]	IDMAC	IDMAC_NFB4EOF_ERR_23	YES	-
IPU_INT_STAT_5[24]	IDMAC	IDMAC_NFB4EOF_ERR_24	YES	-
IPU_INT_STAT_5[27]	IDMAC	IDMAC_NFB4EOF_ERR_27	YES	-
IPU_INT_STAT_5[28]	IDMAC	IDMAC_NFB4EOF_ERR_28	YES	-
IPU_INT_STAT_5[29]	IDMAC	IDMAC_NFB4EOF_ERR_29	YES	-
IPU_INT_STAT_5[31]	IDMAC	IDMAC_NFB4EOF_ERR_31	YES	-
IPU_INT_STAT_6[1]	IDMAC	IDMAC_NFB4EOF_ERR_33	YES	-
IPU_INT_STAT_6[8]	IDMAC	IDMAC_NFB4EOF_ERR_40	YES	-
IPU_INT_STAT_6[9]	IDMAC	IDMAC_NFB4EOF_ERR_41	YES	-
IPU_INT_STAT_6[10]	IDMAC	IDMAC_NFB4EOF_ERR_42	YES	-
IPU_INT_STAT_6[11]	IDMAC	IDMAC_NFB4EOF_ERR_43	YES	-
IPU_INT_STAT_6[12]	IDMAC	IDMAC_NFB4EOF_ERR_44	YES	-
IPU_INT_STAT_6[13]	IDMAC	IDMAC_NFB4EOF_ERR_45	YES	-
IPU_INT_STAT_6[14]	IDMAC	IDMAC_NFB4EOF_ERR_46	YES	-
IPU_INT_STAT_6[15]	IDMAC	IDMAC_NFB4EOF_ERR_47	YES	-
IPU_INT_STAT_6[16]	IDMAC	IDMAC_NFB4EOF_ERR_48	YES	-
IPU_INT_STAT_6[17]	IDMAC	IDMAC_NFB4EOF_ERR_49	YES	-
IPU_INT_STAT_6[18]	IDMAC	IDMAC_NFB4EOF_ERR_50	YES	-
IPU_INT_STAT_6[19]	IDMAC	IDMAC_NFB4EOF_ERR_51	YES	-
IPU_INT_STAT_6[20]	IDMAC	IDMAC_NFB4EOF_ERR_52	YES	-
IPU_INT_STAT_9[0]	IC	VDI_FIFO1_OVF	YES	-
IPU_INT_STAT_9[26]	IC	IC_BAYER_BUF_OVF	YES	-
IPU_INT_STAT_9[27]	IC	IC_ENC_BUF_OVF	YES	-
IPU_INT_STAT_9[28]	IC	IC_VF_BUF_OVF	YES	-
IPU_INT_STAT_9[30]	CSI0	CSI0_PUPE	YES	-
IPU_INT_STAT_9[31]	CSI0	CSI1_PUPE	YES	-

Table continues on the next page...

Table 37-36. Error Interrupts Summary (continued) (continued)

Location	Sub-blocks	Interrupt Status name	panic	Description
IPU_INT_STAT_10[0]	SMFC	SMFC0_FRM_LOST	YES	-
IPU_INT_STAT_10[1]	SMFC	SMFC1_FRM_LOST	YES	-
IPU_INT_STAT_10[2]	SMFC	SMFC2_FRM_LOST	YES	-
IPU_INT_STAT_10[3]	SMFC	SMFC3_FRM_LOST	YES	-
IPU_INT_STAT_10[16]	DC	DC_TEARING_ERR_1	YES	-
IPU_INT_STAT_10[17]	DC	DC_TEARING_ERR_2	YES	-
IPU_INT_STAT_10[18]	DC	DC_TEARING_ERR_6	YES	-
IPU_INT_STAT_10[19]	DI0	DI0_SYNC_DISP_ERR	YES	-
IPU_INT_STAT_10[20]	DI1	DI1_SYNC_DISP_ERR	YES	-
IPU_INT_STAT_10[21]	DI0	DI0_TIME_OUT_ERR	YES	-
IPU_INT_STAT_10[22]	DI1	DI1_TIME_OUT_ERR	YES	-
IPU_INT_STAT_10[24]	IC	IC_VF_FRM_LOST_ERR	YES	-
IPU_INT_STAT_10[25]	IC	IC_ENC_FRM_LOST_ERR	YES	-
IPU_INT_STAT_10[26]	IC	IC_BAYER_FRM_LOST_ERR	YES	-
IPU_INT_STAT_10[28]	CM	NON_PRIVILEGED_ACC_ERR	NO	-
IPU_INT_STAT_10[29]	IDMAC	AXIW_ERR	NO	-
IPU_INT_STAT_10[30]	IDMAC	AXIR_ERR	NO	-

37.4.12.4 SDMA event generator

The IPU provides an SDMA event signal that can be used as trigger to the SoC's SDMA. IPU routes an internal event to the SDMA event signal. The internal event causing the assertion of the SDMA signal is enabled by setting the corresponding bit on the SDMA_EVENT_# registers.

The user is allowed to enable multiple events. When one of these events occurs the ipu_sdma_event signal will be asserted. Similar to interrupts, the ipu_sdma_event signal is cleared by writing one to the corresponding bit in the INT_STAT_# registers.

It is not recommended to use the same internal event for a simultaneous generation of an interrupt signal and an SDMA event. This will require special software care when clearing the corresponding bit in the INT_STAT_# registers.

37.4.12.5 General Configuration Registers

The GCR contains a set of control/status/data registers. It provides IPU interface to the AHB slave bus.

The HCLK rate is equal to the HSP_CLK rate. The detail description of the registers is found in the Programmable Registers section.

37.4.12.6 Shadow Registers Module (SRM)

IPU supports frame by frame task switching. This means that a sub-block can handle a frame with one configuration and handle the following frame with different configuration. Changing the configuration is done by updating the sub-block's parameters.

In order to allow automatic flow without a need of the SW to update all the parameters at the frame boundaries. A Register of a sub-block that has the shadowing capabilities has a shadow register file that resides in the Shadow Register Memory.

The sub-blocks supporting this function may use it in one of the following ways:

37.4.12.6.1 Switching between 2 flows

Upon request from the FSU the SRM switches between the registers and the content in the Shadow Register Memory. When a sub-block uses one of the configuration SW, it is allowed to update the parameters in the memory. This is normally used when 2 flows are supported via one module.

- The current flow's configuration is stored in the module's registers
- The alternate flow configuration is stored in the SRM.
- When switching between flows the current flow's configuration is stored in the SRM. The alternate flow's configuration is written to the module's registers. When the alternate flow ends the configurations are swapped again.

This process is fully controlled by the FSU

37.4.12.6.2 Updating parameters between frames

This mode is used when the user needs to update the parameters of the current task being processed by the module. The updates can't affect the frame that is currently being processed. The update is effective only on the next frame. The SRM performs the parameters update only when there's a specific request by the user.

- The current frame's configuration is stored in the sub-block's registers
- The next frame's configuration is stored in the SRM.
- On frame's boundary the SRM reads the new configuration from the SRM and writes it to the sub-block's registers. The old configuration is lost.

37.4.12.6.3 Updating the memory

In order to avoid data coherency problems, the user should set a flag indicating that he is currently updating the memory region of a specific sub-block. When the flag is set the SRM will not attempt to replace the parameters relevant to the specific sub-block currently being updated by the user.

The user should clear this flag when he completes the update, and the parameters are ready to be used.

The flag is the corresponding **SRM_MODE** field for each sub-block. This field controls the SRM logic that handles the sub-blocks registers

- 00 - ARM platform is allowed to access the sub-block's region in the RAM; The automatic swapping mechanism is disabled.
- 01 - The SRM logic is controlled by the FSU. The update will be done of the next frame.
- 10 - The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame
- 11 - Update now. The SRM is controlled by the ARM platform. The Register will be update now

Each sub-block uses the SRM mechanism according to the sub-block's behavior, the table below summarizes the SRM support for each sub-block

Table 37-37. SRM support per Sub-block

Sub-block	Switching between 2 flows support	Updating parameters between frames	Comment
CSI1, CSI0	NO	YES	SRM_MODE can be 00 or 01. The update will happen only once. When set to 01: after the update the state machine is automatically moved to 00 mode. It is recommended to make sure that the first frame has started by polling the corresponding NFACK bit before setting the SRM_MODE
DI1, DI0	NO	YES	DI0, DI1 parameters are needed when clock change is performed (Clock Change procedure). SRM_MODE can be 00 or 01. The update will happen only once. When set to 01: after the update the state machine is automatically moved to 00 mode
DC	YES	NO	SRM_MODE can be 00 or 10.
DP	YES	NO	SRM_MODE can be 00, 10 or 11. When set to 11: after the update the state machine is automatically moved to 10 mode

Table 37-37. SRM support per Sub-block

Sub-block	Switching between 2 flows support	Updating parameters between frames	Comment
			10 is not supported for SYNC flows

In order to update parameters the user should monitor the SRM_BUSY bit of the corresponding sub-block. When the SRM is not busy the user should set the **SRM_MODE to 00. The user will now update the register file in the memory. When done the user should switch the SRM_MODE field to the desired mode.**

37.4.12.6.4 SRM priority

The SRM updates the registers according to a pre defined priority. The priority is set according to the SRM_PRI bits of each sub-block. The user must set a unique value for each sub-block.

37.4.12.6.5 SRM entries mapping

The table below maps any IPU register to an address in the SRM. The registers marked as NONE do not have an SRM entry

PG column indicates if this register is saved during power gating mode

LPSR column indicates if this register is swapped during low power screen refresh mode (LPSR)

Table 37-38. IPU SRM entries mapping

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CONF	0x0000_0000	NONE	YES	NO
IPU_SISG_CTRL0	0x00000004	NONE	YES	NO
IPU_SISG_CTRL1	0x00000008	NONE	YES	NO
IPU_SISG_SET_1	0x0000000C	NONE	YES	NO
IPU_SISG_SET_2	0x00000010	NONE	YES	NO
IPU_SISG_SET_3	0x00000014	NONE	YES	NO
IPU_SISG_SET_4	0x00000018	NONE	YES	NO
IPU_SISG_SET_5	0x0000001C	NONE	YES	NO
IPU_SISG_SET_6	0x00000020	NONE	YES	NO
IPU_SISG_CLR_1	0x00000024	NONE	YES	NO
IPU_SISG_CLR_2	0x00000028	NONE	YES	NO
IPU_SISG_CLR_3	0x0000002C	NONE	YES	NO
IPU_SISG_CLR_4	0x00000030	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_SISG_CLR_5	0x00000034	NONE	YES	NO
IPU_SISG_CLR_6	0x00000038	NONE	YES	NO
IPU_INT_CTRL_1	0x0000003C	NONE	YES	NO
IPU_INT_CTRL_2	0x00000040	NONE	YES	NO
IPU_INT_CTRL_3	0x00000044	NONE	YES	NO
IPU_INT_CTRL_4	0x00000048	NONE	YES	NO
IPU_INT_CTRL_5	0x0000004C	NONE	YES	NO
IPU_INT_CTRL_6	0x00000050	NONE	YES	NO
IPU_INT_CTRL_7	0x00000054	NONE	YES	NO
IPU_INT_CTRL_8	0x00000058	NONE	YES	NO
IPU_INT_CTRL_9	0x0000005C	NONE	YES	NO
IPU_INT_CTRL_10	0x00000060	NONE	YES	NO
IPU_INT_CTRL_11	0x00000064	NONE	YES	NO
IPU_INT_CTRL_12	0x00000068	NONE	YES	NO
IPU_INT_CTRL_13	0x0000006C	NONE	YES	NO
IPU_INT_CTRL_14	0x00000070	NONE	YES	NO
IPU_INT_CTRL_15	0x00000074	NONE	YES	NO
IPU_SDMA_EVENT_1	0x00000078	NONE	YES	NO
IPU_SDMA_EVENT_2	0x0000007C	NONE	YES	NO
IPU_SDMA_EVENT_3	0x00000080	NONE	YES	NO
IPU_SDMA_EVENT_4	0x00000084	NONE	YES	NO
IPU_SDMA_EVENT_7	0x00000088	NONE	YES	NO
IPU_SDMA_EVENT_8	0x0000008C	NONE	YES	NO
IPU_SDMA_EVENT_11	0x00000090	NONE	YES	NO
IPU_SDMA_EVENT_12	0x00000094	NONE	YES	NO
IPU_SDMA_EVENT_13	0x00000098	NONE	YES	NO
IPU_SDMA_EVENT_14	0x0000009C	NONE	YES	NO
IPU_SRM_PRI1	0x000000A0	NONE	YES	NO
IPU_SRM_PRI2	0x000000A4	NONE	YES	NO
IPU_FS_PROC_FLOW1	0x000000A8	NONE	YES	NO
IPU_FS_PROC_FLOW2	0x000000AC	NONE	YES	NO
IPU_FS_PROC_FLOW3	0x000000B0	NONE	YES	NO
IPU_FS_DISP_FLOW1	0x000000B4	NONE	YES	NO
IPU_FS_DISP_FLOW2	0x000000B8	NONE	YES	NO
IPU_SKIP	0x000000BC	NONE	YES	NO
IPU_DISP_ALT_CONF	0x000000C0	NONE	YES	NO
IPU_DISP_GEN	0x000000C4	NONE	YES	NO
IPU_DISP_ALT1	0x000000C8	NONE	YES	NO
IPU_DISP_ALT2	0x000000CC	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DISP_ALT3	0x000000D0	NONE	YES	NO
IPU_DISP_ALT4	0x000000D4	NONE	YES	NO
IPU_SNOOP	0x000000D8	NONE	YES	NO
IPU_MEM_RST	0x000000DC	NONE	YES	NO
IPU_PM	0x000000E0	NONE	YES	NO
IPU_GPR	0x000000E4	NONE	YES	NO
IPU_CH_DB_MODE_SEL_0	0x00000150	NONE	YES	NO
IPU_CH_DB_MODE_SEL_1	0x00000154	NONE	YES	NO
IPU_ALT_CH_DB_MODE_SEL_0	0x00000168	NONE	YES	NO
IPU_ALT_CH_DB_MODE_SEL_1	0x0000016C	NONE	YES	NO
IPU_CH_TRB_MODE_SEL_0	0x00000178	NONE	YES	NO
IPU_CH_TRB_MODE_SEL_1	0x0000017C	NONE	YES	NO
IPU_INT_STAT_1	0x00000200	NONE	NO	NO
IPU_INT_STAT_2	0x00000204	NONE	NO	NO
IPU_INT_STAT_3	0x00000208	NONE	NO	NO
IPU_INT_STAT_4	0x0000020C	NONE	NO	NO
IPU_INT_STAT_5	0x00000210	NONE	NO	NO
IPU_INT_STAT_6	0x00000214	NONE	NO	NO
IPU_INT_STAT_7	0x00000218	NONE	NO	NO
IPU_INT_STAT_8	0x0000021C	NONE	NO	NO
IPU_INT_STAT_9	0x00000220	NONE	NO	NO
IPU_INT_STAT_10	0x00000224	NONE	NO	NO
IPU_INT_STAT_11	0x00000228	NONE	NO	NO
IPU_INT_STAT_12	0x0000022C	NONE	NO	NO
IPU_INT_STAT_13	0x00000230	NONE	NO	NO
IPU_INT_STAT_14	0x00000234	NONE	NO	NO
IPU_INT_STAT_15	0x00000238	NONE	NO	NO
IPU_CUR_BUF_0	0x0000023C	NONE	NO	NO
IPU_CUR_BUF_1	0x00000240	NONE	NO	NO
IPU_ALT_CUR_BUF_0	0x00000244	NONE	NO	NO
IPU_ALT_CUR_BUF_1	0x00000248	NONE	NO	NO
IPU_SRM_STAT	0x0000024C	NONE	NO	NO
IPU_PROC_TASKS_STAT	0x00000250	NONE	NO	NO
IPU_DISP_TASKS_STAT	0x00000254	NONE	NO	NO
IPU_TRIPLE_CUR_BUF_0	0x00000258	NONE	NO	NO
IPU_TRIPLE_CUR_BUF_1	0x0000025C	NONE	NO	NO
IPU_TRIPLE_CUR_BUF_2	0x00000260	NONE	NO	NO
IPU_TRIPLE_CUR_BUF_3	0x00000264	NONE	NO	NO
IPU_CH_BUF0_RDY0	0x00000268	NONE	NO	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CH_BUF0_RDY1	0x0000026C	NONE	NO	NO
IPU_CH_BUF1_RDY0	0x00000270	NONE	NO	NO
IPU_CH_BUF1_RDY1	0x00000274	NONE	NO	NO
IPU_ALT_CH_BUF0_RDY0	0x00000278	NONE	NO	NO
IPU_ALT_CH_BUF0_RDY1	0x0000027C	NONE	NO	NO
IPU_ALT_CH_BUF1_RDY0	0x00000280	NONE	NO	NO
IPU_ALT_CH_BUF1_RDY1	0x00000284	NONE	NO	NO
IPU_CH_BUF2_RDY0	0x00000288	NONE	NO	NO
IPU_CH_BUF2_RDY1	0x0000028C	NONE	NO	NO
IPU_IDMAC_CONF	0x00008000	NONE	YES	NO
IPU_IDMAC_CH_EN_1	0x00008004	NONE	YES	NO
IPU_IDMAC_CH_EN_2	0x00008008	NONE	YES	NO
IPU_IDMAC_SEP_ALPHA	0x0000800C	NONE	YES	NO
IPU_IDMAC_ALT_SEP_ALPHA	0x00008010	NONE	YES	NO
IPU_IDMAC_CH_PRI_1	0x00008014	NONE	YES	NO
IPU_IDMAC_CH_PRI_2	0x00008018	NONE	YES	NO
IIPU_DMAC_WM_EN_1	0x0000801C	NONE	YES	NO
IPU_IDMAC_WM_EN_2	0x00008020	NONE	YES	NO
IPU_IDMAC_LOCK_EN_1	0x00008024	NONE	YES	NO
IPU_IDMAC_LOCK_EN_2	0x00008028	NONE	YES	NO
IPU_IDMAC_SUB_ADDR_0	0x0000802C	NONE	YES	NO
IPU_IDMAC_SUB_ADDR_1	0x00008030	NONE	YES	NO
IPU_IDMAC_SUB_ADDR_2	0x00008034	NONE	YES	NO
IPU_IDMAC_SUB_ADDR_3	0x00008038	NONE	YES	NO
IPU_IDMAC_SUB_ADDR_4	0x0000803C	NONE	YES	NO
IPU_IDMAC_BNDM_EN_1	0x00008040	NONE	YES	NO
IPU_IDMAC_BNDM_EN_2	0x00008044	NONE	YES	NO
IPU_IDMAC_SC_CORD	0x00008048	NONE	YES	NO
IPU_IDMAC_SC_CORD1	0x0000804C	NONE	YES	NO
IPU_IDMAC_CH_BUSY_1	0x00008100	NONE	NO	NO
IPU_IDMAC_CH_BUSY_2	0x00008104	NONE	NO	NO
IPU_DP_COM_CONF_SYNC	0x1F40000	0x1F40000	YES	YES
IPU_DP_GRAPH_WIND_CTRL_SYNC	0x1F40004	0x1F40004	YES	YES
IPU_DP_FG_POS_SYNC	0x1F40008	0x1F40008	YES	YES
IPU_DP_CUR_POS_SYNC	0x1F4000C	0x1F4000C	YES	YES
IPU_DP_CUR_MAP_SYNC	0x1F40010	0x1F40010	YES	YES
IPU_DP_GAMMA_C_SYNC_0	0x1F40014	0x1F40014	YES	YES
IPU_DP_GAMMA_C_SYNC_1	0x1F40018	0x1F40018	YES	YES
IPU_DP_GAMMA_C_SYNC_2	0x1F4001C	0x1F4001C	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DP_GAMMA_C_SYNC_3	0x1F40020	0x1F40020	YES	YES
IPU_DP_GAMMA_C_SYNC_4	0x1F40024	0x1F40024	YES	YES
IPU_DP_GAMMA_C_SYNC_5	0x1F40028	0x1F40028	YES	YES
IPU_DP_GAMMA_C_SYNC_6	0x1F4002C	0x1F4002C	YES	YES
IPU_DP_GAMMA_C_SYNC_7	0x1F40030	0x1F40030	YES	YES
IPU_DP_GAMMA_S_SYNC_0	0x1F40034	0x1F40034	YES	YES
IPU_DP_GAMMA_S_SYNC_1	0x1F40038	0x1F40038	YES	YES
IPU_DP_GAMMA_S_SYNC_2	0x1F4003C	0x1F4003C	YES	YES
IPU_DP_GAMMA_S_SYNC_3	0x1F40040	0x1F40040	YES	YES
IPU_DP_CSCA_SYNC_0	0x1F40044	0x1F40044	YES	YES
IPU_DP_CSCA_SYNC_1	0x1F40048	0x1F40048	YES	YES
IPU_DP_CSCA_SYNC_2	0x1F4004C	0x1F4004C	YES	YES
IPU_DP_CSCA_SYNC_3	0x1F40050	0x1F40050	YES	YES
IPU_DP_CSC_SYNC_0	0x1F40054	0x1F40054	YES	YES
IPU_DP_CSC_SYNC_1	0x1F40058	0x1F40058	YES	YES
IPU_DP_CUR_POS_ALT	0x1F4005C	0x1F4005C	YES	YES
IPU_DP_COM_CONF_ASYNC0	0x1F40060	0x1F40060	YES	YES
IPU_DP_GRAPH_WIND_CTRL_ASYNC0	0x1F40064	0x1F40064	YES	YES
IPU_DP_FG_POS_ASYNC0	0x1F40068	0x1F40068	YES	YES
IPU_DP_CUR_POS_ASYNC0	0x1F4006C	0x1F4006C	YES	YES
IPU_DP_CUR_MAP_ASYNC0	0x1F40070	0x1F40070	YES	YES
IPU_DP_GAMMA_C_ASYNC0_0	0x1F40074	0x1F40074	YES	YES
IPU_DP_GAMMA_C_ASYNC0_1	0x1F40078	0x1F40078	YES	YES
IPU_DP_GAMMA_C_ASYNC0_2	0x1F4007C	0x1F4007C	YES	YES
IPU_DP_GAMMA_C_ASYNC0_3	0x1F40080	0x1F40080	YES	YES
IPU_DP_GAMMA_C_ASYNC0_4	0x1F40084	0x1F40084	YES	YES
IPU_DP_GAMMA_C_ASYNC0_5	0x1F40088	0x1F40088	YES	YES
IPU_DP_GAMMA_C_ASYNC0_6	0x1F4008C	0x1F4008C	YES	YES
IPU_DP_GAMMA_C_ASYNC0_7	0x1F40090	0x1F40090	YES	YES
IPU_DP_GAMMA_S_ASYNC0_0	0x1F40094	0x1F40094	YES	YES
IPU_DP_GAMMA_S_ASYNC0_1	0x1F40098	0x1F40098	YES	YES
IPU_DP_GAMMA_S_ASYNC0_2	0x1F4009C	0x1F4009C	YES	YES
IPU_DP_GAMMA_S_ASYNC0_3	0x1F400A0	0x1F400A0	YES	YES
IPU_DP_CSCA_ASYNC0_0	0x1F400A4	0x1F400A4	YES	YES
IPU_DP_CSCA_ASYNC0_1	0x1F400A8	0x1F400A8	YES	YES
IPU_DP_CSCA_ASYNC0_2	0x1F400AC	0x1F400AC	YES	YES
IPU_DP_CSCA_ASYNC0_3	0x1F400B0	0x1F400B0	YES	YES
IPU_DP_CSC_ASYNC0_0	0x1F400B4	0x1F400B4	YES	YES
IPU_DP_CSC_ASYNC0_1	0x1F400B8	0x1F400B8	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DP_COM_CONF_ASYNC1	0x1F400BC	0x1F400BC	YES	YES
IPU_DP_GRAPH_WIND_CTRL_ASYNC1	0x1F400C0	0x1F400C0	YES	YES
IPU_DP_FG_POS_ASYNC1	0x1F400C4	0x1F400C4	YES	YES
IPU_DP_CUR_POS_ASYNC1	0x1F400C8	0x1F400C8	YES	YES
IPU_DP_CUR_MAP_ASYNC1	0x1F400CC	0x1F400CC	YES	YES
IPU_DP_GAMMA_C_ASYNC1_0	0x1F400D0	0x1F400D0	YES	YES
IPU_DP_GAMMA_C_ASYNC1_1	0x1F400D4	0x1F400D4	YES	YES
IPU_DP_GAMMA_C_ASYNC1_2	0x1F400D8	0x1F400D8	YES	YES
IPU_DP_GAMMA_C_ASYNC1_3	0x1F400DC	0x1F400DC	YES	YES
IPU_DP_GAMMA_C_ASYNC1_4	0x1F400E0	0x1F400E0	YES	YES
IPU_DP_GAMMA_C_ASYNC1_5	0x1F400E4	0x1F400E4	YES	YES
IPU_DP_GAMMA_C_ASYNC1_6	0x1F400E8	0x1F400E8	YES	YES
IPU_DP_GAMMA_C_ASYNC1_7	0x1F400EC	0x1F400EC	YES	YES
IPU_DP_GAMMA_S_ASYNC1_0	0x1F400F0	0x1F400F0	YES	YES
IPU_DP_GAMMA_S_ASYNC1_1	0x1F400F4	0x1F400F4	YES	YES
IPU_DP_GAMMA_S_ASYNC1_2	0x1F400F8	0x1F400F8	YES	YES
IPU_DP_GAMMA_S_ASYNC1_3	0x1F400FC	0x1F400FC	YES	YES
IPU_DP_CSCA_ASYNC1_0	0x1F40100	0x1F40100	YES	YES
IPU_DP_CSCA_ASYNC1_1	0x1F40104	0x1F40104	YES	YES
IPU_DP_CSCA_ASYNC1_2	0x1F40108	0x1F40108	YES	YES
IPU_DP_CSCA_ASYNC1_3	0x1F4010C	0x1F4010C	YES	YES
IPU_DP_CSC_ASYNC1_0	0x1F40110	0x1F40110	YES	YES
IPU_DP_CSC_ASYNC1_1	0x1F40114	0x1F40114	YES	YES
IPU_DP_DEBUG_CNT	0x000180BC	NONE	NO	NO
IPU_DP_DEBUG_STAT	0x000180C0	NONE	NO	NO
IPU_IC_CONF	0x00020000	NONE	YES	NO
IPU_IC_PRP_ENC_RSC	0x00020004	NONE	YES	NO
IPU_IC_PRP_VF_RSC	0x00020008	NONE	YES	NO
IPU_IC_PP_RSC	0x0002000C	NONE	YES	NO
IPU_IC_CMBP_1	0x00020010	NONE	YES	NO
IPU_IC_CMBP_2	0x00020014	NONE	YES	NO
IPU_IC_IDMAC_1	0x00020018	NONE	YES	NO
IPU_IC_IDMAC_2	0x0002001C	NONE	YES	NO
IPU_IC_IDMAC_3	0x00020020	NONE	YES	NO
IPU_IC_IDMAC_4	0x00020024	NONE	YES	NO
IPU_CSI0_SENS_CONF	0x00030000	NONE	YES	NO
IPU_CSI0_SENS_FRM_SIZE	0x00030004	NONE	YES	NO
IPU_CSI0_ACT_FRM_SIZE	0x00030008	NONE	YES	NO
IPU_CSI0_OUT_FRM_CTRL	0x0003000C	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CSI0_TST_CTRL	0x00030010	NONE	YES	NO
IPU_CSI0_CCIR_CODE_1	0x00030014	NONE	YES	NO
IPU_CSI0_CCIR_CODE_2	0x00030018	NONE	YES	NO
IPU_CSI0_CCIR_CODE_3	0x0003001C	NONE	YES	NO
IPU_CSI0_DI	0x00030020	NONE	YES	NO
IPU_CSI0_SKIP	0x00030024	NONE	YES	NO
IPU_CSI0_CPD_CTRL	0x00030028	0x1F40314	YES	NO
IPU_CSI0_CPD_RC_0	0x0003002C	0x1F40318	YES	NO
IPU_CSI0_CPD_RC_1	0x00030030	0x1F4031C	YES	NO
IPU_CSI0_CPD_RC_2	0x00030034	0x1F40320	YES	NO
IPU_CSI0_CPD_RC_3	0x00030038	0x1F40324	YES	NO
IPU_CSI0_CPD_RC_4	0x0003003C	0x1F40328	YES	NO
IPU_CSI0_CPD_RC_5	0x00030040	0x1F4032C	YES	NO
IPU_CSI0_CPD_RC_6	0x00030044	0x1F40330	YES	NO
IPU_CSI0_CPD_RC_7	0x00030048	0x1F40334	YES	NO
IPU_CSI0_CPD_RS_0	0x0003004C	0x1F40338	YES	NO
IPU_CSI0_CPD_RS_1	0x00030050	0x1F4033C	YES	NO
IPU_CSI0_CPD_RS_2	0x00030054	0x1F40340	YES	NO
IPU_CSI0_CPD_RS_3	0x00030058	0x1F40344	YES	NO
IPU_CSI0_CPD_GRC_0	0x0003005C	0x1F40348	YES	NO
IPU_CSI0_CPD_GRC_1	0x00030060	0x1F4034C	YES	NO
IPU_CSI0_CPD_GRC_2	0x00030064	0x1F40350	YES	NO
IPU_CSI0_CPD_GRC_3	0x00030068	0x1F40354	YES	NO
IPU_CSI0_CPD_GRC_4	0x0003006C	0x1F40358	YES	NO
IPU_CSI0_CPD_GRC_5	0x00030070	0x1F4035C	YES	NO
IPU_CSI0_CPD_GRC_6	0x00030074	0x1F40360	YES	NO
IPU_CSI0_CPD_GRC_7	0x00030078	0x1F40364	YES	NO
IPU_CSI0_CPD_GRS_0	0x0003007C	0x1F40368	YES	NO
IPU_CSI0_CPD_GRS_1	0x00030080	0x1F4036C	YES	NO
IPU_CSI0_CPD_GRS_2	0x00030084	0x1F40370	YES	NO
IPU_CSI0_CPD_GRS_3	0x00030088	0x1F40374	YES	NO
IPU_CSI0_CPD_GBC_0	0x0003008C	0x1F40378	YES	NO
IPU_CSI0_CPD_GBC_1	0x00030090	0x1F4037C	YES	NO
IPU_CSI0_CPD_GBC_2	0x00030094	0x1F40380	YES	NO
IPU_CSI0_CPD_GBC_3	0x00030098	0x1F40384	YES	NO
IPU_CSI0_CPD_GBC_4	0x0003009C	0x1F40388	YES	NO
IPU_CSI0_CPD_GBC_5	0x000300A0	0x1F4038C	YES	NO
IPU_CSI0_CPD_GBC_6	0x000300A4	0x1F40390	YES	NO
IPU_CSI0_CPD_GBC_7	0x000300A8	0x1F40394	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CSI0_CPD_GBS_0	0x000300AC	0x1F40398	YES	NO
IPU_CSI0_CPD_GBS_1	0x000300B0	0x1F4039C	YES	NO
IPU_CSI0_CPD_GBS_2	0x000300B4	0x1F403A0	YES	NO
IPU_CSI0_CPD_GBS_3	0x000300B8	0x1F403A4	YES	NO
IPU_CSI0_CPD_BC_0	0x000300BC	0x1F403A8	YES	NO
IPU_CSI0_CPD_BC_1	0x000300C0	0x1F403AC	YES	NO
IPU_CSI0_CPD_BC_2	0x000300C4	0x1F403B0	YES	NO
IPU_CSI0_CPD_BC_3	0x000300C8	0x1F403B4	YES	NO
IPU_IPU_CSI0_CPD_BC_4	0x000300CC	0x1F403B8	YES	NO
IPU_CSI0_CPD_BC_5	0x000300D0	0x1F403BC	YES	NO
IPU_CSI0_CPD_BC_6	0x000300D4	0x1F403C0	YES	NO
IPU_CSI0_CPD_BC_7	0x000300D8	0x1F403C4	YES	NO
IPU_CSI0_CPD_BS_0	0x000300DC	0x1F403C8	YES	NO
IPU_CSI0_CPD_BS_1	0x000300E0	0x1F403CC	YES	NO
IPU_CSI0_CPD_BS_2	0x000300E4	0x1F403D0	YES	NO
IPU_CSI0_CPD_BS_3	0x000300E8	0x1F403D4	YES	NO
IPU_CSI0_CPD_OFFSET1	0x000300EC	0x1F403D8	YES	NO
IPU_CSI0_CPD_OFFSET2	0x000300F0	0x1F403DC	YES	NO
IPU_CSI1_SENS_CONF	0x00038000	NONE	YES	NO
IPU_CSI1_SENS_FRM_SIZE	0x00038004	NONE	YES	NO
IPU_CSI1_ACT_FRM_SIZE	0x00038008	NONE	YES	NO
IPU_CSI1_OUT_FRM_CTRL	0x0003800C	NONE	YES	NO
IPU_CSI1_TST_CTRL	0x00038010	NONE	YES	NO
IPU_CSI1_CCIR_CODE_1	0x00038014	NONE	YES	NO
IPU_CSI1_CCIR_CODE_2	0x00038018	NONE	YES	NO
IPU_CSI1_CCIR_CODE_3	0x0003801C	NONE	YES	NO
IPU_CSI1_DI	0x00038020	NONE	YES	NO
IPU_CSI1_SKIP	0x00038024	NONE	YES	NO
IPU_CSI1_CPD_CTRL	0x00038028	0x1F403E0	YES	NO
IPU_CSI1_CPD_RC_0	0x0003802C	0x1F403E4	YES	NO
IPU_CSI1_CPD_RC_1	0x00038030	0x1F403E8	YES	NO
IPU_CSI1_CPD_RC_2	0x00038034	0x1F403EC	YES	NO
IPU_CSI1_CPD_RC_3	0x00038038	0x1F403F0	YES	NO
IPU_CSI1_CPD_RC_4	0x0003803C	0x1F403F4	YES	NO
IPU_CSI1_CPD_RC_5	0x00038040	0x1F403F8	YES	NO
IPU_CSI1_CPD_RC_6	0x00038044	0x1F403FC	YES	NO
IPU_CSI1_CPD_RC_7	0x00038048	0x1F40400	YES	NO
IPU_CSI1_CPD_RS_0	0x0003804C	0x1F40404	YES	NO
IPU_CSI1_CPD_RS_1	0x00038050	0x1F40408	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CSI1_CPD_RS_2	0x00038054	0x1F4040C	YES	NO
IPU_CSI1_CPD_RS_3	0x00038058	0x1F40410	YES	NO
IPU_CSI1_CPD_GRC_0	0x0003805C	0x1F40414	YES	NO
IPU_CSI1_CPD_GRC_1	0x00038060	0x1F40418	YES	NO
IPU_CSI1_CPD_GRC_2	0x00038064	0x1F4041C	YES	NO
IPU_CSI1_CPD_GRC_3	0x00038068	0x1F40420	YES	NO
IPU_CSI1_CPD_GRC_4	0x0003806C	0x1F40424	YES	NO
IPU_CSI1_CPD_GRC_5	0x00038070	0x1F40428	YES	NO
IPU_CSI1_CPD_GRC_6	0x00038074	0x1F4042C	YES	NO
IPU_CSI1_CPD_GRC_7	0x00038078	0x1F40430	YES	NO
IPU_CSI1_CPD_GRS_0	0x0003807C	0x1F40434	YES	NO
IPU_CSI1_CPD_GRS_1	0x00038080	0x1F40438	YES	NO
IPU_CSI1_CPD_GRS_2	0x00038084	0x1F4043C	YES	NO
IPU_CSI1_CPD_GRS_3	0x00038088	0x1F40440	YES	NO
IPU_CSI1_CPD_GBC_0	0x0003808C	0x1F40444	YES	NO
IPU_CSI1_CPD_GBC_1	0x00038090	0x1F40448	YES	NO
IPU_CSI1_CPD_GBC_2	0x00038094	0x1F4044C	YES	NO
IPU_CSI1_CPD_GBC_3	0x00038098	0x1F40450	YES	NO
IPU_CSI1_CPD_GBC_4	0x0003809C	0x1F40454	YES	NO
IPU_CSI1_CPD_GBC_5	0x000380A0	0x1F40458	YES	NO
IPU_CSI1_CPD_GBC_6	0x000380A4	0x1F4045C	YES	NO
IPU_CSI1_CPD_GBC_7	0x000380A8	0x1F40460	YES	NO
IPU_CSI1_CPD_GBS_0	0x000380AC	0x1F40464	YES	NO
IPU_CSI1_CPD_GBS_1	0x000380B0	0x1F40468	YES	NO
IPU_CSI1_CPD_GBS_2	0x000380B4	0x1F4046C	YES	NO
IPU_CSI1_CPD_GBS_3	0x000380B8	0x1F40470	YES	NO
IPU_CSI1_CPD_BC_0	0x000380BC	0x1F40474	YES	NO
IPU_CSI1_CPD_BC_1	0x000380C0	0x1F40478	YES	NO
IPU_CSI1_CPD_BC_2	0x000380C4	0x1F4047C	YES	NO
IPU_CSI1_CPD_BC_3	0x000380C8	0x1F40480	YES	NO
IPU_CSI1_CPD_BC_4	0x000380CC	0x1F40484	YES	NO
IPU_CSI1_CPD_BC_5	0x000380D0	0x1F40488	YES	NO
IPU_CSI1_CPD_BC_6	0x000380D4	0x1F4048C	YES	NO
IPU_CSI1_CPD_BC_7	0x000380D8	0x1F40490	YES	NO
IPU_CSI1_CPD_BS_0	0x000380DC	0x1F40494	YES	NO
IPU_CSI1_CPD_BS_1	0x000380E0	0x1F40498	YES	NO
IPU_CSI1_CPD_BS_2	0x000380E4	0x1F4049C	YES	NO
IPU_CSI1_CPD_BS_3	0x000380E8	0x1F404A0	YES	NO
IPU_CSI1_CPD_OFFSET1	0x000380EC	0x1F404A4	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_CSI1_CPD_OFFSET2	0x000380F0	0x1F404A8	YES	NO
IPU_DI0_GENERAL	0x00040000	0x1F404E4	YES	YES
IPU_DI0_BS_CLKGEN0	0x00040004	0x1F404E8	YES	YES
IPU_DI0_BS_CLKGEN1	0x00040008	0x1F404EC	YES	YES
IPU_DI0_SW_GEN0_1	0x0004000C	0x1F404F0	YES	YES
IPU_DI0_SW_GEN0_2	0x00040010	0x1F404F4	YES	YES
IPU_DI0_SW_GEN0_3	0x00040014	0x1F404F8	YES	YES
IPU_DI0_SW_GEN0_4	0x00040018	0x1F404FC	YES	YES
IPU_DI0_SW_GEN0_5	0x0004001C	0x1F40500	YES	YES
IPU_DI0_SW_GEN0_6	0x00040020	0x1F40504	YES	YES
IPU_DI0_SW_GEN0_7	0x00040024	0x1F40508	YES	YES
IPU_DI0_SW_GEN0_8	0x00040028	0x1F4050C	YES	YES
IPU_DI0_SW_GEN0_9	0x0004002C	0x1F40510	YES	YES
IPU_DI0_SW_GEN1_1	0x00040030	0x1F40514	YES	YES
IPU_DI0_SW_GEN1_2	0x00040034	0x1F40518	YES	YES
IPU_DI0_SW_GEN1_3	0x00040038	0x1F4051C	YES	YES
IPU_DI0_SW_GEN1_4	0x0004003C	0x1F40520	YES	YES
IPU_DI0_SW_GEN1_5	0x00040040	0x1F40524	YES	YES
IPU_DI0_SW_GEN1_6	0x00040044	0x1F40528	YES	YES
IPU_DI0_SW_GEN1_7	0x00040048	0x1F4052C	YES	YES
IPU_DI0_SW_GEN1_8	0x0004004C	0x1F40530	YES	YES
IPU_DI0_SW_GEN1_9	0x00040050	0x1F40534	YES	YES
IPU_DI0_SYNC_AS_GEN	0x00040054	0x1F40538	YES	YES
IPU_DI0_DW_GEN_0	0x00040058	0x1F4053C	YES	YES
IPU_DI0_DW_GEN_1	0x0004005C	0x1F40540	YES	YES
IPU_DI0_DW_GEN_2	0x00040060	0x1F40544	YES	YES
IPU_DI0_DW_GEN_3	0x00040064	0x1F40548	YES	YES
IPU_DI0_DW_GEN_4	0x00040068	0x1F4054C	YES	YES
IPU_DI0_DW_GEN_5	0x0004006C	0x1F40550	YES	YES
IPU_DI0_DW_GEN_6	0x00040070	0x1F40554	YES	YES
IPU_DI0_DW_GEN_7	0x00040074	0x1F40558	YES	YES
IPU_DI0_DW_GEN_8	0x00040078	0x1F4055C	YES	YES
IPU_DI0_DW_GEN_9	0x0004007C	0x1F40560	YES	YES
IPU_DI0_DW_GEN_10	0x00040080	0x1F40564	YES	YES
IPU_DI0_DW_GEN_11	0x00040084	0x1F40568	YES	YES
IPU_DI0_DW_SET0_0	0x00040088	0x1F4056C	YES	YES
IPU_DI0_DW_SET0_1	0x0004008C	0x1F40570	YES	YES
IPU_DI0_DW_SET0_2	0x00040090	0x1F40574	YES	YES
IPU_DI0_DW_SET0_3	0x00040094	0x1F40578	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DI0_DW_SET0_4	0x00040098	0x1F4057C	YES	YES
IPU_DI0_DW_SET0_5	0x0004009C	0x1F40580	YES	YES
IPU_DI0_DW_SET0_6	0x000400A0	0x1F40584	YES	YES
IPU_DI0_DW_SET0_7	0x000400A4	0x1F40588	YES	YES
IPU_DI0_DW_SET0_8	0x000400A8	0x1F4058C	YES	YES
IPU_DI0_DW_SET0_9	0x000400AC	0x1F40590	YES	YES
IPU_DI0_DW_SET0_10	0x000400B0	0x1F40594	YES	YES
IPU_DI0_DW_SET0_11	0x000400B4	0x1F40598	YES	YES
IPU_DI0_DW_SET1_0	0x000400B8	0x1F4059C	YES	YES
IPU_DI0_DW_SET1_1	0x000400BC	0x1F405A0	YES	YES
IPU_DI0_DW_SET1_2	0x000400C0	0x1F405A4	YES	YES
IPU_DI0_DW_SET1_3	0x000400C4	0x1F405A8	YES	YES
IPU_DI0_DW_SET1_4	0x000400C8	0x1F405AC	YES	YES
IPU_DI0_DW_SET1_5	0x000400CC	0x1F405B0	YES	YES
IPU_DI0_DW_SET1_6	0x000400D0	0x1F405B4	YES	YES
IPU_DI0_DW_SET1_7	0x000400D4	0x1F405B8	YES	YES
IPU_DI0_DW_SET1_8	0x000400D8	0x1F405BC	YES	YES
IPU_DI0_DW_SET1_9	0x000400DC	0x1F405C0	YES	YES
IPU_DI0_DW_SET1_10	0x000400E0	0x1F405C4	YES	YES
IPU_DI0_DW_SET1_11	0x000400E4	0x1F405C8	YES	YES
IPU_DI0_DW_SET2_0	0x000400E8	0x1F405CC	YES	YES
IPU_DI0_DW_SET2_1	0x000400EC	0x1F405D0	YES	YES
IPU_DI0_DW_SET2_2	0x000400F0	0x1F405D4	YES	YES
IPU_DI0_DW_SET2_3	0x000400F4	0x1F405D8	YES	YES
IPU_DI0_DW_SET2_4	0x000400F8	0x1F405DC	YES	YES
IPU_DI0_DW_SET2_5	0x000400FC	0x1F405E0	YES	YES
IPU_DI0_DW_SET2_6	0x00040100	0x1F405E4	YES	YES
IPU_DI0_DW_SET2_7	0x00040104	0x1F405E8	YES	YES
IPU_DI0_DW_SET2_8	0x00040108	0x1F405EC	YES	YES
IPU_DI0_DW_SET2_9	0x0004010C	0x1F405F0	YES	YES
IPU_DI0_DW_SET2_10	0x00040110	0x1F405F4	YES	YES
IPU_DI0_DW_SET2_11	0x00040114	0x1F405F8	YES	YES
IPU_DI0_DW_SET3_0	0x00040118	0x1F405FC	YES	YES
IPU_DI0_DW_SET3_1	0x0004011C	0x1F40600	YES	YES
IPU_DI0_DW_SET3_2	0x00040120	0x1F40604	YES	YES
IPU_DI0_DW_SET3_3	0x00040124	0x1F40608	YES	YES
IPU_DI0_DW_SET3_4	0x00040128	0x1F4060C	YES	YES
IPU_DI0_DW_SET3_5	0x0004012C	0x1F40610	YES	YES
IPU_DI0_DW_SET3_6	0x00040130	0x1F40614	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DI0_DW_SET3_7	0x00040134	0x1F40618	YES	YES
IPU_DI0_DW_SET3_8	0x00040138	0x1F4061C	YES	YES
IPU_DI0_DW_SET3_9	0x0004013C	0x1F40620	YES	YES
IPU_DI0_DW_SET3_10	0x00040140	0x1F40624	YES	YES
IPU_DI0_DW_SET3_11	0x00040144	0x1F40628	YES	YES
IPU_DI0_STP_REP_1	0x00040148	0x1F4062C	YES	YES
IPU_DI0_STP_REP_2	0x0004014C	0x1F40630	YES	YES
IPU_DI0_STP_REP_3	0x00040150	0x1F40634	YES	YES
IPU_DI0_STP_REP_4	0x00040154	0x1F40638	YES	YES
IPU_DI0_STP_REP_9	0x00040158	0x1F4063C	YES	YES
IPU_DI0_SER_CONF	0x0004015C	0x1F40640	YES	YES
IPU_DI0_SSC	0x00040160	0x1F40644	YES	YES
IPU_DI0_POL	0x00040164	0x1F40648	YES	YES
IPU_DI0_AW0	0x00040168	0x1F4064C	YES	YES
IPU_DI0_AW1	0x0004016C	0x1F40650	YES	YES
IPU_DI0_SCR_CONF	0x00040170	0x1F40654	YES	YES
IPU_DI0_STAT	0x00040174	NONE	NO	NO
IPU_DI1_GENERAL	0x00048000	0x1F40658	YES	YES
IPU_DI1_BS_CLKGEN0	0x00048004	0x1F4065C	YES	YES
IPU_DI1_BS_CLKGEN1	0x00048008	0x1F40660	YES	YES
IPU_DI1_SW_GEN0_1	0x0004800C	0x1F40664	YES	YES
IPU_DI1_SW_GEN0_2	0x00048010	0x1F40668	YES	YES
IPU_DI1_SW_GEN0_3	0x00048014	0x1F4066C	YES	YES
IPU_DI1_SW_GEN0_4	0x00048018	0x1F40670	YES	YES
IPU_DI1_SW_GEN0_5	0x0004801C	0x1F40674	YES	YES
IPU_DI1_SW_GEN0_6	0x00048020	0x1F40678	YES	YES
IPU_DI1_SW_GEN0_7	0x00048024	0x1F4067C	YES	YES
IPU_DI1_SW_GEN0_8	0x00048028	0x1F40680	YES	YES
IPU_DI1_SW_GEN0_9	0x0004802C	0x1F40684	YES	YES
IPU_DI1_SW_GEN1_1	0x00048030	0x1F40688	YES	YES
IPU_DI1_SW_GEN1_2	0x00048034	0x1F4068C	YES	YES
IPU_DI1_SW_GEN1_3	0x00048038	0x1F40690	YES	YES
IPU_DI1_SW_GEN1_4	0x0004803C	0x1F40694	YES	YES
IPU_DI1_SW_GEN1_5	0x00048040	0x1F40698	YES	YES
IPU_DI1_SW_GEN1_6	0x00048044	0x1F4069C	YES	YES
IPU_DI1_SW_GEN1_7	0x00048048	0x1F406A0	YES	YES
IPU_DI1_SW_GEN1_8	0x0004804C	0x1F406A4	YES	YES
IPU_DI1_SW_GEN1_9	0x00048050	0x1F406A8	YES	YES
IPU_DI1_SYNC_AS_GEN	0x00048054	0x1F406AC	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DI1_DW_GEN_0	0x00048058	0x1F406B0	YES	YES
IPU_DI1_DW_GEN_1	0x0004805C	0x1F406B4	YES	YES
IPU_DI1_DW_GEN_2	0x00048060	0x1F406B8	YES	YES
IPU_DI1_DW_GEN_3	0x00048064	0x1F406BC	YES	YES
IPU_DI1_DW_GEN_4	0x00048068	0x1F406C0	YES	YES
IPU_DI1_DW_GEN_5	0x0004806C	0x1F406C4	YES	YES
IPU_DI1_DW_GEN_6	0x00048070	0x1F406C8	YES	YES
IPU_DI1_DW_GEN_7	0x00048074	0x1F406CC	YES	YES
IPU_DI1_DW_GEN_8	0x00048078	0x1F406D0	YES	YES
IPU_DI1_DW_GEN_9	0x0004807C	0x1F406D4	YES	YES
IPU_DI1_DW_GEN_10	0x00048080	0x1F406D8	YES	YES
IPU_DI1_DW_GEN_11	0x00048084	0x1F406DC	YES	YES
IPU_DI1_DW_SET0_0	0x00048088	0x1F406E0	YES	YES
IPU_DI1_DW_SET0_1	0x0004808C	0x1F406E4	YES	YES
IPU_DI1_DW_SET0_2	0x00048090	0x1F406E8	YES	YES
IPU_DI1_DW_SET0_3	0x00048094	0x1F406EC	YES	YES
IPU_DI1_DW_SET0_4	0x00048098	0x1F406F0	YES	YES
IPU_DI1_DW_SET0_5	0x0004809C	0x1F406F4	YES	YES
IPU_DI1_DW_SET0_6	0x000480A0	0x1F406F8	YES	YES
IPU_DI1_DW_SET0_7	0x000480A4	0x1F406FC	YES	YES
IPU_DI1_DW_SET0_8	0x000480A8	0x1F40700	YES	YES
IPU_DI1_DW_SET0_9	0x000480AC	0x1F40704	YES	YES
IPU_DI1_DW_SET0_10	0x000480B0	0x1F40708	YES	YES
IPU_DI1_DW_SET0_11	0x000480B4	0x1F4070C	YES	YES
IPU_DI1_DW_SET1_0	0x000480B8	0x1F40710	YES	YES
IPU_DI1_DW_SET1_1	0x000480BC	0x1F40714	YES	YES
IPU_DI1_DW_SET1_2	0x000480C0	0x1F40718	YES	YES
IPU_DI1_DW_SET1_3	0x000480C4	0x1F4071C	YES	YES
IPU_DI1_DW_SET1_4	0x000480C8	0x1F40720	YES	YES
IPU_DI1_DW_SET1_5	0x000480CC	0x1F40724	YES	YES
IPU_DI1_DW_SET1_6	0x000480D0	0x1F40728	YES	YES
IPU_DI1_DW_SET1_7	0x000480D4	0x1F4072C	YES	YES
IPU_DI1_DW_SET1_8	0x000480D8	0x1F40730	YES	YES
IPU_DI1_DW_SET1_9	0x000480DC	0x1F40734	YES	YES
IPU_DI1_DW_SET1_10	0x000480E0	0x1F40738	YES	YES
IPU_DI1_DW_SET1_11	0x000480E4	0x1F4073C	YES	YES
IPU_DI1_DW_SET2_0	0x000480E8	0x1F40740	YES	YES
IPU_DI1_DW_SET2_1	0x000480EC	0x1F40744	YES	YES
IPU_DI1_DW_SET2_2	0x000480F0	0x1F40748	YES	YES

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DI1_DW_SET2_3	0x000480F4	0x1F4074C	YES	YES
IPU_DI1_DW_SET2_4	0x000480F8	0x1F40750	YES	YES
IPU_DI1_DW_SET2_5	0x000480FC	0x1F40754	YES	YES
IPU_DI1_DW_SET2_6	0x00048100	0x1F40758	YES	YES
IPU_DI1_DW_SET2_7	0x00048104	0x1F4075C	YES	YES
IPU_DI1_DW_SET2_8	0x00048108	0x1F40760	YES	YES
IPU_DI1_DW_SET2_9	0x0004810C	0x1F40764	YES	YES
IPU_DI1_DW_SET2_10	0x00048110	0x1F40768	YES	YES
IPU_DI1_DW_SET2_11	0x00048114	0x1F4076C	YES	YES
IPU_DI1_DW_SET3_0	0x00048118	0x1F40770	YES	YES
IPU_DI1_DW_SET3_1	0x0004811C	0x1F40774	YES	YES
IPU_DI1_DW_SET3_2	0x00048120	0x1F40778	YES	YES
IPU_DI1_DW_SET3_3	0x00048124	0x1F4077C	YES	YES
IPU_DI1_DW_SET3_4	0x00048128	0x1F40780	YES	YES
IPU_DI1_DW_SET3_5	0x0004812C	0x1F40784	YES	YES
IPU_DI1_DW_SET3_6	0x00048130	0x1F40788	YES	YES
IPU_DI1_DW_SET3_7	0x00048134	0x1F4078C	YES	YES
IPU_DI1_DW_SET3_8	0x00048138	0x1F40790	YES	YES
IPU_DI1_DW_SET3_9	0x0004813C	0x1F40794	YES	YES
IPU_DI1_DW_SET3_10	0x00048140	0x1F40798	YES	YES
IPU_DI1_DW_SET3_11	0x00048144	0x1F4079C	YES	YES
IPU_DI1_STP_REP_1	0x00048148	0x1F407A0	YES	YES
IPU_DI1_STP_REP_2	0x0004814C	0x1F407A4	YES	YES
IPU_DI1_STP_REP_3	0x00048150	0x1F407A8	YES	YES
IPU_DI1_STP_REP_4	0x00048154	0x1F407AC	YES	YES
IPU_DI1_STP_REP_9	0x00048158	0x1F407B0	YES	YES
IPU_DI1_SER_CONF	0x0004815C	0x1F407B4	YES	YES
IPU_DI1_SSC	0x00048160	0x1F407B8	YES	YES
IPU_DI1_POL	0x00048164	0x1F407BC	YES	YES
IPU_DI1_AW0	0x00048168	0x1F407C0	YES	YES
IPU_DI1_AW1	0x0004816C	0x1F407C4	YES	YES
IPU_DI1_SCR_CONF	0x00048170	0x1F407C8	YES	YES
IPU_DI1_STAT	0x00048174	NONE	NO	NO
IPU_SMFC_MAP	0x00050000	NONE	YES	NO
IPU_SMFC_WMC	0x00050004	NONE	YES	NO
IPU_SMFC_BS	0x00050008	NONE	YES	NO
IPU_DC_READ_CH_CONF	0x00058000	NONE	YES	NO
IPU_DC_READ_CH_ADDR	0x00058004	NONE	YES	NO
IPU_DC_RL0_CH_0	0x00058008	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DC_RL1_CH_0	0x0005800C	NONE	YES	NO
IPU_DC_RL2_CH_0	0x00058010	NONE	YES	NO
IPU_DC_RL3_CH_0	0x00058014	NONE	YES	NO
IPU_DC_RL4_CH_0	0x00058018	NONE	YES	NO
IPU_DC_WR_CH_CONF_1	0x0005801C	NONE	YES	NO
IPU_DC_WR_CH_ADDR_1	0x00058020	NONE	YES	NO
IPU_DC_RL0_CH_1	0x00058024	NONE	YES	NO
IPU_DC_RL1_CH_1	0x00058028	NONE	YES	NO
IPU_DC_RL2_CH_1	0x0005802C	NONE	YES	NO
IPU_DC_RL3_CH_1	0x00058030	NONE	YES	NO
IPU_DC_RL4_CH_1	0x00058034	NONE	YES	NO
IPU_DC_WR_CH_CONF_2	0x00058038	0x1F404AC	YES	NO
IPU_DC_WR_CH_ADDR_2	0x0005803C	0x1F404B0	YES	NO
IPU_DC_RL0_CH_2	0x00058040	0x1F404B4	YES	NO
IPU_DC_RL1_CH_2	0x00058044	0x1F404B8	YES	NO
IPU_DC_RL2_CH_2	0x00058048	0x1F404BC	YES	NO
IPU_DC_RL3_CH_2	0x0005804C	0x1F404C0	YES	NO
IPU_DC_RL4_CH_2	0x00058050	0x1F404C4	YES	NO
IPU_DC_CMD_CH_CONF_3	0x00058054	NONE	YES	NO
IPU_DC_CMD_CH_CONF_4	0x00058058	NONE	YES	NO
IPU_DC_WR_CH_CONF_5	0x0005805C	NONE	YES	NO
IPU_DC_WR_CH_ADDR_5	0x00058060	NONE	YES	NO
IPU_DC_RL0_CH_5	0x00058064	NONE	YES	NO
IPU_DC_RL1_CH_5	0x00058068	NONE	YES	NO
IPU_DC_RL2_CH_5	0x0005806C	NONE	YES	NO
IPU_DC_RL3_CH_5	0x00058070	NONE	YES	NO
IPU_DC_RL4_CH_5	0x00058074	NONE	YES	NO
IPU_DC_WR_CH_CONF_6	0x00058078	0x1F404C8	YES	NO
IPU_DC_WR_CH_ADDR_6	0x0005807C	0x1F404CC	YES	NO
IPU_DC_RL0_CH_6	0x00058080	0x1F404D0	YES	NO
IPU_DC_RL1_CH_6	0x00058084	0x1F404D4	YES	NO
IPU_DC_RL2_CH_6	0x00058088	0x1F404D8	YES	NO
IPU_DC_RL3_CH_6	0x0005808C	0x1F404DC	YES	NO
IPU_DC_RL4_CH_6	0x00058090	0x1F404E0	YES	NO
IPU_DC_WR_CH_CONF1_8	0x00058094	NONE	YES	NO
IPU_DC_WR_CH_CONF2_8	0x00058098	NONE	YES	NO
IPU_DC_RL1_CH_8	0x0005809C	NONE	YES	NO
IPU_DC_RL2_CH_8	0x000580A0	NONE	YES	NO
IPU_DC_RL3_CH_8	0x000580A4	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DC_RL4_CH_8	0x000580A8	NONE	YES	NO
IPU_DC_RL5_CH_8	0x000580AC	NONE	YES	NO
IPU_DC_RL6_CH_8	0x000580B0	NONE	YES	NO
IPU_DC_WR_CH_CONF1_9	0x000580B4	NONE	YES	NO
IPU_DC_WR_CH_CONF2_9	0x000580B8	NONE	YES	NO
IPU_DC_RL1_CH_9	0x000580BC	NONE	YES	NO
IPU_DC_RL2_CH_9	0x000580C0	NONE	YES	NO
IPU_DC_RL3_CH_9	0x000580C4	NONE	YES	NO
IPU_DC_RL4_CH_9	0x000580C8	NONE	YES	NO
IPU_DC_RL5_CH_9	0x000580CC	NONE	YES	NO
IPU_DC_RL6_CH_9	0x000580D0	NONE	YES	NO
IPU_DC_GEN	0x000580D4	NONE	YES	NO
IPU_DC_DISP_CONF1_0	0x000580D8	NONE	YES	NO
IPU_DC_DISP_CONF1_1	0x000580DC	NONE	YES	NO
IPU_DC_DISP_CONF1_2	0x000580E0	NONE	YES	NO
IPU_DC_DISP_CONF1_3	0x000580E4	NONE	YES	NO
IPU_DC_DISP_CONF2_0	0x000580E8	NONE	YES	NO
IPU_DC_DISP_CONF2_1	0x000580EC	NONE	YES	NO
IPU_DC_DISP_CONF2_2	0x000580F0	NONE	YES	NO
IPU_DC_DISP_CONF2_3	0x000580F4	NONE	YES	NO
IPU_DC_DI0_CONF_1	0x000580F8	NONE	YES	NO
IPU_DC_DI0_CONF_2	0x000580FC	NONE	YES	NO
IPU_DC_DI1_CONF_1	0x00058100	NONE	YES	NO
IPU_DC_DI1_CONF_2	0x00058104	NONE	YES	NO
IPU_DC_MAP_CONF_0	0x00058108	NONE	YES	NO
IPU_DC_MAP_CONF_1	0x0005810C	NONE	YES	NO
IPU_DC_MAP_CONF_2	0x00058110	NONE	YES	NO
IPU_DC_MAP_CONF_3	0x00058114	NONE	YES	NO
IPU_DC_MAP_CONF_4	0x00058118	NONE	YES	NO
IPU_DC_MAP_CONF_5	0x0005811C	NONE	YES	NO
IPU_DC_MAP_CONF_6	0x00058120	NONE	YES	NO
IPU_DC_MAP_CONF_7	0x00058124	NONE	YES	NO
IPU_DC_MAP_CONF_8	0x00058128	NONE	YES	NO
IPU_DC_MAP_CONF_9	0x0005812C	NONE	YES	NO
IPU_DC_MAP_CONF_10	0x00058130	NONE	YES	NO
IPU_DC_MAP_CONF_11	0x00058134	NONE	YES	NO
IPU_DC_MAP_CONF_12	0x00058138	NONE	YES	NO
IPU_DC_MAP_CONF_13	0x0005813C	NONE	YES	NO
IPU_DC_MAP_CONF_14	0x00058140	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DC_MAP_CONF_15	0x00058144	NONE	YES	NO
IPU_DC_MAP_CONF_16	0x00058148	NONE	YES	NO
IPU_DC_MAP_CONF_17	0x0005814C	NONE	YES	NO
IPU_DC_MAP_CONF_18	0x00058150	NONE	YES	NO
IPU_DC_MAP_CONF_19	0x00058154	NONE	YES	NO
IPU_DC_MAP_CONF_20	0x00058158	NONE	YES	NO
IPU_DC_MAP_CONF_21	0x0005815C	NONE	YES	NO
IPU_DC_MAP_CONF_22	0x00058160	NONE	YES	NO
IPU_DC_MAP_CONF_23	0x00058164	NONE	YES	NO
IPU_DC_MAP_CONF_24	0x00058168	NONE	YES	NO
IPU_DC_MAP_CONF_25	0x0005816C	NONE	YES	NO
IPU_DC_MAP_CONF_26	0x00058170	NONE	YES	NO
IPU_DC_UGDE0_0	0x00058174	NONE	YES	NO
IPU_DC_UGDE0_1	0x00058178	NONE	YES	NO
IPU_DC_UGDE0_2	0x0005817C	NONE	YES	NO
IPU_DC_UGDE0_3	0x00058180	NONE	YES	NO
IPU_DC_UGDE1_0	0x00058184	NONE	YES	NO
IPU_DC_UGDE1_1	0x00058188	NONE	YES	NO
IPU_DC_UGDE1_2	0x0005818C	NONE	YES	NO
IPU_DC_UGDE1_3	0x00058190	NONE	YES	NO
IPU_DC_UGDE2_0	0x00058194	NONE	YES	NO
IPU_DC_UGDE2_1	0x00058198	NONE	YES	NO
IPU_DC_UGDE2_2	0x0005819C	NONE	YES	NO
IPU_DC_UGDE2_3	0x000581A0	NONE	YES	NO
IPU_DC_UGDE3_0	0x000581A4	NONE	YES	NO
IPU_DC_UGDE3_1	0x000581A8	NONE	YES	NO
IPU_DC_UGDE3_2	0x000581AC	NONE	YES	NO
IPU_DC_UGDE3_3	0x000581B0	NONE	YES	NO
IPU_DC_LLA0	0x000581B4	NONE	YES	NO
IPU_DC_LLA1	0x000581B8	NONE	YES	NO
IPU_DC_R_LLA0	0x000581BC	NONE	YES	NO
IPU_DC_R_LLA1	0x000581C0	NONE	YES	NO
IPU_DC_WR_CH_ADDR_5_ALT	0x000581C4	NONE	YES	NO
IPU_DC_STAT	0x000581C8	NONE	NO	NO
IPU_DMFC_RD_CHAN	0x00060000	NONE	YES	NO
IPU_DMFC_WR_CHAN	0x00060004	NONE	YES	NO
IPU_DMFC_WR_CHAN_DEF	0x00060008	NONE	YES	NO
IPU_DMFC_DP_CHAN	0x0006000C	NONE	YES	NO
IPU_DMFC_DP_CHAN_DEF	0x00060010	NONE	YES	NO

Table continues on the next page...

Table 37-38. IPU SRM entries mapping (continued)

Register's name	Register's address	SRM entry	PG	LPSR
IPU_DMFC_GENERAL1	0x00060014	NONE	YES	NO
IPU_DMFC_GENERAL2	0x00060018	NONE	YES	NO
IPU_DMFC_IC_CTRL	0x0006001C	NONE	YES	NO
IPU_DMFC_WR_CHAN_ALT	0x00060020	NONE	YES	NO
IPU_DMFC_WR_CHAN_DEF_ALT	0x00060024	NONE	YES	NO
IPU_DMFC_DP_CHAN_ALT	0x00060028	NONE	YES	NO
IPU_DMFC_DP_CHAN_DEF_ALT	0x0006002C	NONE	YES	NO
IPU_DMFC_GENERAL1_ALT	0x00060030	NONE	YES	NO
IPU_DMFC_STAT	0x00060034	NONE	NO	NO
IPU_VDI_FSIZE	0x00068000	NONE	YES	NO
IPU_VDI_C	0x00068004	NONE	YES	NO
IPU_VDI_C2	0x00068008	NONE	YES	NO
IPU_VDI_CMBP_1	0x0006800C	NONE	YES	NO
IPU_VDI_CMBP_2	0x00068010	NONE	YES	NO
IPU_VDI_PS_1	0x00068014	NONE	YES	NO
IPU_VDI_PS_2	0x00068018	NONE	YES	NO
IPU_VDI_PS_3	0x0006801C	NONE	YES	NO
IPU_VDI_PS_4	0x00068020	NONE	YES	NO

37.4.12.7 Memory Access Unit

The Memory Access Unit (MA) supports ARM platform access to the IPU internal memories.

Some of the IPU internal memories are memory mapped. This unit handles accessing these memories. The table below describe the accessible memories and their limitations.

Table 37-41. Internal Memories Access Support and Limitations

Memory	Function	Support and Limitations
lut	IDMAC's look up table	Accessible only when All the IDMAC channels that use the LUT are disabled
cpmem	IDMAC's Channel parameter Memory	This memory can be accessed while tasks are enabled. Must be configured before enabling processing tasks.

Table continues on the next page...

Table 37-41. Internal Memories Access Support and Limitations (continued)

Memory	Function	Support and Limitations
		IDMAC channel parameters must not be changed in the CPM when the corresponding DMA channel is enabled excluding the base addresses (EBA0 and EBA1). One of these parameters can be changed during channel operation if it relates to the non-active double buffer.
tpm	IC's task parameter memory	Must be configured before enabling processing tasks. This memory can be accessed while tasks are enabled. IC task parameters must not be changed in the TPM when the corresponding task is active.
dc_template	DC's template memory	Can be configured before enabling tasks. Must not be accessed while operation. Both read and write access to the DC's template memory are forbidden when there is any enabled channel which can use the template

37.4.12.8 SISG - Still Image Synchronization Generator

The IPU incudes a "Still Image Synchronization Generator" (SISG), providing time-sensitive control signals synchronizing the image sensor with camera peripherals, such as a flash lamp and a mechanical shutter.

The SISG is implemented using a single time base counter, and six Time Compare Units - as described in the following figure.

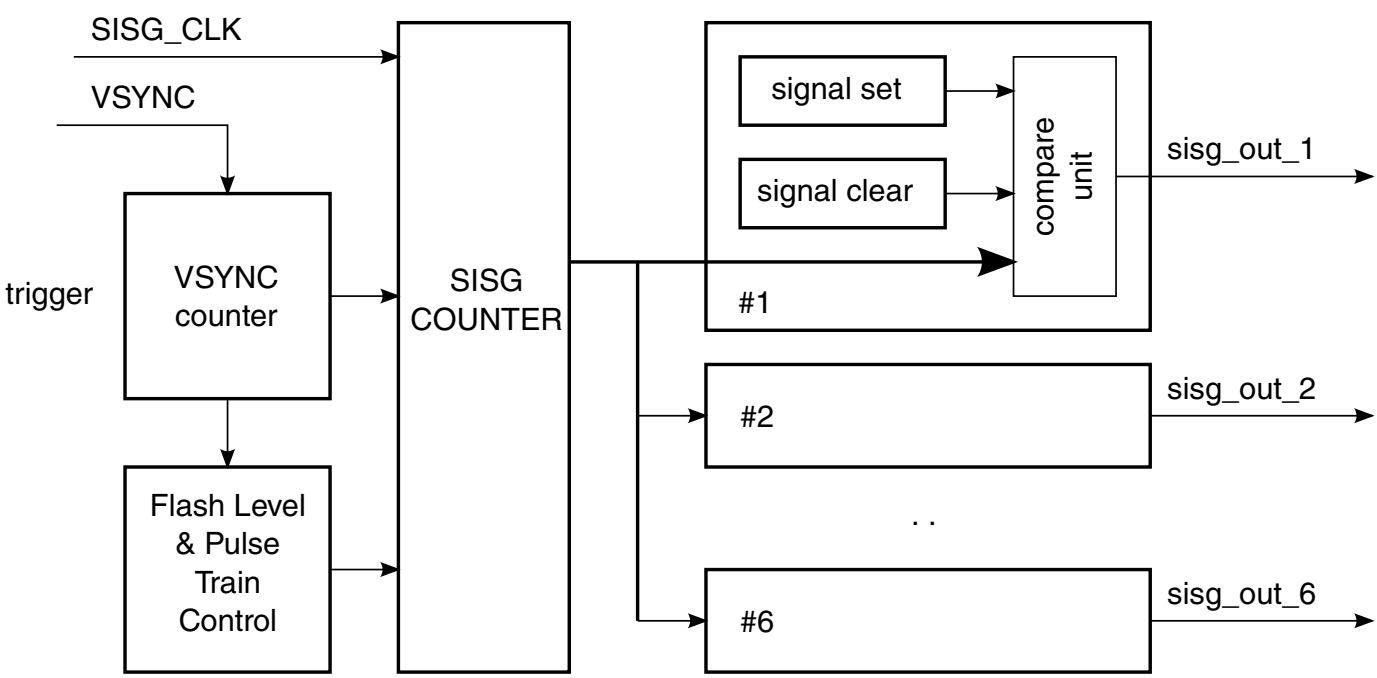


Figure 37-54. Still Image Signal Generator

The SISG inputs are:

- Activation trigger
- VSYNC: the frame-boundary signal from the sensor

The SISG is activated by one of the following triggers:

- The ARM platform by setting the MCU_ACTV_TRIG bit
- A signal generated by an appropriate short packet received through the MIPI/CSI-2 I/F
- An external signal (GPIO pin)

Upon activation, the counter is reset and then there are the following two possibilities:

- Counting starts immediately
- Counting start is delayed until one of the next 7 VSYNC signals (programmable via the NO_OF_VSYNC bits)

During the counting period, the SISG can generate up to 6 output strobes:

- Each strobe can be individually enabled or disabled and has a programmable polarity
- The edges of the strobes are generated at specified counter values - to achieve pixel-level resolution - as specified by programmable SISG_SET& SISG_CLR time tag registers
- The clock has 25 bits, to allow strobe generation during a time period of up to two 12M pixel frames

The SISG can repeat the above sequence for up to 32 cycles (this is provided to generate a train of flash pulses, for anti-red-eye or for measurements in low-light conditions). The repetition is implemented by resetting the counter, which can be triggered by one of the following events:

- A VSYNC signal
- A pre-defined value reached by the counter

After the last sequence, when the counter reaches its maximal value, it stops counting and the SISG remains in idle mode until the next activation.

37.4.12.9 Clock Change procedure

The IPU supports dynamic clock rate changes.

Types of change:

- DVFS transitions: frequent, initiated by the SoC's power modes controller (GPC) and the SoC's clock controller module (CCM)
- Other: infrequent, initiated by SW.

IPU may have on-going activities at this stage.

The display interface clocks may either change or remain unchanged. During screen refresh, the display clock would typically not change. During asynchronous access, it may be appropriate to change also the display interface clock. The choice between these options would be made in advance by the user

If the IPU display interface uses the external clock (DI0_DISP_CLK - ipp_di_0_ext_clk or DI1_DISP_CLK - ipp_di_1_ext_clk) source, it remains unchanged. A change in the rate of this clock is performed fully by SW, without the special HW support described below. In particular, the SW may have to stop explicitly any interaction with the display (e.g. screen refresh) before performing the change.

The user is responsible to make sure that the lowest planned clock (in DVFS transitions) is still high enough to support the expected activities (e.g. data rate through the display bus)

The procedure below describes the IPU handshaking with the CCM.

1. The user prepares 2 sets of clock modes CLOCK_MODE_0 is the default clock mode, CLOCK_MODE_1 is the alternate clock mode. The IPU toggles between these two settings following the next assertion of ipg_clk_change_rq. If the user sets the SRM_CLOCK_CHANGE_MODE bit then he should also prepare the registers in the SRM for each of the DIs. These registers include all the DI settings adjusted to the new clock.
2. CCM asserts the ipg_clk_change_rq signal when a clock change is needed
3. The CM calculates the new clock frequency and send it to the DI (signals are di0_clk_freq, and di1_clk_freq). These signals should be sent to the DI only after getting the di_clk_change_ack signal from the DIs. The values of this field could be.
    - 00 - 1/4 of full frequency
    - 01 - 1/2 of full frequency
    - 10 - full frequency
    - 11 - illegal
4. The CM sends a cm_clk_change_rq signal to the DIs.
5. The DI stops the clock to the display (freeze mode) according to DI0_CLOCK_STOP_MODE & DI1_CLOCK_STOP_MODE bits. If the DI is disable, the ACK from the DI is not needed and the CM will assume that the DI sent an ACK.
6. Once the clock to the display is stopped, the DI sends a signal to the CM called di_clk_change_ack
7. The CM wait for the clock change signals from both of the DIs
8. If the SRM_CLOCK_CHANGE_MODE bit is set the CM should read the new DI settings from the SRM and override the previous DI settings. Then the CM clears the SRM_CLOCK_CHANGE_MODE bit

9. Once the above is complete the CM asserts the ipg_clk_change_ack signal to the CCM
10. The CM sends the signals di0_clk_freq, and di1_clk_freq to the DIs.
11. The CCM will negate the cm_clk_change_rq
12. The CCM will now change the clocks
13. When the new clock arrives the ipu_clk_changed signal will be asserted.
14. The state machine on each DI will now move out of freeze mode and continue working with the new clock.

37.4.12.10 Low Power Modes

IPU supports the following low power modes.

- STOP: on this mode the clock to the IPU is stopped
- LPSR: low power screen refresh.The clock to the IPU is changed to a slower frequency, the IPU performs only screen refresh.

The user should not request LPSR. at the same time. This case is not supported and the results are not predictable.

The CCM may assert one of the 2 signals: stop_clk_at_stop_req OR stop_clk_at_wait_req

The IPU OR them internally as there's no difference between them with regards to IPU's behavior.

In all these modes the clock to the IPU is going to be stopped (assertion of stop_clk_at_stop_req).

IPU should complete all his tasks:

- CSIs complete transferring the last frame. Wait for csi_busy = 0
- IDMAC completes all the flows (all CH_BUSY = 0)
- All the flows in the FSU are complete. New ones do not start.
- CM sends stop request to the DIs
- Once the DI sent all the data to the display, it asserts an ACK signal

Only when all the above occurs, the CM can assert an internal signal called "IPU_IDLE"

IPU_IDLE is the starting point for any of the low power modes.

Note that if the VDIC was in use prior to entrance the low power mode the viewfinder task of the IC will be in WAIT_FOR_READY state (the task's status is reflected in the VF_TSTAT field). The user will need to manually switch that task to IDLE. This is done by performing the following steps:

- Wait for EOF of the viewfinder output channel (IDMAC_EOF_21)
- Set RSW_EN bit
- resume one frame via the viewfinder task.
- Disable the VDI and IC and the corresponding IDMAC channels

37.4.12.10.1 STOP Mode

In this mode the IPU sends the ipu_stby_ack after getting to IPU_IDLE state. The CCM will gate off the clock to the IPU.

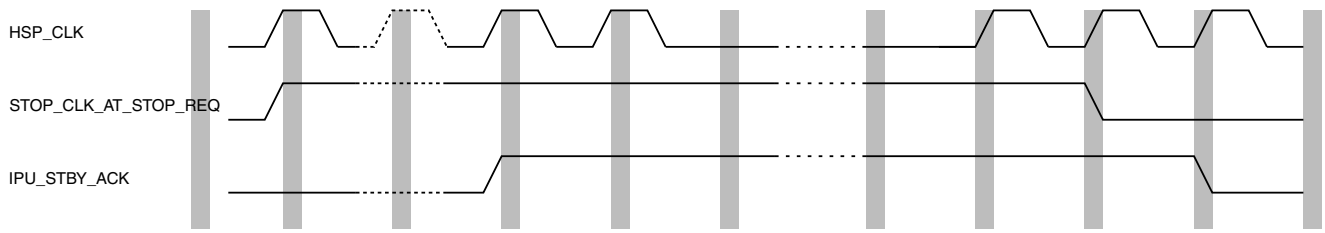


Figure 37-55. Entering and Exiting STOP Mode

Wake up from STOP mode

When the SoC decides to wake up from STOP mode it should:

- Resume the clock to the IPU.
- Negate the stop_clk_at_stop_req or stop_clk_at_wait_req signal.
- The IPU will then negate the ipu_stby_ack signal.
- The IPU will resume screen refresh.

37.4.12.10.2 Low Power Screen Refresh mode - LPSR

In Low Power Screen Refresh mode, the clock to the IPU is changed to a slower frequency, the IPU performs only screen refresh to a single display via the DP (channels 23 & 27).

Preparations

1. The user sets the LPSR_MODE bit indicating that the next assertion of stop_clk_at_stop_req OR stop_clk_at_wait_req activates the LPSR procedure.
2. The user moves the IPU to screen refresh flow. This means that if other tasks are active (flows via CSI or multiple flows to multiple displays) - this tasks needs to be complete and disabled. This is step is fully done by the user (SW). The only flow that remains active is screen refresh to a single display done via the DP (channels 23 & 27)

3. The user stores in the SRM the planned configuration for the sub-blocks involved in the LPSR flow. These configuration will e switched with the active registers' settings on later stage. The relevant sub-blocks are:
    - DI0 & DI1
    - DC
    - DP
    - DMFC
    - IDMAC
    - CM

Entering LPSR

The flow for entering LPSR is the same as stop mode.

1. The IPU follows the same procedure as on STOP mode till getting into IPU_IDLE state.
2. IPU swaps the registers of the relevant blocks with the pre stored content from the SRM. The content of the registers of the current flow is stored in the SRM. The IPU will switch back to this configuration after exiting from LPSR.
3. The IPU sends the ipu_stby_ack
4. The CCM will gate off the clock to the IPU.
5. The CCM will switch to the new clock
6. After changing the clock, the CCM asserts ccm_lpsr_ipu, this signal is synched inside the IPU to the hsp_clk
7. The IPU will resume screen refresh with the new settings.

Exit from LPSR

1. The CCM negates ccm_lpsr_ipu
2. The CM sends standby request to the DI
3. The DI should complete processing the current frame and stop the clock to the display and send an acknowledge signal to the CM.
4. The SRM swaps the registers' configuration. current configuration (LPSR) is saved in the SRM, the previous (original) configuration is stored in the blocks' registers.
5. IPU asserts the ipu_wakeup_ack signal indicating that it is now safe to leave LPSR mode.
6. CCM stops the clocks to IPU
7. CCM resumes the clock to the IPU - This clock is the same clock used prior to entering the LPSR mode.
8. CCM negates the stop_clk_at_stop_req (or stop_clk_at_wait_req)
9. IPU negates the ipu_stby_ack and the ipu_wakeup_ack signals
10. The IPU resumes screen refresh with the original settings.

The diagram below illustrates the procedure for entering and leaving LPSR mode

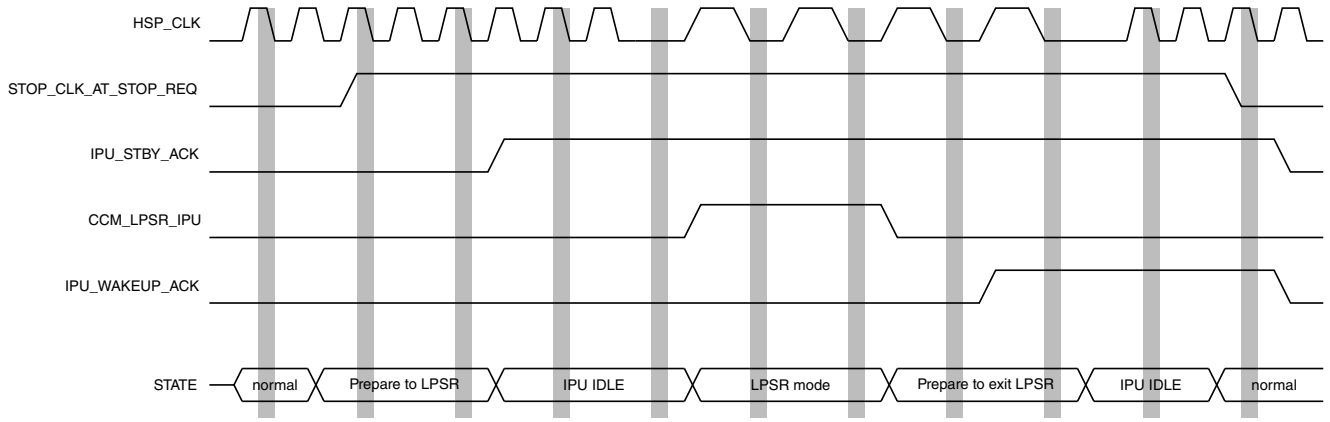


Figure 37-56. Entering and Exiting LPSR mode

37.5 IPU Memory Map/Register Definition

The address space for accesses through the AHB-lite slave port is 4 MB and it is split internally (with 2MB resolution) according to bit [21] of the address. Using the following notation

Address = (IPU_ID[31:25], 1,1,1,MSB[21], LSB[20:0])

the address is used as follows :

1. MSB=0: Low-level access to an external device, with LSB[3:0] = (Lock, CS, RS[1:0])

- LSB[5:4] = RS[1:0] (the address on the display interface)
- LSB[6] = Choice of display's channel (0=channel 8, 1=channel 9)
- LSB[7] = Lock (Lock=1 prevents the use of the display port until the next ARM platform access)

1. MSB=1: access to internal IPU registers, with address LSB

NOTE

The addresses given in the table are relative to the IPU base address defined at SoC's level.

IPU memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
260_0000	Configuration Register (IPU1_CONF)	32	R/W	0000_0000h	37.5.1/2959
260_0004	SISG Control 0 Register (IPU1_SISG_CTRL0)	32	R/W	0000_0000h	37.5.2/2962
260_0008	SISG Control 1 Register (IPU1_SISG_CTRL1)	32	R/W	0000_0000h	37.5.3/2963
260_000C	SISG Set<i> Register (IPU1_SISG_SET_i)	32	R/W	0000_0000h	37.5.4/2963
260_0024	SISG Clear <i> Register (IPU1_SISG_CLR_i)	32	R/W	0000_0000h	37.5.5/2964
260_003C	Interrupt Control Register 1 (IPU1_INT_CTRL_1)	32	R/W	0000_0000h	37.5.6/2964
260_0040	Interrupt Control Register 2 (IPU1_INT_CTRL_2)	32	R/W	0000_0000h	37.5.7/2968
260_0044	Interrupt Control Register 3 (IPU1_INT_CTRL_3)	32	R/W	0000_0000h	37.5.8/2971
260_0048	Interrupt Control Register 4 (IPU1_INT_CTRL_4)	32	R/W	0000_0000h	37.5.9/2975
260_004C	Interrupt Control Register 5 (IPU1_INT_CTRL_5)	32	R/W	0000_0000h	37.5.10/ 2978
260_0050	Interrupt Control Register 6 (IPU1_INT_CTRL_6)	32	R/W	0000_0000h	37.5.11/ 2983
260_0054	Interrupt Control Register 7 (IPU1_INT_CTRL_7)	32	R/W	0000_0000h	37.5.12/ 2986
260_0058	Interrupt Control Register 8 (IPU1_INT_CTRL_8)	32	R/W	0000_0000h	37.5.13/ 2988
260_005C	Interrupt Control Register 9 (IPU1_INT_CTRL_9)	32	R/W	0000_0000h	37.5.14/ 2990
260_0060	Interrupt Control Register 10 (IPU1_INT_CTRL_10)	32	R/W	0000_0000h	37.5.15/ 2992
260_0064	Interrupt Control Register 11 (IPU1_INT_CTRL_11)	32	R/W	0000_0000h	37.5.16/ 2994
260_0068	Interrupt Control Register 12 (IPU1_INT_CTRL_12)	32	R/W	0000_0000h	37.5.17/ 2997
260_006C	Interrupt Control Register 13 (IPU1_INT_CTRL_13)	32	R/W	0000_0000h	37.5.18/ 2999
260_0070	Interrupt Control Register 14 (IPU1_INT_CTRL_14)	32	R/W	0000_0000h	37.5.19/ 3003
260_0074	Interrupt Control Register15 (IPU1_INT_CTRL_15)	32	R/W	0000_0000h	37.5.20/ 3006
260_0078	SDMA Event Control Register 1 (IPU1_SDMA_EVENT_1)	32	R/W	0000_0000h	37.5.21/ 3010
260_007C	SDMA Event Control Register 2 (IPU1_SDMA_EVENT_2)	32	R/W	0000_0000h	37.5.22/ 3014
260_0080	SDMA Event Control Register 3 (IPU1_SDMA_EVENT_3)	32	R/W	0000_0000h	37.5.23/ 3017
260_0084	SDMA Event Control Register 4 (IPU1_SDMA_EVENT_4)	32	R/W	0000_0000h	37.5.24/ 3022
260_0088	SDMA Event Control Register 7 (IPU1_SDMA_EVENT_7)	32	R/W	0000_0000h	37.5.25/ 3025
260_008C	SDMA Event Control Register 8 (IPU1_SDMA_EVENT_8)	32	R/W	0000_0000h	37.5.26/ 3027

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
260_0090	SDMA Event Control Register 11 (IPU1_SDMA_EVENT_11)	32	R/W	0000_0000h	37.5.27/ 3028
260_0094	SDMA Event Control Register 12 (IPU1_SDMA_EVENT_12)	32	R/W	0000_0000h	37.5.28/ 3031
260_0098	SDMA Event Control Register 13 (IPU1_SDMA_EVENT_13)	32	R/W	0000_0000h	37.5.29/ 3033
260_009C	SDMA Event Control Register 14 (IPU1_SDMA_EVENT_14)	32	R/W	0000_0000h	37.5.30/ 3037
260_00A0	Shadow Registers Memory Priority 1 Register (IPU1_SRM_PRI1)	32	R/W	0000_0100h	37.5.31/ 3040
260_00A4	Shadow Registers Memory Priority 2 Register (IPU1_SRM_PRI2)	32	R/W	0605_0803h	37.5.32/ 3041
260_00A8	FSU Processing Flow 1 Register (IPU1_FS_PROC_FLOW1)	32	R/W	0000_0000h	37.5.33/ 3043
260_00AC	FSU Processing Flow 2 Register (IPU1_FS_PROC_FLOW2)	32	R/W	0000_0000h	37.5.34/ 3047
260_00B0	FSU Processing Flow 3 Register (IPU1_FS_PROC_FLOW3)	32	R/W	0000_0000h	37.5.35/ 3050
260_00B4	FSU Displaying Flow 1 Register (IPU1_FS_DISP_FLOW1)	32	R/W	0000_0000h	37.5.36/ 3053
260_00B8	FSU Displaying Flow 2 Register (IPU1_FS_DISP_FLOW2)	32	R/W	0000_0000h	37.5.37/ 3056
260_00BC	SKIP Register (IPU1_SKIP)	32	R/W	0000_0000h	37.5.38/ 3058
260_00C4	Display General Control Register (IPU1_DISP_GEN)	32	R/W	0040_0000h	37.5.39/ 3060
260_00C8	Display Alternate Flow Control Register 1 (IPU1_DISP_ALT1)	32	R/W	0040_0000h	37.5.40/ 3063
260_00CC	Display Alternate Flow Control Register 2 (IPU1_DISP_ALT2)	32	R/W	0000_0000h	37.5.41/ 3064
260_00D0	Display Alternate Flow Control Register 3 (IPU1_DISP_ALT3)	32	R/W	0040_0000h	37.5.42/ 3065
260_00D4	Display Alternate Flow Control Register 4 (IPU1_DISP_ALT4)	32	R/W	0000_0000h	37.5.43/ 3067
260_00DC	Memory Reset Control Register (IPU1_MEM_RST)	32	R/W	0000_0000h	37.5.44/ 3068
260_00E0	Power Modes Control Register (IPU1_PM)	32	R/W	0810_0810h	37.5.45/ 3070
260_00E4	General Purpose Register (IPU1_GPR)	32	R/W	0000_0000h	37.5.46/ 3073
260_0150	Channel Double Buffer Mode Select 0 Register (IPU1_CH_DB_MODE_SEL0)	32	R/W	0000_0000h	37.5.47/ 3075
260_0154	Channel Double Buffer Mode Select 1 Register (IPU1_CH_DB_MODE_SEL1)	32	R/W	0000_0000h	37.5.48/ 3079

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
260_0168	Alternate Channel Double Buffer Mode Select 0 Register (IPU1_ALT_CH_DB_MODE_SEL0)	32	R/W	0000_0000h	37.5.49/ 3082
260_016C	Alternate Channel Double Buffer Mode Select1 Register (IPU1_ALT_CH_DB_MODE_SEL1)	32	R/W	0000_0000h	37.5.50/ 3084
260_0178	Alternate Channel Triple Buffer Mode Select 0 Register (IPU1_ALT_CH_TRB_MODE_SEL0)	32	R/W	0000_0000h	37.5.51/ 3085
260_0200	Interrupt Status Register 1 (IPU1_INT_STAT_1)	32	w1c	0000_0000h	37.5.52/ 3088
260_0204	Interrupt Status Register2 (IPU1_INT_STAT_2)	32	w1c	0000_0000h	37.5.53/ 3093
260_0208	Interrupt Status Register 3 (IPU1_INT_STAT_3)	32	w1c	0000_0000h	37.5.54/ 3096
260_020C	Interrupt Status Register 4 (IPU1_INT_STAT_4)	32	w1c	0000_0000h	37.5.55/ 3100
260_0210	Interrupt Status Register 5 (IPU1_INT_STAT_5)	32	w1c	0000_0000h	37.5.56/ 3103
260_0214	Interrupt Status Register 6 (IPU1_INT_STAT_6)	32	w1c	0000_0000h	37.5.57/ 3108
260_0218	Interrupt Status Register7 1 (IPU1_INT_STAT_7)	32	w1c	0000_0000h	37.5.58/ 3111
260_021C	Interrupt Status Register 8 (IPU1_INT_STAT_8)	32	w1c	0000_0000h	37.5.59/ 3114
260_0220	Interrupt Status Register 9 (IPU1_INT_STAT_9)	32	w1c	0000_0000h	37.5.60/ 3117
260_0224	Interrupt Status Register 10 (IPU1_INT_STAT_10)	32	w1c	0000_0000h	37.5.61/ 3119
260_0228	Interrupt Status Register 11 (IPU1_INT_STAT_11)	32	w1c	0000_0000h	37.5.62/ 3122
260_022C	Interrupt Status Register 12 (IPU1_INT_STAT_12)	32	w1c	0000_0000h	37.5.63/ 3126
260_0230	Interrupt Status Register 13 (IPU1_INT_STAT_13)	32	w1c	0000_0000h	37.5.64/ 3128
260_0234	Interrupt Status Register 14 (IPU1_INT_STAT_14)	32	w1c	0000_0000h	37.5.65/ 3133
260_0238	Interrupt Status Register 15 (IPU1_INT_STAT_15)	32	w1c	0000_0000h	37.5.66/ 3136
260_023C	Current Buffer Register 0 (IPU1_CUR_BUF_0)	32	R	0000_0000h	37.5.67/ 3140
260_0240	Current Buffer Register 1 (IPU1_CUR_BUF_1)	32	R	0000_0000h	37.5.68/ 3145
260_0244	Alternate Current Buffer Register 0 (IPU1_ALT_CUR_0)	32	R	0000_0000h	37.5.69/ 3149
260_0248	Alternate Current Buffer Register 1 (IPU1_ALT_CUR_1)	32	R	0000_0000h	37.5.70/ 3151

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
260_024C	Shadow Registers Memory Status Register (IPU1_SRM_STAT)	32	R	0000_0000h	37.5.71/ 3154
260_0250	Processing Status Tasks Register (IPU1_PROC_TASKS_STAT)	32	R	0000_0000h	37.5.72/ 3156
260_0254	Display Tasks Status Register (IPU1_DISP_TASKS_STAT)	32	R	0000_0000h	37.5.73/ 3158
260_0258	Triple Current Buffer Register 0 (IPU1_TRIPLE_CUR_BUF_0)	32	R	0000_0000h	37.5.74/ 3160
260_025C	Triple Current Buffer Register 1 (IPU1_TRIPLE_CUR_BUF_1)	32	R	0000_0000h	37.5.75/ 3162
260_0268	IPU Channels Buffer 0 Ready 0 Register (IPU1_CH_BUF0_RDY0)	32	R/W	0000_0000h	37.5.76/ 3163
260_026C	IPU Channels Buffer 0 Ready 1 Register (IPU1_CH_BUF0_RDY1)	32	R/W	0000_0000h	37.5.77/ 3167
260_0270	IPU Channels Buffer 1 Ready 0 Register (IPU1_CH_BUF1_RDY0)	32	R/W	0000_0000h	37.5.78/ 3169
260_0274	IPU Channels Buffer 1 Ready 1Register (IPU1_CH_BUF1_RDY1)	32	R/W	0000_0000h	37.5.79/ 3172
260_0278	IPU Alternate Channels Buffer 0 Ready 0 Register (IPU1_ALT_CH_BUF0_RDY0)	32	R/W	0000_0000h	37.5.80/ 3175
260_027C	IPU Alternate Channels Buffer 0 Ready 1 Register (IPU1_ALT_CH_BUF0_RDY1)	32	R/W	0000_0000h	37.5.81/ 3176
260_0280	IPU Alternate Channels Buffer1 Ready 0 Register (IPU1_ALT_CH_BUF1_RDY0)	32	R/W	0000_0000h	37.5.82/ 3177
260_0284	IPU Alternate Channels Buffer 1 Ready 1 Register (IPU1_ALT_CH_BUF1_RDY1)	32	R/W	0000_0000h	37.5.83/ 3178
260_0288	IPU Channels Buffer 2 Ready 0 Register (IPU1_CH_BUF2_RDY0)	32	R/W	0000_0000h	37.5.84/ 3179
260_028C	IPU Channels Buffer 2 Ready 1 Register (IPU1_CH_BUF2_RDY1)	32	R/W	0000_0000h	37.5.85/ 3181
260_8000	IDMAC Configuration Register (IPU1_IDMAC_CONF)	32	R/W	0000_002Fh	37.5.86/ 3182
260_8004	IDMAC Channel Enable 1 Register (IPU1_IDMAC_CH_EN_1)	32	R/W	0000_0000h	37.5.87/ 3184
260_8008	IDMAC Channel Enable 2 Register (IPU1_IDMAC_CH_EN_2)	32	R/W	0000_0000h	37.5.88/ 3187
260_800C	IDMAC Separate Alpha Indication Register (IPU1_IDMAC_SEP_ALPHA)	32	R/W	0000_0000h	37.5.89/ 3189
260_8010	IDMAC Alternate Separate Alpha Indication Register (IPU1_IDMAC_ALT_SEP_ALPHA)	32	R/W	0000_0000h	37.5.90/ 3191
260_8014	IDMAC Channel Priority 1 Register (IPU1_IDMAC_CH_PRI_1)	32	R/W	0000_0000h	37.5.91/ 3193
260_8018	IDMAC Channel Priority 2 Register (IPU1_IDMAC_CH_PRI_2)	32	R/W	0000_0000h	37.5.92/ 3196

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
260_801C	IDMAC Channel Watermark Enable 1 Register (IPU1_IDMAC_WM_EN_1)	32	R/W	0000_0000h	37.5.93/ 3198
260_8020	IDMAC Channel Watermark Enable 2 Register (IPU1_IDMAC_WM_EN_2)	32	R/W	0000_0000h	37.5.94/ 3200
260_8024	IDMAC Channel Lock Enable 1Register (IPU1_IDMAC_LOCK_EN_1)	32	R/W	0000_0000h	37.5.95/ 3201
260_8028	IDMAC Channel Lock Enable 2Register (IPU1_IDMAC_LOCK_EN_2)	32	R/W	0000_0000h	37.5.96/ 3203
260_802C	IDMAC Channel Alternate Address 0 Register (IPU1_IDMAC_SUB_ADDR_0)	32	R/W	0000_0000h	37.5.97/ 3204
260_8030	IDMAC Channel Alternate Address 1 Register (IPU1_IDMAC_SUB_ADDR_1)	32	R/W	0000_0000h	37.5.98/ 3205
260_8034	IDMAC Channel Alternate Address 2 Register (IPU1_IDMAC_SUB_ADDR_2)	32	R/W	0000_0000h	37.5.99/ 3206
260_8038	IDMAC Channel Alternate Address 3 Register (IPU1_IDMAC_SUB_ADDR_3)	32	R/W	0000_0000h	37.5.100/ 3207
260_803C	IDMAC Channel Alternate Address 4 Register (IPU1_IDMAC_SUB_ADDR_4)	32	R/W	0000_0000h	37.5.101/ 3209
260_8040	IDMAC Band Mode Enable 1 Register (IPU1_IDMAC_BNDM_EN_1)	32	R/W	0000_0000h	37.5.102/ 3210
260_8044	IDMAC Band Mode Enable 2 Register (IPU1_IDMAC_BNDM_EN_2)	32	R/W	0000_0000h	37.5.103/ 3213
260_8048	IDMAC Scroll Coordinations Register (IPU1_IDMAC_SC_CORD)	32	R/W	0000_0000h	37.5.104/ 3214
260_804C	IDMAC Scroll Coordinations Register 1 (IPU1_IDMAC_SC_CORD_1)	32	R/W	0000_0000h	37.5.105/ 3215
260_8100	IDMAC Channel Busy 1 Register (IPU1_IDMAC_CH_BUSY_1)	32	R	0000_0000h	37.5.106/ 3216
260_8104	IDMAC Channel Busy 2 Register (IPU1_IDMAC_CH_BUSY_2)	32	R	0000_0000h	37.5.107/ 3222
261_8000	DP Common Configuration Sync Flow Register (IPU1_DP_COM_CONF_SYNC)	32	R/W	0000_0000h	37.5.108/ 3226
261_8004	DP Graphic Window Control Sync Flow Register (IPU1_DP_Graph_Wind_CTRL_SYNC)	32	R/W	0000_0000h	37.5.109/ 3228
261_8008	DP Partial Plane Window Position Sync Flow Register (IPU1_DP_FG_POS_SYNC)	32	R/W	0000_0000h	37.5.110/ 3229
261_800C	DP Cursor Position and Size Sync Flow Register (IPU1_DP_CUR_POS_SYNC)	32	R/W	0000_0000h	37.5.111/ 3229
261_8010	DP Color Cursor Mapping Sync Flow Register (IPU1_DP_CUR_MAP_SYNC)	32	R/W	0000_0000h	37.5.112/ 3230
261_8014	DP Gamma Constants Sync Flow Register i (IPU1_DP_GAMMA_C_SYNC_i)	32	R/W	0000_0000h	37.5.113/ 3231
261_8034	DP Gamma Correction Slope Sync Flow Register i (IPU1_DP_GAMMA_S_SYNC_i)	32	R/W	0000_0000h	37.5.114/ 3231

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
261_8044	DP Color Space Conversion Control Sync Flow Registers (IPU1_DP_CSCA_SYNC_i)	32	R/W	0000_0000h	37.5.115/ 3232
261_8054	DP Color Conversion Control Sync Flow Register 0 (IPU1_DP_SCS_SYNC_0)	32	R/W	0000_0000h	37.5.116/ 3233
261_8058	DP Color Conversion Control Sync Flow Register 1 (IPU1_DP_SCS_SYNC_1)	32	R/W	0000_0000h	37.5.117/ 3233
261_805C	DP Cursor Position and Size Alternate Register (IPU1_DP_CUR_POS_ALT)	32	R/W	0000_0000h	37.5.118/ 3234
261_8060	DP Common Configuration Async 0 Flow Register (IPU1_DP_COM_CONF_ASYNC0)	32	R/W	0000_0000h	37.5.119/ 3235
261_8064	DP Graphic Window Control Async 0 Flow Register (IPU1_DP_GRAPH_WIND_CTRL_ASYNC0)	32	R/W	0000_0000h	37.5.120/ 3237
261_8068	DP Partial Plane Window Position Async 0 Flow Register (IPU1_DP_FG_POS_ASYNC0)	32	R/W	0000_0000h	37.5.121/ 3238
261_806C	DP Cursor Position and Size Async 0 Flow Register (IPU1_DP_CUR_POS_ASYNC0)	32	R/W	0000_0000h	37.5.122/ 3239
261_8070	DP Color Cursor Mapping Async 0 Flow Register (IPU1_DP_CUR_MAP_ASYNC0)	32	R/W	0000_0000h	37.5.123/ 3239
261_8074	DP Gamma Constant Async 0 Flow Register i (IPU1_DP_GAMMA_C_ASYNC0_i)	32	R/W	0000_0000h	37.5.124/ 3240
261_8094	DP Gamma Correction Slope Async 0 Flow Register i (IPU1_DP_GAMMA_S_ASYNC0_i)	32	R/W	0000_0000h	37.5.125/ 3241
261_80A4	DP Color Space Conversion Control Async 0 Flow Register i (IPU1_DP_CSCA_ASYNC0_i)	32	R/W	0000_0000h	37.5.126/ 3241
261_80B4	DP Color Conversion Control Async 0 Flow Register 0 (IPU1_DP_CSC_ASYNC0_0)	32	R/W	0000_0000h	37.5.127/ 3242
261_80B8	DP Color Conversion Control Async 1 Flow Register (IPU1_DP_CSC_ASYNC_1)	32	R/W	0000_0000h	37.5.128/ 3243
261_80BC	DP Common Configuration Async 1 Flow Register (IPU1_DP_COM_CONF_ASYNC1)	32	R/W	0000_0000h	37.5.129/ 3244
261_80BC	DP Debug Control Register (IPU1_DP_DEBUG_CNT)	32	R/W	0000_0000h	37.5.130/ 3246
261_80C0	DP Graphic Window Control Async 1 Flow Register (IPU1_DP_GRAPH_WIND_CTRL_ASYNC1)	32	R/W	0000_0000h	37.5.131/ 3247
261_80C0	DP Debug Status Register (IPU1_DP_DEBUG_STAT)	32	R	0000_0000h	37.5.132/ 3248
261_80C4	DP Partial Plane Window Position Async 1 Flow Register (IPU1_DP_FG_POS_ASYNC1)	32	R/W	0000_0000h	37.5.133/ 3250
261_80C8	DP Cursor Postion and Size Async 1 Flow Register (IPU1_DP_CUR_POS_ASYNC1)	32	R/W	0000_0000h	37.5.134/ 3250
261_80CC	DP Color Cursor Mapping Async 1 Flow Register (IPU1_DP_CUR_MAP_ASYNC1)	32	R/W	0000_0000h	37.5.135/ 3251
261_80D0	DP Gamma Constants Async 1 Flow Register i (IPU1_DP_GAMMA_C_ASYNC1_i)	32	R/W	0000_0000h	37.5.136/ 3252

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
261_80F0	DP Gamma Correction Slope Async 1 Flow Register i (IPU1_DP_GAMMA_S_ASYN1_i)	32	R/W	0000_0000h	37.5.137/ 3253
261_8100	DP Color Space Converstion Control Async 1 Flow Register i (IPU1_DP_CSCA_ASYNC1_i)	32	R/W	0000_0000h	37.5.138/ 3253
261_8110	DP Color Conversion Control Async 1 Flow Register 0 (IPU1_DP_CSC_ASYNC1_0)	32	R/W	0000_0000h	37.5.139/ 3254
261_8114	DP Color Conversion Control Async 1 Flow Register 1 (IPU1_DP_CSC_ASYNC1_1)	32	R/W	0000_0000h	37.5.140/ 3255
262_0000	IC Configuration Register (IPU1_IC_CONF)	32	R/W	0000_0000h	37.5.141/ 3256
262_0004	IC Preprocessing Encoder Resizing Coefficients Register (IPU1_IC_PRP_ENC_RSC)	32	R/W	2000_2000h	37.5.142/ 3258
262_0008	IC Preprocessing View-Finder Resizing Coefficients Register (IPU1_IC_PRP_VF_RSC)	32	R/W	2000_2000h	37.5.143/ 3259
262_000C	IC Postprocessing Encoder Resizing Coefficients Register (IPU1_IC_PP_RSC)	32	R/W	2000_2000h	37.5.144/ 3260
262_0010	IC Combining Parameters Register 1 (IPU1_IC_CMBP_1)	32	R/W	0000_0000h	37.5.145/ 3261
262_0014	IC Combining Parameters Register 2 (IPU1_IC_CMBP_2)	32	R/W	0000_0000h	37.5.146/ 3261
262_0018	IC IDMAC Parameters 1 Register (IPU1_IC_IDMAC_1)	32	R/W	0000_0000h	37.5.147/ 3262
262_001C	IC IDMAC Parameters 2 Register (IPU1_IC_IDMAC_2)	32	R/W	0000_0000h	37.5.148/ 3265
262_0020	IC IDMAC Parameters 3Register (IPU1_IC_IDMAC_3)	32	R/W	0000_0000h	37.5.149/ 3266
262_0024	IC IDMAC Parameters 4 Register (IPU1_IC_IDMAC_4)	32	R/W	0000_0000h	37.5.150/ 3266
263_0000	CSI0 Sensor Configuration Register (IPU1_CSI0_SENS_CONF)	32	R/W	0000_0000h	37.5.151/ 3267
263_0004	CSI0 Sense Frame Size Register (IPU1_CSI0_SENS_FRM_SIZE)	32	R/W	0000_0000h	37.5.152/ 3270
263_0008	CSI0 Actual Frame Size Register (IPU1_CSI0_ACT_FRM_SIZE)	32	R/W	0000_0000h	37.5.153/ 3270
263_000C	CSI0 Output Control Register (IPU1_CSI0_OUT_FRM_CTRL)	32	R/W	0000_0000h	37.5.154/ 3271
263_0010	CSIO Test Control Register (IPU1_CSI0_TST_CTRL)	32	R/W	0000_0000h	37.5.155/ 3272
263_0014	CSIO CCIR Code Register 1 (IPU1_CSI0_CCIR_CODE_1)	32	R/W	0000_0000h	37.5.156/ 3273
263_0018	CSIO CCIR Code Register 2 (IPU1_CSI0_CCIR_CODE_2)	32	R/W	0000_0000h	37.5.157/ 3274
263_001C	CSIO CCIR Code Register 3 (IPU1_CSI0_CCIR_CODE_3)	32	R/W	0000_0000h	37.5.158/ 3275

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
263_0020	CSI0 Data Identifier Register (IPU1_CSI0_DI)	32	R/W	FFFF_FFFFh	37.5.159/ 3275
263_0024	CSI0 SKIP Register (IPU1_CSI0_SKIP)	32	R/W	0000_0000h	37.5.160/ 3276
263_0028	CSI0 Compander Control Register (IPU1_CSIO_CPD_CTRL)	32	R/W	0000_0000h	37.5.161/ 3277
263_002C	CSI0 Red Component Compander Constants Register <i> (IPU1_CSIO_CPD_RC_i)	32	R/W	0000_0000h	37.5.162/ 3278
263_004C	CSI0 Red Component Compander SLOPE Register <i> (IPU1_CSIO_CPD_RS_i)	32	R/W	0000_0000h	37.5.163/ 3279
263_005C	CSI0 GR Component Compander Constants Register <i> (IPU1_CSIO_CPD_GRC_i)	32	R/W	0000_0000h	37.5.164/ 3279
263_007C	CSI0 GR Component Compander SLOPE Register <i> (IPU1_CSIO_CPD_GRS_i)	32	R/W	0000_0000h	37.5.165/ 3280
263_008C	CSI0 GB Component Compander Constants Register <i> (IPU1_CSIO_CPD_GBC_i)	32	R/W	0000_0000h	37.5.166/ 3281
263_00AC	CSI0 GB Component Compander SLOPE Register <i> (IPU1_CSIO_CPD_GBS_i)	32	R/W	0000_0000h	37.5.167/ 3281
263_00BC	CSI0 Blue Component Compander Constants Register <i> (IPU1_CSIO_CPD_BC_i)	32	R/W	0000_0000h	37.5.168/ 3282
263_00DC	CSI0 Blue Component Compander SLOPE Register <i> (IPU1_CSIO_CPD_BS_i)	32	R/W	0000_0000h	37.5.169/ 3283
263_00EC	CSI0 Compander Offset Register 1 (IPU1_CSI0_CPD_OFFSET1)	32	R/W	0000_0000h	37.5.170/ 3283
263_00F0	CSI0 Compander Offset Register 2 (IPU1_CSI0_CPD_OFFSET2)	32	R/W	0000_0000h	37.5.171/ 3284
263_8000	CSI1 Sensor Configuration Register (IPU1_CSI1_SENS_CONF)	32	R/W	0000_0000h	37.5.172/ 3285
263_8004	CSI1 Sense Frame Size Register (IPU1_CSI1_SENS_FRM_SIZE)	32	R/W	0000_0000h	37.5.173/ 3287
263_8008	CSI1 Actual Frame Size Register (IPU1_CSI1_ACT_FRM_SIZE)	32	R/W	0000_0000h	37.5.174/ 3288
263_800C	CSI1 Output Control Register (IPU1_CSI1_OUT_FRM_CTRL)	32	R/W	0000_0000h	37.5.175/ 3289
263_8010	CSI1 Test Control Register (IPU1_CSI1_TST_CTRL)	32	R/W	0000_0000h	37.5.176/ 3290
263_8014	CSI1 CCIR Code Register 1 (IPU1_CSI1_CCIR_CODE_1)	32	R/W	0000_0000h	37.5.177/ 3291
263_8018	CSI1 CCIR Code Register 2 (IPU1_CSI1_CCIR_CODE_2)	32	R/W	0000_0000h	37.5.178/ 3292
263_801C	CSI1 CCIR Code Register 3 (IPU1_CSI1_CCIR_CODE_3)	32	R/W	0000_0000h	37.5.179/ 3293
263_8020	CSI1 Data Identifier Register (IPU1_CSI1_DI)	32	R/W	FFFF_FFFFh	37.5.180/ 3293

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
263_8024	CSI1 SKIP Register (IPU1_CSI1_SKIP)	32	R/W	0000_0000h	37.5.181/ 3294
263_8028	CSI1 Compander Control Register (IPU1_CSI1_CPD_CTRL)	32	R/W	0000_0000h	37.5.182/ 3295
263_802C	CSI1 Red Component Compander Constants Register <i> (IPU1_CSI1_CPD_RC_i)	32	R/W	0000_0000h	37.5.183/ 3296
263_804C	CSI1 Red Component Compander SLOPE Register <i> (IPU1_CSI1_CPD_RS_i)	32	R/W	0000_0000h	37.5.184/ 3296
263_805C	CSI1 GR Component Compander Constants Register <i> (IPU1_CSI1_CPD_GRC_i)	32	R/W	0000_0000h	37.5.185/ 3297
263_807C	CSI1 GR Component Compander SLOPE Register <i> (IPU1_CSI1_CPD_GRS_i)	32	R/W	0000_0000h	37.5.186/ 3298
263_808C	CSI1 GB Component Compander Constants Register <i> (IPU1_CSI1_CPD_GBC_i)	32	R/W	0000_0000h	37.5.187/ 3298
263_80AC	CSI1 GB Component Compander SLOPE Register <i> (IPU1_CSI1_CPD_GBS_i)	32	R/W	0000_0000h	37.5.188/ 3299
263_80BC	CSI1 Blue Component Compander Constants Register <i> (IPU1_CSI1_CPD_BC_i)	32	R/W	0000_0000h	37.5.189/ 3300
263_80DC	CSI1 Blue Component Compander SLOPE Register <i> (IPU1_CSI1_CPD_BS_i)	32	R/W	0000_0000h	37.5.190/ 3300
263_80EC	CSI1 Compander Offset Register 1 (IPU1_CSI1_CPD_OFFSET1)	32	R/W	0000_0000h	37.5.191/ 3301
263_80F0	CSI1 Compander Offset Register 2 (IPU1_CSI1_CPD_OFFSET2)	32	R/W	0000_0000h	37.5.192/ 3302
264_0000	DI0 General Register (IPU1_DI0_GENERAL)	32	R/W	0020_0000h	37.5.193/ 3303
264_0004	DI0 Base Sync Clock Gen 0 Register (IPU1_DI0_BS_CLKGEN0)	32	R/W	0000_0000h	37.5.194/ 3305
264_0008	DI0 Base Sync Clock Gen 1 Register (IPU1_DI0_BS_CLKGEN1)	32	R/W	0000_0000h	37.5.195/ 3306
264_000C	DI0 Sync Wave Gen 1 Register 0 (IPU1_DI0_SW_GEN0_1)	32	R/W	0000_0000h	37.5.196/ 3306
264_0010	DI0 Sync Wave Gen 2 Register 0 (IPU1_DI0_SW_GEN0_2)	32	R/W	0000_0000h	37.5.197/ 3308
264_0014	DI0 Sync Wave Gen 3 Register 0 (IPU1_DI0_SW_GEN0_3)	32	R/W	0000_0000h	37.5.198/ 3309
264_0018	DI0 Sync Wave Gen 4 Register 0 (IPU1_DI0_SW_GEN0_4)	32	R/W	0000_0000h	37.5.199/ 3310
264_001C	DI0 Sync Wave Gen 5 Register 0 (IPU1_DI0_SW_GEN0_5)	32	R/W	0000_0000h	37.5.200/ 3311
264_0020	DI0 Sync Wave Gen 6 Register 0 (IPU1_DI0_SW_GEN0_6)	32	R/W	0000_0000h	37.5.201/ 3313
264_0024	DI0 Sync Wave Gen 7 Register 0 (IPU1_DI0_SW_GEN0_7)	32	R/W	0000_0000h	37.5.202/ 3314

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
264_0028	DI0 Sync Wave Gen 8 Register 0 (IPU1_DI0_SW_GEN0_8)	32	R/W	0000_0000h	37.5.203/ 3315
264_002C	DI0 Sync Wave Gen 9 Register 0 (IPU1_DI0_SW_GEN0_9)	32	R/W	0000_0000h	37.5.204/ 3316
264_0030	DI0 Sync Wave Gen 1 Register 1 (IPU1_DI0_SW_GEN1_1)	32	R/W	0000_0000h	37.5.205/ 3318
264_0034	DI0 Sync Wave Gen 2 Register 1 (IPU1_DI0_SW_GEN1_2)	32	R/W	0000_0000h	37.5.206/ 3320
264_0038	DI0 Sync Wave Gen 3 Register 1 (IPU1_DI0_SW_GEN1_3)	32	R/W	0000_0000h	37.5.207/ 3322
264_003C	DI0 Sync Wave Gen 4 Register 1 (IPU1_DI0_SW_GEN1_4)	32	R/W	0000_0000h	37.5.208/ 3324
264_0040	DI0 Sync Wave Gen 5 Register 1 (IPU1_DI0_SW_GEN1_5)	32	R/W	0000_0000h	37.5.209/ 3326
264_0044	DI0 Sync Wave Gen 6 Register 1 (IPU1_DI0_SW_GEN1_6)	32	R/W	0000_0000h	37.5.210/ 3328
264_0048	DI0 Sync Wave Gen 7 Register 1 (IPU1_DI0_SW_GEN1_7)	32	R/W	0000_0000h	37.5.211/ 3330
264_004C	DI0 Sync Wave Gen 8 Register 1 (IPU1_DI0_SW_GEN1_8)	32	R/W	0000_0000h	37.5.212/ 3332
264_0050	DI0 Sync Wave Gen 9 Register 1 (IPU1_DI0_SW_GEN1_9)	32	R/W	0000_0000h	37.5.213/ 3334
264_0054	DI0 Sync Assistance Gen Register (IPU1_DI0_SYNC_AS_GEN)	32	R/W	0000_0000h	37.5.214/ 3335
264_0058	DI0 Data Wave Gen <i> Register (IPU1_DI0_DW_GEN_i)	32	R/W	0000_0000h	37.5.215/ 3336
264_0088	DI0 Data Wave Set 0 <i> Register (IPU1_DI0_DW_SET0_i)	32	R/W	0000_0000h	37.5.216/ 3339
264_00B8	DI0 Data Wave Set 1 <i> Register (IPU1_DI0_DW_SET1_i)	32	R/W	0000_0000h	37.5.217/ 3339
264_00E8	DI0 Data Wave Set 2 <i> Register (IPU1_DI0_DW_SET2_i)	32	R/W	0000_0000h	37.5.218/ 3340
264_0118	DI0 Data Wave Set 3 <i> Register (IPU1_DI0_DW_SET3_i)	32	R/W	0000_0000h	37.5.219/ 3341
264_0148	DI0 Step Repeat <i> Registers (IPU1_DI0_STP_REP_i)	32	R/W	0000_0000h	37.5.220/ 3341
264_0158	DI0 Step Repeat 9 Registers (IPU1_DI0_STP_REP_9)	32	R/W	0000_0000h	37.5.221/ 3342
264_015C	DI0 Serial Display Control Register (IPU1_DI0_SER_CONF)	32	R/W	0000_0000h	37.5.222/ 3342
264_0160	DI0 Special Signals Control Register (IPU1_DI0_SSC)	32	R/W	0000_0000h	37.5.223/ 3345
264_0164	DI0 Polarity Register (IPU1_DI0_POL)	32	R/W	0000_0000h	37.5.224/ 3347

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
264_0168	DI0 Active Window 0 Register (IPU1_DI0_AW0)	32	R/W	0000_0000h	37.5.225/ 3348
264_016C	DI0 Active Window 1 Register (IPU1_DI0_AW1)	32	R/W	0000_0000h	37.5.226/ 3349
264_0170	DI0 Screen Configuration Register (IPU1_DI0_SCR_CONF)	32	R/W	0000_0000h	37.5.227/ 3350
264_0174	DI0 Status Register (IPU1_DI0_STAT)	32	R	0000_0005h	37.5.228/ 3351
264_8000	DI1General Register (IPU1_DI1_GENERAL)	32	R/W	0020_0000h	37.5.229/ 3353
264_8004	DI1 Base Sync Clock Gen 0 Register (IPU1_DI1_BS_CLKGEN0)	32	R/W	0000_0000h	37.5.230/ 3355
264_8008	DI1 Base Sync Clock Gen 1 Register (IPU1_DI1_BS_CLKGEN1)	32	R/W	0000_0000h	37.5.231/ 3356
264_800C	DI1 Sync Wave Gen 1 Register 0 (IPU1_DI1_SW_GEN0_1)	32	R/W	0000_0000h	37.5.232/ 3356
264_8010	DI1 Sync Wave Gen 2 Register 0 (IPU1_DI1_SW_GEN0_2)	32	R/W	0000_0000h	37.5.233/ 3358
264_8014	DI1 Sync Wave Gen 3 Register 0 (IPU1_DI1_SW_GEN0_3)	32	R/W	0000_0000h	37.5.234/ 3359
264_8018	DI1 Sync Wave Gen 4 Register 0 (IPU1_DI1_SW_GEN0_4)	32	R/W	0000_0000h	37.5.235/ 3360
264_801C	DI1 Sync Wave Gen 5 Register 0 (IPU1_DI1_SW_GEN0_5)	32	R/W	0000_0000h	37.5.236/ 3361
264_8020	DI1 Sync Wave Gen 6 Register 0 (IPU1_DI1_SW_GEN0_6)	32	R/W	0000_0000h	37.5.237/ 3363
264_8024	DI1 Sync Wave Gen 7 Register 0 (IPU1_DI1_SW_GEN0_7)	32	R/W	0000_0000h	37.5.238/ 3364
264_8028	DI1 Sync Wave Gen 8 Register 0 (IPU1_DI1_SW_GEN0_8)	32	R/W	0000_0000h	37.5.239/ 3365
264_802C	DI1Sync Wave Gen 9 Register 0 (IPU1_DI1_SW_GEN0_9)	32	R/W	0000_0000h	37.5.240/ 3366
264_8030	DI1 Sync Wave Gen 1 Register 1 (IPU1_DI1_SW_GEN1_1)	32	R/W	0000_0000h	37.5.241/ 3368
264_8034	DI1 Sync Wave Gen 2 Register 1 (IPU1_DI1_SW_GEN1_2)	32	R/W	0000_0000h	37.5.242/ 3370
264_8038	DI1 Sync Wave Gen 3 Register 1 (IPU1_DI1_SW_GEN1_3)	32	R/W	0000_0000h	37.5.243/ 3372
264_803C	DI1 Sync Wave Gen 4 Register 1 (IPU1_DI1_SW_GEN1_4)	32	R/W	0000_0000h	37.5.244/ 3374
264_8040	DI1 Sync Wave Gen 5 Register 1 (IPU1_DI1_SW_GEN1_5)	32	R/W	0000_0000h	37.5.245/ 3376
264_8044	DI1 Sync Wave Gen 6 Register 1 (IPU1_DI1_SW_GEN1_6)	32	R/W	0000_0000h	37.5.246/ 3378

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
264_8048	DI1Sync Wave Gen 7 Register 1 (IPU1_DI1_SW_GEN1_7)	32	R/W	0000_0000h	37.5.247/ 3380
264_804C	DI1 Sync Wave Gen 8 Register 1 (IPU1_DI1_SW_GEN1_8)	32	R/W	0000_0000h	37.5.248/ 3382
264_8050	DI1 Sync Wave Gen 9 Register 1 (IPU1_DI1_SW_GEN1_9)	32	R/W	0000_0000h	37.5.249/ 3384
264_8054	DI1 Sync Assistance Gen Register (IPU1_DI1_SYNC_AS_GEN)	32	R/W	0000_0000h	37.5.250/ 3385
264_8058	DI1 Data Wave Gen <i> Register (IPU1_DI1_DW_GEN_i)	32	R/W	0000_0000h	37.5.251/ 3386
264_8088	DI1 Data Wave Set 0 <i> Register (IPU1_DI1_DW_SET0_i)	32	R/W	0000_0000h	37.5.252/ 3389
264_80B8	DI1 Data Wave Set 1 <i> Register (IPU1_DI1_DW_SET1_i)	32	R/W	0000_0000h	37.5.253/ 3389
264_80E8	DI1 Data Wave Set 2 <i> Register (IPU1_DI1_DW_SET2_i)	32	R/W	0000_0000h	37.5.254/ 3390
264_8118	DI1 Data Wave Set 3 <i> Register (IPU1_DI1_DW_SET3_i)	32	R/W	0000_0000h	37.5.255/ 3391
264_8148	DI1 Step Repeat <i> Registers (IPU1_D1_STP_REP_i)	32	R/W	0000_0000h	37.5.256/ 3391
264_8158	DI1Step Repeat 9 Registers (IPU1_DI1_STP_REP_9)	32	R/W	0000_0000h	37.5.257/ 3392
264_815C	DI1 Serial Display Control Register (IPU1_DI1_SER_CONF)	32	R/W	0000_0000h	37.5.258/ 3392
264_8160	DI1 Special Signals Control Register (IPU1_DI1_SSC)	32	R/W	0000_0000h	37.5.259/ 3395
264_8164	DI1 Polarity Register (IPU1_DI1_POL)	32	R/W	0000_0000h	37.5.260/ 3397
264_8168	DI1Active Window 0 Register (IPU1_DI1_AW0)	32	R/W	0000_0000h	37.5.261/ 3398
264_816C	DI1 Active Window 1 Register (IPU1_DI1_AW1)	32	R/W	0000_0000h	37.5.262/ 3399
264_8170	DI1 Screen Configuration Register (IPU1_DI1_SCR_CONF)	32	R/W	0000_0000h	37.5.263/ 3400
264_8174	DI1 Status Register (IPU1_DI1_STAT)	32	R	0000_0005h	37.5.264/ 3401
265_0000	SMFC Mapping Register (IPU1_SMFC_MAP)	32	R/W	0000_0000h	37.5.265/ 3402
265_0004	SMFC Watermark Control Register (IPU1_SMFC_WMC)	32	R/W	0000_09A6h	37.5.266/ 3403
265_0008	SMFC Burst Size Register (IPU1_SMFC_BS)	32	R/W	0000_0000h	37.5.267/ 3405
265_8000	DC Read Channel Configuration Register (IPU1_DC_READ_CH_CONF)	32	R/W	FFFF_0000h	37.5.268/ 3406

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_8004	DC Read Channel Start Address Register (IPU1_DC_READ_SH_ADDR)	32	R/W	0000_0000h	37.5.269/ 3407
265_8008	DC Routine Link Register 0 Channel 0 (IPU1_DC_RL0_CH_0)	32	R/W	0000_0000h	37.5.270/ 3408
265_800C	DC Routine Link Register 1 Channel 0 (IPU1_DC_RL1_CH_0)	32	R/W	0000_0000h	37.5.271/ 3409
265_8010	DC Routine Link Register2 Channel 0 (IPU1_DC_RL2_CH_0)	32	R/W	0000_0000h	37.5.272/ 3410
265_8014	DC Routine Link Registe3 Channel 0 (IPU1_DC_RL3_CH_0)	32	R/W	0000_0000h	37.5.273/ 3411
265_8018	DC Routine Link Register 4 Channel 0 (IPU1_DC_RL4_CH_0)	32	R/W	0000_0000h	37.5.274/ 3412
265_801C	DC Write Channel 1 Configuration Register (IPU1_DC_WR_CH_CONF_1)	32	R/W	0000_0000h	37.5.275/ 3413
265_8020	DC Write Channel 1 Address Configuration Register (IPU1_DC_WR_CH_ADDR_1)	32	R/W	0000_0000h	37.5.276/ 3414
265_8024	DC Routine Link Register 0 Channel 1 (IPU1_DC_RL0_CH_1)	32	R/W	0000_0000h	37.5.277/ 3415
265_8028	DC Routine Link Register 1 Channel 1 (IPU1_DC_RL1_CH_1)	32	R/W	0000_0000h	37.5.278/ 3416
265_8030	DC Routine Link Register 2 Channel 1 (IPU1_DC_RL2_CH_1)	32	R/W	0000_0000h	37.5.279/ 3417
265_8032	DC Routine Link Register 3 Channel 1 (IPU1_DC_RL3_CH_1)	32	R/W	0000_0000h	37.5.280/ 3418
265_8034	DC Routine Link Register 4 Channel 1 (IPU1_DC_RL4_CH_1)	32	R/W	0000_0000h	37.5.281/ 3419
265_8038	DC Write Channel 2 Configuration Register (IPU1_DC_WR_CH_CONF_2)	32	R/W	0000_0000h	37.5.282/ 3420
265_803C	DC Write Channel 2 Address Configuration Register (IPU1_DC_WR_CH_ADDR_2)	32	R/W	0000_0000h	37.5.283/ 3421
265_8040	DC Routine Link Register 0 Channel 2 (IPU1_DC_RL0_CH_2)	32	R/W	0000_0000h	37.5.284/ 3422
265_8044	DC Routine Link Register 1 Channel 2 (IPU1_DC_RL1_CH_2)	32	R/W	0000_0000h	37.5.285/ 3423
265_8048	DC Routine Link Register 2 Channel 2 (IPU1_DC_RL2_CH_2)	32	R/W	0000_0000h	37.5.286/ 3424
265_804C	DC Routine Link Register 3 Channel 2 (IPU1_DC_RL3_CH_2)	32	R/W	0000_0000h	37.5.287/ 3425
265_8050	DC Routine Link Register 4 Channel 2 (IPU1_DC_RL4_CH_2)	32	R/W	0000_0000h	37.5.288/ 3426
265_8054	DC Command Channel 3 Configuration Register (IPU1_DC_CMD_CH_CONF_3)	32	R/W	0000_0000h	37.5.289/ 3426
265_8058	DC Command Channel 4 Configuration Register (IPU1_DC_CMD_CH_CONF_4)	32	R/W	0000_0000h	37.5.290/ 3427

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_805C	DC Write Channel 5Configuration Register (IPU1_DC_WR_CH_CONF_5)	32	R/W	0000_0000h	37.5.291/ 3428
265_8060	DC Write Channel 5Address Configuration Register (IPU1_DC_WR_CH_ADDR_5)	32	R/W	0000_0000h	37.5.292/ 3430
265_8064	DC Routine Link Register 0 Channel 5 (IPU1_DC_RL0_CH_5)	32	R/W	0000_0000h	37.5.293/ 3430
265_8068	DC Routine Link Register 1 Channel 5 (IPU1_DC_RL1_CH_5)	32	R/W	0000_0000h	37.5.294/ 3431
265_806C	DC Routine Link Register 2 Channel 5 (IPU1_DC_RL2_CH_5)	32	R/W	0000_0000h	37.5.295/ 3432
265_8070	DC Routine Link Register3 Channel 5 (IPU1_DC_RL3_CH_5)	32	R/W	0000_0000h	37.5.296/ 3433
265_8074	DC Routine Link Register 4 Channel 5 (IPU1_DC_RL4_CH_5)	32	R/W	0000_0000h	37.5.297/ 3434
265_8078	DC Write Channel 6 Configuration Register (IPU1_DC_WR_CH_CONF_6)	32	R/W	0000_0000h	37.5.298/ 3435
265_807C	DC Write Channel 6 Address Configuration Register (IPU1_DC_WR_CH_ADDR_6)	32	R/W	0000_0000h	37.5.299/ 3436
265_8080	DC Routine Link Register 0Channel 6 (IPU1_DC_RL0_CH_6)	32	R/W	0000_0000h	37.5.300/ 3437
265_8084	DC Routine Link Register 1 Channel 6 (IPU1_DC_RL1_CH_6)	32	R/W	0000_0000h	37.5.301/ 3438
265_8088	DC Routine Link Register 2 Channel 6 (IPU1_DC_RL2_CH_6)	32	R/W	0000_0000h	37.5.302/ 3439
265_808C	DC Routine Link Register 3 Channel 6 (IPU1_DC_RL3_CH_6)	32	R/W	0000_0000h	37.5.303/ 3440
265_8090	DC Routine Link Register 4 Channel 6 (IPU1_DC_RL4_CH_6)	32	R/W	0000_0000h	37.5.304/ 3441
265_8094	DC Write Channel 8 Configuration 1Register (IPU1_DC_WR_CH_CONF1_8)	32	R/W	0000_0000h	37.5.305/ 3442
265_8098	DC Write Channel 8 Configuration 2 Register (IPU1_DC_WR_CH_CONF2_8)	32	R/W	0000_0000h	37.5.306/ 3443
265_809C	DC Routine Link Register 1 Channel 8 (IPU1_DC_RL1_CH_8)	32	R/W	0000_0000h	37.5.307/ 3443
265_80A0	DC Routine Link Register 2 Channel 8 (IPU1_DC_RL2_CH_8)	32	R/W	0000_0000h	37.5.308/ 3444
265_80A4	DC Routine Link Register 3 Channel 8 (IPU1_DC_RL3_CH_8)	32	R/W	0000_0000h	37.5.309/ 3445
265_80A8	DC Routine Link Register 4 Channel 8 (IPU1_DC_RL4_CH_8)	32	R/W	0000_0000h	37.5.310/ 3445
265_80AC	DC Routine Link Register 5 Channel 8 (IPU1_DC_RL5_CH_8)	32	R/W	0000_0000h	37.5.311/ 3446
265_80B0	DC Routine Link Register 6 Channel 8 (IPU1_DC_RL6_CH_8)	32	R/W	0000_0000h	37.5.312/ 3447

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_80B4	DC Write Channel 9 Configuration 1Register (IPU1_DC_WR_CH_CONF1_9)	32	R/W	0000_0000h	37.5.313/ 3447
265_80B8	DC Write Channel 9Configuration 2Register (IPU1_DC_WR_CH_CONF2_9)	32	R/W	0000_0000h	37.5.314/ 3448
265_80BC	DC Routine Link Register 1 Channel 9 (IPU1_DC_RL1_CH_9)	32	R/W	0000_0000h	37.5.315/ 3449
265_80C0	DC Routine Link Register 2 Channel 9 (IPU1_DC_RL2_CH_9)	32	R/W	0000_0000h	37.5.316/ 3449
265_80C4	DC Routine Link Register 3Channel 9 (IPU1_DC_RL3_CH_9)	32	R/W	0000_0000h	37.5.317/ 3450
265_80C8	DC Routine Link Register 4 Channel 9 (IPU1_DC_RL4_CH_9)	32	R/W	0000_0000h	37.5.318/ 3451
265_80CC	DC Routine Link Register 5 Channel 9 (IPU1_DC_RL5_CH_9)	32	R/W	0000_0000h	37.5.319/ 3452
265_80D0	DC Routine Link Register 6 Channel 9 (IPU1_DC_RL6_CH_9)	32	R/W	0000_0000h	37.5.320/ 3452
265_80D4	DC General Register (IPU1_DC_GEN)	32	R/W	0000_0060h	37.5.321/ 3453
265_80D8	DC Display Configuration 1 Register 0 (IPU1_DC_DISP_CONF1_0)	32	R/W	0000_0042h	37.5.322/ 3455
265_80DC	DC Display Configuration 1 Register 1 (IPU1_DC_DISP_CONF1_1)	32	R/W	0000_0042h	37.5.323/ 3456
265_80E0	DC Display Configuration 1 Register 2 (IPU1_DC_DISP_CONF1_2)	32	R/W	0000_0042h	37.5.324/ 3458
265_80E4	DC Display Configuration 1 Register 3 (IPU1_DC_DISP_CONF1_3)	32	R/W	0000_0042h	37.5.325/ 3459
265_80E8	DC Display Configuration 2 Register 0 (IPU1_DC_DISP_CONF2_0)	32	R/W	0000_0000h	37.5.326/ 3460
265_80EC	DC Display Configuration 2 Register 1 (IPU1_DC_DISP_CONF2_1)	32	R/W	0000_0000h	37.5.327/ 3461
265_80F0	DC Display Configuration 2 Register 2 (IPU1_DC_DISP_CONF2_2)	32	R/W	0000_0000h	37.5.328/ 3461
265_80F4	DC Display Configuration 2 Register 3 (IPU1_DC_DISP_CONF2_3)	32	R/W	0000_0000h	37.5.329/ 3461
265_80F8	DC DI0Configuration Register 1 (IPU1_DC_DI0_CONF_1)	32	R/W	0000_0000h	37.5.330/ 3462
265_80FC	DC DI0Configuration Register 2 (IPU1_DC_DI0_CONF_2)	32	R/W	0000_0000h	37.5.331/ 3462
265_8100	DC DI1Configuration Register 1 (IPU1_DC_DI1_CONF_1)	32	R/W	0000_0000h	37.5.332/ 3462
265_8104	DC DI1Configuration Register 2 (IPU1_DC_DI1_CONF_2)	32	R/W	0000_0000h	37.5.333/ 3463
265_8108	DC Mapping Configuration Register 0 (IPU1_DC_MAP_CONF_0)	32	R/W	0000_0000h	37.5.334/ 3463

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_810C	DC Mapping Configuration Register 1 (IPU1_DC_MAP_CONF_1)	32	R/W	0000_0000h	37.5.335/ 3464
265_8110	DC Mapping Configuration Register 2 (IPU1_DC_MAP_CONF_2)	32	R/W	0000_0000h	37.5.336/ 3465
265_8114	DC Mapping Configuration Register 3 (IPU1_DC_MAP_CONF_3)	32	R/W	0000_0000h	37.5.337/ 3466
265_8118	DC Mapping Configuration Register 4 (IPU1_DC_MAP_CONF_4)	32	R/W	0000_0000h	37.5.338/ 3467
265_811C	DC Mapping Configuration Register 5 (IPU1_DC_MAP_CONF_5)	32	R/W	0000_0000h	37.5.339/ 3468
265_8120	DC Mapping Configuration Register 6 (IPU1_DC_MAP_CONF_6)	32	R/W	0000_0000h	37.5.340/ 3469
265_8124	DC Mapping Configuration Register 7 (IPU1_DC_MAP_CONF_7)	32	R/W	0000_0000h	37.5.341/ 3470
265_8128	DC Mapping Configuration Register 8 (IPU1_DC_MAP_CONF_8)	32	R/W	0000_0000h	37.5.342/ 3471
265_812C	DC Mapping Configuration Register 9 (IPU1_DC_MAP_CONF_9)	32	R/W	0000_0000h	37.5.343/ 3472
265_8130	DC Mapping Configuration Register 10 (IPU1_DC_MAP_CONF_10)	32	R/W	0000_0000h	37.5.344/ 3473
265_8134	DC Mapping Configuration Register 11 (IPU1_DC_MAP_CONF_11)	32	R/W	0000_0000h	37.5.345/ 3474
265_8138	DC Mapping Configuration Register 12 (IPU1_DC_MAP_CONF_12)	32	R/W	0000_0000h	37.5.346/ 3475
265_813C	DC Mapping Configuration Register 13 (IPU1_DC_MAP_CONF_13)	32	R/W	0000_0000h	37.5.347/ 3476
265_8140	DC Mapping Configuration Register 14 (IPU1_DC_MAP_CONF_14)	32	R/W	0000_0000h	37.5.348/ 3477
265_8144	DC Mapping Configuration Register 15 (IPU1_DC_MAP_CONF_15)	32	R/W	0000_0000h	37.5.349/ 3478
265_8148	DC Mapping Configuration Register 16 (IPU1_DC_MAP_CONF_16)	32	R/W	0000_0000h	37.5.350/ 3478
265_814C	DC Mapping Configuration Register 17 (IPU1_DC_MAP_CONF_17)	32	R/W	0000_0000h	37.5.351/ 3479
265_8150	DC Mapping Configuration Register 18 (IPU1_DC_MAP_CONF_18)	32	R/W	0000_0000h	37.5.352/ 3480
265_8154	DC Mapping Configuration Register 19 (IPU1_DC_MAP_CONF_19)	32	R/W	0000_0000h	37.5.353/ 3480
265_8158	DC Mapping Configuration Register 20 (IPU1_DC_MAP_CONF_20)	32	R/W	0000_0000h	37.5.354/ 3481
265_815C	DC Mapping Configuration Register 21 (IPU1_DC_MAP_CONF_21)	32	R/W	0000_0000h	37.5.355/ 3482
265_8160	DC Mapping Configuration Register 22 (IPU1_DC_MAP_CONF_22)	32	R/W	0000_0000h	37.5.356/ 3482

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_8164	DC Mapping Configuration Register 23 (IPU1_DC_MAP_CONF_23)	32	R/W	0000_0000h	37.5.357/ 3483
265_8168	DC Mapping Configuration Register 24 (IPU1_DC_MAP_CONF_24)	32	R/W	0000_0000h	37.5.358/ 3484
265_816C	DC Mapping Configuration Register 25 (IPU1_DC_MAP_CONF_25)	32	R/W	0000_0000h	37.5.359/ 3484
265_8170	DC Mapping Configuration Register 26 (IPU1_DC_MAP_CONF_26)	32	R/W	0000_0000h	37.5.360/ 3485
265_8174	DC User General Data Event 0 Register 0 (IPU1_DC_UGDE0_0)	32	R/W	0000_0000h	37.5.361/ 3486
265_8178	DC User General Data Event 0 Register 1 (IPU1_DC_UGDE0_1)	32	R/W	0000_0000h	37.5.362/ 3487
265_817C	DC User General Data Event 0 Register2 (IPU1_DC_UGDE0_2)	32	R/W	0000_0000h	37.5.363/ 3488
265_8180	DC User General Data Event 0 Register 3 (IPU1_DC_UGDE0_3)	32	R/W	0000_0000h	37.5.364/ 3488
265_8184	DC User General Data Event 1Register0 (IPU1_DC_UGDE1_0)	32	R/W	0000_0000h	37.5.365/ 3489
265_8188	DC User General Data Event 1 Register 1 (IPU1_DC_UGDE1_1)	32	R/W	0000_0000h	37.5.366/ 3490
265_818C	DC User General Data Event 1Register 2 (IPU1_DC_UGDE1_2)	32	R/W	0000_0000h	37.5.367/ 3491
265_8190	DC User General Data Event 1Register 3 (IPU1_DC_UGDE1_3)	32	R/W	0000_0000h	37.5.368/ 3491
265_8194	DC User General Data Event 2 Register 0 (IPU1_DC_UGDE2_0)	32	R/W	0000_0000h	37.5.369/ 3492
265_8198	DC User General Data Event 2 Register 1 (IPU1_DC_UGDE2_1)	32	R/W	0000_0000h	37.5.370/ 3493
265_819C	DC User General Data Event 2Register 2 (IPU1_DC_UGDE2_2)	32	R/W	0000_0000h	37.5.371/ 3494
265_81A0	DC User General Data Event 2Register 3 (IPU1_DC_UGDE2_3)	32	R/W	0000_0000h	37.5.372/ 3494
265_81A4	DC User General Data Event 3Register 0 (IPU1_DC_UGDE3_0)	32	R/W	0000_0000h	37.5.373/ 3495
265_81A8	DC User General Data Event 3Register 1 (IPU1_DC_UGDE3_1)	32	R/W	0000_0000h	37.5.374/ 3496
265_81AC	DC User General Data Event 3Register 2 (IPU1_DC_UGDE3_2)	32	R/W	0000_0000h	37.5.375/ 3497
265_81B0	DC User General Data Event 3Register 2 (IPU1_DC_UGDE3_3)	32	R/W	0000_0000h	37.5.376/ 3497
265_81B4	DC Low Level Access Control Register 0 (IPU1_DC_LLA0)	32	R/W	0000_0000h	37.5.377/ 3497
265_81B8	DC Low Level Access Control Register 1 (IPU1_DC_LLA1)	32	R/W	0000_0000h	37.5.378/ 3498

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
265_81BC	DC Read Low Level Read Access Control Register 0 (IPU1_DC_R_LLA0)	32	R/W	0000_0000h	37.5.379/ 3498
265_81C0	DC Read Low Level Read Access Control Register1 (IPU1_DC_R_LLA1)	32	R/W	0000_0000h	37.5.380/ 3499
265_81C4	DC Write Channel 5 Configuration Register (IPU1_DC_WR_CH_ADDR_5_ALT)	32	R/W	0000_0000h	37.5.381/ 3499
265_81C8	DC Status Register (IPU1_DC_STAT)	32	R	0000_00AAh	37.5.382/ 3501
266_0000	DMFC Read Channel Register (IPU1_DMFC_RD_CHAN)	32	R/W	0000_0200h	37.5.383/ 3503
266_0004	DMFC Write Channel Register (IPU1_DMFC_WR_CHAN)	32	R/W	0000_0000h	37.5.384/ 3505
266_0008	DMFC Write Channel Definition Register (IPU1_DMFC_WR_CHAN_DEF)	32	R/W	2020_2020h	37.5.385/ 3508
266_000C	DMFC Display Processor Channel Register (IPU1_DMFC_DP_CHAN)	32	R/W	0000_0000h	37.5.386/ 3510
266_0010	DMFC Display Processor Channel Definition Register (IPU1_DMFC_DP_CHAN_DEF)	32	R/W	2020_2020h	37.5.387/ 3513
266_0014	DMFC General 1 Register (IPU1_DMFC_GENERAL_1)	32	R/W	0000_0003h	37.5.388/ 3515
266_0018	DMFC General 2 Register (IPU1_DMFC_GENERAL_2)	32	R/W	0000_0000h	37.5.389/ 3517
266_001C	DMFC IC Interface Control Register (IPU1_DMFC_IC_CTRL)	32	R/W	0000_0002h	37.5.390/ 3518
266_0020	DMFC Write Channel Alternate Register (IPU1_DMFC_WR_CHAN_ALT)	32	R/W	0000_0000h	37.5.391/ 3519
266_0024	DMFC Write Channel Definition Alternate Register (IPU1_DMFC_WR_CHAN_DEF_ALT)	32	R/W	0000_2000h	37.5.392/ 3520
266_0028	DMFC MFC Display Processor Channel Alternate Register (IPU1_DMFC_DP_CHAN_ALT)	32	R/W	0000_0000h	37.5.393/ 3521
266_002C	DMFC Display Channel Definition Alternate Register (IPU1_DMFC_DP_CHAN_DEF_ALT)	32	R/W	2020_0020h	37.5.394/ 3524
266_0030	DMFC General 1 Alternate Register (IPU1_DMFC_GENERAL1_ALT)	32	R/W	0000_0000h	37.5.395/ 3526
266_0034	DMFC Status Register (IPU1_DMFC_STAT)	32	R	02FF_F000h	37.5.396/ 3528
266_8000	VDI Field Size Register (IPU1_VDI_FSIZE)	32	R/W	0000_0000h	37.5.397/ 3529
266_8004	VDI Control Register (IPU1_VDI_C)	32	R/W	0000_0000h	37.5.398/ 3530
266_8008	VDI Control Register 2 (IPU1_VDI_C2_)	32	R/W	0000_0000h	37.5.399/ 3532
266_800C	VDI Combining Parameters Register 1 (IPU1_VDI_CMDP_1)	32	R/W	0000_0000h	37.5.400/ 3533

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
266_8010	VDI Combining Parameters Register 2 (IPU1_VDI_CMDP_2)	32	R/W	0000_0000h	37.5.401/ 3534
266_8014	VDI Plane Size Register 1 (IPU1_VDI_PS_1)	32	R/W	0000_0000h	37.5.402/ 3534
266_8018	VDI Plane Size Register 2 (IPU1_VDI_PS_2)	32	R/W	0000_0000h	37.5.403/ 3535
266_801C	VDI Plane Size Register 3 (IPU1_VDI_PS_3)	32	R/W	0000_0000h	37.5.404/ 3536
266_8020	VDI Plane Size Register 4 (IPU1_VDI_PS_4)	32	R/W	0000_0000h	37.5.405/ 3536
2A0_0000	Configuration Register (IPU2_CONF)	32	R/W	0000_0000h	37.5.1/2959
2A0_0004	SISG Control 0 Register (IPU2_SISG_CTRL0)	32	R/W	0000_0000h	37.5.2/2962
2A0_0008	SISG Control 1 Register (IPU2_SISG_CTRL1)	32	R/W	0000_0000h	37.5.3/2963
2A0_000C	SISG Set<i> Register (IPU2_SISG_SET_i)	32	R/W	0000_0000h	37.5.4/2963
2A0_0024	SISG Clear <i> Register (IPU2_SISG_CLR_i)	32	R/W	0000_0000h	37.5.5/2964
2A0_003C	Interrupt Control Register 1 (IPU2_INT_CTRL_1)	32	R/W	0000_0000h	37.5.6/2964
2A0_0040	Interrupt Control Register 2 (IPU2_INT_CTRL_2)	32	R/W	0000_0000h	37.5.7/2968
2A0_0044	Interrupt Control Register 3 (IPU2_INT_CTRL_3)	32	R/W	0000_0000h	37.5.8/2971
2A0_0048	Interrupt Control Register 4 (IPU2_INT_CTRL_4)	32	R/W	0000_0000h	37.5.9/2975
2A0_004C	Interrupt Control Register 5 (IPU2_INT_CTRL_5)	32	R/W	0000_0000h	37.5.10/ 2978
2A0_0050	Interrupt Control Register 6 (IPU2_INT_CTRL_6)	32	R/W	0000_0000h	37.5.11/ 2983
2A0_0054	Interrupt Control Register 7 (IPU2_INT_CTRL_7)	32	R/W	0000_0000h	37.5.12/ 2986
2A0_0058	Interrupt Control Register 8 (IPU2_INT_CTRL_8)	32	R/W	0000_0000h	37.5.13/ 2988
2A0_005C	Interrupt Control Register 9 (IPU2_INT_CTRL_9)	32	R/W	0000_0000h	37.5.14/ 2990
2A0_0060	Interrupt Control Register 10 (IPU2_INT_CTRL_10)	32	R/W	0000_0000h	37.5.15/ 2992
2A0_0064	Interrupt Control Register 11 (IPU2_INT_CTRL_11)	32	R/W	0000_0000h	37.5.16/ 2994
2A0_0068	Interrupt Control Register 12 (IPU2_INT_CTRL_12)	32	R/W	0000_0000h	37.5.17/ 2997
2A0_006C	Interrupt Control Register 13 (IPU2_INT_CTRL_13)	32	R/W	0000_0000h	37.5.18/ 2999
2A0_0070	Interrupt Control Register 14 (IPU2_INT_CTRL_14)	32	R/W	0000_0000h	37.5.19/ 3003
2A0_0074	Interrupt Control Register15 (IPU2_INT_CTRL_15)	32	R/W	0000_0000h	37.5.20/ 3006
2A0_0078	SDMA Event Control Register 1 (IPU2_SDMA_EVENT_1)	32	R/W	0000_0000h	37.5.21/ 3010

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A0_007C	SDMA Event Control Register 2 (IPU2_SDMA_EVENT_2)	32	R/W	0000_0000h	37.5.22/ 3014
2A0_0080	SDMA Event Control Register 3 (IPU2_SDMA_EVENT_3)	32	R/W	0000_0000h	37.5.23/ 3017
2A0_0084	SDMA Event Control Register 4 (IPU2_SDMA_EVENT_4)	32	R/W	0000_0000h	37.5.24/ 3022
2A0_0088	SDMA Event Control Register 7 (IPU2_SDMA_EVENT_7)	32	R/W	0000_0000h	37.5.25/ 3025
2A0_008C	SDMA Event Control Register 8 (IPU2_SDMA_EVENT_8)	32	R/W	0000_0000h	37.5.26/ 3027
2A0_0090	SDMA Event Control Register 11 (IPU2_SDMA_EVENT_11)	32	R/W	0000_0000h	37.5.27/ 3028
2A0_0094	SDMA Event Control Register 12 (IPU2_SDMA_EVENT_12)	32	R/W	0000_0000h	37.5.28/ 3031
2A0_0098	SDMA Event Control Register 13 (IPU2_SDMA_EVENT_13)	32	R/W	0000_0000h	37.5.29/ 3033
2A0_009C	SDMA Event Control Register 14 (IPU2_SDMA_EVENT_14)	32	R/W	0000_0000h	37.5.30/ 3037
2A0_00A0	Shadow Registers Memory Priority 1 Register (IPU2_SRM_PRI1)	32	R/W	0000_0100h	37.5.31/ 3040
2A0_00A4	Shadow Registers Memory Priority 2 Register (IPU2_SRM_PRI2)	32	R/W	0605_0803h	37.5.32/ 3041
2A0_00A8	FSU Processing Flow 1 Register (IPU2_FS_PROC_FLOW1)	32	R/W	0000_0000h	37.5.33/ 3043
2A0_00AC	FSU Processing Flow 2 Register (IPU2_FS_PROC_FLOW2)	32	R/W	0000_0000h	37.5.34/ 3047
2A0_00B0	FSU Processing Flow 3 Register (IPU2_FS_PROC_FLOW3)	32	R/W	0000_0000h	37.5.35/ 3050
2A0_00B4	FSU Displaying Flow 1 Register (IPU2_FS_DISP_FLOW1)	32	R/W	0000_0000h	37.5.36/ 3053
2A0_00B8	FSU Displaying Flow 2 Register (IPU2_FS_DISP_FLOW2)	32	R/W	0000_0000h	37.5.37/ 3056
2A0_00BC	SKIP Register (IPU2_SKIP)	32	R/W	0000_0000h	37.5.38/ 3058
2A0_00C4	Display General Control Register (IPU2_DISP_GEN)	32	R/W	0040_0000h	37.5.39/ 3060
2A0_00C8	Display Alternate Flow Control Register 1 (IPU2_DISP_ALT1)	32	R/W	0040_0000h	37.5.40/ 3063
2A0_00CC	Display Alternate Flow Control Register 2 (IPU2_DISP_ALT2)	32	R/W	0000_0000h	37.5.41/ 3064
2A0_00D0	Display Alternate Flow Control Register 3 (IPU2_DISP_ALT3)	32	R/W	0040_0000h	37.5.42/ 3065
2A0_00D4	Display Alternate Flow Control Register 4 (IPU2_DISP_ALT4)	32	R/W	0000_0000h	37.5.43/ 3067

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A0_00DC	Memory Reset Control Register (IPU2_MEM_RST)	32	R/W	0000_0000h	37.5.44/ 3068
2A0_00E0	Power Modes Control Register (IPU2_PM)	32	R/W	0810_0810h	37.5.45/ 3070
2A0_00E4	General Purpose Register (IPU2_GPR)	32	R/W	0000_0000h	37.5.46/ 3073
2A0_0150	Channel Double Buffer Mode Select 0 Register (IPU2_CH_DB_MODE_SEL0)	32	R/W	0000_0000h	37.5.47/ 3075
2A0_0154	Channel Double Buffer Mode Select 1 Register (IPU2_CH_DB_MODE_SEL1)	32	R/W	0000_0000h	37.5.48/ 3079
2A0_0168	Alternate Channel Double Buffer Mode Select 0 Register (IPU2_ALT_CH_DB_MODE_SEL0)	32	R/W	0000_0000h	37.5.49/ 3082
2A0_016C	Alternate Channel Double Buffer Mode Select1 Register (IPU2_ALT_CH_DB_MODE_SEL1)	32	R/W	0000_0000h	37.5.50/ 3084
2A0_0178	Alternate Channel Triple Buffer Mode Select 0 Register (IPU2_ALT_CH_TRB_MODE_SEL0)	32	R/W	0000_0000h	37.5.51/ 3085
2A0_0200	Interrupt Status Register 1 (IPU2_INT_STAT_1)	32	w1c	0000_0000h	37.5.52/ 3088
2A0_0204	Interrupt Status Register2 (IPU2_INT_STAT_2)	32	w1c	0000_0000h	37.5.53/ 3093
2A0_0208	Interrupt Status Register 3 (IPU2_INT_STAT_3)	32	w1c	0000_0000h	37.5.54/ 3096
2A0_020C	Interrupt Status Register 4 (IPU2_INT_STAT_4)	32	w1c	0000_0000h	37.5.55/ 3100
2A0_0210	Interrupt Status Register 5 (IPU2_INT_STAT_5)	32	w1c	0000_0000h	37.5.56/ 3103
2A0_0214	Interrupt Status Register 6 (IPU2_INT_STAT_6)	32	w1c	0000_0000h	37.5.57/ 3108
2A0_0218	Interrupt Status Register7 1 (IPU2_INT_STAT_7)	32	w1c	0000_0000h	37.5.58/ 3111
2A0_021C	Interrupt Status Register 8 (IPU2_INT_STAT_8)	32	w1c	0000_0000h	37.5.59/ 3114
2A0_0220	Interrupt Status Register 9 (IPU2_INT_STAT_9)	32	w1c	0000_0000h	37.5.60/ 3117
2A0_0224	Interrupt Status Register 10 (IPU2_INT_STAT_10)	32	w1c	0000_0000h	37.5.61/ 3119
2A0_0228	Interrupt Status Register 11 (IPU2_INT_STAT_11)	32	w1c	0000_0000h	37.5.62/ 3122
2A0_022C	Interrupt Status Register 12 (IPU2_INT_STAT_12)	32	w1c	0000_0000h	37.5.63/ 3126
2A0_0230	Interrupt Status Register 13 (IPU2_INT_STAT_13)	32	w1c	0000_0000h	37.5.64/ 3128
2A0_0234	Interrupt Status Register 14 (IPU2_INT_STAT_14)	32	w1c	0000_0000h	37.5.65/ 3133

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A0_0238	Interrupt Status Register 15 (IPU2_INT_STAT_15)	32	w1c	0000_0000h	37.5.66/ 3136
2A0_023C	Current Buffer Register 0 (IPU2_CUR_BUF_0)	32	R	0000_0000h	37.5.67/ 3140
2A0_0240	Current Buffer Register 1 (IPU2_CUR_BUF_1)	32	R	0000_0000h	37.5.68/ 3145
2A0_0244	Alternate Current Buffer Register 0 (IPU2_ALT_CUR_0)	32	R	0000_0000h	37.5.69/ 3149
2A0_0248	Alternate Current Buffer Register 1 (IPU2_ALT_CUR_1)	32	R	0000_0000h	37.5.70/ 3151
2A0_024C	Shadow Registers Memory Status Register (IPU2_SRM_STAT)	32	R	0000_0000h	37.5.71/ 3154
2A0_0250	Processing Status Tasks Register (IPU2_PROC_TASKS_STAT)	32	R	0000_0000h	37.5.72/ 3156
2A0_0254	Display Tasks Status Register (IPU2_DISP_TASKS_STAT)	32	R	0000_0000h	37.5.73/ 3158
2A0_0258	Triple Current Buffer Register 0 (IPU2_TRIPLE_CUR_BUF_0)	32	R	0000_0000h	37.5.74/ 3160
2A0_025C	Triple Current Buffer Register 1 (IPU2_TRIPLE_CUR_BUF_1)	32	R	0000_0000h	37.5.75/ 3162
2A0_0268	IPU Channels Buffer 0 Ready 0 Register (IPU2_CH_BUF0_RDY0)	32	R/W	0000_0000h	37.5.76/ 3163
2A0_026C	IPU Channels Buffer 0 Ready 1 Register (IPU2_CH_BUF0_RDY1)	32	R/W	0000_0000h	37.5.77/ 3167
2A0_0270	IPU Channels Buffer 1 Ready 0 Register (IPU2_CH_BUF1_RDY0)	32	R/W	0000_0000h	37.5.78/ 3169
2A0_0274	IPU Channels Buffer 1 Ready 1Register (IPU2_CH_BUF1_RDY1)	32	R/W	0000_0000h	37.5.79/ 3172
2A0_0278	IPU Alternate Channels Buffer 0 Ready 0 Register (IPU2_ALT_CH_BUF0_RDY0)	32	R/W	0000_0000h	37.5.80/ 3175
2A0_027C	IPU Alternate Channels Buffer 0 Ready 1 Register (IPU2_ALT_CH_BUF0_RDY1)	32	R/W	0000_0000h	37.5.81/ 3176
2A0_0280	IPU Alternate Channels Buffer1 Ready 0 Register (IPU2_ALT_CH_BUF1_RDY0)	32	R/W	0000_0000h	37.5.82/ 3177
2A0_0284	IPU Alternate Channels Buffer 1 Ready 1 Register (IPU2_ALT_CH_BUF1_RDY1)	32	R/W	0000_0000h	37.5.83/ 3178
2A0_0288	IPU Channels Buffer 2 Ready 0 Register (IPU2_CH_BUF2_RDY0)	32	R/W	0000_0000h	37.5.84/ 3179
2A0_028C	IPU Channels Buffer 2 Ready 1 Register (IPU2_CH_BUF2_RDY1)	32	R/W	0000_0000h	37.5.85/ 3181
2A0_8000	IDMAC Configuration Register (IPU2_IDMAC_CONF)	32	R/W	0000_002Fh	37.5.86/ 3182
2A0_8004	IDMAC Channel Enable 1 Register (IPU2_IDMAC_CH_EN_1)	32	R/W	0000_0000h	37.5.87/ 3184

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A0_8008	IDMAC Channel Enable 2 Register (IPU2_IDMAC_CH_EN_2)	32	R/W	0000_0000h	37.5.88/ 3187
2A0_800C	IDMAC Separate Alpha Indication Register (IPU2_IDMAC_SEP_ALPHA)	32	R/W	0000_0000h	37.5.89/ 3189
2A0_8010	IDMAC Alternate Separate Alpha Indication Register (IPU2_IDMAC_ALT_SEP_ALPHA)	32	R/W	0000_0000h	37.5.90/ 3191
2A0_8014	IDMAC Channel Priority 1 Register (IPU2_IDMAC_CH_PRI_1)	32	R/W	0000_0000h	37.5.91/ 3193
2A0_8018	IDMAC Channel Priority 2 Register (IPU2_IDMAC_CH_PRI_2)	32	R/W	0000_0000h	37.5.92/ 3196
2A0_801C	IDMAC Channel Watermark Enable 1 Register (IPU2_IDMAC_WM_EN_1)	32	R/W	0000_0000h	37.5.93/ 3198
2A0_8020	IDMAC Channel Watermark Enable 2 Register (IPU2_IDMAC_WM_EN_2)	32	R/W	0000_0000h	37.5.94/ 3200
2A0_8024	IDMAC Channel Lock Enable 1Register (IPU2_IDMAC_LOCK_EN_1)	32	R/W	0000_0000h	37.5.95/ 3201
2A0_8028	IDMAC Channel Lock Enable 2Register (IPU2_IDMAC_LOCK_EN_2)	32	R/W	0000_0000h	37.5.96/ 3203
2A0_802C	IDMAC Channel Alternate Address 0 Register (IPU2_IDMAC_SUB_ADDR_0)	32	R/W	0000_0000h	37.5.97/ 3204
2A0_8030	IDMAC Channel Alternate Address 1 Register (IPU2_IDMAC_SUB_ADDR_1)	32	R/W	0000_0000h	37.5.98/ 3205
2A0_8034	IDMAC Channel Alternate Address 2 Register (IPU2_IDMAC_SUB_ADDR_2)	32	R/W	0000_0000h	37.5.99/ 3206
2A0_8038	IDMAC Channel Alternate Address 3 Register (IPU2_IDMAC_SUB_ADDR_3)	32	R/W	0000_0000h	37.5.100/ 3207
2A0_803C	IDMAC Channel Alternate Address 4 Register (IPU2_IDMAC_SUB_ADDR_4)	32	R/W	0000_0000h	37.5.101/ 3209
2A0_8040	IDMAC Band Mode Enable 1 Register (IPU2_IDMAC_BNDM_EN_1)	32	R/W	0000_0000h	37.5.102/ 3210
2A0_8044	IDMAC Band Mode Enable 2 Register (IPU2_IDMAC_BNDM_EN_2)	32	R/W	0000_0000h	37.5.103/ 3213
2A0_8048	IDMAC Scroll Coordinations Register (IPU2_IDMAC_SC_CORD)	32	R/W	0000_0000h	37.5.104/ 3214
2A0_804C	IDMAC Scroll Coordinations Register 1 (IPU2_IDMAC_SC_CORD_1)	32	R/W	0000_0000h	37.5.105/ 3215
2A0_8100	IDMAC Channel Busy 1 Register (IPU2_IDMAC_CH_BUSY_1)	32	R	0000_0000h	37.5.106/ 3216
2A0_8104	IDMAC Channel Busy 2 Register (IPU2_IDMAC_CH_BUSY_2)	32	R	0000_0000h	37.5.107/ 3222
2A1_8000	DP Common Configuration Sync Flow Register (IPU2_DP_COM_CONF_SYNC)	32	R/W	0000_0000h	37.5.108/ 3226
2A1_8004	DP Graphic Window Control Sync Flow Register (IPU2_DP_Graph_Wind_CTRL_SYNC)	32	R/W	0000_0000h	37.5.109/ 3228

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A1_8008	DP Partial Plane Window Position Sync Flow Register (IPU2_DP_FG_POS_SYNC)	32	R/W	0000_0000h	37.5.110/ 3229
2A1_800C	DP Cursor Position and Size Sync Flow Register (IPU2_DP_CUR_POS_SYNC)	32	R/W	0000_0000h	37.5.111/ 3229
2A1_8010	DP Color Cursor Mapping Sync Flow Register (IPU2_DP_CUR_MAP_SYNC)	32	R/W	0000_0000h	37.5.112/ 3230
2A1_8014	DP Gamma Constants Sync Flow Register i (IPU2_DP_GAMMA_C_SYNC_i)	32	R/W	0000_0000h	37.5.113/ 3231
2A1_8034	DP Gamma Correction Slope Sync Flow Register i (IPU2_DP_GAMMA_S_SYNC_i)	32	R/W	0000_0000h	37.5.114/ 3231
2A1_8044	DP Color Space Conversion Control Sync Flow Registers (IPU2_DP_CSCA_SYNC_i)	32	R/W	0000_0000h	37.5.115/ 3232
2A1_8054	DP Color Conversion Control Sync Flow Register 0 (IPU2_DP_SCS_SYNC_0)	32	R/W	0000_0000h	37.5.116/ 3233
2A1_8058	DP Color Conversion Control Sync Flow Register 1 (IPU2_DP_SCS_SYNC_1)	32	R/W	0000_0000h	37.5.117/ 3233
2A1_805C	DP Cursor Position and Size Alternate Register (IPU2_DP_CUR_POS_ALT)	32	R/W	0000_0000h	37.5.118/ 3234
2A1_8060	DP Common Configuration Async 0 Flow Register (IPU2_DP_COM_CONF_ASYNC0)	32	R/W	0000_0000h	37.5.119/ 3235
2A1_8064	DP Graphic Window Control Async 0 Flow Register (IPU2_DP_GRAPH_WIND_CTRL_ASYNC0)	32	R/W	0000_0000h	37.5.120/ 3237
2A1_8068	DP Partial Plane Window Position Async 0 Flow Register (IPU2_DP_FG_POS_ASYNC0)	32	R/W	0000_0000h	37.5.121/ 3238
2A1_806C	DP Cursor Position and Size Async 0 Flow Register (IPU2_DP_CUR_POS_ASYNC0)	32	R/W	0000_0000h	37.5.122/ 3239
2A1_8070	DP Color Cursor Mapping Async 0 Flow Register (IPU2_DP_CUR_MAP_ASYNC0)	32	R/W	0000_0000h	37.5.123/ 3239
2A1_8074	DP Gamma Constant Async 0 Flow Register i (IPU2_DP_GAMMA_C_ASYNC0_i)	32	R/W	0000_0000h	37.5.124/ 3240
2A1_8094	DP Gamma Correction Slope Async 0 Flow Register i (IPU2_DP_GAMMA_S_ASYNC0_i)	32	R/W	0000_0000h	37.5.125/ 3241
2A1_80A4	DP Color Space Conversion Control Async 0 Flow Register i (IPU2_DP_CSCA_ASYNC0_i)	32	R/W	0000_0000h	37.5.126/ 3241
2A1_80B4	DP Color Conversion Control Async 0 Flow Register 0 (IPU2_DP_CSC_ASYNC0_0)	32	R/W	0000_0000h	37.5.127/ 3242
2A1_80B8	DP Color Conversion Control Async 1 Flow Register (IPU2_DP_CSC_ASYNC_1)	32	R/W	0000_0000h	37.5.128/ 3243
2A1_80BC	DP Common Configuration Async 1 Flow Register (IPU2_DP_COM_CONF_ASYNC1)	32	R/W	0000_0000h	37.5.129/ 3244
2A1_80BC	DP Debug Control Register (IPU2_DP_DEBUG_CNT)	32	R/W	0000_0000h	37.5.130/ 3246
2A1_80C0	DP Graphic Window Control Async 1 Flow Register (IPU2_DP_GRAPH_WIND_CTRL_ASYNC1)	32	R/W	0000_0000h	37.5.131/ 3247

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A1_80C0	DP Debug Status Register (IPU2_DP_DEBUG_STAT)	32	R	0000_0000h	37.5.132/ 3248
2A1_80C4	DP Partial Plane Window Position Async 1 Flow Register (IPU2_DP_FG_POS_ASYNC1)	32	R/W	0000_0000h	37.5.133/ 3250
2A1_80C8	DP Cursor Postion and Size Async 1 Flow Register (IPU2_DP_CUR_POS_ASYNC1)	32	R/W	0000_0000h	37.5.134/ 3250
2A1_80CC	DP Color Cursor Mapping Async 1 Flow Register (IPU2_DP_CUR_MAP_ASYNC1)	32	R/W	0000_0000h	37.5.135/ 3251
2A1_80D0	DP Gamma Constants Async 1 Flow Register i (IPU2_DP_GAMMA_C_ASYNC1_i)	32	R/W	0000_0000h	37.5.136/ 3252
2A1_80F0	DP Gamma Correction Slope Async 1 Flow Register i (IPU2_DP_GAMMA_S_ASYN1_i)	32	R/W	0000_0000h	37.5.137/ 3253
2A1_8100	DP Color Space Converstion Control Async 1 Flow Register i (IPU2_DP_CSCA_ASYNC1_i)	32	R/W	0000_0000h	37.5.138/ 3253
2A1_8110	DP Color Conversion Control Async 1 Flow Register 0 (IPU2_DP_CSC_ASYNC1_0)	32	R/W	0000_0000h	37.5.139/ 3254
2A1_8114	DP Color Conversion Control Async 1 Flow Register 1 (IPU2_DP_CSC_ASYNC1_1)	32	R/W	0000_0000h	37.5.140/ 3255
2A2_0000	IC Configuration Register (IPU2_IC_CONF)	32	R/W	0000_0000h	37.5.141/ 3256
2A2_0004	IC Preprocessing Encoder Resizing Coefficients Register (IPU2_IC_PRP_ENC_RSC)	32	R/W	2000_2000h	37.5.142/ 3258
2A2_0008	IC Preprocessing View-Finder Resizing Coefficients Register (IPU2_IC_PRP_VF_RSC)	32	R/W	2000_2000h	37.5.143/ 3259
2A2_000C	IC Postprocessing Encoder Resizing Coefficients Register (IPU2_IC_PP_RSC)	32	R/W	2000_2000h	37.5.144/ 3260
2A2_0010	IC Combining Parameters Register 1 (IPU2_IC_CMBP_1)	32	R/W	0000_0000h	37.5.145/ 3261
2A2_0014	IC Combining Parameters Register 2 (IPU2_IC_CMBP_2)	32	R/W	0000_0000h	37.5.146/ 3261
2A2_0018	IC IDMAC Parameters 1 Register (IPU2_IC_IDMAC_1)	32	R/W	0000_0000h	37.5.147/ 3262
2A2_001C	IC IDMAC Parameters 2 Register (IPU2_IC_IDMAC_2)	32	R/W	0000_0000h	37.5.148/ 3265
2A2_0020	IC IDMAC Parameters 3Register (IPU2_IC_IDMAC_3)	32	R/W	0000_0000h	37.5.149/ 3266
2A2_0024	IC IDMAC Parameters 4 Register (IPU2_IC_IDMAC_4)	32	R/W	0000_0000h	37.5.150/ 3266
2A3_0000	CSI0 Sensor Configuration Register (IPU2_CSI0_SENS_CONF)	32	R/W	0000_0000h	37.5.151/ 3267
2A3_0004	CSI0 Sense Frame Size Register (IPU2_CSI0_SENS_FRM_SIZE)	32	R/W	0000_0000h	37.5.152/ 3270
2A3_0008	CSI0 Actual Frame Size Register (IPU2_CSI0_ACT_FRM_SIZE)	32	R/W	0000_0000h	37.5.153/ 3270

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A3_000C	CSI0 Output Control Register (IPU2_CSI0_OUT_FRM_CTRL)	32	R/W	0000_0000h	37.5.154/ 3271
2A3_0010	CSIO Test Control Register (IPU2_CSI0_TST_CTRL)	32	R/W	0000_0000h	37.5.155/ 3272
2A3_0014	CSIO CCIR Code Register 1 (IPU2_CSI0_CCIR_CODE_1)	32	R/W	0000_0000h	37.5.156/ 3273
2A3_0018	CSIO CCIR Code Register 2 (IPU2_CSI0_CCIR_CODE_2)	32	R/W	0000_0000h	37.5.157/ 3274
2A3_001C	CSIO CCIR Code Register 3 (IPU2_CSI0_CCIR_CODE_3)	32	R/W	0000_0000h	37.5.158/ 3275
2A3_0020	CSI0 Data Identifier Register (IPU2_CSI0_DI)	32	R/W	FFFF_FFFFh	37.5.159/ 3275
2A3_0024	CSI0 SKIP Register (IPU2_CSI0_SKIP)	32	R/W	0000_0000h	37.5.160/ 3276
2A3_0028	CSI0 Compander Control Register (IPU2_CSIO_CPD_CTRL)	32	R/W	0000_0000h	37.5.161/ 3277
2A3_002C	CSI0 Red Component Compander Constants Register <i> (IPU2_CSIO_CPD_RC_i)	32	R/W	0000_0000h	37.5.162/ 3278
2A3_004C	CSI0 Red Component Compander SLOPE Register <i> (IPU2_CSIO_CPD_RS_i)	32	R/W	0000_0000h	37.5.163/ 3279
2A3_005C	CSI0 GR Component Compander Constants Register <i> (IPU2_CSIO_CPD_GRC_i)	32	R/W	0000_0000h	37.5.164/ 3279
2A3_007C	CSI0 GR Component Compander SLOPE Register <i> (IPU2_CSIO_CPD_GRS_i)	32	R/W	0000_0000h	37.5.165/ 3280
2A3_008C	CSI0 GB Component Compander Constants Register <i> (IPU2_CSIO_CPD_GBC_i)	32	R/W	0000_0000h	37.5.166/ 3281
2A3_00AC	CSI0 GB Component Compander SLOPE Register <i> (IPU2_CSIO_CPD_GBS_i)	32	R/W	0000_0000h	37.5.167/ 3281
2A3_00BC	CSI0 Blue Component Compander Constants Register <i> (IPU2_CSIO_CPD_BC_i)	32	R/W	0000_0000h	37.5.168/ 3282
2A3_00DC	CSI0 Blue Component Compander SLOPE Register <i> (IPU2_CSIO_CPD_BS_i)	32	R/W	0000_0000h	37.5.169/ 3283
2A3_00EC	CSI0 Compander Offset Register 1 (IPU2_CSI0_CPD_OFFSET1)	32	R/W	0000_0000h	37.5.170/ 3283
2A3_00F0	CSI0 Compander Offset Register 2 (IPU2_CSI0_CPD_OFFSET2)	32	R/W	0000_0000h	37.5.171/ 3284
2A3_8000	CSI1 Sensor Configuration Register (IPU2_CSI1_SENS_CONF)	32	R/W	0000_0000h	37.5.172/ 3285
2A3_8004	CSI1 Sense Frame Size Register (IPU2_CSI1_SENS_FRM_SIZE)	32	R/W	0000_0000h	37.5.173/ 3287
2A3_8008	CSI1 Actual Frame Size Register (IPU2_CSI1_ACT_FRM_SIZE)	32	R/W	0000_0000h	37.5.174/ 3288
2A3_800C	CSI1 Output Control Register (IPU2_CSI1_OUT_FRM_CTRL)	32	R/W	0000_0000h	37.5.175/ 3289

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A3_8010	CSI1 Test Control Register (IPU2_CSI1_TST_CTRL)	32	R/W	0000_0000h	37.5.176/ 3290
2A3_8014	CSI1 CCIR Code Register 1 (IPU2_CSI1_CCIR_CODE_1)	32	R/W	0000_0000h	37.5.177/ 3291
2A3_8018	CSI1 CCIR Code Register 2 (IPU2_CSI1_CCIR_CODE_2)	32	R/W	0000_0000h	37.5.178/ 3292
2A3_801C	CSI1 CCIR Code Register 3 (IPU2_CSI1_CCIR_CODE_3)	32	R/W	0000_0000h	37.5.179/ 3293
2A3_8020	CSI1 Data Identifier Register (IPU2_CSI1_DI)	32	R/W	FFFF_FFFFh	37.5.180/ 3293
2A3_8024	CSI1 SKIP Register (IPU2_CSI1_SKIP)	32	R/W	0000_0000h	37.5.181/ 3294
2A3_8028	CSI1 Compander Control Register (IPU2_CSI1_CPD_CTRL)	32	R/W	0000_0000h	37.5.182/ 3295
2A3_802C	CSI1 Red Component Compander Constants Register <i> (IPU2_CSI1_CPD_RC_i)	32	R/W	0000_0000h	37.5.183/ 3296
2A3_804C	CSI1 Red Component Compander SLOPE Register <i> (IPU2_CSI1_CPD_RS_i)	32	R/W	0000_0000h	37.5.184/ 3296
2A3_805C	CSI1 GR Component Compander Constants Register <i> (IPU2_CSI1_CPD_GRC_i)	32	R/W	0000_0000h	37.5.185/ 3297
2A3_807C	CSI1 GR Component Compander SLOPE Register <i> (IPU2_CSI1_CPD_GRS_i)	32	R/W	0000_0000h	37.5.186/ 3298
2A3_808C	CSI1 GB Component Compander Constants Register <i> (IPU2_CSI1_CPD_GBC_i)	32	R/W	0000_0000h	37.5.187/ 3298
2A3_80AC	CSI1 GB Component Compander SLOPE Register <i> (IPU2_CSI1_CPD_GBS_i)	32	R/W	0000_0000h	37.5.188/ 3299
2A3_80BC	CSI1 Blue Component Compander Constants Register <i> (IPU2_CSI1_CPD_BC_i)	32	R/W	0000_0000h	37.5.189/ 3300
2A3_80DC	CSI1 Blue Component Compander SLOPE Register <i> (IPU2_CSI1_CPD_BS_i)	32	R/W	0000_0000h	37.5.190/ 3300
2A3_80EC	CSI1 Compander Offset Register 1 (IPU2_CSI1_CPD_OFFSET1)	32	R/W	0000_0000h	37.5.191/ 3301
2A3_80F0	CSI1 Compander Offset Register 2 (IPU2_CSI1_CPD_OFFSET2)	32	R/W	0000_0000h	37.5.192/ 3302
2A4_0000	DI0 General Register (IPU2_DI0_GENERAL)	32	R/W	0020_0000h	37.5.193/ 3303
2A4_0004	DI0 Base Sync Clock Gen 0 Register (IPU2_DI0_BS_CLKGEN0)	32	R/W	0000_0000h	37.5.194/ 3305
2A4_0008	DI0 Base Sync Clock Gen 1 Register (IPU2_DI0_BS_CLKGEN1)	32	R/W	0000_0000h	37.5.195/ 3306
2A4_000C	DI0 Sync Wave Gen 1 Register 0 (IPU2_DI0_SW_GEN0_1)	32	R/W	0000_0000h	37.5.196/ 3306
2A4_0010	DI0 Sync Wave Gen 2 Register 0 (IPU2_DI0_SW_GEN0_2)	32	R/W	0000_0000h	37.5.197/ 3308

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A4_0014	DI0 Sync Wave Gen 3 Register 0 (IPU2_DI0_SW_GEN0_3)	32	R/W	0000_0000h	37.5.198/ 3309
2A4_0018	DI0 Sync Wave Gen 4 Register 0 (IPU2_DI0_SW_GEN0_4)	32	R/W	0000_0000h	37.5.199/ 3310
2A4_001C	DI0 Sync Wave Gen 5 Register 0 (IPU2_DI0_SW_GEN0_5)	32	R/W	0000_0000h	37.5.200/ 3311
2A4_0020	DI0 Sync Wave Gen 6 Register 0 (IPU2_DI0_SW_GEN0_6)	32	R/W	0000_0000h	37.5.201/ 3313
2A4_0024	DI0 Sync Wave Gen 7 Register 0 (IPU2_DI0_SW_GEN0_7)	32	R/W	0000_0000h	37.5.202/ 3314
2A4_0028	DI0 Sync Wave Gen 8 Register 0 (IPU2_DI0_SW_GEN0_8)	32	R/W	0000_0000h	37.5.203/ 3315
2A4_002C	DI0 Sync Wave Gen 9 Register 0 (IPU2_DI0_SW_GEN0_9)	32	R/W	0000_0000h	37.5.204/ 3316
2A4_0030	DI0 Sync Wave Gen 1 Register 1 (IPU2_DI0_SW_GEN1_1)	32	R/W	0000_0000h	37.5.205/ 3318
2A4_0034	DI0 Sync Wave Gen 2 Register 1 (IPU2_DI0_SW_GEN1_2)	32	R/W	0000_0000h	37.5.206/ 3320
2A4_0038	DI0 Sync Wave Gen 3 Register 1 (IPU2_DI0_SW_GEN1_3)	32	R/W	0000_0000h	37.5.207/ 3322
2A4_003C	DI0 Sync Wave Gen 4 Register 1 (IPU2_DI0_SW_GEN1_4)	32	R/W	0000_0000h	37.5.208/ 3324
2A4_0040	DI0 Sync Wave Gen 5 Register 1 (IPU2_DI0_SW_GEN1_5)	32	R/W	0000_0000h	37.5.209/ 3326
2A4_0044	DI0 Sync Wave Gen 6 Register 1 (IPU2_DI0_SW_GEN1_6)	32	R/W	0000_0000h	37.5.210/ 3328
2A4_0048	DI0 Sync Wave Gen 7 Register 1 (IPU2_DI0_SW_GEN1_7)	32	R/W	0000_0000h	37.5.211/ 3330
2A4_004C	DI0 Sync Wave Gen 8 Register 1 (IPU2_DI0_SW_GEN1_8)	32	R/W	0000_0000h	37.5.212/ 3332
2A4_0050	DI0 Sync Wave Gen 9 Register 1 (IPU2_DI0_SW_GEN1_9)	32	R/W	0000_0000h	37.5.213/ 3334
2A4_0054	DI0 Sync Assistance Gen Register (IPU2_DI0_SYNC_AS_GEN)	32	R/W	0000_0000h	37.5.214/ 3335
2A4_0058	DI0 Data Wave Gen <i> Register (IPU2_DI0_DW_GEN_i)	32	R/W	0000_0000h	37.5.215/ 3336
2A4_0088	DI0 Data Wave Set 0 <i> Register (IPU2_DI0_DW_SET0_i)	32	R/W	0000_0000h	37.5.216/ 3339
2A4_00B8	DI0 Data Wave Set 1 <i> Register (IPU2_DI0_DW_SET1_i)	32	R/W	0000_0000h	37.5.217/ 3339
2A4_00E8	DI0 Data Wave Set 2 <i> Register (IPU2_DI0_DW_SET2_i)	32	R/W	0000_0000h	37.5.218/ 3340
2A4_0118	DI0 Data Wave Set 3 <i> Register (IPU2_DI0_DW_SET3_i)	32	R/W	0000_0000h	37.5.219/ 3341

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A4_0148	DI0 Step Repeat <i> Registers (IPU2_DI0_STP_REP_i)	32	R/W	0000_0000h	37.5.220/ 3341
2A4_0158	DI0 Step Repeat 9 Registers (IPU2_DI0_STP_REP_9)	32	R/W	0000_0000h	37.5.221/ 3342
2A4_015C	DI0 Serial Display Control Register (IPU2_DI0_SER_CONF)	32	R/W	0000_0000h	37.5.222/ 3342
2A4_0160	DI0 Special Signals Control Register (IPU2_DI0_SSC)	32	R/W	0000_0000h	37.5.223/ 3345
2A4_0164	DI0 Polarity Register (IPU2_DI0_POL)	32	R/W	0000_0000h	37.5.224/ 3347
2A4_0168	DI0 Active Window 0 Register (IPU2_DI0_AW0)	32	R/W	0000_0000h	37.5.225/ 3348
2A4_016C	DI0 Active Window 1 Register (IPU2_DI0_AW1)	32	R/W	0000_0000h	37.5.226/ 3349
2A4_0170	DI0 Screen Configuration Register (IPU2_DI0_SCR_CONF)	32	R/W	0000_0000h	37.5.227/ 3350
2A4_0174	DI0 Status Register (IPU2_DI0_STAT)	32	R	0000_0005h	37.5.228/ 3351
2A4_8000	DI1General Register (IPU2_DI1_GENERAL)	32	R/W	0020_0000h	37.5.229/ 3353
2A4_8004	DI1 Base Sync Clock Gen 0 Register (IPU2_DI1_BS_CLKGEN0)	32	R/W	0000_0000h	37.5.230/ 3355
2A4_8008	DI1 Base Sync Clock Gen 1 Register (IPU2_DI1_BS_CLKGEN1)	32	R/W	0000_0000h	37.5.231/ 3356
2A4_800C	DI1 Sync Wave Gen 1 Register 0 (IPU2_DI1_SW_GEN0_1)	32	R/W	0000_0000h	37.5.232/ 3356
2A4_8010	DI1 Sync Wave Gen 2 Register 0 (IPU2_DI1_SW_GEN0_2)	32	R/W	0000_0000h	37.5.233/ 3358
2A4_8014	DI1 Sync Wave Gen 3 Register 0 (IPU2_DI1_SW_GEN0_3)	32	R/W	0000_0000h	37.5.234/ 3359
2A4_8018	DI1 Sync Wave Gen 4 Register 0 (IPU2_DI1_SW_GEN0_4)	32	R/W	0000_0000h	37.5.235/ 3360
2A4_801C	DI1 Sync Wave Gen 5 Register 0 (IPU2_DI1_SW_GEN0_5)	32	R/W	0000_0000h	37.5.236/ 3361
2A4_8020	DI1 Sync Wave Gen 6 Register 0 (IPU2_DI1_SW_GEN0_6)	32	R/W	0000_0000h	37.5.237/ 3363
2A4_8024	DI1 Sync Wave Gen 7 Register 0 (IPU2_DI1_SW_GEN0_7)	32	R/W	0000_0000h	37.5.238/ 3364
2A4_8028	DI1 Sync Wave Gen 8 Register 0 (IPU2_DI1_SW_GEN0_8)	32	R/W	0000_0000h	37.5.239/ 3365
2A4_802C	DI1Sync Wave Gen 9 Register 0 (IPU2_DI1_SW_GEN0_9)	32	R/W	0000_0000h	37.5.240/ 3366
2A4_8030	DI1 Sync Wave Gen 1 Register 1 (IPU2_DI1_SW_GEN1_1)	32	R/W	0000_0000h	37.5.241/ 3368

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A4_8034	DI1 Sync Wave Gen 2 Register 1 (IPU2_DI1_SW_GEN1_2)	32	R/W	0000_0000h	37.5.242/ 3370
2A4_8038	DI1 Sync Wave Gen 3 Register 1 (IPU2_DI1_SW_GEN1_3)	32	R/W	0000_0000h	37.5.243/ 3372
2A4_803C	DI1 Sync Wave Gen 4 Register 1 (IPU2_DI1_SW_GEN1_4)	32	R/W	0000_0000h	37.5.244/ 3374
2A4_8040	DI1 Sync Wave Gen 5 Register 1 (IPU2_DI1_SW_GEN1_5)	32	R/W	0000_0000h	37.5.245/ 3376
2A4_8044	DI1 Sync Wave Gen 6 Register 1 (IPU2_DI1_SW_GEN1_6)	32	R/W	0000_0000h	37.5.246/ 3378
2A4_8048	DI1Sync Wave Gen 7 Register 1 (IPU2_DI1_SW_GEN1_7)	32	R/W	0000_0000h	37.5.247/ 3380
2A4_804C	DI1 Sync Wave Gen 8 Register 1 (IPU2_DI1_SW_GEN1_8)	32	R/W	0000_0000h	37.5.248/ 3382
2A4_8050	DI1 Sync Wave Gen 9 Register 1 (IPU2_DI1_SW_GEN1_9)	32	R/W	0000_0000h	37.5.249/ 3384
2A4_8054	DI1 Sync Assistance Gen Register (IPU2_DI1_SYNC_AS_GEN)	32	R/W	0000_0000h	37.5.250/ 3385
2A4_8058	DI1 Data Wave Gen <i> Register (IPU2_DI1_DW_GEN_i)	32	R/W	0000_0000h	37.5.251/ 3386
2A4_8088	DI1 Data Wave Set 0 <i> Register (IPU2_DI1_DW_SET0_i)	32	R/W	0000_0000h	37.5.252/ 3389
2A4_80B8	DI1 Data Wave Set 1 <i> Register (IPU2_DI1_DW_SET1_i)	32	R/W	0000_0000h	37.5.253/ 3389
2A4_80E8	DI1 Data Wave Set 2 <i> Register (IPU2_DI1_DW_SET2_i)	32	R/W	0000_0000h	37.5.254/ 3390
2A4_8118	DI1 Data Wave Set 3 <i> Register (IPU2_DI1_DW_SET3_i)	32	R/W	0000_0000h	37.5.255/ 3391
2A4_8148	DI1 Step Repeat <i> Registers (IPU2_D1_STP_REP_i)	32	R/W	0000_0000h	37.5.256/ 3391
2A4_8158	DI1Step Repeat 9 Registers (IPU2_DI1_STP_REP_9)	32	R/W	0000_0000h	37.5.257/ 3392
2A4_815C	DI1 Serial Display Control Register (IPU2_DI1_SER_CONF)	32	R/W	0000_0000h	37.5.258/ 3392
2A4_8160	DI1 Special Signals Control Register (IPU2_DI1_SSC)	32	R/W	0000_0000h	37.5.259/ 3395
2A4_8164	DI1 Polarity Register (IPU2_DI1_POL)	32	R/W	0000_0000h	37.5.260/ 3397
2A4_8168	DI1Active Window 0 Register (IPU2_DI1_AW0)	32	R/W	0000_0000h	37.5.261/ 3398
2A4_816C	DI1 Active Window 1 Register (IPU2_DI1_AW1)	32	R/W	0000_0000h	37.5.262/ 3399
2A4_8170	DI1 Screen Configuration Register (IPU2_DI1_SCR_CONF)	32	R/W	0000_0000h	37.5.263/ 3400

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A4_8174	DI1 Status Register (IPU2_DI1_STAT)	32	R	0000_0005h	37.5.264/ 3401
2A5_0000	SMFC Mapping Register (IPU2_SMFC_MAP)	32	R/W	0000_0000h	37.5.265/ 3402
2A5_0004	SMFC Watermark Control Register (IPU2_SMFC_WMC)	32	R/W	0000_09A6h	37.5.266/ 3403
2A5_0008	SMFC Burst Size Register (IPU2_SMFC_BS)	32	R/W	0000_0000h	37.5.267/ 3405
2A5_8000	DC Read Channel Configuration Register (IPU2_DC_READ_CH_CONF)	32	R/W	FFFF_0000h	37.5.268/ 3406
2A5_8004	DC Read Channel Start Address Register (IPU2_DC_READ_SH_ADDR)	32	R/W	0000_0000h	37.5.269/ 3407
2A5_8008	DC Routine Link Register 0 Channel 0 (IPU2_DC_RL0_CH_0)	32	R/W	0000_0000h	37.5.270/ 3408
2A5_800C	DC Routine Link Register 1 Channel 0 (IPU2_DC_RL1_CH_0)	32	R/W	0000_0000h	37.5.271/ 3409
2A5_8010	DC Routine Link Register2 Channel 0 (IPU2_DC_RL2_CH_0)	32	R/W	0000_0000h	37.5.272/ 3410
2A5_8014	DC Routine Link Registe3 Channel 0 (IPU2_DC_RL3_CH_0)	32	R/W	0000_0000h	37.5.273/ 3411
2A5_8018	DC Routine Link Register 4 Channel 0 (IPU2_DC_RL4_CH_0)	32	R/W	0000_0000h	37.5.274/ 3412
2A5_801C	DC Write Channel 1 Configuration Register (IPU2_DC_WR_CH_CONF_1)	32	R/W	0000_0000h	37.5.275/ 3413
2A5_8020	DC Write Channel 1 Address Configuration Register (IPU2_DC_WR_CH_ADDR_1)	32	R/W	0000_0000h	37.5.276/ 3414
2A5_8024	DC Routine Link Register 0 Channel 1 (IPU2_DC_RL0_CH_1)	32	R/W	0000_0000h	37.5.277/ 3415
2A5_8028	DC Routine Link Register 1 Channel 1 (IPU2_DC_RL1_CH_1)	32	R/W	0000_0000h	37.5.278/ 3416
2A5_8030	DC Routine Link Register 2 Channel 1 (IPU2_DC_RL2_CH_1)	32	R/W	0000_0000h	37.5.279/ 3417
2A5_8032	DC Routine Link Register 3 Channel 1 (IPU2_DC_RL3_CH_1)	32	R/W	0000_0000h	37.5.280/ 3418
2A5_8034	DC Routine Link Register 4 Channel 1 (IPU2_DC_RL4_CH_1)	32	R/W	0000_0000h	37.5.281/ 3419
2A5_8038	DC Write Channel 2 Configuration Register (IPU2_DC_WR_CH_CONF_2)	32	R/W	0000_0000h	37.5.282/ 3420
2A5_803C	DC Write Channel 2 Address Configuration Register (IPU2_DC_WR_CH_ADDR_2)	32	R/W	0000_0000h	37.5.283/ 3421
2A5_8040	DC Routine Link Register 0 Channel 2 (IPU2_DC_RL0_CH_2)	32	R/W	0000_0000h	37.5.284/ 3422
2A5_8044	DC Routine Link Register 1 Channel 2 (IPU2_DC_RL1_CH_2)	32	R/W	0000_0000h	37.5.285/ 3423

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A5_8048	DC Routine Link Register 2 Channel 2 (IPU2_DC_RL2_CH_2)	32	R/W	0000_0000h	37.5.286/ 3424
2A5_804C	DC Routine Link Register 3 Channel 2 (IPU2_DC_RL3_CH_2)	32	R/W	0000_0000h	37.5.287/ 3425
2A5_8050	DC Routine Link Register 4 Channel 2 (IPU2_DC_RL4_CH_2)	32	R/W	0000_0000h	37.5.288/ 3426
2A5_8054	DC Command Channel 3 Configuration Register (IPU2_DC_CMD_CH_CONF_3)	32	R/W	0000_0000h	37.5.289/ 3426
2A5_8058	DC Command Channel 4 Configuration Register (IPU2_DC_CMD_CH_CONF_4)	32	R/W	0000_0000h	37.5.290/ 3427
2A5_805C	DC Write Channel 5Configuration Register (IPU2_DC_WR_CH_CONF_5)	32	R/W	0000_0000h	37.5.291/ 3428
2A5_8060	DC Write Channel 5Address Configuration Register (IPU2_DC_WR_CH_ADDR_5)	32	R/W	0000_0000h	37.5.292/ 3430
2A5_8064	DC Routine Link Register 0 Channel 5 (IPU2_DC_RL0_CH_5)	32	R/W	0000_0000h	37.5.293/ 3430
2A5_8068	DC Routine Link Register 1 Channel 5 (IPU2_DC_RL1_CH_5)	32	R/W	0000_0000h	37.5.294/ 3431
2A5_806C	DC Routine Link Register 2 Channel 5 (IPU2_DC_RL2_CH_5)	32	R/W	0000_0000h	37.5.295/ 3432
2A5_8070	DC Routine Link Register3 Channel 5 (IPU2_DC_RL3_CH_5)	32	R/W	0000_0000h	37.5.296/ 3433
2A5_8074	DC Routine Link Register 4 Channel 5 (IPU2_DC_RL4_CH_5)	32	R/W	0000_0000h	37.5.297/ 3434
2A5_8078	DC Write Channel 6 Configuration Register (IPU2_DC_WR_CH_CONF_6)	32	R/W	0000_0000h	37.5.298/ 3435
2A5_807C	DC Write Channel 6 Address Configuration Register (IPU2_DC_WR_CH_ADDR_6)	32	R/W	0000_0000h	37.5.299/ 3436
2A5_8080	DC Routine Link Register 0Channel 6 (IPU2_DC_RL0_CH_6)	32	R/W	0000_0000h	37.5.300/ 3437
2A5_8084	DC Routine Link Register 1 Channel 6 (IPU2_DC_RL1_CH_6)	32	R/W	0000_0000h	37.5.301/ 3438
2A5_8088	DC Routine Link Register 2 Channel 6 (IPU2_DC_RL2_CH_6)	32	R/W	0000_0000h	37.5.302/ 3439
2A5_808C	DC Routine Link Register 3 Channel 6 (IPU2_DC_RL3_CH_6)	32	R/W	0000_0000h	37.5.303/ 3440
2A5_8090	DC Routine Link Register 4 Channel 6 (IPU2_DC_RL4_CH_6)	32	R/W	0000_0000h	37.5.304/ 3441
2A5_8094	DC Write Channel 8 Configuration 1Register (IPU2_DC_WR_CH_CONF1_8)	32	R/W	0000_0000h	37.5.305/ 3442
2A5_8098	DC Write Channel 8 Configuration 2 Register (IPU2_DC_WR_CH_CONF2_8)	32	R/W	0000_0000h	37.5.306/ 3443
2A5_809C	DC Routine Link Register 1 Channel 8 (IPU2_DC_RL1_CH_8)	32	R/W	0000_0000h	37.5.307/ 3443

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A5_80A0	DC Routine Link Register 2 Channel 8 (IPU2_DC_RL2_CH_8)	32	R/W	0000_0000h	37.5.308/ 3444
2A5_80A4	DC Routine Link Register 3 Channel 8 (IPU2_DC_RL3_CH_8)	32	R/W	0000_0000h	37.5.309/ 3445
2A5_80A8	DC Routine Link Register 4 Channel 8 (IPU2_DC_RL4_CH_8)	32	R/W	0000_0000h	37.5.310/ 3445
2A5_80AC	DC Routine Link Register 5 Channel 8 (IPU2_DC_RL5_CH_8)	32	R/W	0000_0000h	37.5.311/ 3446
2A5_80B0	DC Routine Link Register 6 Channel 8 (IPU2_DC_RL6_CH_8)	32	R/W	0000_0000h	37.5.312/ 3447
2A5_80B4	DC Write Channel 9 Configuration 1Register (IPU2_DC_WR_CH_CONF1_9)	32	R/W	0000_0000h	37.5.313/ 3447
2A5_80B8	DC Write Channel 9Configuration 2Register (IPU2_DC_WR_CH_CONF2_9)	32	R/W	0000_0000h	37.5.314/ 3448
2A5_80BC	DC Routine Link Register 1 Channel 9 (IPU2_DC_RL1_CH_9)	32	R/W	0000_0000h	37.5.315/ 3449
2A5_80C0	DC Routine Link Register 2 Channel 9 (IPU2_DC_RL2_CH_9)	32	R/W	0000_0000h	37.5.316/ 3449
2A5_80C4	DC Routine Link Register 3Channel 9 (IPU2_DC_RL3_CH_9)	32	R/W	0000_0000h	37.5.317/ 3450
2A5_80C8	DC Routine Link Register 4 Channel 9 (IPU2_DC_RL4_CH_9)	32	R/W	0000_0000h	37.5.318/ 3451
2A5_80CC	DC Routine Link Register 5 Channel 9 (IPU2_DC_RL5_CH_9)	32	R/W	0000_0000h	37.5.319/ 3452
2A5_80D0	DC Routine Link Register 6 Channel 9 (IPU2_DC_RL6_CH_9)	32	R/W	0000_0000h	37.5.320/ 3452
2A5_80D4	DC General Register (IPU2_DC_GEN)	32	R/W	0000_0060h	37.5.321/ 3453
2A5_80D8	DC Display Configuration 1 Register 0 (IPU2_DC_DISP_CONF1_0)	32	R/W	0000_0042h	37.5.322/ 3455
2A5_80DC	DC Display Configuration 1 Register 1 (IPU2_DC_DISP_CONF1_1)	32	R/W	0000_0042h	37.5.323/ 3456
2A5_80E0	DC Display Configuration 1 Register 2 (IPU2_DC_DISP_CONF1_2)	32	R/W	0000_0042h	37.5.324/ 3458
2A5_80E4	DC Display Configuration 1 Register 3 (IPU2_DC_DISP_CONF1_3)	32	R/W	0000_0042h	37.5.325/ 3459
2A5_80E8	DC Display Configuration 2 Register 0 (IPU2_DC_DISP_CONF2_0)	32	R/W	0000_0000h	37.5.326/ 3460
2A5_80EC	DC Display Configuration 2 Register 1 (IPU2_DC_DISP_CONF2_1)	32	R/W	0000_0000h	37.5.327/ 3461
2A5_80F0	DC Display Configuration 2 Register 2 (IPU2_DC_DISP_CONF2_2)	32	R/W	0000_0000h	37.5.328/ 3461
2A5_80F4	DC Display Configuration 2 Register 3 (IPU2_DC_DISP_CONF2_3)	32	R/W	0000_0000h	37.5.329/ 3461

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A5_80F8	DC DI0Configuration Register 1 (IPU2_DC_DI0_CONF_1)	32	R/W	0000_0000h	37.5.330/ 3462
2A5_80FC	DC DI0Configuration Register 2 (IPU2_DC_DI0_CONF_2)	32	R/W	0000_0000h	37.5.331/ 3462
2A5_8100	DC DI1Configuration Register 1 (IPU2_DC_DI1_CONF_1)	32	R/W	0000_0000h	37.5.332/ 3462
2A5_8104	DC DI1Configuration Register 2 (IPU2_DC_DI1_CONF_2)	32	R/W	0000_0000h	37.5.333/ 3463
2A5_8108	DC Mapping Configuration Register 0 (IPU2_DC_MAP_CONF_0)	32	R/W	0000_0000h	37.5.334/ 3463
2A5_810C	DC Mapping Configuration Register 1 (IPU2_DC_MAP_CONF_1)	32	R/W	0000_0000h	37.5.335/ 3464
2A5_8110	DC Mapping Configuration Register 2 (IPU2_DC_MAP_CONF_2)	32	R/W	0000_0000h	37.5.336/ 3465
2A5_8114	DC Mapping Configuration Register 3 (IPU2_DC_MAP_CONF_3)	32	R/W	0000_0000h	37.5.337/ 3466
2A5_8118	DC Mapping Configuration Register 4 (IPU2_DC_MAP_CONF_4)	32	R/W	0000_0000h	37.5.338/ 3467
2A5_811C	DC Mapping Configuration Register 5 (IPU2_DC_MAP_CONF_5)	32	R/W	0000_0000h	37.5.339/ 3468
2A5_8120	DC Mapping Configuration Register 6 (IPU2_DC_MAP_CONF_6)	32	R/W	0000_0000h	37.5.340/ 3469
2A5_8124	DC Mapping Configuration Register 7 (IPU2_DC_MAP_CONF_7)	32	R/W	0000_0000h	37.5.341/ 3470
2A5_8128	DC Mapping Configuration Register 8 (IPU2_DC_MAP_CONF_8)	32	R/W	0000_0000h	37.5.342/ 3471
2A5_812C	DC Mapping Configuration Register 9 (IPU2_DC_MAP_CONF_9)	32	R/W	0000_0000h	37.5.343/ 3472
2A5_8130	DC Mapping Configuration Register 10 (IPU2_DC_MAP_CONF_10)	32	R/W	0000_0000h	37.5.344/ 3473
2A5_8134	DC Mapping Configuration Register 11 (IPU2_DC_MAP_CONF_11)	32	R/W	0000_0000h	37.5.345/ 3474
2A5_8138	DC Mapping Configuration Register 12 (IPU2_DC_MAP_CONF_12)	32	R/W	0000_0000h	37.5.346/ 3475
2A5_813C	DC Mapping Configuration Register 13 (IPU2_DC_MAP_CONF_13)	32	R/W	0000_0000h	37.5.347/ 3476
2A5_8140	DC Mapping Configuration Register 14 (IPU2_DC_MAP_CONF_14)	32	R/W	0000_0000h	37.5.348/ 3477
2A5_8144	DC Mapping Configuration Register 15 (IPU2_DC_MAP_CONF_15)	32	R/W	0000_0000h	37.5.349/ 3478
2A5_8148	DC Mapping Configuration Register 16 (IPU2_DC_MAP_CONF_16)	32	R/W	0000_0000h	37.5.350/ 3478
2A5_814C	DC Mapping Configuration Register 17 (IPU2_DC_MAP_CONF_17)	32	R/W	0000_0000h	37.5.351/ 3479

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A5_8150	DC Mapping Configuration Register 18 (IPU2_DC_MAP_CONF_18)	32	R/W	0000_0000h	37.5.352/ 3480
2A5_8154	DC Mapping Configuration Register 19 (IPU2_DC_MAP_CONF_19)	32	R/W	0000_0000h	37.5.353/ 3480
2A5_8158	DC Mapping Configuration Register 20 (IPU2_DC_MAP_CONF_20)	32	R/W	0000_0000h	37.5.354/ 3481
2A5_815C	DC Mapping Configuration Register 21 (IPU2_DC_MAP_CONF_21)	32	R/W	0000_0000h	37.5.355/ 3482
2A5_8160	DC Mapping Configuration Register 22 (IPU2_DC_MAP_CONF_22)	32	R/W	0000_0000h	37.5.356/ 3482
2A5_8164	DC Mapping Configuration Register 23 (IPU2_DC_MAP_CONF_23)	32	R/W	0000_0000h	37.5.357/ 3483
2A5_8168	DC Mapping Configuration Register 24 (IPU2_DC_MAP_CONF_24)	32	R/W	0000_0000h	37.5.358/ 3484
2A5_816C	DC Mapping Configuration Register 25 (IPU2_DC_MAP_CONF_25)	32	R/W	0000_0000h	37.5.359/ 3484
2A5_8170	DC Mapping Configuration Register 26 (IPU2_DC_MAP_CONF_26)	32	R/W	0000_0000h	37.5.360/ 3485
2A5_8174	DC User General Data Event 0 Register 0 (IPU2_DC_UGDE0_0)	32	R/W	0000_0000h	37.5.361/ 3486
2A5_8178	DC User General Data Event 0 Register 1 (IPU2_DC_UGDE0_1)	32	R/W	0000_0000h	37.5.362/ 3487
2A5_817C	DC User General Data Event 0 Register2 (IPU2_DC_UGDE0_2)	32	R/W	0000_0000h	37.5.363/ 3488
2A5_8180	DC User General Data Event 0 Register 3 (IPU2_DC_UGDE0_3)	32	R/W	0000_0000h	37.5.364/ 3488
2A5_8184	DC User General Data Event 1Register0 (IPU2_DC_UGDE1_0)	32	R/W	0000_0000h	37.5.365/ 3489
2A5_8188	DC User General Data Event 1 Register 1 (IPU2_DC_UGDE1_1)	32	R/W	0000_0000h	37.5.366/ 3490
2A5_818C	DC User General Data Event 1Register 2 (IPU2_DC_UGDE1_2)	32	R/W	0000_0000h	37.5.367/ 3491
2A5_8190	DC User General Data Event 1Register 3 (IPU2_DC_UGDE1_3)	32	R/W	0000_0000h	37.5.368/ 3491
2A5_8194	DC User General Data Event 2 Register 0 (IPU2_DC_UGDE2_0)	32	R/W	0000_0000h	37.5.369/ 3492
2A5_8198	DC User General Data Event 2 Register 1 (IPU2_DC_UGDE2_1)	32	R/W	0000_0000h	37.5.370/ 3493
2A5_819C	DC User General Data Event 2Register 2 (IPU2_DC_UGDE2_2)	32	R/W	0000_0000h	37.5.371/ 3494
2A5_81A0	DC User General Data Event 2Register 3 (IPU2_DC_UGDE2_3)	32	R/W	0000_0000h	37.5.372/ 3494
2A5_81A4	DC User General Data Event 3Register 0 (IPU2_DC_UGDE3_0)	32	R/W	0000_0000h	37.5.373/ 3495

Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A5_81A8	DC User General Data Event 3Register 1 (IPU2_DC_UGDE3_1)	32	R/W	0000_0000h	37.5.374/ 3496
2A5_81AC	DC User General Data Event 3Register 2 (IPU2_DC_UGDE3_2)	32	R/W	0000_0000h	37.5.375/ 3497
2A5_81B0	DC User General Data Event 3Register 2 (IPU2_DC_UGDE3_3)	32	R/W	0000_0000h	37.5.376/ 3497
2A5_81B4	DC Low Level Access Control Register 0 (IPU2_DC_LLA0)	32	R/W	0000_0000h	37.5.377/ 3497
2A5_81B8	DC Low Level Access Control Register 1 (IPU2_DC_LLA1)	32	R/W	0000_0000h	37.5.378/ 3498
2A5_81BC	DC Read Low Level Read Access Control Register 0 (IPU2_DC_R_LLA0)	32	R/W	0000_0000h	37.5.379/ 3498
2A5_81C0	DC Read Low Level Read Access Control Register1 (IPU2_DC_R_LLA1)	32	R/W	0000_0000h	37.5.380/ 3499
2A5_81C4	DC Write Channel 5 Configuration Register (IPU2_DC_WR_CH_ADDR_5_ALT)	32	R/W	0000_0000h	37.5.381/ 3499
2A5_81C8	DC Status Register (IPU2_DC_STAT)	32	R	0000_00AAh	37.5.382/ 3501
2A6_0000	DMFC Read Channel Register (IPU2_DMFC_RD_CHAN)	32	R/W	0000_0200h	37.5.383/ 3503
2A6_0004	DMFC Write Channel Register (IPU2_DMFC_WR_CHAN)	32	R/W	0000_0000h	37.5.384/ 3505
2A6_0008	DMFC Write Channel Definition Register (IPU2_DMFC_WR_CHAN_DEF)	32	R/W	2020_2020h	37.5.385/ 3508
2A6_000C	DMFC Display Processor Channel Register (IPU2_DMFC_DP_CHAN)	32	R/W	0000_0000h	37.5.386/ 3510
2A6_0010	DMFC Display Processor Channel Definition Register (IPU2_DMFC_DP_CHAN_DEF)	32	R/W	2020_2020h	37.5.387/ 3513
2A6_0014	DMFC General 1 Register (IPU2_DMFC_GENERAL_1)	32	R/W	0000_0003h	37.5.388/ 3515
2A6_0018	DMFC General 2 Register (IPU2_DMFC_GENERAL_2)	32	R/W	0000_0000h	37.5.389/ 3517
2A6_001C	DMFC IC Interface Control Register (IPU2_DMFC_IC_CTRL)	32	R/W	0000_0002h	37.5.390/ 3518
2A6_0020	DMFC Write Channel Alternate Register (IPU2_DMFC_WR_CHAN_ALT)	32	R/W	0000_0000h	37.5.391/ 3519
2A6_0024	DMFC Write Channel Definition Alternate Register (IPU2_DMFC_WR_CHAN_DEF_ALT)	32	R/W	0000_2000h	37.5.392/ 3520
2A6_0028	DMFC MFC Display Processor Channel Alternate Register (IPU2_DMFC_DP_CHAN_ALT)	32	R/W	0000_0000h	37.5.393/ 3521
2A6_002C	DMFC Display Channel Definition Alternate Register (IPU2_DMFC_DP_CHAN_DEF_ALT)	32	R/W	2020_0020h	37.5.394/ 3524
2A6_0030	DMFC General 1 Alternate Register (IPU2_DMFC_GENERAL1_ALT)	32	R/W	0000_0000h	37.5.395/ 3526

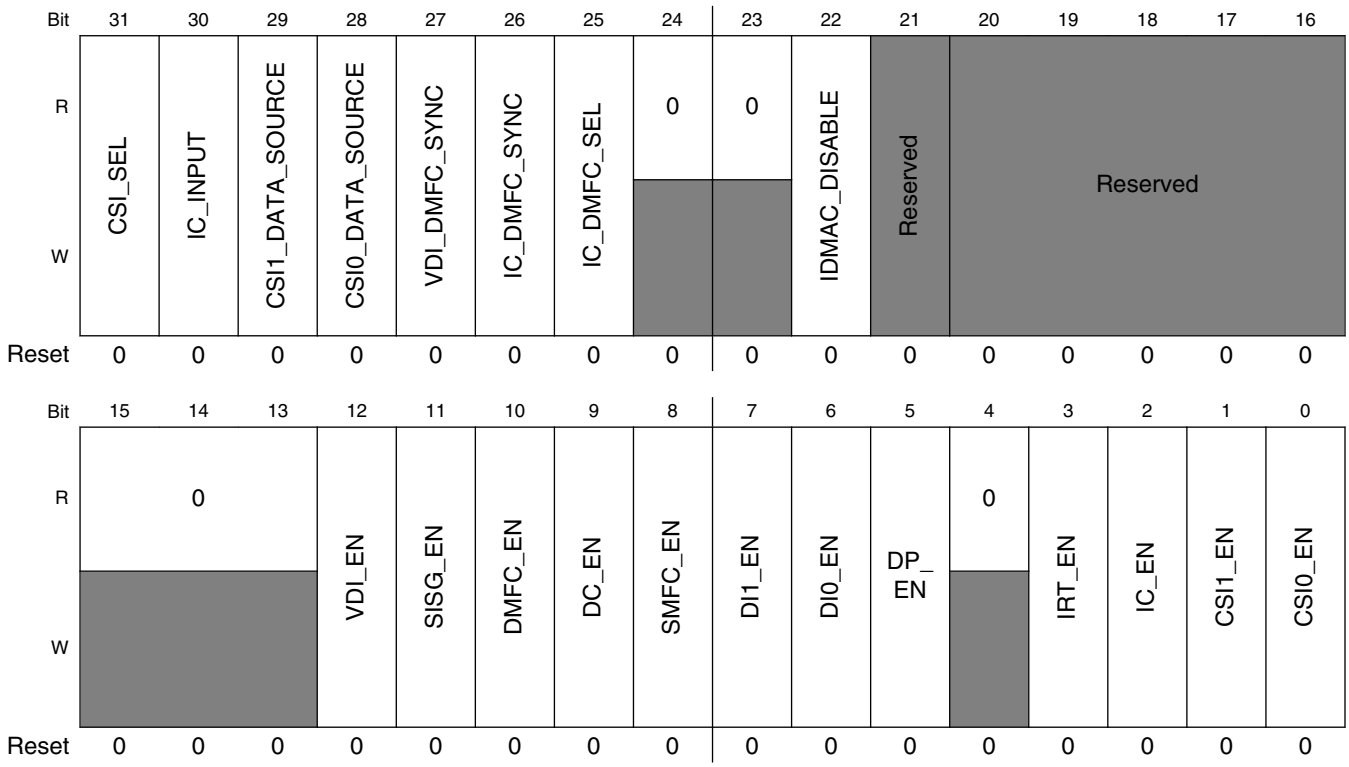
Table continues on the next page...

IPU memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
2A6_0034	DMFC Status Register (IPU2_DMFC_STAT)	32	R	02FF_F000h	37.5.396/ 3528
2A6_8000	VDI Field Size Register (IPU2_VDI_FSIZE)	32	R/W	0000_0000h	37.5.397/ 3529
2A6_8004	VDI Control Register (IPU2_VDI_C)	32	R/W	0000_0000h	37.5.398/ 3530
2A6_8008	VDI Control Register 2 (IPU2_VDI_C2_)	32	R/W	0000_0000h	37.5.399/ 3532
2A6_800C	VDI Combining Parameters Register 1 (IPU2_VDI_CMDP_1)	32	R/W	0000_0000h	37.5.400/ 3533
2A6_8010	VDI Combining Parameters Register 2 (IPU2_VDI_CMDP_2)	32	R/W	0000_0000h	37.5.401/ 3534
2A6_8014	VDI Plane Size Register 1 (IPU2_VDI_PS_1)	32	R/W	0000_0000h	37.5.402/ 3534
2A6_8018	VDI Plane Size Register 2 (IPU2_VDI_PS_2)	32	R/W	0000_0000h	37.5.403/ 3535
2A6_801C	VDI Plane Size Register 3 (IPU2_VDI_PS_3)	32	R/W	0000_0000h	37.5.404/ 3536
2A6_8020	VDI Plane Size Register 4 (IPU2_VDI_PS_4)	32	R/W	0000_0000h	37.5.405/ 3536

37.5.1 Configuration Register (IPUx_CONF)

Address: Base address + 0h offset



IPUx_CONF field descriptions

Field	Description
31 CSI_SEL	CSI select bit; This bit selects manually between the 2 CSI's. This bit defines which CSI is the input to the IC. This bit is effective only if IC_INPUT is bit cleared 0 CSI0 is selected 1 CSI1 is selected
30 IC_INPUT	IC Input select bit. This bit selects manually between the 2 inputs to the IC 0 CSI0/1 is selected; In order to select between the CSIs use the CSI_SEL bit. 1 VDI
29 CSI1_DATA_ SOURCE	CSI1 data Source This bit selects what is the data source for the CSI1. This is a static mux that should not be changed while CSI1 is working. Data is handles differently if the source is MCT (MIPI) or parallel interface. 0 Parallel interface is connected to CSI1 1 MCT (MIPI) is connected to CSI1
28 CSI0_DATA_ SOURCE	CSI0 data Source This bit selects what is the data source for the CSI0. This is a static mux that should not be changed while CSI0 is working. Data is handles differently if the source is MCT (MIPI) or parallel interface.

Table continues on the next page...

IPUx_CONF field descriptions (continued)

Field	Description
	0 Parallel interface is connected to CSI0 1 MCT (MIPI) is connected to CSI0
27 VDI_DMFC_SYNC	This bit enables the direct path VDIC -> IC_VF -> DMFC for sync flow. If this bit is set IC_DMFC_SEL must be set. 0 the flow is disabled 1 the flow is enabled
26 IC_DMFC_SYNC	IC to DMFC Sync flow This bit defines if the direct flow between IC to DMFC is synchronous or asynchronous 0 async flow 1 Sync flow
25 IC_DMFC_SEL	IC to DMFC select Selects the DMAIC_1 (channel 21) channel's connectivity between the IC and the DMFC 0 DMAIC_1 (channel 21) is routed to the IDMAC 1 DMAIC_1 (channel 21) is routed to DMFC In case DMFC was selected the IDMAC_CH_EN[21] must be clear.
24 Reserved	This read-only field is reserved and always has the value 0.
23 Reserved	This read-only field is reserved and always has the value 0.
22 IDMAC_DISABLE	Image DMA controller (IDMAC) disable bit. This bit allows the user to turn off the clock of the IDMAC if the use case permits it. By default the IDMAC is enabled. 0 IDMAC is enabled 1 IDMAC is disabled
21 -	This field is reserved. Reserved.
20–16 -	This field is reserved. Reserved
15–13 Reserved	This read-only field is reserved and always has the value 0.
12 VDI_EN	VDI enable bit. This bit must be cleared if the ISP_EN bit is set. 0 VDIC is disabled 1 VDIC is enabled
11 SISG_EN	Still Image Synchronization Generator (SISG) Enable bit 0 SISG is disabled 1 SISG is enabled
10 DMFC_EN	Display's Multi FIFO Controller sub-block (DMFC) Enable bit 0 DMFC is disabled 1 DMFC is enabled
9 DC_EN	Display Controller sub-block (DC) Enable bit

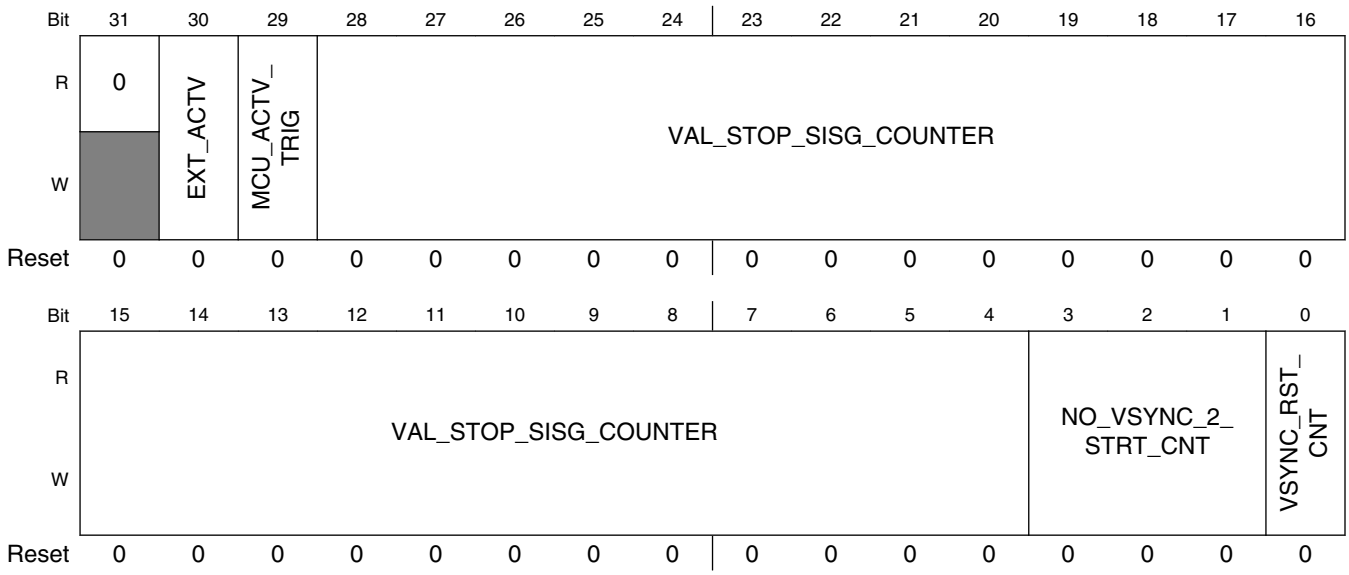
Table continues on the next page...

IPUx_CONF field descriptions (continued)

Field	Description
	0 DC is disabled 1 DC is enabled
8 SMFC_EN	Sensor's Multi FIFO Controller Sub-block (SMFC) Enable bit 0 SMFC is disabled 1 SMFC is enabled
7 DI1_EN	Display Interface Sub-block 1 Enable bit 0 DI1 is disabled 1 DI1 is enabled
6 DI0_EN	Display interface Sub-block 0 Enable bit 0 DI0 is disabled 1 DI0 is enabled
5 DP_EN	Display processor Sub-block Enable bit 0 DP is disabled 1 DP is enabled
4 Reserved	This read-only field is reserved and always has the value 0.
3 IRT_EN	Image Rotation Sub-Block Enable bit 0 IRT is disabled 1 IRT is enabled
2 IC_EN	Image Conversion Sub-Block Enable bit 0 IC is disabled 1 IC is enabled
1 CSI1_EN	Camera Sensor Interface 1 Enable bit 0 CSI1 is disabled 1 CSI1 is enabled
0 CSI0_EN	Camera Sensor Interface 0 Enable bit 0 CSI0 is disabled 1 CSI0 is enabled

37.5.2 SISG Control 0 Register (IPUx_SISG_CTRL0)

Address: Base address + 4h offset

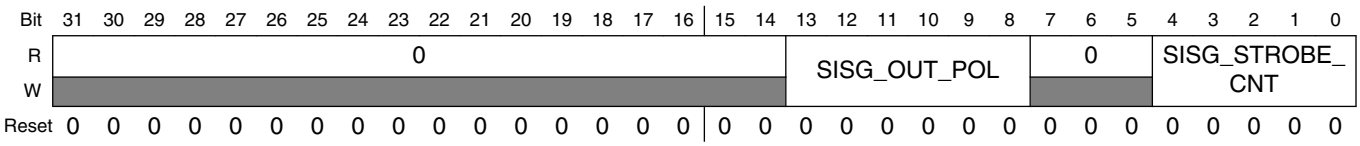


IPUx_SISG_CTRL0 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30 EXT_ACTV	External Active Define if an external active trigger will start the counters. The external active trigger is an input signal to the IPU called ext_actv_trig
29 MCU_ACTV_TRIG	Reserved, should be cleared.
28–4 VAL_STOP_SISG_COUNTER	SISG Stop Counters value. This is a predefined value that stops the SISG counters. The user should write to this field the N-1 value of the desired value.
3–1 NO_VSYNC_2_STRT_CNT	VSYCs to Start Counter This bits define how many VSYNCs signals will be counter before activating the SISG counters. If set to 0 starts immediately. If set to N (1..7) starts after N VSYNCs.
0 VSYNC_RST_CNT	VSYNC Resets counters Defines if the counters are stooped following VSYNC or when the counters reach a pre defined value (VAL_STOP_SISG_COUNTER) 1 The counters are stopped at VSYNC 0 The counters are stooped when the counters reach the VAL_STOP_SISG_COUNTER value.

37.5.3 SISG Control 1 Register (IPUx_SISG_CTRL1)

Address: Base address + 8h offset

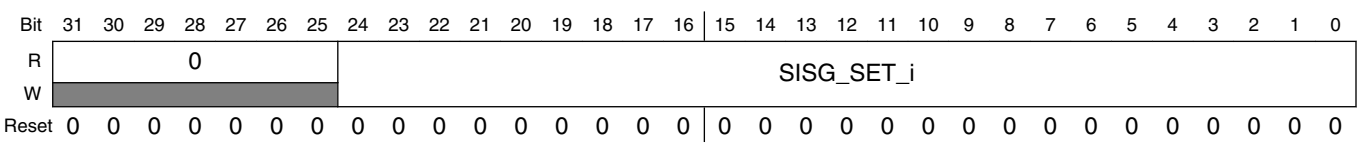


IPUx_SISG_CTRL1 field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13–8 SISG_OUT_POL	SISG_OUT_POL This bits defines the polarity of the SISG output signals 1 active high 0 active low
7–5 Reserved	This read-only field is reserved and always has the value 0.
SISG_STROBE_ CNT	SISG Strobe Count The SISG can repeat the sequence for up to 32 cycles; this is used for generating a train of pulses.

37.5.4 SISG Set<i> Register (IPUx_SISG_SET_i)

Address: Base address + Ch offset



IPUx_SISG_SET_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
SISG_SET_i	SISG SET <i> value These bits define the set value of the SISG counter #<i>

37.5.5 SISG Clear <i> Register (IPUx_SISG_CLR_i)

Address: Base address + 24h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								SISG_CLEAR_i																							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SISG_CLR_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
SISG_CLEAR_i	SISG CLR <i> value These bits define the clear value of the SISG counter #<i>

37.5.6 Interrupt Control Register 1 (IPUx_INT_CTRL_1)

This register contains part of IPU interrupts controls. The controls of EOF (end of frame) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 3Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_EOF_EN_31	0	IDMAC_EOF_EN_29	IDMAC_EOF_EN_28	IDMAC_EOF_EN_27	IDMAC_EOF_EN_26	IDMAC_EOF_EN_25	IDMAC_EOF_EN_24	IDMAC_EOF_EN_23	IDMAC_EOF_EN_22	IDMAC_EOF_EN_21	IDMAC_EOF_EN_20	IDMAC_EOF_EN_19	IDMAC_EOF_EN_18	IDMAC_EOF_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_EOF_EN_15	IDMAC_EOF_EN_14	IDMAC_EOF_EN_13	IDMAC_EOF_EN_12	IDMAC_EOF_EN_11	IDMAC_EOF_EN_10	IDMAC_EOF_EN_9	IDMAC_EOF_EN_8	0		IDMAC_EOF_EN_5	0	IDMAC_EOF_EN_3	IDMAC_EOF_EN_2	IDMAC_EOF_EN_1	IDMAC_EOF_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_1 field descriptions

Field	Description
31 IDMAC_EOF_EN_31	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 IDMAC_EOF_EN_29	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
28 IDMAC_EOF_EN_28	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
27 IDMAC_EOF_EN_27	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_EOF_EN_26	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_EOF_EN_25	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IDMAC_EOF_EN_24	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 IDMAC_EOF_EN_23	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
22 IDMAC_EOF_EN_22	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_1 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
21 IDMAC_EOF_EN_21	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_EOF_EN_20	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_EOF_EN_19	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_EOF_EN_18	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_EOF_EN_17	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_EOF_EN_15	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_EOF_EN_14	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_EOF_EN_13	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_1 field descriptions (continued)

Field	Description
12 IDMAC_EOF_EN_12	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_EOF_EN_11	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_EOF_EN_10	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_EOF_EN_9	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_EOF_EN_8	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
5 IDMAC_EOF_EN_5	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
3 IDMAC_EOF_EN_3	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
2 IDMAC_EOF_EN_2	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

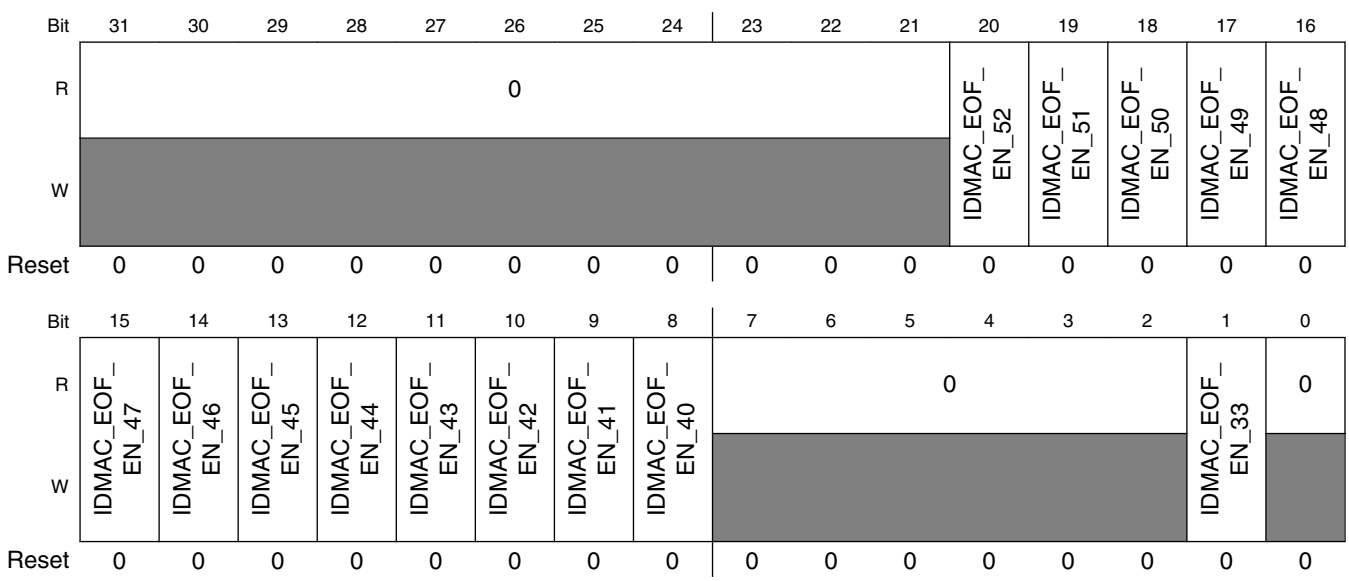
IPUx_INT_CTRL_1 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_EOF_EN_1	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 IDMAC_EOF_EN_0	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.7 Interrupt Control Register 2 (IPUx_INT_CTRL_2)

This register contains part of IPU interrupts controls. The controls of EOF (end of frame) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 40h offset



IPUx_INT_CTRL_2 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_INT_CTRL_2 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_EOF_EN_52	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_EOF_EN_51	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_EOF_EN_50	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_EOF_EN_49	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 IDMAC_EOF_EN_48	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_EOF_EN_47	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_EOF_EN_46	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_EOF_EN_45	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_EOF_EN_44	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_2 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_EOF_EN_43	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_EOF_EN_42	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_EOF_EN_41	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_EOF_EN_40	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_EOF_EN_33	Enable End of Frame of Channel interrupt. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.8 Interrupt Control Register 3 (IPUx_INT_CTRL_3)

This register contains part of IPU interrupts controls. The controls of NFACK (New Frame Ack) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 44h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_NFACK_EN_31	0	IDMAC_NFACK_EN_29	IDMAC_NFACK_EN_28	IDMAC_NFACK_EN_27	IDMAC_NFACK_EN_26	IDMAC_NFACK_EN_25	IDMAC_NFACK_EN_24	IDMAC_NFACK_EN_23	IDMAC_NFACK_EN_22	IDMAC_NFACK_EN_21	IDMAC_NFACK_EN_20	IDMAC_NFACK_EN_19	IDMAC_NFACK_EN_18	IDMAC_NFACK_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFACK_EN_15	IDMAC_NFACK_EN_14	IDMAC_NFACK_EN_13	IDMAC_NFACK_EN_12	IDMAC_NFACK_EN_11	IDMAC_NFACK_EN_10	IDMAC_NFACK_EN_9	IDMAC_NFACK_EN_8	0		IDMAC_NFACK_EN_5	0	IDMAC_NFACK_EN_3	IDMAC_NFACK_EN_2	IDMAC_NFACK_EN_1	IDMAC_NFACK_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_3 field descriptions

Field	Description
31 IDMAC_NFACK_EN_31	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 IDMAC_NFACK_EN_29	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
28 IDMAC_NFACK_EN_28	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_3 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
27 IDMAC_NFACK_EN_27	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_NFACK_EN_26	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_NFACK_EN_25	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IDMAC_NFACK_EN_24	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 IDMAC_NFACK_EN_23	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
22 IDMAC_NFACK_EN_22	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
21 IDMAC_NFACK_EN_21	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_NFACK_EN_20	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_NFACK_EN_19	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_3 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_NFACK_EN_18	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_NFACK_EN_17	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_NFACK_EN_15	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_NFACK_EN_14	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_NFACK_EN_13	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_NFACK_EN_12	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_NFACK_EN_11	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_NFACK_EN_10	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

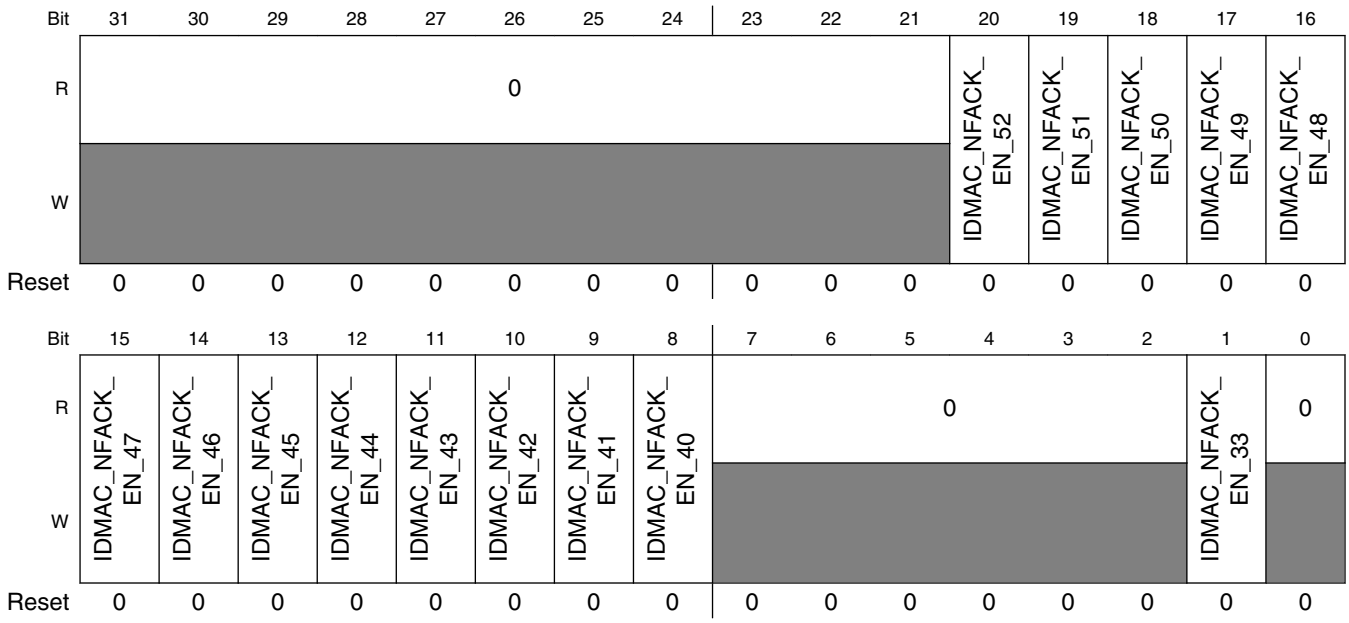
IPUx_INT_CTRL_3 field descriptions (continued)

Field	Description
9 IDMAC_NFACK_EN_9	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_NFACK_EN_8	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
5 IDMAC_NFACK_EN_5	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
3 IDMAC_NFACK_EN_3	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
2 IDMAC_NFACK_EN_2	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_NFACK_EN_1	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 IDMAC_NFACK_EN_0	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.9 Interrupt Control Register 4 (IPUx_INT_CTRL_4)

This register contains part of IPU interrupts controls. The controls of NFACK (New Frame Ack) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 48h offset



IPUx_INT_CTRL_4 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.
20 IDMAC_NFACK_EN_52	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_NFACK_EN_51	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_NFACK_EN_50	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_4 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_NFACK_EN_49	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 IDMAC_NFACK_EN_48	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_NFACK_EN_47	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_NFACK_EN_46	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_NFACK_EN_45	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_NFACK_EN_44	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_NFACK_EN_43	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_NFACK_EN_42	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_NFACK_EN_41	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_4 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_NFACK_EN_40	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_NFACK_EN_33	Enable New Frame Ack of Channel interrupt. This bit is the control of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.10 Interrupt Control Register 5 (IPUx_INT_CTRL_5)

This register contains part of IPU interrupts controls. The controls of the New-frame before end-of-frame indication (NFB4EOF) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 4Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_NFB4EOF_EN_31	0	IDMAC_NFB4EOF_EN_29	IDMAC_NFB4EOF_EN_28	IDMAC_NFB4EOF_EN_27	IDMAC_NFB4EOF_EN_26	IDMAC_NFB4EOF_EN_25	IDMAC_NFB4EOF_EN_24	IDMAC_NFB4EOF_EN_23	IDMAC_NFB4EOF_EN_22	IDMAC_NFB4EOF_EN_21	IDMAC_NFB4EOF_EN_20	IDMAC_NFB4EOF_EN_19	IDMAC_NFB4EOF_EN_18	IDMAC_NFB4EOF_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFB4EOF_EN_15	IDMAC_NFB4EOF_EN_14	IDMAC_NFB4EOF_EN_13	IDMAC_NFB4EOF_EN_12	IDMAC_NFB4EOF_EN_11	IDMAC_NFB4EOF_EN_10	IDMAC_NFB4EOF_EN_9	IDMAC_NFB4EOF_EN_8	0		IDMAC_NFB4EOF_EN_5	0	IDMAC_NFB4EOF_EN_3	IDMAC_NFB4EOF_EN_2	IDMAC_NFB4EOF_EN_1	IDMAC_NFB4EOF_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_5 field descriptions

Field	Description
31 IDMAC_NFB4EOF_EN_31	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 IDMAC_NFB4EOF_EN_29	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_5 field descriptions (continued)

Field	Description
28 IDMAC_ NFB4EOF_EN_ 28	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
27 IDMAC_ NFB4EOF_EN_ 27	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_ NFB4EOF_EN_ 26	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_ NFB4EOF_EN_ 25	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IDMAC_ NFB4EOF_EN_ 24	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 IDMAC_ NFB4EOF_EN_ 23	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
22 IDMAC_ NFB4EOF_EN_ 22	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
21 IDMAC_ NFB4EOF_EN_ 21	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_5 field descriptions (continued)

Field	Description
20 IDMAC_ NFB4EOF_EN_ 20	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_ NFB4EOF_EN_ 19	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_ NFB4EOF_EN_ 18	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_ NFB4EOF_EN_ 17	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_ NFB4EOF_EN_ 15	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_ NFB4EOF_EN_ 14	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_ NFB4EOF_EN_ 13	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_5 field descriptions (continued)

Field	Description
12 IDMAC_ NFB4EOF_EN_ 12	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_ NFB4EOF_EN_ 11	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_ NFB4EOF_EN_ 10	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_ NFB4EOF_EN_9	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_ NFB4EOF_EN_8	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
5 IDMAC_ NFB4EOF_EN_5	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
3 IDMAC_ NFB4EOF_EN_3	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

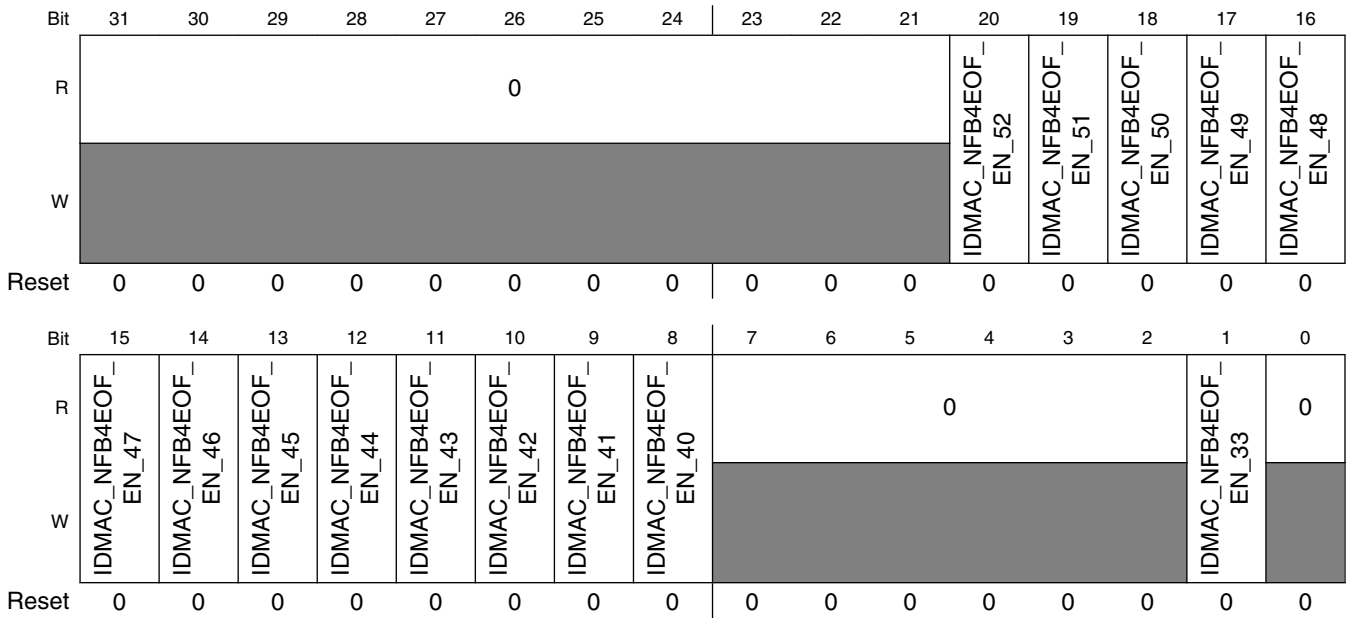
IPUx_INT_CTRL_5 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
2 IDMAC_ NFB4EOF_EN_2	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_ NFB4EOF_EN_1	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 IDMAC_ NFB4EOF_EN_0	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.11 Interrupt Control Register 6 (IPUx_INT_CTRL_6)

This register contains part of IPU interrupts controls. The controls of the New-frame before end-of-frame indication (NFB4EOF_ERR) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 50h offset



IPUx_INT_CTRL_6 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_ NFB4EOF_EN_ 52	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_ NFB4EOF_EN_ 51	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_6 field descriptions (continued)

Field	Description
18 IDMAC_ NFB4EOF_EN_ 50	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_ NFB4EOF_EN_ 49	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 IDMAC_ NFB4EOF_EN_ 48	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_ NFB4EOF_EN_ 47	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_ NFB4EOF_EN_ 46	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_ NFB4EOF_EN_ 45	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_ NFB4EOF_EN_ 44	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_ NFB4EOF_EN_ 43	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

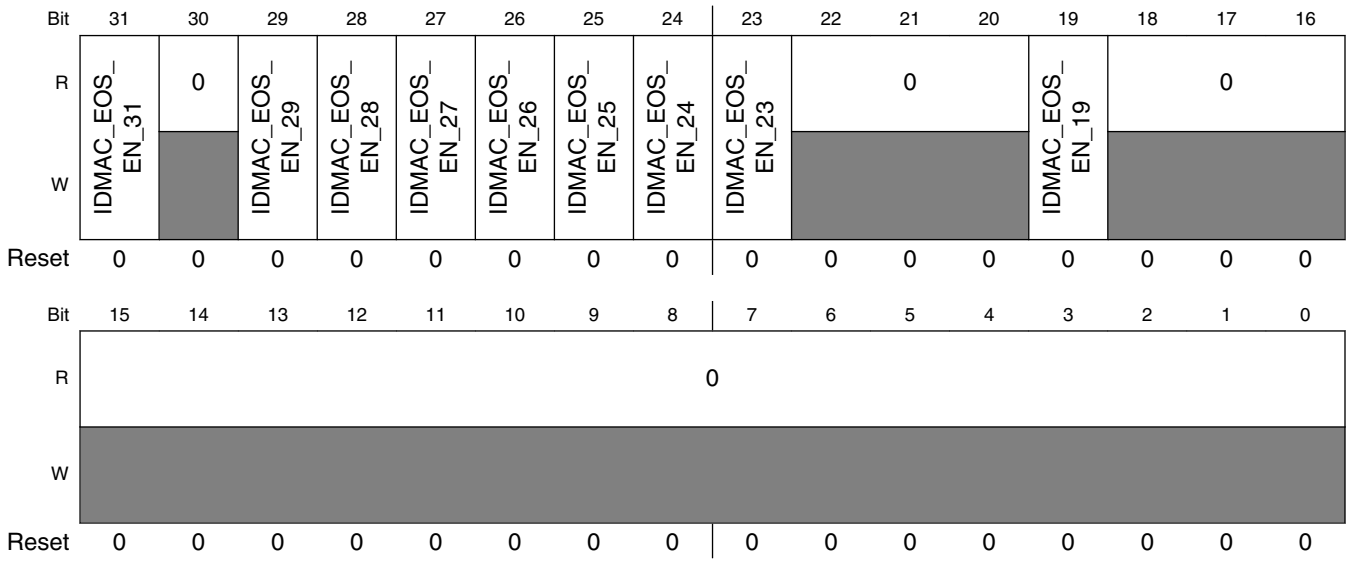
IPUx_INT_CTRL_6 field descriptions (continued)

Field	Description
10 IDMAC_ NFB4EOF_EN_ 42	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_ NFB4EOF_EN_ 41	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_ NFB4EOF_EN_ 40	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_ NFB4EOF_EN_ 33	New Frame before end-of-frame error indication of Channel interrupt. This bit is the control of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.12 Interrupt Control Register 7 (IPUx_INT_CTRL_7)

This register contains part of IPUIPU interrupts controls. The controls of the End-of-Scroll indication (EOS) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 54h offset



IPUx_INT_CTRL_7 field descriptions

Field	Description
31 IDMAC_EOS_EN_31	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 IDMAC_EOS_EN_29	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
28 IDMAC_EOS_EN_28	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

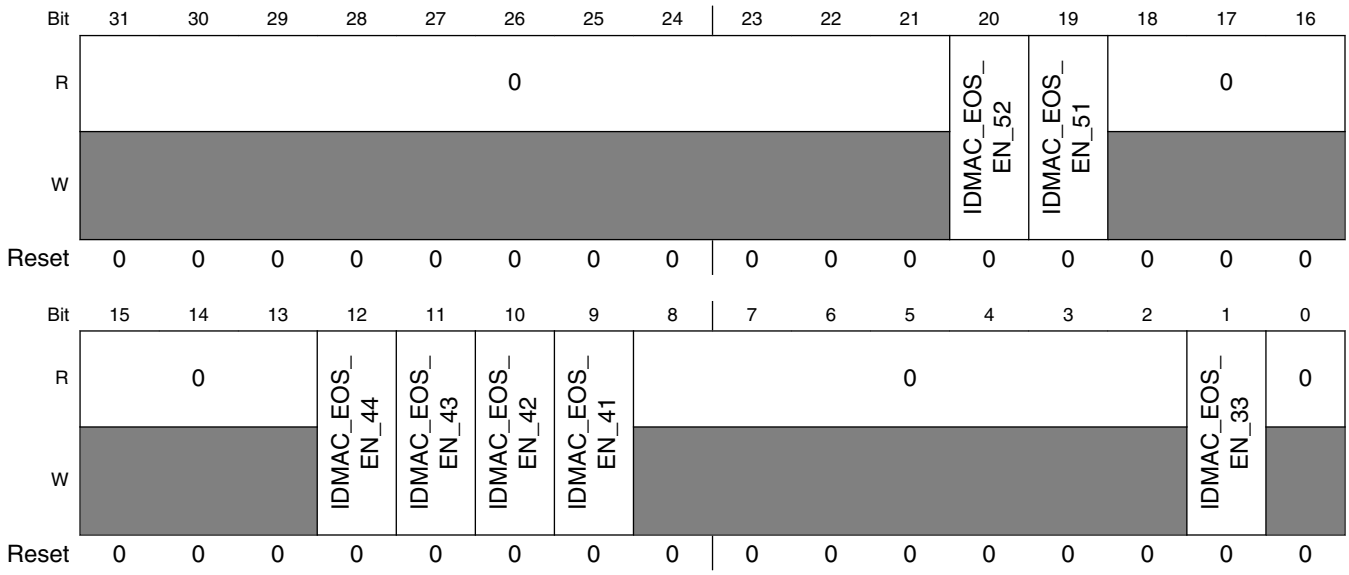
IPUx_INT_CTRL_7 field descriptions (continued)

Field	Description
27 IDMAC_EOS_EN_27	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_EOS_EN_26	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_EOS_EN_25	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IDMAC_EOS_EN_24	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 IDMAC_EOS_EN_23	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
22–20 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_EOS_EN_19	End of Scroll indication of Channel interrupt. This bit is the control of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.13 Interrupt Control Register 8 (IPUx_INT_CTRL_8)

This register contains part of IPU interrupts controls. The controls of the End of Scroll indication (EOS) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 58h offset



IPUx_INT_CTRL_8 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_EOS_EN_52	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_EOS_EN_51	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18–13 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

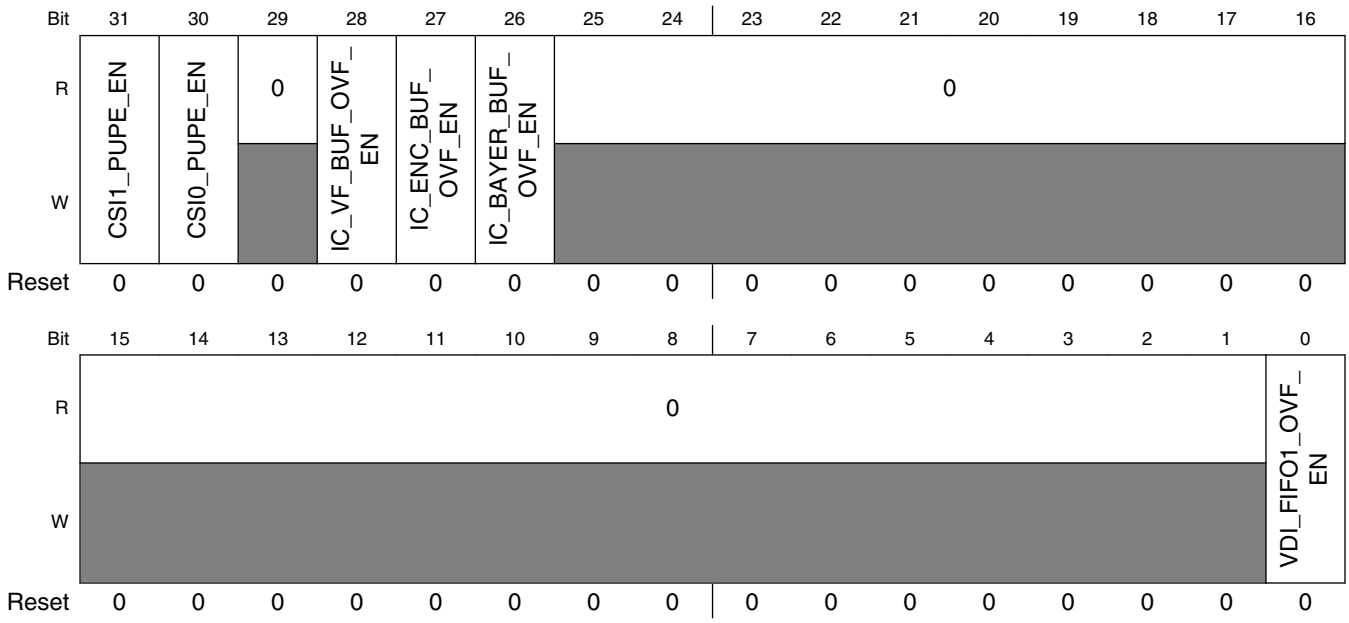
IPUx_INT_CTRL_8 field descriptions (continued)

Field	Description
12 IDMAC_EOS_EN_44	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_EOS_EN_43	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_EOS_EN_42	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_EOS_EN_41	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_EOS_EN_33	End of Scroll of Channel interrupt. This bit is the control of End Of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.14 Interrupt Control Register 9 (IPUx_INT_CTRL_9)

This register contains part of IPU interrupts controls. This register controls error interrupt signals coming from different sub-blocks within

Address: Base address + 5Ch offset



IPUx_INT_CTRL_9 field descriptions

Field	Description
31 CSI1_PUPE_EN	CSI1_PUPE_EN - CSI1 parameters update error interrupt enable. This bit enables an interrupt that is a result of an error generated by the CSI1. The error is generated in case where new frame arrived from the CSI1 before the completion of the CSI1's parameters update by the SRM 0 Interrupt is disabled. 1 Interrupt is enabled.
30 CSI0_PUPE_EN	CSI0_PUPE_EN - CSI0 parameters update error interrupt enable. This bit enables an interrupt that is a result of an error generated by the CSI0. The error is generated in case where new frame arrived from the CSI0 before the completion of the CSI0's parameters update by the SRM 0 Interrupt is disabled. 1 Interrupt is enabled.
29 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

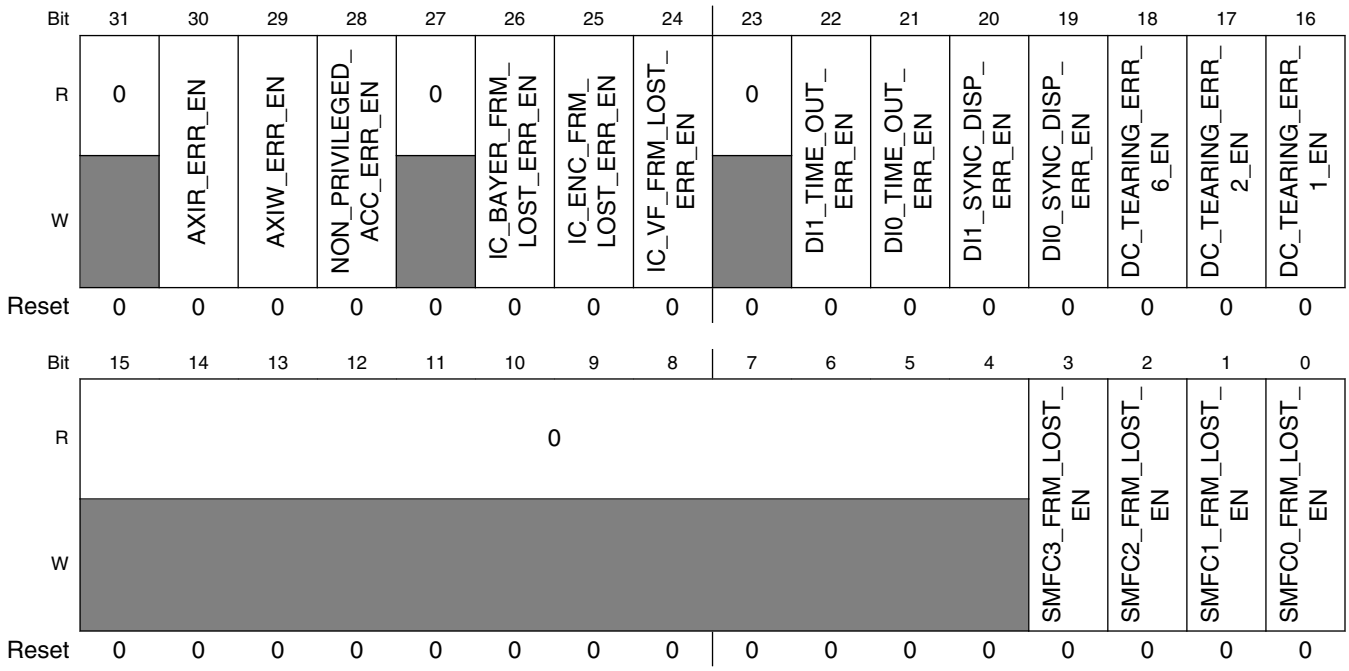
IPUx_INT_CTRL_9 field descriptions (continued)

Field	Description
28 IC_VF_BUF_ OVF_EN	This bit enables an interrupt that is a result of the IC Buffer overflow for view finder coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is disabled. 1 Interrupt is enabled.
27 IC_ENC_BUF_ OVF_EN	This bit enables an interrupt that is a result of the IC Buffer overflow for encoding coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IC_BAYER_ BUF_OVF_EN	This bit enables an interrupt that is a result of the IC Buffer overflow for bayer coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is disabled. 1 Interrupt is enabled.
25–1 Reserved	This read-only field is reserved and always has the value 0.
0 VDI_FIFO1_ OVF_EN	FIFO1 overflow Interrupt1 Enable The VDIC generates FIFO1 overflow interrupt1 when the write pointer of FIFO1 overruns read pointer. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.15 Interrupt Control Register 10 (IPUx_INT_CTRL_10)

This register contains part of IPU interrupts controls. This register controls error interrupt signals coming from different modules within

Address: Base address + 60h offset



IPUx_INT_CTRL_10 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30 AXIR_ERR_EN	This bit enables an interrupt that is a result of AXI read access resulted with error response. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 AXIW_ERR_EN	This bit enables an interrupt that is a result of AXI write access resulted with error response. 0 Interrupt is disabled. 1 Interrupt is enabled.
28 NON_ PRIVILEGED_ ACC_ERR_EN	Non Privileged Access Error interrupt enable. The CPMEM and the DP can be accessed by the ARM platform in privileged mode only HPROT[1] =1. An attempt to access these regions in user mode will issue an interrupt. This bit enables the interrupt. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_10 field descriptions (continued)

Field	Description
27 Reserved	This read-only field is reserved and always has the value 0.
26 IC_BAYER_ FRM_LOST_ ERR_EN	This bit enables an interrupt that is a result of IC's Bayer frame lost. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IC_ENC_FRM_ LOST_ERR_EN	This bit enables an interrupt that is a result of IC's encoding frame lost. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IC_VF_FRM_ LOST_ERR_EN	This bit enables an interrupt that is a result of IC's view finder frame lost. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 Reserved	This read-only field is reserved and always has the value 0.
22 DI1_TIME_OUT_ ERR_EN	DI1 time out error interrupt enable This bit enables the interrupt that is a result of a time out error during a read access via DI1 0 Interrupt is disabled. 1 Interrupt is enabled.
21 DI0_TIME_OUT_ ERR_EN	DI0 time outwore interrupt enable This bit enables the interrupt that is a result of a time out error during a read access via DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
20 DI1_SYNC_ DISP_ERR_EN	DI1 Synchronous display error enable This bit enables the interrupt that is a result of an error during access to a synchronous display via DI1 0 Interrupt is disabled. 1 Interrupt is enabled.
19 DI0_SYNC_ DISP_ERR_EN	DI0 Synchronous display error enable This bit enables the interrupt that is a result of an error during access to a synchronous display via DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
18 DC_TEARING_ ERR_6_EN	Tearing Error #6 enable This bit enables the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 6 0 Interrupt is disabled. 1 Interrupt is enabled.
17 DC_TEARING_ ERR_2_EN	Tearing Error #2 enable This bit enables the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 2

Table continues on the next page...

IPUx_INT_CTRL_10 field descriptions (continued)

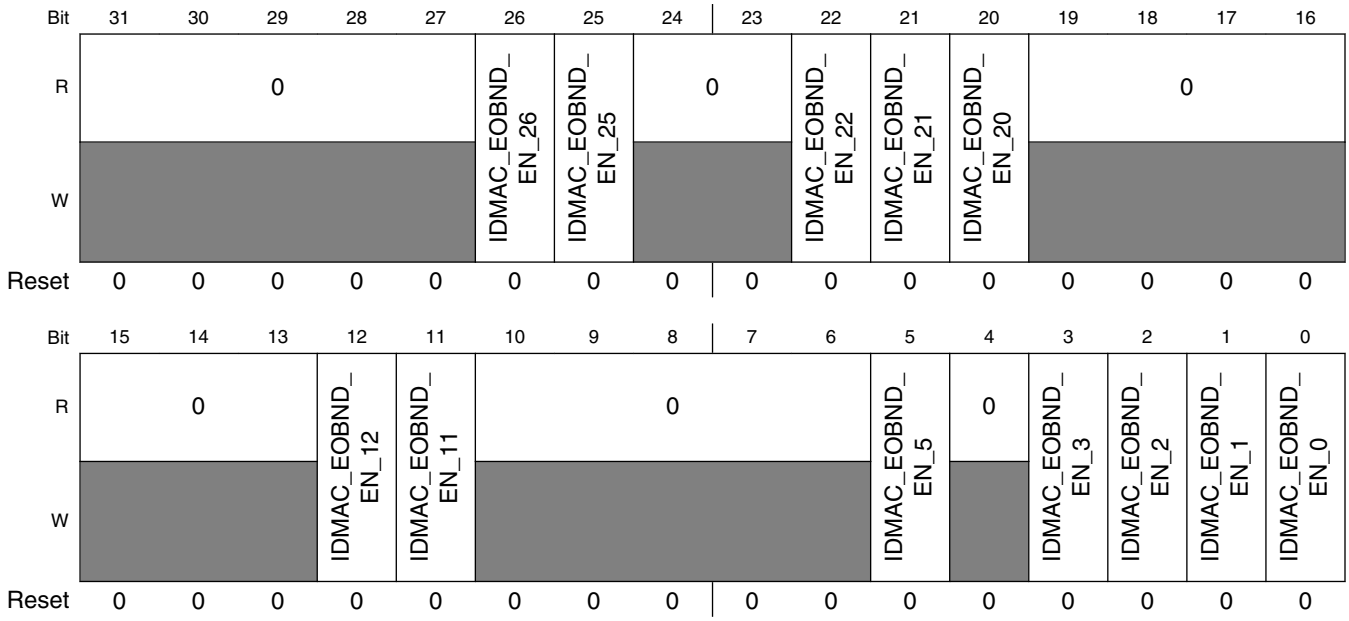
Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
16 DC_TEARING_ERR_1_EN	Tearing Error #1 enable This bit enables the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 1 0 Interrupt is disabled. 1 Interrupt is enabled.
15–4 Reserved	This read-only field is reserved and always has the value 0.
3 SMFC3_FRM_LOST_EN	Frame Lost of SMFC channel 3 interrupt enable bit This bit enables an interrupt that is a result of a Frame Lost of SMFC channel 3. 0 Interrupt is disabled. 1 Interrupt is enabled.
2 SMFC2_FRM_LOST_EN	Frame Lost of SMFC channel 2 interrupt enable bit This bit enables an interrupt that is a result of a Frame Lost of SMFC channel 2. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 SMFC1_FRM_LOST_EN	Frame Lost of SMFC channel 1 interrupt enable bit This bit enables an interrupt that is a result of a Frame Lost of SMFC channel 1. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 SMFC0_FRM_LOST_EN	Frame Lost of SMFC channel 0 interrupt enable bit This bit enables an interrupt that is a result of a Frame Lost of SMFC channel 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.16 Interrupt Control Register 11 (IPUx_INT_CTRL_11)

This register contains part of IPU interrupts controls. The controls of the end-of-band indication (EOBND) of DMA Channels interrupts [31:0] can be found in this register.

- Hide VDOA_SYNC for all versions
- Show VDOA_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.

Address: Base address + 64h offset



IPUx_INT_CTRL_11 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_EOBND_EN_26	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_EOBND_EN_25	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24–23 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
22 IDMAC_EOBND_EN_22	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
21 IDMAC_EOBND_EN_21	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_11 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_EOBND_EN_20	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19–13 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_EOBND_EN_12	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_EOBND_EN_11	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
5 IDMAC_EOBND_EN_5	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
3 IDMAC_EOBND_EN_3	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
2 IDMAC_EOBND_EN_2	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_EOBND_EN_1	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

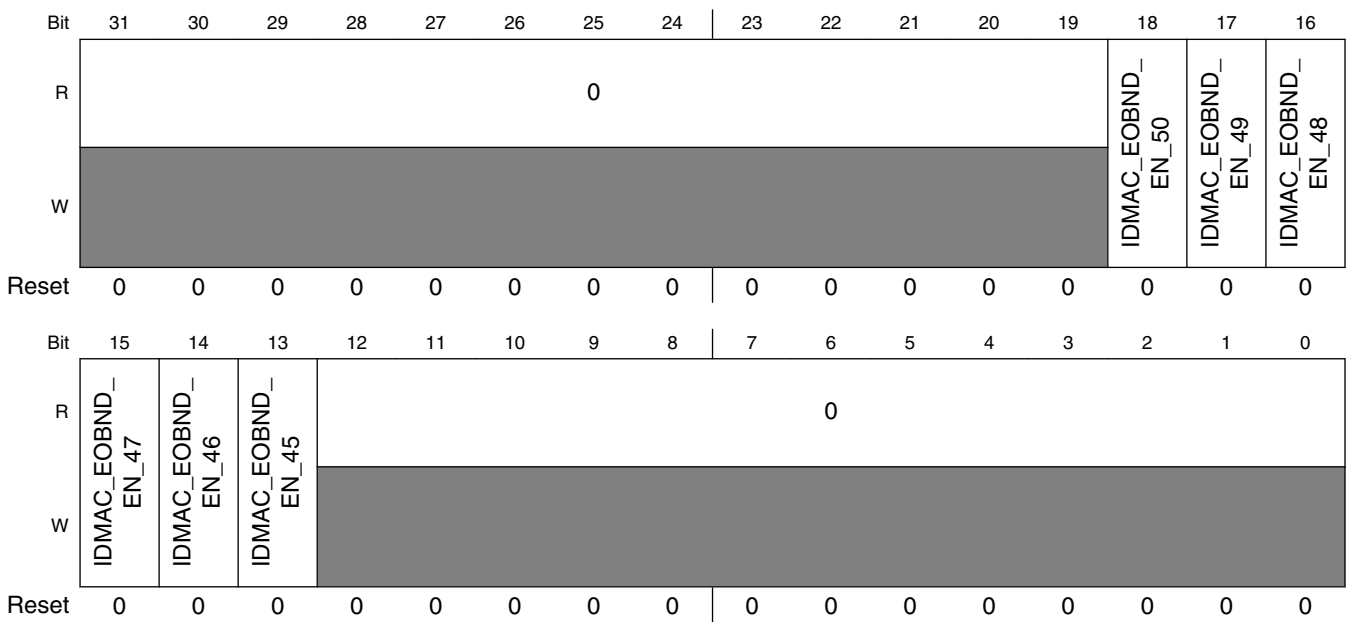
IPUx_INT_CTRL_11 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
0 IDMAC_EOBND_EN_0	end-of-band indication of Channel interrupt. This bit is the control of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.17 Interrupt Control Register 12 (IPUx_INT_CTRL_12)

This register contains part of IPU interrupts controls. The controls of the end-of-band indication (EOBND) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 68h offset



IPUx_INT_CTRL_12 field descriptions

Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0.
	0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_EOBND_EN_50	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_12 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_EOBND_EN_49	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 IDMAC_EOBND_EN_48	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_EOBND_EN_47	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_EOBND_EN_46	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_EOBND_EN_45	end-of-band indication of Channel interrupt. This bit is the control end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.18 Interrupt Control Register 13 (IPUx_INT_CTRL_13)

This register contains part of IPU interrupts controls. The controls of the threshold crossing indication (TH) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 6Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_TH_EN_31	0	IDMAC_TH_EN_29	IDMAC_TH_EN_28	IDMAC_TH_EN_27	IDMAC_TH_EN_26	IDMAC_TH_EN_25	IDMAC_TH_EN_24	IDMAC_TH_EN_23	IDMAC_TH_EN_22	IDMAC_TH_EN_21	IDMAC_TH_EN_20	IDMAC_TH_EN_19	IDMAC_TH_EN_18	IDMAC_TH_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_TH_EN_15	IDMAC_TH_EN_14	IDMAC_TH_EN_13	IDMAC_TH_EN_12	IDMAC_TH_EN_11	IDMAC_TH_EN_10	IDMAC_TH_EN_9	IDMAC_TH_EN_8	0		IDMAC_TH_EN_5	0	IDMAC_TH_EN_3	IDMAC_TH_EN_2	IDMAC_TH_EN_1	IDMAC_TH_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_13 field descriptions

Field	Description
31 IDMAC_TH_EN_31	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
29 IDMAC_TH_EN_29	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
28 IDMAC_TH_EN_28	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_13 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
27 IDMAC_TH_EN_27	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
26 IDMAC_TH_EN_26	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IDMAC_TH_EN_25	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
24 IDMAC_TH_EN_24	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 IDMAC_TH_EN_23	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
22 IDMAC_TH_EN_22	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
21 IDMAC_TH_EN_21	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_TH_EN_20	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_13 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_TH_EN_19	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_TH_EN_18	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_TH_EN_17	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_TH_EN_15	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_TH_EN_14	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
13 IDMAC_TH_EN_13	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_TH_EN_12	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_13 field descriptions (continued)

Field	Description
11 IDMAC_TH_EN_11	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_TH_EN_10	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_TH_EN_9	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_TH_EN_8	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
5 IDMAC_TH_EN_5	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
3 IDMAC_TH_EN_3	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
2 IDMAC_TH_EN_2	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_CTRL_13 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_TH_EN_ 1	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 IDMAC_TH_EN_ 0	Threshold crossing indication of Channel interrupt. This bit is the control of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.19 Interrupt Control Register 14 (IPUx_INT_CTRL_14)

This register contains part of IPU interrupts controls. The controls of the Threshold crossing indication (TH) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 70h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_TH_EN_52	IDMAC_TH_EN_51	IDMAC_TH_EN_50	IDMAC_TH_EN_49	IDMAC_TH_EN_48
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_TH_EN_47	IDMAC_TH_EN_46	IDMAC_TH_EN_45	IDMAC_TH_EN_44	IDMAC_TH_EN_43	IDMAC_TH_EN_42	IDMAC_TH_EN_41	IDMAC_TH_EN_40	0						IDMAC_TH_EN_33	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_14 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
20 IDMAC_TH_EN_52	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
19 IDMAC_TH_EN_51	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
18 IDMAC_TH_EN_50	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
17 IDMAC_TH_EN_49	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
16 IDMAC_TH_EN_48	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
15 IDMAC_TH_EN_47	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
14 IDMAC_TH_EN_46	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.

Table continues on the next page...

IPUx_INT_CTRL_14 field descriptions (continued)

Field	Description
13 IDMAC_TH_EN_45	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
12 IDMAC_TH_EN_44	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
11 IDMAC_TH_EN_43	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
10 IDMAC_TH_EN_42	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
9 IDMAC_TH_EN_41	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
8 IDMAC_TH_EN_40	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is disabled. 1 Interrupt is enabled.
1 IDMAC_TH_EN_33	Threshold crossing indication of Channel interrupt. This bit is the control Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is disabled. 1 Interrupt is enabled.
0 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_INT_CTRL_14 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.20 Interrupt Control Register15 (IPUx_INT_CTRL_15)

This register contains part of IPU interrupts controls. The controls of general purpose interrupts can be found in this register.

Address: Base address + 74h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DI1_CNT_EN_PRE_8_EN	DI1_CNT_EN_PRE_3_EN	DI1_DISP_CLK_EN_PRE_EN	DI0_CNT_EN_PRE_10_EN	DI0_CNT_EN_PRE_9_EN	DI0_CNT_EN_PRE_8_EN	DI0_CNT_EN_PRE_7_EN	DI0_CNT_EN_PRE_6_EN	DI0_CNT_EN_PRE_5_EN	DI0_CNT_EN_PRE_4_EN	DI0_CNT_EN_PRE_3_EN	DI0_CNT_EN_PRE_2_EN	DI0_CNT_EN_PRE_1_EN	DI0_CNT_EN_PRE_0_EN	DC_ASYNC_STOP_EN	DC_DP_START_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI_VSYNC_PRE_1_EN	DI_VSYNC_PRE_0_EN	DC_FC_6_EN	DC_FC_4_EN	DC_FC_3_EN	DC_FC_2_EN	DC_FC_1_EN	DC_FC_0_EN	DP_ASF_BRAKE_EN	DP_SF_BRAKE_EN	DP_ASF_END_EN	DP_ASF_START_EN	DP_SF_END_EN	DP_SF_START_EN	SNOOPING2_INT_EN	SNOOPING1_INT_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_CTRL_15 field descriptions

Field	Description
31 DI1_CNT_EN_PRE_8_EN	This bit enables the interrupt that is a result of a trigger generated by counter #8 of DI1 0 Interrupt is disabled. 1 Interrupt is enabled.
30 DI1_CNT_EN_PRE_3_EN	This bit enables the interrupt that is a result of a trigger generated by counter #3 of DI1 0 Interrupt is disabled. 1 Interrupt is enabled.
29 DI1_DISP_CLK_EN_PRE_EN	DI1_DISP_CLK_EN_PRE_EN

Table continues on the next page...

IPUx_INT_CTRL_15 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
28 DI0_CNT_EN_ PRE_10_EN	This bit enables the interrupt that is a result of a trigger generated by counter #10 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
27 DI0_CNT_EN_ PRE_9_EN	This bit enables the interrupt that is a result of a trigger generated by counter #9 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
26 DI0_CNT_EN_ PRE_8_EN	This bit enables the interrupt that is a result of a trigger generated by counter #8 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
25 DI0_CNT_EN_ PRE_7_EN	This bit enables the interrupt that is a result of a trigger generated by counter #7 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
24 DI0_CNT_EN_ PRE_6_EN	This bit enables the interrupt that is a result of a trigger generated by counter #6 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
23 DI0_CNT_EN_ PRE_5_EN	This bit enables the interrupt that is a result of a trigger generated by counter #5 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
22 DI0_CNT_EN_ PRE_4_EN	This bit enables the interrupt that is a result of a trigger generated by counter #4 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
21 DI0_CNT_EN_ PRE_3_EN	This bit enables the interrupt that is a result of a trigger generated by counter #3 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
20 DI0_CNT_EN_ PRE_2_EN	This bit enables the interrupt that is a result of a trigger generated by counter #2 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
19 DI0_CNT_EN_ PRE_1_EN	This bit enables the interrupt that is a result of a trigger generated by counter #1 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
18 DI0_CNT_EN_ PRE_0_EN	This bit enables the interrupt that is a result of a trigger generated by counter #0 of DI0 0 Interrupt is disabled. 1 Interrupt is enabled.
17 DC_ASYNC_ STOP_EN	This bit enables the interrupt asserted anytime the DP stops an async flow and moves to a sync flow

Table continues on the next page...

IPUx_INT_CTRL_15 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
16 DC_DP_START_EN	This bit enables the interrupt asserted anytime the DP start a new sync or async flow or when an async flow is interrupted by a sync flow 0 Interrupt is disabled. 1 Interrupt is enabled.
15 DI_VSYNC_PRE_1_EN	This bit enables the DI1 interrupt indicating of a VSYNC signal asserted 2 rows before the VSYNC sent to the display 0 Interrupt is disabled. 1 Interrupt is enabled.
14 DI_VSYNC_PRE_0_EN	This bit enables the DI0 interrupt indicating of a VSYNC signal asserted 2 rows before the VSYNC sent to the display 0 Interrupt is disabled. 1 Interrupt is enabled.
13 DC_FC_6_EN	This bit enables the DC Frame Complete on channel #6 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
12 DC_FC_4_EN	This bit enables the DC Frame Complete on channel #4 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
11 DC_FC_3_EN	This bit enables the DC Frame Complete on channel #3 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
10 DC_FC_2_EN	This bit enables the DC Frame Complete on channel #2 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
9 DC_FC_1_EN	This bit enables they'd Frame Complete on channel #1 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
8 DC_FC_0_EN	This bit enables they'd Frame Complete on channel #0 interrupt 0 Interrupt is disabled. 1 Interrupt is enabled.
7 DP_ASF_BRAKE_EN	DP Async Flow Brake enable bit. This bit enables the interrupt that is a result of the async flow brake at the DP 0 Interrupt is disabled. 1 Interrupt is enabled.
6 DP_SF_BRAKE_EN	DP Sync Flow Brake enable bit. This bit enables the interrupt that is a result of the Sync flow brake at the DP

Table continues on the next page...

IPUx_INT_CTRL_15 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
5 DP_ASF_END_EN	DP Async Flow End enable bit. This bit enables the interrupt that is a result of the Async flow end at the DP 0 Interrupt is disabled. 1 Interrupt is enabled.
4 DP_ASF_START_EN	DP Async Flow Start enable bit. This bit enables the interrupt that is a result of the Async flow start at the DP 0 Interrupt is disabled. 1 Interrupt is enabled.
3 DP_SF_END_EN	DP Sync Flow End enable bit. This bit enables the interrupt that is a result of the Sync flow end at the DP 0 Interrupt is disabled. 1 Interrupt is enabled.
2 DP_SF_START_EN	DP Sync Flow Start enable bit. This bit enables the interrupt that is a result of the Sync flow start at the DP 0 Interrupt is disabled. 1 Interrupt is enabled.
1 SNOOPING2_INT_EN	IPU snooping 2 interrupt enable bit. This bit enables the interrupt that is a result of the detection of a snooping 2 signal assertion coming to the IPU 0 Interrupt is disabled. 1 Interrupt is enabled.
0 SNOOPING1_INT_EN	IPU snooping 1 interrupt enable bit. This bit enables the interrupt that is a result of the detection of a snooping 1 signal assertion coming to the IPU 0 Interrupt is disabled. 1 Interrupt is enabled.

37.5.21 SDMA Event Control Register 1 (IPUx_SDMA_EVENT_1)

This register contains part of IPU SDMA events controls. The controls of EOF (end of frame) of DMA Channels SDMA events [31:0] can be found in this register.

Address: Base address + 78h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_EOF_SDMA_EN_31	0	IDMAC_EOF_SDMA_EN_29	IDMAC_EOF_SDMA_EN_28	IDMAC_EOF_SDMA_EN_27	IDMAC_EOF_SDMA_EN_26	IDMAC_EOF_SDMA_EN_25	IDMAC_EOF_SDMA_EN_24	IDMAC_EOF_SDMA_EN_23	IDMAC_EOF_SDMA_EN_22	IDMAC_EOF_SDMA_EN_21	IDMAC_EOF_SDMA_EN_20	IDMAC_EOF_SDMA_EN_19	IDMAC_EOF_SDMA_EN_18	IDMAC_EOF_SDMA_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_EOF_SDMA_EN_15	IDMAC_EOF_SDMA_EN_14	IDMAC_EOF_SDMA_EN_13	IDMAC_EOF_SDMA_EN_12	IDMAC_EOF_SDMA_EN_11	IDMAC_EOF_SDMA_EN_10	IDMAC_EOF_SDMA_EN_9	IDMAC_EOF_SDMA_EN_8	0		IDMAC_EOF_SDMA_EN_5	0	IDMAC_EOF_SDMA_EN_3	IDMAC_EOF_SDMA_EN_2	IDMAC_EOF_SDMA_EN_1	IDMAC_EOF_SDMA_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SDMA_EVENT_1 field descriptions

Field	Description
31 IDMAC_EOF_SDMA_EN_31	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
29 IDMAC_EOF_SDMA_EN_29	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
28 IDMAC_EOF_SDMA_EN_28	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_1 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
27 IDMAC_EOF_ SDMA_EN_27	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
26 IDMAC_EOF_ SDMA_EN_26	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
25 IDMAC_EOF_ SDMA_EN_25	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
24 IDMAC_EOF_ SDMA_EN_24	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
23 IDMAC_EOF_ SDMA_EN_23	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
22 IDMAC_EOF_ SDMA_EN_22	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
21 IDMAC_EOF_ SDMA_EN_21	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_EOF_ SDMA_EN_20	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_EOF_ SDMA_EN_19	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_1 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
18 IDMAC_EOF_SDMA_EN_18	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_EOF_SDMA_EN_17	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_EOF_SDMA_EN_15	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_EOF_SDMA_EN_14	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_EOF_SDMA_EN_13	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_EOF_SDMA_EN_12	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_EOF_SDMA_EN_11	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10 IDMAC_EOF_SDMA_EN_10	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

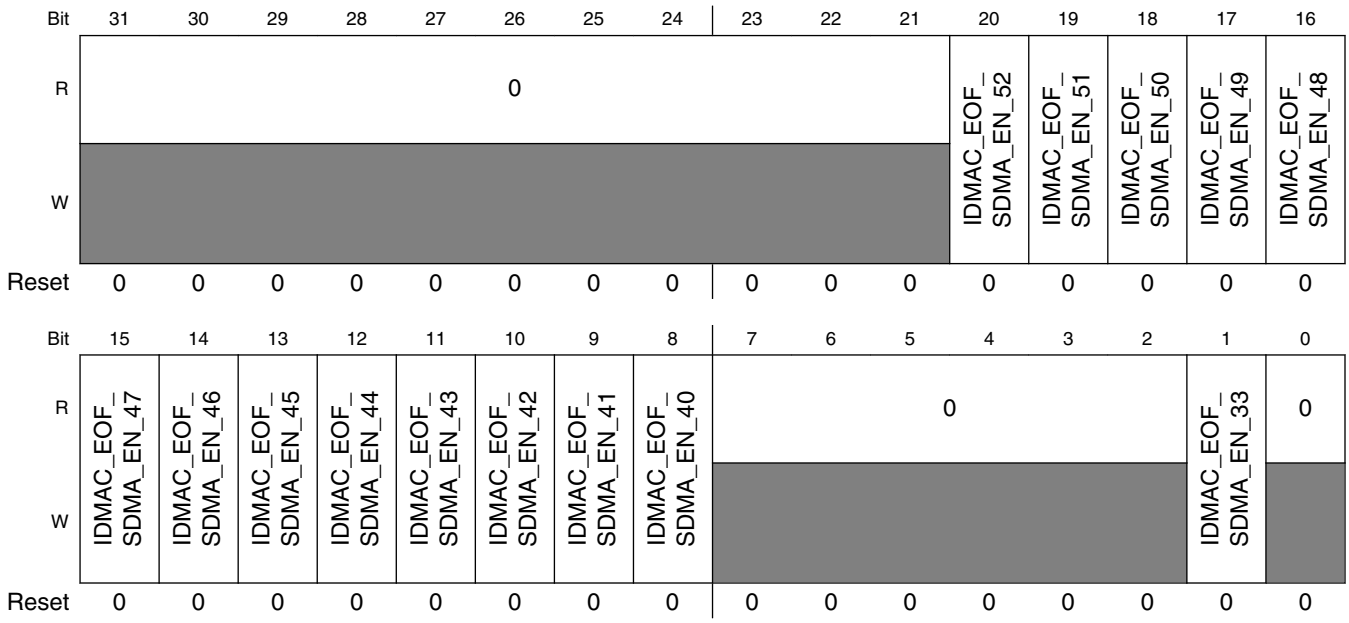
IPUx_SDMA_EVENT_1 field descriptions (continued)

Field	Description
9 IDMAC_EOF_ SDMA_EN_9	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_EOF_ SDMA_EN_8	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
5 IDMAC_EOF_ SDMA_EN_5	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
3 IDMAC_EOF_ SDMA_EN_3	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
2 IDMAC_EOF_ SDMA_EN_2	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_EOF_ SDMA_EN_1	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 IDMAC_EOF_ SDMA_EN_0	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.22 SDMA Event Control Register 2 (IPUx_SDMA_EVENT_2)

This register contains part of IPU SDMA events controls. The controls of EOF (end of frame) of DMA Channels SDMA events [63:32] can be found in this register.

Address: Base address + 7Ch offset



IPUx_SDMA_EVENT_2 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_EOF_SDMA_EN_52	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_EOF_SDMA_EN_51	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
18 IDMAC_EOF_SDMA_EN_50	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_2 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_EOF_ SDMA_EN_49	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 IDMAC_EOF_ SDMA_EN_48	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_EOF_ SDMA_EN_47	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_EOF_ SDMA_EN_46	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_EOF_ SDMA_EN_45	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_EOF_ SDMA_EN_44	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_EOF_ SDMA_EN_43	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10 IDMAC_EOF_ SDMA_EN_42	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
9 IDMAC_EOF_ SDMA_EN_41	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_2 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_EOF_SDMA_EN_40	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_EOF_SDMA_EN_33	Enable End of Frame of Channel SDMA event. This bit is the control of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.23 SDMA Event Control Register 3 (IPUx_SDMA_EVENT_3)

This register contains part of IPU SDMA events controls. The controls of NFACK (New Frame Acknowledge) of DMA Channels SDMA events [31:0] can be found in this register.

Address: Base address + 80h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_NFACK_SDMA_EN_31	0	IDMAC_NFACK_SDMA_EN_29	IDMAC_NFACK_SDMA_EN_28	IDMAC_NFACK_SDMA_EN_27	IDMAC_NFACK_SDMA_EN_26	IDMAC_NFACK_SDMA_EN_25	IDMAC_NFACK_SDMA_EN_24	IDMAC_NFACK_SDMA_EN_23	IDMAC_NFACK_SDMA_EN_22	IDMAC_NFACK_SDMA_EN_21	IDMAC_NFACK_SDMA_EN_20	IDMAC_NFACK_SDMA_EN_19	IDMAC_NFACK_SDMA_EN_18	IDMAC_NFACK_SDMA_EN_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFACK_SDMA_EN_15	IDMAC_NFACK_SDMA_EN_14	IDMAC_NFACK_SDMA_EN_13	IDMAC_NFACK_SDMA_EN_12	IDMAC_NFACK_SDMA_EN_11	IDMAC_NFACK_SDMA_EN_10	IDMAC_NFACK_SDMA_EN_9	IDMAC_NFACK_SDMA_EN_8	0		IDMAC_NFACK_SDMA_EN_5	0	IDMAC_NFACK_SDMA_EN_3	IDMAC_NFACK_SDMA_EN_2	IDMAC_NFACK_SDMA_EN_1	IDMAC_NFACK_SDMA_EN_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SDMA_EVENT_3 field descriptions

Field	Description
31 IDMAC_NFACK_SDMA_EN_31	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
29 IDMAC_NFACK_SDMA_EN_29	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_3 field descriptions (continued)

Field	Description
28 IDMAC_NFACK_SDMA_EN_28	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
27 IDMAC_NFACK_SDMA_EN_27	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
26 IDMAC_NFACK_SDMA_EN_26	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
25 IDMAC_NFACK_SDMA_EN_25	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
24 IDMAC_NFACK_SDMA_EN_24	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
23 IDMAC_NFACK_SDMA_EN_23	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
22 IDMAC_NFACK_SDMA_EN_22	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
21 IDMAC_NFACK_SDMA_EN_21	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_3 field descriptions (continued)

Field	Description
20 IDMAC_NFACK_SDMA_EN_20	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_NFACK_SDMA_EN_19	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
18 IDMAC_NFACK_SDMA_EN_18	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_NFACK_SDMA_EN_17	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_NFACK_SDMA_EN_15	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_NFACK_SDMA_EN_14	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_NFACK_SDMA_EN_13	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_3 field descriptions (continued)

Field	Description
12 IDMAC_NFACK_SDMA_EN_12	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_NFACK_SDMA_EN_11	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10 IDMAC_NFACK_SDMA_EN_10	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
9 IDMAC_NFACK_SDMA_EN_9	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_NFACK_SDMA_EN_8	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
5 IDMAC_NFACK_SDMA_EN_5	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
3 IDMAC_NFACK_SDMA_EN_3	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

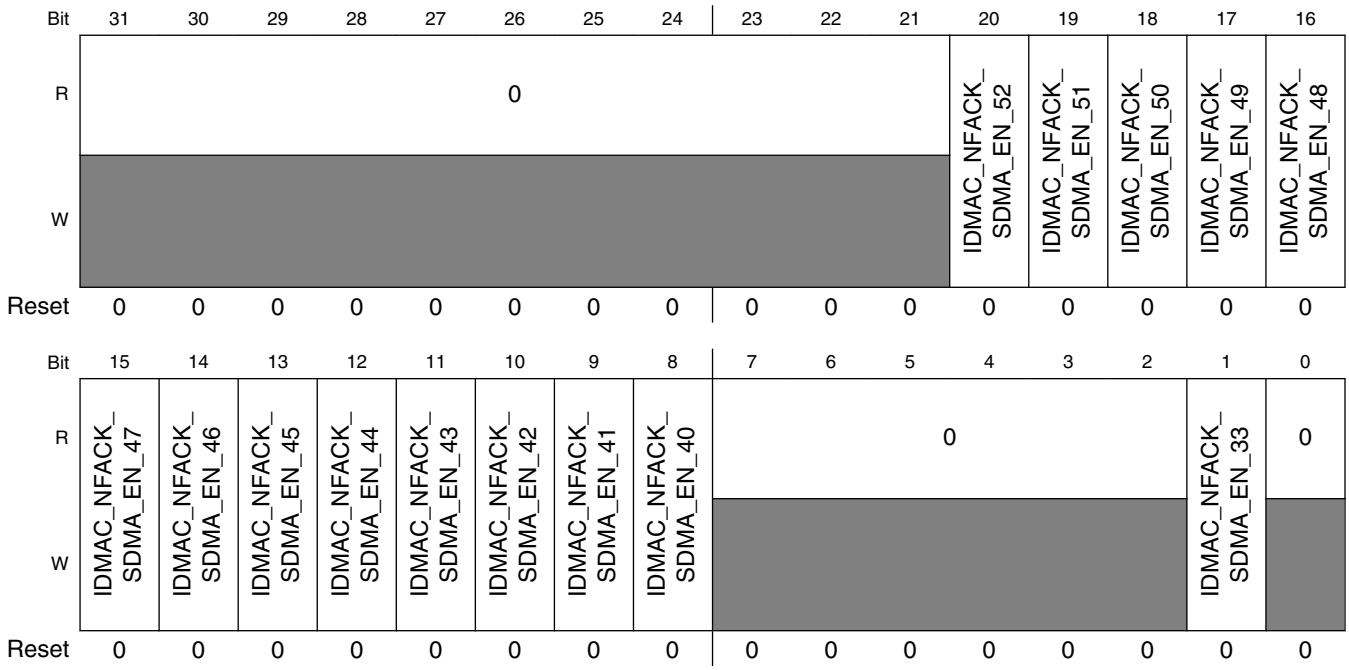
IPUx_SDMA_EVENT_3 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
2 IDMAC_NFACK_SDMA_EN_2	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_NFACK_SDMA_EN_1	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 IDMAC_NFACK_SDMA_EN_0	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.24 SDMA Event Control Register 4 (IPUx_SDMA_EVENT_4)

This register contains part of IPU SDMA events controls. The controls of NFACK (New Frame Acknowledge) of DMA Channels SDMA events [63:32] can be found in this register.

Address: Base address + 84h offset



IPUx_SDMA_EVENT_4 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_NFACK_SDMA_EN_52	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_NFACK_SDMA_EN_51	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_4 field descriptions (continued)

Field	Description
18 IDMAC_NFACK_SDMA_EN_50	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_NFACK_SDMA_EN_49	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 IDMAC_NFACK_SDMA_EN_48	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_NFACK_SDMA_EN_47	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_NFACK_SDMA_EN_46	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_NFACK_SDMA_EN_45	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_NFACK_SDMA_EN_44	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_NFACK_SDMA_EN_43	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

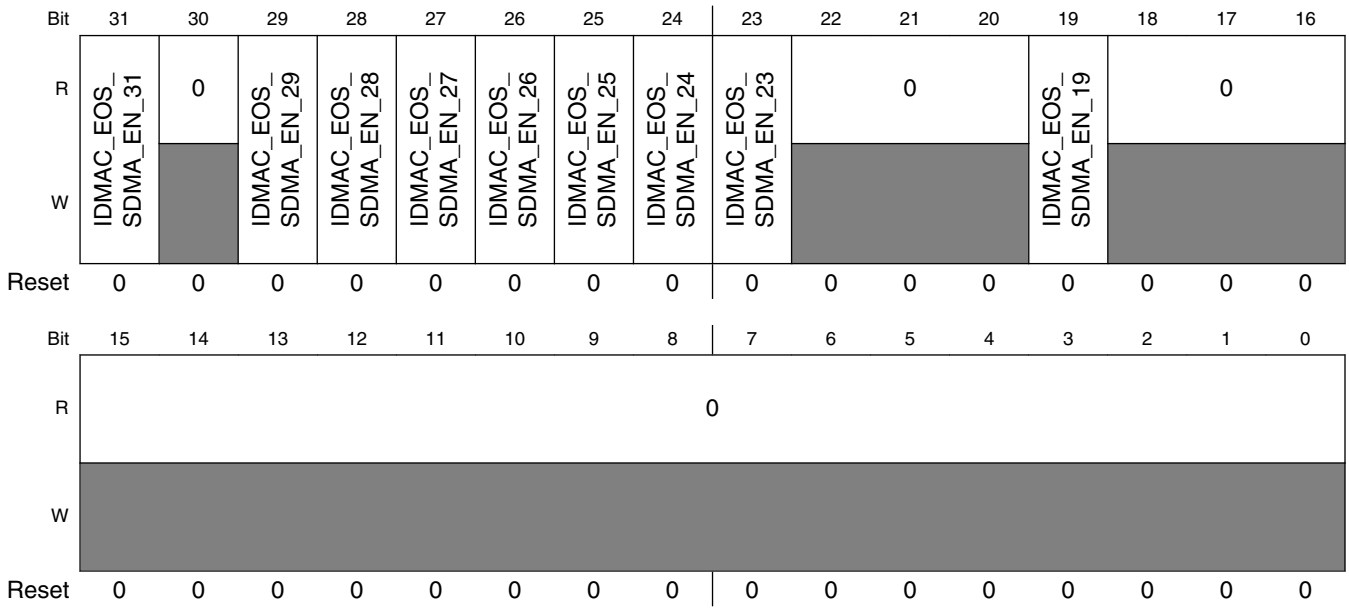
IPUx_SDMA_EVENT_4 field descriptions (continued)

Field	Description
10 IDMAC_NFACK_SDMA_EN_42	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
9 IDMAC_NFACK_SDMA_EN_41	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_NFACK_SDMA_EN_40	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_NFACK_SDMA_EN_33	Enable New Frame Acknowledge of Channel SDMA event. This bit is the control of New Frame Acknowledge of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.25 SDMA Event Control Register 7 (IPUx_SDMA_EVENT_7)

This register contains part of IPU SDMA events controls. The controls of EOS (End of Scroll) of DMA Channels SDMA events [31:0] can be found in this register.

Address: Base address + 88h offset



IPUx_SDMA_EVENT_7 field descriptions

Field	Description
31 IDMAC_EOS_ SDMA_EN_31	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
29 IDMAC_EOS_ SDMA_EN_29	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
28 IDMAC_EOS_ SDMA_EN_28	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

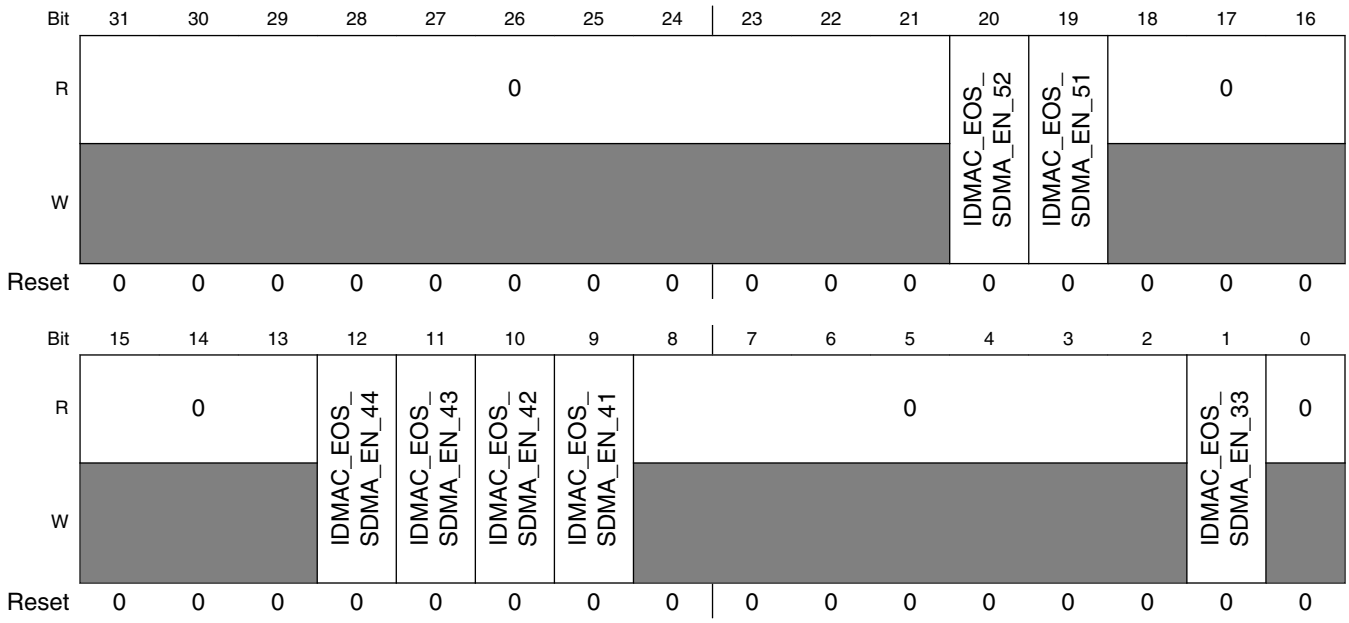
IPUx_SDMA_EVENT_7 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
27 IDMAC_EOS_ SDMA_EN_27	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
26 IDMAC_EOS_ SDMA_EN_26	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
25 IDMAC_EOS_ SDMA_EN_25	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
24 IDMAC_EOS_ SDMA_EN_24	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
23 IDMAC_EOS_ SDMA_EN_23	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
22–20 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_EOS_ SDMA_EN_19	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.26 SDMA Event Control Register 8 (IPUx_SDMA_EVENT_8)

This register contains part of IPU SDMA events controls. The controls of EOS (End of Scroll) of DMA Channels SDMA events [63:32] can be found in this register.

Address: Base address + 8Ch offset



IPUx_SDMA_EVENT_8 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_EOS_ SDMA_EN_52	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_EOS_ SDMA_EN_51	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
18–13 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_8 field descriptions (continued)

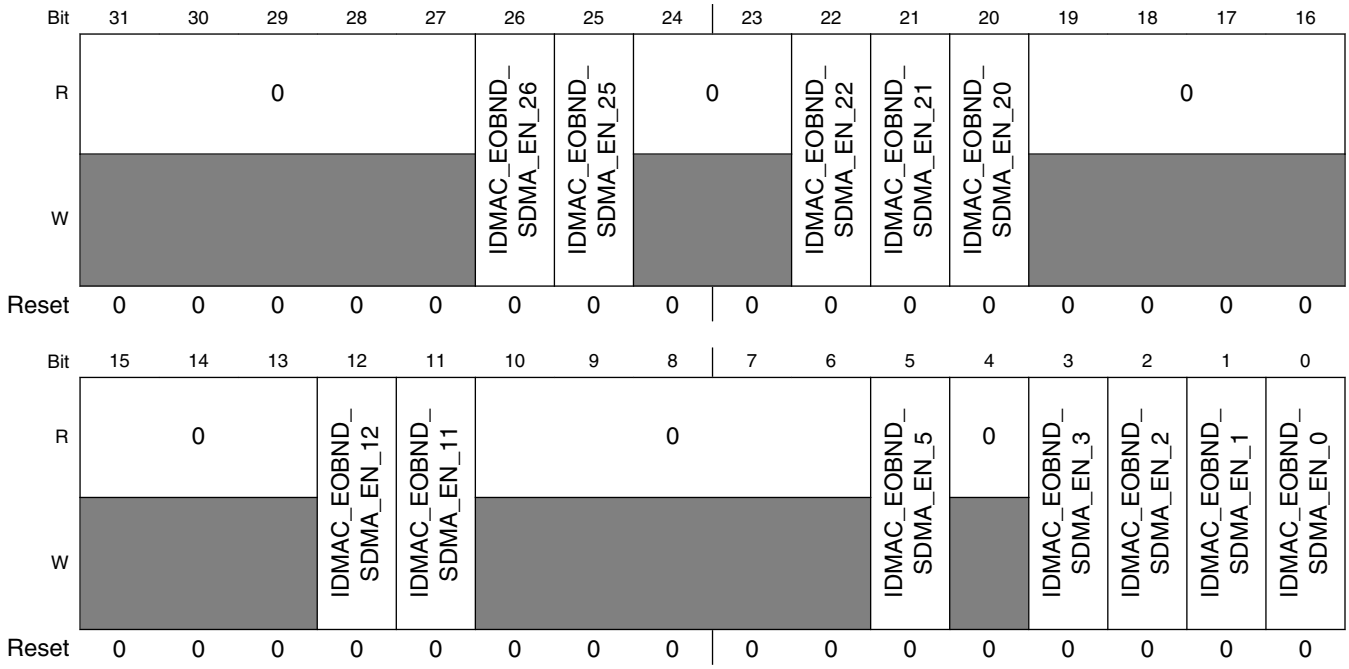
Field	Description
12 IDMAC_EOS_ SDMA_EN_44	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_EOS_ SDMA_EN_43	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10 IDMAC_EOS_ SDMA_EN_42	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
9 IDMAC_EOS_ SDMA_EN_41	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8–2 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_EOS_ SDMA_EN_33	Enable End of Scroll of Channel SDMA event. This bit is the control of End Of Scroll of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.27 SDMA Event Control Register 11 (IPUx_SDMA_EVENT_11)

This register contains part of IPU SDMA events controls. The controls of EOBND (End of Band) of DMA Channels SDMA events [31:0] can be found in this register.

- Hide VDOA_SYNC for all versions
- Show VDOA_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.

Address: Base address + 90h offset



IPUx_SDMA_EVENT_11 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
26 IDMAC_EOBND_SDMA_EN_26	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
25 IDMAC_EOBND_SDMA_EN_25	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
24–23 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
22 IDMAC_EOBND_SDMA_EN_22	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
21 IDMAC_EOBND_SDMA_EN_21	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_11 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_EOBND_SDMA_EN_20	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19–13 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_EOBND_SDMA_EN_12	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_EOBND_SDMA_EN_11	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10–6 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
5 IDMAC_EOBND_SDMA_EN_5	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
3 IDMAC_EOBND_SDMA_EN_3	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
2 IDMAC_EOBND_SDMA_EN_2	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_EOBND_SDMA_EN_1	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

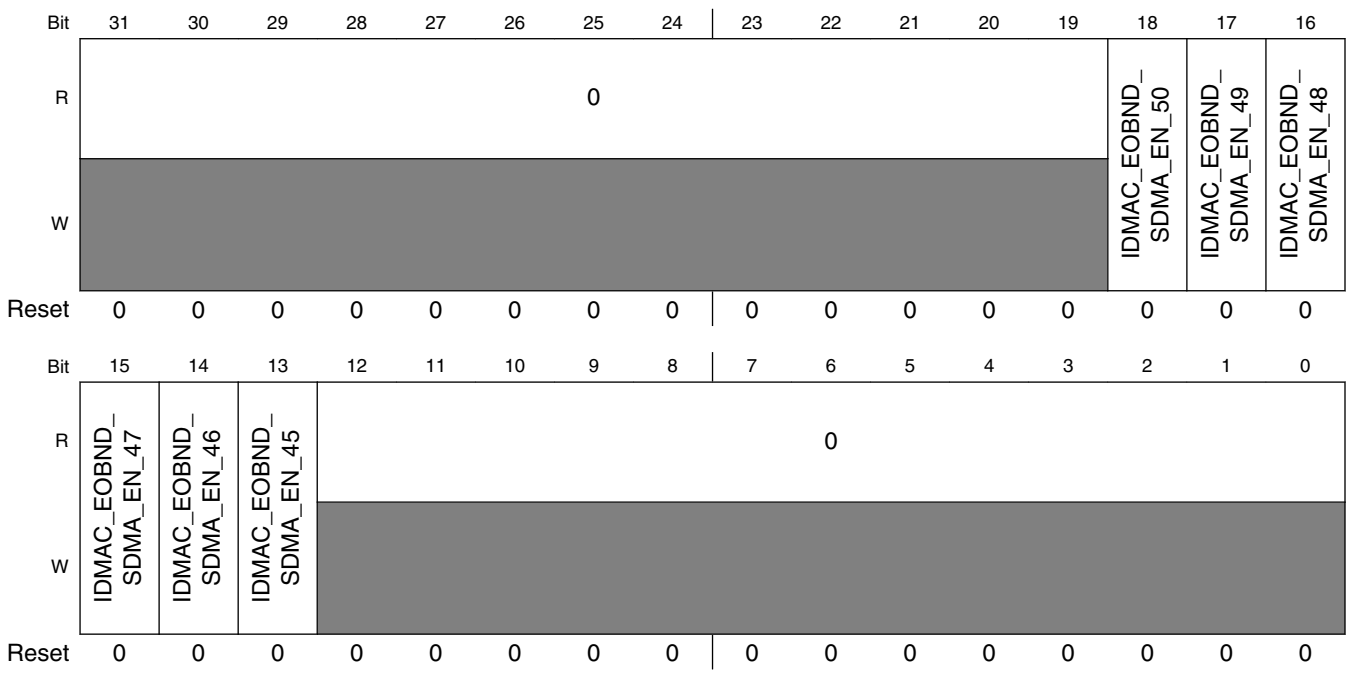
IPUx_SDMA_EVENT_11 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
0 IDMAC_EOBND_ SDMA_EN_0	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.28 SDMA Event Control Register 12 (IPUx_SDMA_EVENT_12)

This register contains part of IPU SDMA events controls. The controls of EOBND (End of Band) of DMA Channels SDMA events [63:32] can be found in this register.

Address: Base address + 94h offset



IPUx_SDMA_EVENT_12 field descriptions

Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0.
	0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

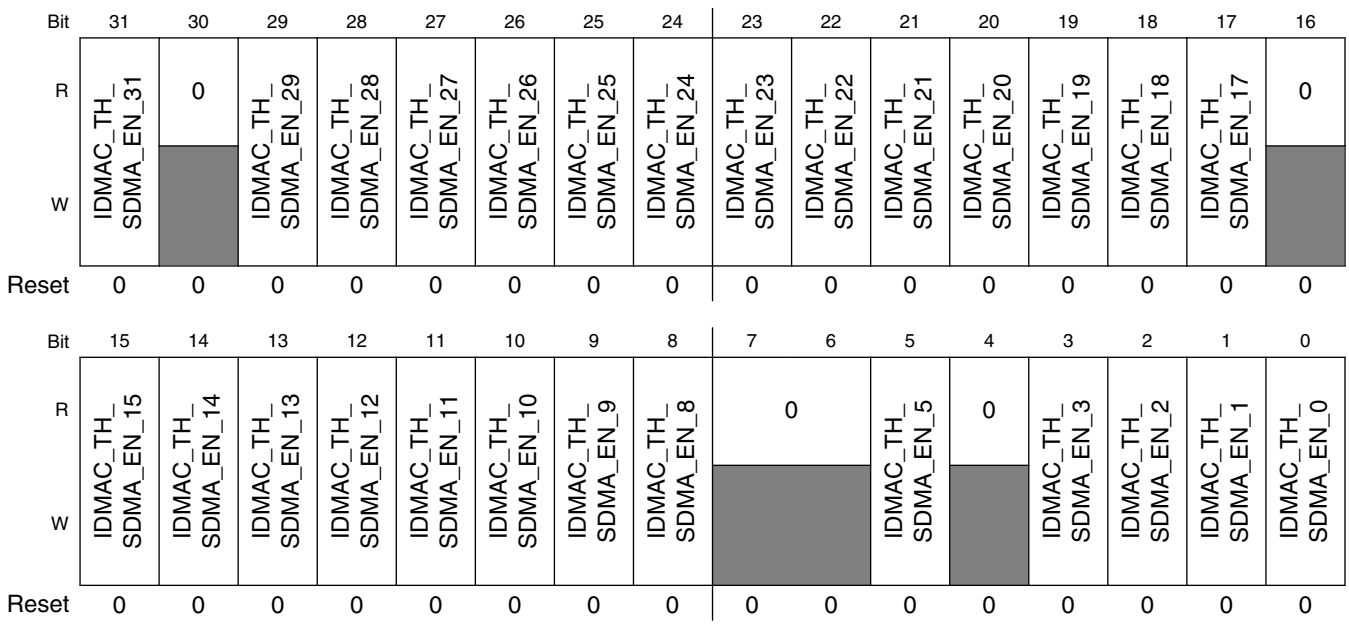
IPUx_SDMA_EVENT_12 field descriptions (continued)

Field	Description
18 IDMAC_EOBND_ SDMA_EN_50	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_EOBND_ SDMA_EN_49	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 IDMAC_EOBND_ SDMA_EN_48	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_EOBND_ SDMA_EN_47	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_EOBND_ SDMA_EN_46	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_EOBND_ SDMA_EN_45	Enable End of Band of Channel SDMA event. This bit is the control of End Of Band of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.29 SDMA Event Control Register 13 (IPUx_SDMA_EVENT_13)

This register contains part of IPU SDMA events controls. The controls of TH (Threshold) of DMA Channels SDMA events [31:0] can be found in this register.

Address: Base address + 98h offset



IPUx_SDMA_EVENT_13 field descriptions

Field	Description
31 IDMAC_TH_ SDMA_EN_31	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
30 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
29 IDMAC_TH_ SDMA_EN_29	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
28 IDMAC_TH_ SDMA_EN_28	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_13 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
27 IDMAC_TH_ SDMA_EN_27	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
26 IDMAC_TH_ SDMA_EN_26	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
25 IDMAC_TH_ SDMA_EN_25	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
24 IDMAC_TH_ SDMA_EN_24	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
23 IDMAC_TH_ SDMA_EN_23	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
22 IDMAC_TH_ SDMA_EN_22	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
21 IDMAC_TH_ SDMA_EN_21	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
20 IDMAC_TH_ SDMA_EN_20	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_TH_ SDMA_EN_19	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_SDMA_EVENT_13 field descriptions (continued)

Field	Description
	0 SDMA event is disabled. 1 SDMA event is enabled.
18 IDMAC_TH_ SDMA_EN_18	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_TH_ SDMA_EN_17	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_TH_ SDMA_EN_15	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_TH_ SDMA_EN_14	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_TH_ SDMA_EN_13	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_TH_ SDMA_EN_12	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_TH_ SDMA_EN_11	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
10 IDMAC_TH_ SDMA_EN_10	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

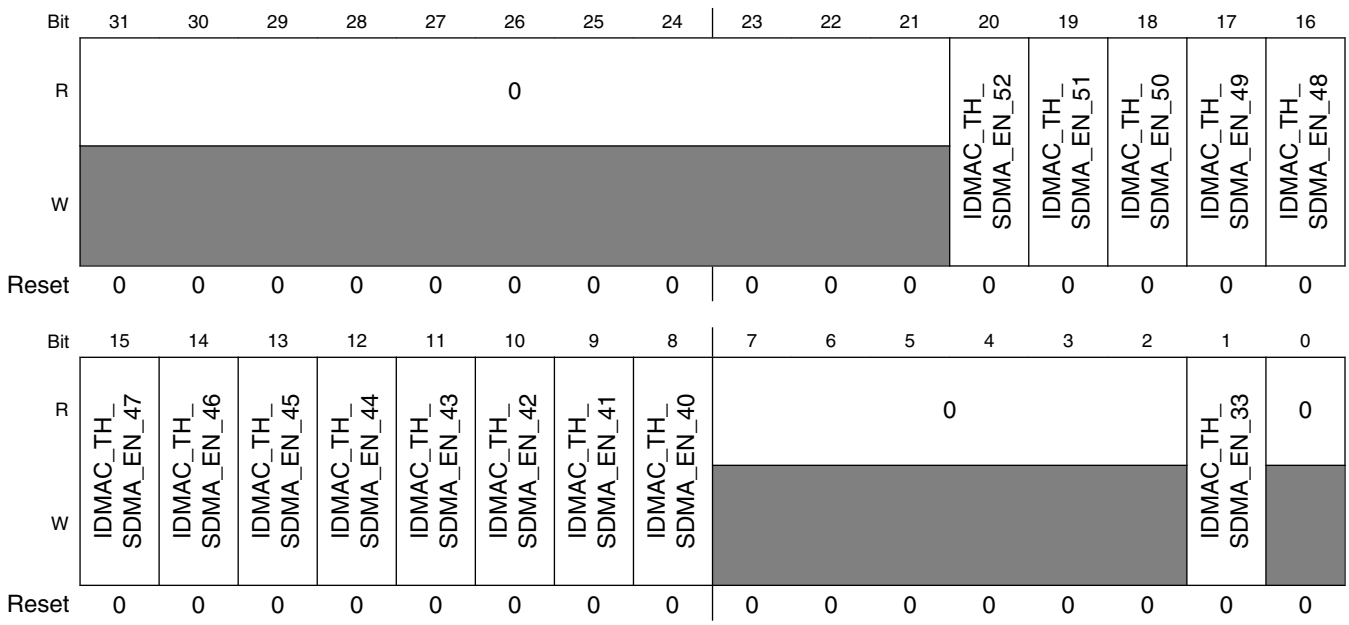
IPUx_SDMA_EVENT_13 field descriptions (continued)

Field	Description
9 IDMAC_TH_ SDMA_EN_9	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_TH_ SDMA_EN_8	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
5 IDMAC_TH_ SDMA_EN_5	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
4 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
3 IDMAC_TH_ SDMA_EN_3	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
2 IDMAC_TH_ SDMA_EN_2	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_TH_ SDMA_EN_1	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 IDMAC_TH_ SDMA_EN_0	Enable Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.30 SDMA Event Control Register 14 (IPUx_SDMA_EVENT_14)

This register contains part of IPU SDMA events controls. The controls of TH (Threshold) of DMA Channels SDMA events [63:32] can be found in this register.

Address: Base address + 9Ch offset



IPUx_SDMA_EVENT_14 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.
20 IDMAC_TH_ SDMA_EN_52	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
19 IDMAC_TH_ SDMA_EN_51	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_14 field descriptions (continued)

Field	Description
18 IDMAC_TH_ SDMA_EN_50	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
17 IDMAC_TH_ SDMA_EN_49	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
16 IDMAC_TH_ SDMA_EN_48	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
15 IDMAC_TH_ SDMA_EN_47	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
14 IDMAC_TH_ SDMA_EN_46	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
13 IDMAC_TH_ SDMA_EN_45	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
12 IDMAC_TH_ SDMA_EN_44	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
11 IDMAC_TH_ SDMA_EN_43	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.

Table continues on the next page...

IPUx_SDMA_EVENT_14 field descriptions (continued)

Field	Description
10 IDMAC_TH_ SDMA_EN_42	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
9 IDMAC_TH_ SDMA_EN_41	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
8 IDMAC_TH_ SDMA_EN_40	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.
1 IDMAC_TH_ SDMA_EN_33	Threshold of Channel SDMA event. This bit is the control of Threshold of Channel #n. n Indicates the corresponding DMA channel number. *n* Indicates the corresponding DMA channel number. 0 SDMA event is disabled. 1 SDMA event is enabled.
0 Reserved	This read-only field is reserved and always has the value 0. 0 SDMA event is disabled. 1 SDMA event is enabled.

37.5.31 Shadow Registers Memory Priority 1 Register (IPUx_SRM_PRI1)

The register controls the priority of SRM updates. The priority level for each block that has a shadow of its registers in the SRM should be unique. The priority level defines the order of SRM updates. A block with priority set to 010 will be updated before a block with priority set to 001.

Address: Base address + A0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			CSI0_SRM_MODE		CSI0_SRM_PRI			0			CSI1_SRM_MODE		CSI1_SRM_PRI		
W																
Reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

IPUx_SRM_PRI1 field descriptions

Field	Description
31–13 Reserved	This read-only field is reserved and always has the value 0.
12–11 CSI0_SRM_MODE	CSI0 SRM Mode This field controls the SRM logic that handles the CSI0 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the CSI1's region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM Platform. The Register will be update now
10–8 CSI0_SRM_PRI	CSI0 SRM priority This bits define the priority of the CSI1 block
7–5 Reserved	This read-only field is reserved and always has the value 0.
4–3 CSI1_SRM_MODE	CSI1 SRM Mode This field controls the SRM logic that handles the CSI1 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the CSI0's region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
CSI1_SRM_PRI	CSI1 SRM priority This bits define the priority of the CSI0 module

37.5.32 Shadow Registers Memory Priority 2 Register (IPUx_SRM_PRI2)

The register controls the priority of SRM updates. The priority level for each block that has a shadow of its registers in the SRM should be unique. The priority level defines the order of SRM updates. a block with priority set to 010 will be updated before a block with priority set to 001.

Address: Base address + A4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0			DI1_SRM_MODE		DI1_SRM_PRI			0			DI0_SRM_MCU_USE		DI0_SRM_PRI		
W																
Reset	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DC_6_SRM_MODE		DC_2_SRM_MODE		DC_SRM_PRI			DP_A1_SRM_MODE	DP_A0_SRM_MODE			DP_S_SRM_MODE		DP_SRM_PRI		
W																
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1

IPUx_SRM_PRI2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–27 DI1_SRM_MODE	DCI1 SRM Mode This field controls the SRM logic that handles the DI1 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DI1 region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
26–24 DI1_SRM_PRI	DI1 SRM priority This bits define the priority of the DI1 module
23–21 Reserved	This read-only field is reserved and always has the value 0.
20–19 DI0_SRM_MCU_USE	DI0 SRM is used by ARM platform This bit indicates that the registers of the DI0 are currently being updated by the ARM platform. The ARM platform should set this bit before accessing the SRM part that is relevant to the DI0. The ARM platform should clear this bit when the update procedure is finished. When this bit is set the SRM mechanism will not update the DI0's registers to avoid data coherency problems. 1 DI0 SRM is currently updated by the ARM platform 0 DI0 SRM s currently not updated by the ARM platform
18–16 DI0_SRM_PRI	DI0 SRM priority This bits define the priority of the DI0 module

Table continues on the next page...

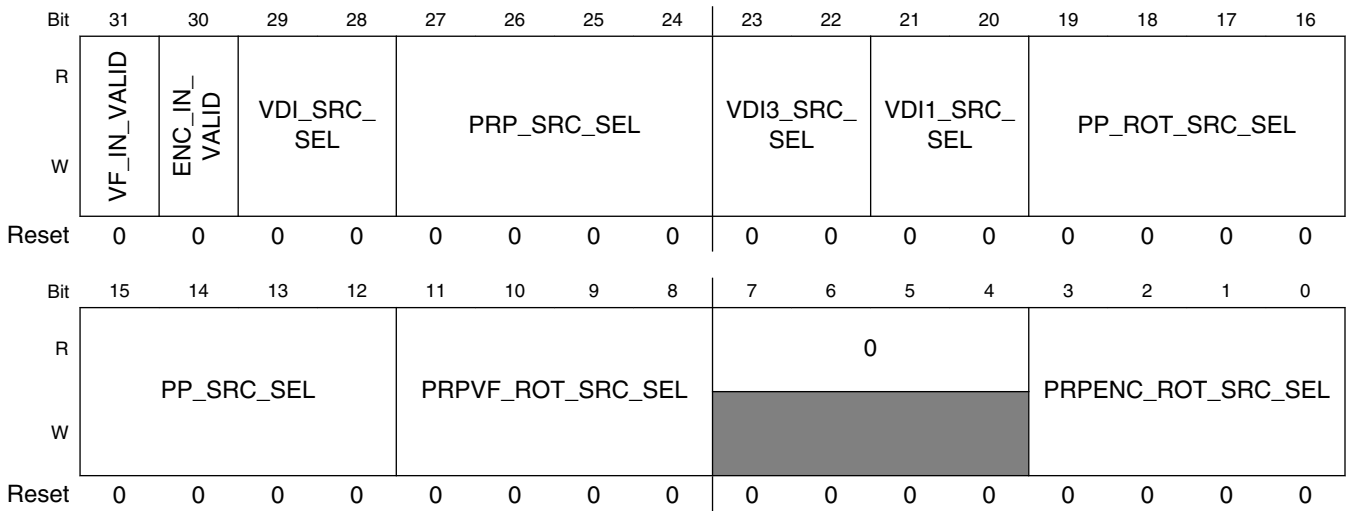
IPUx_SRM_PRI2 field descriptions (continued)

Field	Description
15–14 DC_6_SRM_ MODE	DC Group #6 SRM Mode This field controls the SRM logic that handles the DC Group #6 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DC Group #6's region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
13–12 DC_2_SRM_ MODE	DC Group #2 SRM Mode This field controls the SRM logic that handles the DC Group #2 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DC Group #2's region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
11–9 DC_SRM_PRI	DC SRM priority This bits define the priority of the DC module
8–7 DP_A1_SRM_ MODE	DP Async flow #1 SRM Mode This field controls the SRM logic that handles the DP Async flow #1 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DP Async flow #1 region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
6–5 DP_A0_SRM_ MODE	DP Async flow #0 SRM Mode This field controls the SRM logic that handles the DP Async flow #0 registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DP Async flow #0 region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done of the next frame. 10 The SRM logic is controlled by the FSU. Registers are swapped continuously frame by frame 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
4–3 DP_S_SRM_ MODE	DP sync flow SRM Mode This field controls the SRM logic that handles the DP sync flow registers 00 Automatic swapping is disabled; ARM platform is allowed to access the DP sync flow region in the RAM 01 The SRM logic is controlled by the FSU. The update will be done on the next frame. 10 Reserved 11 Update now. The SRM is controlled by the ARM platform. The Register will be update now
DP_SRM_PRI	DP SRM priority This bits define the priority of the DP module

37.5.33 FSU Processing Flow 1 Register (IPUx_FS_PROC_FLOW1)

This register contain controls for IPU's tasks.

Address: Base address + A8h offset



IPUx_FS_PROC_FLOW1 field descriptions

Field	Description
31 VF_IN_VALID	View-finder Input valid. Setting this bit indicates that the buffer in memory for viewfinder is validated by the ARM platform (valid only when RWS_EN is '1'). 0 View-finder should skip buffer in memory. 1 View-finder should use buffer in memory.
30 ENC_IN_VALID	Encoding Input valid. Setting this bit indicates that the buffer in memory for encoding is validated by the ARM platform (valid only when RWS_EN is '1'). 0 Encoding should skip buffer in memory. 1 Encoding should use buffer in memory.
29–28 VDI_SRC_SEL	Source select for the VDIC This field is relevant if the VDIC works in de-interlacing mode (when VDI_CMB_EN bit is clear) 00 ARM platform 01 CSI direct (cb7) 10 Reserved 10 VDOA 11 Reserved
27–24 PRP_SRC_SEL	Source select for the Pre Processing Task 0000 ARM platform

Table continues on the next page...

IPUx_FS_PROC_FLOW1 field descriptions (continued)

Field	Description
	0001 capture0 (smfc0) — — — 0011 capture2 (smfc2) — — — 0101 IC direct (cb7) — 0110 IRT Encoding 0111 IRT viewfinder 1000 Reserved 1001 Reserved 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
23–22 VDI3_SRC_SEL	Source select for the VDIC plane #3 (IDMAC's CH 25) 00 ARM platform This field is relevant only if the VDIC works in combining mode (VDI_CMB_EN bit is set) 01 IRT viewfinder (ch 49) 10 IRT playback (ch 50) 11 post-processing (ch 22)
21–20 VDI1_SRC_SEL	Source select for the VDIC plane #1 (IDMAC's CH26) This field is relevant only if the VDIC works in combining mode (VDI_CMB_EN bit is set) 00 ARM platform 01 IRT viewfinder 10 IRT playback 11 post-processing
19–16 PP_ROT_SRC_SEL	Source select for the pre processing task of the IRT (CH 50) 0000 ARM platform 0001 capture0 (smfc0) — 0010 Reserved 0011 capture2 (smfc2) — 0100 Reserved 0101 Post-processing 0110 Reserved 0111 Reserved —

Table continues on the next page...

IPUx_FS_PROC_FLOW1 field descriptions (continued)

Field	Description
	1000 Reserved — 1001 Reserved 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
15–12 PP_SRC_SEL	Source select for the pre processing task of the IC 0000 ARM platform 0001 capture0 (smfc0) — 0010 Reserved 0011 capture2 (smfc2) — 0100 Reserved 0101 Reserved 0110 Rotation for post-processing 0111 Reserved — 1000 Reserved 1000 VDOA — 1001 Reserved 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
11–8 PRPVF_ROT_ SRC_SEL	Source select for the view finder task of the IRT 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture1 (smfc1) — 0011 capture2 (smfc2) — 0100 capture3 (smfc3) — 0101 IC direct (cb7) — 0110 Reserved 0111 Reserved

Table continues on the next page...

IPUx_FS_PROC_FLOW1 field descriptions (continued)

Field	Description
	1000 View-finder 1001 Reserved 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
7–4 Reserved	This read-only field is reserved and always has the value 0.
PRPENC_ROT_SRC_SEL	Source select for the encoding task of the IRT 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture1 (smfc1) — 0011 capture2 (smfc2) — 0100 capture3 (smfc3) — 0101 IC direct (cb7) — 0110 Reserved 0111 encoding 1000 Reserved 1001 Reserved 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2

37.5.34 FSU Processing Flow 2 Register (IPUx_FS_PROC_FLOW2)

This register contains controls for IPU's tasks.

Address: Base address + ACh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				PRP_DEST_SEL				PRPENC_ROT_DEST_SEL				PP_ROT_DEST_SEL				PP_DEST_SEL				PRPVF_ROT_DEST_SEL				PRPVF_DEST_SEL				PRP_ENC_DEST_SEL			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_FS_PROC_FLOW2 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–24 PRP_DEST_SEL	**Pre processing destination select (for channel DMAIC_7)** 0000 ARM platform 0001 IC input buffer (ch12) 0010 PP (ch11) 0011 PP_ROT (ch47) 0100 DC1 (ch28) 0101 DC2 (ch41) 0110 DP_ASYNC1 (ch24) 0111 DP_ASYNC0 (ch29) 1000 DP_SYNC1 (ch27) 1001 DP_SYNC0 (ch23) 1010 Alt DC2 (ch41) 1011 Alt DP_ASYNC1 (ch24) 1100 Alt DP_ASYNC0 (ch29) 1111 Reserved
23–20 PRPENC_ROT_DEST_SEL	Destination select for Rotation task coming from the Encoding input 0000 ARM platform 0001 Reserved 0010 Reserved — 0011 Reserved — 0100 Reserved 0101 IC Pre Processing 0110 Reserved 0111 DC1 (ch28) 1000 DC2 (ch41)

Table continues on the next page...

IPUx_FS_PROC_FLOW2 field descriptions (continued)

Field	Description
	1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
19–16 PP_ROT_DEST_SEL	Destination select for Rotation task coming from the Post Processing input 0000 ARM platform 0001 Reserved 0010 Reserved 0011 Reserved 0100 IC Playback (Post Processing) — — 0101 VDI_PLANE3 (Ch 25) — — 0110 VDI_PLANE1 (Ch 26) 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
15–12 PP_DEST_SEL	Destination select for post processing task 0000 ARM platform 0001 Reserved 0010 Reserved 0011 IRT playback — 0100 VDI_PLANE3 (Ch 25) — 0101 VDI_PLANE1 (Ch 26) 0110 Reserved 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41)

Table continues on the next page...

IPUx_FS_PROC_FLOW2 field descriptions (continued)

Field	Description
	1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
11–8 PRPVF_ROT_ DEST_SEL	Destination select for Rotation task coming from the View finder input 0000 ARM platform 0001 Reserved 0010 Reserved — — 0011 VDI_PLANE3 (Ch 25) — — 0100 VDI_PLANE1 (Ch 26) 0101 IC Pre Processing 0110 Reserved 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
7–4 PRPVF_DEST_ SEL	Destination select for View finder task 0000 ARM platform 0001 IRT viewfinder 0010 Reserved 0011 Reserved 0100 Reserved 0101 Reserved 0110 Reserved 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
PRP_ENC_ DEST_SEL	Destination select for Encoding task 0000 ARM platform 0001 IRT Encoding 0010 Reserved

Table continues on the next page...

IPUx_FS_PROC_FLOW2 field descriptions (continued)

Field	Description
	0011 Reserved 0100 Reserved 0101 Reserved 0110 Reserved 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**

37.5.35 FSU Processing Flow 3 Register (IPUx_FS_PROC_FLOW3)

This register contains controls for IPU's tasks.

- Hide VPU_SUB_FRAME_SYNC for all versions
- Show VPU_SUB_FRAME_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.
- This requires some sophisticated conditional tag settings

Address: Base address + B0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0						VPU_DEST_SEL		EXT_SRC2_DEST_SEL		EXT_SRC1_DEST_SEL		0		VDOA_DEST_SEL	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0		SMFC3_DEST_SEL			SMFC2_DEST_SEL			SMFC1_DEST_SEL				SMFC0_DEST_SEL			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_FS_PROC_FLOW3 field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25–24 VPU_DEST_SEL	This bits selects the corresponding IDMAC channel's EOL indication to be used for sub frame synchronization with the VPU. The corresponding IDMAC channel's EOLI bit at the CPMEM has to be set as well.

Table continues on the next page...

IPUx_FS_PROC_FLOW3 field descriptions (continued)

Field	Description
	00 disabled 01 capture0 (smfc0) **(ch0)** 10 capture2 (smfc2) **(ch2)** 11 IC viewfinder (ch21)
23–22 EXT_SRC2_ DEST_SEL	Destination select for External Source 2 00 disabled 01 **DP_SYNC0 (ch23)** 10 **DP_SYNC1 (ch27)** 11 DC1 (ch28)
21–20 EXT_SRC1_ DEST_SEL	Destination select for External Source 1 00 disabled 01 **DP_SYNC0 (ch23)** 10 **DP_SYNC1 (ch27)** 11 DC1 (ch28)
19–18 Reserved	This read-only field is reserved and always has the value 0.
17–16 VDOA_DEST_ SEL	**Destination select for VDOA** 00 disabled 01 IC Playback (Post Processing) 10 VDI (ch8,ch9 & ch10 or ch9 according to VDI_MOT_SEL settings) 11 Reserved
15–14 Reserved	This read-only field is reserved and always has the value 0.
13–11 SMFC3_DEST_ SEL	Destination select for SMFC3 000 ARM platform 001 IRT Encoding 010 IRT viewfinder 011 IRT playback 100 IC Playback (Post Processing) 101 IC Pre Processing — 111 Reserved
10–7 SMFC2_DEST_ SEL	Destination select for SMFC2 0000 ARM platform 0001 IRT Encoding 0010 IRT viewfinder 0011 IRT playback 0100 IC Playback (Post Processing) 0101 IC Pre Processing — 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)**

Table continues on the next page...

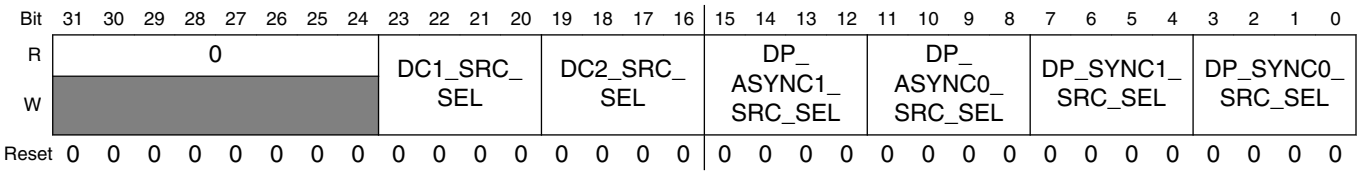
IPUx_FS_PROC_FLOW3 field descriptions (continued)

Field	Description
	1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**
6–4 SMFC1_DEST_SEL	Destination select for SMFC1 000 ARM platform 001 IRT Encoding 010 IRT viewfinder 011 IRT playback 100 IC Playback (Post Processing) 101 IC Pre Processing — 111 Reserved
SMFC0_DEST_SEL	Destination select for SMFC0 0000 ARM platform 0001 IRT Encoding 0010 IRT viewfinder 0011 IRT playback 0100 IC Playback (Post Processing) 0101 IC Pre Processing — 0111 DC1 (ch28) 1000 DC2 (ch41) 1001 **DP_SYNC0 (ch23)** 1010 **DP_SYNC1 (ch27)** 1011 **DP_ASYNC1 (ch24)** 1100 **DP_ASYNC0 (ch29)** 1101 Alt DC2 (ch41) 1110 Alt **DP_ASYNC1 (ch24)** 1111 Alt **DP_ASYNC0 (ch29)**

37.5.36 FSU Displaying Flow 1 Register (IPUx_FS_DISP_FLOW1)

This register contains controls for IPU's tasks.

Address: Base address + B4h offset



IPUx_FS_DISP_FLOW1 field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–20 DC1_SRC_SEL	Source select for DS1/DS2 - MG (graphics) plane (ch28) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder 0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 — 1101 External source #1 (e.g. an external block like GPU) 1110 snoop1 — 1111 External source #2 (e.g. an external block like GPU)
19–16 DC2_SRC_SEL	Source select for DS3 (ch41) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2)

Table continues on the next page...

IPUx_FS_DISP_FLOW1 field descriptions (continued)

Field	Description
	—
	0011 IC encoding
	0100 IC viewfinder
	0101 IC playback
	0110 IRT Encoding
	0111 IRT viewfinder
	1000 IRT playback
	—
	1001 Reserved
	—
	1010 Reserved
	1011 autoref
	1100 autoref+snoop1
	1101 autoref+snoop2
	1110 snoop1
	1111 snoop2
15–12 DP_ASYNC1_ SRC_SEL	Source select for DS1/DS2 - Vx (video) plane (ch24) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder 0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
11–8 DP_ASYNC0_ SRC_SEL	Source select for DS2 - MG (graphics) plane (ch29) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder

Table continues on the next page...

IPUx_FS_DISP_FLOW1 field descriptions (continued)

Field	Description
	0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
7–4 DP_SYNC1_ SRC_SEL	Source select for DS1/DS2 - Vx (video) plane (ch27) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder 0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved — — — — — — 1110 snoop1 1111 snoop2
DP_SYNC0_ SRC_SEL	Source select for DS2 - MG (graphics) plane (ch23) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder

Table continues on the next page...

IPUx_FS_DISP_FLOW1 field descriptions (continued)

Field	Description
	0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved — — — — — — 1110 snoop1 1111 snoop2

37.5.37 FSU Displaying Flow 2 Register (IPUx_FS_DISP_FLOW2)

This register contains controls for IPU's tasks.

Address: Base address + B8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0												DC2_ALT_SRC_SEL				0								DP_ASYNC1_ALT_SRC_SEL				DP_ASYNC0_ALT_SRC_SEL			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_FS_DISP_FLOW2 field descriptions

Field	Description
31–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 DC2_ALT_SRC_SEL	Source select for Alternate DS3 (ch41) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder

Table continues on the next page...

IPUx_FS_DISP_FLOW2 field descriptions (continued)

Field	Description
	0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
15–8 Reserved	This read-only field is reserved and always has the value 0.
7–4 DP_ASYNC1_ ALT_SRC_SEL	Source select for alternate DS1/DS2 - Vx (video) plane (ch24) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder 0101 IC playback 0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2
DP_ASYNC0_ ALT_SRC_SEL	Source select for alternate DS2 - MG (graphics) plane (ch29) 0000 ARM platform 0001 capture0 (smfc0) — 0010 capture2 (smfc2) — 0011 IC encoding 0100 IC viewfinder 0101 IC playback

Table continues on the next page...

IPUx_FS_DISP_FLOW2 field descriptions (continued)

Field	Description
	0110 IRT Encoding 0111 IRT viewfinder 1000 IRT playback — 1001 Reserved — 1010 Reserved 1011 autoref 1100 autoref+snoop1 1101 autoref+snoop2 1110 snoop1 1111 snoop2

37.5.38 SKIP Register (IPUx_SKIP)

This register controls the different frame skipping supported by the IPU.

Address: Base address + BCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	VDI_SKIP												VDI_MAX_ RATIO_SKIP				CSI_SKIP_IC_VF					CSI_ MAX_ RATIO_ SKIP_IC_ VF			CSI_SKIP_IC_ ENC					CSI_ MAX_ RATIO_ SKIP_IC_ ENC		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SKIP field descriptions

Field	Description
31–20 VDI_SKIP	VDI_SKIP These 12 bits define the skipping pattern of the frames send from the VDIC. The VDIC avoids reading fields from the memory if the output frame is skipped. Skipping is relevant only if the source to the VDIC is coming from the CSI. Skipping is done for a set of frames. The number of frames in a set is defined at VDI_MAX_RATIO_SKIP. when VDI_MAX_RATIO_SKIP = 1 => VDI_SKIP[1:0] is used; other bits are ignored when VDI_MAX_RATIO_SKIP = 2 => VDI_SKIP[2:0] are used; other bits are ignored .. .. when VDI_MAX_RATIO_SKIP = 11 => VDI_SKIP[11:0] are used;
19–16 VDI_MAX_ RATIO_SKIP	Maximum Ratio Skip for VDIC

Table continues on the next page...

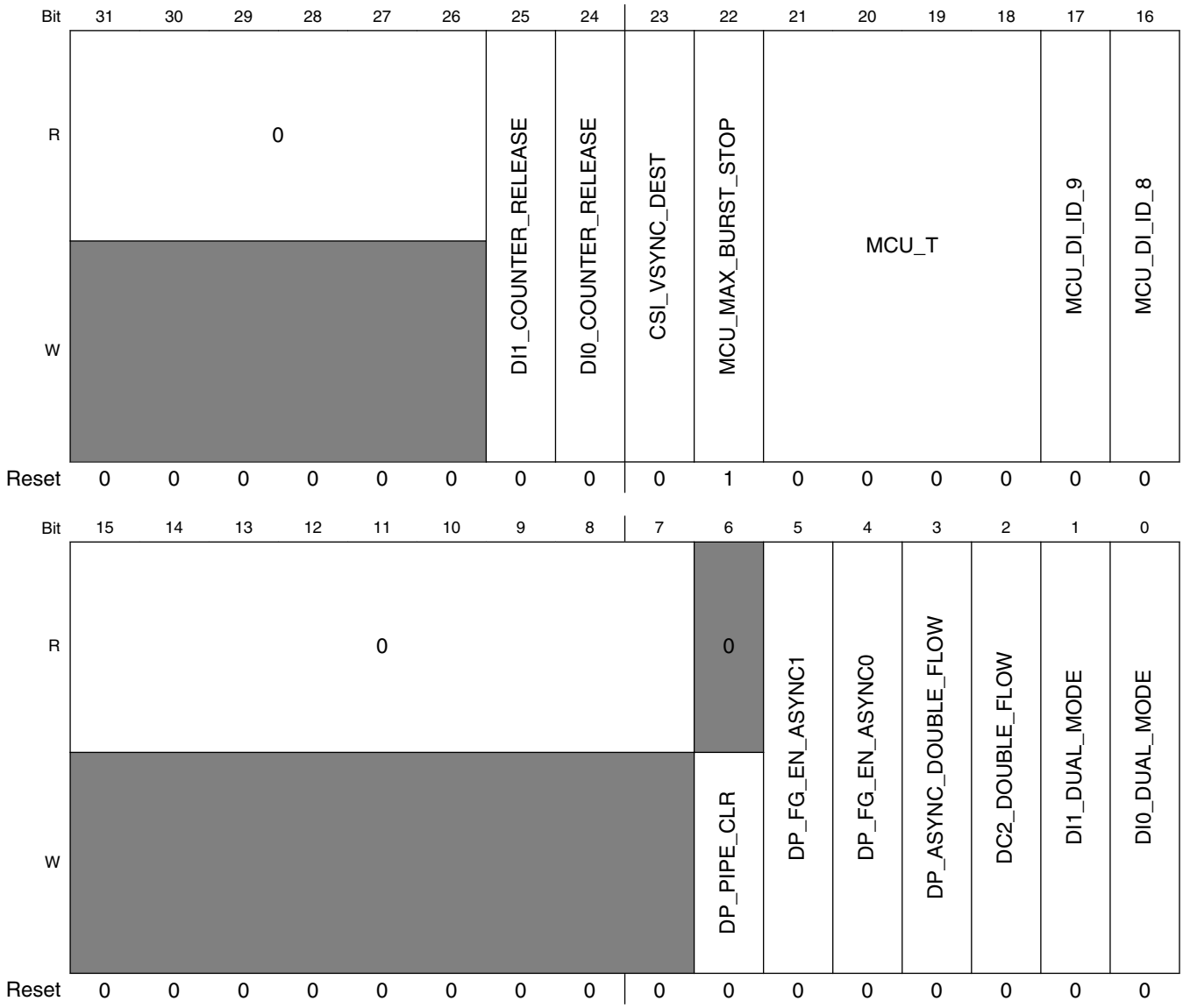
IPUx_SKIP field descriptions (continued)

Field	Description
	These bits define the number of frames in a skipping set. The maximum value of this bits is 11. When set to 0 the skipping is disabled.
15–11 CSI_SKIP_IC_VF	CSI SKIP IC_VF These 5 bits define the skipping pattern of the frames send to the IC for view finder task from one of the CSIs as defined on the CSI_SEL and IC_INPUT bits Skipping is done for a set of frames. The number of frames in a set is defined at CSI_MAX_RATIO_SKIP_IC_VF. when CSI_MAX_RATIO_SKIP_IC_VF = 1 => CSI_SKIP_IC_VF[1:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_VF = 2 => CSI_SKIP_IC_VF[2:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_VF =3 => CSI_SKIP_IC_VF[3:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_VF = 4 => CSI_SKIP_IC_VF[4:0] are used; Setting bit #n of CSI_SKIP_IC_VF means that the #n frame in the set is skipped. For example: if CSI_MAX_RATIO_SKIP_IC_VF = 4 and CSI_SKIP_IC_VF = 11010 Frames #0 & Frame #2 will not be skipped as bit0 and bit2 are cleared Frames #1 & Frame #3 will be skipped as bit1 and bit3 are set bit #4 is ignored as CSI_MAX_RATIO_SKIP_IC_VF is set to 4
10–8 CSI_MAX_ RATIO_SKIP_ IC_VF	CSI Maximum Ratio Skip for IC (view finder task) These bits define the number of frames in a skipping set. The maximum value of this bits is 4. When set to 0 the skipping is disabled.
7–3 CSI_SKIP_IC_ ENC	CSI SKIP IC_ENC These 5 bits define the skipping pattern of the frames send to the IC for encoding task from one of the CSIs as defined on the CSI_SEL and IC_INPUT bits Skipping is done for a set of frames. The number of frames in a set is defined at CSI_MAX_RATIO_SKIP_IC_ENC. when CSI_MAX_RATIO_SKIP_IC_ENC = 1 => CSI_SKIP_IC_ENC[1:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_ENC = 2 => CSI_SKIP_IC_ENC[2:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_ENC = 3 => CSI_SKIP_IC_ENC[3:0] are used; other bits are ignored when CSI_MAX_RATIO_SKIP_IC_ENC = 4 => CSI_SKIP_IC_ENC[4:0] are used; Setting bit #n of CSI_SKIP_IC_ENC means that the #n frame in the set is skipped. For example: if CSI_MAX_RATIO_SKIP_IC_ENC = 4 and CSI_SKIP_IC_ENC = 11010 Frames #0 & Frame #2 will not be skipped as bit0 and bit2 are cleared Frames #1 & Frame #3 will be skipped as bit1 and bit3 are set bit #4 is ignored as CSI_MAX_RATIO_SKIP_IC_ENC is set to 4
CSI_MAX_ RATIO_SKIP_ IC_ENC	CSI Maximum Ratio Skip for IC (encoding task) These bits define the number of frames in a skipping set. The maximum value of this bits is 4. When set to 0 the skipping is disabled.

37.5.39 Display General Control Register (IPUx_DISP_GEN)

This register controls various aspects of the display port.

Address: Base address + C4h offset



IPUx_DISP_GEN field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DISP_GEN field descriptions (continued)

Field	Description
25 DI1_COUNTER_RELEASE	DI1 Counter release By default the DI0 counters responsible for waveform generation for sync flow are frozen. For the first attempt to use the DI in sync flow the user should set this bit 1 counter is released and running 0 counter is cleared and stopped
24 DI0_COUNTER_RELEASE	DI1 Counter release By default the DI0 counters responsible for waveform generation for sync flow are frozen. For the first attempt to use the DI in sync flow the user should set this bit 1 counter is released and running 0 counter is cleared and stopped
23 CSI_VSYNC_DEST	CSI_VSYNC destination This bit defines the destination of the VSYNC coming from the CSI's 1 csi1_vsync is connected to DI0; csi0_vsync is connected to DI1 0 csi0_vsync is connected to DI0; csi1_vsync is connected to DI1
22 MCU_MAX_BURST_STOP	ARM platform Maximal burst This bit limit the maximal unspecified length burst. 1 The maximum unspecified burst length is 8-beat 0 The unspecified burst length is unlimited
21–18 MCU_T	The address space for accesses through the AHB-lite slave port is MB and it is split internally (with 32MB resolution) according to bits [28:25] of the address. Using the following notation: Address = (ID[31:29], MSB[28:25], LSB[24:0]) The address is used as follows ("T" is a configurable integer between 0 and 13): MSB<T: access to an external device, with address = (MSB, LSB) T<=MSB<14: access to an external device, with address (MSB-T, LSB)
17 MCU_DI_ID_9	MCU_DI_ID_9 - DI ID via DC channel 9. This bit defines the DI that the ARM platform DC's access via channel #9 1 ARM platform accesses DC's channel #9 via DI1. 0 ARM platform accesses DC's channel #9 via DI0.
16 MCU_DI_ID_8	MCU_DI_ID_8 - DI ID via DC channel 8. This bit defines the DI that the ARM platform DC's access via channel #8 1 ARM platform accesses DC's channel #8 via DI1. 0 ARM platform accesses DC's channel #8 via DI0.
15–7 Reserved	This read-only field is reserved and always has the value 0.
6 DP_PIPE_CLR	DP Pipe Clear This bit clears the internal pipe of the DP. The user may use this bit in case of an error condition This is a self clear bit

Table continues on the next page...

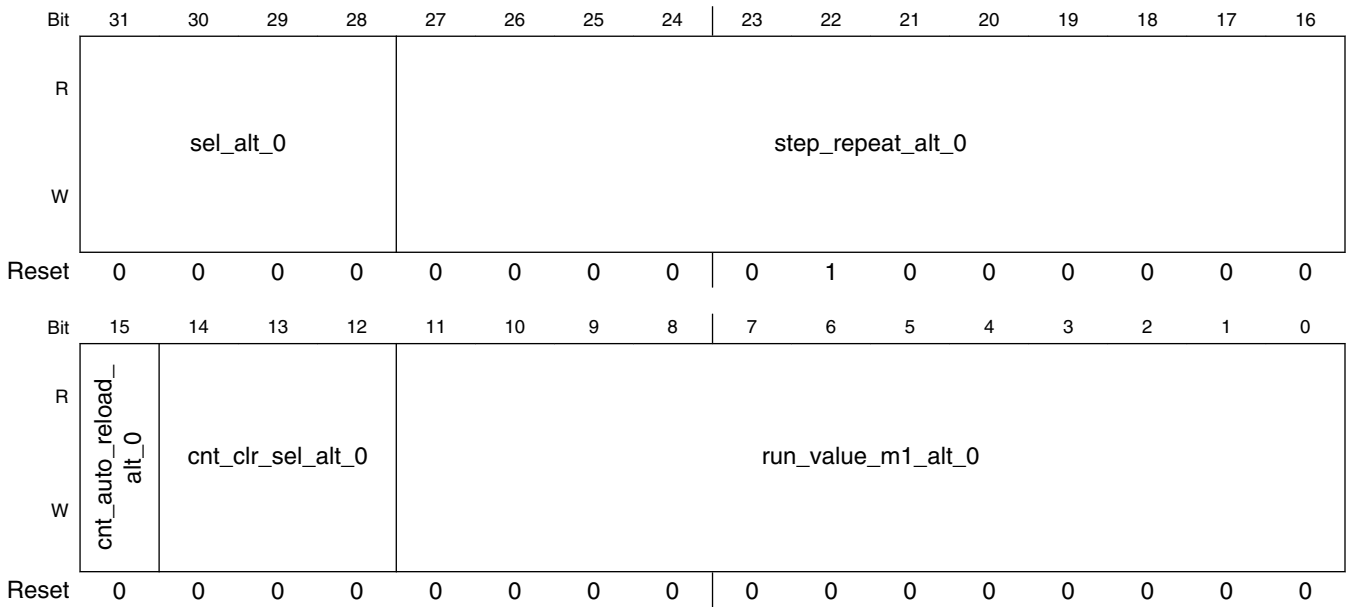
IPUx_DISP_GEN field descriptions (continued)

Field	Description
	1 Clear the internal pipe of the DP 0 Idle - does nothing
5 DP_FG_EN_ASYNC1	FG_EN - partial plane Enable for async flow 1. This bit enables the partial plane channel. 1 partial plane channel is enabled. 0 partial plane channel is disabled.
4 DP_FG_EN_ASYNC0	FG_EN - partial plane Enable for async flow 0. This bit enables the partial plane channel. 1 partial plane channel is enabled. 0 partial plane channel is disabled.
3 DP_ASYNC_DOUBLE_FLOW	DP Async Double Flow. This bit define how many async flows are currently handles via DP channel (ch24+29) 1 2 flows are handled via DP 0 single flow is handled via DP
2 DC2_DOUBLE_FLOW	DC2 Double Flow. This bit define how many flows are currently handles via DC2 channel (ch41) 1 2 flows are handled via DC2 0 single flow is handled via DC2
1 DI1_DUAL_MODE	DI1 dual mode control 1 DI1 operates in dual mode 0 DI1 is not in dual mode
0 DI0_DUAL_MODE	DI0 dual mode control 1 DI0 operates in dual mode 0 DI0 is not in dual mode

37.5.40 Display Alternate Flow Control Register 1 (IPUx_DISP_ALT1)

This register controls various aspects of the display port.

Address: Base address + C8h offset



IPUx_DISP_ALT1 field descriptions

Field	Description
31–28 sel_alt_0	Select alternative parameters instead of DI Sync Wave Gen counter#. The DI is selected according to DP's synchronous channel destination 0000-disable 0001 instead of counter 1 0010 instead of counter 2 1000 instead of counter 8
27–16 step_repeat_alt_0	This fields defines the amount of repetitions that will be performed by the counter
15 cnt_auto_reload_alt_0	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the step_repeat_alt_0 field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the step_repeat_alt_0 field
14–12 cnt_clr_sel_alt_0	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock.

Table continues on the next page...

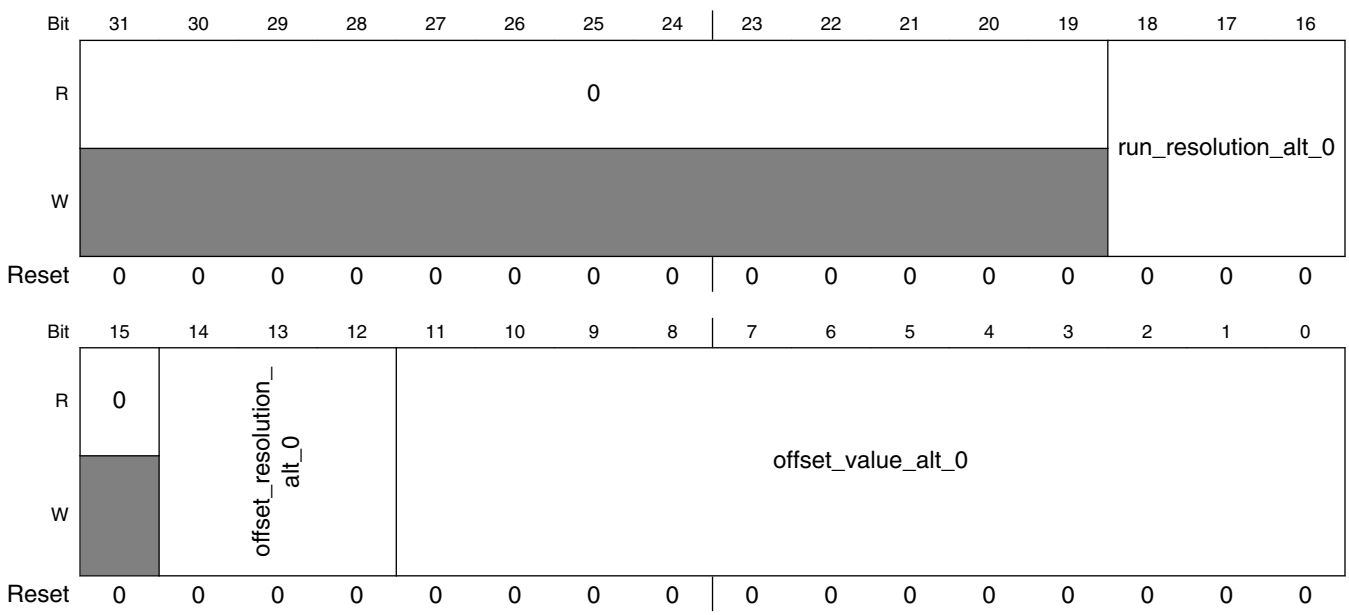
IPUx_DISP_ALT1 field descriptions (continued)

Field	Description
	010 Reserved 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
run_value_m1_alt_0	Counter pre defined value This fields defines the counter pre defines value. real value- 1

37.5.41 Display Alternate Flow Control Register 2 (IPUx_DISP_ALT2)

This register controls various aspects of the display port.

Address: Base address + CCh offset



IPUx_DISP_ALT2 field descriptions

Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DISP_ALT2 field descriptions (continued)

Field	Description
18–16 run_resolution_alt_0	Counter Run Resolution This field defines the trigger causing the counter to increment. The counter run resolution should be defined in the same way as in original DI's counter#
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 offset_resolution_alt_0	Counter offset Resolution This field defines the trigger causing the offset counter to increment The counter offset resolution should be defined in the same way as in original DI's counter#
offset_value_alt_0	Counter offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by

37.5.42 Display Alternate Flow Control Register 3 (IPUx_DISP_ALT3)

This register controls various aspects of the display port.

Address: Base address + D0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	sel_alt_1				step_repeat_alt_1											
Reset	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	cnt_auto_reload_alt_1	cnt_clr_sel_alt_1			run_value_m1_alt_1											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DISP_ALT3 field descriptions

Field	Description
31–28 sel_alt_1	Select alternative parameters instead of DI Sync Wave Gen counter#. The DI is selected according to DP's synchronous channel destination

Table continues on the next page...

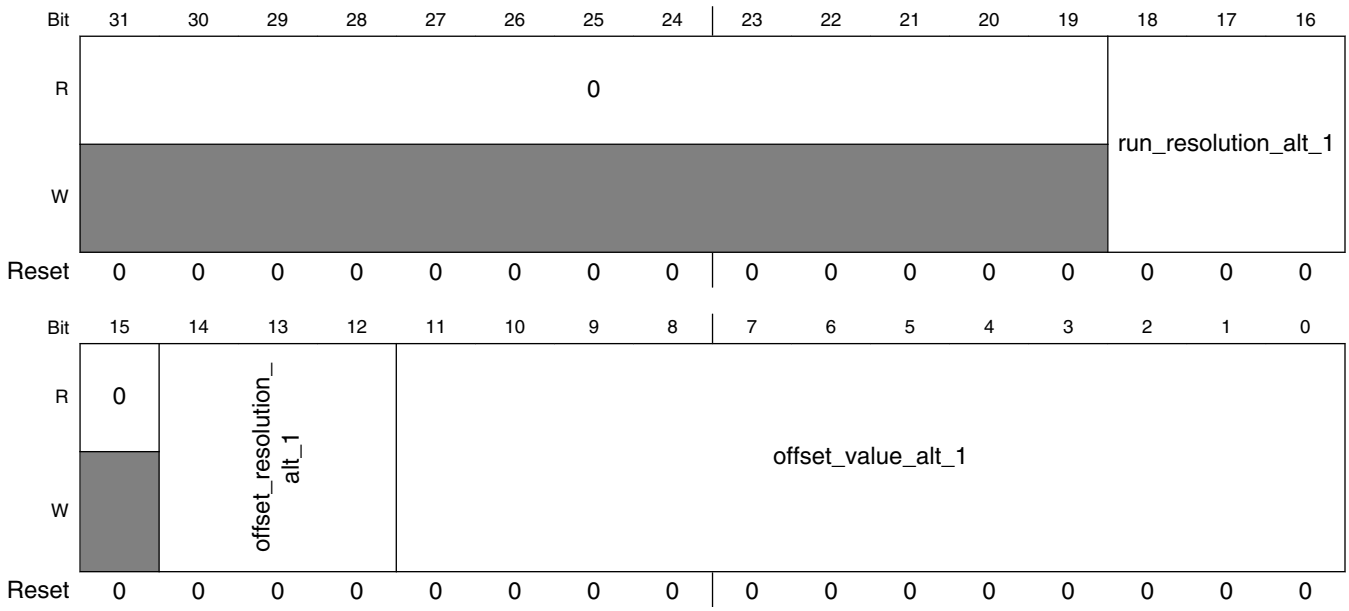
IPUx_DISP_ALT3 field descriptions (continued)

Field	Description
	0000 disable 0001 instead of counter 1 0010 instead of counter 2 1000 instead of counter 8
27–16 step_repeat_alt_1	This fields defines the amount of repetitions that will be performed by the counter
15 cnt_auto_reload_alt_1	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the step_repeat_alt_0 field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the step_repeat_alt_0 field
14–12 cnt_clr_sel_alt_1	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 Reserved 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
run_value_m1_alt_1	Counter pre defined value This fields defines the counter pre defines value. real value- 1

37.5.43 Display Alternate Flow Control Register 4 (IPUx_DISP_ALT4)

This register controls various aspects of the display port.

Address: Base address + D4h offset



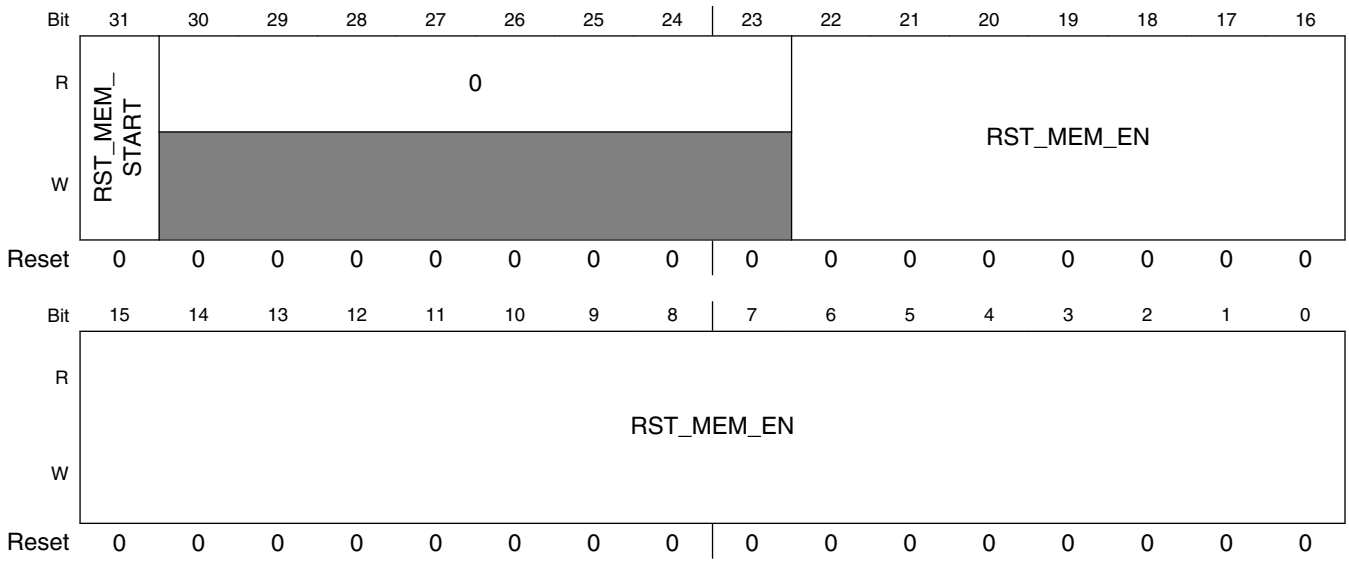
IPUx_DISP_ALT4 field descriptions

Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0.
18–16 run_resolution_alt_1	Counter Run Resolution This field defines the trigger causing the counter to increment. The counter run resolution should be defined in the same way as in original DI's counter#
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 offset_resolution_alt_1	Counter offset Resolution This field defines the trigger causing the offset counter to increment The counter offset resolution should be defined in the same way as in original DI's counter#
offset_value_alt_1	Counter offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by

37.5.44 Memory Reset Control Register (IPUx_MEM_RST)

This register controls the memory reset mechanism. IPU has a hardware mechanism for clearing the content of the internal memories. This allows the user to clear the content of or more of the internal memories without the need to perform write accesses to the memories.

Address: Base address + DCh offset



IPUx_MEM_RST field descriptions

Field	Description
31 RST_MEM_ START	Memory Reset Start Writing one to this bit activate the memory reset mechanism. The memories that their corresponding RST_MEM_EN bit is set will be cleared. When the memory reset mechanism completes the memory clearing procedure this bit will be automatically cleared. 1 The memory reset mechanism is activated and busy 0 Idle, the memory reset mechanism is not working.
30–23 Reserved	This read-only field is reserved and always has the value 0.
RST_MEM_EN	Reset Memory Enable Each bit on this field enables the memory reset mechanism for a specific memory. The user should set the relevant bits for the memories that need to be cleared. Below is the list of memories and their corresponding bit. srm = rst_mem_en[0] alpha = rst_mem_en[1] cpmem = rst_mem_en[2]

Table continues on the next page...

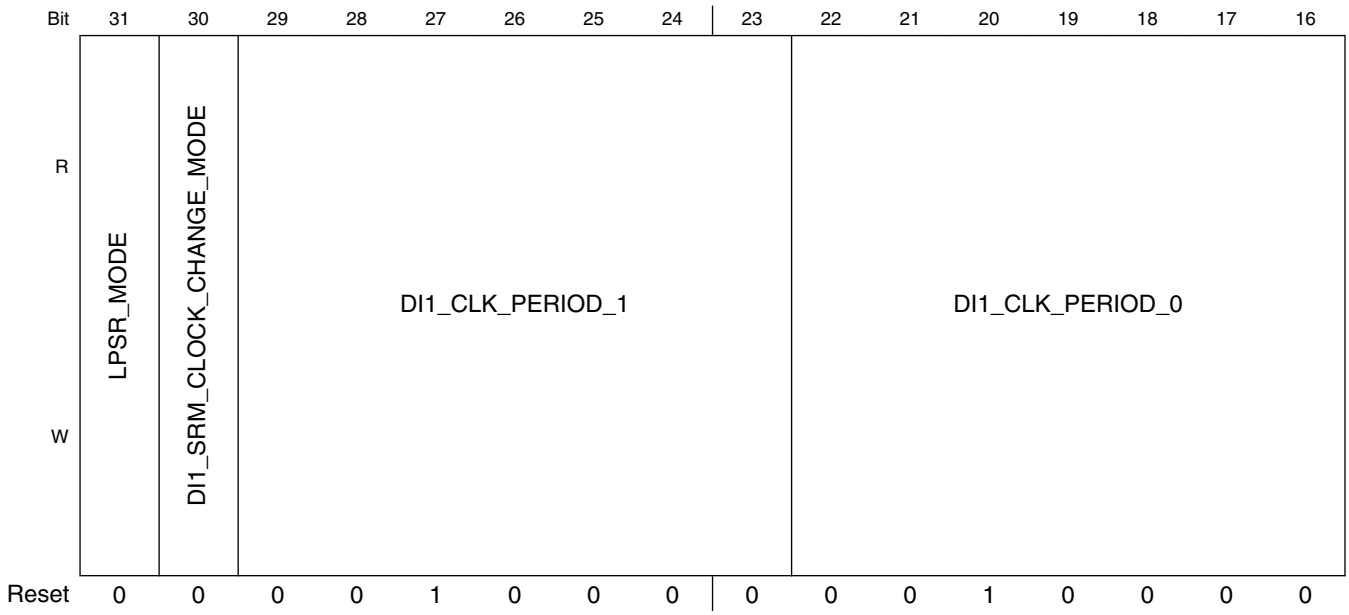
IPUx_MEM_RST field descriptions (continued)

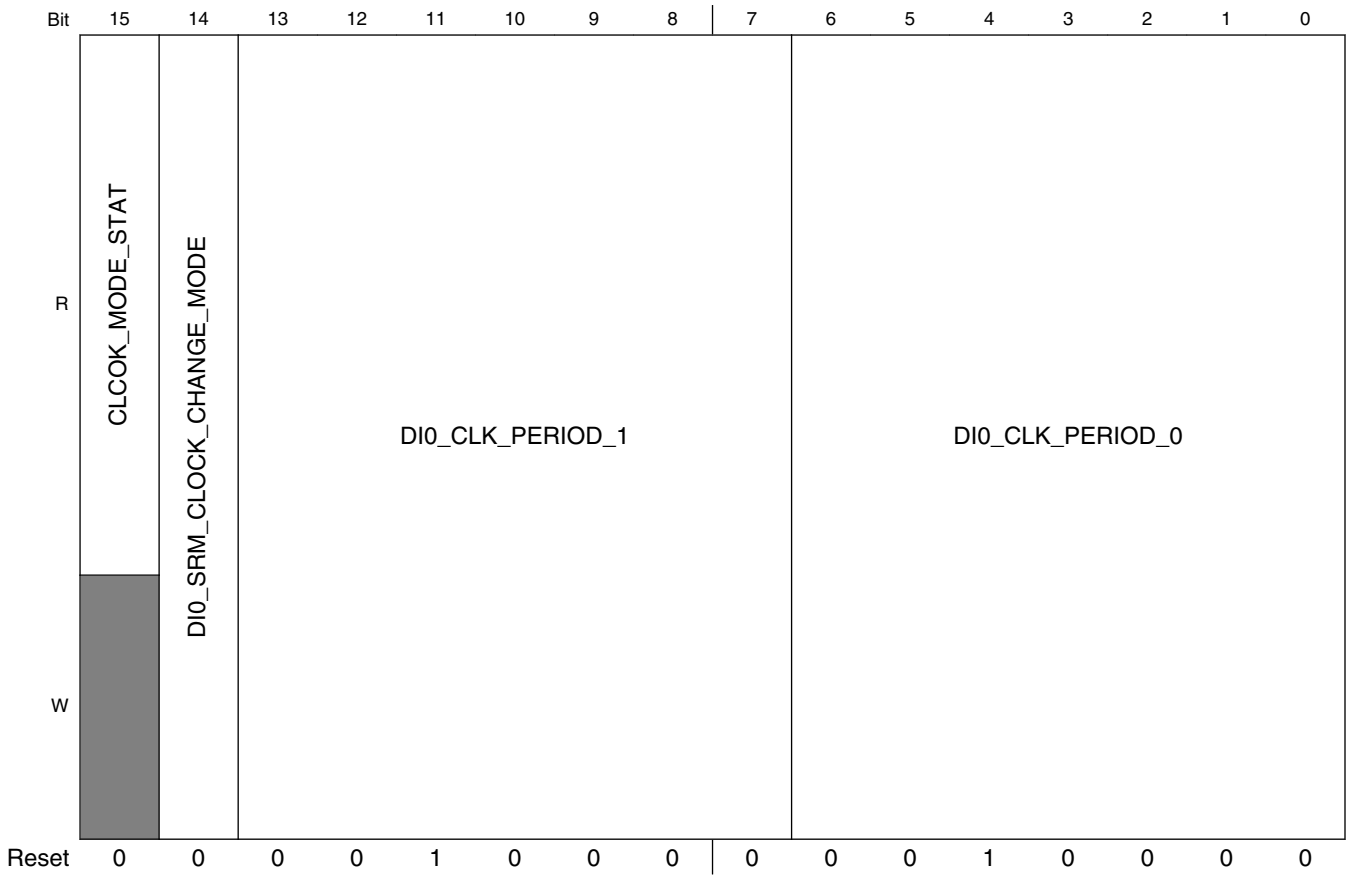
Field	Description
	tpm = rst_mem_en[3] mpm = rst_mem_en[4] bm = rst_mem_en[5] rm = rst_mem_en[6] dstm = rst_mem_en[7] dsom = rst_mem_en[8] lut0 = rst_mem_en[9] lut1 = rst_mem_en[10] ram_smfc = rst_mem_en[11] vdi_fifo2 = rst_mem_en[12] vdi_fifo3 = rst_mem_en[13] icb = rst_mem_en[14] vdi_fifo1 = rst_mem_en[15] dc_template = rst_mem_en[20] dmfc_rd = rst_mem_en[21] dmfc_wr = rst_mem_en[22]

37.5.45 Power Modes Control Register (IPUx_PM)

This register controls the automatic transitions of the IPU between different power modes of the SoC and handles the clock change modes.

Address: Base address + E0h offset





IPUx_PM field descriptions

Field	Description
31 LPSR_MODE	LPSR Mode This bit indicates that the next attempt for entering low power mode is an attempt to move to LPST mode. Setting this bit by the user is essential in order to assure proper response of the IPU to the assertion of the stop request from the CCM. 1 Next low power mode will be LPSR 0 Next low power mode is not LPSR
30 DI1_SRM_ CLOCK_ CHANGE_MODE	SRM clock change mode When the clock is going to be changed to any new ratio other then 1:1, 1:2. 1:4. The user needs to prepare an alternate set of DI setting in the SRM. This bit enable this mode. This bit is self cleared. 1 SRM clock change mode is enabled; the next clock change will be done by updating the DI settings from the SRM 0 SRM clock change mode is disabled.
29–23 DI1_CLK_ PERIOD_1	DI1_CLK period option 1. This parameter defines the period of the clock that the DI1 works with. This parameter contains integer part (bits [6:4]) and fractional part (bits [3:0]). Setting this value to 1.0 (default) means that the DI1 works on the fastest possible clock. Setting a value smaller than 1.0 is not allowed.

Table continues on the next page...

IPUx_PM field descriptions (continued)

Field	Description
	The value to be programmed to the DI1_CLK_PERIOD_1 field is equal to: Fast_freq/Target_freq Where: Target_freq = The frequency that the DI clock works with Fast_freq = fastest possible clock that the DI can work with
22–16 DI1_CLK_ PERIOD_0	DI1_CLK period option 0. This parameter defines the period of the clock that the DI1 works with. This parameter contains integer part (bits [6:4]) and fractional part (bits [3:0]). Setting this value to 1.0 (default) means that the DI1 works on the fastest possible clock. Setting a value smaller than 1.0 is not allowed. The value to be programmed to the DI1_CLK_PERIOD_1 field is equal to: Fast_freq/Target_freq Where: Target_freq = The frequency that the DI clock works with Fast_freq = fastest possible clock that the DI can work with
15 CLCOK_MODE_ STAT	Clock mode status This is a read only bit indicating what is the current clock mode 1 current clock mode is 1 0 current clock mode is 0
14 DI0_SRM_ CLOCK_ CHANGE_MODE	SRM clock change mode When the clock is going to be changed to any new ratio other then 1:1, 1:2. 1:4. The user needs to prepare an alternate set of DI setting in the SRM. This bit enable this mode. This bit is self cleared. 1 SRM clock change mode is enabled; the next clock change will be done by updating the DI settings from the SRM 0 SRM clock change mode is disabled.
13–7 DI0_CLK_ PERIOD_1	DI0_CLK period option 1. This parameter defines the period of the clock that the DI0 works with. This parameter contains integer part (bits [6:4]) and fractional part (bits [3:0]). Setting this value to 1.0 (default) means that the DI0 works on the fastest possible clock. Setting a value smaller than 1.0 is not allowed. The value to be programmed to the DI0_CLK_PERIOD_1 field is equal to: Fast_freq/Target_freq Where: Target_freq = The frequency that the DI clock works with Fast_freq = fastest possible clock that the DI can work with
DI0_CLK_ PERIOD_0	DI0_CLK period option 0. This parameter defines the period of the clock that the DI0 works with. This parameter contains integer part (bits [6:4]) and fractional part (bits [3:0]).

Table continues on the next page...

IPUx_PM field descriptions (continued)

Field	Description
	Setting this value to 1.0 (default) means that the DI0 works on the fastest possible clock. Setting a value smaller than 1.0 is not allowed. The value to be programmed to the DI0_CLK_PERIOD_1 field is equal to: Fast_freq/Target_freq Where: Target_freq = The frequency that the DI clock works with Fast_freq = fastest possible clock that the DI can work with

37.5.46 General Purpose Register (IPUx_GPR)

The register contains general purpose bits.

Address: Base address + E4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	IPU_CH_BUF1_RDY1_CLR	IPU_CH_BUF1_RDY0_CLR	IPU_CH_BUF0_RDY1_CLR	IPU_CH_BUF0_RDY0_CLR	IPU_ALT_CH_BUF1_RDY1_CLR	IPU_ALT_CH_BUF1_RDY0_CLR	IPU_ALT_CH_BUF0_RDY1_CLR	IPU_ALT_CH_BUF0_RDY0_CLR	IPU_DI1_CLK_CHANGE_ACK_DIS	IPU_DI0_CLK_CHANGE_ACK_DIS	IPU_CH_BUF2_RDY1_CLR	IPU_CH_BUF2_RDY0_CLR	IPU_GPn			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IPU_GPn															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_GPR field descriptions

Field	Description
31 IPU_CH_BUF1_RDY1_CLR	This bit defines the IPU_CH_BUF1_RDY1 properties. This register can be a write one to clear OR write one to set.

Table continues on the next page...

IPUx_GPR field descriptions (continued)

Field	Description
	1 writing one to a bit of this register clears this bit; **IPU_CH_BUF1_RDY1** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF1_RDY1** is w1s register
30 IPU_CH_BUF1_RDY0_CLR	This bit defines the **IPU_CH_BUF1_RDY0** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_CH_BUF1_RDY0** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF1_RDY0** is w1s register
29 IPU_CH_BUF0_RDY1_CLR	This bit defines the **IPU_CH_BUF0_RDY1** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_CH_BUF0_RDY1** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF0_RDY1** is w1s register
28 IPU_CH_BUF0_RDY0_CLR	This bit defines the **IPU_CH_BUF0_RDY0** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_CH_BUF0_RDY0** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF0_RDY0** is w1s register
27 IPU_ALT_CH_BUF1_RDY1_CLR	This bit defines the **IPU_ALT_CH_BUF1_RDY1** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_ALT_CH_BUF1_RDY1** is w1c register 0 writing one to a bit of this register sets this bit **IPU_ALT_CH_BUF1_RDY1** is w1s register
26 IPU_ALT_CH_BUF1_RDY0_CLR	This bit defines the **IPU_ALT_CH_BUF1_RDY0** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_ALT_CH_BUF1_RDY0** is w1c register 0 writing one to a bit of this register sets this bit **IPU_ALT_CH_BUF1_RDY0** is w1s register
25 IPU_ALT_CH_BUF0_RDY1_CLR	This bit defines the **IPU_ALT_CH_BUF0_RDY1** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_ALT_CH_BUF0_RDY1** is w1c register 0 writing one to a bit of this register sets this bit **IPU_ALT_CH_BUF0_RDY1** is w1s register
24 IPU_ALT_CH_BUF0_RDY0_CLR	This bit defines the **IPU_ALT_CH_BUF0_RDY0** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_ALT_CH_BUF0_RDY0** is w1c register 0 writing one to a bit of this register sets this bit **IPU_ALT_CH_BUF0_RDY0** is w1s register
23 IPU_DI1_CLK_CHANGE_ACK_DIS	Disable DI1's clock change mechanism. 1 clock change mechanism is disabled. DI automatically acknowledges a clock change request 0 clock change mechanism is disabled. DI performs the clock change procedure
22 IPU_DI0_CLK_CHANGE_ACK_DIS	Disable DI0's clock change mechanism. 1 clock change mechanism is disabled. DI automatically acknowledges a clock change request 0 clock change mechanism is disabled. DI performs the clock change procedure
21 IPU_CH_BUF2_RDY1_CLR	This bit defines the **IPU_CH_BUF2_RDY1** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_CH_BUF2_RDY1** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF2_RDY1** is w1s register

Table continues on the next page...

IPUx_GPR field descriptions (continued)

Field	Description
20 IPU_CH_BUF2_RDY0_CLR	This bit defines the **IPU_CH_BUF2_RDY0** properties. This register can be a write one to clear OR write one to set. 1 writing one to a bit of this register clears this bit; **IPU_CH_BUF2_RDY0** is w1c register 0 writing one to a bit of this register sets this bit **IPU_CH_BUF2_RDY0** is w1s register
IPU_GPn	IPU General Purpose bit. *n* Indicates the corresponding DMA channel number. This bits are general Read/Write bits, reserved for future use

37.5.47 Channel Double Buffer Mode Select 0 Register (IPUx_CH_DB_MODE_SEL0)

The register contains double buffer mode select control information for 32 IPU's DMA channels.

Address: Base address + 150h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	DMA_CH_DB_MODE_SEL_31	0	DMA_CH_DB_MODE_SEL_29	DMA_CH_DB_MODE_SEL_28	DMA_CH_DB_MODE_SEL_27	DMA_CH_DB_MODE_SEL_26	DMA_CH_DB_MODE_SEL_25	DMA_CH_DB_MODE_SEL_24	DMA_CH_DB_MODE_SEL_23	DMA_CH_DB_MODE_SEL_22	DMA_CH_DB_MODE_SEL_21	DMA_CH_DB_MODE_SEL_20	DMA_CH_DB_MODE_SEL_19	DMA_CH_DB_MODE_SEL_18	DMA_CH_DB_MODE_SEL_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DMA_CH_DB_MODE_SEL_15	DMA_CH_DB_MODE_SEL_14	DMA_CH_DB_MODE_SEL_13	DMA_CH_DB_MODE_SEL_12	DMA_CH_DB_MODE_SEL_11	DMA_CH_DB_MODE_SEL_10	DMA_CH_DB_MODE_SEL_9	DMA_CH_DB_MODE_SEL_8	0		DMA_CH_DB_MODE_SEL_5	0	DMA_CH_DB_MODE_SEL_3	DMA_CH_DB_MODE_SEL_2	DMA_CH_DB_MODE_SEL_1	DMA_CH_DB_MODE_SEL_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CH_DB_MODE_SEL0 field descriptions

Field	Description
31 DMA_CH_DB_MODE_SEL_31	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_CH_DB_MODE_SEL0 field descriptions (continued)

Field	Description
	0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
29 DMA_CH_DB_ MODE_SEL_29	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
28 DMA_CH_DB_ MODE_SEL_28	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
27 DMA_CH_DB_ MODE_SEL_27	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
26 DMA_CH_DB_ MODE_SEL_26	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
25 DMA_CH_DB_ MODE_SEL_25	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
24 DMA_CH_DB_ MODE_SEL_24	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
23 DMA_CH_DB_ MODE_SEL_23	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
22 DMA_CH_DB_ MODE_SEL_22	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

Table continues on the next page...

IPUx_CH_DB_MODE_SEL0 field descriptions (continued)

Field	Description
21 DMA_CH_DB_ MODE_SEL_21	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
20 DMA_CH_DB_ MODE_SEL_20	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
19 DMA_CH_DB_ MODE_SEL_19	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
18 DMA_CH_DB_ MODE_SEL_18	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
17 DMA_CH_DB_ MODE_SEL_17	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
15 DMA_CH_DB_ MODE_SEL_15	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
14 DMA_CH_DB_ MODE_SEL_14	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
13 DMA_CH_DB_ MODE_SEL_13	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
12 DMA_CH_DB_ MODE_SEL_12	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_CH_DB_MODE_SEL0 field descriptions (continued)

Field	Description
	0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
11 DMA_CH_DB_ MODE_SEL_11	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
10 DMA_CH_DB_ MODE_SEL_10	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
9 DMA_CH_DB_ MODE_SEL_9	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
8 DMA_CH_DB_ MODE_SEL_8	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
5 DMA_CH_DB_ MODE_SEL_5	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
3 DMA_CH_DB_ MODE_SEL_3	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
2 DMA_CH_DB_ MODE_SEL_2	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

Table continues on the next page...

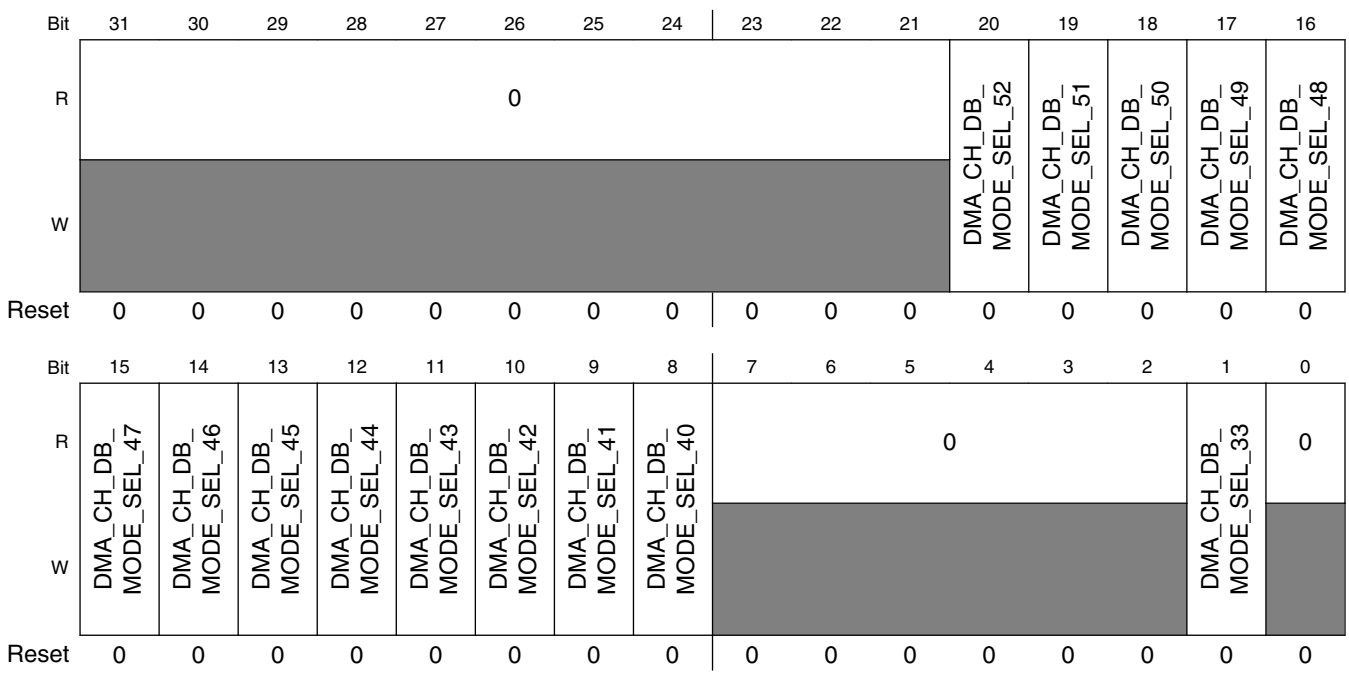
IPUx_CH_DB_MODE_SEL0 field descriptions (continued)

Field	Description
1 DMA_CH_DB_ MODE_SEL_1	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
0 DMA_CH_DB_ MODE_SEL_0	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

37.5.48 Channel Double Buffer Mode Select 1 Register (IPUx_CH_DB_MODE_SEL1)

The register contains double buffer mode select control information for 32 IPU's DMA channels.

Address: Base address + 154h offset



IPUx_CH_DB_MODE_SEL1 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_CH_DB_MODE_SEL1 field descriptions (continued)

Field	Description
	0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
20 DMA_CH_DB_ MODE_SEL_52	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
19 DMA_CH_DB_ MODE_SEL_51	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
18 DMA_CH_DB_ MODE_SEL_50	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
17 DMA_CH_DB_ MODE_SEL_49	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
16 DMA_CH_DB_ MODE_SEL_48	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
15 DMA_CH_DB_ MODE_SEL_47	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
14 DMA_CH_DB_ MODE_SEL_46	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
13 DMA_CH_DB_ MODE_SEL_45	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
12 DMA_CH_DB_ MODE_SEL_44	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

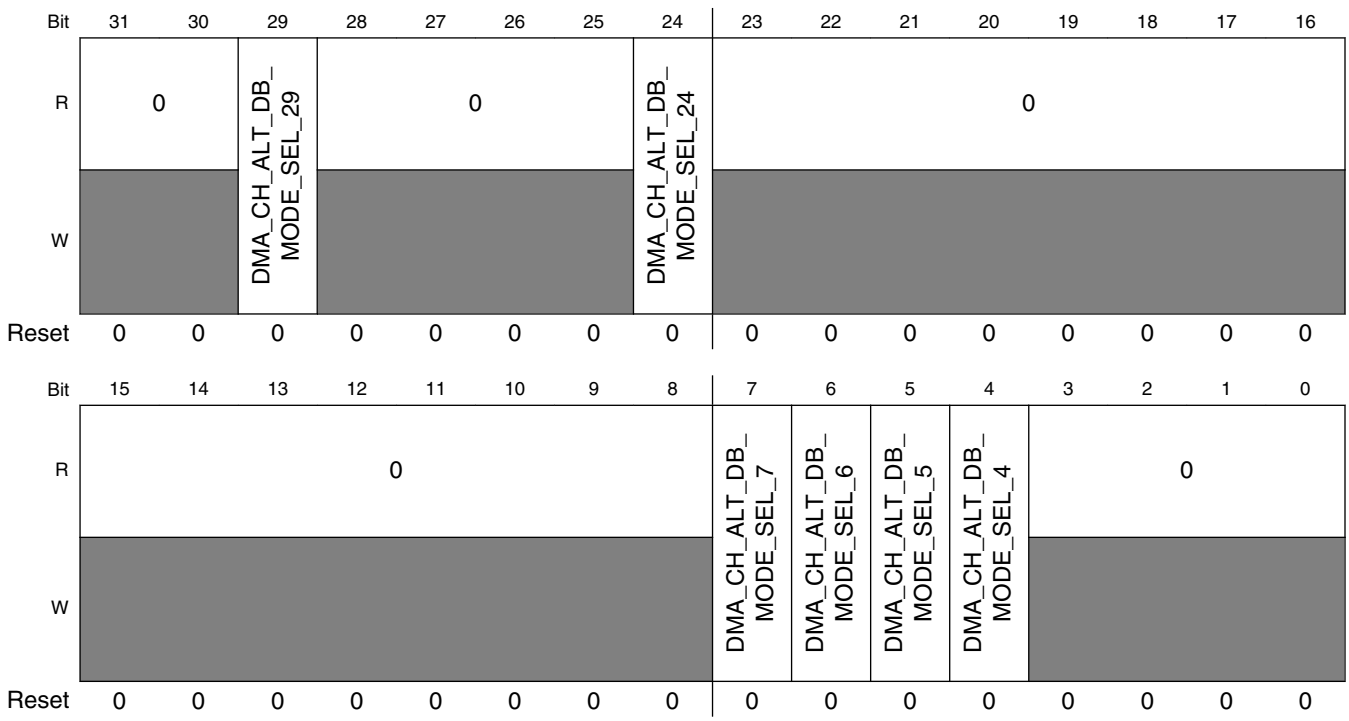
IPUx_CH_DB_MODE_SEL1 field descriptions (continued)

Field	Description
	0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
11 DMA_CH_DB_ MODE_SEL_43	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
10 DMA_CH_DB_ MODE_SEL_42	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
9 DMA_CH_DB_ MODE_SEL_41	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
8 DMA_CH_DB_ MODE_SEL_40	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
1 DMA_CH_DB_ MODE_SEL_33	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

37.5.49 Alternate Channel Double Buffer Mode Select 0 Register (IPUx_ALT_CH_DB_MODE_SEL0)

The register contains double buffer mode select control information for 32 IPU's DMA channels.

Address: Base address + 168h offset



IPUx_ALT_CH_DB_MODE_SEL0 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
29 DMA_CH_ALT_DB_MODE_SEL_29	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
28–25 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

Table continues on the next page...

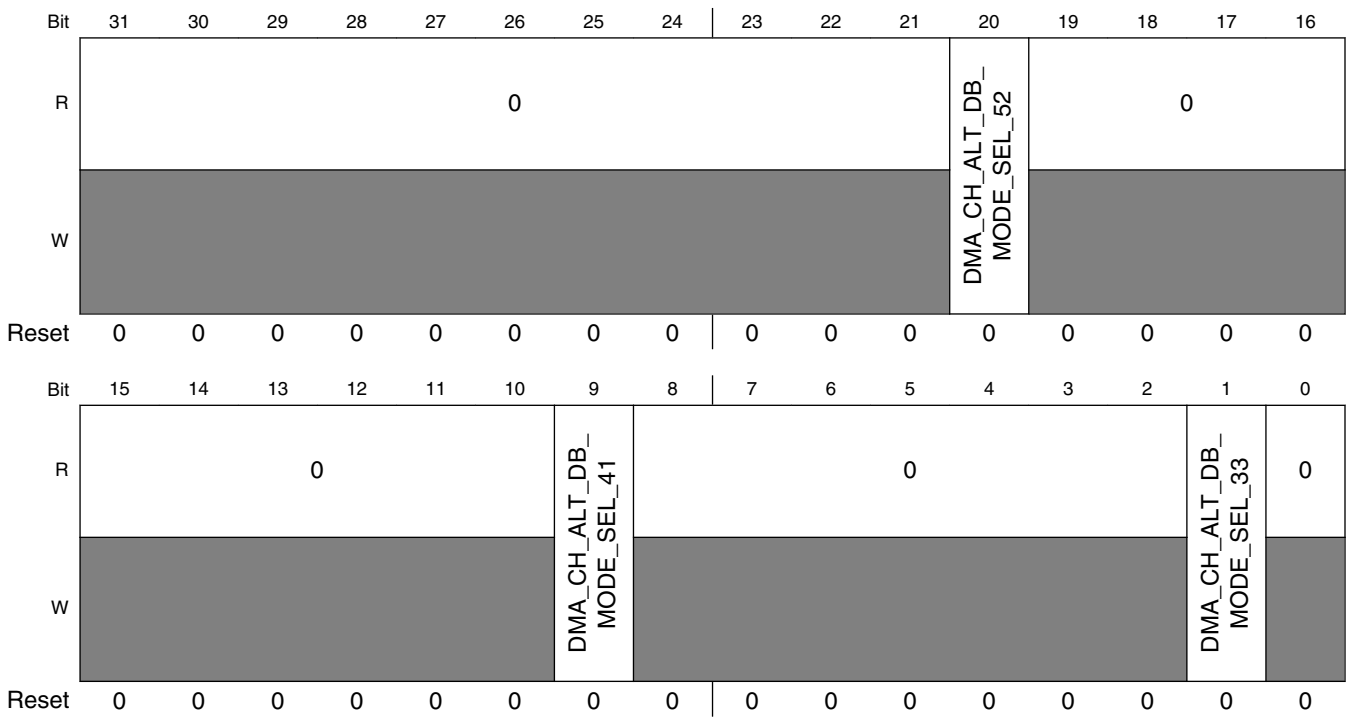
IPUx_ALT_CH_DB_MODE_SEL0 field descriptions (continued)

Field	Description
24 DMA_CH_ALT_DB_MODE_SEL_24	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
23–8 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
7 DMA_CH_ALT_DB_MODE_SEL_7	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
6 DMA_CH_ALT_DB_MODE_SEL_6	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
5 DMA_CH_ALT_DB_MODE_SEL_5	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
4 DMA_CH_ALT_DB_MODE_SEL_4	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

37.5.50 Alternate Channel Double Buffer Mode Select1 Register (IPUx_ALT_CH_DB_MODE_SEL1)

The register contains double buffer mode select control information for 32 IPU's DMA channels.

Address: Base address + 16Ch offset



IPUx_ALT_CH_DB_MODE_SEL1 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
20 DMA_CH_ALT_DB_MODE_SEL_52	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
19–10 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

Table continues on the next page...

IPUx_ALT_CH_DB_MODE_SEL1 field descriptions (continued)

Field	Description
9 DMA_CH_ALT_DB_MODE_SEL_41	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
8–2 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
1 DMA_CH_ALT_DB_MODE_SEL_33	Double Buffer Mode Select. This bit indicates if a double buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Double buffer is not used for this channel. 1 Double buffer is used for this channel.

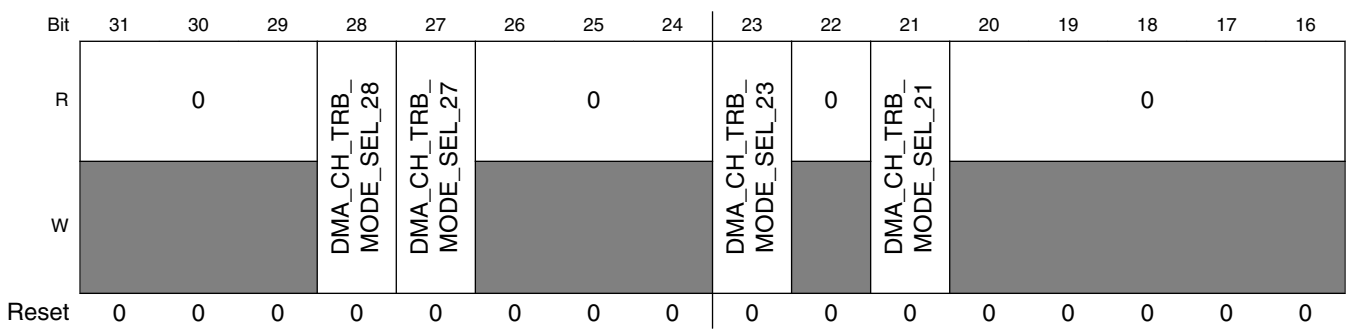
37.5.51 Alternate Channel Triple Buffer Mode Select 0 Register (IPUx_ALT_CH_TRB_MODE_SEL0)

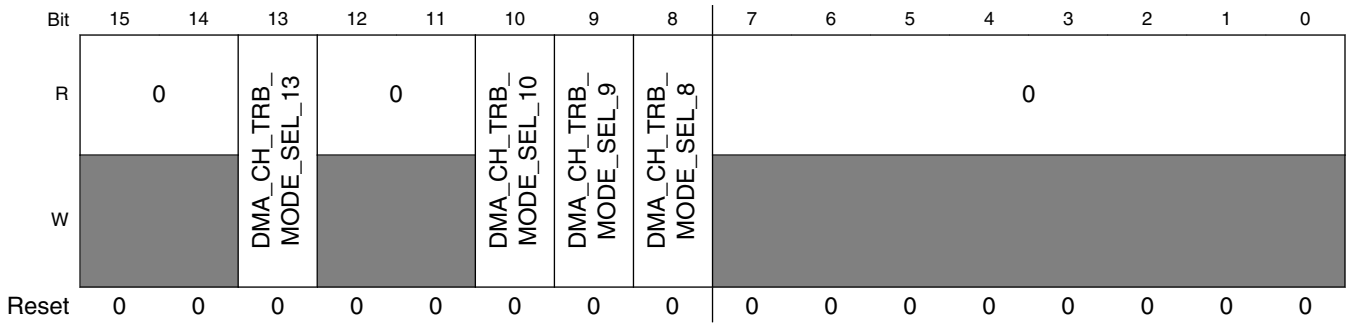
The register contains triple buffer mode select control information for 32 IPU's DMA channels.

When the channel is configured for triple buffer mode. The double buffer mode settings configured on the corresponding DB_MODE_SEL bit are overridden.

- Hide VPU_SUB_FRAME_SYNC for all versions
- Show VPU_SUB_FRAME_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.
- This requires some sophisticated conditional tag settings

Address: Base address + 178h offset





IPUx_ALT_CH_TRB_MODE_SEL0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
28 DMA_CH_TRB_MODE_SEL_28	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
27 DMA_CH_TRB_MODE_SEL_27	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
26–24 Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
23 DMA_CH_TRB_MODE_SEL_23	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
22 Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
21 DMA_CH_TRB_MODE_SEL_21	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
20–14 Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.

Table continues on the next page...

IPUx_ALT_CH_TRB_MODE_SEL0 field descriptions (continued)

Field	Description
13 DMA_CH_TRB_ MODE_SEL_13	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
12–11 Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
10 DMA_CH_TRB_ MODE_SEL_10	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
9 DMA_CH_TRB_ MODE_SEL_9	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
8 DMA_CH_TRB_ MODE_SEL_8	Triple Buffer Mode Select. This bit indicates if a triple buffer is used for this channel. *n* Indicates the corresponding DMA channel number. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.
Reserved	This read-only field is reserved and always has the value 0. 0 Triple buffer is not used for this channel. 1 Triple buffer is used for this channel.

37.5.52 Interrupt Status Register 1 (IPUx_INT_STAT_1)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of EOF (end of frame) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 200h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_EOF_31	0	IDMAC_EOF_29	IDMAC_EOF_28	IDMAC_EOF_27	IDMAC_EOF_26	IDMAC_EOF_25	IDMAC_EOF_24	IDMAC_EOF_23	IDMAC_EOF_22	IDMAC_EOF_21	IDMAC_EOF_20	IDMAC_EOF_19	IDMAC_EOF_18	IDMAC_EOF_17	0
W	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_EOF_15	IDMAC_EOF_14	IDMAC_EOF_13	IDMAC_EOF_12	IDMAC_EOF_11	IDMAC_EOF_10	IDMAC_EOF_9	IDMAC_EOF_8	0		IDMAC_EOF_5	0	IDMAC_EOF_3	IDMAC_EOF_2	IDMAC_EOF_1	IDMAC_EOF_0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c			w1c		w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_1 field descriptions

Field	Description
31 IDMAC_EOF_31	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_1 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
29 IDMAC_EOF_29	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
28 IDMAC_EOF_28	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IDMAC_EOF_27	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_EOF_26	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_EOF_25	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24 IDMAC_EOF_24	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
23 IDMAC_EOF_23	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
22 IDMAC_EOF_22	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_1 field descriptions (continued)

Field	Description
21 IDMAC_EOF_21	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_EOF_20	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_EOF_19	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_EOF_18	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_EOF_17	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_EOF_15	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_EOF_14	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_EOF_13	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_EOF_12	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_1 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_EOF_11	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_EOF_10	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_EOF_9	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_EOF_8	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
5 IDMAC_EOF_5	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
3 IDMAC_EOF_3	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
2 IDMAC_EOF_2	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_EOF_1	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n.

Table continues on the next page...

IPUx_INT_STAT_1 field descriptions (continued)

Field	Description
	n Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 IDMAC_EOF_0	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.53 Interrupt Status Register2 (IPUx_INT_STAT_2)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of EOF (end of frame) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 204h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_EOF_52	IDMAC_EOF_51	IDMAC_EOF_50	IDMAC_EOF_49	IDMAC_EOF_48
W												w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_EOF_47	IDMAC_EOF_46	IDMAC_EOF_45	IDMAC_EOF_44	IDMAC_EOF_43	IDMAC_EOF_42	IDMAC_EOF_41	IDMAC_EOF_40	0						IDMAC_EOF_33	0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c							w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_2 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.
	0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_2 field descriptions (continued)

Field	Description
20 IDMAC_EOF_52	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_EOF_51	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_EOF_50	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_EOF_49	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 IDMAC_EOF_48	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_EOF_47	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_EOF_46	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_EOF_45	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_EOF_44	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_2 field descriptions (continued)

Field	Description
11 IDMAC_EOF_43	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_EOF_42	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_EOF_41	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_EOF_40	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_EOF_33	End of Frame of Channel interrupt. This bit is the status bit of End Of Frame of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.54 Interrupt Status Register 3 (IPUx_INT_STAT_3)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of NFACK (New Frame Ack) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 208h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_NFACK_31	0	IDMAC_NFACK_29	IDMAC_NFACK_28	IDMAC_NFACK_27	IDMAC_NFACK_26	IDMAC_NFACK_25	IDMAC_NFACK_24	IDMAC_NFACK_23	IDMAC_NFACK_22	IDMAC_NFACK_21	IDMAC_NFACK_20	IDMAC_NFACK_19	IDMAC_NFACK_18	IDMAC_NFACK_17	0
W	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFACK_15	IDMAC_NFACK_14	IDMAC_NFACK_13	IDMAC_NFACK_12	IDMAC_NFACK_11	IDMAC_NFACK_10	IDMAC_NFACK_9	IDMAC_NFACK_8	0		IDMAC_NFACK_5	0	IDMAC_NFACK_3	IDMAC_NFACK_2	IDMAC_NFACK_1	IDMAC_NFACK_0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c			w1c		w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_3 field descriptions

Field	Description
31 IDMAC_NFACK_31	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
29 IDMAC_NFACK_29	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_3 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
28 IDMAC_NFACK_28	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IDMAC_NFACK_27	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_NFACK_26	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_NFACK_25	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24 IDMAC_NFACK_24	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
23 IDMAC_NFACK_23	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
22 IDMAC_NFACK_22	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
21 IDMAC_NFACK_21	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_NFACK_20	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_3 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_NFACK_19	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_NFACK_18	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_NFACK_17	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_NFACK_15	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_NFACK_14	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_NFACK_13	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_NFACK_12	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_NFACK_11	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_3 field descriptions (continued)

Field	Description
10 IDMAC_NFACK_10	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_NFACK_9	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_NFACK_8	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
5 IDMAC_NFACK_5	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
3 IDMAC_NFACK_3	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
2 IDMAC_NFACK_2	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_NFACK_1	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 IDMAC_NFACK_0	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_3 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.

37.5.55 Interrupt Status Register 4 (IPUx_INT_STAT_4)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of NFACK (New Frame Ack) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 20Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_NFACK_52	IDMAC_NFACK_51	IDMAC_NFACK_50	IDMAC_NFACK_49	IDMAC_NFACK_48
W												w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFACK_47	IDMAC_NFACK_46	IDMAC_NFACK_45	IDMAC_NFACK_44	IDMAC_NFACK_43	IDMAC_NFACK_42	IDMAC_NFACK_41	IDMAC_NFACK_40	0						IDMAC_NFACK_33	0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c							w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_4 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_NFACK_52	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_4 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_NFACK_51	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_NFACK_50	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_NFACK_49	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 IDMAC_NFACK_48	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_NFACK_47	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_NFACK_46	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_NFACK_45	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_NFACK_44	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_NFACK_43	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_4 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_NFACK_42	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_NFACK_41	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_NFACK_40	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_NFACK_33	Enable New Frame Ack of Channel interrupt. This bit is the status bits of New Frame Ack of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.56 Interrupt Status Register 5 (IPUx_INT_STAT_5)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the New-frame before end-of-frame indication (NFB4EOF_ERR) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 210h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_NFB4EOF_ERR_31	0	IDMAC_NFB4EOF_ERR_29	IDMAC_NFB4EOF_ERR_28	IDMAC_NFB4EOF_ERR_27	IDMAC_NFB4EOF_ERR_26	IDMAC_NFB4EOF_ERR_25	IDMAC_NFB4EOF_ERR_24	IDMAC_NFB4EOF_ERR_23	IDMAC_NFB4EOF_ERR_22	IDMAC_NFB4EOF_ERR_21	IDMAC_NFB4EOF_ERR_20	IDMAC_NFB4EOF_ERR_19	IDMAC_NFB4EOF_ERR_18	IDMAC_NFB4EOF_ERR_17	0
W	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFB4EOF_ERR_15	IDMAC_NFB4EOF_ERR_14	IDMAC_NFB4EOF_ERR_13	IDMAC_NFB4EOF_ERR_12	IDMAC_NFB4EOF_ERR_11	IDMAC_NFB4EOF_ERR_10	IDMAC_NFB4EOF_ERR_9	IDMAC_NFB4EOF_ERR_8	0		IDMAC_NFB4EOF_ERR_5	0	IDMAC_NFB4EOF_ERR_3	IDMAC_NFB4EOF_ERR_2	IDMAC_NFB4EOF_ERR_1	IDMAC_NFB4EOF_ERR_0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c			w1c		w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_5 field descriptions

Field	Description
31 IDMAC_ NFB4EOF_ERR_ 31	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
29 IDMAC_ NFB4EOF_ERR_ 29	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
28 IDMAC_ NFB4EOF_ERR_ 28	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IDMAC_ NFB4EOF_ERR_ 27	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_ NFB4EOF_ERR_ 26	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_ NFB4EOF_ERR_ 25	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24 IDMAC_ NFB4EOF_ERR_ 24	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_5 field descriptions (continued)

Field	Description
23 IDMAC_ NFB4EOF_ERR_ 23	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
22 IDMAC_ NFB4EOF_ERR_ 22	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
21 IDMAC_ NFB4EOF_ERR_ 21	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_ NFB4EOF_ERR_ 20	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_ NFB4EOF_ERR_ 19	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_ NFB4EOF_ERR_ 18	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_ NFB4EOF_ERR_ 17	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_5 field descriptions (continued)

Field	Description
15 IDMAC_ NFB4EOF_ERR_ 15	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_ NFB4EOF_ERR_ 14	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_ NFB4EOF_ERR_ 13	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_ NFB4EOF_ERR_ 12	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_ NFB4EOF_ERR_ 11	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_ NFB4EOF_ERR_ 10	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_ NFB4EOF_ERR_ 9	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_ NFB4EOF_ERR_ 8	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_5 field descriptions (continued)

Field	Description
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
5 IDMAC_ NFB4EOF_ERR_ 5	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
3 IDMAC_ NFB4EOF_ERR_ 3	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
2 IDMAC_ NFB4EOF_ERR_ 2	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_ NFB4EOF_ERR_ 1	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 IDMAC_ NFB4EOF_ERR_ 0	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.57 Interrupt Status Register 6 (IPUx_INT_STAT_6)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the New-frame before end-of-frame indication (NFB4EOF_ERR) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 214h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_NFB4EOF_ERR_52	IDMAC_NFB4EOF_ERR_51	IDMAC_NFB4EOF_ERR_50	IDMAC_NFB4EOF_ERR_49	IDMAC_NFB4EOF_ERR_48
W												w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_NFB4EOF_ERR_47	IDMAC_NFB4EOF_ERR_46	IDMAC_NFB4EOF_ERR_45	IDMAC_NFB4EOF_ERR_44	IDMAC_NFB4EOF_ERR_43	IDMAC_NFB4EOF_ERR_42	IDMAC_NFB4EOF_ERR_41	IDMAC_NFB4EOF_ERR_40	0						IDMAC_NFB4EOF_ERR_33	0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c							w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_6 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_ NFB4EOF_ERR_ 52	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_ NFB4EOF_ERR_ 51	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_ NFB4EOF_ERR_ 50	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_ NFB4EOF_ERR_ 49	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 IDMAC_ NFB4EOF_ERR_ 48	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_ NFB4EOF_ERR_ 47	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_ NFB4EOF_ERR_ 46	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_6 field descriptions (continued)

Field	Description
13 IDMAC_ NFB4EOF_ERR_ 45	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_ NFB4EOF_ERR_ 44	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_ NFB4EOF_ERR_ 43	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_ NFB4EOF_ERR_ 42	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_ NFB4EOF_ERR_ 41	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_ NFB4EOF_ERR_ 40	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_ NFB4EOF_ERR_ 33	New Frame before end-of-frame error indication of Channel interrupt. This bit is the status bit of New Frame before end-of-frame error interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_INT_STAT_6 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.

37.5.58 Interrupt Status Register7 1 (IPUx_INT_STAT_7)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the End-of-Scroll indication (EOS) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 218h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_EOS_EN_31	0	IDMAC_EOS_EN_29	IDMAC_EOS_EN_28	IDMAC_EOS_EN_27	IDMAC_EOS_EN_26	IDMAC_EOS_EN_25	IDMAC_EOS_EN_24	IDMAC_EOS_EN_23	0			IDMAC_EOS_EN_19	0		
W	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c				w1c			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_7 field descriptions

Field	Description
31 IDMAC_EOS_EN_31	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
29 IDMAC_EOS_EN_29	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
28 IDMAC_EOS_EN_28	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IDMAC_EOS_EN_27	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_EOS_EN_26	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_EOS_EN_25	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24 IDMAC_EOS_EN_24	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_7 field descriptions (continued)

Field	Description
23 IDMAC_EOS_EN_23	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
22–20 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_EOS_EN_19	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.59 Interrupt Status Register 8 (IPUx_INT_STAT_8)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. All the status bits of the End of Scroll indication (EOS) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 21Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_EOS_EN_52	IDMAC_EOS_EN_51	0		
W												w1c	w1c			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			IDMAC_EOS_EN_44	IDMAC_EOS_EN_43	IDMAC_EOS_EN_42	IDMAC_EOS_EN_41	0							IDMAC_EOS_EN_33	0
W				w1c	w1c	w1c	w1c								w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_8 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_INT_STAT_8 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_EOS_EN_52	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_EOS_EN_51	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18–13 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_EOS_EN_44	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_EOS_EN_43	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_EOS_EN_42	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_EOS_EN_41	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_8 field descriptions (continued)

Field	Description
1 IDMAC_EOS_EN_33	End of Scroll indication of Channel interrupt. This bit is the status bit of End of Scroll interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.60 Interrupt Status Register 9 (IPUx_INT_STAT_9)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. This register holds the error interrupt indications coming from different modules within All the bits in this register are write one to clear.

Address: Base address + 220h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	CSI1_PUPE	CSI0_PUPE	0	IC_VF_BUF_OVF	IC_ENC_BUF_OVF	IC_BAYER_BUF_OVF	0									
W	w1c	w1c		w1c	w1c	w1c										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0															VDI_FIFO1_OVF
W																w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_9 field descriptions

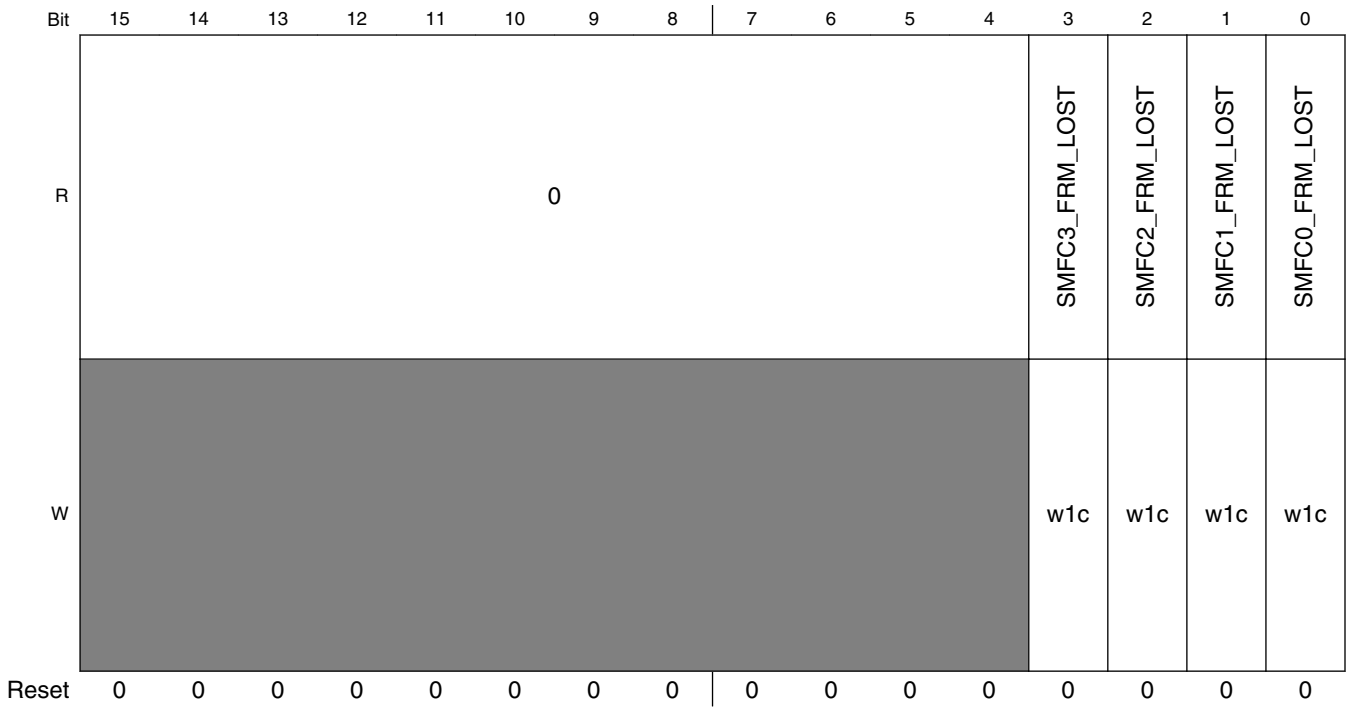
Field	Description
31 CSI1_PUPE	CSI1_PUPE - CSI1 parameters update error interrupt. This bit indicates on an interrupt that is a result of an error generated by the CSI1. The error is generated in case where new frame arrived from the CSI1 before the completion of the CSI1's parameters update by the SRM 0 Interrupt is cleared. 1 Interrupt is requested.
30 CSI0_PUPE	CSI0_PUPE - CSI0 parameters update error interrupt. This bit indicates on an interrupt that is a result of an error generated by the CSI0. The error is generated in case where new frame arrived from the CSI0 before the completion of the CSI0's parameters update by the SRM 0 Interrupt is cleared. 1 Interrupt is requested.
29 Reserved	This read-only field is reserved and always has the value 0.
28 IC_VF_BUF_OVF	This bit indicates on an interrupt that is a result of the IC Buffer overflow for view finder coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IC_ENC_BUF_OVF	This bit indicates on an interrupt that is a result of the IC Buffer overflow for encoding coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IC_BAYER_BUF_OVF	This bit indicates on an interrupt that is a result of the IC Buffer overflow for Bayer coming from the IC. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is cleared. 1 Interrupt is requested.
25–1 Reserved	This read-only field is reserved and always has the value 0.
0 VDI_FIFO1_OVF	FIFO1 overflow Interrupt1 The VDIC generate FIFO1 overflow interrupt1 when write pointer of FIFO1 overrun read pointer. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.61 Interrupt Status Register 10 (IPUx_INT_STAT_10)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. This register holds error interrupt indications coming from different modules within All the bits in this register are write one to clear.

Address: Base address + 224h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	AXIR_ERR	AXIW_ERR	NON_PRIVILEGED_ACC_ERR	0	IC_BAYER_FRM_LOST_ERR	IC_ENC_FRM_LOST_ERR	IC_VF_FRM_LOST_ERR	0	DI1_TIME_OUT_ERR	DI0_TIME_OUT_ERR	DI1_SYNC_DISP_ERR	DI0_SYNC_DISP_ERR	DC_TEARING_ERR_6	DC_TEARING_ERR_2	DC_TEARING_ERR_1
W		w1c	w1c	w1c		w1c	w1c	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_INT_STAT_10 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30 AXIR_ERR	This bit indicates on an interrupt that is a result of AXI read access resulted with error response. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is cleared. 1 Interrupt is requested.
29 AXIW_ERR	This bit indicates on an interrupt that is a result of AXI write access resulted with error response. The user needs to write 1 to this bit in order to clear it. 0 Interrupt is cleared. 1 Interrupt is requested.
28 NON_ PRIVILEGED_ ACC_ERR	Non Privileged Access Error interrupt. This bit indicates on an interrupt that is a result of access the CPMEM or the DP memory in user mode 0 Interrupt is cleared. 1 Interrupt is requested.
27 Reserved	This read-only field is reserved and always has the value 0.
26 IC_BAYER_ FRM_LOST_ ERR	This bit indicates on an interrupt that is a result of IC's Bayer frame lost. 0 Interrupt is disabled. 1 Interrupt is enabled.
25 IC_ENC_FRM_ LOST_ERR	This bit indicates on an interrupt that is a result of IC's encoding frame lost.

Table continues on the next page...

IPUx_INT_STAT_10 field descriptions (continued)

Field	Description
	0 Interrupt is disabled. 1 Interrupt is enabled.
24 IC_VF_FRM_LOST_ERR	This bit indicates on an interrupt that is a result of IC's view finder frame lost. 0 Interrupt is disabled. 1 Interrupt is enabled.
23 Reserved	This read-only field is reserved and always has the value 0.
22 DI1_TIME_OUT_ERR	DI1 time out error interrupt This bit indicates on the interrupt that is a result of a time out error during a read access via DI1
21 DI0_TIME_OUT_ERR	DI0 time out error interrupt This bit indicates on the interrupt that is a result of a time out error during a read access via DI0
20 DI1_SYNC_DISP_ERR	DI1 Synchronous display error interrupt This bit indicates on the interrupt that is a result of an error during access to a synchronous display via DI1
19 DI0_SYNC_DISP_ERR	DI0 Synchronous display error interrupt This bit indicates on the interrupt that is a result of an error during access to a synchronous display via DI0
18 DC_TEARING_ERR_6	Tearing Error #6 interrupt This bit indicates on the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 6
17 DC_TEARING_ERR_2	Tearing Error #2 interrupt This bit indicates on the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 2
16 DC_TEARING_ERR_1	Tearing Error #1 interrupt This bit indicates on the interrupt that is a result of tearing error while the anti tearing mechanism is activated for DC channel 1
15–4 Reserved	This read-only field is reserved and always has the value 0.
3 SMFC3_FRM_LOST	Frame Lost of SMFC channel 3 interrupt. This bit indicates on an interrupt that is a result of a result of a Frame Lost of SMFC channel 3 0 Interrupt is cleared. 1 Interrupt is requested.
2 SMFC2_FRM_LOST	Frame Lost of SMFC channel 2 interrupt. This bit indicates on an interrupt that is a result of a result of a Frame Lost of SMFC channel 2 0 Interrupt is cleared. 1 Interrupt is requested.
1 SMFC1_FRM_LOST	Frame Lost of SMFC channel 1 interrupt. This bit indicates on an interrupt that is a result of a result of a Frame Lost of SMFC channel 1 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_10 field descriptions (continued)

Field	Description
0 SMFC0_FRM_LOST	Frame Lost of SMFC channel 0 interrupt. This bit indicates on an interrupt that is a result of a result of a Frame Lost of SMFC channel 0 0 Interrupt is cleared. 1 Interrupt is requested.

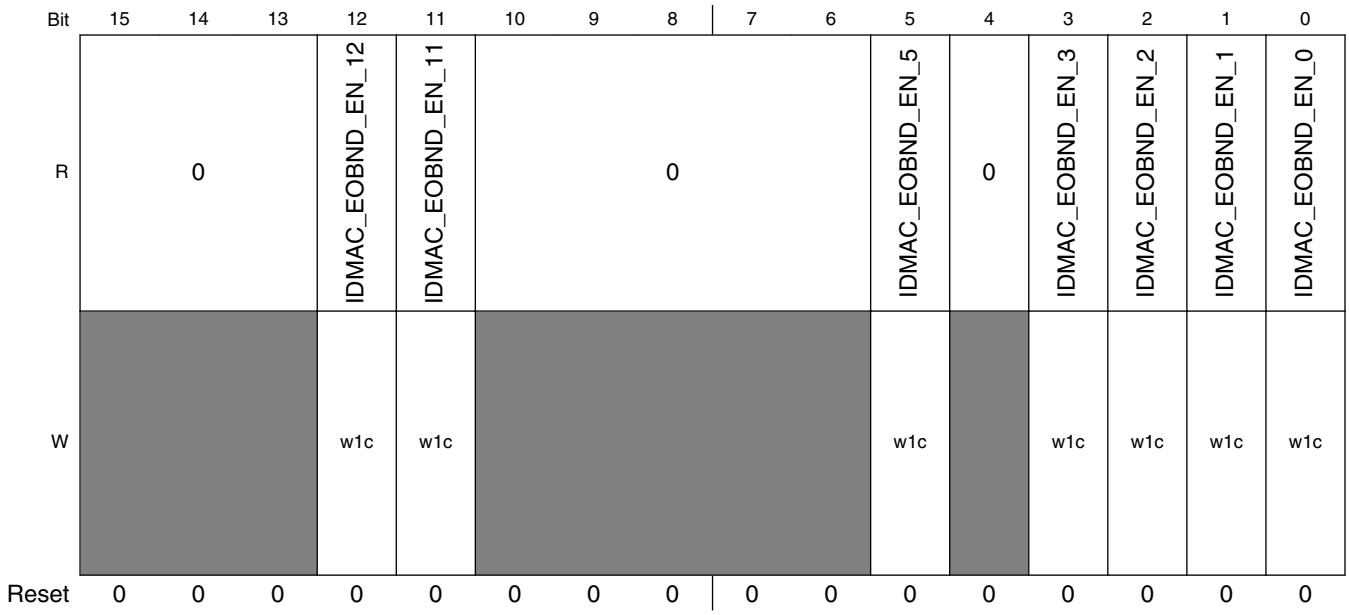
37.5.62 Interrupt Status Register 11 (IPUx_INT_STAT_11)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the end-of-band indication (EOBND) of DMA Channels interrupts [31:0] can be found in this register.

- Hide VDOA_SYNC for all versions
- Show VDOA_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.

Address: Base address + 228h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0					IDMAC_EOBND_EN_26	IDMAC_EOBND_EN_25	0		IDMAC_EOBND_EN_22	IDMAC_EOBND_EN_21	IDMAC_EOBND_EN_20	0			
W						w1c	w1c			w1c	w1c	w1c				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			IDMAC_EOBND_EN_12	IDMAC_EOBND_EN_11	0					IDMAC_EOBND_EN_5	0	IDMAC_EOBND_EN_3	IDMAC_EOBND_EN_2	IDMAC_EOBND_EN_1	IDMAC_EOBND_EN_0
W				w1c	w1c						w1c		w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_11 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_EOBND_EN_26	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_EOBND_EN_25	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24–23 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
22 IDMAC_EOBND_EN_22	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
21 IDMAC_EOBND_EN_21	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n.

Table continues on the next page...

IPUx_INT_STAT_11 field descriptions (continued)

Field	Description
	n Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_EOBND_EN_20	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19–13 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_EOBND_EN_12	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_EOBND_EN_11	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
5 IDMAC_EOBND_EN_5	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
3 IDMAC_EOBND_EN_3	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_11 field descriptions (continued)

Field	Description
2 IDMAC_EOBND_EN_2	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_EOBND_EN_1	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 IDMAC_EOBND_EN_0	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.63 Interrupt Status Register 12 (IPUx_INT_STAT_12)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the end-of-band indication (EOBND) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 22Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0													IDMAC_EOBND_EN_50	IDMAC_EOBND_EN_49	IDMAC_EOBND_EN_48
W														w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_EOBND_EN_47	IDMAC_EOBND_EN_46	IDMAC_EOBND_EN_45	0												
W	w1c	w1c	w1c													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_12 field descriptions

Field	Description
31–19 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_EOBND_EN_50	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_EOBND_EN_49	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 IDMAC_EOBND_EN_48	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_EOBND_EN_47	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_EOBND_EN_46	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_EOBND_EN_45	end-of-band indication of Channel interrupt. This bit is the status bit of end-of-band interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.64 Interrupt Status Register 13 (IPUx_INT_STAT_13)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the Threshold crossing indication (TH) of DMA Channels interrupts [31:0] can be found in this register.

Address: Base address + 230h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_TH_31	0	IDMAC_TH_29	IDMAC_TH_28	IDMAC_TH_27	IDMAC_TH_26	IDMAC_TH_25	IDMAC_TH_24	IDMAC_TH_23	IDMAC_TH_22	IDMAC_TH_21	IDMAC_TH_20	IDMAC_TH_19	IDMAC_TH_18	IDMAC_TH_17	0
W	w1c		w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_TH_15	IDMAC_TH_14	IDMAC_TH_13	IDMAC_TH_12	IDMAC_TH_11	IDMAC_TH_10	IDMAC_TH_9	IDMAC_TH_8	0		IDMAC_TH_5	0	IDMAC_TH_3	IDMAC_TH_2	IDMAC_TH_1	IDMAC_TH_0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c			w1c		w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_13 field descriptions

Field	Description
31 IDMAC_TH_31	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_13 field descriptions (continued)

Field	Description
30 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
29 IDMAC_TH_29	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
28 IDMAC_TH_28	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
27 IDMAC_TH_27	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
26 IDMAC_TH_26	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
25 IDMAC_TH_25	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
24 IDMAC_TH_24	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
23 IDMAC_TH_23	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
22 IDMAC_TH_22	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n.

Table continues on the next page...

IPUx_INT_STAT_13 field descriptions (continued)

Field	Description
	n Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
21 IDMAC_TH_21	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_TH_20	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_TH_19	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_TH_18	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_TH_17	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_TH_15	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_TH_14	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_13 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_TH_13	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_TH_12	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_TH_11	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_TH_10	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_TH_9	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_TH_8	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
5 IDMAC_TH_5	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_13 field descriptions (continued)

Field	Description
4 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
3 IDMAC_TH_3	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
2 IDMAC_TH_2	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_TH_1	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 IDMAC_TH_0	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.65 Interrupt Status Register 14 (IPUx_INT_STAT_14)

IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of the Threshold crossing indication (TH) of DMA Channels interrupts [63:32] can be found in this register.

Address: Base address + 234h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											IDMAC_TH_52	IDMAC_TH_51	IDMAC_TH_50	IDMAC_TH_49	IDMAC_TH_48
W												w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	IDMAC_TH_47	IDMAC_TH_46	IDMAC_TH_45	IDMAC_TH_44	IDMAC_TH_43	IDMAC_TH_42	IDMAC_TH_41	IDMAC_TH_40	0						IDMAC_TH_33	0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c							w1c	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_14 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.
	0 Interrupt is cleared. 1 Interrupt is requested.
20 IDMAC_TH_52	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_14 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
19 IDMAC_TH_51	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
18 IDMAC_TH_50	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
17 IDMAC_TH_49	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
16 IDMAC_TH_48	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
15 IDMAC_TH_47	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
14 IDMAC_TH_46	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
13 IDMAC_TH_45	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
12 IDMAC_TH_44	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_INT_STAT_14 field descriptions (continued)

Field	Description
	0 Interrupt is cleared. 1 Interrupt is requested.
11 IDMAC_TH_43	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
10 IDMAC_TH_42	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
9 IDMAC_TH_41	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
8 IDMAC_TH_40	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.
1 IDMAC_TH_33	Threshold crossing indication of Channel interrupt. This bit is the status bit of Threshold crossing interrupt of Channel #n. *n* Indicates the corresponding DMA channel number. 0 Interrupt is cleared. 1 Interrupt is requested.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.66 Interrupt Status Register 15 (IPUx_INT_STAT_15)

IPU status registers are not stored in the SRM during power gating mode.IPU status registers are not stored in the SRM during power gating mode.This register contains part of IPU interrupts status bits. The status bits of general purpose interrupts can be found in this register.

Address: Base address + 238h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	DI1_CNT_EN_PRE_8	DI1_CNT_EN_PRE_3	DI1_DISP_CLK_EN_PRE	DI0_CNT_EN_PRE_10	DI0_CNT_EN_PRE_9	DI0_CNT_EN_PRE_8	DI0_CNT_EN_PRE_7	DI0_CNT_EN_PRE_6	DI0_CNT_EN_PRE_5	DI0_CNT_EN_PRE_4	DI0_CNT_EN_PRE_3	DI0_CNT_EN_PRE_2	DI0_CNT_EN_PRE_1	DI0_CNT_EN_PRE_0	DC_ASYNC_STOP	DC_DP_START
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DI_VSYNC_PRE_1	DI_VSYNC_PRE_0	DC_FC_6	DC_FC_4	DC_FC_3	DC_FC_2	DC_FC_1	DC_FC_0	DP_ASF_BRAKE	DP_SF_BRAKE	DP_ASF_END	DP_ASF_START	DP_SF_END	DP_SF_START	SNOOPING2_INT	SNOOPING1_INT
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_INT_STAT_15 field descriptions

Field	Description
31 DI1_CNT_EN_PRE_8	This bit indicates on the interrupt that is a result of a trigger generated by counter #8 of DI1 0 Interrupt is cleared. 1 Interrupt is requested.
30 DI1_CNT_EN_PRE_3	This bit indicates on the interrupt that is a result of a trigger generated by counter #3 of DI1 0 Interrupt is cleared. 1 Interrupt is requested.
29 DI1_DISP_CLK_EN_PRE	**DI1_DISP_CLK_EN_PRE** 0 Interrupt is cleared. 1 Interrupt is requested.
28 DI0_CNT_EN_PRE_10	This bit indicates on the interrupt that is a result of a trigger generated by counter #10 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
27 DI0_CNT_EN_PRE_9	This bit indicates on the interrupt that is a result of a trigger generated by counter #9 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
26 DI0_CNT_EN_PRE_8	This bit indicates on the interrupt that is a result of a trigger generated by counter #8 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
25 DI0_CNT_EN_PRE_7	This bit indicates on the interrupt that is a result of a trigger generated by counter #7 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
24 DI0_CNT_EN_PRE_6	This bit indicates on the interrupt that is a result of a trigger generated by counter #6 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
23 DI0_CNT_EN_PRE_5	This bit indicates on the interrupt that is a result of a trigger generated by counter #5 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
22 DI0_CNT_EN_PRE_4	This bit indicates on the interrupt that is a result of a trigger generated by counter #4 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
21 DI0_CNT_EN_PRE_3	This bit indicates on the interrupt that is a result of a trigger generated by counter #3 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
20 DI0_CNT_EN_PRE_2	This bit indicates on the interrupt that is a result of a trigger generated by counter #2 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_15 field descriptions (continued)

Field	Description
19 DI0_CNT_EN_PRE_1	This bit indicates on the interrupt that is a result of a trigger generated by counter #1 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
18 DI0_CNT_EN_PRE_0	This bit indicates on the interrupt that is a result of a trigger generated by counter #0 of DI0 0 Interrupt is cleared. 1 Interrupt is requested.
17 DC_ASYNC_STOP	This bit indicates on an interrupt asserted anytime the DP stops an async flow and moves to a sync flow 0 Interrupt is cleared. 1 Interrupt is requested.
16 DC_DP_START	This bit indicates on an interrupt asserted anytime the DP start a new sync or async flow or when an async flow is interrupted by a sync flow 0 Interrupt is cleared. 1 Interrupt is requested.
15 DI_VSYNC_PRE_1	DI1 interrupt indicating of a VSYNC signal asserted 2 rows before the VSYNC sent to the display 0 Interrupt is cleared. 1 Interrupt is requested.
14 DI_VSYNC_PRE_0	DI0 interrupt indicating of a VSYNC signal asserted 2 rows before the VSYNC sent to the display 0 Interrupt is cleared. 1 Interrupt is requested.
13 DC_FC_6	DC Frame Complete on channel #6 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.
12 DC_FC_4	DC Frame Complete on channel #4 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.
11 DC_FC_3	DC Frame Complete on channel #3 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.
10 DC_FC_2	DC Frame Complete on channel #2 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.
9 DC_FC_1	DC Frame Complete on channel #1 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.
8 DC_FC_0	DC Frame Complete on channel #0 interrupt indication 0 Interrupt is cleared. 1 Interrupt is requested.

Table continues on the next page...

IPUx_INT_STAT_15 field descriptions (continued)

Field	Description
7 DP_ASF_BRAKE	DP Async Flow Brake indication interrupt. This bit indicates on the interrupt that is a result of the async flow brake at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
6 DP_SF_BRAKE	DP Sync Flow Brake indication interrupt. This bit indicates on the interrupt that is a result of the Sync flow brake at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
5 DP_ASF_END	DP Async Flow End indication interrupt. This bit indicates on the interrupt that is a result of the Async flow end at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
4 DP_ASF_START	DP Async Flow Start indication interrupt. This bit indicates on the interrupt that is a result of the Async flow start at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
3 DP_SF_END	DP Sync Flow End indication interrupt. This bit indicates on the interrupt that is a result of the Sync flow end at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
2 DP_SF_START	DP Sync Flow Start indication interrupt. This bit indicates on the interrupt that is a result of the Sync flow start at the DP 0 Interrupt is cleared. 1 Interrupt is requested.
1 SNOOPING2_ INT	IPU snooping 2 event indication interrupt. This bit indicates on the interrupt that is a result of the detection of a snooping 2 signal assertion coming to the IPU 0 Interrupt is cleared. 1 Interrupt is requested.
0 SNOOPING1_ INT	IPU snooping 1 event indication interrupt. This bit indicates on the interrupt that is a result of the detection of a snooping 1 signal assertion coming to the 0 Interrupt is cleared. 1 Interrupt is requested.

37.5.67 Current Buffer Register 0 (IPUx_CUR_BUF_0)

This register contains the current buffer status information bit for each DMA channel.

Address: Base address + 23Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	DMA_CH_CUR_BUF_31	0	DMA_CH_CUR_BUF_29	DMA_CH_CUR_BUF_28	DMA_CH_CUR_BUF_27	DMA_CH_CUR_BUF_26	DMA_CH_CUR_BUF_25	DMA_CH_CUR_BUF_24	DMA_CH_CUR_BUF_23	DMA_CH_CUR_BUF_22	DMA_CH_CUR_BUF_21	DMA_CH_CUR_BUF_20	DMA_CH_CUR_BUF_19	DMA_CH_CUR_BUF_18	DMA_CH_CUR_BUF_17	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DMA_CH_CUR_BUF_15	DMA_CH_CUR_BUF_14	DMA_CH_CUR_BUF_13	DMA_CH_CUR_BUF_12	DMA_CH_CUR_BUF_11	DMA_CH_CUR_BUF_10	DMA_CH_CUR_BUF_9	DMA_CH_CUR_BUF_8	0		DMA_CH_CUR_BUF_5	0	DMA_CH_CUR_BUF_3	DMA_CH_CUR_BUF_2	DMA_CH_CUR_BUF_1	DMA_CH_CUR_BUF_0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CUR_BUF_0 field descriptions

Field	Description
31 DMA_CH_CUR_BUF_31	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
30 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
29 DMA_CH_CUR_BUF_29	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
28 DMA_CH_CUR_BUF_28	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

Table continues on the next page...

IPUx_CUR_BUF_0 field descriptions (continued)

Field	Description
27 DMA_CH_CUR_BUF_27	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
26 DMA_CH_CUR_BUF_26	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
25 DMA_CH_CUR_BUF_25	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
24 DMA_CH_CUR_BUF_24	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
23 DMA_CH_CUR_BUF_23	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
22 DMA_CH_CUR_BUF_22	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
21 DMA_CH_CUR_BUF_21	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
20 DMA_CH_CUR_BUF_20	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
19 DMA_CH_CUR_BUF_19	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

Table continues on the next page...

IPUx_CUR_BUF_0 field descriptions (continued)

Field	Description
18 DMA_CH_CUR_BUF_18	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
17 DMA_CH_CUR_BUF_17	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
16 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
15 DMA_CH_CUR_BUF_15	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
14 DMA_CH_CUR_BUF_14	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
13 DMA_CH_CUR_BUF_13	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
12 DMA_CH_CUR_BUF_12	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
11 DMA_CH_CUR_BUF_11	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
10 DMA_CH_CUR_BUF_10	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
9 DMA_CH_CUR_BUF_9	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

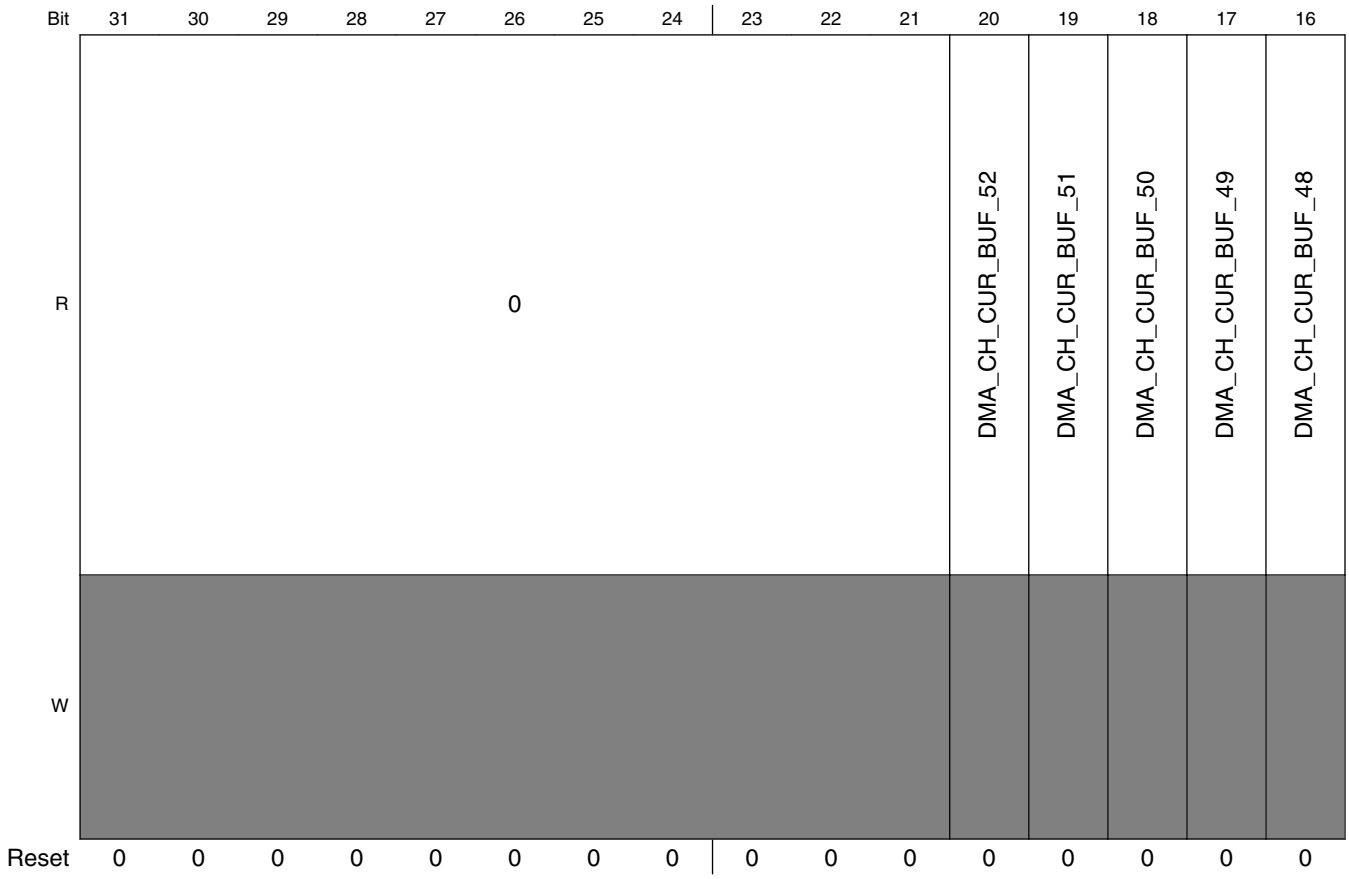
IPUx_CUR_BUF_0 field descriptions (continued)

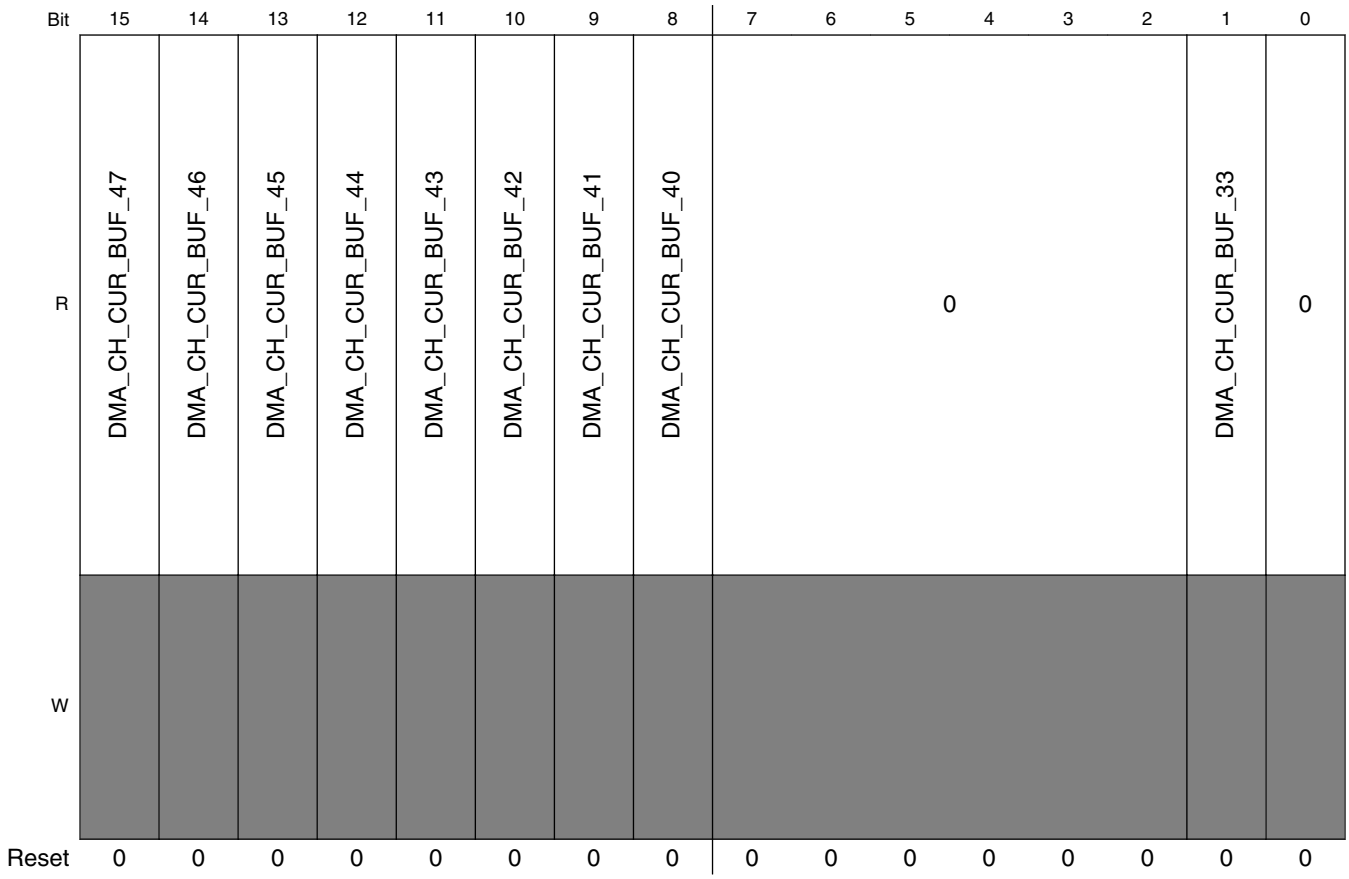
Field	Description
	0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
8 DMA_CH_CUR_BUF_8	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
7–6 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
5 DMA_CH_CUR_BUF_5	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
4 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
3 DMA_CH_CUR_BUF_3	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
2 DMA_CH_CUR_BUF_2	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
1 DMA_CH_CUR_BUF_1	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
0 DMA_CH_CUR_BUF_0	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

37.5.68 Current Buffer Register 1 (IPUx_CUR_BUF_1)

This register contains the current buffer status information bit for each DMA channel.

Address: Base address + 240h offset





IPUx_CUR_BUF_1 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0.
20 DMA_CH_CUR_BUF_52	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
19 DMA_CH_CUR_BUF_51	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
18 DMA_CH_CUR_BUF_50	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

Table continues on the next page...

IPUx_CUR_BUF_1 field descriptions (continued)

Field	Description
17 DMA_CH_CUR_BUF_49	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
16 DMA_CH_CUR_BUF_48	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
15 DMA_CH_CUR_BUF_47	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
14 DMA_CH_CUR_BUF_46	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
13 DMA_CH_CUR_BUF_45	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
12 DMA_CH_CUR_BUF_44	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
11 DMA_CH_CUR_BUF_43	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
10 DMA_CH_CUR_BUF_42	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
9 DMA_CH_CUR_BUF_41	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

Table continues on the next page...

IPUx_CUR_BUF_1 field descriptions (continued)

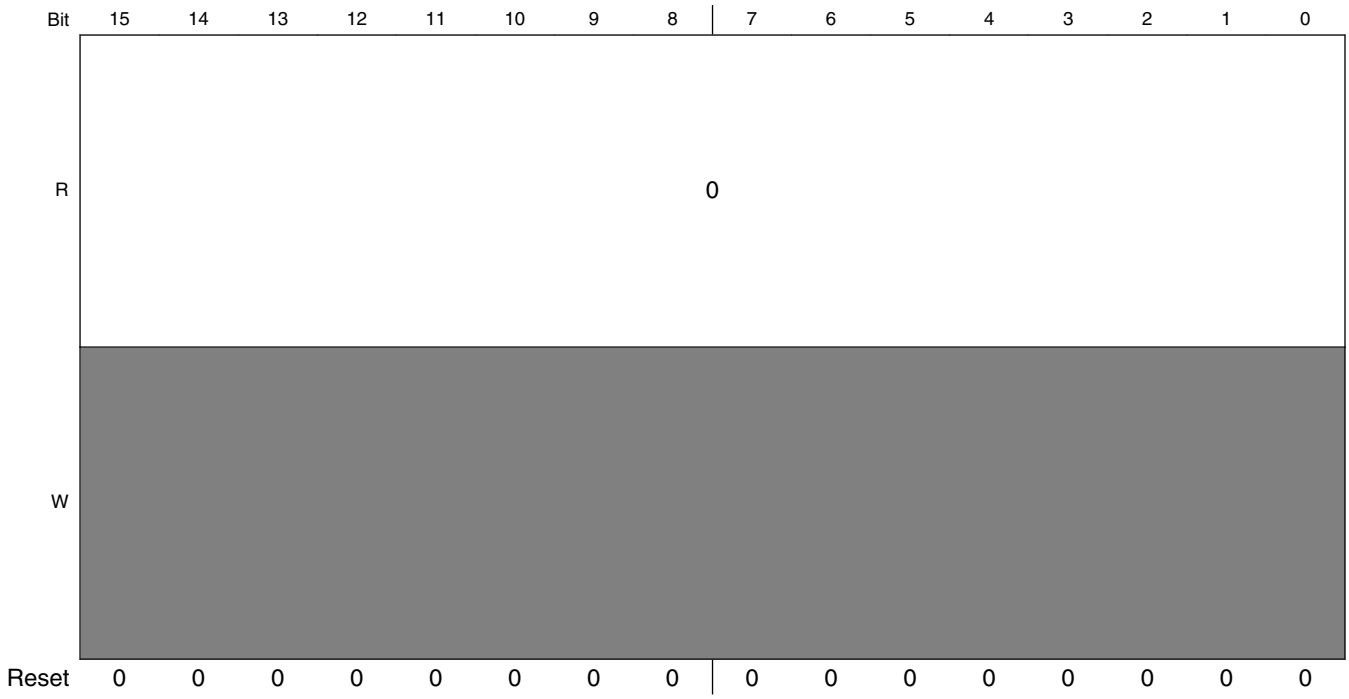
Field	Description
8 DMA_CH_CUR_BUF_40	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
7–2 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
1 DMA_CH_CUR_BUF_33	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

37.5.69 Alternate Current Buffer Register 0 (IPUx_ALT_CUR_0)

This register contains the current buffer status information bit for each DMA channel.

Address: Base address + 244h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0		DMA_CH_ALT_CUR_BUF_29	0				DMA_CH_ALT_CUR_BUF_24	0							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_ALT_CUR_0 field descriptions

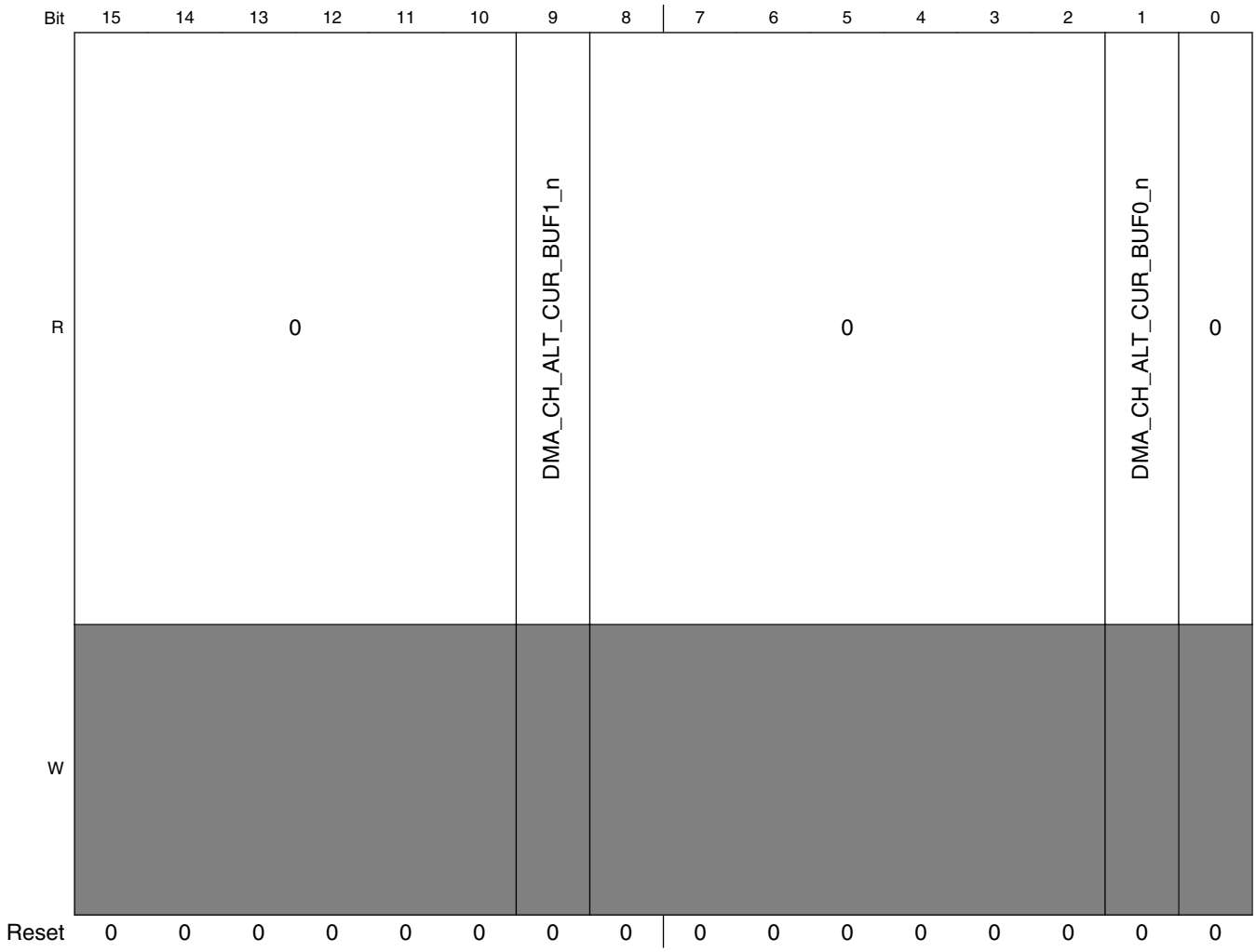
Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
29 DMA_CH_ALT_ CUR_BUF_29	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
28–25 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
24 DMA_CH_ALT_ CUR_BUF_24	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

37.5.70 Alternate Current Buffer Register 1 (IPUx_ALT_CUR_1)

This register contains the current buffer status information bit for each DMA channel. The register is shown in VDI Plane Size Register 4 , and the register fields are described in VDI Plane Size Register 4 .

Address: Base address + 248h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0											DMA_CH_ALT_CUR_BUF_52	0			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_ALT_CUR_1 field descriptions

Field	Description
31–21 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
20 DMA_CH_ALT_CUR_BUF_52	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
19–10 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
9 DMA_CH_ALT_CUR_BUF1_n	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number.

Table continues on the next page...

IPUx_ALT_CUR_1 field descriptions (continued)

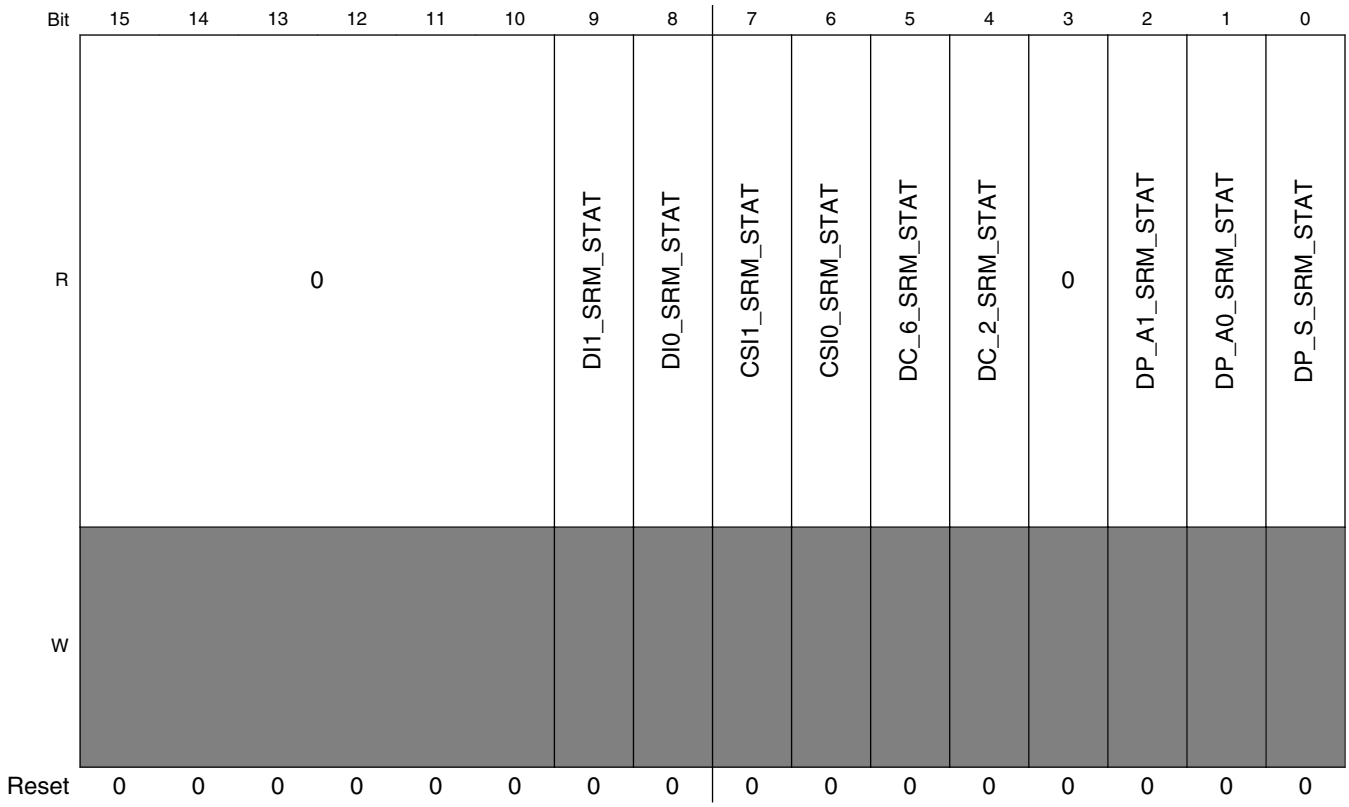
Field	Description
	0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
8–2 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
1 DMA_CH_ALT_ CUR_BUF0_n	Current Buffer. This bit indicates which buffer is in use by DMA when double buffer mode is selected. *n* Indicates the corresponding DMA channel number. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.
0 Reserved	This read-only field is reserved and always has the value 0. 0 Current buffer used by DMA is buffer 0. 1 Current buffer used by DMA is buffer 1.

37.5.71 Shadow Registers Memory Status Register (IPUx_SRM_STAT)

The register contains status bits of SRM updates. There is a bit for each block. The bit is set when the SRM is currently updating the module's registers. When the SRM completes updating the registers of the block the bit is cleared. SW should not update the block's registers while it is being updated by the SRM.

Address: Base address + 24Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_SRM_STAT field descriptions

Field	Description
31–10 Reserved	This read-only field is reserved and always has the value 0.
9 DI1_SRM_STAT	DI1 SRM STAT This bit indicates that the SRM is currently updating the DI1 registers 1 SRM is busy updating the DI1 registers 0 SRM is not updating the DI1 registers
8 DI0_SRM_STAT	DI0 SRM STAT This bit indicates that the SRM is currently updating the DI0 registers 1 SRM is busy updating the DI0 registers 0 SRM is not updating the DI0 registers
7 CSI1_SRM_ STAT	CSI1_SRM_STAT
6 CSI0_SRM_ STAT	CSI1_SRM_STAT
5 DC_6_SRM_ STAT	DC group #6 SRM STAT This bit indicates that the SRM is currently updating the DC group #6 registers

Table continues on the next page...

IPUx_SRM_STAT field descriptions (continued)

Field	Description
	1 SRM is busy updating the DC registers 0 SRM is not updating the DC registers
4 DC_2_SRM_STAT	DC group #2 SRM STAT This bit indicates that the SRM is currently updating the DC group #2 registers 1 SRM is busy updating the DC group #6 registers 0 SRM is not updating the DC group #2 registers
3 Reserved	This read-only field is reserved and always has the value 0.
2 DP_A1_SRM_STAT	DP ASYNC1 FLOW SRM STAT This bit indicates that the SRM is currently updating the DP async flow 1 registers 1 SRM is busy updating the DP sync flow registers 0 SRM is not updating the DP sync flow registers
1 DP_A0_SRM_STAT	DP ASYNC0 FLOW SRM STAT This bit indicates that the SRM is currently updating the DP async flow 0 registers 1 SRM is busy updating the DP async flow 0 registers 0 SRM is not updating the DP async flow 0 registers
0 DP_S_SRM_STAT	DP SYNC FLOW SRM STAT This bit indicates that the SRM is currently updating the DP sync flow registers 1 SRM is busy updating the DP sync flow registers 0 SRM is not updating the DP sync flow registers

37.5.72 Processing Status Tasks Register (IPUx_PROC_TASKS_STAT)

This register contains status bits for IPU's tasks.

Address: Base address + 250h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MEM2PRP_TSTAT			PP_ROT_TSTAT		VF_ROT_TSTAT		ENC_ROT_TSTAT		PP_TSTAT		VF_TSTAT		ENC_TSTAT	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_PROC_TASKS_STAT field descriptions

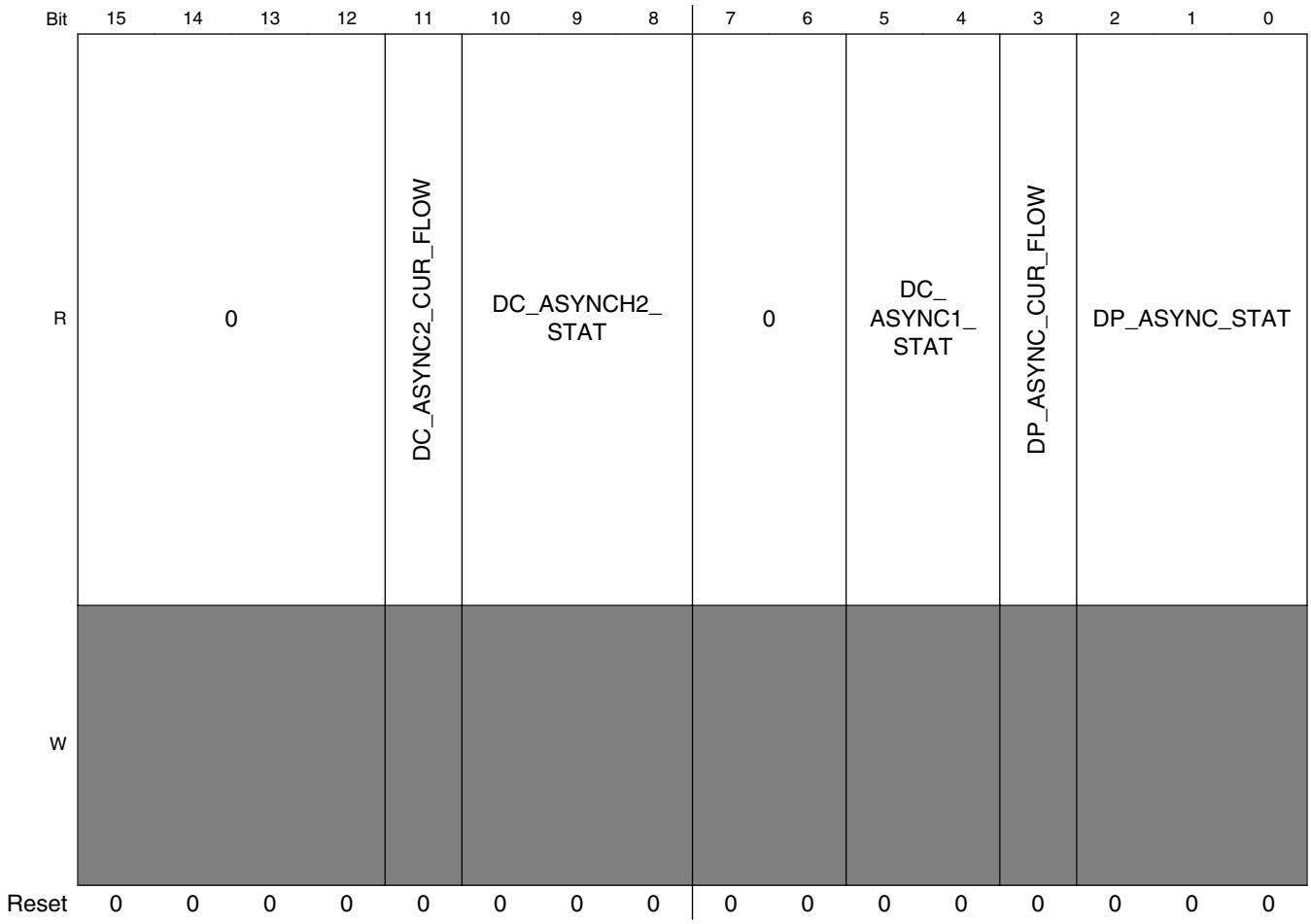
Field	Description
31–15 Reserved	This read-only field is reserved and always has the value 0.
14–12 MEM2PRP_ TSTAT	Status of the pre processing tasks (viewfinder and encoding) when the source is coming from the memory. 000 IDLE - Both pre processing tasks are idle 001 BOTH_ACTIVE - Both pre processing tasks are idle 010 ENC_ACTIVE - Encoding task is active 011 VF_ACTIVE - View finder task is active 100 BOTH_PAUSE - both tasks are paused 101 Reserved 110 Reserved 111 Reserve
11–10 PP_ROT_TSTAT	Status of the rotation for post processing task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
9–8 VF_ROT_TSTAT	Status of the rotation for viewfinder task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
7–6 ENC_ROT_ TSTAT	Status of the rotation for encoding task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
5–4 PP_TSTAT	Status of the post processing task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
3–2 VF_TSTAT	Status of the viewfinder task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
ENC_TSTAT	Status of the encoding task 00 IDLE - The task is idle 01 ACTIVE - The primary flow of this task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready

37.5.73 Display Tasks Status Register (IPUx_DISP_TASKS_STAT)

This register contains status bits for IPU's tasks.

Address: Base address + 254h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_DISP_TASKS_STAT field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
11 DC_ASYNC2_CUR_FLOW	Current asynchronous #2 flow via the DC 1 alternate flow 0 main flow
10–8 DC_ASYNCH2_STAT	Status of the Asynchronous flow #2 through the DC 000 IDLE - the task is idle 001 PRIM_ACTIVE - The primary flow of this task is currently active 010 ALT_ACTIVE - The alternate flow of this task is currently active 011 UPDATE_PARAM - The FSU is busy updating parameters from the SRM 100 PAUSE - The task is paused 101 Reserved 110 Reserved 111 Reserved
7–6 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DISP_TASKS_STAT field descriptions (continued)

Field	Description
5–4 DC_ASYNC1_ STAT	Status of the Asynchronous flow #1 through the DC (ch 28) 00 IDLE - The task is idle 01 ACTIVE - This task is currently active 10 WAIT_FOR_ READY - The task is waiting for a buffer to be ready
3 DP_ASYNC_ CUR_FLOW	Current asynchronous flow via the DP 1 alternate flow 0 main flow
DP_ASYNC_ STAT	Status of the Asynchronous flow through the DP 000 IDLE - the task is idle 001 PRIM_ACTIVE - The primary flow of this task is currently active 010 ALT_ACTIVE - The alternate flow of this task is currently active 011 UPDATE_PARAM - The FSU is busy updating parameters from the SRM 100 PAUSE - The task is paused 101 Reserved 110 Reserved 111 Reserved

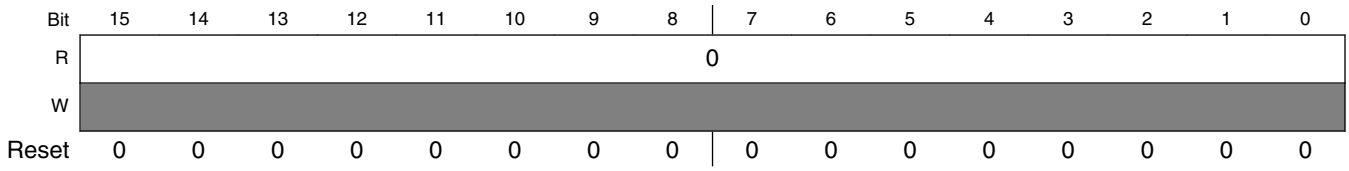
37.5.74 Triple Current Buffer Register 0 (IPUx_TRIPLE_CUR_BUF_0)

This register contains the current buffer status information for triple buffer mode for each DMA channel.

- Hide VPU_SUB_FRAME_SYNC for all versions
- Show VPU_SUB_FRAME_SYNC for IPUv3H version.
- The table below tagged with other settings (like IPU3M_only) should be hidden in IPUv3H version.
- This requires some sophisticated conditional tag settings

Address: Base address + 258h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0				DMA_CH_TRIPLE_CUR_BUF_13		0				DMA_CH_TRIPLE_CUR_BUF_10		DMA_CH_TRIPLE_CUR_BUF_9		DMA_CH_TRIPLE_CUR_BUF_8	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



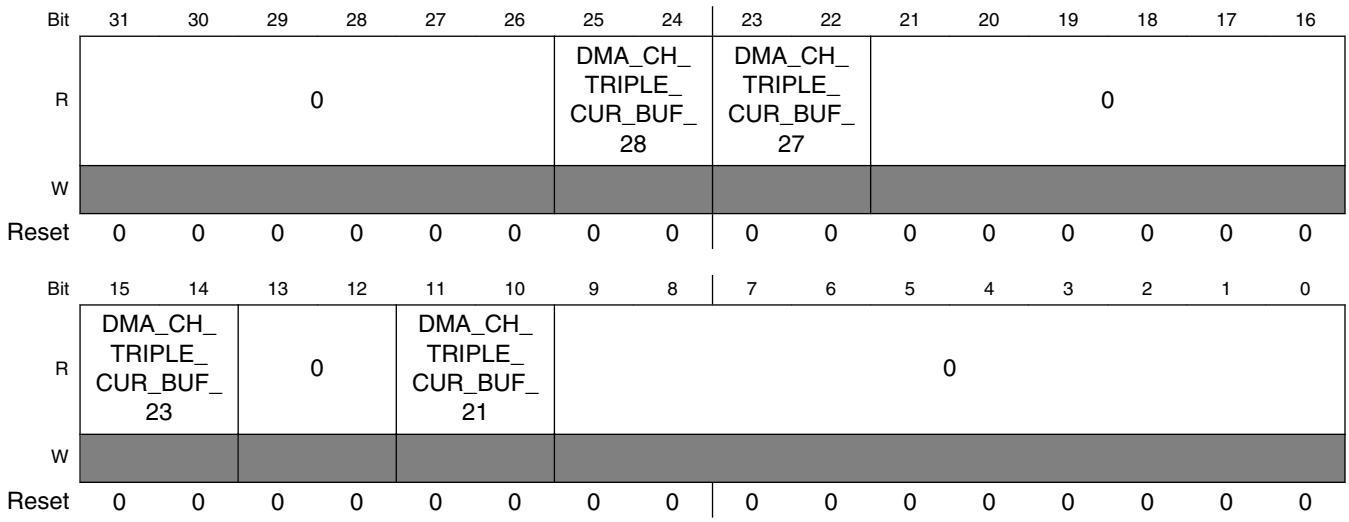
IPUx_TRIPLE_CUR_BUF_0 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–26 DMA_CH_ TRIPLE_CUR_ BUF_13	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
25–22 Reserved	This read-only field is reserved and always has the value 0.
21–20 DMA_CH_ TRIPLE_CUR_ BUF_10	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
19–18 DMA_CH_ TRIPLE_CUR_ BUF_9	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
17–16 DMA_CH_ TRIPLE_CUR_ BUF_8	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
Reserved	This read-only field is reserved and always has the value 0.

37.5.75 Triple Current Buffer Register 1 (IPUx_TRIPLE_CUR_BUF_1)

This register contains the current buffer status information for triple buffer mode for each DMA channel.

Address: Base address + 25Ch offset



IPUx_TRIPLE_CUR_BUF_1 field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25–24 DMA_CH_TRIPLE_CUR_BUF_28	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
23–22 DMA_CH_TRIPLE_CUR_BUF_27	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.

Table continues on the next page...

IPUx_TRIPLE_CUR_BUF_1 field descriptions (continued)

Field	Description
21–16 Reserved	This read-only field is reserved and always has the value 0.
15–14 DMA_CH_ TRIPLE_CUR_ BUF_23	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
13–12 Reserved	This read-only field is reserved and always has the value 0.
11–10 DMA_CH_ TRIPLE_CUR_ BUF_21	Current Buffer for triple buffer mode. This bits indicate which buffer is in use by the DMA when triple buffer mode is selected. Each pair of bits indicate the corresponding DMA channel (bits [1:0] correspond to ch #0; (bits [3:2] correspond to ch #1, etc.) 11 NA 00 Current buffer used by DMA is buffer 0. 01 Current buffer used by DMA is buffer 1. 10 Current buffer used by DMA is buffer 2.
Reserved	This read-only field is reserved and always has the value 0.

37.5.76 IPU Channels Buffer 0 Ready 0 Register (IPUx_CH_BUF0_RDY0)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (31-0). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF0_RDY0_CLR bit**.

The register is shown in IPU Channels Buffer 0 Ready 0 Register (IPU_CH_BUF0_RDY0), and the register fields are described in IPU Channels Buffer 0 Ready 0 Register (IPU_CH_BUF0_RDY0).

Address: Base address + 268h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DMA_CH_BUF0_RDY_31	Reserved	DMA_CH_BUF0_RDY_29	DMA_CH_BUF0_RDY_28	DMA_CH_BUF0_RDY_27	Reserved		DMA_CH_BUF0_RDY_24	DMA_CH_BUF0_RDY_23	DMA_CH_BUF0_RDY_22	DMA_CH_BUF0_RDY_21	DMA_CH_BUF0_RDY_20	Reserved	DMA_CH_BUF0_RDY_18	DMA_CH_BUF0_RDY_17	Reserved
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DMA_CH_BUF0_RDY_15	DMA_CH_BUF0_RDY_14	DMA_CH_BUF0_RDY_13	DMA_CH_BUF0_RDY_12	DMA_CH_BUF0_RDY_11	DMA_CH_BUF0_RDY_10	DMA_CH_BUF0_RDY_9	DMA_CH_BUF0_RDY_8	DMA_CH_BUF0_RDY_7	DMA_CH_BUF0_RDY_6	DMA_CH_BUF0_RDY_5	DMA_CH_BUF0_RDY_4	DMA_CH_BUF0_RDY_3	DMA_CH_BUF0_RDY_2	DMA_CH_BUF0_RDY_1	DMA_CH_BUF0_RDY_0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CH_BUF0_RDY0 field descriptions

Field	Description
31 DMA_CH_BUF0_RDY_31	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
30 -	This field is reserved. Reserved.
29 DMA_CH_BUF0_RDY_29	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
28 DMA_CH_BUF0_RDY_28	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
27 DMA_CH_BUF0_RDY_27	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
26–25 -	This field is reserved. Reserved.
24 DMA_CH_BUF0_RDY_24	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_CH_BUF0_RDY0 field descriptions (continued)

Field	Description
23 DMA_CH_BUF0_RDY_23	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
22 DMA_CH_BUF0_RDY_22	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
21 DMA_CH_BUF0_RDY_21	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
20 DMA_CH_BUF0_RDY_20	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
19 -	This field is reserved. Reserved.
18 DMA_CH_BUF0_RDY_18	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
17 DMA_CH_BUF0_RDY_17	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
16 -	This field is reserved. Reserved.
15 DMA_CH_BUF0_RDY_15	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
14 DMA_CH_BUF0_RDY_14	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
13 DMA_CH_BUF0_RDY_13	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
12 DMA_CH_BUF0_RDY_12	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
11 DMA_CH_BUF0_RDY_11	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_CH_BUF0_RDY0 field descriptions (continued)

Field	Description
10 DMA_CH_BUF0_RDY_10	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
9 DMA_CH_BUF0_RDY_9	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
8 DMA_CH_BUF0_RDY_8	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
7 DMA_CH_BUF0_RDY_7	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
6 DMA_CH_BUF0_RDY_6	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
5 DMA_CH_BUF0_RDY_5	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
4 DMA_CH_BUF0_RDY_4	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
3 DMA_CH_BUF0_RDY_3	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
2 DMA_CH_BUF0_RDY_2	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
1 DMA_CH_BUF0_RDY_1	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
0 DMA_CH_BUF0_RDY_0	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

37.5.77 IPU Channels Buffer 0 Ready 1 Register (IPUx_CH_BUF0_RDY1)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (63-32). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF0_RDY1_CLR bit**.

Address: Base address + 26Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved											DMA_CH_BUF0_RDY_52	DMA_CH_BUF0_RDY_51	DMA_CH_BUF0_RDY_50	DMA_CH_BUF0_RDY_49	DMA_CH_BUF0_RDY_48
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DMA_CH_BUF0_RDY_47	DMA_CH_BUF0_RDY_46	DMA_CH_BUF0_RDY_45	DMA_CH_BUF0_RDY_44	DMA_CH_BUF0_RDY_43	DMA_CH_BUF0_RDY_42	DMA_CH_BUF0_RDY_41	DMA_CH_BUF0_RDY_40	Reserved						DMA_CH_BUF0_RDY_33	Reserved
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CH_BUF0_RDY1 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved.
20 DMA_CH_BUF0_RDY_52	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
19 DMA_CH_BUF0_RDY_51	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_CH_BUF0_RDY1 field descriptions (continued)

Field	Description
18 DMA_CH_BUF0_RDY_50	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
17 DMA_CH_BUF0_RDY_49	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
16 DMA_CH_BUF0_RDY_48	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
15 DMA_CH_BUF0_RDY_47	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
14 DMA_CH_BUF0_RDY_46	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
13 DMA_CH_BUF0_RDY_45	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
12 DMA_CH_BUF0_RDY_44	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
11 DMA_CH_BUF0_RDY_43	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
10 DMA_CH_BUF0_RDY_42	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
9 DMA_CH_BUF0_RDY_41	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
8 DMA_CH_BUF0_RDY_40	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
7–2 -	This field is reserved. Reserved.
1 DMA_CH_BUF0_RDY_33	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory.

Table continues on the next page...

IPUx_CH_BUF0_RDY1 field descriptions (continued)

Field	Description
	0 Buffer 0 is not ready. 1 Buffer 0 is ready.
0 -	This field is reserved. Reserved.

37.5.78 IPU Channels Buffer 1 Ready 0 Register (IPUx_CH_BUF1_RDY0)

The register contains buffer 1 ready control information for 32 IPU's DMA channels (31-0). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF1_RDY0_CLR bit**.

Address: Base address + 270h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DMA_CH_BUF1_RDY_31	Reserved	DMA_CH_BUF1_RDY_29	DMA_CH_BUF1_RDY_28	DMA_CH_BUF1_RDY_27	DMA_CH_BUF1_RDY_26	DMA_CH_BUF1_RDY_25	DMA_CH_BUF1_RDY_24	DMA_CH_BUF1_RDY_23	DMA_CH_BUF1_RDY_22	DMA_CH_BUF1_RDY_21	DMA_CH_BUF1_RDY_20	DMA_CH_BUF1_RDY_19	DMA_CH_BUF1_RDY_18	DMA_CH_BUF1_RDY_17	Reserved
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DMA_CH_BUF1_RDY_15	DMA_CH_BUF1_RDY_14	DMA_CH_BUF1_RDY_13	DMA_CH_BUF1_RDY_12	DMA_CH_BUF1_RDY_	DMA_CH_BUF1_RDY_10	DMA_CH_BUF1_RDY_9	DMA_CH_BUF1_RDY_8	Reserved		DMA_CH_BUF1_RDY_5	Reserved	DMA_CH_BUF1_RDY_3	DMA_CH_BUF1_RDY_2	DMA_CH_BUF1_RDY_1	DMA_CH_BUF1_RDY_0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CH_BUF1_RDY0 field descriptions

Field	Description
31 DMA_CH_BUF1_ RDY_31	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory.

Table continues on the next page...

IPUx_CH_BUF1_RDY0 field descriptions (continued)

Field	Description
	0 Buffer 0 is not ready. 1 Buffer 0 is ready.
30 -	This field is reserved. Reserved.
29 DMA_CH_BUF1_RDY_29	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
28 DMA_CH_BUF1_RDY_28	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
27 DMA_CH_BUF1_RDY_27	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
26 DMA_CH_BUF1_RDY_26	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
25 DMA_CH_BUF1_RDY_25	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
24 DMA_CH_BUF1_RDY_24	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
23 DMA_CH_BUF1_RDY_23	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
22 DMA_CH_BUF1_RDY_22	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
21 DMA_CH_BUF1_RDY_21	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
20 DMA_CH_BUF1_RDY_20	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
19 DMA_CH_BUF1_RDY_19	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_CH_BUF1_RDY0 field descriptions (continued)

Field	Description
18 DMA_CH_BUF1_RDY_18	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
17 DMA_CH_BUF1_RDY_17	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
16 -	This field is reserved. Reserved.
15 DMA_CH_BUF1_RDY_15	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
14 DMA_CH_BUF1_RDY_14	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
13 DMA_CH_BUF1_RDY_13	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
12 DMA_CH_BUF1_RDY_12	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
11 DMA_CH_BUF1_RDY_	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
10 DMA_CH_BUF1_RDY_10	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
9 DMA_CH_BUF1_RDY_9	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
8 DMA_CH_BUF1_RDY_8	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
7–6 -	This field is reserved. Reserved.
5 DMA_CH_BUF1_RDY_5	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_CH_BUF1_RDY0 field descriptions (continued)

Field	Description
4 -	This field is reserved. Reserved.
3 DMA_CH_BUF1_RDY_3	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
2 DMA_CH_BUF1_RDY_2	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
1 DMA_CH_BUF1_RDY_1	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
0 DMA_CH_BUF1_RDY_0	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

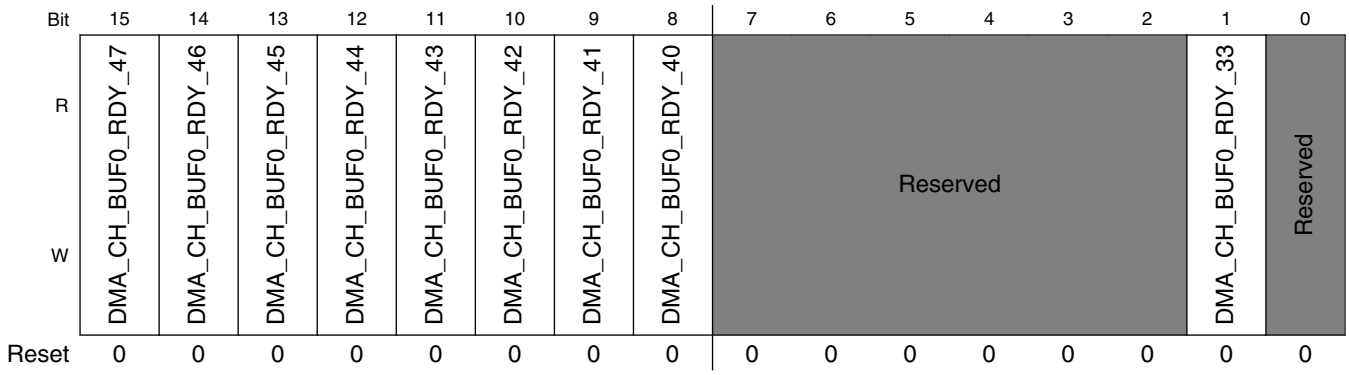
37.5.79 IPU Channels Buffer 1 Ready 1Register (IPUx_CH_BUF1_RDY1)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (63-32). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF1_RDY1_CLR bit**.

The register is shown in IPU Channels Buffer 1 Ready 1Register (IPU_CH_BUF1_RDY1), and the register fields are described in IPU Channels Buffer 1 Ready 1Register (IPU_CH_BUF1_RDY1).

Address: Base address + 274h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved											DMA_CH_BUF0_RDY_52	DMA_CH_BUF0_RDY_51	DMA_CH_BUF0_RDY_50	DMA_CH_BUF0_RDY_49	DMA_CH_BUF0_RDY_48
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_CH_BUF1_RDY1 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved.
20 DMA_CH_BUF0_RDY_52	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
19 DMA_CH_BUF0_RDY_51	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
18 DMA_CH_BUF0_RDY_50	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
17 DMA_CH_BUF0_RDY_49	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
16 DMA_CH_BUF0_RDY_48	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
15 DMA_CH_BUF0_RDY_47	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
14 DMA_CH_BUF0_RDY_46	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
13 DMA_CH_BUF0_RDY_45	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.

Table continues on the next page...

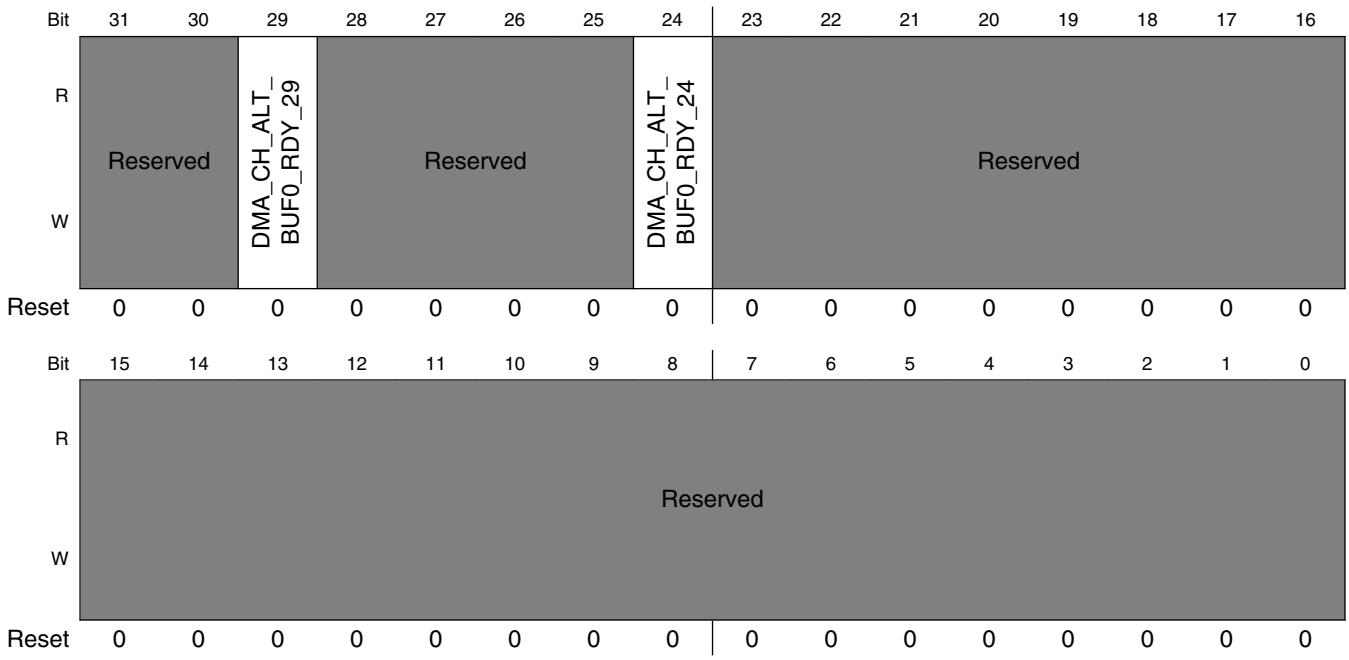
IPUx_CH_BUF1_RDY1 field descriptions (continued)

Field	Description
12 DMA_CH_BUF0_RDY_44	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
11 DMA_CH_BUF0_RDY_43	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
10 DMA_CH_BUF0_RDY_42	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
9 DMA_CH_BUF0_RDY_41	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
8 DMA_CH_BUF0_RDY_40	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
7–2 -	This field is reserved. Reserved.
1 DMA_CH_BUF0_RDY_33	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 Buffer 1 is ready.
0 -	This field is reserved. Reserved.

37.5.80 IPU Alternate Channels Buffer 0 Ready 0 Register (IPUx_ALT_CH_BUF0_RDY0)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (31-0). Writing "1" to each field will set each bit. Writing "0" to each field simultaneously will clear all the bits.

Address: Base address + 278h offset



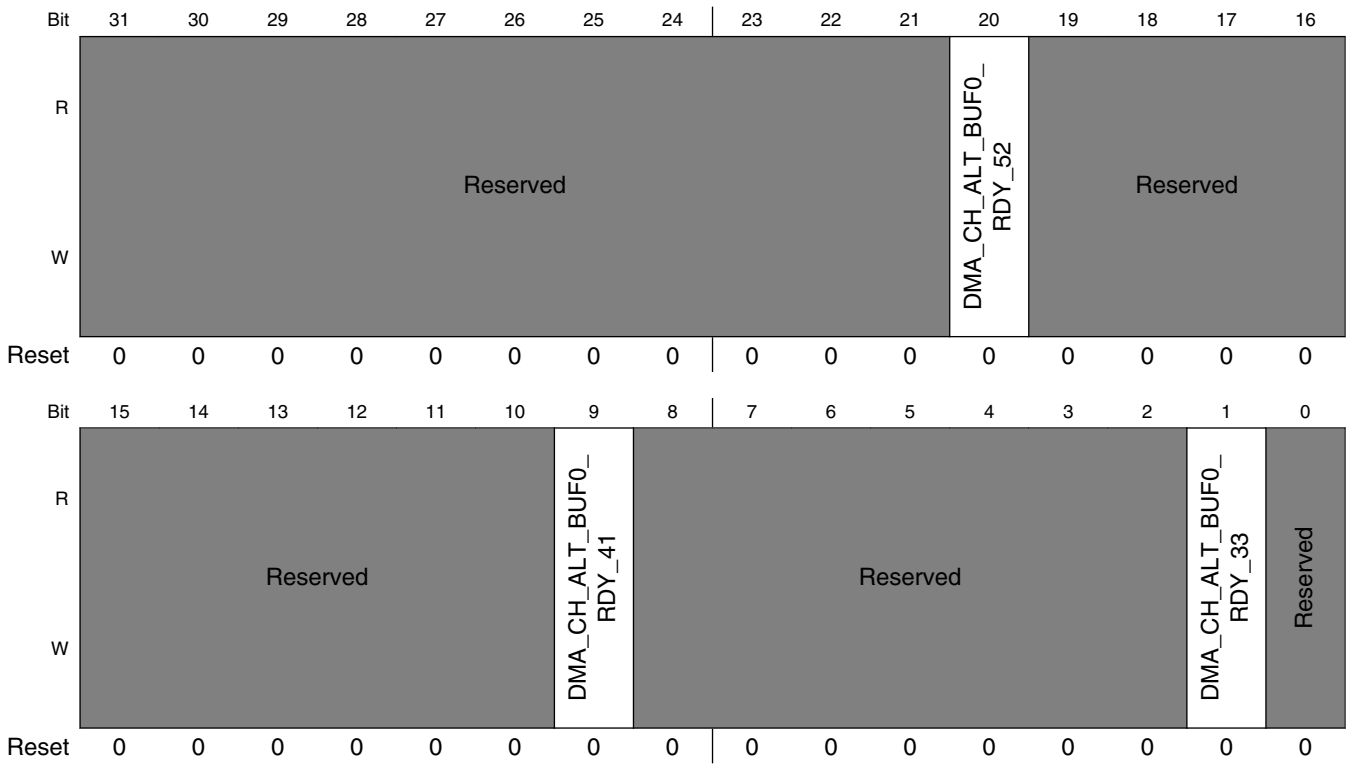
IPUx_ALT_CH_BUF0_RDY0 field descriptions

Field	Description
31–30 -	This field is reserved. Reserved.
29 DMA_CH_ALT_BUF0_RDY_29	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
28–25 -	This field is reserved. Reserved.
24 DMA_CH_ALT_BUF0_RDY_24	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
-	This field is reserved. Reserved.

37.5.81 IPU Alternate Channels Buffer 0 Ready 1 Register (IPUx_ALT_CH_BUF0_RDY1)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (63-32). Writing "1" to each field will set each bit. Writing "0" to each field simultaneously will clear all the bits.

Address: Base address + 27Ch offset



IPUx_ALT_CH_BUF0_RDY1 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved.
20 DMA_CH_ALT_BUF0_RDY_52	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
19–10 -	This field is reserved. Reserved.
9 DMA_CH_ALT_BUF0_RDY_41	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.

Table continues on the next page...

IPUx_ALT_CH_BUF0_RDY1 field descriptions (continued)

Field	Description
8–2 -	This field is reserved. Reserved.
1 DMA_CH_ALT_BUF0_RDY_33	Buffer 0 is ready. This bit indicates that ARM platform finished/writing reading buffer 0 in memory. 0 Buffer 0 is not ready. 1 Buffer 0 is ready.
0 -	This field is reserved. Reserved.

37.5.82 IPU Alternate Channels Buffer1 Ready 0 Register (IPUx_ALT_CH_BUF1_RDY0)

The register contains buffer 1 ready control information for 32 IPU's DMA channels (31-0). Writing "1" to each field will set each bit. Writing "0" to each field simultaneously will clear all the bits.

The register is shown in IPU Alternate Channels Buffer1 Ready 0 Register (IPU_ALT_CH_BUF1_RDY0), and the register fields are described in IPU Alternate Channels Buffer1 Ready 0 Register (IPU_ALT_CH_BUF1_RDY0).

Address: Base address + 280h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved		DMA_CH_ALT_BUF1_RDY_29	Reserved				DMA_CH_ALT_BUF1_RDY_24	Reserved							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_ALT_CH_BUF1_RDY0 field descriptions

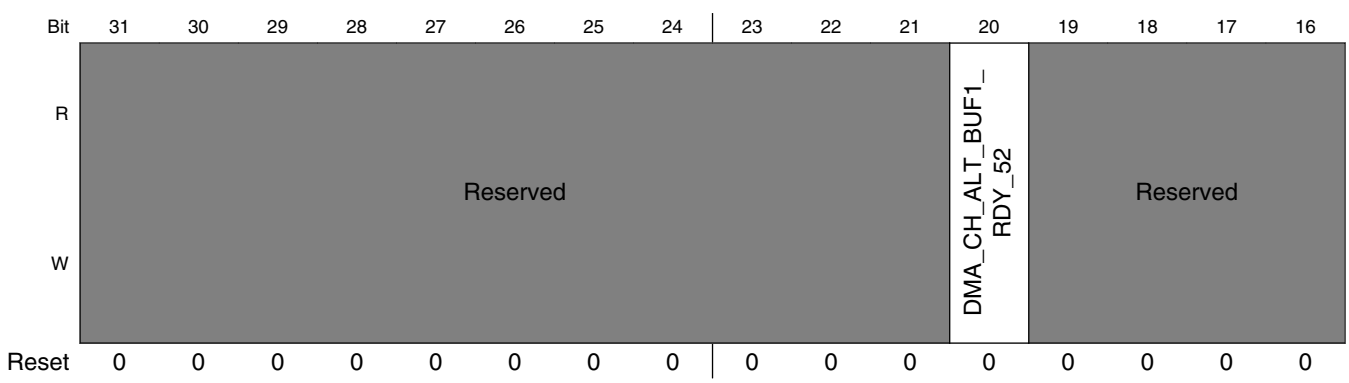
Field	Description
31–30 -	This field is reserved. Reserved.
29 DMA_CH_ALT_BUF1_RDY_29	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 buffer 1 is ready.
28–25 -	This field is reserved. Reserved.
24 DMA_CH_ALT_BUF1_RDY_24	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 buffer 1 is ready.
-	This field is reserved. Reserved.

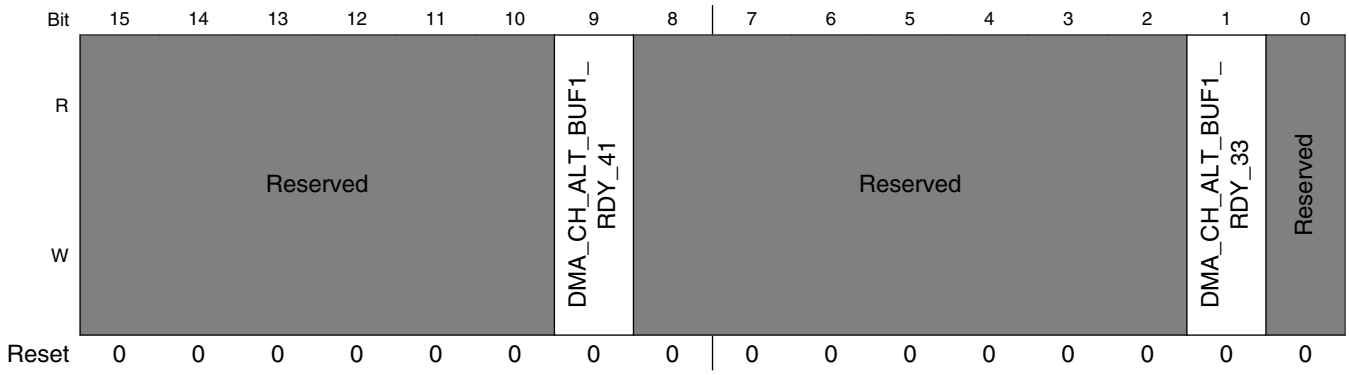
37.5.83 IPU Alternate Channels Buffer 1 Ready 1 Register (IPUx_ALT_CH_BUF1_RDY1)

The register contains buffer 0 ready control information for 32 IPU's DMA channels (63-32). Writing "1" to each field will set each bit. Writing "0" to each field simultaneously will clear all the bits.

The register is shown in IPU Alternate Channels Buffer 1 Ready 1 Register (IPU_ALT_CH_BUF1_RDY1), and the register fields are described in IPU Alternate Channels Buffer 1 Ready 1 Register (IPU_ALT_CH_BUF1_RDY1).

Address: Base address + 284h offset





IPUx_ALT_CH_BUF1_RDY1 field descriptions

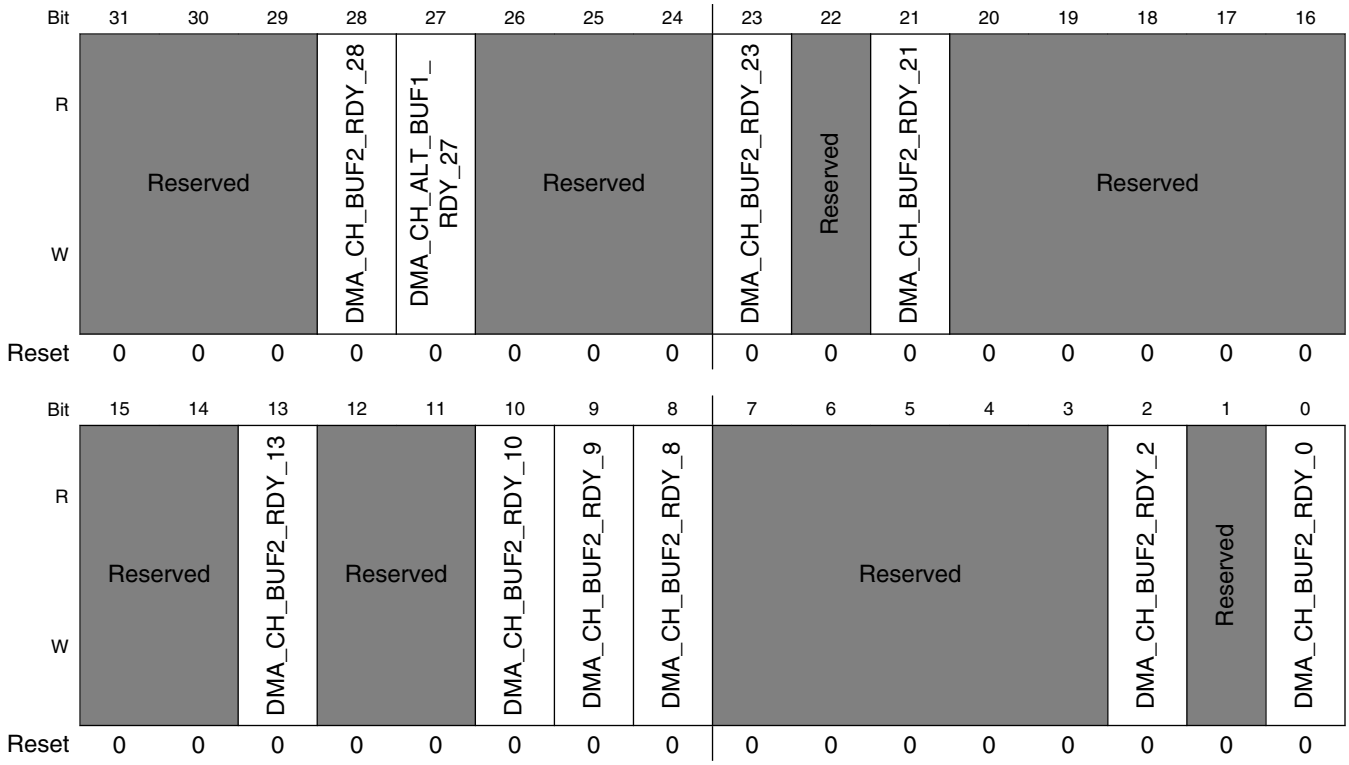
Field	Description
31–21 -	This field is reserved. Reserved.
20 DMA_CH_ALT_BUF1_RDY_52	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 buffer 1 is ready.
19–10 -	This field is reserved. Reserved.
9 DMA_CH_ALT_BUF1_RDY_41	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 buffer 1 is ready.
8–2 -	This field is reserved. Reserved.
1 DMA_CH_ALT_BUF1_RDY_33	Buffer 1 is ready. This bit indicates that ARM platform finished/writing reading buffer 1 in memory. 0 buffer 1 is not ready. 1 buffer 1 is ready.
0 -	This field is reserved. Reserved.

37.5.84 IPU Channels Buffer 2 Ready 0 Register (IPUx_CH_BUF2_RDY0)

The register contains buffer 2 ready control information for 32 IPU's DMA channels (31-0). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF2_RDY0_CLR bit**.

The register is shown in IPU Channels Buffer 2 Ready 0 Register (IPU_CH_BUF2_RDY0), and the register fields are described in IPU Channels Buffer 2 Ready 0 Register (IPU_CH_BUF2_RDY0).

Address: Base address + 288h offset



IPUx_CH_BUF2_RDY0 field descriptions

Field	Description
31–29 -	This field is reserved. Reserved.
28 DMA_CH_BUF2_RDY_28	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
27 DMA_CH_ALT_BUF1_RDY_27	buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 buffer 2 is not ready. 1 buffer 2 is ready.
26–24 -	This field is reserved. Reserved.
23 DMA_CH_BUF2_RDY_23	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
22 -	This field is reserved. Reserved.
21 DMA_CH_BUF2_RDY_21	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
20–14 -	This field is reserved. Reserved.

Table continues on the next page...

IPUx_CH_BUF2_RDY0 field descriptions (continued)

Field	Description
13 DMA_CH_BUF2_RDY_13	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
12–11 -	This field is reserved. Reserved.
10 DMA_CH_BUF2_RDY_10	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
9 DMA_CH_BUF2_RDY_9	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
8 DMA_CH_BUF2_RDY_8	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
7–3 -	This field is reserved. Reserved.
2 DMA_CH_BUF2_RDY_2	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.
1 -	This field is reserved. Reserved.
0 DMA_CH_BUF2_RDY_0	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.

37.5.85 IPU Channels Buffer 2 Ready 1 Register (IPUx_CH_BUF2_RDY1)

The register contains buffer 2 ready control information for 32 IPU's DMA channels (63-32). This register can be a write one to set or a write one to clear according to the **IPU_CH_BUF2_RDY1_CLR bit.**

The register is shown in IPU Channels Buffer 2 Ready 1 Register (IPU_CH_BUF2_RDY1), and the register fields are described in IPU Channels Buffer 2 Ready 1 Register (IPU_CH_BUF2_RDY1).

Address: Base address + 28Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DMA_CH_BUF2_RDY_x																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CH_BUF2_RDY1 field descriptions

Field	Description
DMA_CH_BUF2_RDY_x	Buffer 2 is ready. This bit indicates that ARM platform finished/writing reading buffer 2 in memory. 0 Buffer 2 is not ready. 1 Buffer 2 is ready.

37.5.86 IDMAC Configuration Register (IPUx_IDMAC_CONF)

Address: Base address + 8000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved						USED_BUFS_EN_R	USED_BUFS_MAX_R				USED_BUFS_EN_W	USED_BUFS_MAX_W			P_ENDIAN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved										RDI	WIDPT	MAX_REQ_READ			
Reset	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1

IPUx_IDMAC_CONF field descriptions

Field	Description
31–26 -	This field is reserved. Reserved, should be cleared.
25 USED_BUFS_EN_R	Enables the limit on the number of pending non real time read requests.
24–21 USED_BUFS_MAX_R	Limit the number of pending non real time read requests. The value can be between 0 to 8. This field has no affect if USED_BUFS_EN_R is cleared
20 USED_BUFS_EN_W	Enables the limit on the number of pending non real time write requests.

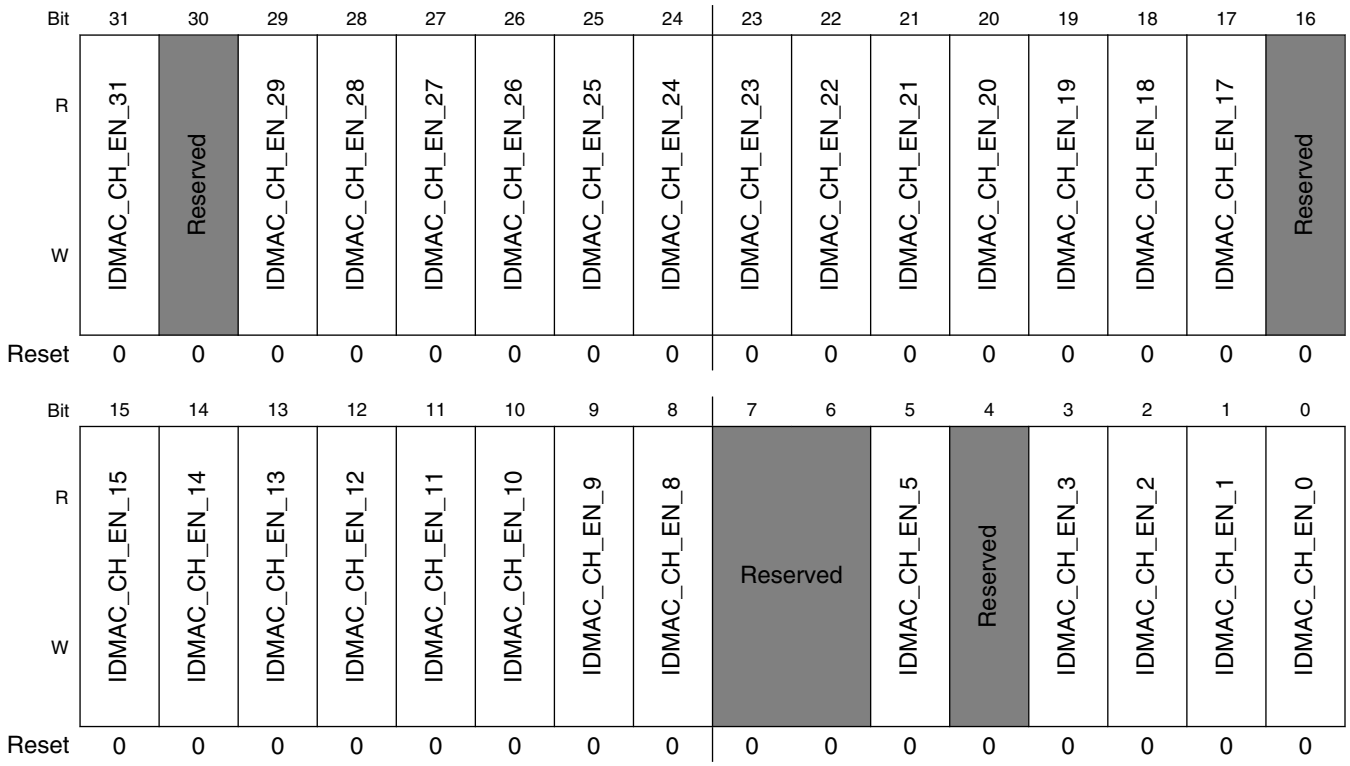
Table continues on the next page...

IPUx_IDMAC_CONF field descriptions (continued)

Field	Description
19–17 USED_BUFS_MAX_W	Limit the number of pending non real time write requests. The value can be between 0 to 6. This field has no affect if USED_BUFS_EN_W is cleared
16 P_ENDIAN	Pixel Endianness. The pixel Endianness must not be changed while any of the IDMAC channels is enabled. 0 little endian 1 Big endian
15–6 -	This field is reserved. Reserved, should be cleared.
5 RDI	Read Data Interleaving. This bit must match the slave read data interleaving support. If the AXI slave connected to the IPU supports read data interleaving then this bit must be set. If the AXI slave does not support read data interleaving then the IDMAC can utilize this and issue more address phases on read. In that case it is recommended to have this bit cleared. 0 The AXI slave does not support read data interleaving 1 The AXI slave supports read data interleaving
4–3 WIDPT	Write Interleaving Depth These 2 bits define the Write Interleaving Depth of the AXI port. This bits should be configured by the user according to the AXI slave's Write Interleaving Depth. WIDPT defines the maximal number of active bursts (yet to be responded) with different IDs. IDMAC will block data phase if the next data's ID is new (no such ID active) and the number of active IDs is equal to WIDPT. 00 Write Interleaving Depth of 1 01 Write Interleaving Depth of 2 10 Write Interleaving Depth of 3 11 Write Interleaving Depth of 4
MAX_REQ_READ	Maximum Read Requests. This fields sets the maximum pending requests allowed in the AXI Read requests queue.

37.5.87 IDMAC Channel Enable 1 Register (IPUx_IDMAC_CH_EN_1)

Address: Base address + 8004h offset



IPUx_IDMAC_CH_EN_1 field descriptions

Field	Description
31 IDMAC_CH_EN_31	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
30 -	This field is reserved. Reserved.
29 IDMAC_CH_EN_29	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
28 IDMAC_CH_EN_28	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
27 IDMAC_CH_EN_27	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled

Table continues on the next page...

IPUx_IDMAC_CH_EN_1 field descriptions (continued)

Field	Description
26 IDMAC_CH_EN_26	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
25 IDMAC_CH_EN_25	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
24 IDMAC_CH_EN_24	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
23 IDMAC_CH_EN_23	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
22 IDMAC_CH_EN_22	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
21 IDMAC_CH_EN_21	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
20 IDMAC_CH_EN_20	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
19 IDMAC_CH_EN_19	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
18 IDMAC_CH_EN_18	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
17 IDMAC_CH_EN_17	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
16 -	This field is reserved. Reserved.
15 IDMAC_CH_EN_15	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
14 IDMAC_CH_EN_14	IDMAC Channel enable bit [i]

Table continues on the next page...

IPUx_IDMAC_CH_EN_1 field descriptions (continued)

Field	Description
	0 IDMAC channel is disabled 1 IDMAC channel is enabled
13 IDMAC_CH_EN_13	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
12 IDMAC_CH_EN_12	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
11 IDMAC_CH_EN_11	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
10 IDMAC_CH_EN_10	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
9 IDMAC_CH_EN_9	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
8 IDMAC_CH_EN_8	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
7–6 -	This field is reserved. Reserved.
5 IDMAC_CH_EN_5	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
4 -	This field is reserved. Reserved.
3 IDMAC_CH_EN_3	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
2 IDMAC_CH_EN_2	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
1 IDMAC_CH_EN_1	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
0 IDMAC_CH_EN_0	IDMAC Channel enable bit [i]

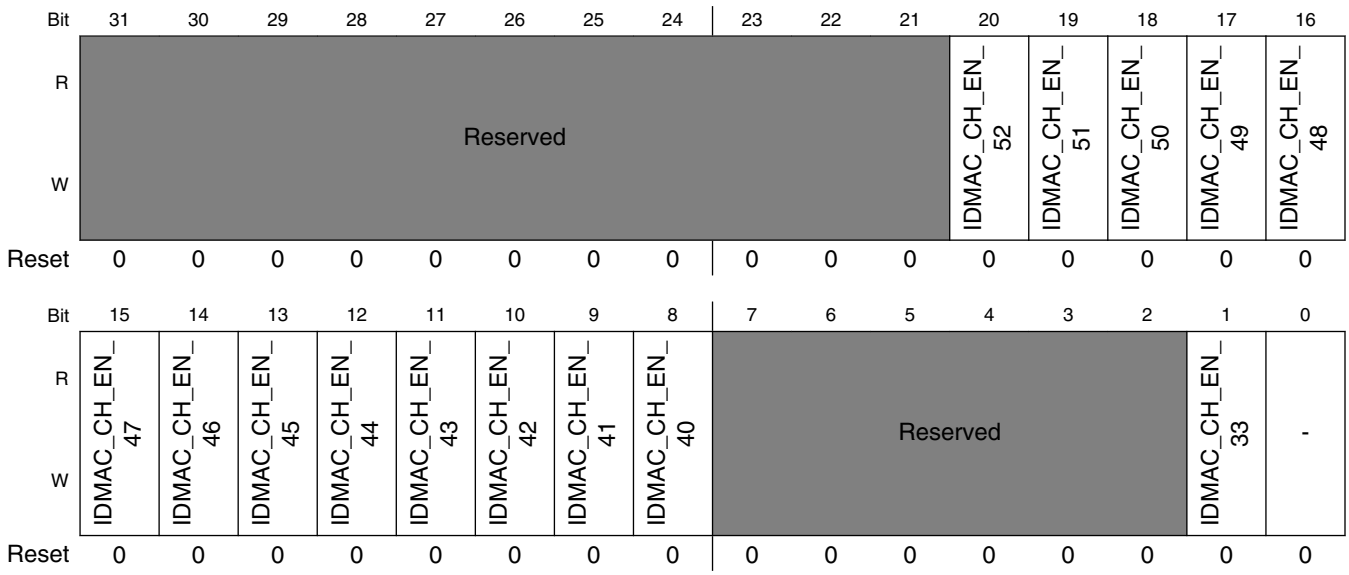
Table continues on the next page...

IPUx_IDMAC_CH_EN_1 field descriptions (continued)

Field	Description
	0 IDMAC channel is disabled 1 IDMAC channel is enabled

37.5.88 IDMAC Channel Enable 2 Register (IPUx_IDMAC_CH_EN_2)

Address: Base address + 8008h offset



IPUx_IDMAC_CH_EN_2 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved.
20 IDMAC_CH_EN_52	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
19 IDMAC_CH_EN_51	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
18 IDMAC_CH_EN_50	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
17 IDMAC_CH_EN_49	IDMAC Channel enable bit [i]

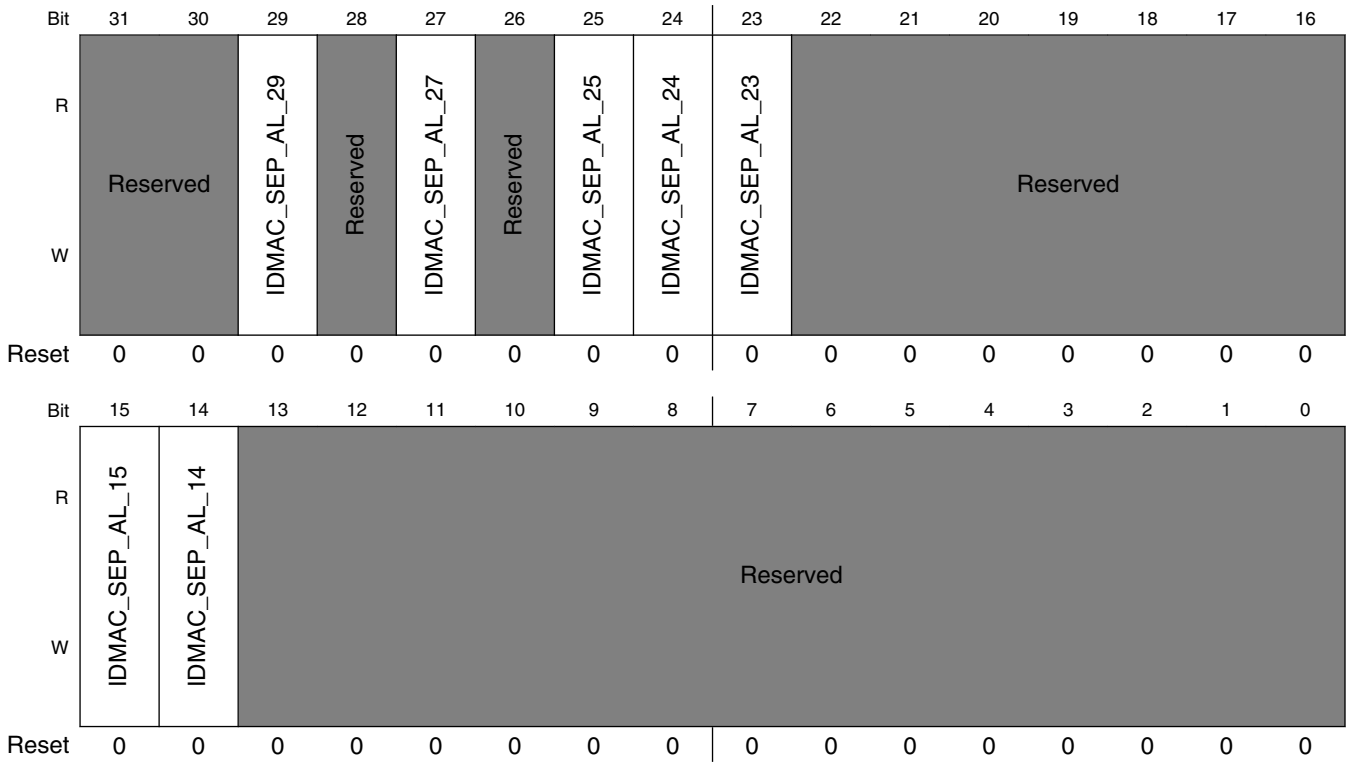
Table continues on the next page...

IPUx_IDMAC_CH_EN_2 field descriptions (continued)

Field	Description
	0 IDMAC channel is disabled 1 IDMAC channel is enabled
16 IDMAC_CH_EN_48	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
15 IDMAC_CH_EN_47	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
14 IDMAC_CH_EN_46	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
13 IDMAC_CH_EN_45	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
12 IDMAC_CH_EN_44	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
11 IDMAC_CH_EN_43	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
10 IDMAC_CH_EN_42	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
9 IDMAC_CH_EN_41	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
8 IDMAC_CH_EN_40	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
7–2 -	This field is reserved. Reserved.
1 IDMAC_CH_EN_33	IDMAC Channel enable bit [i] 0 IDMAC channel is disabled 1 IDMAC channel is enabled
0 -	Reserved.

37.5.89 IDMAC Separate Alpha Indication Register (IPUx_IDMAC_SEP_ALPHA)

Address: Base address + 800Ch offset



IPUx_IDMAC_SEP_ALPHA field descriptions

Field	Description
31–30 -	This field is reserved. Reserved.
29 IDMAC_SEP_AL_29	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
28 -	This field is reserved. Reserved.
27 IDMAC_SEP_AL_27	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel.

Table continues on the next page...

IPUx_IDMAC_SEP_ALPHA field descriptions (continued)

Field	Description
	In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
26 -	This field is reserved. Reserved.
25 IDMAC_SEP_AL_25	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
24 IDMAC_SEP_AL_24	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
23 IDMAC_SEP_AL_23	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
22–16 -	This field is reserved. Reserved.
15 IDMAC_SEP_AL_15	IDMAC Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
14 IDMAC_SEP_AL_14	IDMAC Separate alpha indication bit [i]

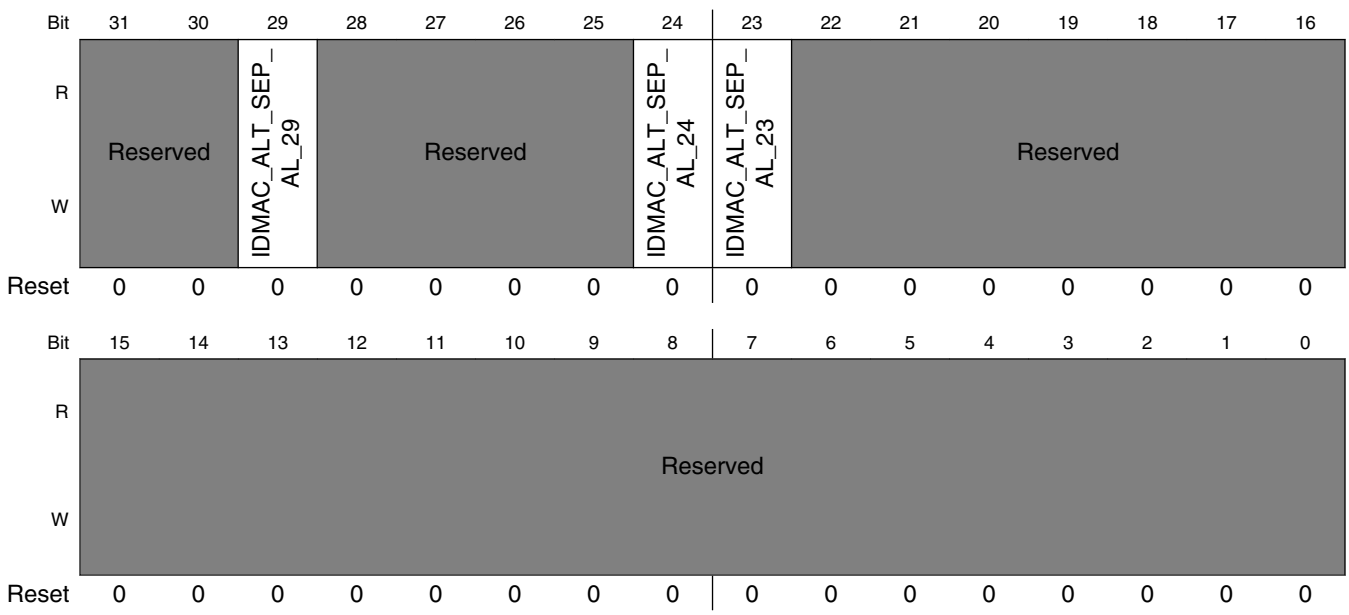
Table continues on the next page...

IPUx_IDMAC_SEP_ALPHA field descriptions (continued)

Field	Description
	A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
-	This field is reserved. Reserved.

37.5.90 IDMAC Alternate Separate Alpha Indication Register (IPUx_IDMAC_ALT_SEP_ALPHA)

Address: Base address + 8010h offset



IPUx_IDMAC_ALT_SEP_ALPHA field descriptions

Field	Description
31–30 -	This field is reserved. Reserved.
29 IDMAC_ALT_SEP_AL_29	IDMAC Alternate Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding.

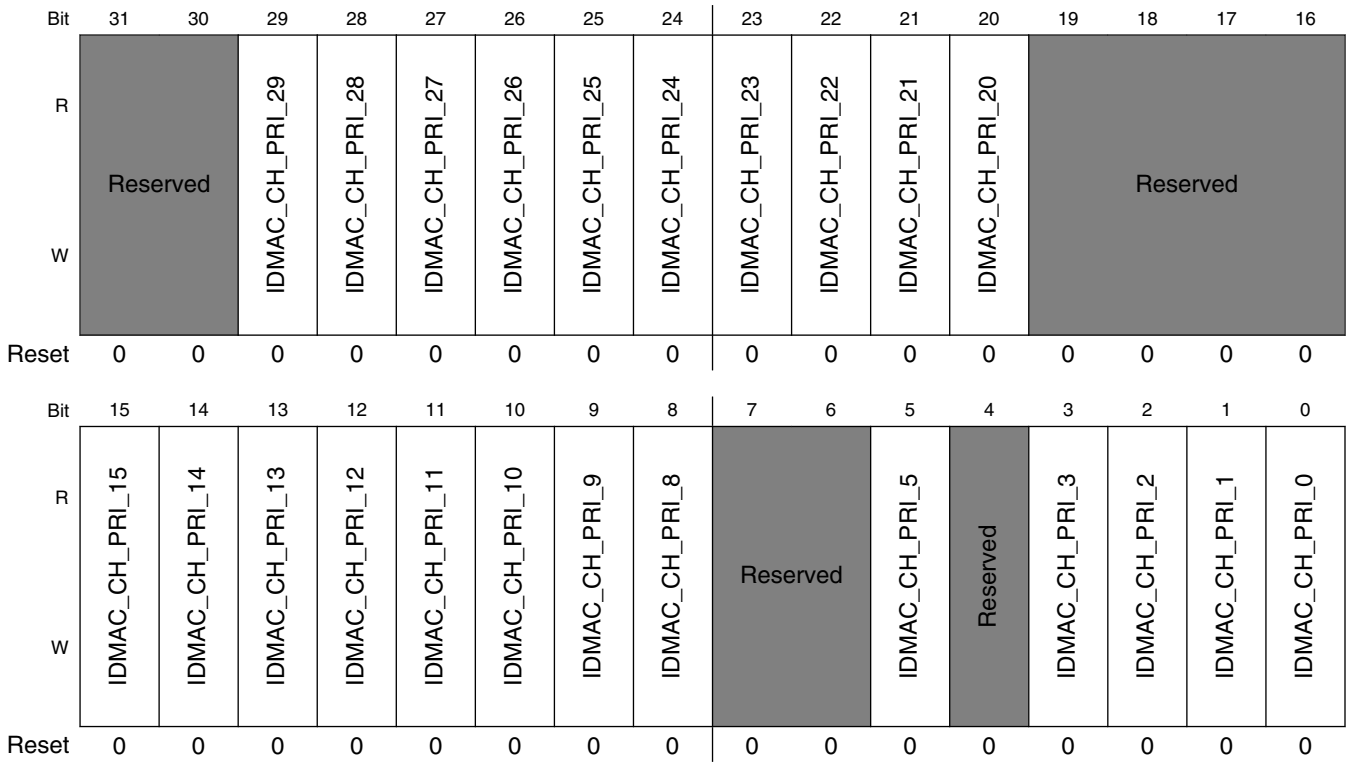
Table continues on the next page...

IPUx_IDMAC_ALT_SEP_ALPHA field descriptions (continued)

Field	Description
	For channels that may have alternate flow and may use separate alpha. This bit indicates the mode of the alpha for the alternate flow 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
28–25 -	This field is reserved. Reserved.
24 IDMAC_ALT_ SEP_AL_24	IDMAC Alternate Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. For channels that may have alternate flow and may use separate alpha. This bit indicates the mode of the alpha for the alternate flow 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
23 IDMAC_ALT_ SEP_AL_23	IDMAC Alternate Separate alpha indication bit [i] A sub block may need to read data from the system's memory where the pixel data and the alpha transparency data are located in separate buffers. In that case the Alpha transparency data is read by the special Alpha channel. In a case where the alpha should be read from a separate buffer, the user should set the channels corresponding. For channels that may have alternate flow and may use separate alpha. This bit indicates the mode of the alpha for the alternate flow 0 Channel [i] does not read Alpha transparency data from a separate buffer. 1 Channel [i] reads Alpha transparency data from a separate buffer.
-	This field is reserved. Reserved.

37.5.91 IDMAC Channel Priority 1 Register (IPUx_IDMAC_CH_PRI_1)

Address: Base address + 8014h offset



IPUx_IDMAC_CH_PRI_1 field descriptions

Field	Description
31–30 -	This field is reserved. Reserved.
29 IDMAC_CH_ PRI_29	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
28 IDMAC_CH_ PRI_28	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
27 IDMAC_CH_ PRI_27	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
26 IDMAC_CH_ PRI_26	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority

Table continues on the next page...

IPUx_IDMAC_CH_PRI_1 field descriptions (continued)

Field	Description
25 IDMAC_CH_ PRI_25	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
24 IDMAC_CH_ PRI_24	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
23 IDMAC_CH_ PRI_23	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
22 IDMAC_CH_ PRI_22	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
21 IDMAC_CH_ PRI_21	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
20 IDMAC_CH_ PRI_20	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
19–16 -	This field is reserved. Reserved.
15 IDMAC_CH_ PRI_15	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
14 IDMAC_CH_ PRI_14	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
13 IDMAC_CH_ PRI_13	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
12 IDMAC_CH_ PRI_12	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
11 IDMAC_CH_ PRI_11	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
10 IDMAC_CH_ PRI_10	IDMAC Channel enable bit [i]

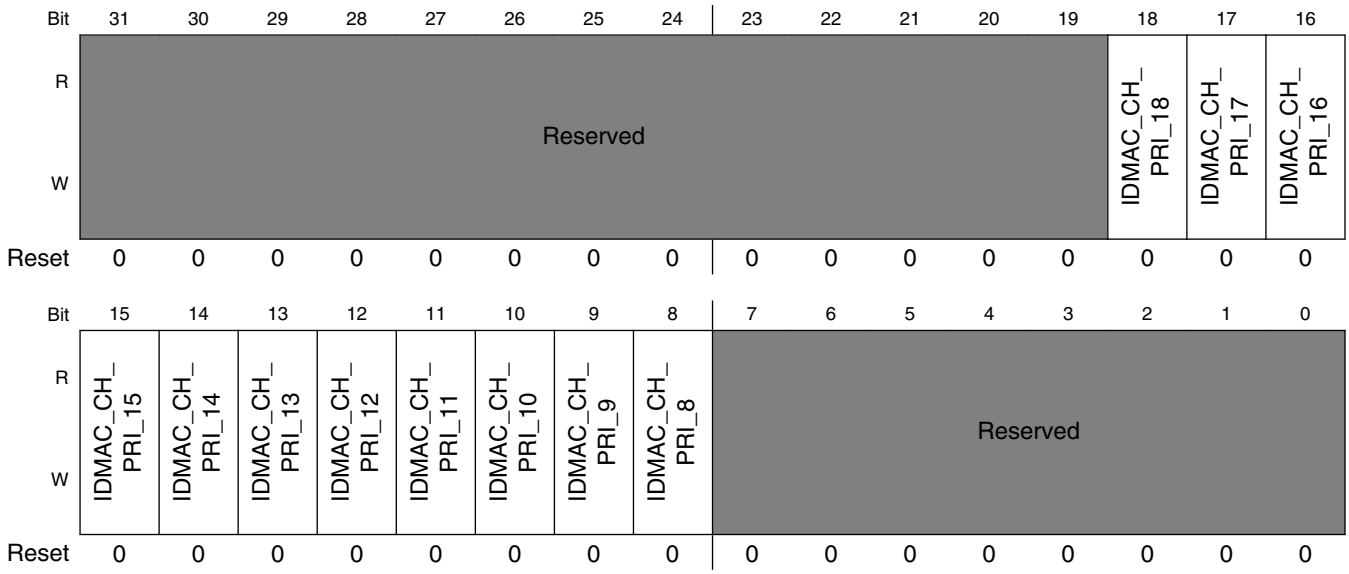
Table continues on the next page...

IPUx_IDMAC_CH_PRI_1 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
9 IDMAC_CH_PRI_9	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
8 IDMAC_CH_PRI_8	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
7–6 -	This field is reserved. Reserved.
5 IDMAC_CH_PRI_5	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
4 -	This field is reserved. Reserved.
3 IDMAC_CH_PRI_3	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
2 IDMAC_CH_PRI_2	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
1 IDMAC_CH_PRI_1	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
0 IDMAC_CH_PRI_0	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority

37.5.92 IDMAC Channel Priority 2 Register (IPUx_IDMAC_CH_PRI_2)

Address: Base address + 8018h offset



IPUx_IDMAC_CH_PRI_2 field descriptions

Field	Description
31–19 -	This field is reserved. Reserved.
18 IDMAC_CH_PRI_18	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
17 IDMAC_CH_PRI_17	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
16 IDMAC_CH_PRI_16	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
15 IDMAC_CH_PRI_15	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
14 IDMAC_CH_PRI_14	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority

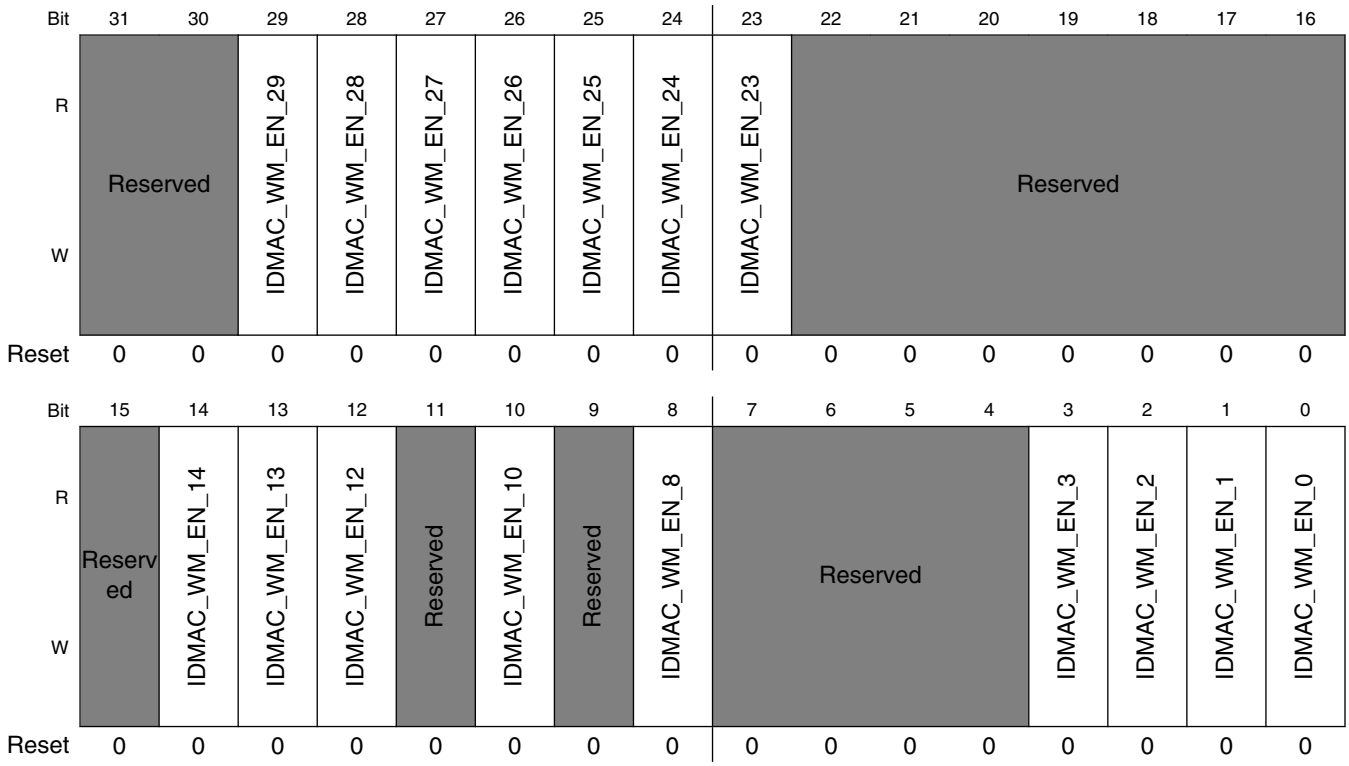
Table continues on the next page...

IPUx_IDMAC_CH_PRI_2 field descriptions (continued)

Field	Description
13 IDMAC_CH_ PRI_13	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
12 IDMAC_CH_ PRI_12	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
11 IDMAC_CH_ PRI_11	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
10 IDMAC_CH_ PRI_10	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
9 IDMAC_CH_ PRI_9	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
8 IDMAC_CH_ PRI_8	IDMAC Channel enable bit [i] 0 IDMAC channel [i] is in low priority 1 IDMAC channel [i] is in high priority
-	This field is reserved. Reserved.

37.5.93 IDMAC Channel Watermark Enable 1 Register (IPUx_IDMAC_WM_EN_1)

Address: Base address + 801Ch offset



IPUx_IDMAC_WM_EN_1 field descriptions

Field	Description
31–30 -	This field is reserved. Reserved.
29 IDMAC_WM_EN_29	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
28 IDMAC_WM_EN_28	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
27 IDMAC_WM_EN_27	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
26 IDMAC_WM_EN_26	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled

Table continues on the next page...

IPUx_IDMAC_WM_EN_1 field descriptions (continued)

Field	Description
25 IDMAC_WM_EN_25	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
24 IDMAC_WM_EN_24	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
23 IDMAC_WM_EN_23	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
22–15 -	This field is reserved. Reserved.
14 IDMAC_WM_EN_14	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
13 IDMAC_WM_EN_13	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
12 IDMAC_WM_EN_12	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
11 -	This field is reserved. Reserved.
10 IDMAC_WM_EN_10	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
9 -	This field is reserved. Reserved.
8 IDMAC_WM_EN_8	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
7–4 -	This field is reserved. Reserved.
3 IDMAC_WM_EN_3	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
2 IDMAC_WM_EN_2	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled

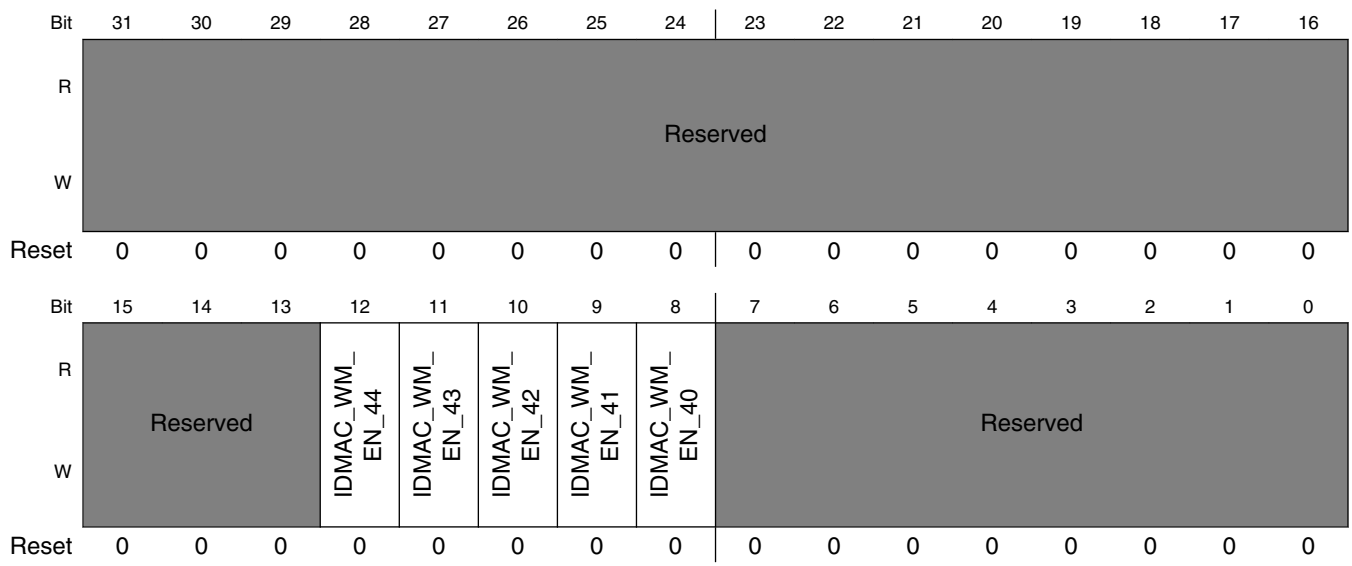
Table continues on the next page...

IPUx_IDMAC_WM_EN_1 field descriptions (continued)

Field	Description
1 IDMAC_WM_ EN_1	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
0 IDMAC_WM_ EN_0	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled

37.5.94 IDMAC Channel Watermark Enable 2 Register (IPUx_IDMAC_WM_EN_2)

Address: Base address + 8020h offset



IPUx_IDMAC_WM_EN_2 field descriptions

Field	Description
31–13 -	This field is reserved. Reserved.
12 IDMAC_WM_ EN_44	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
11 IDMAC_WM_ EN_43	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled

Table continues on the next page...

IPUx_IDMAC_WM_EN_2 field descriptions (continued)

Field	Description
10 IDMAC_WM_EN_42	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
9 IDMAC_WM_EN_41	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
8 IDMAC_WM_EN_40	IDMAC Watermark enable bit [i] 0 IDMAC channel [i] watermark feature is disabled 1 IDMAC channel [i] watermark feature is enabled
-	This field is reserved. Reserved.

37.5.95 IDMAC Channel Lock Enable 1Register (IPUx_IDMAC_LOCK_EN_1)

Address: Base address + 8024h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved										IDMAC_LOCK_EN_28		IDMAC_LOCK_EN_27		IDMAC_LOCK_EN_23	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	IDMAC_LOCK_EN_22		IDMAC_LOCK_EN_21		IDMAC_LOCK_EN_20		IDMAC_LOCK_EN_15		IDMAC_LOCK_EN_14		IDMAC_LOCK_EN_12		IDMAC_LOCK_EN_11		IDMAC_LOCK_EN_5	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IDMAC_LOCK_EN_1 field descriptions

Field	Description
31–22 -	This field is reserved. Reserved.
21–20 IDMAC_LOCK_EN_28	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
19–18 IDMAC_LOCK_EN_27	IDMAC lock bits for channel [i]

Table continues on the next page...

IPUx_IDMAC_LOCK_EN_1 field descriptions (continued)

Field	Description
	00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
17–16 IDMAC_LOCK_EN_23	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
15–14 IDMAC_LOCK_EN_22	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
13–12 IDMAC_LOCK_EN_21	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
11–10 IDMAC_LOCK_EN_20	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
9–8 IDMAC_LOCK_EN_15	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
7–6 IDMAC_LOCK_EN_14	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.

Table continues on the next page...

IPUx_IDMAC_LOCK_EN_1 field descriptions (continued)

Field	Description
5–4 IDMAC_LOCK_EN_12	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
3–2 IDMAC_LOCK_EN_11	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
IDMAC_LOCK_EN_5	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.

37.5.96 IDMAC Channel Lock Enable 2Register (IPUx_IDMAC_LOCK_EN_2)

Address: Base address + 8028h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				IDMAC_LOCK_50		IDMAC_LOCK_49		IDMAC_LOCK_48		IDMAC_LOCK_47		IDMAC_LOCK_46		IDMAC_LOCK_45	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IDMAC_LOCK_EN_2 field descriptions

Field	Description
31–12 -	This field is reserved. Reserved
11–10 IDMAC_LOCK_50	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request.

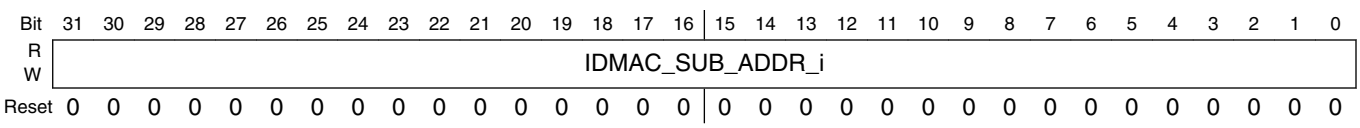
Table continues on the next page...

IPUx_IDMAC_LOCK_EN_2 field descriptions (continued)

Field	Description
	10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
9–8 IDMAC_LOCK_49	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
7–6 IDMAC_LOCK_48	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
5–4 IDMAC_LOCK_47	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
3–2 IDMAC_LOCK_46	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.
IDMAC_LOCK_45	IDMAC lock bits for channel [i] 00 The lock feature is disabled. The IDMAC will generate one AXI burst upon the assertion of the DMA request. 01 The IDMAC will generate two AXI bursts upon the assertion of the DMA request. 10 The IDMAC will generate four AXI bursts upon the assertion of the DMA request. 11 The IDMAC will generate eight AXI bursts upon the assertion of the DMA request.

37.5.97 IDMAC Channel Alternate Address 0 Register (IPUx_IDMAC_SUB_ADDR_0)

Address: Base address + 802Ch offset

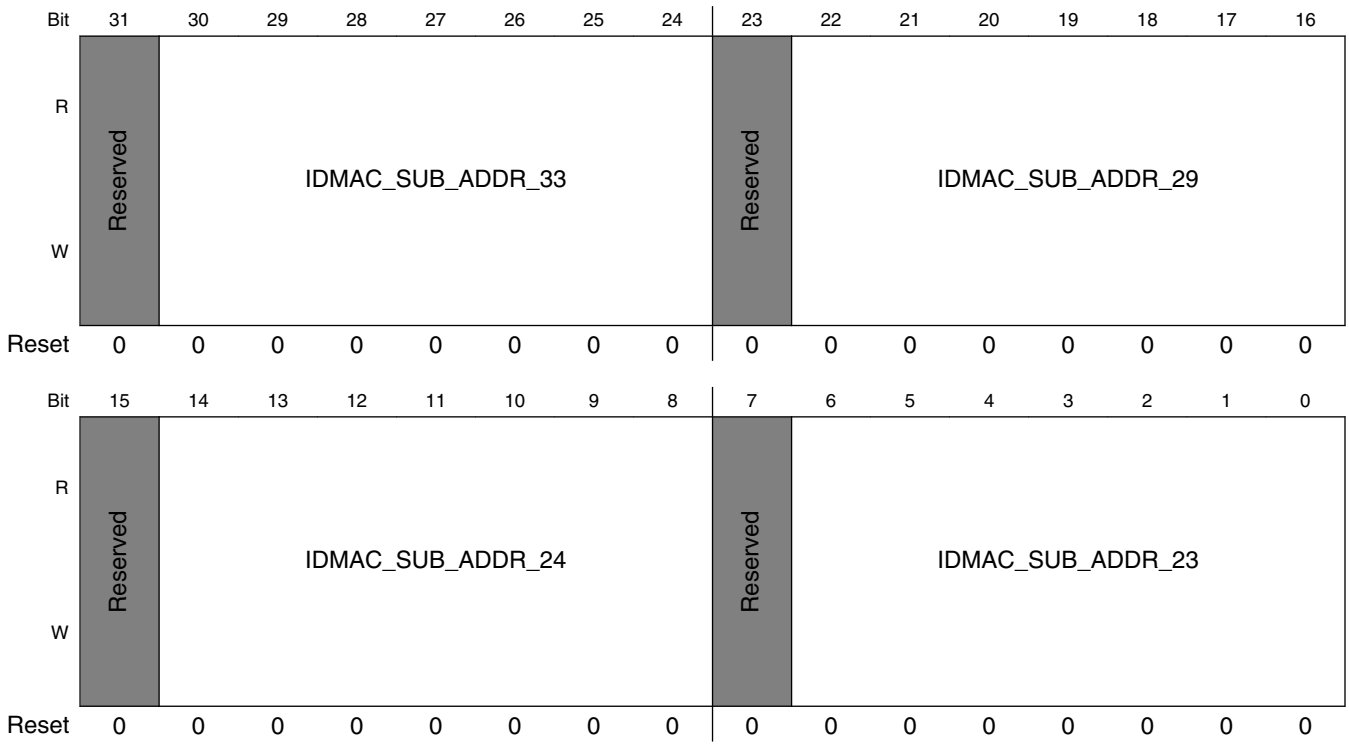


IPUx_IDMAC_SUB_ADDR_0 field descriptions

Field	Description
IDMAC_SUB_ADDR_i	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

37.5.98 IDMAC Channel Alternate Address 1 Register (IPUx_IDMAC_SUB_ADDR_1)

Address: Base address + 8030h offset



IPUx_IDMAC_SUB_ADDR_1 field descriptions

Field	Description
31 -	This field is reserved. Reserved.
30–24 IDMAC_SUB_ADDR_33	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
23 -	This field is reserved. Reserved.
22–16 IDMAC_SUB_ADDR_29	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

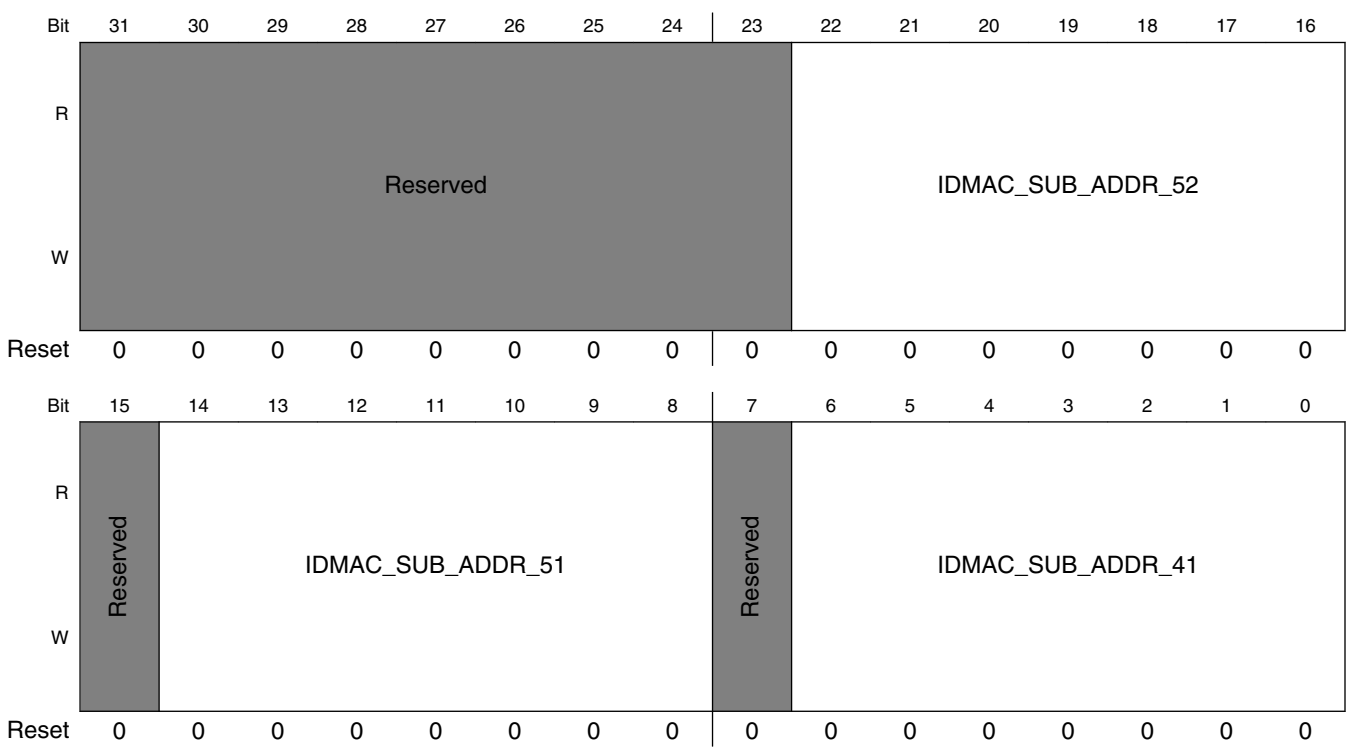
Table continues on the next page...

IPUx_IDMAC_SUB_ADDR_1 field descriptions (continued)

Field	Description
15 -	This field is reserved. Reserved.
14–8 IDMAC_SUB_ADDR_24	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
7 -	This field is reserved. Reserved.
IDMAC_SUB_ADDR_23	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

37.5.99 IDMAC Channel Alternate Address 2 Register (IPUx_IDMAC_SUB_ADDR_2)

Address: Base address + 8034h offset



IPUx_IDMAC_SUB_ADDR_2 field descriptions

Field	Description
31–23 -	This field is reserved. Reserved.

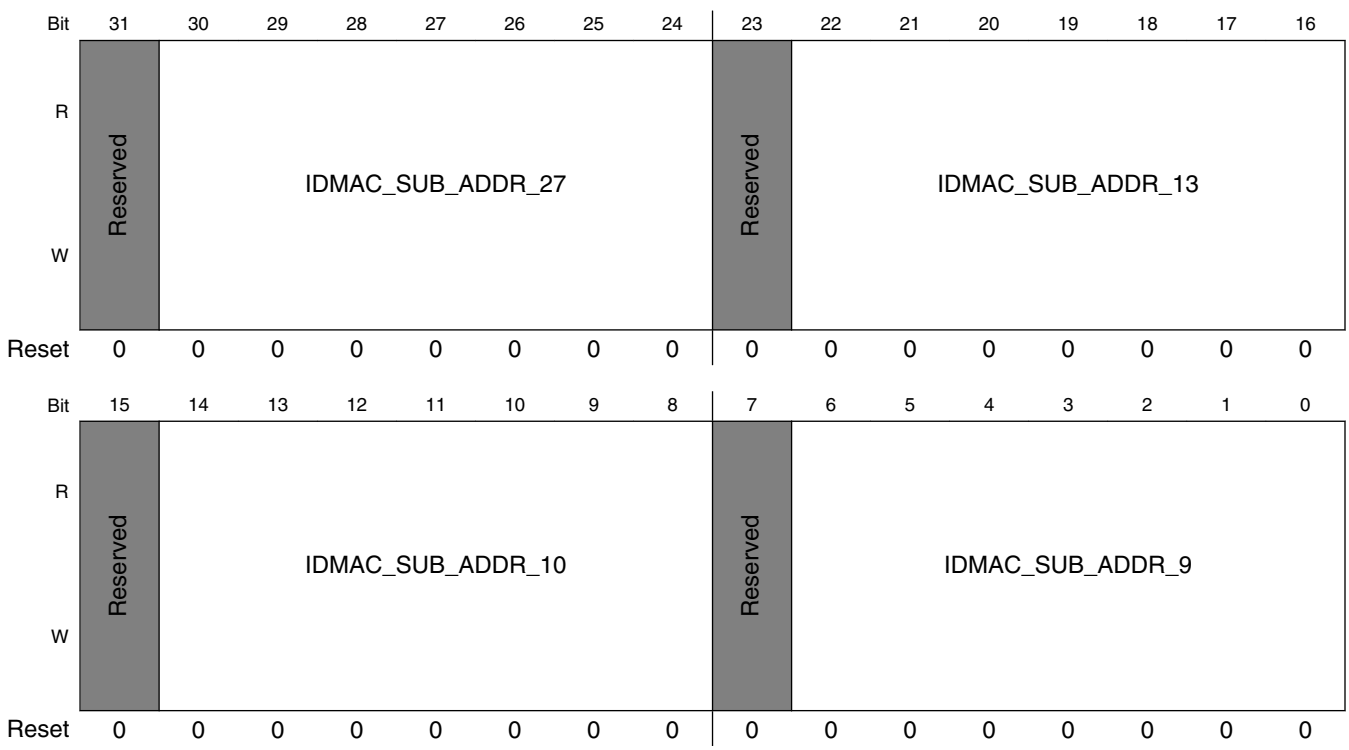
Table continues on the next page...

IPUx_IDMAC_SUB_ADDR_2 field descriptions (continued)

Field	Description
22–16 IDMAC_SUB_ ADDR_52	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
15 -	This field is reserved. Reserved.
14–8 IDMAC_SUB_ ADDR_51	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
7 -	This field is reserved. Reserved.
IDMAC_SUB_ ADDR_41	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

37.5.100 IDMAC Channel Alternate Address 3 Register (IPUx_IDMAC_SUB_ADDR_3)

Address: Base address + 8038h offset

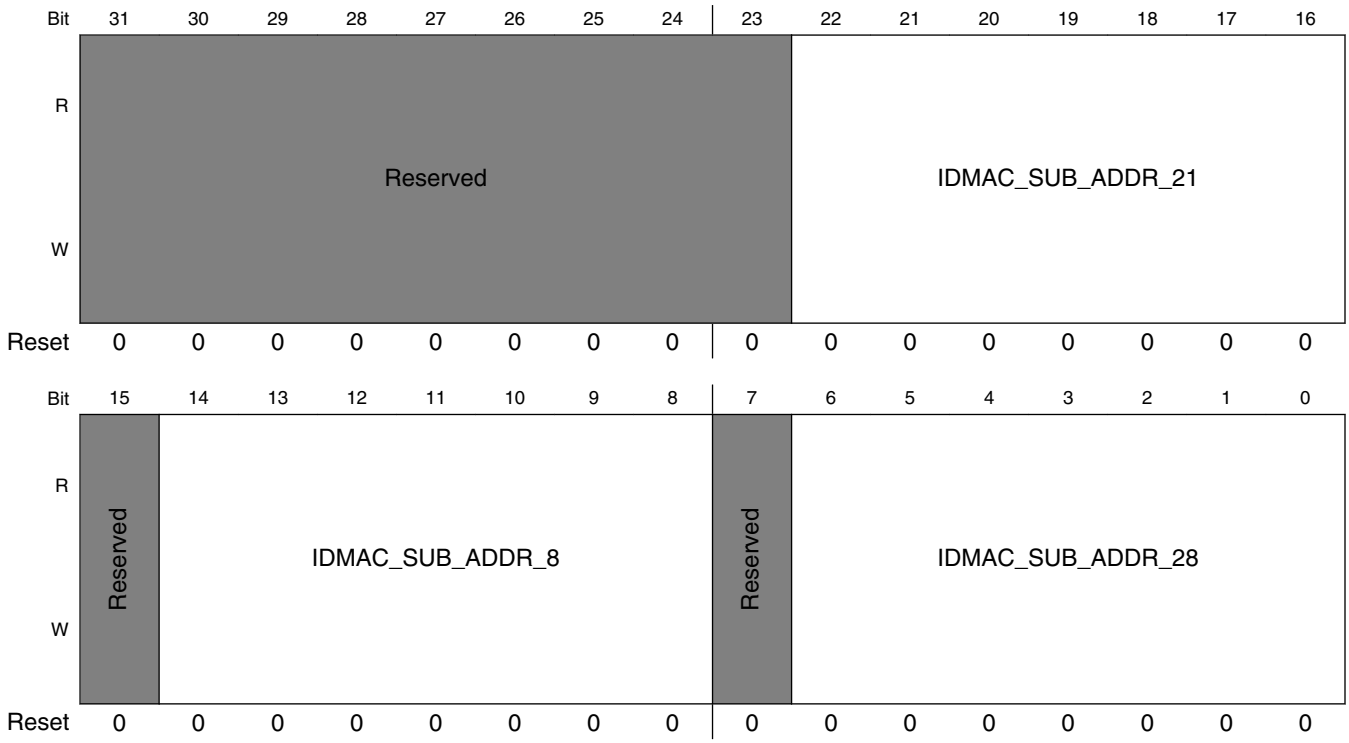


IPUx_IDMAC_SUB_ADDR_3 field descriptions

Field	Description
31 -	This field is reserved. Reserved.
30–24 IDMAC_SUB_ADDR_27	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
23 -	This field is reserved. Reserved.
22–16 IDMAC_SUB_ADDR_13	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
15 -	This field is reserved. Reserved.
14–8 IDMAC_SUB_ADDR_10	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
7 -	This field is reserved. Reserved.
IDMAC_SUB_ADDR_9	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

37.5.101 IDMAC Channel Alternate Address 4 Register (IPUx_IDMAC_SUB_ADDR_4)

Address: Base address + 803Ch offset

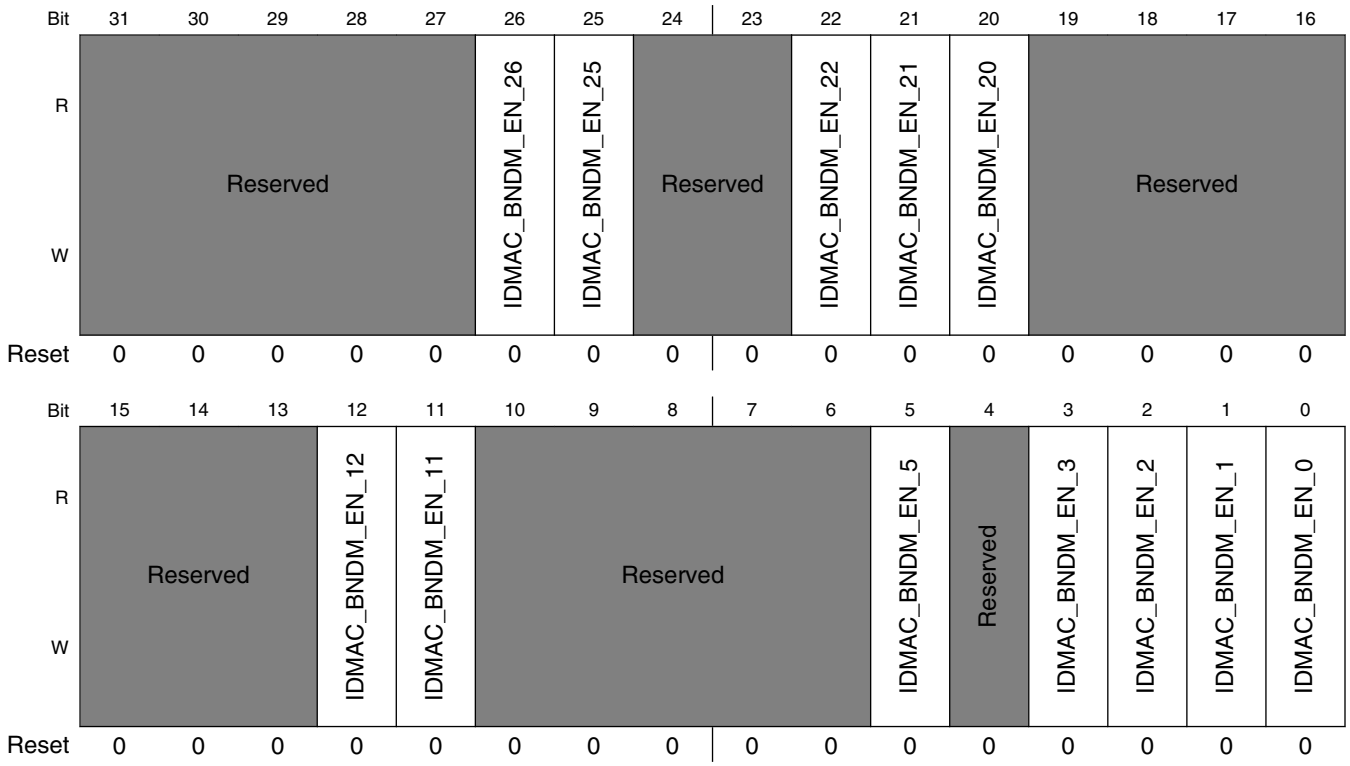


IPUx_IDMAC_SUB_ADDR_4 field descriptions

Field	Description
31–23 -	This field is reserved. Reserved.
22–16 IDMAC_SUB_ADDR_21	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
15 -	This field is reserved. Reserved.
14–8 IDMAC_SUB_ADDR_8	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.
7 -	This field is reserved. Reserved.
IDMAC_SUB_ADDR_28	The CPMEM alternative entry [i] holds the parameters of the channel that is number appears in IDMAC_SUB_ADDR_i. The user must set each IDMAC_SUB_ADDR_i field to a unique channel. Alternative CPMEM entry is relevant only for channels supporting alternate flows.

37.5.102 IDMAC Band Mode Enable 1 Register (IPUx_IDMAC_BNDM_EN_1)

Address: Base address + 8040h offset



IPUx_IDMAC_BNDM_EN_1 field descriptions

Field	Description
31–27 -	This field is reserved. Reserved.
26 IDMAC_BNDM_EN_26	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
25 IDMAC_BNDM_EN_25	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode

Table continues on the next page...

IPUx_IDMAC_BNDM_EN_1 field descriptions (continued)

Field	Description
24–23 -	This field is reserved. Reserved.
22 IDMAC_BNDM_EN_22	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
21 IDMAC_BNDM_EN_21	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
20 IDMAC_BNDM_EN_20	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
19–13 -	This field is reserved. Reserved.
12 IDMAC_BNDM_EN_12	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
11 IDMAC_BNDM_EN_11	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
10–6 -	This field is reserved. Reserved.
5 IDMAC_BNDM_EN_5	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration.

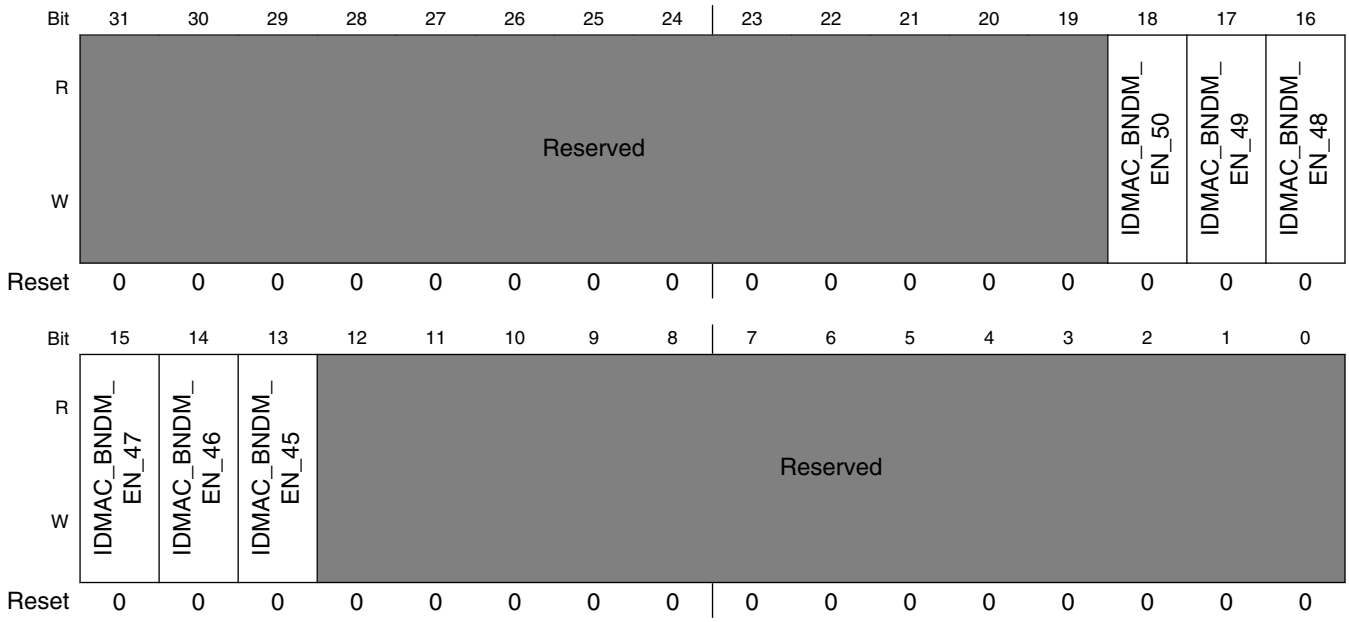
Table continues on the next page...

IPUx_IDMAC_BNDM_EN_1 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
4 -	This field is reserved. Reserved.
3 IDMAC_BNDM_EN_3	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
2 IDMAC_BNDM_EN_2	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
1 IDMAC_BNDM_EN_1	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
0 IDMAC_BNDM_EN_0	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode

37.5.103 IDMAC Band Mode Enable 2 Register (IPUx_IDMAC_BNDM_EN_2)

Address: Base address + 8044h offset



IPUx_IDMAC_BNDM_EN_2 field descriptions

Field	Description
31–19 -	This field is reserved. Reserved.
18 IDMAC_BNDM_EN_50	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
17 IDMAC_BNDM_EN_49	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
16 IDMAC_BNDM_EN_48	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration.

Table continues on the next page...

IPUx_IDMAC_BNDM_EN_2 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
15 IDMAC_BNDM_EN_47	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
14 IDMAC_BNDM_EN_46	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
13 IDMAC_BNDM_EN_45	IDMAC Band Mode Enable bit [i] This bit controls if the channel currently works in band mode. When alternate flow is running via this channel, both the main flow and the alternate flow should have the same band mode configuration. 0 IDMAC channel [i] is not in band mode 1 IDMAC channel [i] is in band mode
-	This field is reserved. Reserved.

37.5.104 IDMAC Scroll Coordinations Register (IPUx_IDMAC_SC_CORD)

Address: Base address + 8048h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				SX0												Reserved					SY0											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IDMAC_SC_CORD field descriptions

Field	Description
31–28 -	This field is reserved. Reserved, should be cleared.
27–16 SX0	Scroll X coordination This field indicates the X coordinate of the scroll. This parameter has an affect on continuos scroll mode only. Units are pixels. When the alpha buffer resides in a separate buffer and the alpha is different than 8 BPP this field should be 0.

Table continues on the next page...

IPUx_IDMAC_SC_CORD field descriptions (continued)

Field	Description
15–11 -	This field is reserved. Reserved, should be cleared.
SY0	Scroll Y coordination This field indicates the Y coordinate of the scroll. This parameter has an affect on continuos scroll mode only. Units are pixels. When the alpha buffer resides in a separate buffer and the alpha is different than 8 BPP this field should be 0.

37.5.105 IDMAC Scroll Coordinations Register 1 (IPUx_IDMAC_SC_CORD_1)

Address: Base address + 804Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				SX1												Reserved					SY1											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

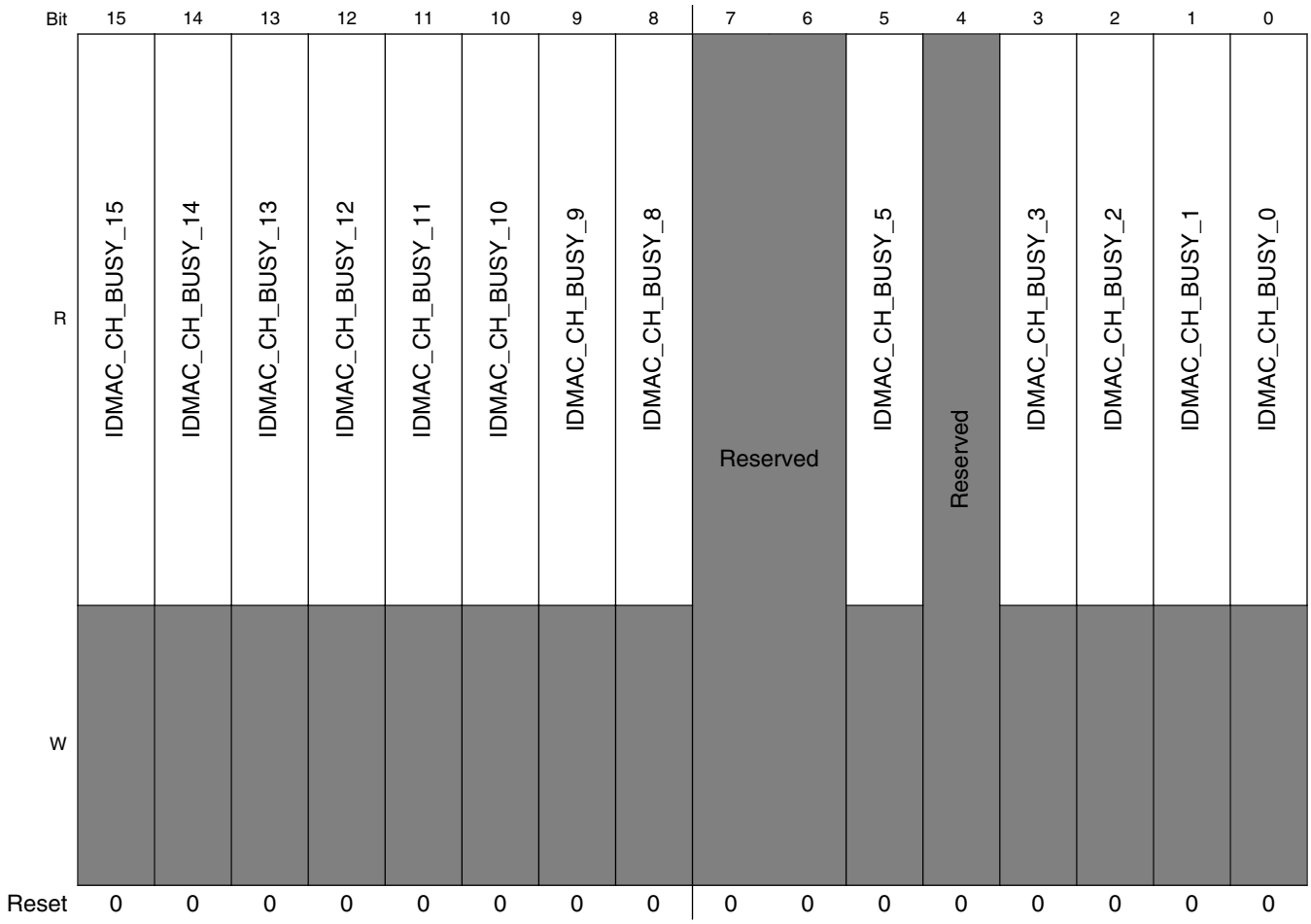
IPUx_IDMAC_SC_CORD_1 field descriptions

Field	Description
31–28 -	This field is reserved. Reserved, should be cleared.
27–16 SX1	Scroll X coordination (2nd set) This field indicates the X coordinate of the scroll. This parameter has an affect on continuos scroll mode only. Units are pixels. When the alpha buffer resides in a separate buffer and the alpha is different than 8 BPP this field should be 0.
15–11 -	This field is reserved. Reserved, should be cleared.
SY1	Scroll Y coordination (2nd set) This field indicates the Y coordinate of the scroll. This parameter has an affect on continuos scroll mode only. Units are pixels. When the alpha buffer resides in a separate buffer and the alpha is different than 8 BPP this field should be 0.

37.5.106 IDMAC Channel Busy 1 Register (IPUx_IDMAC_CH_BUSY_1)

Address: Base address + 8100h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	IDMAC_CH_BUSY_	Reserved	IDMAC_CH_BUSY_29	IDMAC_CH_BUSY_28	IDMAC_CH_BUSY_27	IDMAC_CH_BUSY_26	IDMAC_CH_BUSY_25	IDMAC_CH_BUSY_24	IDMAC_CH_BUSY_23	IDMAC_CH_BUSY_22	IDMAC_CH_BUSY_21	IDMAC_CH_BUSY_20	Reserved	IDMAC_CH_BUSY_18	IDMAC_CH_BUSY_17	Reserved
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_IDMAC_CH_BUSY_1 field descriptions

Field	Description
31 IDMAC_CH_BUSY_	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
30 -	This field is reserved. Reserved.
29 IDMAC_CH_BUSY_29	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
28 IDMAC_CH_BUSY_28	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC.

Table continues on the next page...

IPUx_IDMAC_CH_BUSY_1 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
27 IDMAC_CH_ BUSY_27	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
26 IDMAC_CH_ BUSY_26	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
25 IDMAC_CH_ BUSY_25	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
24 IDMAC_CH_ BUSY_24	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
23 IDMAC_CH_ BUSY_23	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
22 IDMAC_CH_ BUSY_22	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
21 IDMAC_CH_ BUSY_21	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy

Table continues on the next page...

IPUx_IDMAC_CH_BUSY_1 field descriptions (continued)

Field	Description
20 IDMAC_CH_BUSY_20	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
19 -	This field is reserved. Reserved.
18 IDMAC_CH_BUSY_18	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
17 IDMAC_CH_BUSY_17	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
16 -	This field is reserved. Reserved.
15 IDMAC_CH_BUSY_15	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
14 IDMAC_CH_BUSY_14	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
13 IDMAC_CH_BUSY_13	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
12 IDMAC_CH_BUSY_12	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC.

Table continues on the next page...

IPUx_IDMAC_CH_BUSY_1 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
11 IDMAC_CH_BUSY_11	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
10 IDMAC_CH_BUSY_10	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
9 IDMAC_CH_BUSY_9	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
8 IDMAC_CH_BUSY_8	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
7–6 -	This field is reserved. Reserved.
5 IDMAC_CH_BUSY_5	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
4 -	This field is reserved. Reserved.
3 IDMAC_CH_BUSY_3	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
2 IDMAC_CH_BUSY_2	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC.

Table continues on the next page...

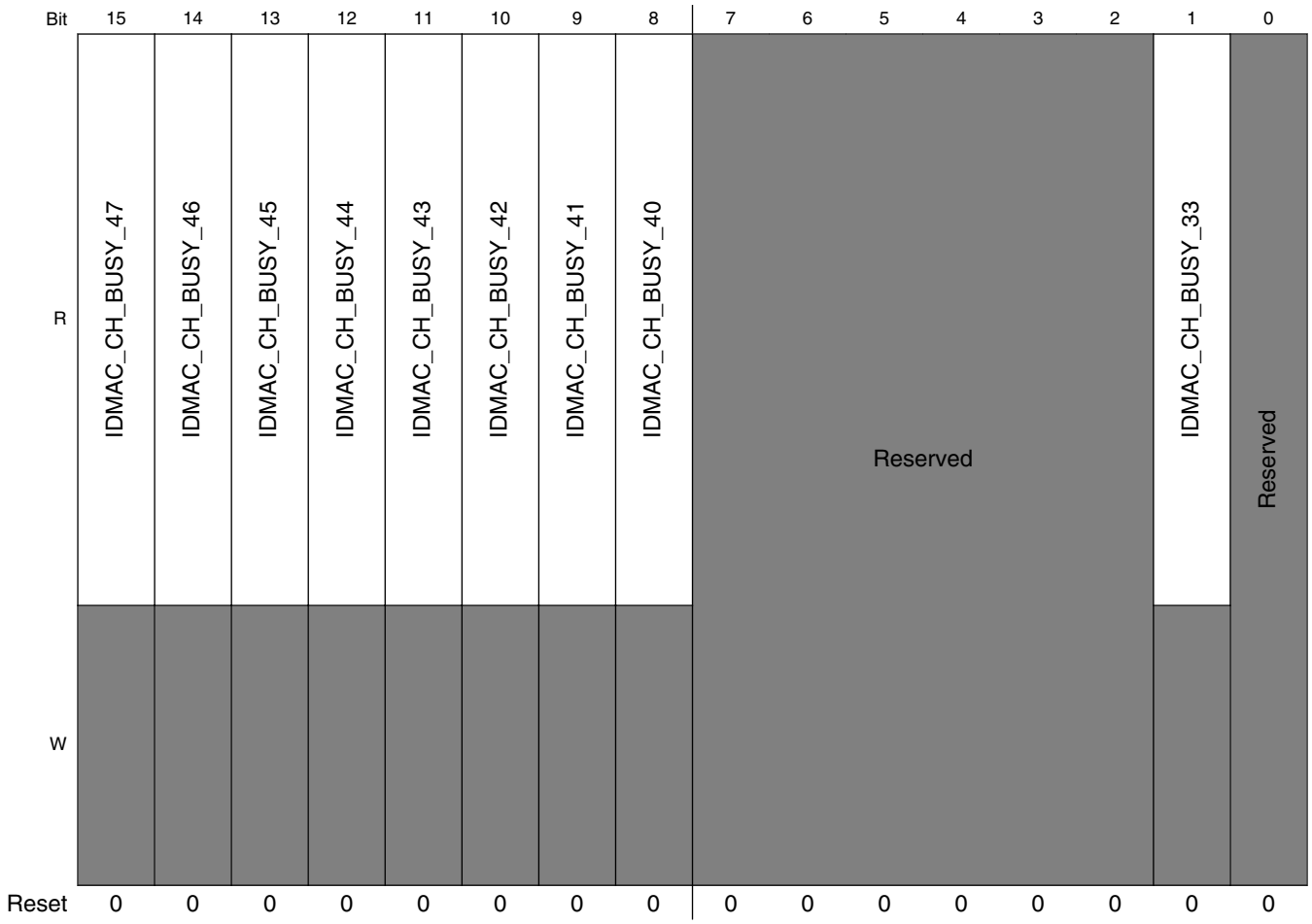
IPUx_IDMAC_CH_BUSY_1 field descriptions (continued)

Field	Description
	This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
1 IDMAC_CH_ BUSY_1	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
0 IDMAC_CH_ BUSY_0	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy

37.5.107 IDMAC Channel Busy 2 Register (IPUx_IDMAC_CH_BUSY_2)

Address: Base address + 8104h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved											IDMAC_CH_BUSY_52	IDMAC_CH_BUSY_51	IDMAC_CH_BUSY_50	IDMAC_CH_BUSY_49	IDMAC_CH_BUSY_48
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



IPUx_IDMAC_CH_BUSY_2 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved.
20 IDMAC_CH_ BUSY_52	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
19 IDMAC_CH_ BUSY_51	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
18 IDMAC_CH_ BUSY_50	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC.

Table continues on the next page...

IPUx_IDMAC_CH_BUSY_2 field descriptions (continued)

Field	Description
	0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
17 IDMAC_CH_BUSY_49	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
16 IDMAC_CH_BUSY_48	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
15 IDMAC_CH_BUSY_47	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
14 IDMAC_CH_BUSY_46	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
13 IDMAC_CH_BUSY_45	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
12 IDMAC_CH_BUSY_44	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
11 IDMAC_CH_BUSY_43	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy

Table continues on the next page...

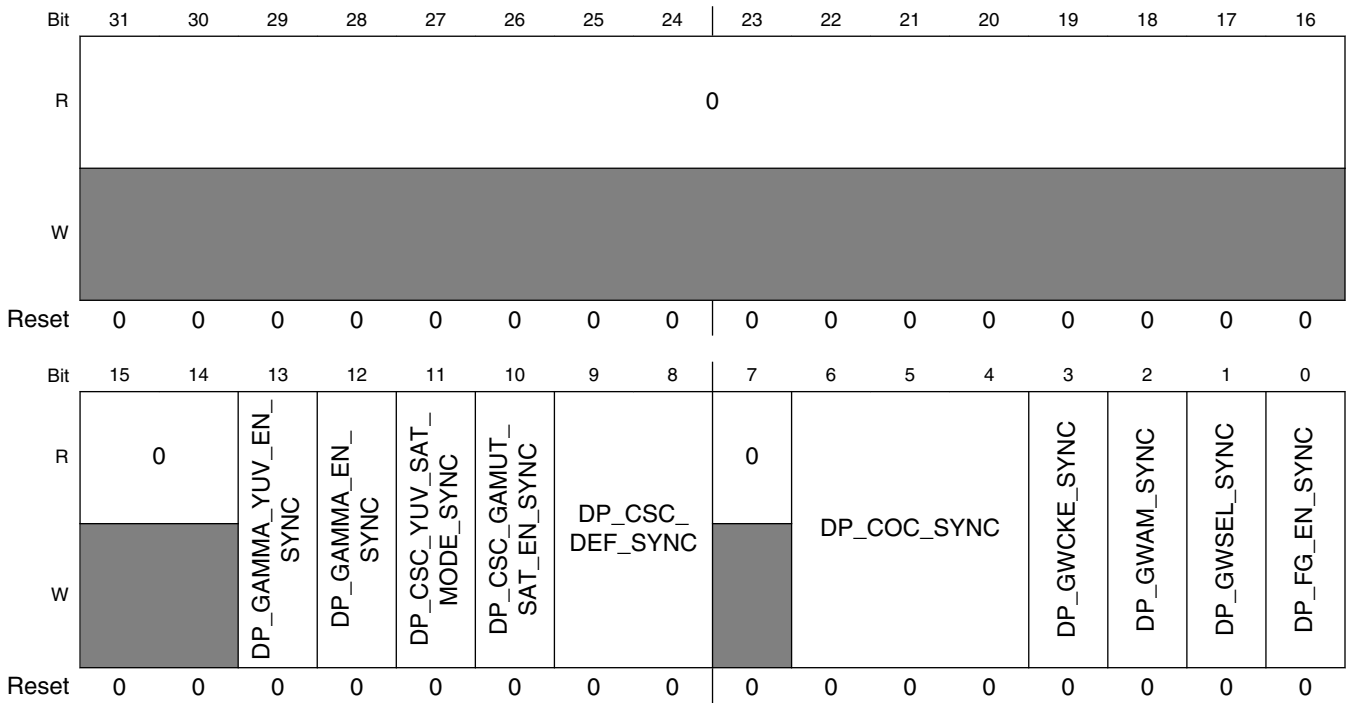
IPUx_IDMAC_CH_BUSY_2 field descriptions (continued)

Field	Description
10 IDMAC_CH_ BUSY_42	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
9 IDMAC_CH_ BUSY_41	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
8 IDMAC_CH_ BUSY_40	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
7–2 -	This field is reserved. Reserved.
1 IDMAC_CH_ BUSY_33	IDMAC Channel busy bit [i] This bit indicates if the channel is currently served by the IDMAC. This bit is self cleared by the IDMAC. 0 IDMAC channel [i] is not busy 1 IDMAC channel [i] is busy
0 -	This field is reserved. Reserved.

37.5.108 DP Common Configuration Sync Flow Register (IPUx_DP_COM_CONF_SYNC)

This register contains common configuration parameters for the DP.

Address: Base address + 1_8000h offset



IPUx_DP_COM_CONF_SYNC field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13 DP_GAMMA_YUV_EN_SYNC	GAMMA's YUV mode enable for sync flow 0 YUV mode is OFF. 1 YUV mode is ON.
12 DP_GAMMA_EN_SYNC	GAMMA_EN - Gamma correction block enable bit 0 disable 1 enable
11 DP_CSC_YUV_SAT_MODE_SYNC	CSC_YUV_SAT_MODE YUV saturation mode for color space conversion 0 Y/U/V range 0 -255 1 Y range 16-235, U/V range 16-240

Table continues on the next page...

IPUx_DP_COM_CONF_SYNC field descriptions (continued)

Field	Description
10 DP_CSC_ GAMUT_SAT_ EN_SYNC	CSC_GAMUT_SAT_EN Indicate if GAMUT saturation is enabled 0 disable GAMUT mapping 1 enable GAMUT mapping
9–8 DP_CSC_DEF_ SYNC	CSC_DEF Enable or disable Color Space Conversion. 00 CSC disable 01 CSC enable after combining 10 CSC enable before combining on BG channel 11 CSC enable before combining on FG channel
7 Reserved	This read-only field is reserved and always has the value 0.
6–4 DP_COC_SYNC	COC - Cursor Operation Control Controls the format of the cursor and the type of arithmetic operations 000 Transparent, cursor is disabled. 001 Full cursor. 010 Reversed cursor. 011 AND between full plane and cursor. 100 Reserved 101 OR between full plane and cursor. 110 XOR between full plane and cursor. 111 Reserved.
3 DP_GWCKE_ SYNC	GWCKE - Graphic Window Color Keying Enable Enable or disable graphic window color keying. 1 Enable color keying of graphic window 0 Disable color keying of graphic window
2 DP_GWAM_ SYNC	GWAM - Graphic Window Alpha Mode Select the use of Alpha to be global or local. 1 Global Alpha. 0 Local Alpha.
1 DP_GWSEL_ SYNC	GWSEL - Graphic Window Select Select graphic window to be on partial plane or full plane. 1 Graphic window is partial plane. 0 Graphic window is full plane.
0 DP_FG_EN_ SYNC	FG_EN - partial plane Enable. This bit enables the partial plane channel. 1 partial plane channel is enabled. 0 partial plane channel is disabled.

37.5.109 DP Graphic Window Control Sync Flow Register (IPUx_DP_Graph_Wind_CTRL_SYNC)

This register contains common configuration parameters for the DP.

Address: Base address + 1_8004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_GWAV_SYNC								DP_GWCKR_SYNC								DP_GWCKG_SYNC								DP_GWCKB_SYNC							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_Graph_Wind_CTRL_SYNC field descriptions

Field	Description
31–24 DP_GWAV_SYNC	GWAV - Graphic Window Alpha Value Defines the alpha value of graphic window used for alpha blending between graphic window and full plane. The Value the number that writing to GWAV register. The Actual Value the number, that insert to calculation in Combining Module. If Value = < 0.5- 1/256 (01111111) then Actual Value = Value. If Value >= 0.5 (10000000), then Actual Value = Value + 1/256. 00000000 Actual value is 00000000; Graphic window totally opaque i.e. overlay on LCD screen 01111111 Actual value is 01111111; 10000000 Actual value is 10000001 10000001 Actual value is 10000010 11111110 Actual value is 11111111 11111111 Actual value is 100000000;Graphic window totally transparent i.e. not displayed on LCD screen
23–16 DP_GWCKR_SYNC	GWCKR - Graphic Window Color Keying Red Component Defines the red component of graphic window color keying. 00000000 No red 11111111 Full red
15–8 DP_GWCKG_SYNC	GWCKG - Graphic Window Color Keying Green Component Defines the green component of graphic window color keying. 00000000 No Green 11111111 Full Green
DP_GWCKB_SYNC	GWCKB - Graphic Window Color Keying Blue Component Defines the blue component of graphic window color keying. 00000000 No blue 11111111 Full blue

37.5.110 DP Partial Plane Window Position Sync Flow Register (IPUx_DP_FG_POS_SYNC)

The DP partial plane Window Position Register is used to determine the starting position of the partial plane window relative to BG window position.

This Register is used for the synchronous flow only.

Address: Base address + 1_8008h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					DP_FGXP_SYNC											0					DP_FGYP_SYNC										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_FG_POS_SYNC field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 DP_FGXP_ SYNC	FGXP partial plane Window X Position. partial plane Window X Position - Specifies the number of pixels between the start of full plane window X position and the beginning of the first data of new line.
15–11 Reserved	This read-only field is reserved and always has the value 0.
DP_FGYP_ SYNC	FGYP partial plane window Y position partial plane Window Y Position - Specifies the number of lines between the start of full plane windows Y position and the beginning of the first data.

37.5.111 DP Cursor Position and Size Sync Flow Register (IPUx_DP_CUR_POS_SYNC)

The LCD Cursor Position Register is used to determine the starting position of the cursor relative to BG windows position.

Address: Base address + 1_800Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DP_CYP_SYNC					DP_CYH_SYNC											DP_CXP_SYNC					DP_CXW_SYNC										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_POS_SYNC field descriptions

Field	Description
31–27 DP_CYP_SYNC	CYP - Cursor Y Position Represents the cursors vertical starting position Y in pixel count (from 0 to CYH).Live View Resolution Mode.
26–16 DP_CYH_SYNC	CYH - Cursor Height Specifies the height of the hardware cursor in pixels.
15–11 DP_CXP_SYNC	CXP - Cursor X Position Represents the cursors horizontal starting position X in pixel count (from 0 to CXW).
DP_CXW_SYNC	CXW - Cursor Width. Specifies the width of the hardware cursor in pixels.

37.5.112 DP Color Cursor Mapping Sync Flow Register (IPUx_DP_CUR_MAP_SYNC)

The LCD Color Cursor Mapping Register defines the color of the cursor in passive or TFT color modes.

Address: Base address + 1_8010h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								DP_CUR_COL_B_SYNC								DP_CUR_COL_G_SYNC								DP_CUR_COL_R_SYNC							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_MAP_SYNC field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 DP_CUR_COL_B_SYNC	CUR_COL_B - Cursor Blue Field Defines the Blue component of the cursor color in color mode 00000000 No Blue. 11111111 Full Blue.
15–8 DP_CUR_COL_G_SYNC	CUR_COL_G - Cursor Green Field Defines the Green component of the cursor color in color mode 00000000 No Green. 11111111 Full Green.
DP_CUR_COL_R_SYNC	CUR_COL_B - Cursor Red Field Defines the Red component of the cursor color in color mode

Table continues on the next page...

IPUx_DP_CUR_MAP_SYNC field descriptions (continued)

Field	Description
	00000000 No Red. 11111111 Full Red.

37.5.113 DP Gamma Constants Sync Flow Register i (IPUx_DP_GAMMA_C_SYNC_i)

This registers contains CONSTANTi parameters used for gamma correction inside the display processor (DP).

Address: Base address + 1_8014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							DP_GAMMA_C_SYNC_2i_1									0							DP_GAMMA_C_SYNC_2i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_C_SYNC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 DP_GAMMA_C_SYNC_2i_1	CONSTANTi+1 parameter of Gamma Correction.
15–9 Reserved	This read-only field is reserved and always has the value 0.
DP_GAMMA_C_SYNC_2i	CONSTANTi parameter of Gamma Correction.

37.5.114 DP Gamma Correction Slope Sync Flow Register i (IPUx_DP_GAMMA_S_SYNC_i)

This registers contains SLOPEi parameters used for Gamma Correction inside the display processor (DP).

Address: Base address + 1_8034h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DP_GAMMA_S_SYNC_4i_3								DP_GAMMA_S_SYNC_4i_2								DP_GAMMA_S_SYNC_4i_1								DP_GAMMA_S_SYNC_4i							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_S_SYNC_i field descriptions

Field	Description
31–24 DP_GAMMA_S_SYNC_4i_3	SLOPE<4*i+3> parameter of Gamma Correction.
23–16 DP_GAMMA_S_SYNC_4i_2	SLOPE<4*i+2> parameter of Gamma Correction.
15–8 DP_GAMMA_S_SYNC_4i_1	SLOPE<4*i+1> parameter of Gamma Correction.
DP_GAMMA_S_SYNC_4i	SLOPE<4*i> parameter of Gamma Correction.

37.5.115 DP Color Space Conversion Control Sync Flow Registers (IPUx_DP_CSCA_SYNC_i)

Address: Base address + 1_8044h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0						DP_CSC_A_SYNC_2i_1										0						DP_CSC_A_SYNC_2i									
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CSCA_SYNC_i field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25–16 DP_CSC_A_SYNC_2i_1	A<2*i+1> parameter of color conversion
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A_SYNC_2i	A<2*i> parameter of color conversion.

37.5.116 DP Color Conversion Control Sync Flow Register 0 (IPUx_DP_SCS_SYNC_0)

Address: Base address + 1_8054h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_S0_SYNC		DP_CSC_B0_SYNC													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0						DP_CSC_A8_SYNC									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_SCS_SYNC_0 field descriptions

Field	Description
31–30 DP_CSC_S0_SYNC	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B0_SYNC	B0 parameter of color conversion.
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A8_SYNC	A9 parameter of color conversion.

37.5.117 DP Color Conversion Control Sync Flow Register 1 (IPUx_DP_SCS_SYNC_1)

Address: Base address + 1_8058h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_S2_SYNC		DP_CSC_B2_SYNC													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_CSC_S1_SYNC		DP_CSC_B1_SYNC													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_SCS_SYNC_1 field descriptions

Field	Description
31–30 DP_CSC_S2_SYNC	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B2_SYNC	B0 parameter of color conversion.
15–14 DP_CSC_S1_SYNC	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
DP_CSC_B1_SYNC	B0 parameter of color conversion.

37.5.118 DP Cursor Position and Size Alternate Register (IPUx_DP_CUR_POS_ALT)

The LCD Cursor Position Register is used to determine the starting position of the cursor relative to BG windows position for the alternative flow.

Address: Base address + 1_805Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_CYP_SYNC_ALT					DP_CYH_SYNC_ALT											DP_CXP_SYNC_ALT					DP_CXW_SYNC_ALT										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_POS_ALT field descriptions

Field	Description
31–27 DP_CYP_SYNC_ALT	CYP_ALT - Cursor Y Position Represents the cursors vertical starting position Y in pixel count (from 0 to CYH).Live View Resolution Mode for the alternative flow.
26–16 DP_CYH_SYNC_ALT	CYH_ALT - Cursor Height Specifies the height of the hardware cursor in pixels.
15–11 DP_CXP_SYNC_ALT	CXP_ALT - Cursor X Position Represents the cursors horizontal starting position X in pixel count (from 0 to CXW) for the alternative flow.

Table continues on the next page...

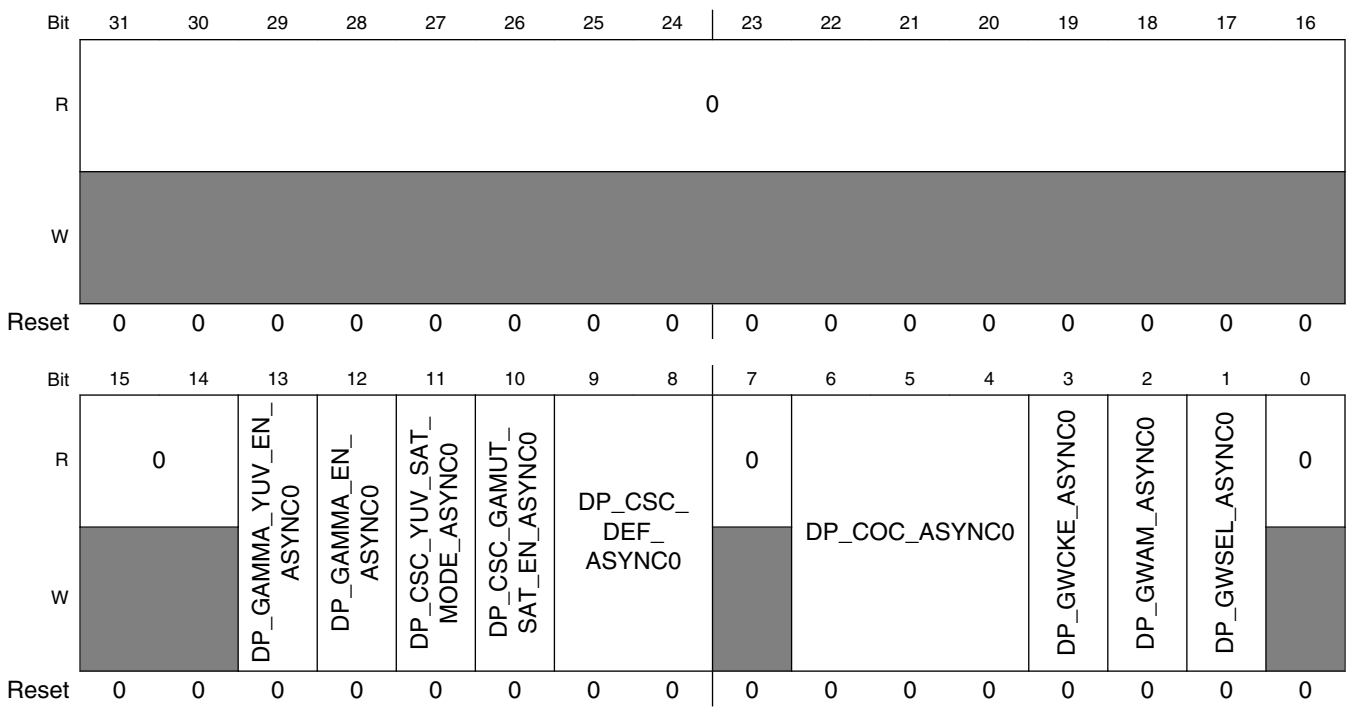
IPUx_DP_CUR_POS_ALT field descriptions (continued)

Field	Description
DP_CXW_ SYNC_ALT	CXW_ALT - Cursor Width. Specifies the width of the hardware cursor in pixels for the alternative flow.

37.5.119 DP Common Configuration Async 0 Flow Register (IPUx_DP_COM_CONF_ASYNC0)

This register contains common configuration parameters for the DP.

Address: Base address + 1_8060h offset



IPUx_DP_COM_CONF_ASYNC0 field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13 DP_GAMMA_ YUV_EN_ ASYNC0	GAMMA's YUV mode enable for async flow 0 0 YUV mode is OFF. 1 YUV mode is ON.
12 DP_GAMMA_ EN_ASYNC0	GAMMA_EN - Gamma correction block enable bit

Table continues on the next page...

IPUx_DP_COM_CONF_ASYNC0 field descriptions (continued)

Field	Description
	0 disable 1 enable
11 DP_CSC_YUV_SAT_MODE_ASYNC0	CSC_YUV_SAT_MODE YUV saturation mode for color space conversion 0 Y/U/V range 0 -255 1 Y range 16-235, U/V range 16-240
10 DP_CSC_GAMUT_SAT_EN_ASYNC0	CSC_GAMUT_SAT_EN Indicate if GAMUT saturation is enabled 0 disable GAMUT mapping 1 enable GAMUT mapping
9–8 DP_CSC_DEF_ASYNC0	CSC_DEF Enable or disable Color Space Conversion. 00 CSC disable 01 CSC enable after combining 10 CSC enable before combining on BG channel 11 CSC enable before combining on FG channel
7 Reserved	This read-only field is reserved and always has the value 0.
6–4 DP_COC_ASYNC0	COC - Cursor Operation Control Controls the format of the cursor and the type of arithmetic operations 000 Transparent, cursor is disabled. 001 Full cursor. 010 Reversed cursor. 011 AND between full plane and cursor. 100 Reserved 101 OR between full plane and cursor. 110 XOR between full plane and cursor. 111 Reserved
3 DP_GWCKE_ASYNC0	GWCKE - Graphic Window Color Keying Enable Enable or disable graphic window color keying. 1 Enable color keying of graphic window 0 Disable color keying of graphic window
2 DP_GWAM_ASYNC0	GWAM - Graphic Window Alpha Mode Select the use of Alpha to be global or local. 1 Global Alpha. 0 Local Alpha.
1 DP_GWSEL_ASYNC0	GWSEL - Graphic Window Select Select graphic window to be on partial plane or full plane.

Table continues on the next page...

IPUx_DP_COM_CONF_ASYNC0 field descriptions (continued)

Field	Description
	1 Graphic window is partial plane. 0 Graphic window is full plane.5
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.120 DP Graphic Window Control Async 0 Flow Register (IPUx_DP_GRAPH_WIND_CTRL_ASYNC0)

This register contains common configuration parameters for the DP.

Address: Base address + 1_8064h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_GWAV_ASYNC0								DP_GWCKR_ASYNC0								DP_GWCKG_ASYNC0								DP_GWCKB_ASYNC0							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GRAPH_WIND_CTRL_ASYNC0 field descriptions

Field	Description
31–24 DP_GWAV_ ASYNC0	GWAV - Graphic Window Alpha Value Defines the alpha value of graphic window used for alpha blending between graphic window and full plane. The Value the number that writing to GWAV register. The Actual Value the number, that insert to calculation in Combining Module. If Value = < 0.5- 1/256 (01111111) then Actual Value = Value. If Value >= 0.5 (10000000), then Actual Value = Value + 1/256. 00000000 Actual value is 00000000; Graphic window totally opaque i.e. overlay on LCD screen 01111111 Actual value is 01111111; 10000000 Actual value is 10000001 10000001 Actual value is 10000010 11111110 Actual value is 11111111 11111111 Actual value is 100000000;Graphic window totally transparent i.e. not displayed on LCD screen
23–16 DP_GWCKR_ ASYNC0	GWCKR - Graphic Window Color Keying Red Component Defines the red component of graphic window color keying. 00000000 No red 11111111 Full red
15–8 DP_GWCKG_ ASYNC0	GWCKG - Graphic Window Color Keying Green Component Defines the green component of graphic window color keying. 00000000 No Green 11111111 Full Green

Table continues on the next page...

IPUx_DP_GRAPH_WIND_CTRL_ASYNC0 field descriptions (continued)

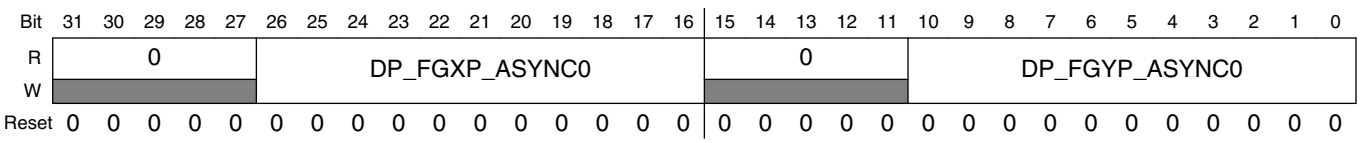
Field	Description
DP_GWCKB_ ASYNC0	GWCKB - Graphic Window Color Keying Blue Component Defines the blue component of graphic window color keying. 00000000 No blue 11111111 Full blue

37.5.121 DP Partial Plane Window Position Async 0 Flow Register (IPUx_DP_FG_POS_ASYNC0)

The DP partial plane Window Position Register is used to determine the starting position of the partial plane window relative to BG window position.

This Register is used for the synchronous flow only.

Address: Base address + 1_8068h offset



IPUx_DP_FG_POS_ASYNC0 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 DP_FGXP_ ASYNC0	FGXP partial plane Window X Position. partial plane Window X Position - Specifies the number of pixels between the start of full plane window X position and the beginning of the first data of new line.
15–11 Reserved	This read-only field is reserved and always has the value 0.
DP_FGYP_ ASYNC0	FGYP partial plane window Y position partial plane Window Y Position - Specifies the number of lines between the start of full plane windows Y position and the beginning of the first data.

37.5.122 DP Cursor Position and Size Async 0 Flow Register (IPUx_DP_CUR_POS_ASYNC0)

The LCD Cursor Position Register is used to determine the starting position of the cursor relative to BG windows position.

Address: Base address + 1_806Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_CYP_ ASYNC0					DP_CYH_ASYNC0											DP_CXP_ ASYNC0					DP_CXW_ASYNC0										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_POS_ASYNC0 field descriptions

Field	Description
31–27 DP_CYP_ ASYNC0	CYP - Cursor Y Position Represents the cursors vertical starting position Y in pixel count (from 0 to CYH).Live View Resolution Mode.
26–16 DP_CYH_ ASYNC0	CYH - Cursor Height Specifies the height of the hardware cursor in pixels.
15–11 DP_CXP_ ASYNC0	CXP - Cursor X Position Represents the cursors horizontal starting position X in pixel count (from 0 to CXW).
DP_CXW_ ASYNC0	CXW - Cursor Width. Specifies the width of the hardware cursor in pixels.

37.5.123 DP Color Cursor Mapping Async 0 Flow Register (IPUx_DP_CUR_MAP_ASYNC0)

The LCD Color Cursor Mapping Register defines the color of the cursor in passive or TFT color modes.

Address: Base address + 1_8070h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0								DP_CUR_COL_B_ASYNC0								DP_CUR_COL_G_ASYNC0								DP_CUR_COL_R_ASYNC0							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_MAP_ASYNC0 field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 DP_CUR_COL_B_ASYNC0	CUR_COL_B - Cursor Blue Field Defines the Blue component of the cursor color in color mode 00000000 No Blue. 11111111 Full Blue.
15–8 DP_CUR_COL_G_ASYNC0	CUR_COL_G - Cursor Green Field Defines the Green component of the cursor color in color mode 00000000 No Green. 11111111 Full Green.
DP_CUR_COL_R_ASYNC0	CUR_COL_R - Cursor Red Field Defines the Red component of the cursor color in color mode 00000000 No Red. 11111111 Full Red.

37.5.124 DP Gamma Constant Async 0 Flow Register i (IPUx_DP_GAMMA_C_ASYNC0_i)

This registers contains CONSTANTi parameters used for gamma correction inside the display processor (DP).

Address: Base address + 1_8074h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				DP_GAMMA_C_ASYNC0_2i_1												0							DP_GAMMA_C_ASYNC0_2i									
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_C_ASYNC0_i field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 DP_GAMMA_C_ASYNC0_2i_1	CONSTANTi+1 parameter of Gamma Correction.
15–9 Reserved	This read-only field is reserved and always has the value 0.
DP_GAMMA_C_ASYNC0_2i	CONSTANTi parameter of Gamma Correction.

37.5.125 DP Gamma Correction Slope Async 0 Flow Register i (IPUx_DP_GAMMA_S_ASYNC0_i)

This registers contains SLOPEi parameters used for Gamma Correction inside the display processor (DP).

Address: Base address + 1_8094h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_GAMMA_S_ASYNC0_4i_3								DP_GAMMA_S_ASYNC0_4i_2								DP_GAMMA_S_ASYNC0_4i_1								DP_GAMMA_S_ASYNC0_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_S_ASYNC0_i field descriptions

Field	Description
31–24 DP_GAMMA_S_ASYNC0_4i_3	SLOPE<4*i+3> parameter of Gamma Correction.
23–16 DP_GAMMA_S_ASYNC0_4i_2	SLOPE<4*i+2> parameter of Gamma Correction.
15–8 DP_GAMMA_S_ASYNC0_4i_1	SLOPE<4*i+1> parameter of Gamma Correction.
DP_GAMMA_S_ASYNC0_4i	SLOPE<4*i> parameter of Gamma Correction.

37.5.126 DP Color Space Conversion Control Async 0 Flow Register i (IPUx_DP_CSCA_ASYNC0_i)

Address: Base address + 1_80A4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0						DP_CSC_A_ASYNC0_2i_1										0						DP_CSC_A_ASYNC0_2i										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CSCA_ASYNC0_i field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DP_CSCA_ASYNC0_i field descriptions (continued)

Field	Description
25–16 DP_CSC_A_ ASYNC0_2i_1	A<2*i+1> parameter of color conversion
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A_ ASYNC0_2i	A<2*i> parameter of color conversion.

37.5.127 DP Color Conversion Control Async 0 Flow Register 0 (IPUx_DP_CSC_ASYNC0_0)

Address: Base address + 1_80B4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_ S0_ASYNC0		DP_CSC_B0_ASYNC0													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0						DP_CSC_A8_ASYNC0									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CSC_ASYNC0_0 field descriptions

Field	Description
31–30 DP_CSC_S0_ ASYNC0	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B0_ ASYNC0	B0 parameter of color conversion.
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A8_ ASYNC0	A9 parameter of color conversion.

37.5.128 DP Color Conversion Control Async 1 Flow Register (IPUx_DP_CSC_ASYNC_1)

Address: Base address + 1_80B8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_S2_ASYNC0		DP_CSC_B2_ASYNC0													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_CSC_S1_ASYNC0		DP_CSC_B1_ASYNC0													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

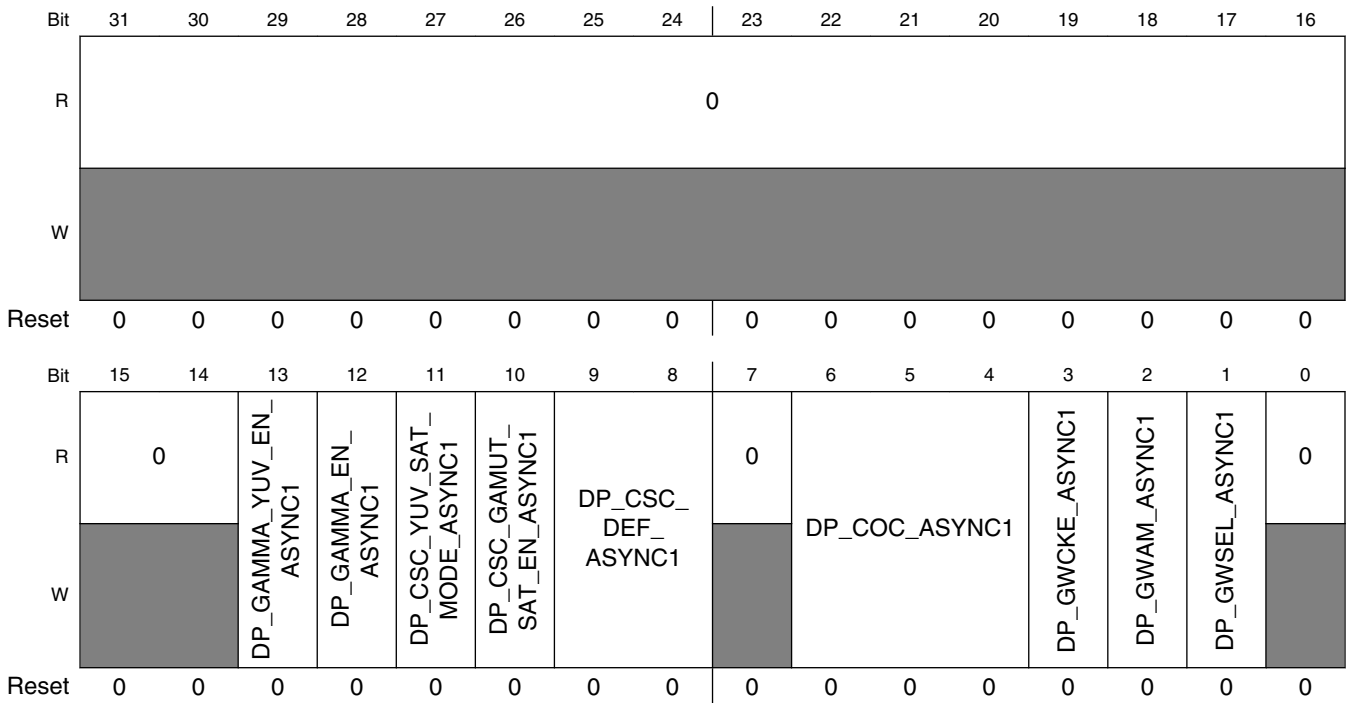
IPUx_DP_CSC_ASYNC_1 field descriptions

Field	Description
31–30 DP_CSC_S2_ASYNC0	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B2_ASYNC0	B0 parameter of color conversion.
15–14 DP_CSC_S1_ASYNC0	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
DP_CSC_B1_ASYNC0	B0 parameter of color conversion.

37.5.129 DP Common Configuration Async 1 Flow Register (IPUx_DP_COM_CONF_ASYNC1)

This register contains common configuration parameters for the DP.

Address: Base address + 1_80BCh offset



IPUx_DP_COM_CONF_ASYNC1 field descriptions

Field	Description
31–14 Reserved	This read-only field is reserved and always has the value 0.
13 DP_GAMMA_YUV_EN_ASYNC1	GAMMA's YUV mode enable for async flow 1 0 YUV mode is OFF. 1 YUV mode is ON.
12 DP_GAMMA_EN_ASYNC1	GAMMA_EN - Gamma correction block enable bit 0 disable 1 enable
11 DP_CSC_YUV_SAT_MODE_ASYNC1	CSC_YUV_SAT_MODE YUV saturation mode for color space conversion 0 Y/U/V range 0 -255 1 Y range 16-235, U/V range 16-240

Table continues on the next page...

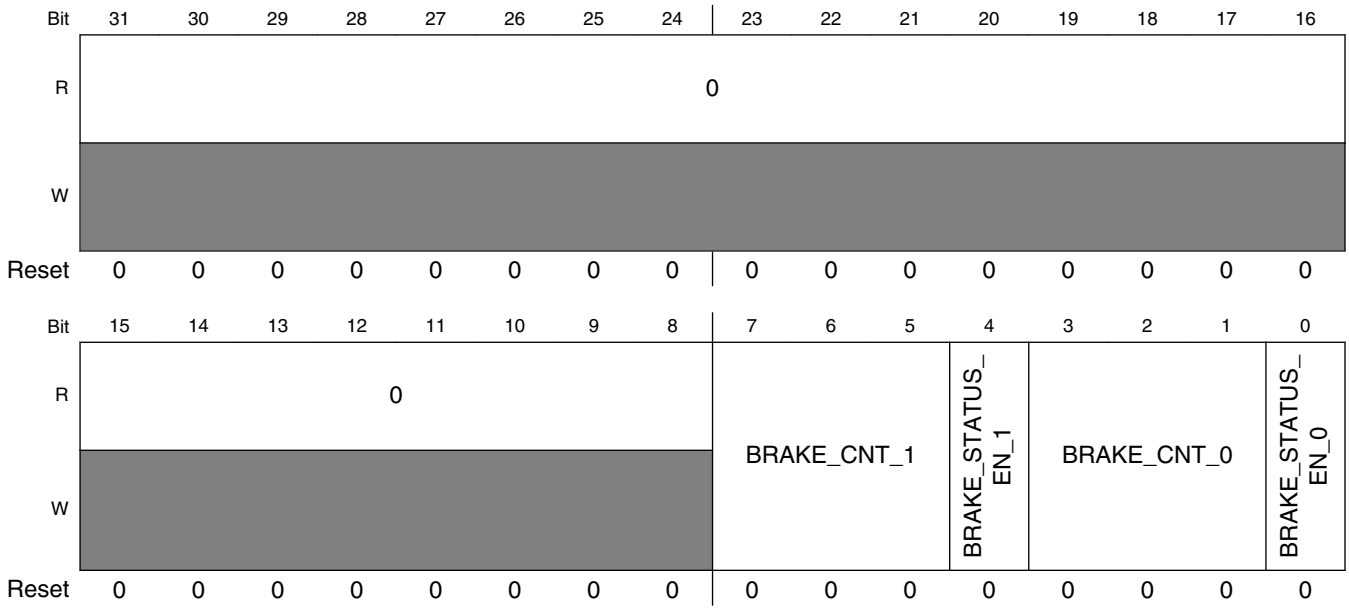
IPUx_DP_COM_CONF_ASYNC1 field descriptions (continued)

Field	Description
10 DP_CSC_ GAMUT_SAT_ EN_ASYNC1	CSC_GAMUT_SAT_EN Indicate if GAMUT saturation is enabled 0 disable GAMUT mapping 1 enable GAMUT mapping
9–8 DP_CSC_DEF_ ASYNC1	CSC_DEF Enable or disable Color Space Conversion. 00 CSC disable 01 CSC enable after combining 10 CSC enable before combining on BG channel 11 CSC enable before combining on FG channel
7 Reserved	This read-only field is reserved and always has the value 0.
6–4 DP_COC_ ASYNC1	COC - Cursor Operation Control Controls the format of the cursor and the type of arithmetic operations 000 Transparent, cursor is disabled. 001 Full cursor. 010 Reversed cursor. 011 AND between full plane and cursor. 100 Reserved 101 OR between full plane and cursor. 110 XOR between full plane and cursor. 111 Reserved
3 DP_GWCKE_ ASYNC1	GWCKE - Graphic Window Color Keying Enable Enable or disable graphic window color keying. 1 Enable color keying of graphic window 0 Disable color keying of graphic window
2 DP_GWAM_ ASYNC1	GWAM - Graphic Window Alpha Mode Select the use of Alpha to be global or local. 1 Global Alpha. 0 Local Alpha.
1 DP_GWSEL_ ASYNC1	GWSEL - Graphic Window Select Select graphic window to be on partial plane or full plane. 1 Graphic window is partial plane. 0 Graphic window is full plane.
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.130 DP Debug Control Register (IPUx_DP_DEBUG_CNT)

This is the debug unit control register. This register is not stored in the SRM.

Address: Base address + 1_80BCh offset



IPUx_DP_DEBUG_CNT field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7–5 BRAKE_CNT_1	The async flow can be broken multiple times. It possible to control which breaking event will cause the interrupt. This field counts the breaking events for unit #1
4 BRAKE_ STATUS_EN_1	This bit enables the break/status unit #1
3–1 BRAKE_CNT_0	The async flow can be broken multiple times. It possible to control which breaking event will cause the interrupt. This field counts the breaking events for unit #0
0 BRAKE_ STATUS_EN_0	This bit enables the break/status unit #0

37.5.131 DP Graphic Window Control Async 1 Flow Register (IPUx_DP_GRAPH_WIND_CTRL_ASYNC1)

This register contains common configuration parameters for the DP.

Address: Base address + 1_80C0h offset

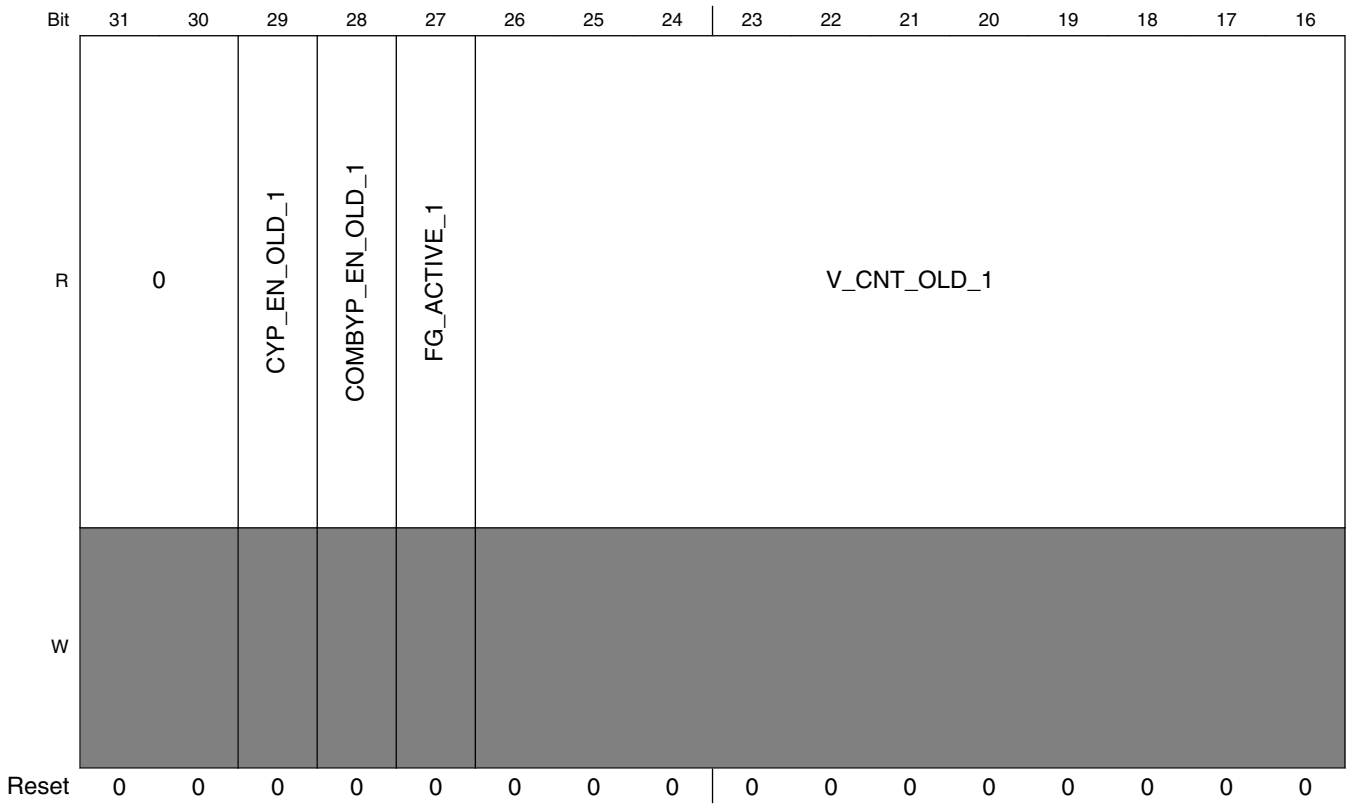
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_GWAV_ASYNC1								DP_GWCKR_ASYNC1								DP_GWCKG_ASYNC1								DP_GWCKB_ASYNC1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

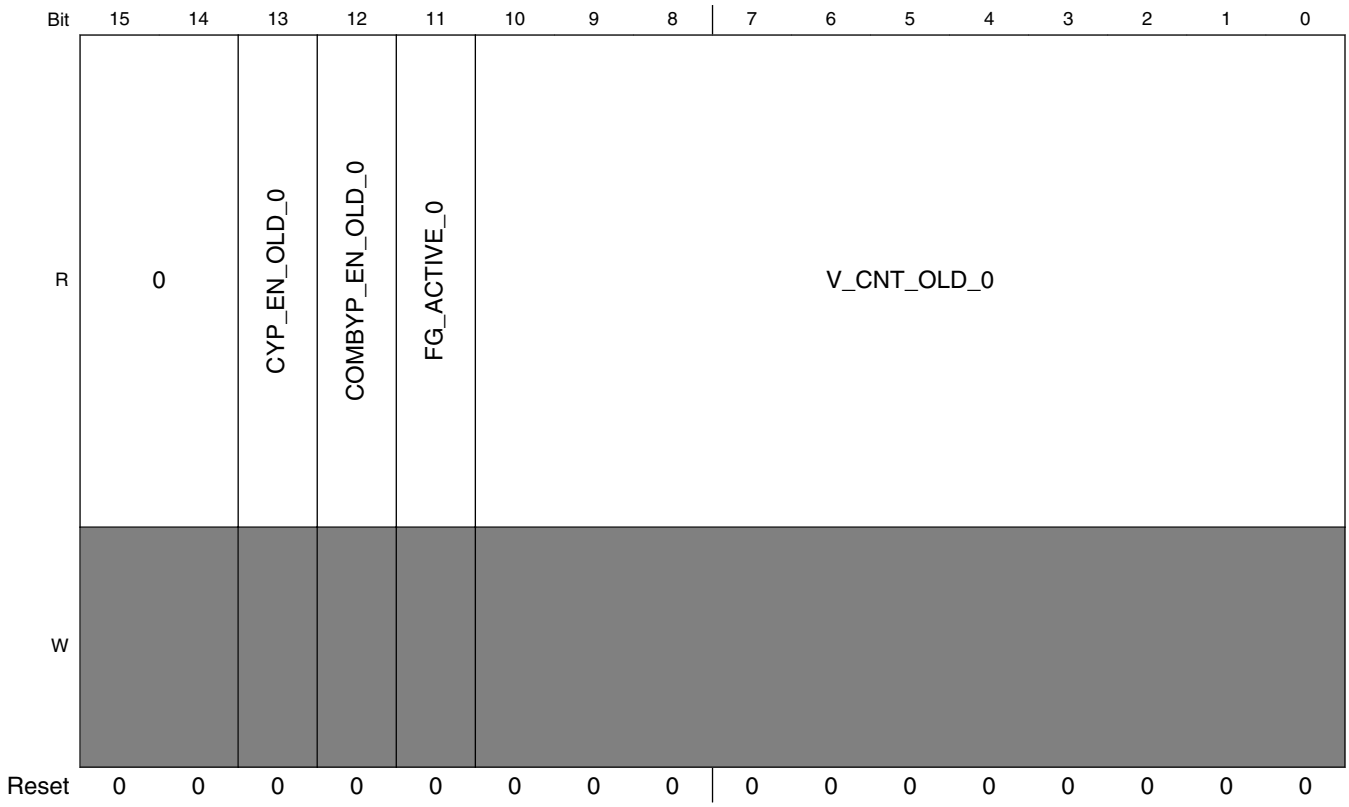
IPUx_DP_GRAPH_WIND_CTRL_ASYNC1 field descriptions

Field	Description
31–24 DP_GWAV_ASYNC1	GWAV - Graphic Window Alpha Value Defines the alpha value of graphic window used for alpha blending between graphic window and full plane. The Value the number that writing to GWAV register. The Actual Value the number, that insert to calculation in Combining Module. If Value = < 0.5- 1/256 (01111111) then Actual Value = Value. If Value >= 0.5 (10000000), then Actual Value = Value + 1/256. 00000000 Actual value is 00000000; Graphic window totally opaque i.e. overlay on LCD screen 01111111 Actual value is 01111111; 10000000 Actual value is 10000001 10000001 Actual value is 10000010 11111110 Actual value is 11111111 11111111 Actual value is 100000000;Graphic window totally transparent i.e. not displayed on LCD screen
23–16 DP_GWCKR_ASYNC1	GWCKR - Graphic Window Color Keying Red Component Defines the red component of graphic window color keying. 00000000 No red 11111111 Full red
15–8 DP_GWCKG_ASYNC1	GWCKG - Graphic Window Color Keying Green Component Defines the green component of graphic window color keying. 00000000 No Green 11111111 Full Green
DP_GWCKB_ASYNC1	GWCKB - Graphic Window Color Keying Blue Component Defines the blue component of graphic window color keying. 00000000 No blue 11111111 Full blue

37.5.132 DP Debug Status Register (IPUx_DP_DEBUG_STAT)

Address: Base address + 1_80C0h offset





IPUx_DP_DEBUG_STAT field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
29 CYP_EN_OLD_1	The async flow has been broken in the middle of a cursor (This filed is relevant for debug unit #1)
28 COMBYP_EN_OLD_1	the async1 flow has been broken in the middle of combining (This filed is relevant for debug unit #1)
27 FG_ACTIVE_1	Displaying the partial frame has been started (This filed is relevant for debug unit #1)
26–16 V_CNT_OLD_1	The exact row where the async flow has been broken (This filed is relevant for debug unit #0)
15–14 Reserved	This read-only field is reserved and always has the value 0.
13 CYP_EN_OLD_0	The async flow has been broken in the middle of a cursor (This filed is relevant for debug unit #0)
12 COMBYP_EN_OLD_0	the async flow has been broken in the middle of combining (This filed is relevant for debug unit #0)
11 FG_ACTIVE_0	Displaying the partial frame has been started for async flow (This filed is relevant for debug unit #0)
V_CNT_OLD_0	The exact row where the async flow has been broken (This filed is relevant for debug unit #0)

37.5.133 DP Partial Plane Window Position Async 1 Flow Register (IPUx_DP_FG_POS_ASYNC1)

The DP partial plane Window Position Register is used to determine the starting position of the partial plane window relative to BG window position.

This Register is used for the synchronous flow only.

Address: Base address + 1_80C4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					DP_FGXP_ASYNC1											0					DP_FGYP_ASYNC1										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_FG_POS_ASYNC1 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 DP_FGXP_ ASYNC1	FGXP partial plane Window X Position. partial plane Window X Position - Specifies the number of pixels between the start of full plane window X position and the beginning of the first data of new line.
15–11 Reserved	This read-only field is reserved and always has the value 0.
DP_FGYP_ ASYNC1	FGYP partial plane window Y position partial plane Window Y Position - Specifies the number of lines between the start of full plane windows Y position and the beginning of the first data.

37.5.134 DP Cursor Postion and Size Async 1 Flow Register (IPUx_DP_CUR_POS_ASYNC1)

The LCD Cursor Position Register is used to determine the starting position of the cursor relative to BG windows position.

Address: Base address + 1_80C8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DP_CYP_ ASYNC1					DP_CYH_ASYNC1											DP_CXP_ ASYNC1					DP_CXW_ASYNC1										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_POS_ASYNC1 field descriptions

Field	Description
31–27 DP_CYP_ ASYNC1	CYP - Cursor Y Position Represents the cursors vertical starting position Y in pixel count (from 0 to CYH).Live View Resolution Mode.
26–16 DP_CYH_ ASYNC1	CYH - Cursor Height Specifies the height of the hardware cursor in pixels.
15–11 DP_CXP_ ASYNC1	CXP - Cursor X Position Represents the cursors horizontal starting position X in pixel count (from 0 to CXW).
DP_CXW_ ASYNC1	CXW - Cursor Width. Specifies the width of the hardware cursor in pixels.

37.5.135 DP Color Cursor Mapping Async 1 Flow Register (IPUx_DP_CUR_MAP_ASYNC1)

The LCD Color Cursor Mapping Register defines the color of the cursor in passive or TFT color modes.

Address: Base address + 1_80CCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								DP_CUR_COL_B_ASYNC1								DP_CUR_COL_G_ASYNC1								DP_CUR_COL_R_ASYNC1							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CUR_MAP_ASYNC1 field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 DP_CUR_COL_ B_ASYNC1	CUR_COL_B - Cursor Blue Field Defines the Blue component of the cursor color in color mode 00000000 No Blue. 11111111 Full Blue.
15–8 DP_CUR_COL_ G_ASYNC1	CUR_COL_G - Cursor Green Field Defines the Green component of the cursor color in color mode 00000000 No Green. 11111111 Full Green.
DP_CUR_COL_ R_ASYNC1	CUR_COL_R - Cursor Red Field Defines the Red component of the cursor color in color mode

Table continues on the next page...

IPUx_DP_CUR_MAP_ASYNC1 field descriptions (continued)

Field	Description
	00000000 No Red. 11111111 Full Red.

37.5.136 DP Gamma Constants Async 1 Flow Register i (IPUx_DP_GAMMA_C_ASYNC1_i)

This registers contains CONSTANTi parameters used for gamma correction inside the display processor (DP).

Address: Base address + 1_80D0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							DP_GAMMA_C_ASYNC1_2i_1									0							DP_GAMMA_C_ASYNC1_2i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_C_ASYNC1_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 DP_GAMMA_C_ASYNC1_2i_1	CONSTANTi+1 parameter of Gamma Correction.
15–9 Reserved	This read-only field is reserved and always has the value 0.
DP_GAMMA_C_ASYNC1_2i	CONSTANTi parameter of Gamma Correction.

37.5.137 DP Gamma Correction Slope Async 1 Flow Register i (IPUx_DP_GAMMA_S_ASYN1_i)

This registers contains SLOPEi parameters used for Gamma Correction inside the display processor (DP).

Address: Base address + 1_80F0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_GAMMA_S_ASYNC1_4i_3								DP_GAMMA_S_ASYNC1_4i_2								DP_GAMMA_S_ASYNC1_4i_1								DP_GAMMA_S_ASYNC1_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_GAMMA_S_ASYN1_i field descriptions

Field	Description
31–24 DP_GAMMA_S_ASYNC1_4i_3	SLOPE<4*i+3> parameter of Gamma Correction.
23–16 DP_GAMMA_S_ASYNC1_4i_2	SLOPE<4*i+2> parameter of Gamma Correction.
15–8 DP_GAMMA_S_ASYNC1_4i_1	SLOPE<4*i+1> parameter of Gamma Correction.
DP_GAMMA_S_ASYNC1_4i	SLOPE<4*i> parameter of Gamma Correction.

37.5.138 DP Color Space Converstion Control Async 1 Flow Register i (IPUx_DP_CSCA_ASYNC1_i)

Address: Base address + 1_8100h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0						DP_CSC_A_ASYNC1_2i_1										0						DP_CSC_A_ASYNC1_2i									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CSCA_ASYNC1_i field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DP_CSCA_ASYNC1_i field descriptions (continued)

Field	Description
25–16 DP_CSC_A_ ASYNC1_2i_1	A<2*i+1> parameter of color conversion.
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A_ ASYNC1_2i	A<2*i> parameter of color conversion.

37.5.139 DP Color Conversion Control Async 1 Flow Register 0 (IPUx_DP_CSC_ASYNC1_0)

Address: Base address + 1_8110h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_ S0_ASYNC1		DP_CSC_B0_ASYNC1													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0						DP_CSC_A8_ASYNC1									
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DP_CSC_ASYNC1_0 field descriptions

Field	Description
31–30 DP_CSC_S0_ ASYNC1	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B0_ ASYNC1	B0 parameter of color conversion.
15–10 Reserved	This read-only field is reserved and always has the value 0.
DP_CSC_A8_ ASYNC1	A9 parameter of color conversion.

37.5.140 DP Color Conversion Control Async 1 Flow Register 1 (IPUx_DP_CSC_ASYNC1_1)

Address: Base address + 1_8114h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DP_CSC_S2_ASYNC1		DP_CSC_B2_ASYNC1													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DP_CSC_S1_ASYNC1		DP_CSC_B1_ASYNC1													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

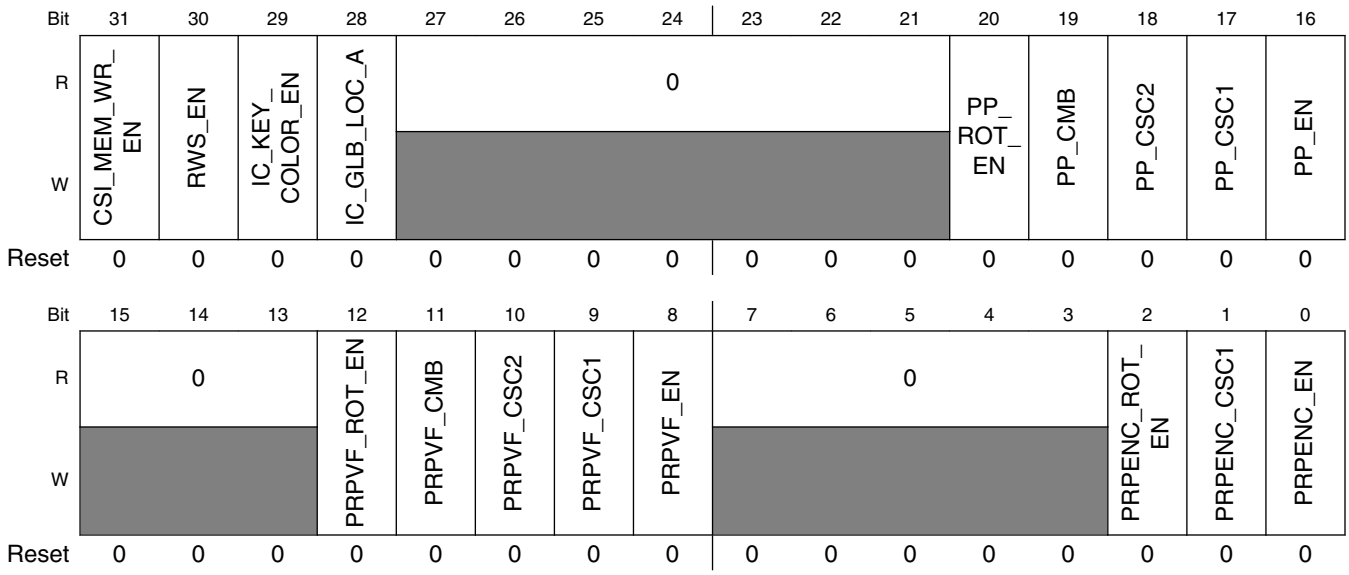
IPUx_DP_CSC_ASYNC1_1 field descriptions

Field	Description
31–30 DP_CSC_S2_ASYNC1	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
29–16 DP_CSC_B2_ASYNC1	B0 parameter of color conversion.
15–14 DP_CSC_S1_ASYNC1	S0 parameter of color conversion. 00 scale factor of 2 01 scale factor of 1 10 scale factor of 0 11 scale factor of -1
DP_CSC_B1_ASYNC1	B0 parameter of color conversion.

37.5.141 IC Configuration Register (IPUx_IC_CONF)

This register contains control parameter for IC 3 tasks (pre-processing for encoding, pre-processing for view-finder and post processing).

Address: Base address + 2_0000h offset



IPUx_IC_CONF field descriptions

Field	Description
31 CSI_MEM_WR_EN	CSI direct memory write enable. This bit enables writing data from sensor directly to memory even when a raw sensor is not attached. 0 CSI direct writing to memory is disabled. 1 CSI direct writing to memory is enabled.
30 RWS_EN	Raw sensor enable. This bit indicate if a Raw sensor is attached (Bayer format). This bit is used together with the CSI_MEM_WR_EN bit as follows: CSI_MEM_WR_EN=0, RWS_EN=0 - data is fed from the CSI to the IC for processing; CSI_MEM_WR_EN=1, RWS_EN=0 - data is fed from the CSI to the IC for processing and also for writing to the system memory; CSI_MEM_WR_EN=0, RWS_EN=1 - data is fed from the CSI to the system memory (via the IC) and from the system memory to the IC for processing; CSI_MEM_WR_EN=1, RWS_EN=1 - non-valid configuration. 0 Raw sensor is not attached. 1 Raw sensor is attached.
29 IC_KEY_COLOR_EN	Key Color enable. This bit enables the key color feature.

Table continues on the next page...

IPUx_IC_CONF field descriptions (continued)

Field	Description
	0 Key color is disabled. 1 Key color is enabled.
28 IC_GLB_LOC_A	Global Alpha. This bit select the source of Alpha parameter. 0 Alpha parameter is local. 1 Alpha parameter is global.
27–21 Reserved	This read-only field is reserved and always has the value 0.
20 PP_ROT_EN	Post-Processing Rotation Task enable. This bit enable Post-Processing Rotation Task. 0 Rotation is disabled. 1 Rotation is enabled.
19 PP_CMB	Post-Processing Task combining enable. This bit enables combining. 0 Combining is disabled. 1 Combining is enabled.
18 PP_CSC2	Post-Processing Task color conversion RGB-->YUV enable. This bit enables YUV-->RGB. Reserved 0 RGB-->YUV is disabled. 1 RGB-->YUV is enabled.
17 PP_CSC1	Post-Processing Task color conversion YUV-->RGB enable. This bit enables YUV-->RGB. 0 YUV-->RGB is disabled. 1 YUV-->RGB is enabled.
16 PP_EN	Post-Processing Task enable. This bit enables the Post-Processing task. 0 Task is disabled. 1 Task is enabled.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12 PRPVF_ROT_EN	Preprocessing Rotation Task for viewfinder enable. This bit enable Preprocessing Rotation Task for viewfinder. 0 Rotation is disabled. 1 Rotation is enabled.
11 PRPVF_CMB	Preprocessing Task for View-Finder combining enable. This bit enables combining. 0 Combining is disabled. 1 Combining is enabled.
10 PRPVF_CSC2	Reserved
9 PRPVF_CSC1	Pre-processing task for view-finder first color conversion enable. This bit enables first color conversion. 0 First color conversion is disabled. 1 First color conversion is enabled.
8 PRPVF_EN	Preprocessing Task for View-Finder enable. This bit enables the View-Finder task.

Table continues on the next page...

IPUx_IC_CONF field descriptions (continued)

Field	Description
	0 Task is disabled. 1 Task is enabled.
7–3 Reserved	This read-only field is reserved and always has the value 0.
2 PRPENC_ROT_EN	Preprocessing Rotation Task for encoding enable. This bit enable Preprocessing Rotation Task for encoding. 0 Rotation is disabled. 1 Rotation is enabled.
1 PRPENC_CSC1	Preprocessing Task for encoding color conversion enable. This bit enables color conversion. 0 Color conversion is disabled. 1 Color conversion is enabled.
0 PRPENC_EN	Preprocessing Task for encoding enable. This bit enables the encoding task. 0 Task is disabled. 1 Task is enabled.

37.5.142 IC Preprocessing Encoder Resizing Coefficients Register (IPUx_IC_PRP_ENC_RSC)

This register contains the resizing and downsizing parameters for Preprocessing task for encoding.

Address: Base address + 2_0004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PRPENC_DS_R_V		PRPENC_RS_R_V														PRPENC_DS_R_H		PRPENC_RS_R_H													
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_PRP_ENC_RSC field descriptions

Field	Description
31–30 PRPENC_DS_R_V	Preprocessing task for encoding Downsizing vertical Ratio. This field contains the downsizing vertical coefficient of Preprocessing for Encoding.
29–16 PRPENC_RS_R_V	Preprocessing task for encoding Resizing vertical Ratio. This field contains the resizing vertical coefficient of Preprocessing for Encoding. Resizing Ratio is equal to PRPENC_RS_R_V: M

Table continues on the next page...

IPUx_IC_PRP_ENC_RSC field descriptions (continued)

Field	Description
	Where M = 2^13; SI - input size; SO - output size PRPENC_RS_R_V = floor(M*(SI-1)/(SO-1));
15–14 PRPENC_DS_R_H	Preprocessing task for encoding Downsizing horizontal Ratio. This field contains the downsizing horizontal coefficient of Preprocessing for Encoding. Values: 00 1 01 2 10 4 11 RSV
PRPENC_RS_R_H	Preprocessing task for encoding Resizing horizontal Ratio. This field contains the resizing horizontal coefficient of Preprocessing for Encoding. Resizing Ratio is equal to PRPENC_RS_R_H: M Where M = 2^13; SI - input size; SO - output size PRPENC_RS_R_H = floor(M*(SI-1)/(SO-1));

37.5.143 IC Preprocessing View-Finder Resizing Coefficients Register (IPUx_IC_PRP_VF_RSC)

This register contains the resizing and downsizing parameters for preprocessing task for display.

Address: Base address + 2_0008h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	PRPVF_DS_R_V		PRPVF_RS_R_V													
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PRPVF_DS_R_H		PRPVF_RS_R_H													
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_PRP_VF_RSC field descriptions

Field	Description
31–30 PRPVF_DS_R_V	Preprocessing task for encoding Downsizing vertical Ratio. This field contains the downsizing vertical coefficient of Preprocessing for View-Finder.
29–16 PRPVF_RS_R_V	Preprocessing task for encoding Resizing vertical Ratio. This field contains the resizing vertical coefficient of Preprocessing for View-Finder. Resizing Ratio is equal to PRPVF_RS_R_V: M

Table continues on the next page...

IPUx_IC_PRP_VF_RSC field descriptions (continued)

Field	Description
	Where M = 2^13; SI - input size; SO - output size PRPVF_RS_R_V = floor(M*(SI-1)/(SO-1));
15–14 PRPVF_DS_R_H	Preprocessing task for encoding Downsizing horizontal Ratio. This field contains the downsizing horizontal coefficient of Preprocessing for View-Finder. Values: 00 1 01 2 10 4 11 RSV
PRPVF_RS_R_H	Preprocessing task for view-finding resizing horizontal ratio. This field contains the resizing horizontal coefficient of preprocessing Task For View-finder. Resizing Ratio is equal to PRPVF_RS_R_H: M Where M = 2^13; SI - input size; SO - output size PRPVF_RS_R_H = floor(M*(SI-1)/(SO-1));

37.5.144 IC Postprocessing Encoder Resizing Coefficients Register (IPUx_IC_PP_RSC)

This register contains the resizing and downsizing parameters for Post-Processing task for display.

Address: Base address + 2_000Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	PP_DS_R_V		PP_RS_R_V														PP_DS_R_H		PP_RS_R_H													
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_PP_RSC field descriptions

Field	Description
31–30 PP_DS_R_V	Post-Processing task Downsizing vertical Ratio. This field contains the downsizing vertical coefficient of Post-Processing.
29–16 PP_RS_R_V	Post-Processing task Resizing vertical Ratio. This field contains the resizing vertical coefficient of Post-Processing. Resizing Ratio is equal to PP_RS_R_V: M Where M = 2^13; SI - input size; SO - output size PP_RS_R_V = floor(M*(SI-1)/(SO-1));

Table continues on the next page...

IPUx_IC_PP_RSC field descriptions (continued)

Field	Description
15–14 PP_DS_R_H	Post-Processing task Downsizing horizontal Ratio. This field contains the downsizing horizontal coefficient of Post-Processing. 00 1 01 2 10 4 11 RSV
PP_RS_R_H	Post-Processing task Resizing horizontal Ratio. This field contains the resizing horizontal coefficient of Post-Processing. Resizing Ratio is equal to PP_RS_R_H: M Where M = 2^13; SI - input size; SO - output size PP_RS_R_H = floor(M*(SI-1)/(SO-1));

37.5.145 IC Combining Parameters Register 1 (IPUx_IC_CMBP_1)

Address: Base address + 2_0010h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																IC_PP_ALPHA_V								IC_PRPVF_ALPHA_V							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_CMBP_1 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 IC_PP_ALPHA_V	Post-Processing task Global Alpha. This field contains the Global Alpha value of Post-Processing.
IC_PRPVF_ALPHA_V	Preprocessing task for viewfinder Global Alpha. This field contains the Global Alpha value of Preprocessing for viewfinder.

37.5.146 IC Combining Parameters Register 2 (IPUx_IC_CMBP_2)

Address: Base address + 2_0014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								IC_KEY_COLOR_R								IC_KEY_COLOR_G								IC_KEY_COLOR_B							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_CMBP_2 field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 IC_KEY_ COLOR_R	Key Color Red.
15–8 IC_KEY_ COLOR_G	Key Color Green.
IC_KEY_ COLOR_B	Key Color Blue.

37.5.147 IC IDMAC Parameters 1 Register (IPUx_IC_IDMAC_1)

Address: Base address + 2_0018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0						ALT_CB7_BURST_16	ALT_CB6_BURST_16	0	T3_FLIP_RS	T2_FLIP_RS	T1_FLIP_RS	T3_FLIP_UD	T3_FLIP_LR	T3_ROT	T2_FLIP_UD
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	T2_FLIP_LR	T2_ROT	T1_FLIP_UD	T1_FLIP_LR	T1_ROT	0			CB7_BURST_16	CB6_BURST_16	CB5_BURST_16	CB4_BURST_16	CB3_BURST_16	CB2_BURST_16	CB1_BURST_16	CB0_BURST_16
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_IDMAC_1 field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25 ALT_CB7_ BURST_16	Reserved
24 ALT_CB6_ BURST_16	Reserved
23 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_IC_IDMAC_1 field descriptions (continued)

Field	Description
22 T3_FLIP_RS	LEFT/RIGHT flip for Post Processing (PP) task; his bit affect the flipping done on the resizing unit. The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip
21 T2_FLIP_RS	LEFT/RIGHT flip for View Finder (VF) task; this bit affect the flipping done on the resizing unit. The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip
20 T1_FLIP_RS	LEFT/RIGHT flip for Encoding (ENC) task; this bit affect the flipping done on the resizing unit. The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip
19 T3_FLIP_UD	UP/DOWN flip for Post Processing (PP) task The value of this field must be identical to the corresponding channel's VF parameters in the IDMAC's CPMEM 1 Vertical flip enable 0 no flip
18 T3_FLIP_LR	LEFT/RIGHT flip for Post Processing (PP) task; this bit affect the flipping done on the rotation unit The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip
17 T3_ROT	Rotation for Post Processing (PP) task The value of this field must be identical to the corresponding channel's ROT parameters in the IDMAC's CPMEM 1 90 degree rotation clockwise 0 no rotation
16 T2_FLIP_UD	UP/DOWN flip for View Finder (VF) task The value of this field must be identical to the corresponding channel's VF parameters in the IDMAC's CPMEM 1 Vertical flip enable 0 no flip
15 T2_FLIP_LR	LEFT/RIGHT flip for View Finder (VF) task; this bit affect the flipping done on the rotation unit The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip

Table continues on the next page...

IPUx_IC_IDMAC_1 field descriptions (continued)

Field	Description
14 T2_ROT	Rotation for View Finder (VF) task The value of this field must be identical to the corresponding channel's ROT parameters in the IDMAC's CPMEM 1 90 degree rotation clockwise 0 no rotation
13 T1_FLIP_UD	UP/DOWN flip for Encoding (ENC) task The value of this field must be identical to the corresponding channel's VF parameters in the IDMAC's CPMEM 1 Vertical flip enable 0 no flip
12 T1_FLIP_LR	LEFT/RIGHT flip for Encoding (ENC) task; this bit affect the flipping done on the rotation unit The value of this field must be identical to the corresponding channel's HF parameter in the IDMAC's CPMEM 1 horizontal flip enabled 0 no flip
11 T1_ROT	Rotation for Encoding (ENC) task The value of this field must be identical to the corresponding channel's ROT parameters in the IDMAC's CPMEM 1 90 degree rotation clockwise 0 no rotation
10–8 Reserved	This read-only field is reserved and always has the value 0.
7 CB7_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB7 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
6 CB6_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB6 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
5 CB5_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB5 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
4 CB4_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB4 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM

Table continues on the next page...

IPUx_IC_IDMAC_1 field descriptions (continued)

Field	Description
	0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
3 CB3_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB3 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
2 CB2_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB2 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
1 CB1_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB1 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111
0 CB0_BURST_16	This bit defines the number of active cycles within a burst (burst size) coming from the IDMAC for IC's CB0 For pixel data the number of pixels should match the NPB[6:2] value on the IDMAC's CPMEM 0 Burst size is 8 pixels; The Matching NPB[6:2] should be 00111 1 Burst size is 16 pixels; The Matching NPB[6:2] should be 01111

37.5.148 IC IDMAC Parameters 2 Register (IPUx_IC_IDMAC_2)

Address: Base address + 2_001Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0		T3_FR_HEIGHT										T2_FR_HEIGHT										T1_FR_HEIGHT									
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_IDMAC_2 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
29–20 T3_FR_HEIGHT	Frame Height for Post Processing (PP) task The value of this field must be identical to the corresponding FH channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1
19–10 T2_FR_HEIGHT	Frame Height for View Finder (VF) task

Table continues on the next page...

IPUx_IC_IDMAC_2 field descriptions (continued)

Field	Description
	The value of this field must be identical to the corresponding FH channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1
T1_FR_HEIGHT	Frame Height for Encoding (ENC) task The value of this field must be identical to corresponding FH channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1

37.5.149 IC IDMAC Parameters 3Register (IPUx_IC_IDMAC_3)

Address: Base address + 2_0020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0		T3_FR_WIDTH														T2_FR_WIDTH												T1_FR_WIDTH			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_IDMAC_3 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
29–20 T3_FR_WIDTH	Frame Width for Post Processing (PP) task The value of this field must be identical to the corresponding FW channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1
19–10 T2_FR_WIDTH	Frame Width for View Finder (VF) task The value of this field must be identical to the corresponding FW channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1
T1_FR_WIDTH	Frame Width for Encoding (ENC) task The value of this field must be identical to corresponding FW channel's parameters in the IDMAC's CPMEM. This parameter refers to the output's size -1

37.5.150 IC IDMAC Parameters 4 Register (IPUx_IC_IDMAC_4)

Address: Base address + 2_0024h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																rm_brdg_ max_rq				ibm_brdg_ max_rq				mpm_dmfc_ brdg_max_rq				mpm_rw_ brdg_max_rq			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_IC_IDMAC_4 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–12 rm_brdg_max_rq	RM memory Bridge Max Requests 0000 Feature is disabled 0001 Max request is 1 1111 Max request are15
11–8 ibm_brdg_max_rq	IBM memory Bridge Max Requests 0000 Feature is disabled 0001 Max request is 1 1111 Max request are15
7–4 mpm_dmfc_brdg_max_rq	MPM memory Bridge Max Requests for the IC DMFC interface 0000 Feature is disabled 0001 Max request is 1 1111 Max request are15
mpm_rw_brdg_max_rq	MPM memory Bridge Max Requests between MPM's read and writes 0000 Feature is disabled 0001 Max request is 1 1111 Max request are15

37.5.151 CSI0 Sensor Configuration Register (IPUx_CSI0_SENS_CONF)

Address: Base address + 3_0000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CSI0_DATA_EN_POL	0	CSI0_FORCE_EOF	CSI0_JPEG_MODE	CSI0_JPEG8_EN	CSI0_DATA_DEST			CSI0_DIV_RATIO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI0_EXT_VSYNC	CSI0_DATA_WIDTH				CSI0_SENS_DATA_FORMAT			CSI0_PACK_TIGHT	CSI0_SENS_PRTCL			CSI0_SENS_PIX_CLK_POL	CSI0_DATA_POL	CSI0_HSYNC_POL	CSI0_VSYNC_POL
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI0_SENS_CONF field descriptions

Field	Description
31 CSI0_DATA_EN_POL	Invert IPP_IND_SENSB_DATA_EN input. This bit selects the polarity of IPP_IND_SENSB_DATA_EN signal. 0 IPP_IND_SENSB_DATA_EN is directly applied to internal circuitry. 1 IPP_IND_SENSB_DATA_EN is inverted before applied to internal circuitry.
30 Reserved	This read-only field is reserved and always has the value 0.
29 CSI0_FORCE_EOF	Force End of frame This is a self clear bit allowing the user to force an End-of-frame event; This bit can be used in cases where the frame sent by the sensor was not completed. 1 force end of frame 0 no action
28 CSI0_JPEG_MODE	JPEG Mode - this bit defines the mode of the control signals when working in JPEG mode 1 The data is valid as long as HSYNC and VSYNC signals are active; HSYNC is valid for single frame 0 The frame starts withe the assertion of VSYNC. The frame ends on the next VSYNC or by setting the **CSI0_FORCE_EOF bit.**
27 CSI0_JPEG8_EN	JPEG8 enable bit 1 JPEG8 detection is enabled 0 JPEG8 is disabled
26–24 CSI0_DATA_DEST	These bits enable the destination of the data coming from the CSI. CSI0_DATA_DEST[0] - Reserved CSI0_DATA_DEST[1] - destination is IC CSI0_DATA_DEST[2] - destination is IDMAC via SMFC
23–16 CSI0_DIV_RATIO	DIV Ratio Clock division ratio minus 1. This field defines the division ratio of HSP_CLK into SENSB_MCLK: SENSB_MCLK rate = HSP_CLK rate /(DIV_RATIO+1)
15 CSI0_EXT_VSYNC	External VSYNC enable. This bits select between external and internal VSYNC. 0 Internal VSYNC mode. 1 External VSYNC mode.
14–11 CSI0_DATA_WIDTH	Data width. This field defines the number of bits per color. Values: 0000 4 bits per color 0001 8 bits per color 0010 9 bits per color 0011 10 bits per color 0100 11 bits per color 0101 12 bits per color 0110 13 bits per color 0111 14 bits per color 1000 15 bits per color 1001 16 bits per color

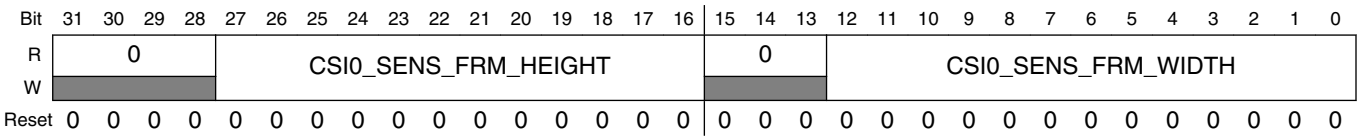
Table continues on the next page...

IPUx_CSI0_SENS_CONF field descriptions (continued)

Field	Description
10–8 CSI0_SENS_ DATA_FORMAT	Data format from the sensor. This field defines the data format for the input of the CSI sensor. Values: 000 full RGB or YUV444 001 YUV422 (YUYV...) 010 YUV422 (UYVY...) 011 Bayer or Generic data 100 RGB565 101 RGB555 110 RGB444 111 JPEG
7 CSI0_PACK_ TIGHT	CSI0 Pack Tight When the data format is YUV or RGB and the component's width is 9-16 bits, it can be sent to the memory in 2 different ways. 1 Three 10 bits components are packed into a 32 bit word. Color extension/reduction is performed 0 Each component is written as a 16 bit word where the MSB is written to bit #15, color extension is done for the remaining least significant bits.
6–4 CSI0_SENS_ PRTCL	Sensor Protocol. This bit defines the Sensor timing/data mode protocol. Values: 000 Gated clock mode 001 Non-gated clock mode 010 CCIR progressive mode (BT.656) 011 CCIR interlaced mode (BT.656) 100 CCIR progressive (BT.1120 DDR mode: data arrives on every edge of the clock) 101 CCIR progressive (BT.1120 SDR mode: data arrives only on the positive edge of the clock) 110 CCIR interlaced mode (BT.1120 DDR mode: data arrives on every edge of the clock) 111 CCIR interlaced mode (BT.1120 SDR mode: data arrives only on the positive edge of the clock)
3 CSI0_SENS_ PIX_CLK_POL	Invert Pixel clock input. This bit selects the polarity of pixel clock. 0 pixel clock is directly applied to internal circuitry. 1 pixel clock is inverted before applied to internal circuitry.
2 CSI0_DATA_ POL	Invert data input. This bit selects the polarity of data input. 0 data lines are directly applied to internal circuitry. 1 data lines are inverted before applied to internal circuitry.
1 CSI0_HSYNC_ POL	Invert IPP_IND_SENSB_HSYNC input. This bit selects the polarity of IPP_IND_SENSB_HSYNC signal. 0 IPP_IND_SENSB_HSYNC is directly applied to internal circuitry. 1 IPP_IND_SENSB_HSYNC is inverted before applied to internal circuitry.
0 CSI0_VSYNC_ POL	Invert IPP_IND_SENSB_VSYNC input. This bit selects the polarity of IPP_IND_SENSB_VSYNC signal. 0 IPP_IND_SENSB_VSYNC is not inverted before applied to internal circuitry. 1 IPP_IND_SENSB_VSYNC is inverted before applied to internal circuitry.

37.5.152 CSI0 Sense Frame Size Register (IPUx_CSI0_SENS_FRM_SIZE)

Address: Base address + 3_0004h offset

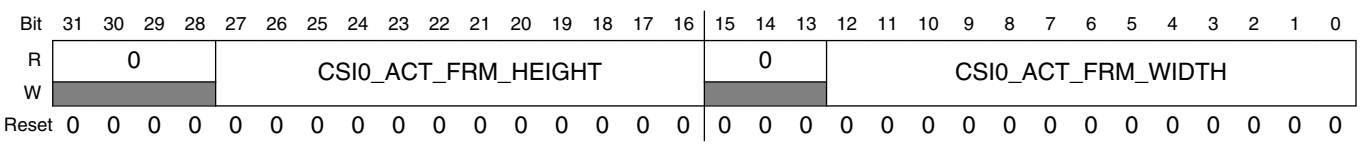


IPUx_CSI0_SENS_FRM_SIZE field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 CSI0_SENS_ FRM_HEIGHT	Sensor frame height minus 1. This field defines the sensor frame rows number minus 1.
15–13 Reserved	This read-only field is reserved and always has the value 0.
CSI0_SENS_ FRM_WIDTH	Sensor frame width minus 1. This field defines the sensor frame column number minus 1.

37.5.153 CSI0 Actual Frame Size Register (IPUx_CSI0_ACT_FRM_SIZE)

Address: Base address + 3_0008h offset

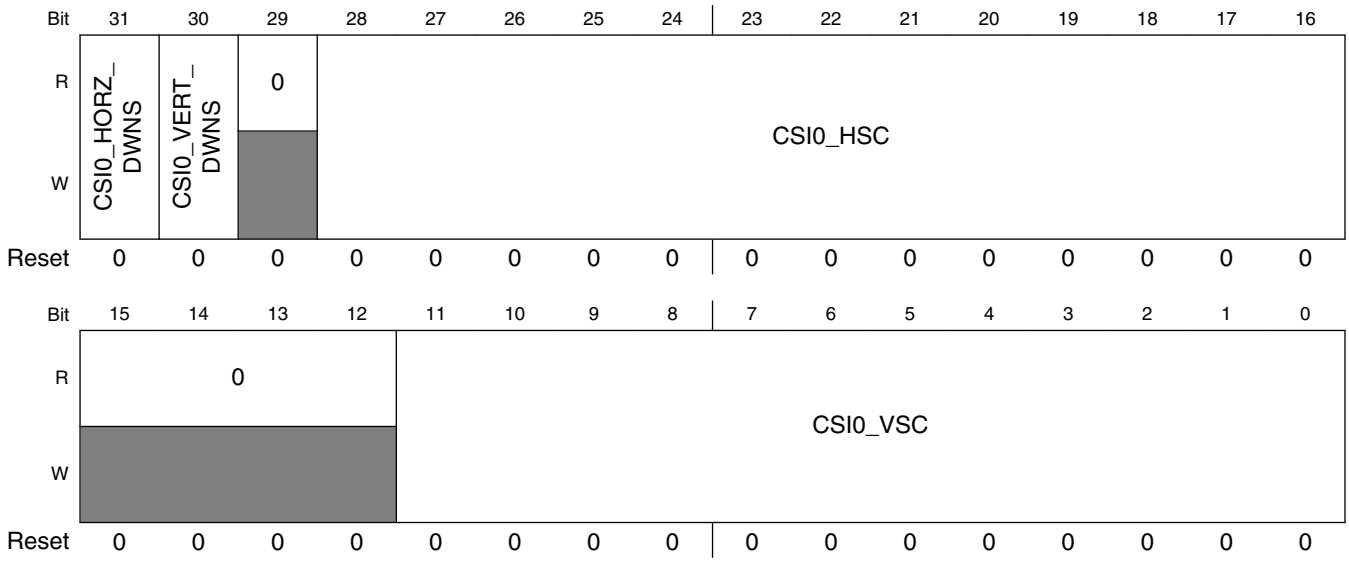


IPUx_CSI0_ACT_FRM_SIZE field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 CSI0_ACT_ FRM_HEIGHT	Actual frame height minus 1. This field defines the CSI output frame rows number minus 1.
15–13 Reserved	This read-only field is reserved and always has the value 0.
CSI0_ACT_ FRM_WIDTH	Actual frame width minus 1. This field defines the CSI output frame columns number minus 1.

37.5.154 CSI0 Output Control Register (IPUx_CSI0_OUT_FRM_CTRL)

Address: Base address + 3_000Ch offset

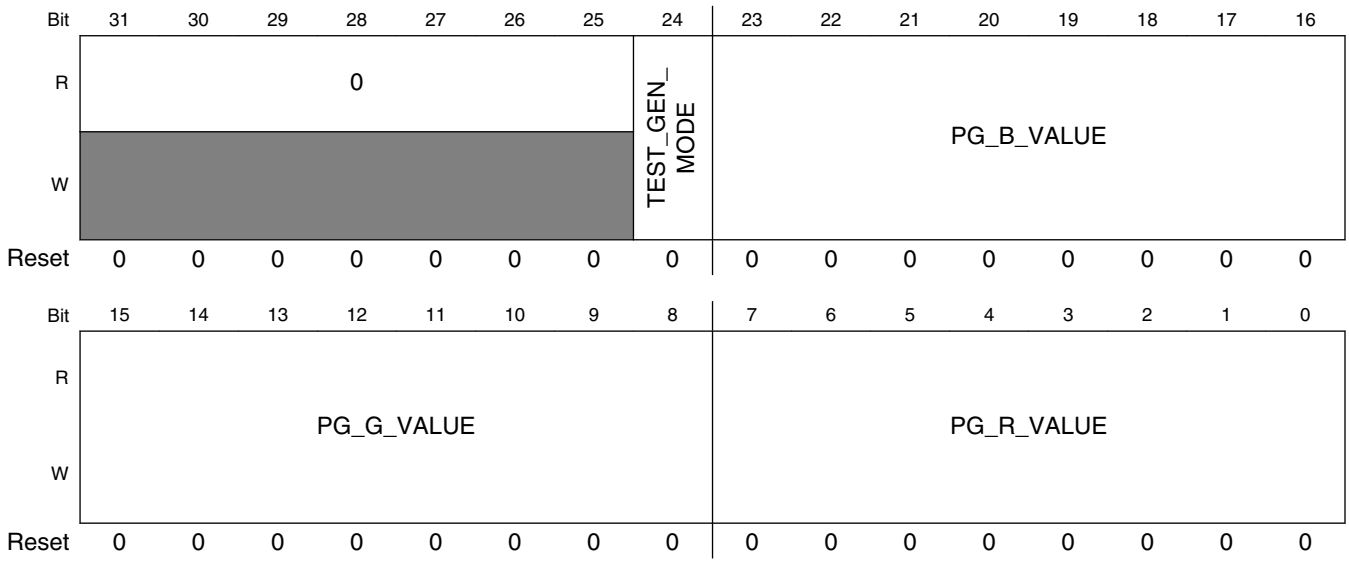


IPUx_CSI0_OUT_FRM_CTRL field descriptions

Field	Description
31 CSI0_HORZ_DWNS	Enable horizontal downsizing (decimation) by 2. 0 Downsizing disabled 1 Downsizing enabled
30 CSI0_VERT_DWNS	Enable vertical downsizing (decimation) by 2. 0 Downsizing disabled 1 Downsizing enabled
29 Reserved	This read-only field is reserved and always has the value 0.
28–16 CSI0_HSC	Horizontal skip. This field defines the number of columns to skip. In Interlaced mode this number refers to the number of lines per field.
15–12 Reserved	This read-only field is reserved and always has the value 0.
CSI0_VSC	Vertical skip. This field defines the number of rows to skip.

37.5.155 CSIO Test Control Register (IPUx_CSI0_TST_CTRL)

Address: Base address + 3_0010h offset

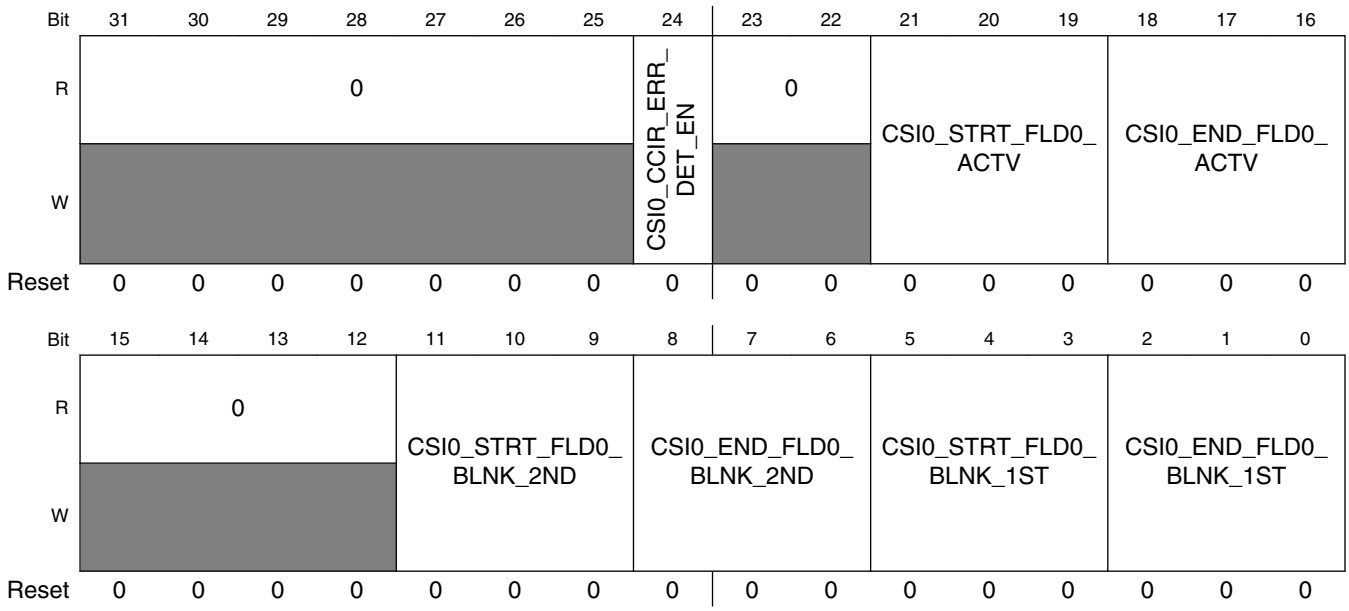


IPUx_CSI0_TST_CTRL field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 TEST_GEN_ MODE	Test generator mode. This bit activates the signal generation. 0 Test signal generator is inactive. 1 Test signal generator is active.
23–16 PG_B_VALUE	Pattern generator B value. This field selects the B value for the generated pattern of even pixel.
15–8 PG_G_VALUE	Pattern generator G value. This field selects the G value for the generated pattern of even pixel.
PG_R_VALUE	Pattern generator R value. This field selects the R value for the generated pattern of even pixel.

37.5.156 CSIO CCIR Code Register 1 (IPUx_CSI0_CCIR_CODE_1)

Address: Base address + 3_0014h offset



IPUx_CSI0_CCIR_CODE_1 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 CSI0_CCIR_ERR_DET_EN	Enable error detection and correction for CCIR interlaced mode with protection bit. 0 Error detection and correction is disabled. 1 Error detection and correction is enabled.
23–22 Reserved	This read-only field is reserved and always has the value 0.
21–19 CSI0_STRT_FLD0_ACTV	Start of field 0 active line command (interlaces mode). (In progressive mode, start of active line command mode).
18–16 CSI0_END_FLD0_ACTV	End of field 0 active line command (interlaces mode). (In progressive mode, end of active line command mode).
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 CSI0_STRT_FLD0_BLNK_2ND	Start of field 0 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
8–6 CSI0_END_FLD0_BLNK_2ND	End of field 0 second blanking line command (interlaces mode). (In progressive mode this field is ignored).

Table continues on the next page...

IPUx_CSI0_CCIR_CODE_1 field descriptions (continued)

Field	Description
5–3 CSI0_STRT_ FLD0_BLNK_ 1ST	Start of field 0 first blanking line command (interlaces mode). (In progressive mode this field indicates start of blanking line command).
CSI0_END_ FLD0_BLNK_ 1ST	End of field 0 first blanking line command (interlaces mode). (In progressive mode this field is ignored).

37.5.157 CSIO CCIR Code Register 2 (IPUx_CSI0_CCIR_CODE_2)

Address: Base address + 3_0018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0										CSI0_STRT_FLD1_ACTV			CSI0_END_FLD1_ACTV			0				CSI0_STRT_FLD1_BLNK_2ND			CSI0_END_FLD1_BLNK_2ND			CSI0_STRT_FLD1_BLNK_1ST			CSI0_END_FLD1_BLNK_1ST		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI0_CCIR_CODE_2 field descriptions

Field	Description
31–22 Reserved	This read-only field is reserved and always has the value 0.
21–19 CSI0_STRT_ FLD1_ACTV	Start of field 1 active line command (interlaces mode). (In progressive mode this field is ignored).
18–16 CSI0_END_ FLD1_ACTV	End of field 1 active line command (interlaces mode). (In progressive mode this field is ignored).
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 CSI0_STRT_ FLD1_BLNK_ 2ND	Start of field 1 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
8–6 CSI0_END_ FLD1_BLNK_ 2ND	End of field 1 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
5–3 CSI0_STRT_ FLD1_BLNK_ 1ST	Start of field 1 first blanking line command (interlaces mode). (In progressive mode this field is ignored).
CSI0_END_ FLD1_BLNK_ 1ST	End of field 1 first blanking line command (interlaces mode). (In progressive mode this field is ignored).

37.5.158 CSIO CCIR Code Register 3 (IPUx_CSI0_CCIR_CODE_3)

Address: Base address + 3_001Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								CSI0_CCIR_PRECOM																							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI0_CCIR_CODE_3 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CCIR_ PRECOM	CCIR pre command. This field defines the sequence which comes before the CCIR command. For BT.656 the code should be written to bits [23:0] while bits [29:24] are ignored (3X8bit) For BT.1120 the code should be written to bits [29:0] (3X10bit)

37.5.159 CSI0 Data Identifier Register (IPUx_CSI0_DI)

Address: Base address + 3_0020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CSI0_MIPI_DI3								CSI0_MIPI_DI2								CSI0_MIPI_DI1								CSI0_MIPI_DI0							
W																																
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

IPUx_CSI0_DI field descriptions

Field	Description
31–24 CSI0_MIPI_DI3	CSI0_MIPI_DI3 This field holds the Data Identifier #3 handled by the CSI.
23–16 CSI0_MIPI_DI2	CSI0_MIPI_DI2 This field holds the Data Identifier #2 handled by the CSI.
15–8 CSI0_MIPI_DI1	CSI0_MIPI_DI1 This field holds the Data Identifier #1 handled by the CSI
CSI0_MIPI_DI0	CSI0_MIPI_DI0 This field holds the Data Identifier #0 handled by the CSI; This is the main stream.

37.5.160 CSI0 SKIP Register (IPUx_CSI0_SKIP)

This register controls the frame skipping supported between CSI0 and the SMFC.

Address: Base address + 3_0024h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0						CSI0_ID_2_SKIP		CSI0_SKIP_SMFC					CSI0_MAX_RATIO_SKIP_SMFC		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI0_SKIP field descriptions

Field	Description
31–10 Reserved	This read-only field is reserved and always has the value 0.
9–8 CSI0_ID_2_SKIP	CSI0 to SMFC Skipping ID. Data from the CSI0 to the SMFC has an ID associated with it. The ID is received from the MIPI interface. The skipping mechanism between the CSI0 and the SMFC can be used for only one ID. There is no skipping for data coming with ID different from the ID programed in this bits 00 - Skipping mechanism is activated on frames with ID equal to 00 01 - Skipping mechanism is activated on frames with ID equal to 01 10 - Skipping mechanism is activated on frames with ID equal to 10 11 - Skipping mechanism is activated on frames with ID equal to 11
7–3 CSI0_SKIP_SMFC	CSI0 SKIP SMFC These 5 bits define the skipping pattern of the frames send to the SMFC. Skipping is done for a set of frames. The number of frames in a set is defined at CSI0_MAX_RATIO_SKIP_SMFC. when CSI0_MAX_RATIO_SKIP_SMFC = 1 => CSI0_SKIP_SMFC[0] is used; other bits are ignored when CSI0_MAX_RATIO_SKIP_SMFC = 2 => CSI0_SKIP_SMFC[1:0] are used; other bits are ignored when CSI0_MAX_RATIO_SKIP_SMFC = 3 => CSI0_SKIP_SMFC[2:0] are used; other bits are ignored when CSI0_MAX_RATIO_SKIP_SMFC = 4 => CSI0_SKIP_SMFC[3:0] are used; other bits are ignored when CSI0_MAX_RATIO_SKIP_SMFC = 5 => CSI0_SKIP_SMFC[4:0] are used; Setting bit #n of CSI0_SKIP_SMFC means that the #n frame in the set is skipped. For example: if CSI0_MAX_RATIO_SKIP_SMFC = 4 and CSI0_SKIP_SMFC = 11010 Frames #0 & Frame #2 will not be skipped as bit0 and bit2 are cleared Frames #1 & Frame #3 will be skipped as bit1 and bit3 are set bit #4 is ignored as CSI0_MAX_RATIO_SKIP_SMFC is set to 4

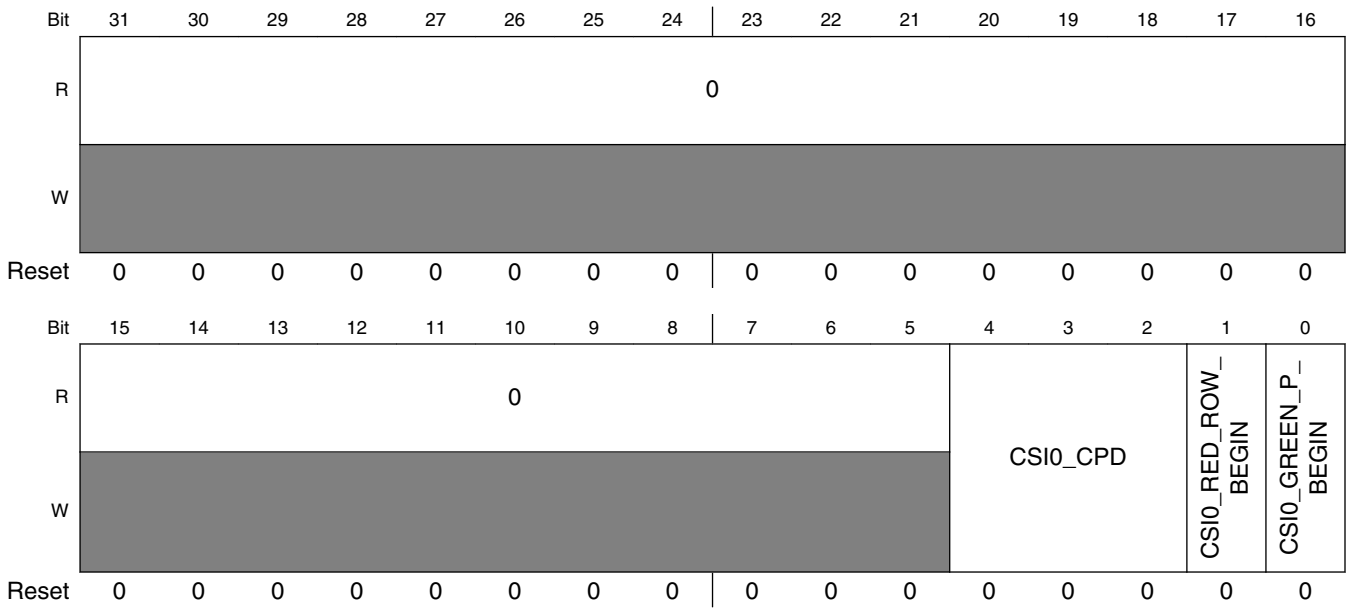
Table continues on the next page...

IPUx_CSI0_SKIP field descriptions (continued)

Field	Description
CSI0_MAX_ RATIO_SKIP_ SMFC	CSI0 Maximum Ratio Skip for SMFC These bits define the number of frames in a skipping set. The skipping number is equal to CSI0_MAX_RATIO_SKIP_SMFC+1; The maximum value of this bits is 5. When set to 0 the skipping is disabled.

37.5.161 CSI0 Compander Control Register (IPUx_CSIO_CPD_CTRL)

Address: Base address + 3_0028h offset



IPUx_CSIO_CPD_CTRL field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4–2 CSI0_CPD	CSI0_CPD These bits enable the compander in the path to different destination. CSI0_CPD[0] - CSI0_CPD[1] - Enable for the compander for data sent to the IC CSI0_CPD[2] - Enable for the compander for data sent to the IDMAC via SMFC If all the 3 bits are zero the compander is disabled Reserved
1 CSI0_RED_ ROW_BEGIN	Color of first row in the frame. Reserved

Table continues on the next page...

IPUx_CSIO_CPD_CTRL field descriptions (continued)

Field	Description
	0 First row in the frame is GBGB. 1 First row in the frame is GRGR.
0 CSI0_GREEN_P_BEGIN	Color of first component in the frame. Reserved 0 First component in the frame is blue or red, depending from RED_ROW bit. 1 First component in the frame is green

37.5.162 CSI0 Red Component Compander Constants Register <i> (IPUx_CSIO_CPD_RC_i)

These registers contain CONSTANT <i> parameters used for companding of red component.

Address: Base address + 3_002Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							CSI0_CPD_RC_2i_1									0							CSI0_CPD_RC_2i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_RC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI0_CPD_RC_2i_1	CONSTANT <2*i+1> parameter of Compander, Red component. Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CPD_RC_2i	CONSTANT <2*i> parameter of Compander, Red component. Reserved

37.5.163 CSI0 Red Component Compander SLOPE Register <i> (IPUx_CSIO_CPD_RS_i)

These registers contain SLOPE <i> parameters used for companding of red component.

Address: Base address + 3_004Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI0_CPD_RS_4i_3								CSI0_CPD_RS_4i_2								CSI0_CPD_RS_4i_1								CSI0_CPD_RS_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_RS_i field descriptions

Field	Description
31–24 CSI0_CPD_RS_4i_3	Reserved
23–16 CSI0_CPD_RS_4i_2	Reserved
15–8 CSI0_CPD_RS_4i_1	Reserved
CSI0_CPD_RS_4i	Reserved

37.5.164 CSI0 GR Component Compander Constants Register <i> (IPUx_CSIO_CPD_GRC_i)

These registers contain CONSTANTi parameters used for companding of green components in GRGR rows.

Address: Base address + 3_005Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0							CSI0_CPD_GRC_2i_1									0							CSI0_CPD_GRC_2i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_GRC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI0_CPD_ GRC_2i_1	Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CPD_ GRC_2i	Reserved

37.5.165 CSI0 GR Component Compander SLOPE Register <i> (IPUx_CSIO_CPD_GRS_i)

These registers contain SLOPEi parameters used for companding of green components in GRGR rows.

Address: Base address + 3_007Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI0_CPD_GRS_4i_3								CSI0_CPD_GRS_4i_2								CSI0_CPD_GRS_4i_1								CSI0_CPD_GRS_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_GRS_i field descriptions

Field	Description
31–24 CSI0_CPD_ GRS_4i_3	Reserved
23–16 CSI0_CPD_ GRS_4i_2	Reserved
15–8 CSI0_CPD_ GRS_4i_1	Reserved
CSI0_CPD_ GRS_4i	Reserved

37.5.166 CSI0 GB Component Compander Constants Register <i> (IPUx_CSIO_CPD_GBC_i)

These registers contain CONSTANTi parameters used for companding of green components in GBGB rows.

Address: Base address + 3_008Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							CSI0_CPD_GBC_2i_1									0							CSI0_CPD_GBC_2i							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_GBC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI0_CPD_ GBC_2i_1	Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CPD_ GBC_2i	Reserved

37.5.167 CSI0 GB Component Compander SLOPE Register <i> (IPUx_CSIO_CPD_GBS_i)

These registers contain SLOPEi parameters used for companding of green components in GBGB rows.

Address: Base address + 3_00ACh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CSI0_CPD_GBS_4i_3								CSI0_CPD_GBS_4i_2								CSI0_CPD_GBS_4i_1								CSI0_CPD_GBS_4i							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

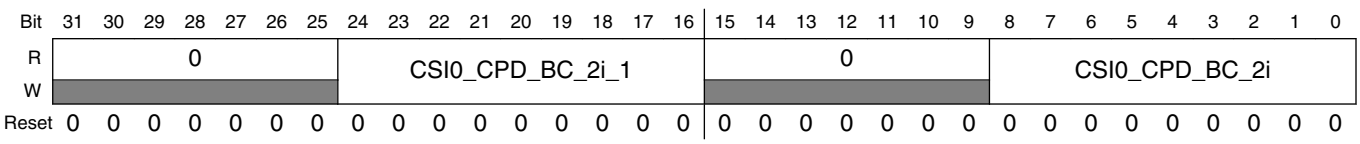
IPUx_CSIO_CPD_GBS_i field descriptions

Field	Description
31–24 CSI0_CPD_ GBS_4i_3	Reserved
23–16 CSI0_CPD_ GBS_4i_2	Reserved
15–8 CSI0_CPD_ GBS_4i_1	Reserved
CSI0_CPD_ GBS_4i	Reserved

37.5.168 CSI0 Blue Component Compander Constants Register <i> (IPUx_CSIO_CPD_BC_i)

These registers contain CONSTANTi parameters used for companding of blue component.

Address: Base address + 3_00BCh offset



IPUx_CSIO_CPD_BC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI0_CPD_BC_ 2i_1	Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CPD_BC_ 2i	Reserved

37.5.169 CSI0 Blue Component Compander SLOPE Register <i> (IPUx_CSIO_CPD_BS_i)

These registers contain SLOPEi parameters used for companding of red component.

Address: Base address + 3_00DCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI0_CPD_BS_4i_3								CSI0_CPD_BS_4i_2								CSI0_CPD_BS_4i_1								CSI0_CPD_BS_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSIO_CPD_BS_i field descriptions

Field	Description
31–24 CSI0_CPD_BS_4i_3	Reserved
23–16 CSI0_CPD_BS_4i_2	Reserved
15–8 CSI0_CPD_BS_4i_1	Reserved
CSI0_CPD_BS_4i	Reserved I

37.5.170 CSI0 Compander Offset Register 1 (IPUx_CSI0_CPD_OFFSET1)

These registers contain Offset parameters used for companding.

Address: Base address + 3_00ECh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0		CSI0_CPD_B_OFFSET										CSI0_GB_OFFSET										CSI0_GR_OFFSET									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI0_CPD_OFFSET1 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

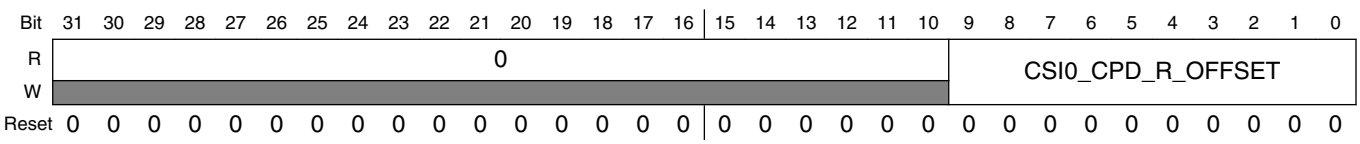
IPUx_CSI0_CPD_OFFSET1 field descriptions (continued)

Field	Description
29–20 CSI0_CPD_B_ OFFSET	Reserved
19–10 CSI0_GB_ OFFSET	Reserved
CSI0_GR_ OFFSET	Reserved

37.5.171 CSI0 Compander Offset Register 2 (IPUx_CSI0_CPD_OFFSET2)

This register contain Offset parameters used for companding.

Address: Base address + 3_00F0h offset



IPUx_CSI0_CPD_OFFSET2 field descriptions

Field	Description
31–10 Reserved	This read-only field is reserved and always has the value 0.
CSI0_CPD_R_ OFFSET	CSI0 Red component offset The value is between -512 to 511. The value is added to the red component before companding. Clipping: If the result of the red components value + the offset is smaller than 0, the result is zero If the result of the red components value + the offset is greater than 1023, the result is 1023 Reserved

37.5.172 CSI1 Sensor Configuration Register (IPUx_CSI1_SENS_CONF)

Address: Base address + 3_8000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	CSI0_DATA_EN_POL	0	CSI1_FORCE_EOF	CSI1_JPEG_MODE	CSI1_JPEG8_EN	CSI1_DATA_DEST			CSI1_DIV_RATIO							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CSI1_EXT_VSYNC	CSI1_DATA_WIDTH				CSI1_SENS_DATA_FORMAT			CSI1_PACK_TIGHT	CSI1_SENS_PRTCL			CSI1_SENS_PIX_CLK_POL	CSI1_DATA_POL	CSI1_HSYNC_POL	CSI1_VSYNC_POL
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_SENS_CONF field descriptions

Field	Description
31 CSI0_DATA_EN_POL	Invert IPP_IND_SENSB_DATA_EN input. This bit selects the polarity of IPP_IND_SENSB_DATA_EN signal. 0 IPP_IND_SENSB_DATA_EN is directly applied to internal circuitry. 1 IPP_IND_SENSB_DATA_EN is inverted before applied to internal circuitry.
30 Reserved	This read-only field is reserved and always has the value 0.
29 CSI1_FORCE_EOF	Force End of frame This is a self clear bit allowing the user to force an End-of-frame event; This bit can be used in cases where the frame sent by the sensor was not completed. 1 force end of frame 0 no action
28 CSI1_JPEG_MODE	JPEG Mode - this bit defines the mode of the control signals when working in JPEG mode 1 The data is valid as long as HSYNC and VSYNC signals are active; HSYNC is valid for single frame 0 The frame starts withe the assertion of VSYNC. The frame ends on the next VSYNC or by setting the **CSI0_FORCE_EOF bit**
27 CSI1_JPEG8_EN	JPEG8 enable bit 1 JPEG8 detection is enabled 0 JPEG8 is disabled

Table continues on the next page...

IPUx_CSI1_SENS_CONF field descriptions (continued)

Field	Description
26–24 CSI1_DATA_DEST	These bits enable the destination of the data coming from the CSI. CSI1_DATA_DEST[0] - Reserved CSI1_DATA_DEST[1] - destination is IC CSI1_DATA_DEST[2] - destination is IDMAC via SMFC
23–16 CSI1_DIV_RATIO	DIV Ratio Clock division ratio minus 1. This field defines the division ratio of HSP_CLK into SENSB_MCLK: SENSB_MCLK rate = HSP_CLK rate /(DIV_RATIO+1)
15 CSI1_EXT_VSYNC	External VSYNC enable. This bits select between external and internal VSYNC. 0 Internal VSYNC mode. 1 External VSYNC mode.
14–11 CSI1_DATA_WIDTH	Data width. This fields defines the number of bits per color. Values: 0000 4 bits per color 0000 Reserved 0001 8 bits per color 0010 9 bits per color 0010 Reserved 0011 10 bits per color 0100 11 bits per color 0100 Reserved 0101 12 bits per color 0101 Reserved 0110 13 bits per color 0110 Reserved 0111 14 bits per color 0111 Reserved 1000 15 bits per color 1000 Reserved 1001 16 bits per color
10–8 CSI1_SENS_DATA_FORMAT	Data format from the sensor. This field defines the data format for the input of the CSI sensor. Values: 000 full RGB or YUV444 001 YUV422 (YUYV...) 010 YUV422 (UYVY...) 011 Bayer or Generic data 100 RGB565 101 RGB555 110 RGB444 111 JPEG
7 CSI1_PACK_TIGHT	**CSI1 Pack Tight** When the data format is YUV or RGB and the component's width is 9-16 bits, it can be sent to the memory in 2 different ways

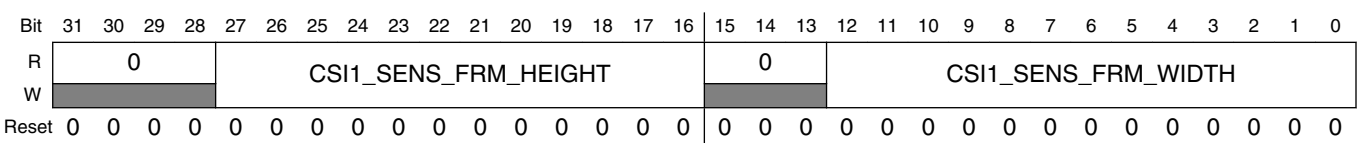
Table continues on the next page...

IPUx_CSI1_SENS_CONF field descriptions (continued)

Field	Description
	1 Three 10 bits components are packed into a 32 bit word. Color extension/reduction is performed 0 Each component is written as a 16 bit word where the MSB is written to bit #15, color extension is done for the remaining least significant bits.
6–4 CSI1_SENS_PRTCL	Sensor Protocol. This bit defines the Sensor timing/data mode protocol. Values: 000 Gated clock mode 001 Non-gated clock mode 010 CCIR progressive mode (BT.656) 011 CCIR interlaced mode (BT.656) 100 CCIR progressive (BT.1120 DDR mode: data arrives on every edge of the clock) 101 CCIR progressive (BT.1120 SDR mode: data arrives only on the positive edge of the clock) 110 CCIR interlaced mode (BT.1120 DDR mode: data arrives on every edge of the clock) 111 CCIR interlaced mode (BT.1120 SDR mode: data arrives only on the positive edge of the clock)
3 CSI1_SENS_PIX_CLK_POL	Invert Pixel clock input. This bit selects the polarity of pixel clock. 0 pixel clock is directly applied to internal circuitry. 1 pixel clock is inverted before applied to internal circuitry.
2 CSI1_DATA_POL	Invert data input. This bit selects the polarity of data input. 0 data lines are directly applied to internal circuitry. 1 data lines are inverted before applied to internal circuitry.
1 CSI1_HSYNC_POL	Invert IPP_IND_SENSB_HSYNC input. This bit selects the polarity of IPP_IND_SENSB_HSYNC signal. 0 IPP_IND_SENSB_HSYNC is directly applied to internal circuitry. 1 IPP_IND_SENSB_HSYNC is inverted before applied to internal circuitry.
0 CSI1_VSYNC_POL	Invert IPP_IND_SENSB_VSYNC input. This bit selects the polarity of IPP_IND_SENSB_VSYNC signal. 0 IPP_IND_SENSB_VSYNC is not inverted before applied to internal circuitry. 1 IPP_IND_SENSB_VSYNC is inverted before applied to internal circuitry.

37.5.173 CSI1 Sense Frame Size Register (IPUx_CSI1_SENS_FRM_SIZE)

Address: Base address + 3_8004h offset



IPUx_CSI1_SENS_FRM_SIZE field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_CSI1_SENS_FRM_SIZE field descriptions (continued)

Field	Description
27–16 CSI1_SENS_ FRM_HEIGHT	Sensor frame height minus 1. This field defines the sensor frame rows number minus 1.
15–13 Reserved	This read-only field is reserved and always has the value 0.
CSI1_SENS_ FRM_WIDTH	Sensor frame width minus 1. This field defines the sensor frame column number minus 1.

37.5.174 CSI1 Actual Frame Size Register (IPUx_CSI1_ACT_FRM_SIZE)

Address: Base address + 3_8008h offset

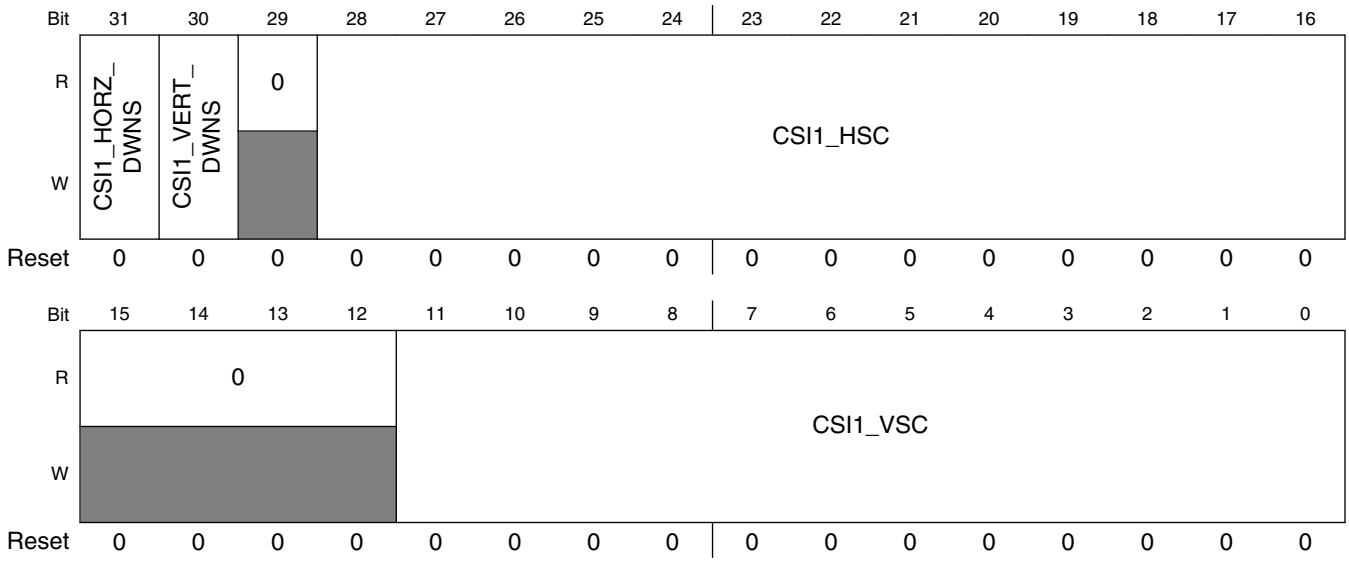
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				CSI1_ACT_FRM_HEIGHT												0			CSI1_ACT_FRM_WIDTH													
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_ACT_FRM_SIZE field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 CSI1_ACT_ FRM_HEIGHT	Actual frame height minus 1. This field defines the CSI output frame rows number minus 1.
15–13 Reserved	This read-only field is reserved and always has the value 0.
CSI1_ACT_ FRM_WIDTH	Actual frame width minus 1. This field defines the CSI output frame columns number minus 1.

37.5.175 CSI1 Output Control Register (IPUx_CSI1_OUT_FRM_CTRL)

Address: Base address + 3_800Ch offset

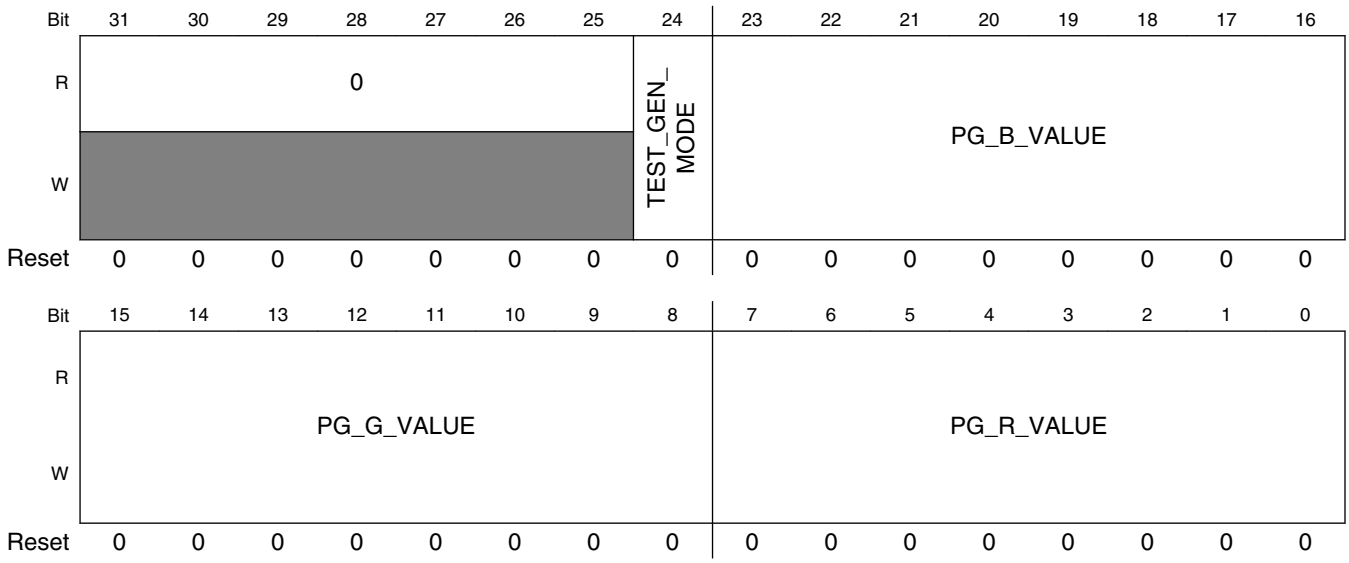


IPUx_CSI1_OUT_FRM_CTRL field descriptions

Field	Description
31 CSI1_HORZ_DWNS	Enable horizontal downsizing (decimation) by 2. 0 Downsizing disabled 1 Downsizing enabled
30 CSI1_VERT_DWNS	Enable vertical downsizing (decimation) by 2. 0 Downsizing disabled 1 Downsizing enabled
29 Reserved	This read-only field is reserved and always has the value 0.
28–16 CSI1_HSC	Horizontal skip. This field defines the number of columns to skip. In Interlaced mode this number refers to the number of lines per field
15–12 Reserved	This read-only field is reserved and always has the value 0.
CSI1_VSC	Vertical skip. This field defines the number of rows to skip.

37.5.176 CSI1 Test Control Register (IPUx_CSI1_TST_CTRL)

Address: Base address + 3_8010h offset

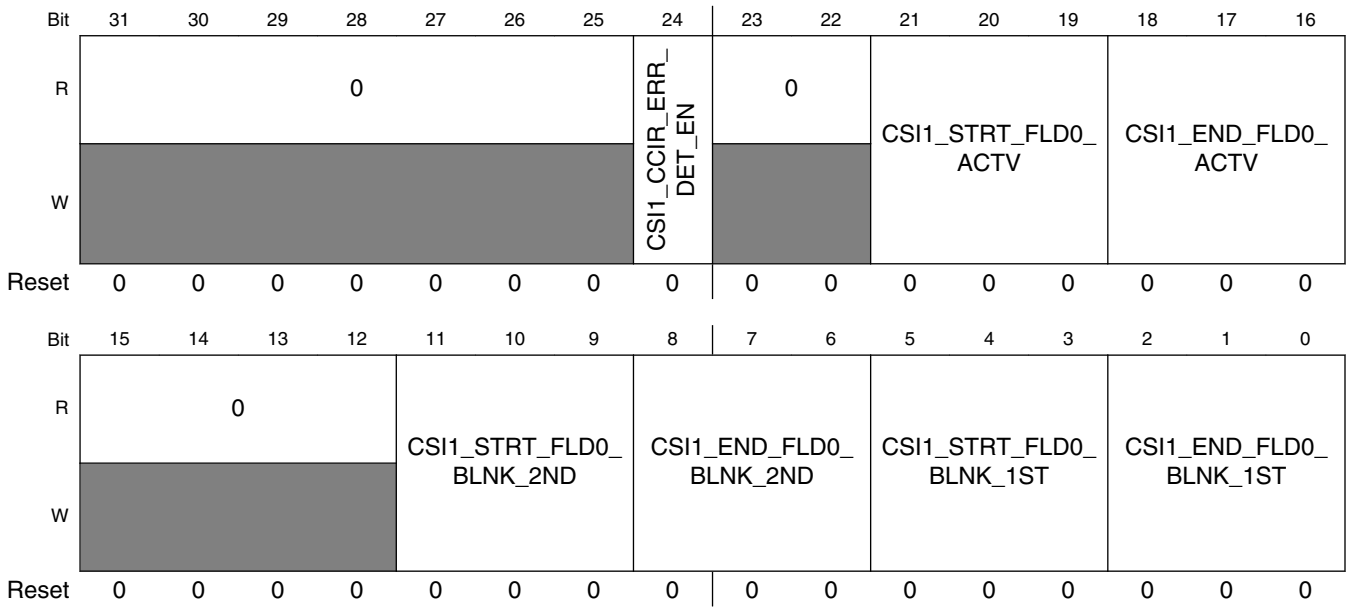


IPUx_CSI1_TST_CTRL field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 TEST_GEN_ MODE	Test generator mode. This bit activates the signal generation. 0 Test signal generator is inactive. 1 Test signal generator is active.
23–16 PG_B_VALUE	Pattern generator B value. This field selects the B value for the generated pattern of even pixel.
15–8 PG_G_VALUE	Pattern generator G value. This field selects the G value for the generated pattern of even pixel.
PG_R_VALUE	Pattern generator R value. This field selects the R value for the generated pattern of even pixel.

37.5.177 CSI1 CCIR Code Register 1 (IPUx_CSI1_CCIR_CODE_1)

Address: Base address + 3_8014h offset



IPUx_CSI1_CCIR_CODE_1 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 CSI1_CCIR_ERR_DET_EN	Enable error detection and correction for CCIR interlaced mode with protection bit. 0 Error detection and correction is disabled. 1 Error detection and correction is enabled.
23–22 Reserved	This read-only field is reserved and always has the value 0.
21–19 CSI1_STRT_FLD0_ACTV	Start of field 0 active line command (interlaces mode). (In progressive mode, start of active line command mode).
18–16 CSI1_END_FLD0_ACTV	End of field 0 active line command (interlaces mode). (In progressive mode, end of active line command mode).
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 CSI1_STRT_FLD0_BLNK_2ND	Start of field 0 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
8–6 CSI1_END_FLD0_BLNK_2ND	End of field 0 second blanking line command (interlaces mode). (In progressive mode this field is ignored).

Table continues on the next page...

IPUx_CSI1_CCIR_CODE_1 field descriptions (continued)

Field	Description
5–3 CSI1_STRT_ FLD0_BLNK_ 1ST	Start of field 0 first blanking line command (interlaces mode). (In progressive mode this field indicates start of blanking line command).
CSI1_END_ FLD0_BLNK_ 1ST	End of field 0 first blanking line command (interlaces mode). (In progressive mode this field is ignored).

37.5.178 CSI1 CCIR Code Register 2 (IPUx_CSI1_CCIR_CODE_2)

Address: Base address + 3_8018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0										CSI1_STRT_FLD1_ACTV			CSI1_END_FLD1_ACTV			0				CSI1_STRT_FLD1_BLNK_2ND			CSI1_END_FLD1_BLNK_2ND			CSI1_STRT_FLD1_BLNK_1ST			CSI1_END_FLD1_BLNK_1ST		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CCIR_CODE_2 field descriptions

Field	Description
31–22 Reserved	This read-only field is reserved and always has the value 0.
21–19 CSI1_STRT_ FLD1_ACTV	Start of field 1 active line command (interlaces mode). (In progressive mode this field is ignored).
18–16 CSI1_END_ FLD1_ACTV	End of field 1 active line command (interlaces mode). (In progressive mode this field is ignored).
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 CSI1_STRT_ FLD1_BLNK_ 2ND	Start of field 1 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
8–6 CSI1_END_ FLD1_BLNK_ 2ND	End of field 1 second blanking line command (interlaces mode). (In progressive mode this field is ignored).
5–3 CSI1_STRT_ FLD1_BLNK_ 1ST	Start of field 1 first blanking line command (interlaces mode). (In progressive mode this field is ignored).
CSI1_END_ FLD1_BLNK_ 1ST	End of field 1 first blanking line command (interlaces mode). (In progressive mode this field is ignored).

37.5.179 CSI1 CCIR Code Register 3 (IPUx_CSI1_CCIR_CODE_3)

Address: Base address + 3_801Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								CSI1_CCIR_PRECOM																							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CCIR_CODE_3 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CCIR_ PRECOM	CCIR pre command. This field defines the sequence which comes before the CCIR command. For BT.656 the code should be written to bits [23:0] while bits [29:24] are ignored (3X8bit) For BT.1120 the code should be written to bits [29:0] (3X10bit)

37.5.180 CSI1 Data Identifier Register (IPUx_CSI1_DI)

Address: Base address + 3_8020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CSI1_MIPI_DI3								CSI1_MIPI_DI2								CSI0_MIPI_DI1								CSI1_MIPI_DI0							
W																																
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

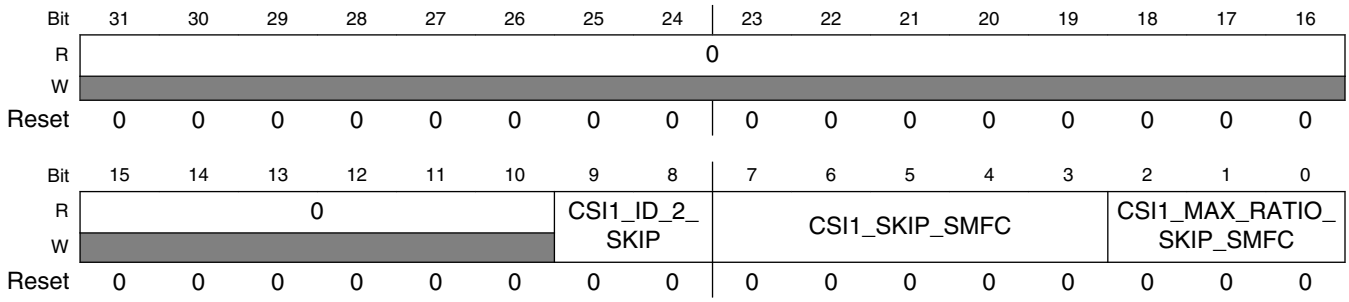
IPUx_CSI1_DI field descriptions

Field	Description
31–24 CSI1_MIPI_DI3	CSI1_MIPI_DI3 This field holds the Data Identifier #3 handled by the CSI.
23–16 CSI1_MIPI_DI2	CSI1_MIPI_DI2 This field holds the Data Identifier #2 handled by the CSI.
15–8 CSI0_MIPI_DI1	CSI1_MIPI_DI1 This field holds the Data Identifier #1 handled by the CSI
CSI1_MIPI_DI0	CSI1_MIPI_DI0 This field holds the Data Identifier #0 handled by the CSI; This is the main stream.

37.5.181 CSI1 SKIP Register (IPUx_CSI1_SKIP)

This register control the frame skipping supported between CSI1 and the SMFC.

Address: Base address + 3_8024h offset



IPUx_CSI1_SKIP field descriptions

Field	Description
31–10 Reserved	This read-only field is reserved and always has the value 0.
9–8 CSI1_ID_2_SKIP	CSI1 to SMFC Skipping ID. Data from the CSI1 to the SMFC has an ID associated with it. The ID is received from the MIPI interface. The skipping mechanism between the CSI1 and the SMFC can be used for only one ID. There is no skipping for data coming with ID different from the ID programed in this bits 00 - Skipping mechanism is activated on frames with ID equal to 00 01 - Skipping mechanism is activated on frames with ID equal to 01 10 - Skipping mechanism is activated on frames with ID equal to 10 11 - Skipping mechanism is activated on frames with ID equal to 11
7–3 CSI1_SKIP_SMFC	CSI1 SKIP SMFC These 5 bits define the skipping pattern of the frames send to the SMFC. Skipping is done for a set of frames. The number of frames in a set is defined at CSI1_MAX_RATIO_SKIP_SMFC. when CSI1_MAX_RATIO_SKIP_SMFC = 1 => CSI1_SKIP_SMFC[0] is used; other bits are ignored when CSI1_MAX_RATIO_SKIP_SMFC = 2 => CSI1_SKIP_SMFC[1:0] are used; other bits are ignored when CSI1_MAX_RATIO_SKIP_SMFC = 3 => CSI1_SKIP_SMFC[2:0] are used; other bits are ignored when CSI1_MAX_RATIO_SKIP_SMFC = 4 => CSI1_SKIP_SMFC[3:0] are used; other bits are ignored when CSI1_MAX_RATIO_SKIP_SMFC = 5 => CSI1_SKIP_SMFC[4:0] are used; Setting bit #n of CSI1_SKIP_SMFC means that the #n frame in the set is skipped. For example: if CSI1_MAX_RATIO_SKIP_SMFC = 4 and CSI1_SKIP_SMFC = 11010 Frames #0 & Frame #2 will not be skipped as bit0 and bit2 are cleared Frames #1 & Frame #3 will be skipped as bit1 and bit3 are set bit #4 is ignored as CSI1_MAX_RATIO_SKIP_SMFC is set to 4

Table continues on the next page...

IPUx_CSI1_SKIP field descriptions (continued)

Field	Description
CSI1_MAX_ RATIO_SKIP_ SMFC	CSI1 Maximum Ratio Skip for SMFC These bits define the number of frames in a skipping set. These bits define the number of frames in a skipping set. The skipping number is equal to CSI1_MAX_RATIO_SKIP_SMFC+1; The maximum value of this bits is 5. When set to 0 the skipping is disabled.

37.5.182 CSI1 Compander Control Register (IPUx_CSI1_CPD_CTRL)

Address: Base address + 3_8028h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											0			0	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

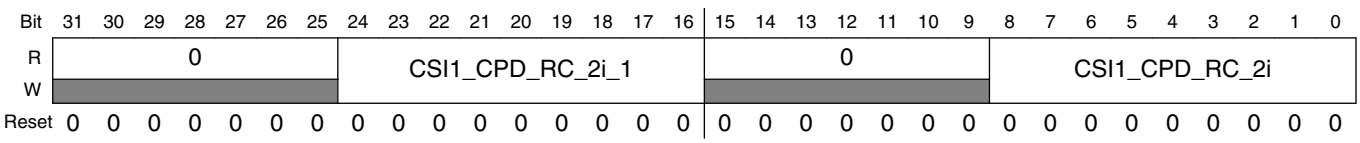
IPUx_CSI1_CPD_CTRL field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4–2 Reserved	This read-only field is reserved and always has the value 0.
1 Reserved	This read-only field is reserved and always has the value 0.
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.183 CSI1 Red Component Compander Constants Register <i> (IPUx_CSI1_CPD_RC_i)

These registers contain CONSTANT <i> parameters used for companding of red component.

Address: Base address + 3_802Ch offset



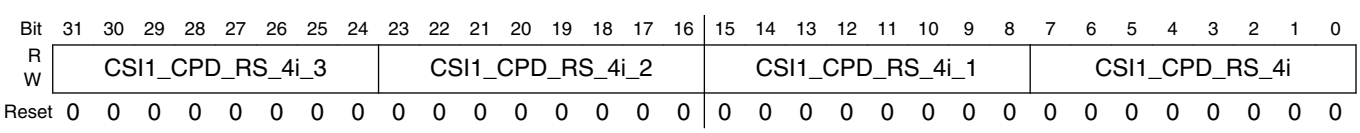
IPUx_CSI1_CPD_RC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI1_CPD_RC_2i_1	CONSTANT <2*i+1> parameter of Compander, Red component. Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CPD_RC_2i	CONSTANT <2*i> parameter of Compander, Red component. Reserved

37.5.184 CSI1 Red Component Compander SLOPE Register <i> (IPUx_CSI1_CPD_RS_i)

These registers contain SLOPE <i> parameters used for companding of red component.

Address: Base address + 3_804Ch offset



IPUx_CSI1_CPD_RS_i field descriptions

Field	Description
31–24 CSI1_CPD_RS_4i_3	SLOPE<4*i+3> parameter of Compander, Red component. Reserved
23–16 CSI1_CPD_RS_4i_2	SLOPE<4*i+2> parameter of Compander, Red component. Reserved
15–8 CSI1_CPD_RS_4i_1	SLOPE<4*i+1> parameter of Compander, Red component. Reserved
CSI1_CPD_RS_4i	SLOPE<4*i> parameter of Compander, Red component. Reserved

37.5.185 CSI1 GR Component Compander Constants Register <i> (IPUx_CSI1_CPD_GRC_i)

These registers contain CONSTANTi parameters used for companding of green components in GRGR rows.

Address: Base address + 3_805Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							CSI1_CPD_GRC_2i_1									0							CSI1_CPD_GRC_2i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_GRC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI1_CPD_GRC_2i_1	CONST<2*i+1> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRC should be equal to CSI1_CPD_GBC Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CPD_GRC_2i	CONSTANT<2*i> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRC should be equal to CSI1_CPD_GBC Reserved

37.5.186 CSI1 GR Component Compander SLOPE Register <i> (IPUx_CSI1_CPD_GRS_i)

These registers contain SLOPEi parameters used for companding of green components in GRGR rows.

Address: Base address + 3_807Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI1_CPD_GRS_4i_3								CSI1_CPD_GRS_4i_2								CSI1_CPD_GRS_4i_1								CSI1_CPD_GRS_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_GRS_i field descriptions

Field	Description
31–24 CSI1_CPD_ GRS_4i_3	SLOPE<4*i+3> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRS should be equal to CSI1_CPD_GBS Reserved
23–16 CSI1_CPD_ GRS_4i_2	SLOPE<4*i+2> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRS should be equal to CSI1_CPD_GBS Reserved
15–8 CSI1_CPD_ GRS_4i_1	SLOPE<4*i+1> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRS should be equal to CSI1_CPD_GBS Reserved
CSI1_CPD_ GRS_4i	SLOPE<4*i> parameter of Compander, GR component. If the input format is RGB/YUV then CSI1_CPD_GRS should be equal to CSI1_CPD_GBS Reserved

37.5.187 CSI1 GB Component Compander Constants Register <i> (IPUx_CSI1_CPD_GBC_i)

These registers contain CONSTANTi parameters used for companding of green components in GBGB rows.

Address: Base address + 3_808Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	0								CSI1_CPD_GBC_2i_1								0								CSI1_CPD_GBC_2i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_GBC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI1_CPD_GBC_2i_1	CONSTi+1 parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBC should be equal to CSI1_CPD_GRC Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CPD_GBC_2i	CONSTANTi parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBC should be equal to CSI1_CPD_GRC Reserved

37.5.188 CSI1 GB Component Compander SLOPE Register <i> (IPUx_CSI1_CPD_GBS_i)

These registers contain SLOPEi parameters used for companding of green components in GBGB rows.

Address: Base address + 3_80ACh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CSI1_CPD_GBS_4i_3								CSI1_CPD_GBS_4i_2								CSI1_CPD_GBS_4i_1								CSI1_CPD_GBS_4i							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_GBS_i field descriptions

Field	Description
31–24 CSI1_CPD_GBS_4i_3	SLOPE<4*i+3> parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBS should be equal to CSI1_CPD_GRS Reserved
23–16 CSI1_CPD_GBS_4i_2	SLOPE<4*i+2> parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBS should be equal to CSI1_CPD_GRS Reserved
15–8 CSI1_CPD_GBS_4i_1	SLOPE<4*i+1> parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBS should be equal to CSI1_CPD_GRS Reserved
CSI1_CPD_GBS_4i	SLOPE<4*i> parameter of Compander, GB component. If the input format is RGB/YUV then CSI1_CPD_GBS should be equal to CSI1_CPD_GRS Reserved

37.5.189 CSI1 Blue Component Compander Constants Register <i> (IPUx_CSI1_CPD_BC_i)

These registers contend CONSTANTi parameters used for companding of blue component.

Address: Base address + 3_80BCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							CSI1_CPD_BC_2i_1									0							CSI1_CPD_BC_2i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_BC_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 CSI1_CPD_BC_2i_1	CONSTANT<2*i+1> parameter of Compander, Blue component. Reserved
15–9 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CPD_BC_2i	CONSTANT<2*i> parameter of Compander, Blue component. Reserved

37.5.190 CSI1 Blue Component Compander SLOPE Register <i> (IPUx_CSI1_CPD_BS_i)

This registers contain SLOPEi parameters used for companding of red component.

Address: Base address + 3_80DCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CSI1_CPD_BS_4i_3								CSI1_CPD_BS_4i_2								CSI1_CPD_BS_4i_1								CSI1_CPD_BS_4i							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_BS_i field descriptions

Field	Description
31–24 CSI1_CPD_BS_4i_3	SLOPE<4*i+3> parameter of Compander, Blue component. Reserved
23–16 CSI1_CPD_BS_4i_2	SLOPE<4*i+2> parameter of Compander, Blue component. Reserved
15–8 CSI1_CPD_BS_4i_1	SLOPE<4*i+1> parameter of Compander, Blue component. Reserved
CSI1_CPD_BS_4i	SLOPE<4*i> parameter of Compander, Blue component. Reserved

37.5.191 CSI1 Compander Offset Register 1 (IPUx_CSI1_CPD_OFFSET1)

These registers contain Offset parameters used for companding.

Address: Base address + 3_80ECh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0		CSI1_CPD_B_OFFSET										CSI1_CPD_GB_OFFSET										CSI1_CPD_GR_OFFSET									
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_CSI1_CPD_OFFSET1 field descriptions

Field	Description
31–30 Reserved	This read-only field is reserved and always has the value 0.
29–20 CSI1_CPD_B_OFFSET	CSI1 Blue component offset The value is between -512 to 511. The value is added to the blue component before companding. Clipping: If the result of the blue components value + the offset is smaller than 0, the result is zero If the result of the blue components value + the offset is greater than 1023, the result is 1023 Reserved
19–10 CSI1_CPD_GB_OFFSET	CSI1 Green Blue component offset The value is between -512 to 511. The value is added to the blue component before companding. Clipping: If the result of the green-blue components value + the offset is smaller than 0, the result is zero If the result of the green-blue components value + the offset is greater than 1023, the result is 1023

Table continues on the next page...

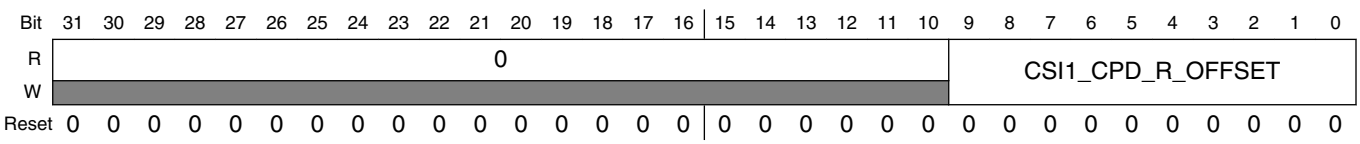
IPUx_CSI1_CPD_OFFSET1 field descriptions (continued)

Field	Description
	If the input format is RGB/YUV then CSI1_GB_OFFSET must be equal to CSI1_GR_OFFSET Reserved
CSI1_CPD_GR_OFFSET	CSI1 Green Red component offset The value is between -512 to 511. The value is added to the green-red component before companding. Clipping: If the result of the green-red components value + the offset is smaller than 0, the result is zero If the result of the green-red components value + the offset is greater than 1023, the result is 1023 Reserved

37.5.192 CSI1 Compander Offset Register 2 (IPUx_CSI1_CPD_OFFSET2)

These registers contain Offset parameters used for companding.

Address: Base address + 3_80F0h offset



IPUx_CSI1_CPD_OFFSET2 field descriptions

Field	Description
31–10 Reserved	This read-only field is reserved and always has the value 0.
CSI1_CPD_R_OFFSET	CSI1 Red component offset The value is between -512 to 511. The value is added to the red component before companding. Clipping: If the result of the red components value + the offset is smaller than 0, the result is zero If the result of the red components value + the offset is greater than 1023, the result is 1023 Reserved

37.5.193 DI0 General Register (IPUx_DI0_GENERAL)

Address: Base address + 4_0000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	di0_pin8_pin15_sel	di0_disp_y_sel			DI0_CLOCK_STOP_MODE				DI0_DISP_CLOCK_INIT	di0_mask_sel	di0_vsync_ext	di0_clk_ext	DI0_WATCHDOG_MODE		di0_polarity_disp_clk	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	di0_sync_count_sel				di0_err_treatment	di0_erm_vsync_sel	di0_polarity_cs1	di0_polarity_cs0	di0_polarity_i_1							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_GENERAL field descriptions

Field	Description
31 di0_pin8_pin15_sel	This bit routes PIN8 over PIN15 1 PIN8 is routed to PIN15, PIN8 is also routed to PIN8 0 PIN15 is routed to PIN15, PIN8 is routed to PIN8
30–28 di0_disp_y_sel	DI0 Display Vertical coordinate (Y) select. This field defines which one of the 8 counters will be used as a display's line counter. 000 counter #1 is selected 111 counter #8 is selected
27–24 DI0_CLOCK_STOP_MODE	DI clock stop mode When performing a clock change. The DI stops the clock to the display. These field defines when the clock will be stopped. Stopping at EOL/EOF is supported for the case where the data is coming from the IDMAC (DMA access). In case that only direct accesses is performed, the user should set this field to 0000 0001-1001 stop at the next event of one of the counters (counter #1 to counter #9) 0000 stop at the next edge of the display clock 1100 stop at EOL (end of a line), but if stop request is during blanking interval, stop now 1101 stop at EOF (end of a frame), but if stop request is during blanking interval, stop now 1110 stop at EOL (end of a line), but if stop request is during blanking interval, stop at the end of the next line 1111 stop at EOF (end of a frame), but if stop request is during blanking interval, stop at the end of the next frame

Table continues on the next page...

IPUx_DI0_GENERAL field descriptions (continued)

Field	Description
23 DI0_DISP_CLOCK_INIT	Display clock's initial mode For synchronization error conditions the display clock can be stopped on the next VSYNC 1 The display's clock is running after the next VSYNC (indicating new frame) 0 The display's clock is stopped after the next VSYNC (indicating new frame)
22 di0_mask_sel	DI0 Mask select. IPP_PIN_2 output of the DI that functions as MASK signal can come from 2 sources: counter #2 or extracted from the MASK data coming from the memory. 1 IPP_PIN_2 is coming from extracted MASK data coming from the memory 0 IPP_PIN_2 is coming from counter #2
21 di0_vsync_ext	DI0 External VSYNC. This bit selects the source of the VSYNC signal 1 External to the IPU 0 Internally generated by the IPU
20 di0_clk_ext	DI0 External Clock. This bit selects the source of the DI0's clock 1 The source of the clock is external to the IPU 0 The clock is internally generated by the IPU
19–18 DI0_WATCHDOG_MODE	DI0 watchdog mode In case of a display error where the DI clock is stopped (defined at di0_err_treatment). An internal watchdog counts DI clocks. If this timer reached its pre defined value the DI will skip the current frame and restart on the frame. This 2 bits define the number of DI clock cycles that the timer counts. 00 The timer counts 4 DI cycles 01 The timer counts 16 DI cycles 10 The timer counts 64 DI cycles 11 The timer counts 128 DI cycles
17 di0_polarity_disp_clk	DI0 Output Clock's polarity This bits define the polarity of the DI0's clock. 1 The output clock is active high 0 The output clock is active low
16 Reserved	This read-only field is reserved and always has the value 0.
15–12 di0_sync_count_sel	For synchronous flow error: selects synchronous flow synchronization counter in DI:
11 di0_err_treatment	In case of synchronous flow error there are 2 ways to handle the display 1 to wait (stop clock) 0 Drive the last component
10 di0_erm_vsync_sel	DI0 error recovery block's VSYNC source select The error recovery block detect a case where the DI's VSYNC is asserted before the EOF. This bit selects the source of the VSYNC signal monitored by this mechanism.

Table continues on the next page...

IPUx_DI0_GENERAL field descriptions (continued)

Field	Description
	1 vsync_post - an internal VSYNC signal asserted 2 lines after the DI's VSYNC 0 vsync_pre - an internal VSYNC signal asserted 2 lines before the DI's VSYNC
9 di0_polarity_cs1	DI0 Chip Select's 1 polarity This bits define the polarity of the DI's CS1. 1 The CS1 is active high 0 The CS1 is active low
8 di0_polarity_cs0	DI0 Chip Select's 0 polarity This bits define the polarity of the DI's CS0. 1 The CS0 is active high 0 The CS0 is active low
di0_polarity_i_1	DI0 output pin's polarity This bits define the polarity of each of the DI's outputs. 1 The output pin is active high 0 The output pin is active low

37.5.194 DI0 Base Sync Clock Gen 0 Register (IPUx_DI0_BS_CLKGEN0)

Address: Base address + 4_0004h offset

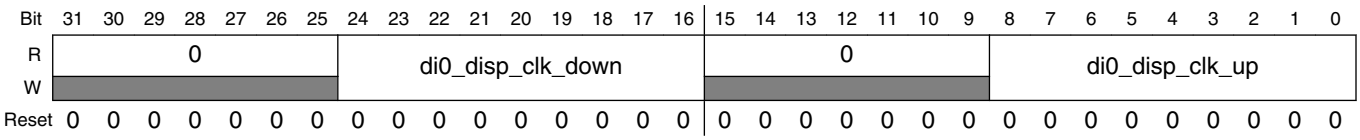
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di0_disp_clk_offset									0				di0_disp_clk_period											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_BS_CLKGEN0 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_disp_clk_ offset	DI0 Display Clock Offset The DI has the ability to delay the display's clock This field defines the amount of IPU's clock cycles added as delay on this clock.
15–12 Reserved	This read-only field is reserved and always has the value 0.
di0_disp_clk_ period	DI0 Display Clock Period This field defines the Display interface clock period for display write access. This parameter contains an integer part (bits 11:4) and a fractional part (bits 3:0). It defines a fractional division ratio of the HSP_CLK clock for generation of the display's interface clock.

37.5.195 DI0 Base Sync Clock Gen 1 Register (IPUx_DI0_BS_CLKGEN1)

Address: Base address + 4_0008h offset

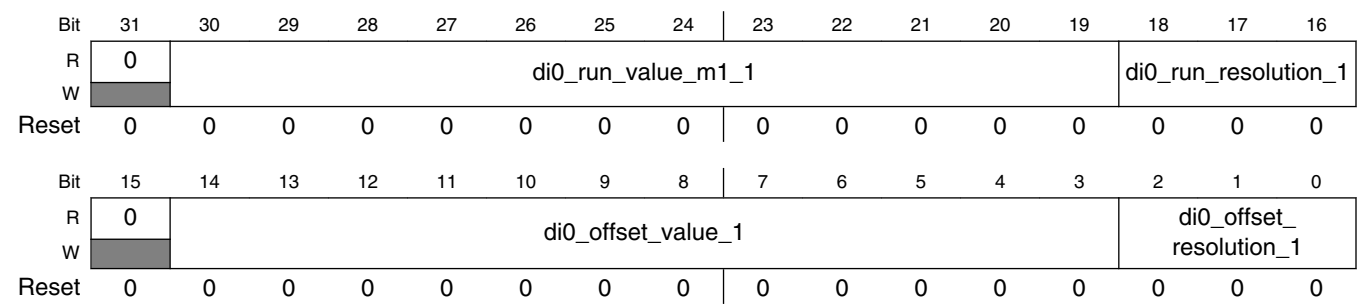


IPUx_DI0_BS_CLKGEN1 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_disp_clk_down	DI0 display clock falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is a time interval between display's access start point and display's interface clock falling edge.
15–9 Reserved	This read-only field is reserved and always has the value 0.
di0_disp_clk_up	DI0 display clock rising edge position This parameter contains an integer part (bits 8:1) and a fractional part (bit 0). The position value is a time interval between display's access start point and display's interface clock rising edge.

37.5.196 DI0 Sync Wave Gen 1 Register 0 (IPUx_DI0_SW_GEN0_1)

Address: Base address + 4_000Ch offset



IPUx_DI0_SW_GEN0_1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.

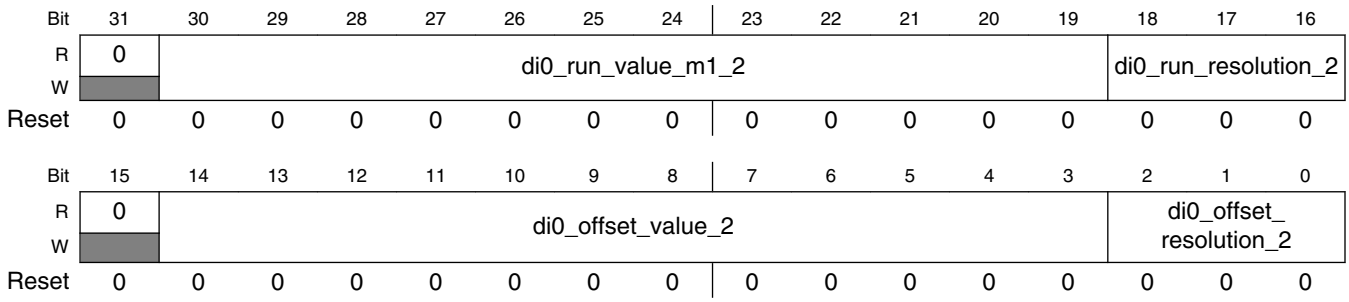
Table continues on the next page...

IPUx_DI0_SW_GEN0_1 field descriptions (continued)

Field	Description
30–19 di0_run_value_m1_1	DI0 counter #1 pre defined value This fields defines the counter #1 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_1 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_resolution_1	DI0 counter #1 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 NA 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSIs according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_1	DI0 counter #1 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_resolution_1	DI0 counter #1 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock 010 NA 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSIs according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.197 DI0 Sync Wave Gen 2 Register 0 (IPUx_DI0_SW_GEN0_2)

Address: Base address + 4_0010h offset



IPUx_DI0_SW_GEN0_2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_m1_2	DI0 counter #2 pre defined value This fields defines the counter #2 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_2 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_resolution_2	DI0 counter #2 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock 010 The Counter is triggered by counter #1 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_2	DI0 counter #2 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_resolution_2	DI0 counter #2 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

Table continues on the next page...

IPUx_DI0_SW_GEN0_2 field descriptions (continued)

Field	Description
	011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSIs according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.198 DI0 Sync Wave Gen 3 Register 0 (IPUx_DI0_SW_GEN0_3)

Address: Base address + 4_0014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_3												di0_run_resolution_3		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_3												di0_offset_ resolution_3		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_ m1_3	DI0 counter #3 pre defined value This fields defines the counter #3 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_3 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_ resolution_3	DI0 counter #3 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

Table continues on the next page...

IPUx_DI0_SW_GEN0_3 field descriptions (continued)

Field	Description
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_3	DI0 counter #3 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_resolution_3	DI0 counter #3 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.199 DI0 Sync Wave Gen 4 Register 0 (IPUx_DI0_SW_GEN0_4)

Address: Base address + 4_0018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_4												di0_run_resolution_4		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_4												di0_offset_resolution_4		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_4 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_m1_4	DI0 counter #4 pre defined value This fields defines the counter #4 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_4 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.

Table continues on the next page...

IPUx_DI0_SW_GEN0_4 field descriptions (continued)

Field	Description
18–16 di0_run_ resolution_4	DI0 counter #4 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_ 4	DI0 counter #4 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_ resolution_4	DI0 counter #4 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.200 DI0 Sync Wave Gen 5 Register 0 (IPUx_DI0_SW_GEN0_5)

Address: Base address + 4_001Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_5												di0_run_resolution_5		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_5												di0_offset_ resolution_5		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_5 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_m1_5	DI0 counter #5 pre defined value This fields defines the counter #5 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_5 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_resolution_5	DI0 counter #5 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_5	DI0 counter #5 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_resolution_5	DI0 counter #5 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 -The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.

37.5.201 DI0 Sync Wave Gen 6 Register 0 (IPUx_DI0_SW_GEN0_6)

Address: Base address + 4_0020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_6												di0_run_resolution_6		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_6												di0_offset_ resolution_6		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_6 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_ m1_6	DI0 counter #6 pre defined value This fields defines the counter #6 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_6 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_ resolution_6	DI0 counter #6 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_ 6	DI0 counter #6 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_ resolution_6	DI0 counter #6 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3

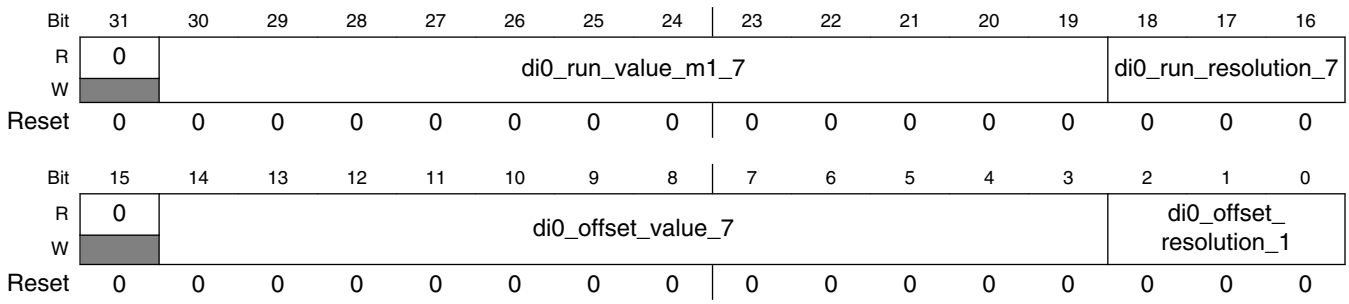
Table continues on the next page...

IPUx_DI0_SW_GEN0_6 field descriptions (continued)

Field	Description
	101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.202 DI0 Sync Wave Gen 7 Register 0 (IPUx_DI0_SW_GEN0_7)

Address: Base address + 4_0024h offset



IPUx_DI0_SW_GEN0_7 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_ m1_7	DI0 counter #7 pre defined value This fields defines the counter #7 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_7 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_ resolution_7	DI0 counter #1 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_ 7	DI0 counter #7 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by

Table continues on the next page...

IPUx_DI0_SW_GEN0_7 field descriptions (continued)

Field	Description
di0_offset_ resolution_1	DI0 counter #7 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.203 DI0 Sync Wave Gen 8 Register 0 (IPUx_DI0_SW_GEN0_8)

Address: Base address + 4_0028h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_8												di0_run_resolution_8		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_8												di0_offset_ resolution_8		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_8 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_ m1_8	DI0 counter #8 pre defined value This fields defines the counter #8 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_8 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.
18–16 di0_run_ resolution_8	DI0 counter #8 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4

Table continues on the next page...

IPUx_DI0_SW_GEN0_8 field descriptions (continued)

Field	Description
	110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_8	DI0 counter #8 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_resolution_8	DI0 counter #8 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.204 DI0 Sync Wave Gen 9 Register 0 (IPUx_DI0_SW_GEN0_9)

Address: Base address + 4_002Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di0_run_value_m1_9												di0_run_resolution_9		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di0_offset_value_9												di0_offset_resolution_9		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SW_GEN0_9 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di0_run_value_m1_9	DI0 counter #9 pre defined value This fields defines the counter #9 pre defines value. When the counter is auto reload mode (di0_cnt_auto_reload_9 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di0_step_repeat field.

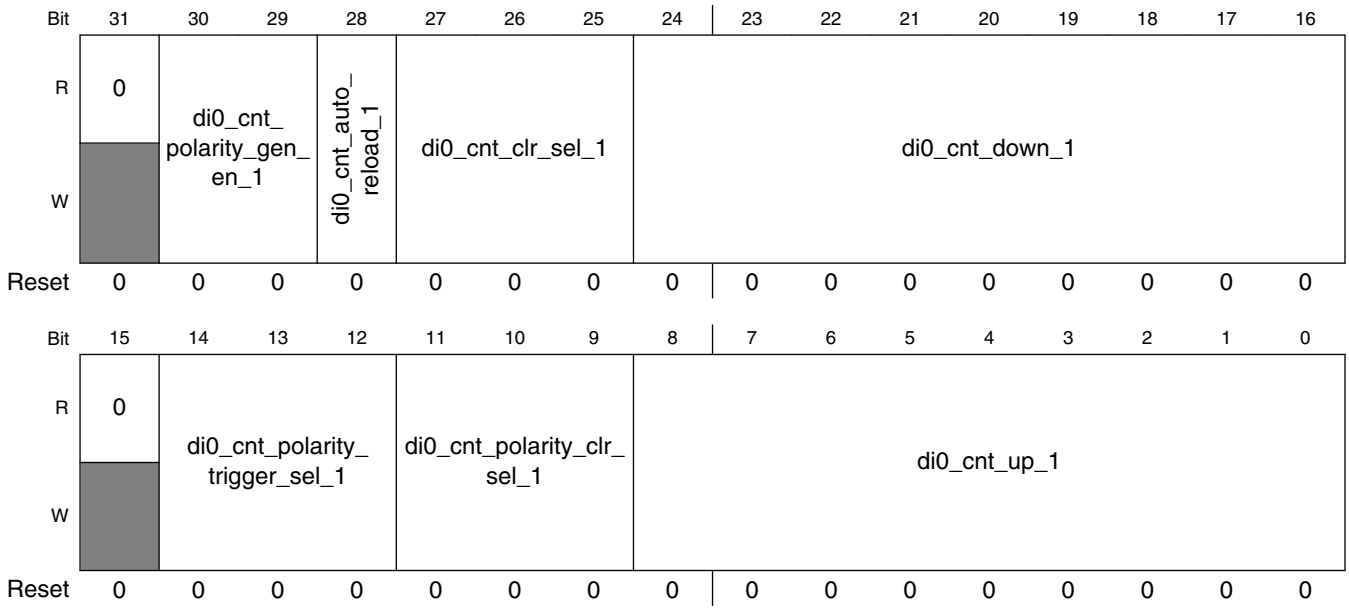
Table continues on the next page...

IPUx_DI0_SW_GEN0_9 field descriptions (continued)

Field	Description
18–16 di0_run_ resolution_9	DI0 counter #9 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di0_offset_value_ 9	DI0 counter #9 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di0_offset_ resolution_9	DI0 counter #9 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.205 DI0 Sync Wave Gen 1 Register 1 (IPUx_DI0_SW_GEN1_1)

Address: Base address + 4_0030h offset



IPUx_DI0_SW_GEN1_1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_1	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_1	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_1 field
27–25 di0_cnt_clr_sel_1	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 Reserved

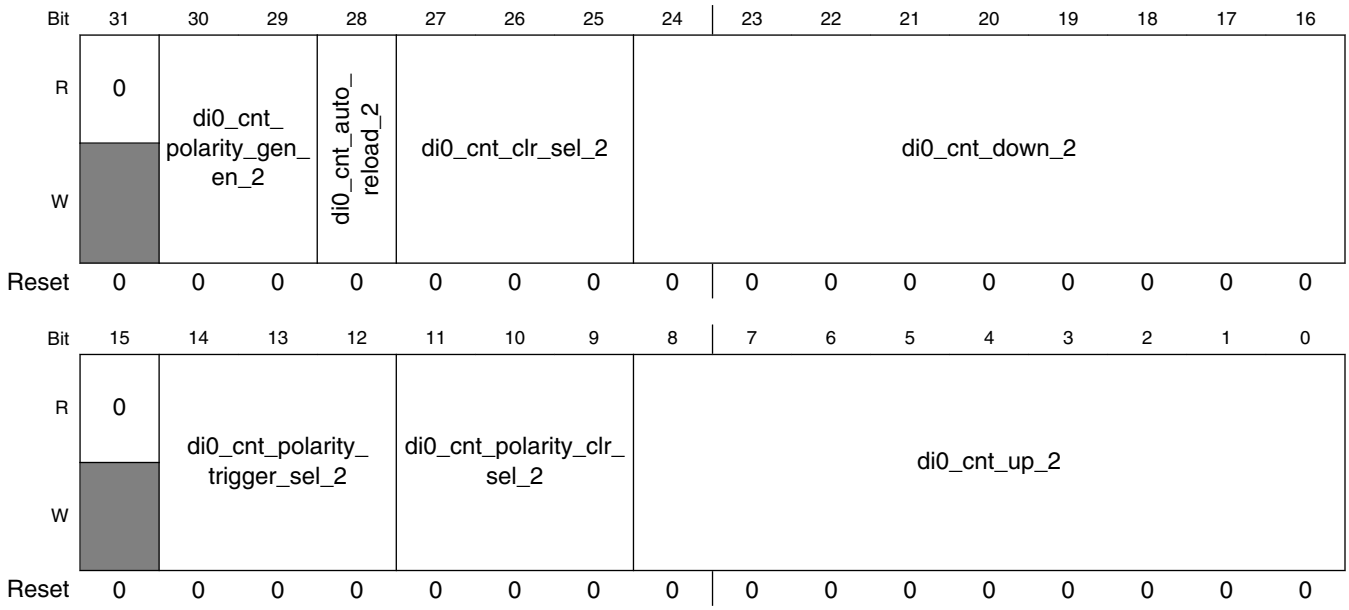
Table continues on the next page...

IPUx_DI0_SW_GEN1_1 field descriptions (continued)

Field	Description
	011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di0_cnt_down_1	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_1	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 Reserved 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_1	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Reserved 011 Reserved 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di0_cnt_up_1	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.206 DI0 Sync Wave Gen 2 Register 1 (IPUx_DI0_SW_GEN1_2)

Address: Base address + 4_0034h offset



IPUx_DI0_SW_GEN1_2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_2	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_2	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_2	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

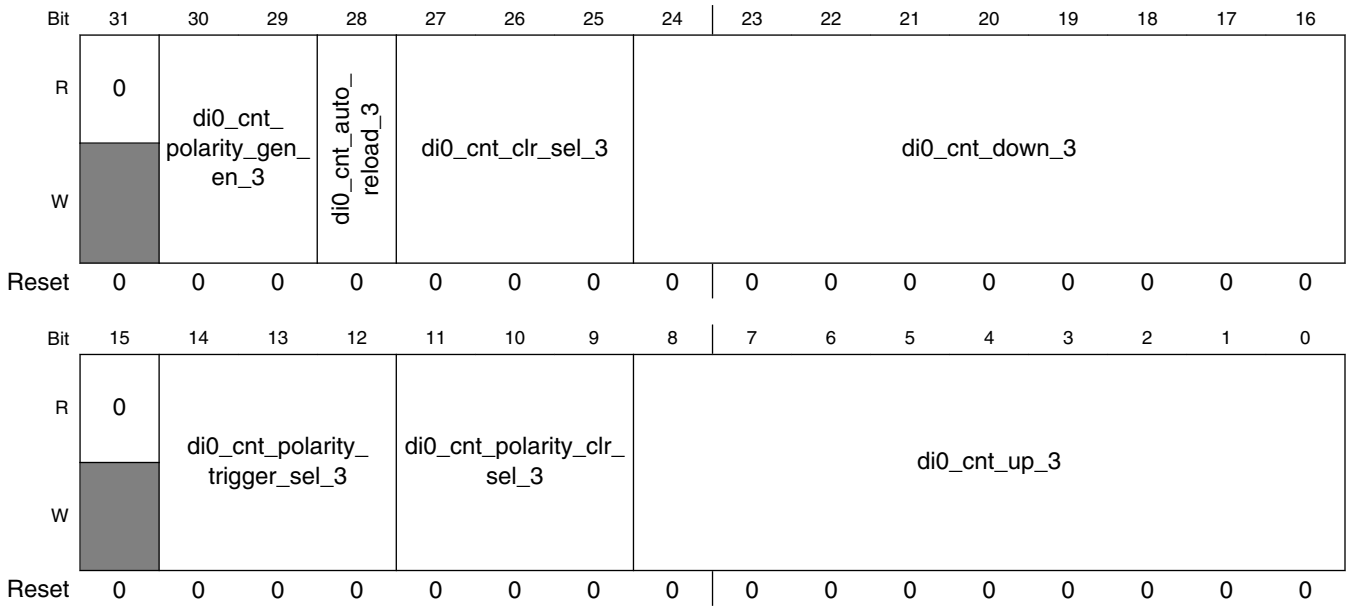
Table continues on the next page...

IPUx_DI0_SW_GEN1_2 field descriptions (continued)

Field	Description
	011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di0_cnt_down_2	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_2	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_2	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Reserved 011 Reserved 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di0_cnt_up_2	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.207 DI0 Sync Wave Gen 3 Register 1 (IPUx_DI0_SW_GEN1_3)

Address: Base address + 4_0038h offset



IPUx_DI0_SW_GEN1_3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_3	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_3	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_3	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

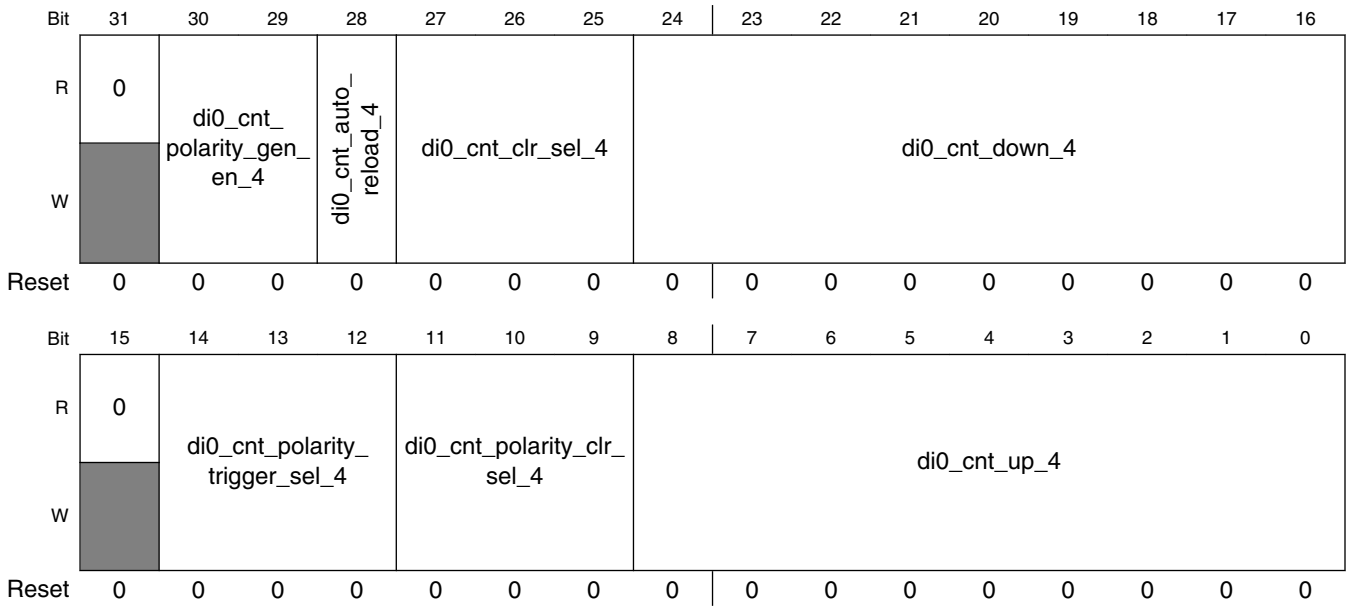
Table continues on the next page...

IPUx_DI0_SW_GEN1_3 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di0_cnt_down_3	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_3	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_3	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di0_cnt_up_3	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.208 DI0 Sync Wave Gen 4 Register 1 (IPUx_DI0_SW_GEN1_4)

Address: Base address + 4_003Ch offset



IPUx_DI0_SW_GEN1_4 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_4	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_4	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_4	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

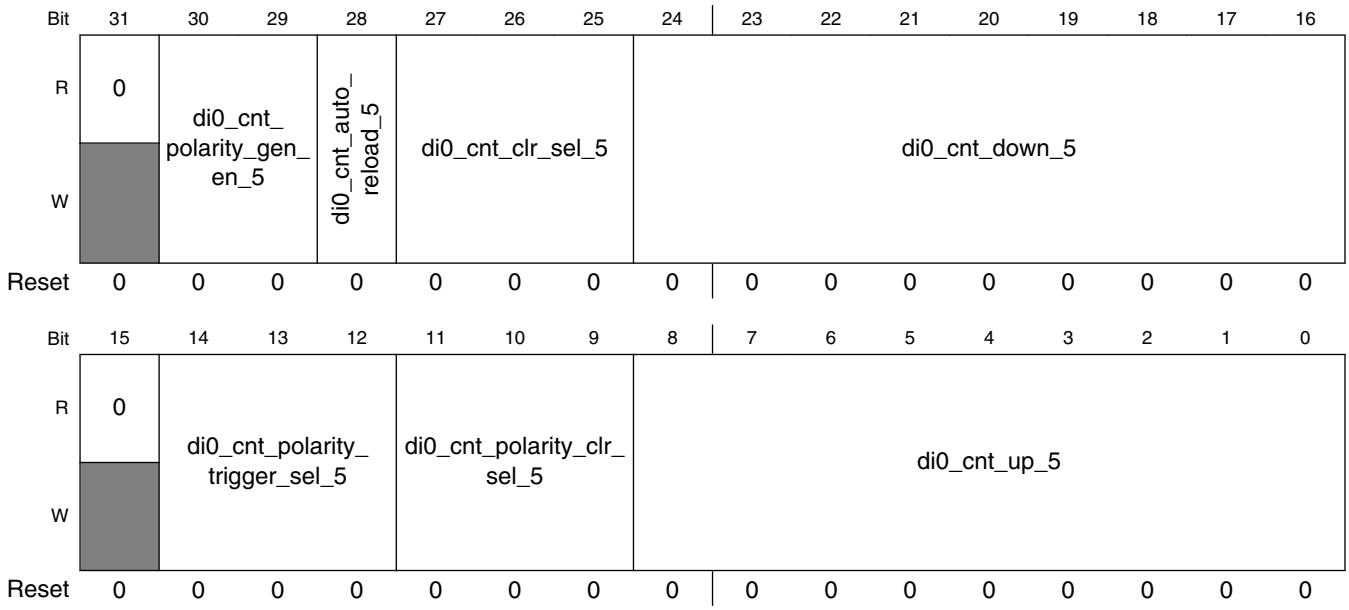
Table continues on the next page...

IPUx_DI0_SW_GEN1_4 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di0_cnt_down_4	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_4	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_4	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Reserved 110 Reserved 111 Reserved
di0_cnt_up_4	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.209 DI0 Sync Wave Gen 5 Register 1 (IPUx_DI0_SW_GEN1_5)

Address: Base address + 4_0040h offset



IPUx_DI0_SW_GEN1_5 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_5	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_5	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_5	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

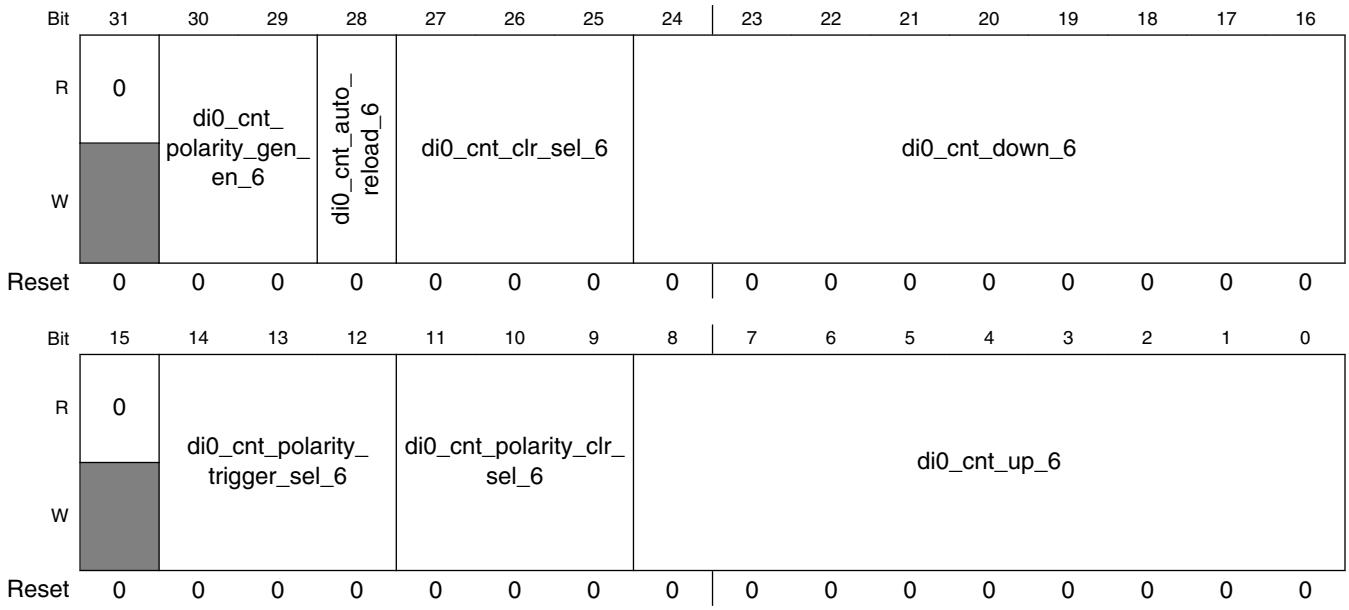
Table continues on the next page...

IPUx_DI0_SW_GEN1_5 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
24–16 di0_cnt_down_5	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_5	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_5	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Reserved 111 Reserved
di0_cnt_up_5	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.210 DI0 Sync Wave Gen 6 Register 1 (IPUx_DI0_SW_GEN1_6)

Address: Base address + 4_0044h offset



IPUx_DI0_SW_GEN1_6 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_6	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_6	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_6	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

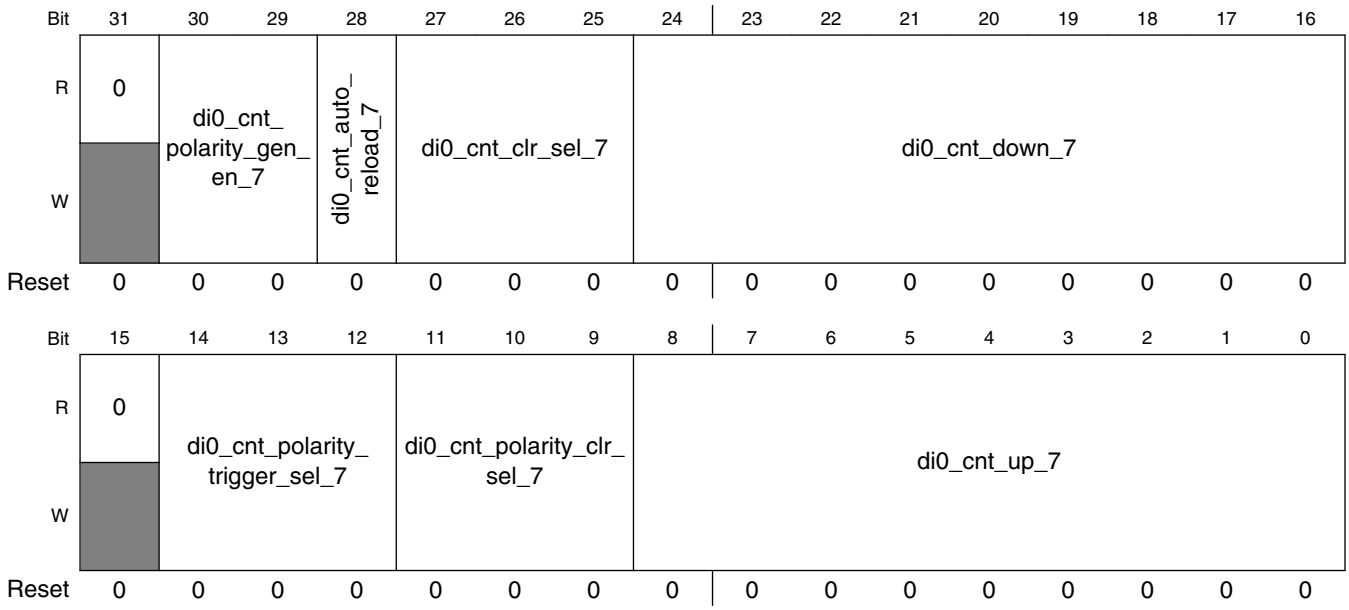
Table continues on the next page...

IPUx_DI0_SW_GEN1_6 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di0_cnt_down_6	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_6	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_6	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Reserved
di0_cnt_up_6	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.211 DI0 Sync Wave Gen 7 Register 1 (IPUx_DI0_SW_GEN1_7)

Address: Base address + 4_0048h offset



IPUx_DI0_SW_GEN1_7 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_7	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_7	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_7	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

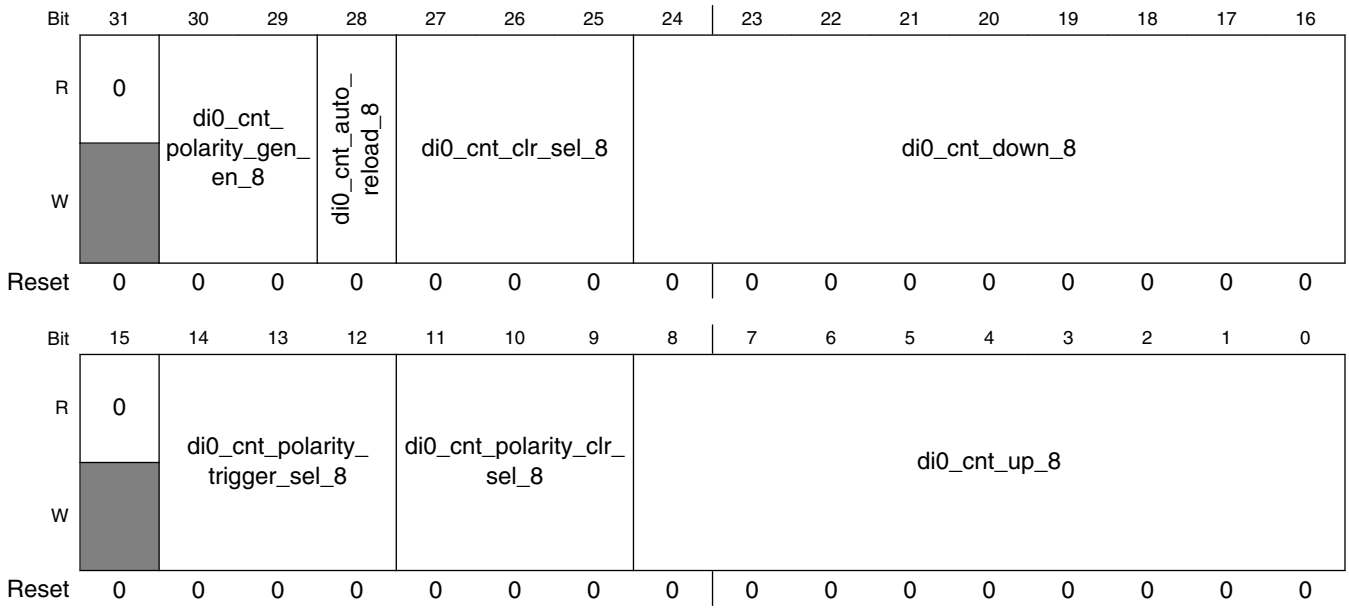
Table continues on the next page...

IPUx_DI0_SW_GEN1_7 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di0_cnt_down_7	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_7	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_7	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Output is inverted if the output of counter #6 is set
di0_cnt_up_7	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.212 DI0 Sync Wave Gen 8 Register 1 (IPUx_DI0_SW_GEN1_8)

Address: Base address + 4_004Ch offset



IPUx_DI0_SW_GEN1_8 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di0_cnt_polarity_gen_en_8	DI0 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di0_cnt_auto_reload_8	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_8	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

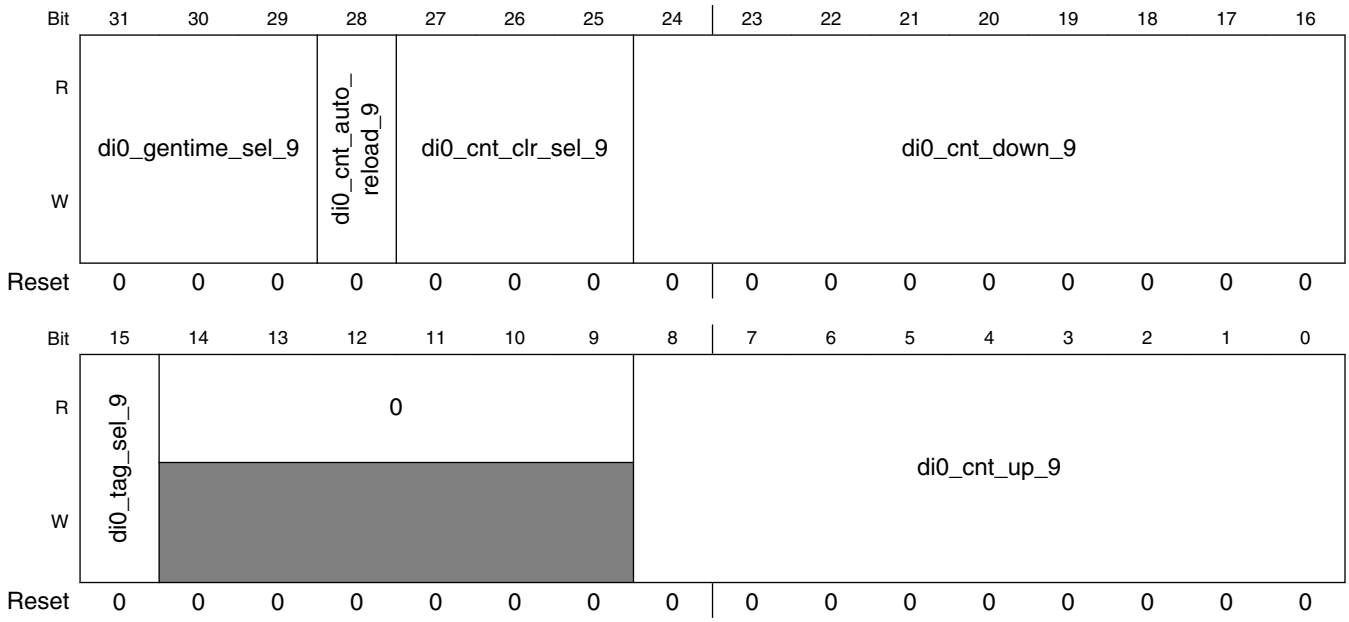
Table continues on the next page...

IPUx_DI0_SW_GEN1_8 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di0_cnt_down_8	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di0_cnt_polarity_trigger_sel_8	DI0 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di0_cnt_polarity_clr_sel_8	DI0 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Output is inverted if the output of counter #6 is set
di0_cnt_up_8	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.213 DI0 Sync Wave Gen 9 Register 1 (IPUx_DI0_SW_GEN1_9)

Address: Base address + 4_0050h offset



IPUx_DI0_SW_GEN1_9 field descriptions

Field	Description
31–29 di0_gentime_sel_9	Counter #9 main waveform select This field defines the counter that counter #9's auxiliary waveform will be attached too. 000 Counter #9's waveform is attached to counter #1's waveform 001 Counter #9's waveform is attached to counter #2's waveform 010 Counter #9's waveform is attached to counter #3's waveform 011 Counter #9's waveform is attached to counter #4's waveform 100 Counter #9's waveform is attached to counter #5's waveform 101 Counter #9's waveform is attached to counter #6's waveform 110 Counter #9's waveform is attached to counter #7's waveform 111 Counter #9's waveform is attached to counter #8's waveform
28 di0_cnt_auto_reload_9	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di0_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di0_step_repeat_<i> field
27–25 di0_cnt_clr_sel_9	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

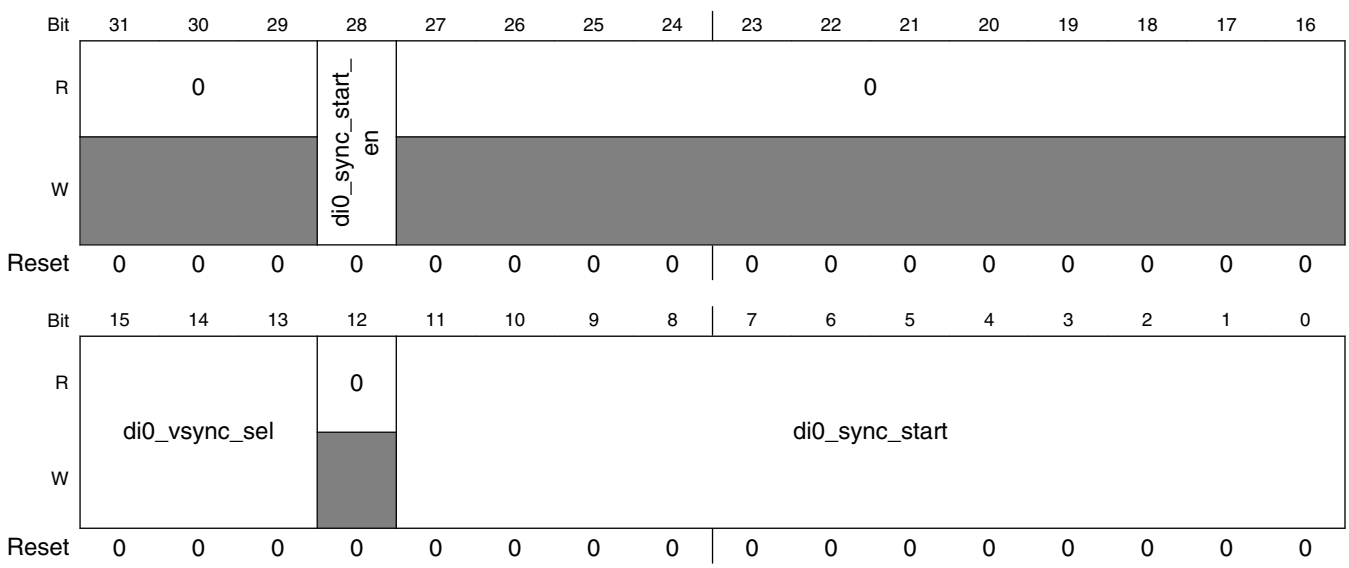
Table continues on the next page...

IPUx_DI0_SW_GEN1_9 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di0_cnt_down_9	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 di0_tag_sel_9	Counter #9 can send a synchronous tag when counter #9 reach its predefined value or when it's triggering counter reaches its pre defined value. 1 tag source is counter #9 0 Tag's source is the triggering counter.
14–9 Reserved	This read-only field is reserved and always has the value 0.
di0_cnt_up_9	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.214 DI0 Sync Assistance Gen Register (IPUx_DI0_SYNC_AS_GEN)

Address: Base address + 4_0054h offset



IPUx_DI0_SYNC_AS_GEN field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28 di0_sync_start_en	di0_sync_start_en
27–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 di0_vsync_sel	VSYNC select This field defines which of the counters functions as VSYNC signal 000 VSYNC is coming from counter #1 001 VSYNC is coming from counter #2 111 VSYNC is coming from counter #8
12 Reserved	This read-only field is reserved and always has the value 0.
di0_sync_start	DI0 Sync start This field defines the number of low (including blanking rows) on the which the DI0 starts preparing the data for the next frame.

37.5.215 DI0 Data Wave Gen <i> Register (IPUx_DI0_DW_GEN_i)

The DI0_DW_GEN_<i> register holds pointers for the waveform generators.

These registers have different bit arrangements for parallel and serial display. When using a parallel display VDI Plane Size Register 4 is applicable. When using a serial interface VDI Plane Size Register 4 is applicable.

Table 37-667. Register Field Descriptions for Serial Display

Field	Description
31-24 di0_serial_period_<i>	DI0 Serial Period <i> This field defines the period of the time base serial display clock. The units are the internal DI clock
23-16 di0_start_period_<i>	DI0 start period This field defines the amount of cycles between the point where the access is ready to be launched to the actual point where the time base serial display clock restarts. The units are the internal DI clock
15-14 di0_cst_<i>	DI0 Chip Select pointer for waveform <i> This field points to a register that defines the waveform of the CS pin. For serial displays the down value as defined on DI0_DW_SET*_<i> is measured from the assertion of the last serial display time base clock. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i>

Table continues on the next page...

Table 37-667. Register Field Descriptions for Serial Display (continued)

Field	Description
	10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
13-9	Reserved
8-4 di0_serial_valid_bits<i>	DI0 Serial valid bits. This field defines the amount of valid bits to be transmitted within the 32 bits internal word aligned to bit[0]. The actual amount of valid bits is di0_serial_valid_bits_<i> + 1
3-2 di0_serial_rs_<i>	DI0 Serial RS This field points to a register that defines the waveform of the RS pin. For serial displays the down value as defined on DI0_DW_SET*_<i> is measured from the assertion of the last serial display time base clock. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
1-0 di0_serial_clk_<i>	DI0 serial clock<i> This field points to a register that defines the waveform of the Serial clock pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>

Address: Base address + 4_0058h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	di0_access_size_i								di0_componnent_size_i								di0_cst_i		di0_pt_6_i		di0_pt_5_i		di0_pt_4_i		di0_pt_3_i		di0_pt_2_i		di0_pt_1_i		di0_pt_0_i	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_DW_GEN_i field descriptions

Field	Description
31–24 di0_access_size_i	DI0 Access Size <i> This field defines the amount of IPU cycles between any 2 accesses (an access may be a pixel or generic data that may have more one component)
23–16 di0_componnent_size_i	DI0 component Size This field defines the amount of IPU cycles between any 2 components
15–14 di0_cst_i	DI0 Chip Select pointer for waveform <i> This field points to a register that defines the waveform of the CS pin. The CS is automatically mapped to a specific display

Table continues on the next page...

IPUx_DI0_DW_GEN_i field descriptions (continued)

Field	Description
	00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
13–12 di0_pt_6_i	DI0 PIN_17 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_17 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
11–10 di0_pt_5_i	DI0 PIN_16 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_16 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
9–8 di0_pt_4_i	DI0 PIN_15 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_15 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
7–6 di0_pt_3_i	DI0 PIN_14 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_14 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
5–4 di0_pt_2_i	DI0 PIN_13 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_13 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>
3–2 di0_pt_1_i	DI0 PIN_12 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_12 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>

Table continues on the next page...

IPUx_DI0_DW_GEN_i field descriptions (continued)

Field	Description
di0_pt_0_i	DI0 PIN_11 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_11 pin. 00 The waveform is defined according to the settings on DI0_DW_SET0_<i> 01 The waveform is defined according to the settings on DI0_DW_SET1_<i> 10 The waveform is defined according to the settings on DI0_DW_SET2_<i> 11 The waveform is defined according to the settings on DI0_DW_SET3_<i>

37.5.216 DI0 Data Wave Set 0 <i> Register (IPUx_DI0_DW_SET0_i)

Address: Base address + 4_0088h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di0_data_cnt_down0_i									0							di0_data_cnt_up0_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_DW_SET0_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_data_cnt_down0_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a point according to the corresponding di0_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di0_data_cnt_up0_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a point according to the corresponding di0_pt_*_<i>

37.5.217 DI0 Data Wave Set 1 <i> Register (IPUx_DI0_DW_SET1_i)

Address: Base address + 4_00B8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di0_data_cnt_down1_i									0							di0_data_cnt_up1_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_DW_SET1_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_data_cnt_down1_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di0_data_cnt_up1_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>

37.5.218 DI0 Data Wave Set 2 <i> Register (IPUx_DI0_DW_SET2_i)

Address: Base address + 4_00E8h offset

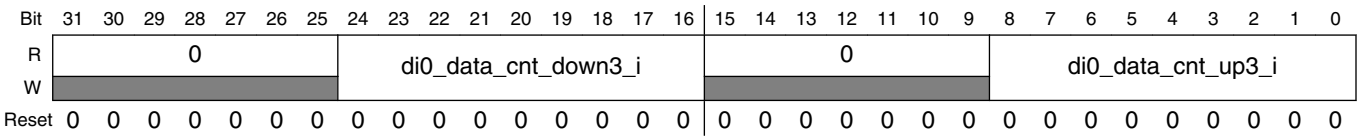
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di0_data_cnt_down2_i									0							di0_data_cnt_up2_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_DW_SET2_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_data_cnt_down2_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di0_data_cnt_up2_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>

37.5.219 DI0 Data Wave Set 3 <i> Register (IPUx_DI0_DW_SET3_i)

Address: Base address + 4_0118h offset

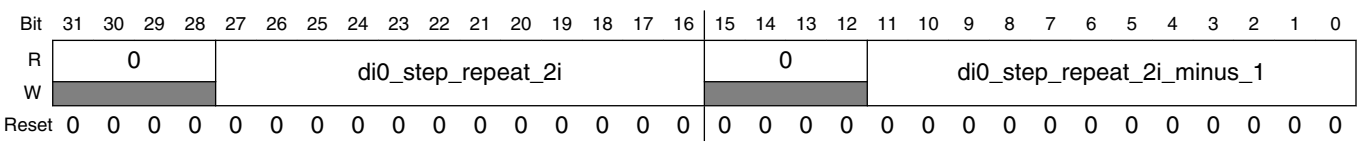


IPUx_DI0_DW_SET3_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di0_data_cnt_down3_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di0_data_cnt_up3_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di0_pt_*_<i>

37.5.220 DI0 Step Repeat <i> Registers (IPUx_DI0_STP_REP_i)

Address: Base address + 4_0148h offset



IPUx_DI0_STP_REP_i field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 di0_step_repeat_2i	Step Repeat <i> This fields defines the amount of repetitions that will be performed by the counter <i>
15–12 Reserved	This read-only field is reserved and always has the value 0.
di0_step_repeat_2i_minus_1	Step Repeat <i> This fields defines the amount of repetitions that will be performed by the counter <i>

37.5.221 DI0 Step Repeat 9 Registers (IPUx_DI0_STP_REP_9)

Address: Base address + 4_0158h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																				di0_step_repeat_9											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_STP_REP_9 field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
di0_step_repeat_9	Step Repeat 9 This fields defines the amount of repetitions that will be performed by the counter 9

37.5.222 DI0 Serial Display Control Register (IPUx_DI0_SER_CONF)

Address: Base address + 4_015Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	DI0_SERIAL_LLA_PNTR_RS_R_1				DI0_SERIAL_LLA_PNTR_RS_R_0				DI0_SERIAL_LLA_PNTR_RS_W_1				DI0_SERIAL_LLA_PNTR_RS_W_0			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DI0_SERIAL_LATCH								0		DI0_LLA_SER_ACCESS	DI0_SER_CLK_POLARITY	DI0_SERIAL_DATA_POLARITY	DI0_SERIAL_RS_POLARITY	DI0_SERIAL_CS_POLARITY	DI0_WAIT4SERIAL
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_SER_CONF field descriptions

Field	Description
31–28 DI0_SERIAL_LLA_PNTR_RS_R_1	RS 3 waveform pointer for read low level access This pointer defines which waveform set will be chosen when the read low level access is targeted to RS group 1. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
27–24 DI0_SERIAL_LLA_PNTR_RS_R_0	RS 2 waveform pointer for read low level access This pointer defines which waveform set will be chosen when the read low level access is targeted to RS group 0. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
23–20 DI0_SERIAL_LLA_PNTR_RS_W_1	RS 1 waveform pointer for write low level access This pointer defines which waveform set will be chosen when the low level write access is targeted to RS group 1. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
19–16 DI0_SERIAL_LLA_PNTR_RS_W_0	RS 0 waveform pointer for write low level access This pointer defines which waveform set will be chosen when the low level write access is targeted to RS group 0. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
15–8 DI0_SERIAL_LATCH	DI0 Serial Latch This field defines how many cycles to insert between serial read accesses start to data sampling in the
7–6 Reserved	This read-only field is reserved and always has the value 0.

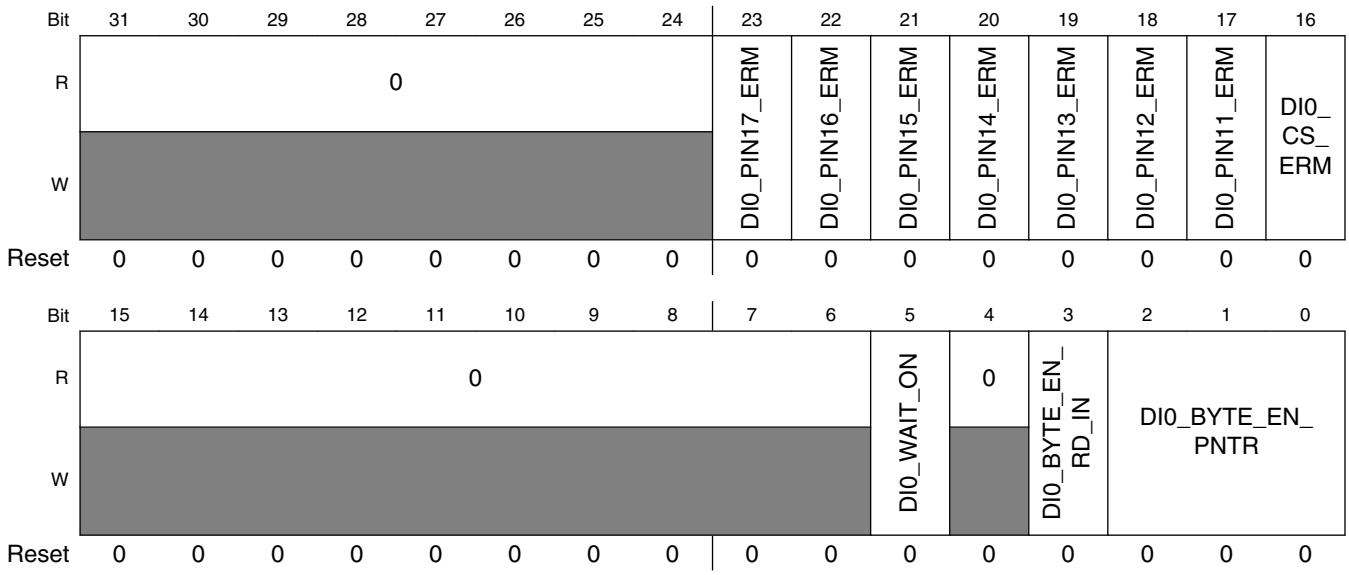
Table continues on the next page...

IPUx_DI0_SER_CONF field descriptions (continued)

Field	Description
5 DI0_LLA_SER_ACCESS	Direct Low Level Access to Serial display 1 ARM platform access is performed via a direct path to the serial display in LLA mode, in this mode only the ARM platform in LLA mode can access the serial port 0 ARM platform access to the serial display port is not done directly, hence other source are allowed to access the serial port. The arbitration is done automatically
4 DI0_SER_CLK_POLARITY	Serial Clock Polarity The output polarity of the SER_CLK pin 1 The clock is inverted 0 The clock is not inverted
3 DI0_SERIAL_DATA_POLARITY	Serial Data Polarity The output polarity of the SER_DATA pin 1 The data is inverted 0 The data is not inverted
2 DI0_SERIAL_RS_POLARITY	Serial RS Polarity The output polarity of the SER_RS pin 1 The RS is inverted 0 The RS is not inverted
1 DI0_SERIAL_CS_POLARITY	Serial Chip Select Polarity The output polarity of the SER_CS pin 1 The CS is inverted 0 The CS is not inverted
0 DI0_WAIT4SERIAL	Wait for Serial When the parallel display share pins with the serial port. Accessing the parallel port is not allowed till the serial port completes its access. 1 The parallel port should wait to the serial port as the pins are shared 0 The parallel port should not wait to the serial port as the pins are not shared

37.5.223 DI0 Special Signals Control Register (IPUx_DI0_SSC)

Address: Base address + 4_0160h offset



IPUx_DI0_SSC field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23 DI0_PIN17_ERM	DI0 PIN17 error recovery mode. This bit defines the error recovery mode of the PIN17 pin. 1 The PIN17 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN17 pin following a display error detection
22 DI0_PIN16_ERM	DI0 PIN16 error recovery mode. This bit defines the error recovery mode of the PIN16 pin. 1 The PIN16 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN16 pin following a display error detection
21 DI0_PIN15_ERM	DI0 PIN15 error recovery mode. This bit defines the error recovery mode of the PIN15 pin. 1 The PIN15 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN15 pin following a display error detection
20 DI0_PIN14_ERM	DI0 PIN14 error recovery mode. This bit defines the error recovery mode of the PIN14 pin.

Table continues on the next page...

IPUx_DI0_SSC field descriptions (continued)

Field	Description
	1 The PIN14 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN14 pin following a display error detection
19 DI0_PIN13_ERM	DI0 PIN13 error recovery mode. This bit defines the error recovery mode of the PIN13 pin. 1 The PIN13 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN13 pin following a display error detection
18 DI0_PIN12_ERM	DI0 PIN12 error recovery mode. This bit defines the error recovery mode of the PIN12 pin. 1 The PIN12 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN12 pin following a display error detection
17 DI0_PIN11_ERM	DI0 PIN11 error recovery mode. This bit defines the error recovery mode of the PIN11 pin. 1 The PIN11 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC 0 Nothing is done to the PIN11 pin following a display error detection
16 DI0_CS_ERM	DI0 GLUELOGIC error recovery mode. This bit defines the error recovery mode of the GLUELOGIC. 1 The GLUELOGIC is release in case of a synchronous display error. The release will be done on the next VSYNC 0 Nothing is done to the GLUELOGIC following a display error detection
15–6 Reserved	This read-only field is reserved and always has the value 0.
5 DI0_WAIT_ON	Wait On This field defines the DC's response to WAIT signal 1 The DC holds the flow as long as WAIT is asserted 0 The DC continues the flow regardless the WAIT signal
4 Reserved	This read-only field is reserved and always has the value 0.
3 DI0_BYTE_EN_RD_IN	Byte Enable Read In This bit selects the source of the byte enable pins 1 The write byte enable signals are routed via bits [17:16] of the display's data, The read byte enable signals are routed via bits [19:18 of the display's data 0 The byte enable signals are routed via bits [17:16] of the display's data for both read and write
DI0_BYTE_EN_PNTR	Byte Enable Pointer This pointer selects the pin asserted along with the byte enables signals 000 wave form of byte enable as pin_11

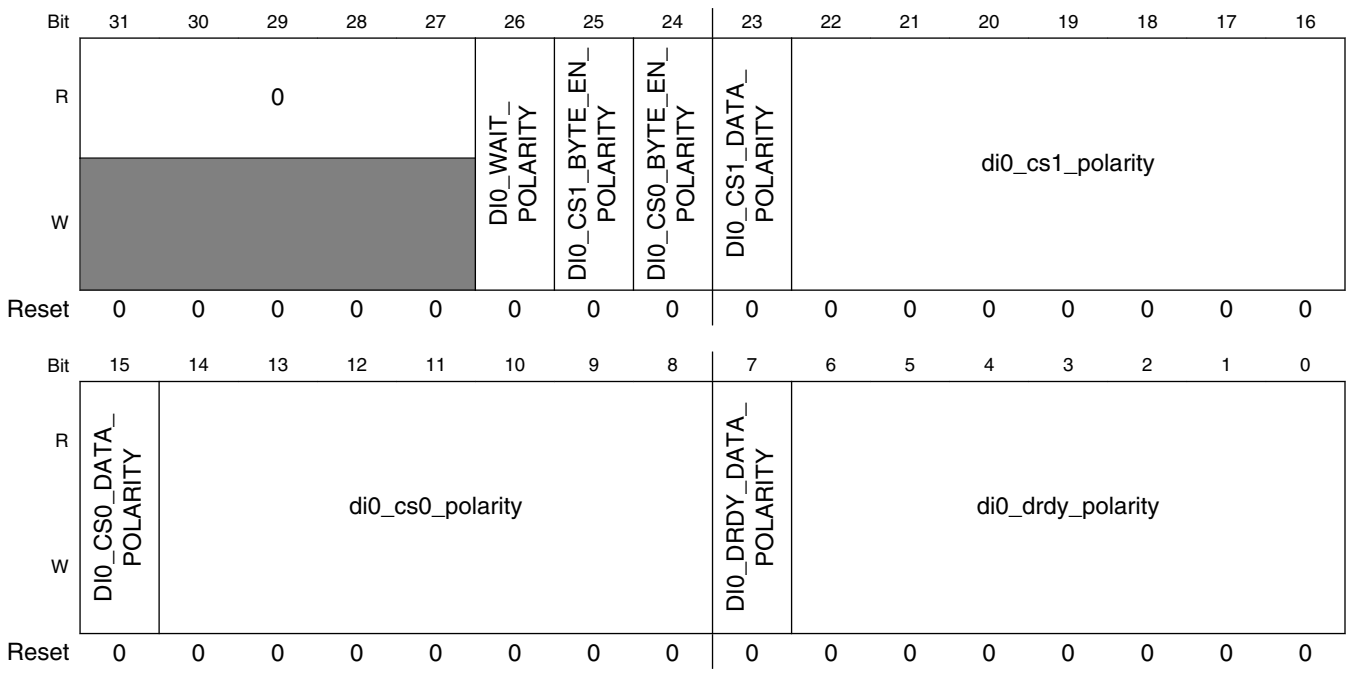
Table continues on the next page...

IPUx_DI0_SSC field descriptions (continued)

Field	Description
	001 wave form of byte enable as pin_12 111 wave form of byte enable as suitable CS pin

37.5.224 DI0 Polarity Register (IPUx_DI0_POL)

Address: Base address + 4_0164h offset



IPUx_DI0_POL field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26 DI0_WAIT_POLARITY	WAIT polarity This bit defines the polarity of the wait signal input coming from the displa1 1 active high 0 active low
25 DI0_CS1_BYTE_EN_POLARITY	Byte Enable associated with CS1 polarity This bit defines the polarity of the byte enable signals to the display 1 active high 0 active low
24 DI0_CS0_BYTE_EN_POLARITY	Byte Enable associated with CS0 polarity This bit defines the polarity of the byte enable signals to the display

Table continues on the next page...

IPUx_DI0_POL field descriptions (continued)

Field	Description
	1 active high 0 active low
23 DI0_CS1_DATA_POLARITY	Data Polarity associated with CS1
22–16 di0_cs1_polarity	DI0 output pin's polarity for CS1 This bits define the polarity of each of the DI's outputs when CS1 is asserted 1 The output pin is active high 0 The output pin is active low
15 DI0_CS0_DATA_POLARITY	Data Polarity associated with CS0
14–8 di0_cs0_polarity	DI0 output pin's polarity for CS1 This bits define the polarity of each of the DI's outputs when CS1 is asserted 1 The output pin is active high 0 The output pin is active low
7 DI0_DRDY_DATA_POLARITY	Data Polarity associated with DRDY
di0_drdy_polarity	DI0 output dynamic pin's polarity for synchronous access This bits define the polarity of each of the DI's outputs when synchronous display access is asserted The pins' default polarity is the same as defined in the di0_drdy_polarity bits 1 The output pin is active high 0 The output pin is active low

37.5.225 DI0 Active Window 0 Register (IPUx_DI0_AW0)

Address: Base address + 4_0168h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI0_AW_TRIG_SEL				DI0_AW_HEND												DI0_AW_HCOUNT_SEL				DI0_AW_HSTART											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_AW0 field descriptions

Field	Description
31–28 DI0_AW_TRIG_SEL	This field selects the trigger for sending data during the display's active window 000 disabled 001 The trigger is the same trigger that triggers the displays clock. 010 The trigger is counter #1

Table continues on the next page...

IPUx_DI0_AW0 field descriptions (continued)

Field	Description
	011 The trigger is counter #2 100 The trigger is counter #3 101 The trigger is counter #4 110 The trigger is counter #5 111 The trigger is always on.
27–16 DI0_AW_HEND	This field defines the horizontal end of the active window
15–12 DI0_AW_ HCOUNT_SEL	GM: This field selects the counter that counts the horizontal position of the display's active window 0000 disabled 0001 reserved 0010 The counter is counter #1 0011 The counter is counter #2 0100 The counter is counter #3 0101 The counter is counter #4 0110 The counter is counter #5 1001 The counter is counter #8
DI0_AW_ HSTART	This field defines the horizontal start of the active window DI0_AW_HSTART < DI0_AW_HEND

37.5.226 DI0 Active Window 1 Register (IPUx_DI0_AW1)

Address: Base address + 4_016Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				DI0_AW_VEND												DI0_AW_ VCOUNT_ SEL				DI0_AW_VSTART											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI0_AW1 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 DI0_AW_VEND	This field defines the vertical end of the active window
15–12 DI0_AW_ VCOUNT_SEL	This field selects the counter that counts the vertical position of the display's active window 0000 disabled 0001 reserved 0010 The counter is counter #1 0011 The counter is counter #2 0100 The counter is counter #3 0101 The counter is counter #4

Table continues on the next page...

IPUx_DI0_AW1 field descriptions (continued)

Field	Description
	0110 The counter is counter #5 1001 The counter is counter #8
DI0_AW_ VSTART	This field defines the vertical start of the active window DI0_AW_VSTART < DI0_AW_VEND

37.5.227 DI0 Screen Configuration Register (IPUx_DI0_SCR_CONF)

Address: Base address + 4_0170h offset

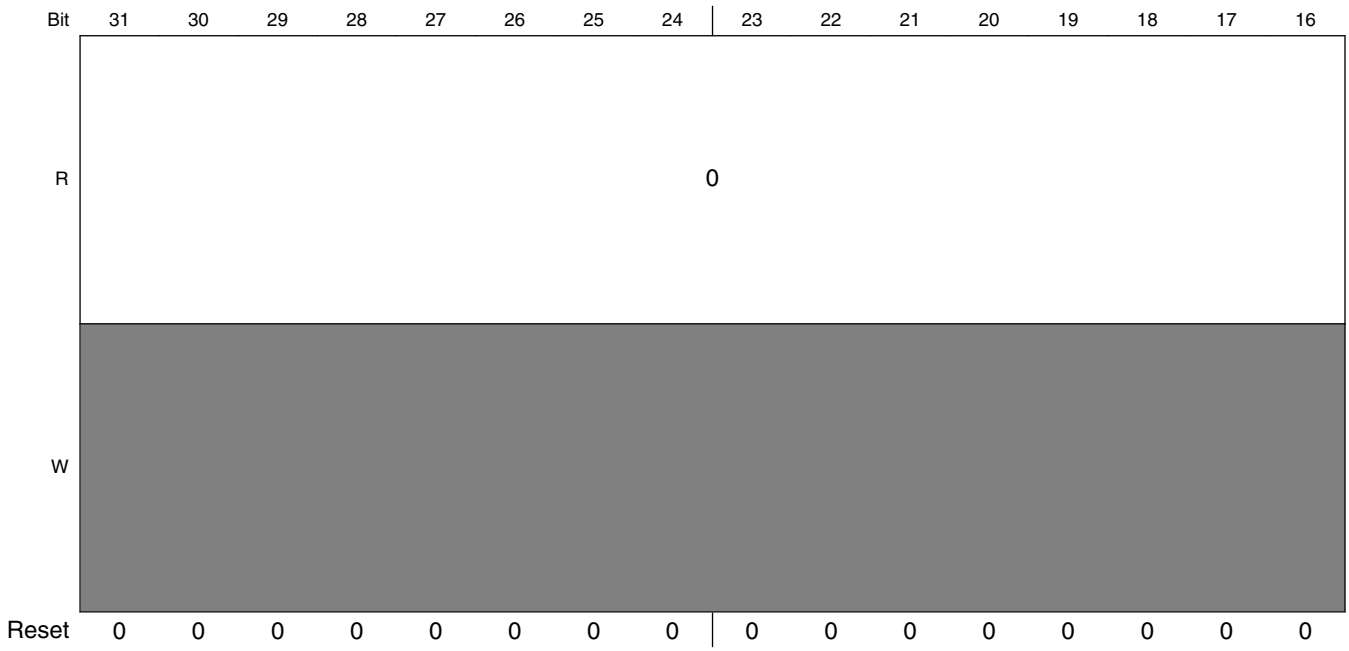
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																DI0_SCREEN_HEIGHT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

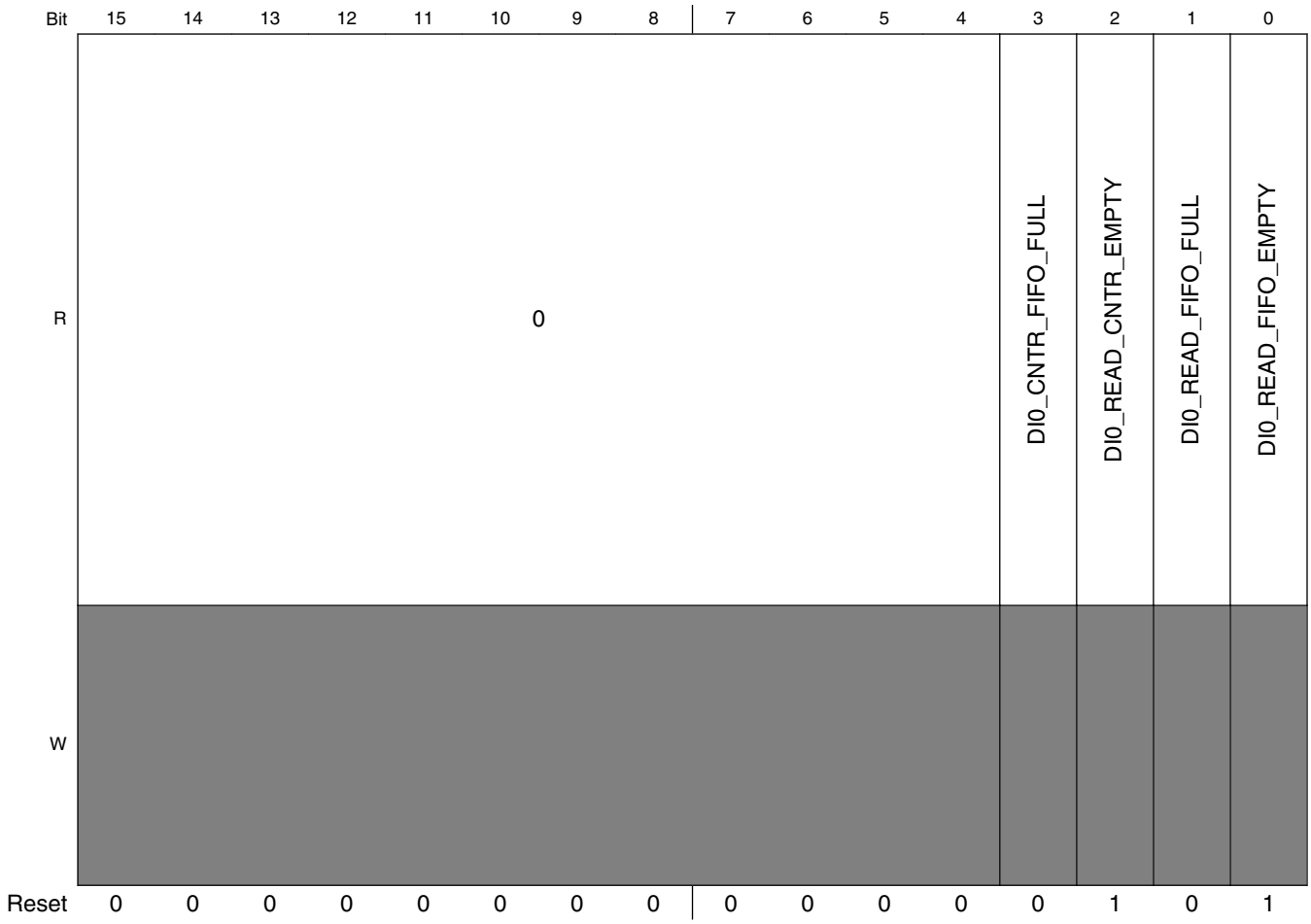
IPUx_DI0_SCR_CONF field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
DI0_SCREEN_ HEIGHT	This field defines the number of display rows (Number_of_ROWS = DI0_SCREEN_HEIGHT+1) This field is used for VSYNC calculation and for anti-tearing

37.5.228 DI0 Status Register (IPUx_DI0_STAT)

Address: Base address + 4_0174h offset





IPUx_DI0_STAT field descriptions

Field	Description
31–4 Reserved	This read-only field is reserved and always has the value 0.
3 DI0_CNTR_ FIFO_FULL	This bit indicates a full state of the DI0 FIFO. This FIFO is part of the DI0 synchronizer.
2 DI0_READ_ CNTR_EMPTY	This bit indicates an empty state of the DI0 FIFO. This FIFO is part of the DI0 synchronizer.
1 DI0_READ_ FIFO_FULL	This bit indicates a full state of the DI0 FIFO when performing a read. This FIFO is part of the DI0 synchronizer.
0 DI0_READ_ FIFO_EMPTY	This bit indicates an empty state of the DI0 FIFO when performing a read. This FIFO is part of the DI0 synchronizer.

37.5.229 DI1General Register (IPUx_DI1_GENERAL)

Address: Base address + 4_8000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	di1_pin8_pin15_sel	di1_disp_y_sel			DI1_CLOCK_STOP_MODE				DI1_DISP_CLOCK_INIT	di1_mask_sel	di1_vsync_ext	di1_clk_ext	DI1_WATCHDOG_MODE		di1_polarity_disp_clk	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	di1_sync_count_sel				di1_err_treatment	di1_erm_vsync_sel	di1_polarity_cs1	di1_polarity_cs0	di1_polarity_i_1							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_GENERAL field descriptions

Field	Description
31 di1_pin8_pin15_sel	This bit routes PIN8 over PIN15 1 PIN8 is routed to PIN15, PIN8 is also routed to PIN8 0 PIN15 is routed to PIN15, PIN8 is routed to PIN8
30–28 di1_disp_y_sel	DI1 Display Vertical coordinate (Y) select. This field defines which one of the 8 counters will be used as a display's line counter. 000 counter #1 is selected 111 counter #8 is selected
27–24 DI1_CLOCK_STOP_MODE	DI clock stop mode When performing a clock change. The DI stops the clock to the display. These field defines when the clock will be stopped Stopping at EOL/EOF is supported for the case where the data is coming from the IDMAC (DMA access). In case that only direct accesses is performed, the user should set this field to 0000 0001-1001 stop at the next event of one of the counters (counter #1 to counter #9) 0000 stop at the next edge of the display clock 1100 stop at EOL (end of a line), but if stop request is during blanking interval, stop now 1101 stop at EOF (end of a frame), but if stop request is during blanking interval, stop now 1110 stop at EOL (end of a line), but if stop request is during blanking interval, stop at the end of the next line 1111 stop at EOF (end of a frame), but if stop request is during blanking interval, stop at the end of the next frame

Table continues on the next page...

IPUx_DI1_GENERAL field descriptions (continued)

Field	Description
23 DI1_DISP_CLOCK_INIT	Display clock's initial mode For synchronization error conditions the display clock can be stopped on the next VSYNC 1 The display's clock is running after the next VSYNC (indicating new frame) 0 The display's clock is stopped after the next VSYNC (indicating new frame)
22 di1_mask_sel	DI1 Mask select. IPP_PIN_2 output of the DI that functions as MASK signal can come from 2 sources: counter #2 or extracted from the MASK data coming from the memory. 1 IPP_PIN_2 is coming from extracted MASK data coming from the memory 0 IPP_PIN_2 is coming from counter #2
21 di1_vsync_ext	DI1 External VSYNC. This bit selects the source of the VSYNC signal 1 External to the IPU 0 Internally generated by the IPU
20 di1_clk_ext	DI1 External Clock. This bit selects the source of the DI's clock 1 The source of the clock is external to the IPU 0 The clock is internally generated by the IPU
19–18 DI1_WATCHDOG_MODE	DI1 watchdog mode In case of a display error where the DI clock is stopped (defined at di0_err_treatment). An internal watchdog counts DI clocks. If this timer reached its pre defined value the DI will skip the current frame and restart on the frame. This 2 bits define the number of DI clock cycles that the timer counts. 00 The timer counts 4 DI cycles 01 The timer counts 16 DI cycles 10 The timer counts 64 DI cycles 11 The timer counts 128 DI cycles
17 di1_polarity_disp_clk	DI1 Output Clock's polarity This bits define the polarity of the DI's clock. 1 The output clock is active high 0 The output clock is active low
16 Reserved	This read-only field is reserved and always has the value 0.
15–12 di1_sync_count_sel	For synchronous flow error: selects synchronous flow synchronization counter in DI:
11 di1_err_treatment	In case of synchronous flow error there are 2 ways to handle the display 1 to wait (i.e. stop clock) 0 Drive the last component
10 di1_erm_vsync_sel	DI1 error recovery module's VSYNC source select The error recovery block detect a case where the DI's VSYNC is asserted before the EOF. This bit selects the source of the VSYNC signal monitored by this mechanism.

Table continues on the next page...

IPUx_DI1_GENERAL field descriptions (continued)

Field	Description
	1 vsync_post - an internal VSYNC signal asserted 2 lines after the DI's VSYNC 0 vsync_pre - an internal VSYNC signal asserted 2 lines before the DI's VSYNC
9 di1_polarity_cs1	DI1 Chip Select's 1 polarity This bits define the polarity of the DI's CS1. 1 The CS1 is active high 0 The CS1 is active low
8 di1_polarity_cs0	DI1 Chip Select's 0 polarity This bits define the polarity of the DI's CS0. 1 The CS0 is active high 0 The CS0 is active low
di1_polarity_i_1	DI1 output pin's polarity This bits define the polarity of each of the DI's outputs. 1 The output pin is active high 0 The output pin is active low

37.5.230 DI1 Base Sync Clock Gen 0 Register (IPUx_DI1_BS_CLKGEN0)

Address: Base address + 4_8004h offset

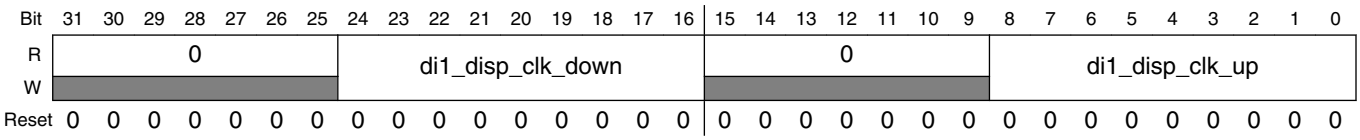
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di1_disp_clk_offset									0				di1_disp_clk_period											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_BS_CLKGEN0 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_disp_clk_ offset	DI1 Display Clock Offset The DI has the ability to delay the display's clock This field defines the amount of IPU's clock cycles added as delay on this clock.
15–12 Reserved	This read-only field is reserved and always has the value 0.
di1_disp_clk_ period	DI1 Display Clock Period This field defines the Display interface clock period for display write access. This parameter contains an integer part (bits 11:4) and a fractional part (bits 3:0). It defines a fractional division ratio of the HSP_CLK clock for generation of the display's interface clock.

37.5.231 DI1 Base Sync Clock Gen 1 Register (IPUx_DI1_BS_CLKGEN1)

Address: Base address + 4_8008h offset

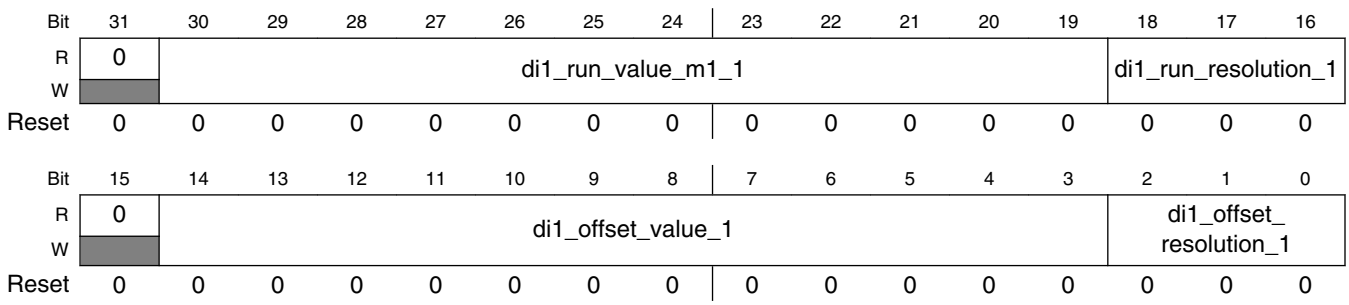


IPUx_DI1_BS_CLKGEN1 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_disp_clk_down	DI1 display clock falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is a time interval between display's access start point and display 's interface clock falling edge.
15–9 Reserved	This read-only field is reserved and always has the value 0.
di1_disp_clk_up	DI1 display clock rising edge position This parameter contains an integer part (bits 8:1) and a fractional part (bit 0). The position value is a time interval between display's access start point and display 's interface clock rising edge.

37.5.232 DI1 Sync Wave Gen 1 Register 0 (IPUx_DI1_SW_GEN0_1)

Address: Base address + 4_800Ch offset



IPUx_DI1_SW_GEN0_1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.

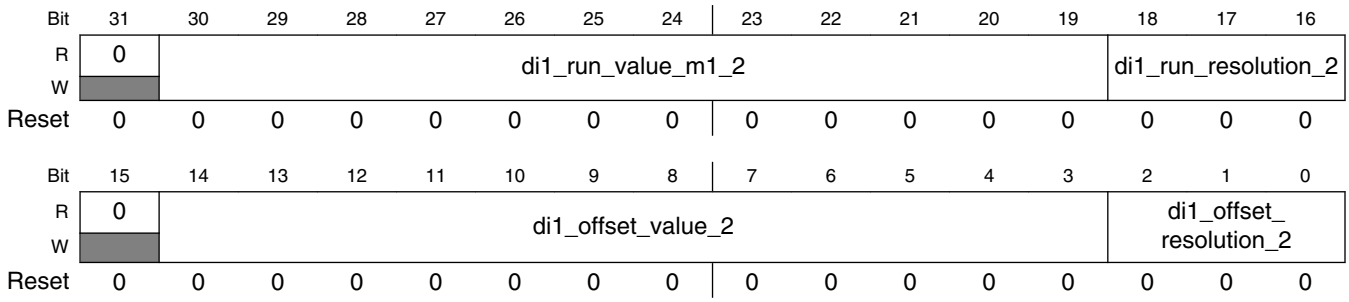
Table continues on the next page...

IPUx_DI1_SW_GEN0_1 field descriptions (continued)

Field	Description
30–19 di1_run_value_m1_1	DI1 counter #1 pre defined value This fields defines the counter #1 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_1 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_resolution_1	DI1 counter #1 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 NA 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_1	DI1 counter #1 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_resolution_1	DI1 counter #1 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 NA 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.233 DI1 Sync Wave Gen 2 Register 0 (IPUx_DI1_SW_GEN0_2)

Address: Base address + 4_8010h offset



IPUx_DI1_SW_GEN0_2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_m1_2	DI1 counter #2 pre defined value This fields defines the counter #2 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_2 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_resolution_2	DI1 counter #2 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_2	DI1 counter #2 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_resolution_2	DI1 counter #2 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

Table continues on the next page...

IPUx_DI1_SW_GEN0_2 field descriptions (continued)

Field	Description
	011 NA 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.234 DI1 Sync Wave Gen 3 Register 0 (IPUx_DI1_SW_GEN0_3)

Address: Base address + 4_8014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_3												di1_run_resolution_3		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_3												di1_offset_ resolution_3		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_ m1_3	DI1 counter #3 pre defined value This fields defines the counter #3 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_3 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_ resolution_3	DI1 counter #3 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

Table continues on the next page...

IPUx_DI1_SW_GEN0_3 field descriptions (continued)

Field	Description
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_3	DI1 counter #3 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_resolution_3	DI1 counter #3 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 NA 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.235 DI1 Sync Wave Gen 4 Register 0 (IPUx_DI1_SW_GEN0_4)

Address: Base address + 4_8018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_4												di1_run_resolution_4		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_4												di1_offset_resolution_4		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_4 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_m1_4	DI1 counter #4 pre defined value This fields defines the counter #4 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_4 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.

Table continues on the next page...

IPUx_DI1_SW_GEN0_4 field descriptions (continued)

Field	Description
18–16 di1_run_ resolution_4	DI1 counter #4 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_ 4	DI1 counter #4 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_ resolution_4	DI1 counter #4 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.

37.5.236 DI1 Sync Wave Gen 5 Register 0 (IPUx_DI1_SW_GEN0_5)

Address: Base address + 4_801Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_5												di1_run_resolution_5		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_5												di1_offset_ resolution_5		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_5 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_m1_5	DI1 counter #5 pre defined value This fields defines the counter #5 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_5 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_resolution_5	DI1 counter #5 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_5	DI1 counter #5 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_resolution_5	DI1 counter #5 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 -The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.

37.5.237 DI1 Sync Wave Gen 6 Register 0 (IPUx_DI1_SW_GEN0_6)

Address: Base address + 4_8020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_6												di1_run_resolution_6		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_6												di1_offset_ resolution_6		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_6 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_ m1_6	DI1 counter #6 pre defined value This fields defines the counter #6 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_6 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_ resolution_6	DI1 counter #6 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_ 6	DI1 counter #6 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_ resolution_6	DI1 counter #6 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3

Table continues on the next page...

IPUx_DI1_SW_GEN0_6 field descriptions (continued)

Field	Description
	101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.238 DI1 Sync Wave Gen 7 Register 0 (IPUx_DI1_SW_GEN0_7)

Address: Base address + 4_8024h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_7												di1_run_resolution_7		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_7												di1_offset_ resolution_1		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_7 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_ m1_7	DI1 counter #7 pre defined value This fields defines the counter #7 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_7 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_ resolution_7	DI1 counter #1 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_ 7	DI1 counter #7 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by

Table continues on the next page...

IPUx_DI1_SW_GEN0_7 field descriptions (continued)

Field	Description
di1_offset_ resolution_1	DI1 counter #7 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.239 DI1 Sync Wave Gen 8 Register 0 (IPUx_DI1_SW_GEN0_8)

Address: Base address + 4_8028h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_8												di1_run_resolution_8		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_8												di1_offset_ resolution_8		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_8 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_ m1_8	DI1 counter #8 pre defined value This fields defines the counter #8 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_8 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.
18–16 di1_run_ resolution_8	DI1 counter #8 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4

Table continues on the next page...

IPUx_DI1_SW_GEN0_8 field descriptions (continued)

Field	Description
	110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_8	DI1 counter #8 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_resolution_8	DI1 counter #8 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.240 DI1Sync Wave Gen 9 Register 0 (IPUx_DI1_SW_GEN0_9)

Address: Base address + 4_802Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	di1_run_value_m1_9												di1_run_resolution_9		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	di1_offset_value_9												di1_offset_resolution_9		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SW_GEN0_9 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–19 di1_run_value_m1_9	DI1 counter #9 pre defined value This fields defines the counter #9 pre defines value. When the counter is auto reload mode (di1_cnt_auto_reload_9 bit is set), the counter automatically restarts. otherwise the counter restarts for the amount of time defined on the corresponding di1_step_repeat field.

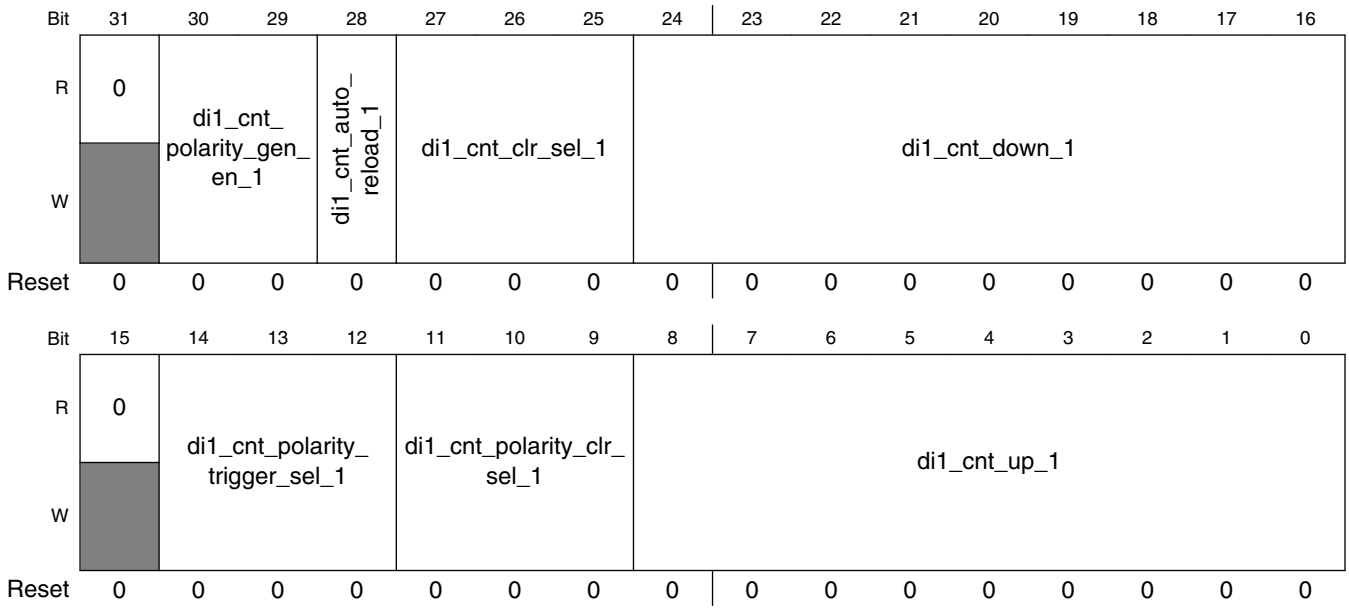
Table continues on the next page...

IPUx_DI1_SW_GEN0_9 field descriptions (continued)

Field	Description
18–16 di1_run_ resolution_9	DI1 counter #9 Run Resolution This field defines the trigger causing the counter to increment. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
15 Reserved	This read-only field is reserved and always has the value 0.
14–3 di1_offset_value_ 9	DI1 counter #9 offset value The counter can start counting after a pre defined delay This field defines the amount of cycles that the counter will be delayed by
di1_offset_ resolution_9	DI1 counter #9 offset Resolution This field defines the trigger causing the offset counter to increment 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.

37.5.241 DI1 Sync Wave Gen 1 Register 1 (IPUx_DI1_SW_GEN1_1)

Address: Base address + 4_8030h offset



IPUx_DI1_SW_GEN1_1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_1	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_1	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_1 field
27–25 di1_cnt_clr_sel_1	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 Reserved

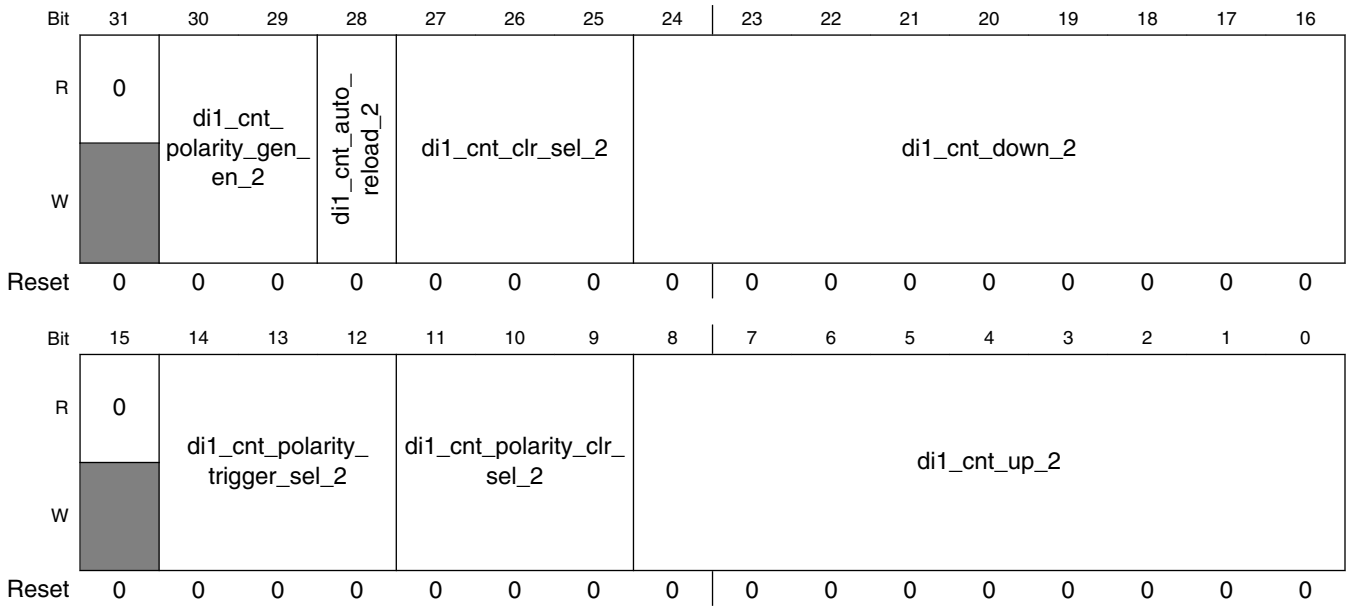
Table continues on the next page...

IPUx_DI1_SW_GEN1_1 field descriptions (continued)

Field	Description
	011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di1_cnt_down_1	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_1	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 Reserved 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_1	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Reserved 011 Reserved 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di1_cnt_up_1	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.242 DI1 Sync Wave Gen 2 Register 1 (IPUx_DI1_SW_GEN1_2)

Address: Base address + 4_8034h offset



IPUx_DI1_SW_GEN1_2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_2	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_2	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_2	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

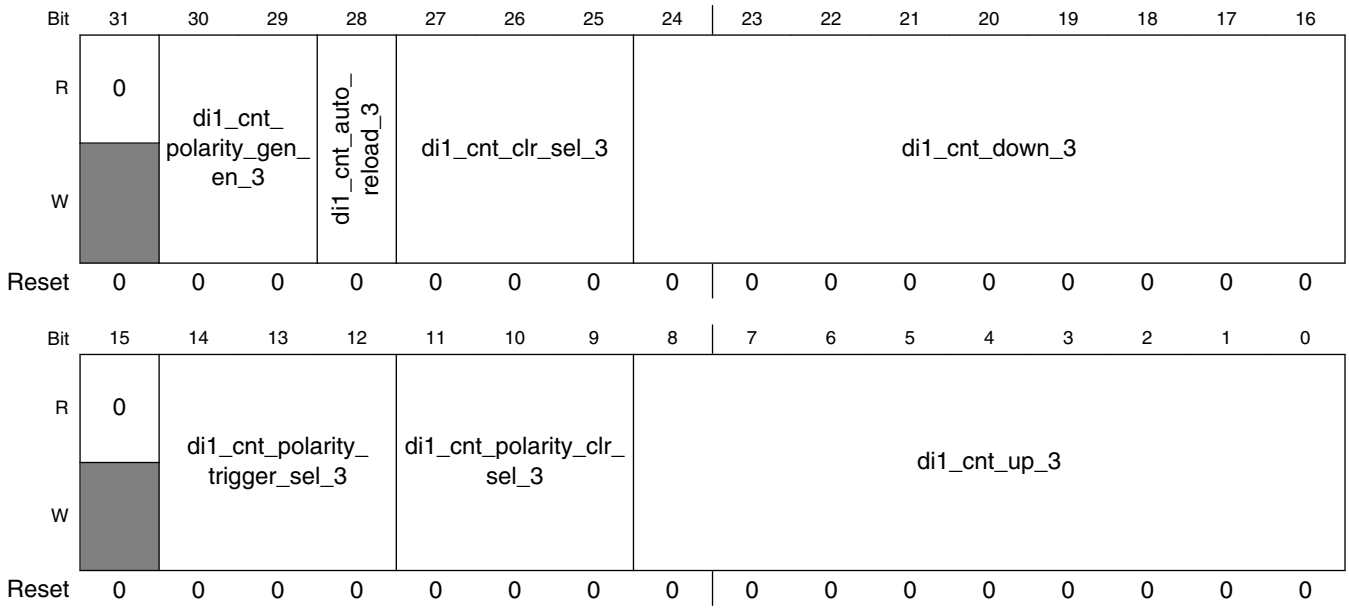
Table continues on the next page...

IPUx_DI1_SW_GEN1_2 field descriptions (continued)

Field	Description
	011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di1_cnt_down_2	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_2	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 Reserved 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_2	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Reserved 011 Reserved 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di1_cnt_up_2	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.243 DI1 Sync Wave Gen 3 Register 1 (IPUx_DI1_SW_GEN1_3)

Address: Base address + 4_8038h offset



IPUx_DI1_SW_GEN1_3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_3	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_3	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_3	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

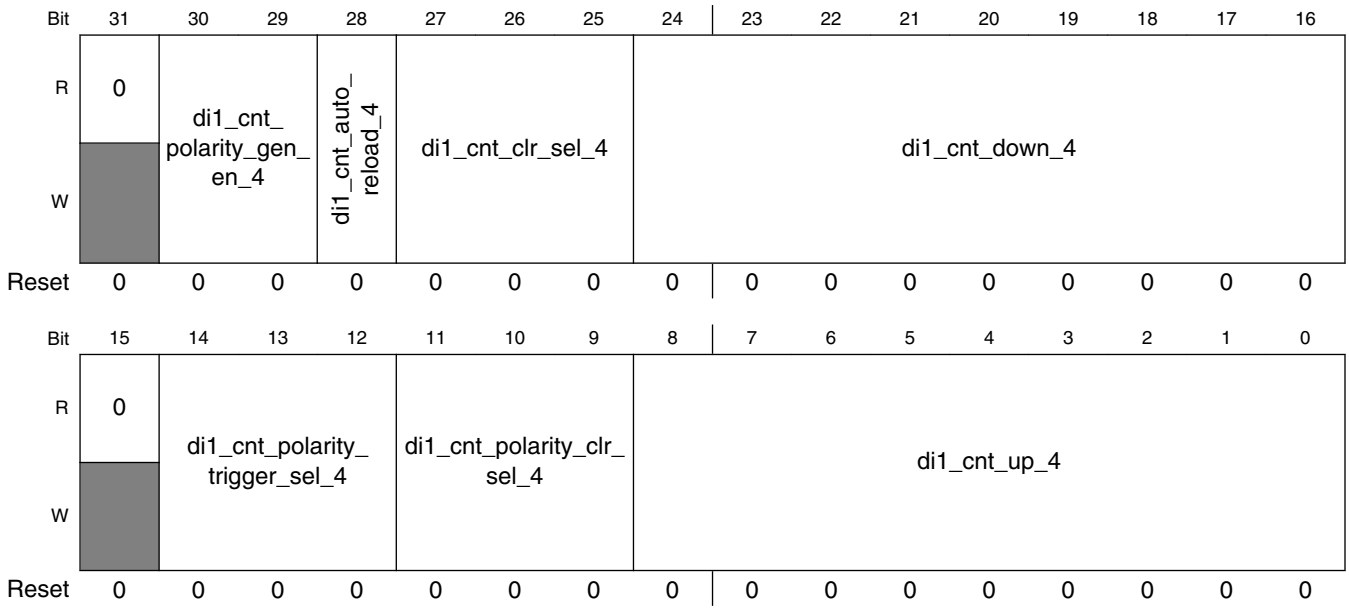
Table continues on the next page...

IPUx_DI1_SW_GEN1_3 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di1_cnt_down_3	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_3	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 Reserved 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_3	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Reserved 101 Reserved 110 Reserved 111 Reserved
di1_cnt_up_3	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.244 DI1 Sync Wave Gen 4 Register 1 (IPUx_DI1_SW_GEN1_4)

Address: Base address + 4_803Ch offset



IPUx_DI1_SW_GEN1_4 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_4	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_4	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_4	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

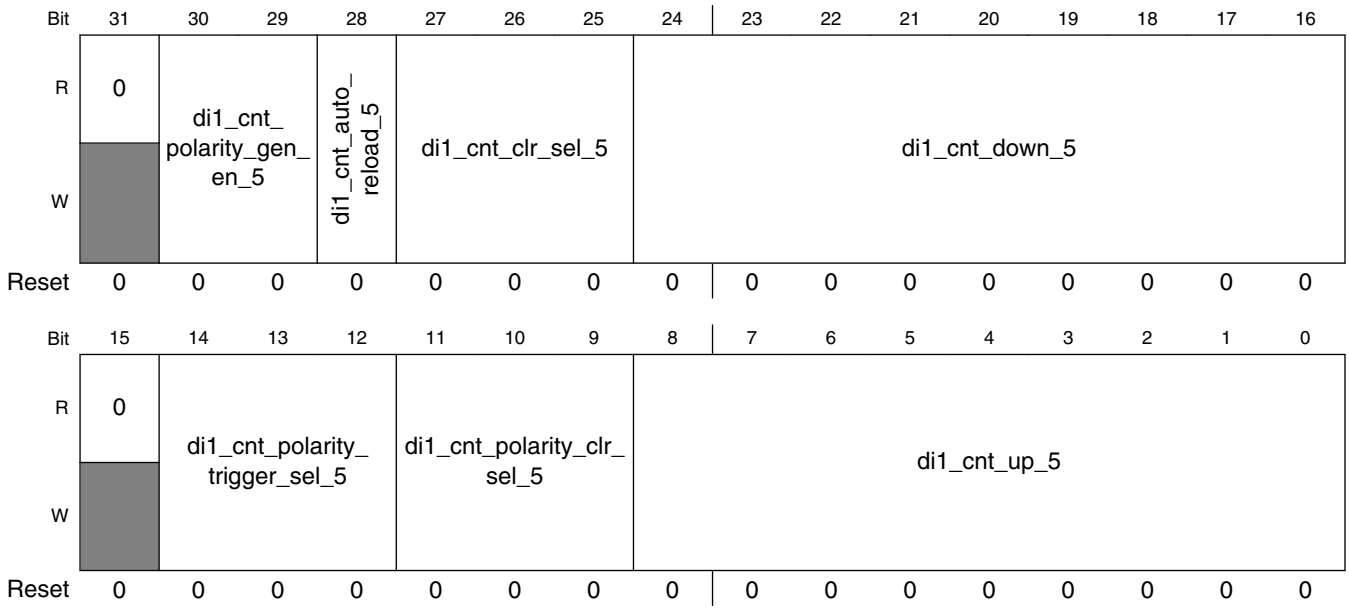
Table continues on the next page...

IPUx_DI1_SW_GEN1_4 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
24–16 di1_cnt_down_4	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_4	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 - Counter is disabled 001 - The counter is triggered by the same trigger that triggers the displays clock. 010 - The Counter is triggered by counter #1 011 - The Counter is triggered by counter #2 100 - The Counter is triggered by counter #3 101 - CSI VSYNC. The VSYNC is a trigger coming from one of the CSI's according to the CSI_VSYNC_DEST bit. — 110 External VSYNC 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_4	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Reserved 110 Reserved 111 Reserved
di1_cnt_up_4	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.245 DI1 Sync Wave Gen 5 Register 1 (IPUx_DI1_SW_GEN1_5)

Address: Base address + 4_8040h offset



IPUx_DI1_SW_GEN1_5 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_5	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_5	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_5	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

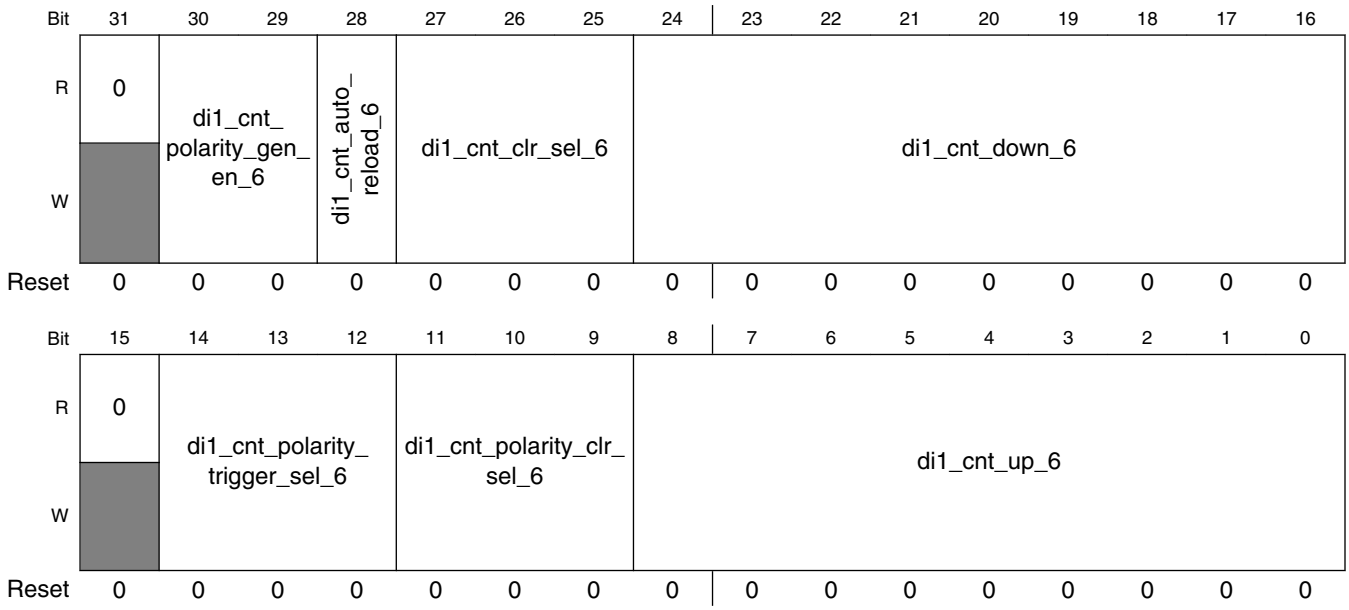
Table continues on the next page...

IPUx_DI1_SW_GEN1_5 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
24–16 di1_cnt_down_5	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_5	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 External VSYNC 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_5	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Reserved 111 Reserved
di1_cnt_up_5	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.246 DI1 Sync Wave Gen 6 Register 1 (IPUx_DI1_SW_GEN1_6)

Address: Base address + 4_8044h offset



IPUx_DI1_SW_GEN1_6 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_6	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_6	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_6	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

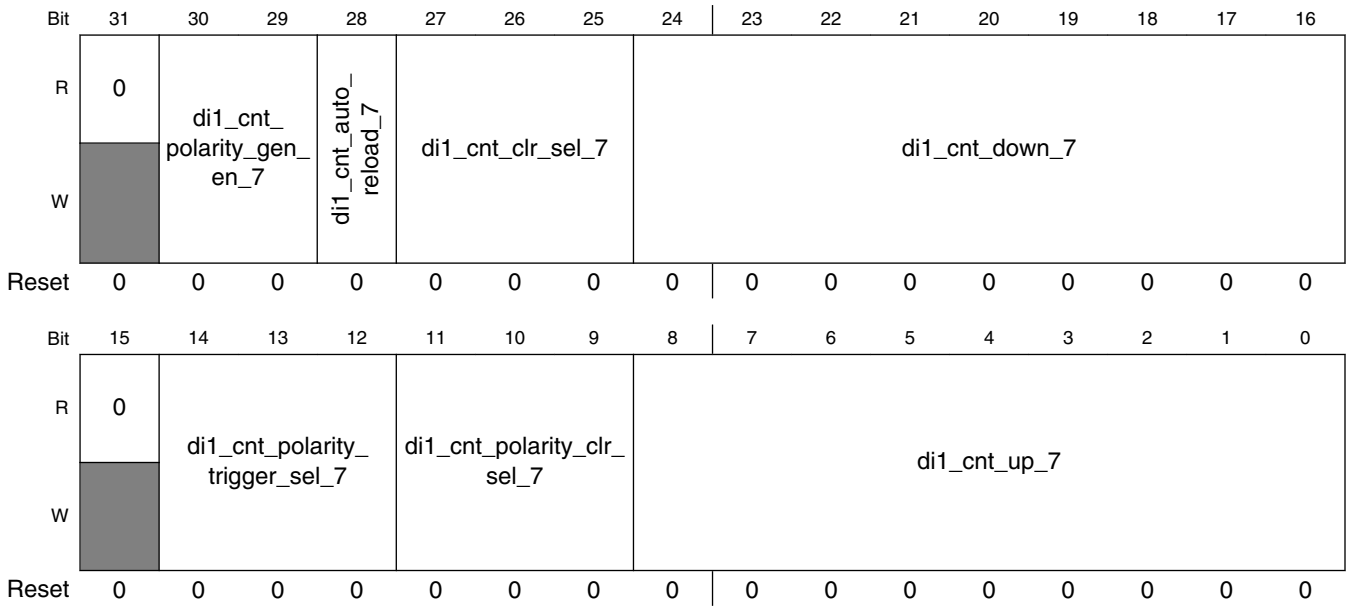
Table continues on the next page...

IPUx_DI1_SW_GEN1_6 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di1_cnt_down_6	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_6	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_6	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Reserved
di1_cnt_up_6	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.247 DI1Sync Wave Gen 7 Register 1 (IPUx_DI1_SW_GEN1_7)

Address: Base address + 4_8048h offset



IPUx_DI1_SW_GEN1_7 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_7	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_7	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_7	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

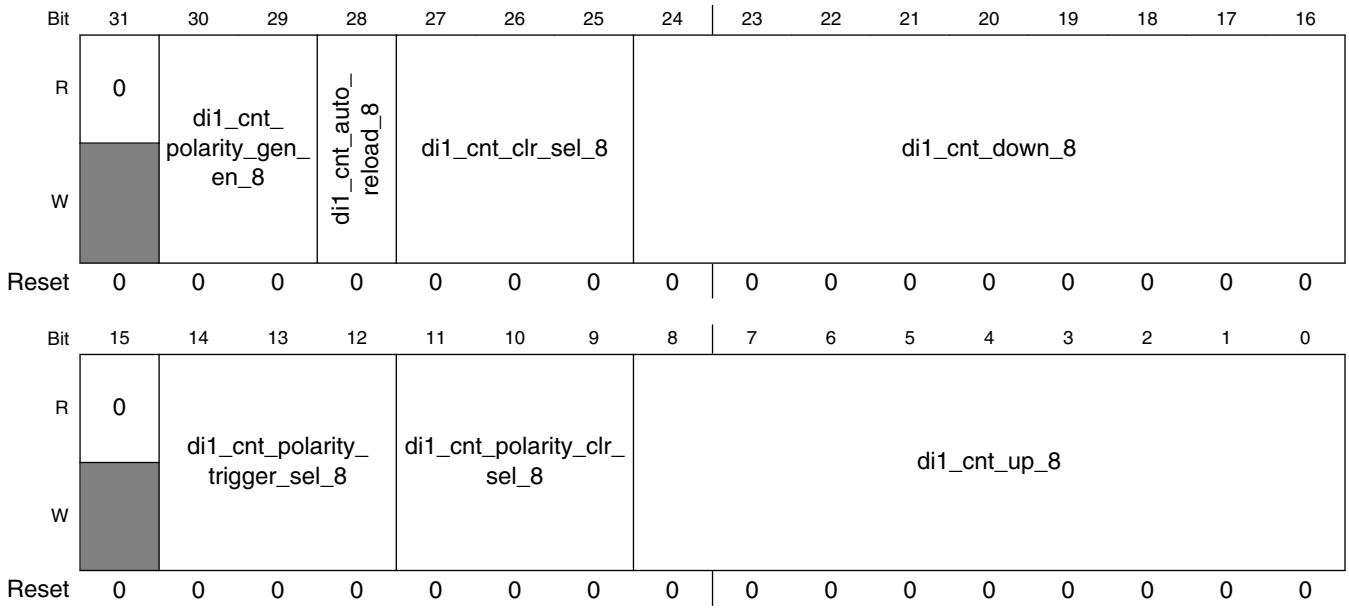
Table continues on the next page...

IPUx_DI1_SW_GEN1_7 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di1_cnt_down_7	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_7	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_7	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Output is inverted if the output of counter #6 is set
di1_cnt_up_7	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.248 DI1 Sync Wave Gen 8 Register 1 (IPUx_DI1_SW_GEN1_8)

Address: Base address + 4_804Ch offset



IPUx_DI1_SW_GEN1_8 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–29 di1_cnt_polarity_gen_en_8	DI1 Counter polarity generator enable The counter's output polarity can be changed on the fly. 00 Dynamic polarity change is disabled 01 The counters UP and DOWN value are calculated according to the trigger selected by cnt_polarity_trigger_sel field. When selecting this mode the UP and DOWN values has to be a multiple of 2. 10 Dynamic polarity change is enabled 11 Outputs polarity is dynamically changed every time the counter reaches its pre defined value
28 di1_cnt_auto_reload_8	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_8	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

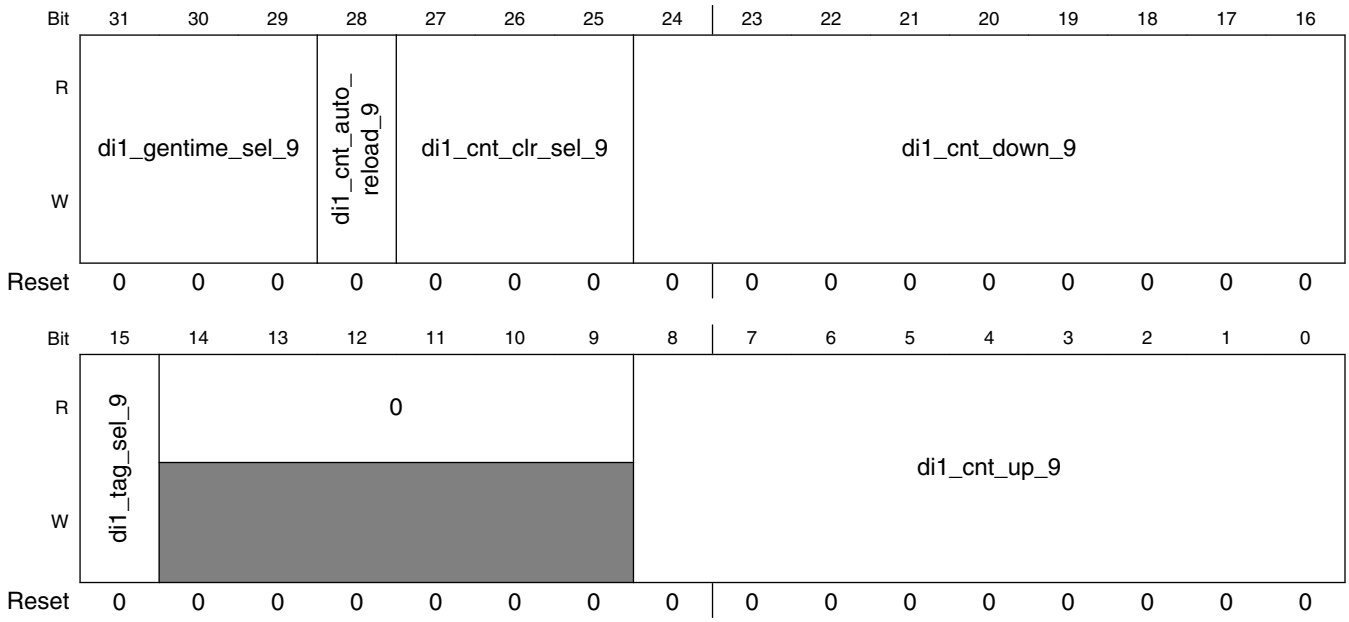
Table continues on the next page...

IPUx_DI1_SW_GEN1_8 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di1_cnt_down_8	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 Reserved	This read-only field is reserved and always has the value 0.
14–12 di1_cnt_polarity_trigger_sel_8	DI1 Counter's toggling trigger select This field selects the counter's trigger causing the output to toggle 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1 011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
11–9 di1_cnt_polarity_clr_sel_8	DI1 counter's polarity Clear select This field selects the input to the counter telling the counter wether to invert the output 000 Output is always inverted 001 Output is kept the same (no inversion) 010 Output is inverted if the output of counter #1 is set 011 Output is inverted if the output of counter #2 is set 100 Output is inverted if the output of counter #3 is set 101 Output is inverted if the output of counter #4 is set 110 Output is inverted if the output of counter #5 is set 111 Output is inverted if the output of counter #6 is set
di1_cnt_up_8	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.249 DI1 Sync Wave Gen 9 Register 1 (IPUx_DI1_SW_GEN1_9)

Address: Base address + 4_8050h offset



IPUx_DI1_SW_GEN1_9 field descriptions

Field	Description
31–29 di1_gentime_sel_9	Counter #9 main waveform select This field defines the counter that counter #9's auxiliary waveform will be attached too. 000 Counter #9's waveform is attached to counter #1's waveform 001 Counter #9's waveform is attached to counter #2's waveform 010 Counter #9's waveform is attached to counter #3's waveform 011 Counter #9's waveform is attached to counter #4's waveform 100 Counter #9's waveform is attached to counter #5's waveform 101 Counter #9's waveform is attached to counter #6's waveform 110 Counter #9's waveform is attached to counter #7's waveform 111 Counter #9's waveform is attached to counter #8's waveform
28 di1_cnt_auto_reload_9	Counter auto reload mode 1 The counter will automatically be reloaded forever, ignoring the value of the di1_step_repeat_<i> field 0 The counter will not be automatically reloaded, It will be reloaded for the amount of repeat times defined on the di1_step_repeat_<i> field
27–25 di1_cnt_clr_sel_9	Counter Clear select This field defines the source of the signals that clears the counter. 000 Counter is disabled 001 The counter is triggered by the same trigger that triggers the displays clock. 010 The Counter is triggered by counter #1

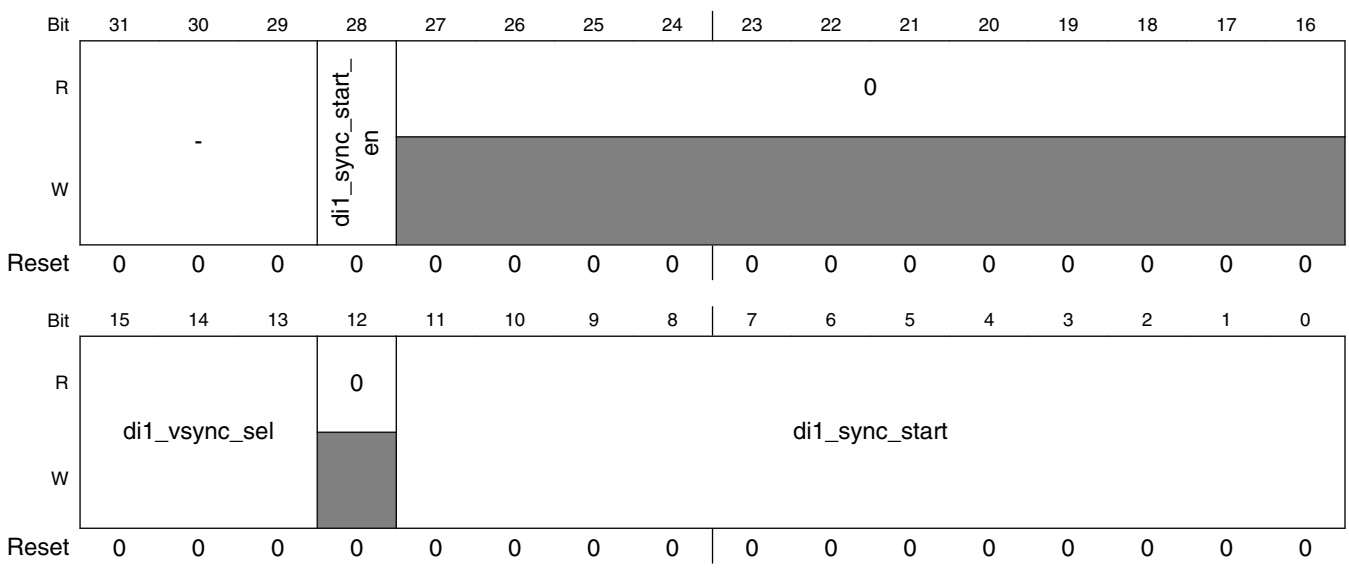
Table continues on the next page...

IPUx_DI1_SW_GEN1_9 field descriptions (continued)

Field	Description
	011 The Counter is triggered by counter #2 100 The Counter is triggered by counter #3 101 The Counter is triggered by counter #4 110 The Counter is triggered by counter #5 111 Counter is always on.
24–16 di1_cnt_down_9	Counter falling edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's falling edge.
15 di1_tag_sel_9	Counter #9 can send a synchronous tag when counter #9 reach its predefined value or when it's triggering counter reaches its pre defined value. 1 tag source is counter #9 0 Tag's source is the triggering counter.
14–9 Reserved	This read-only field is reserved and always has the value 0.
di1_cnt_up_9	Counter rising edge position This parameter contains an integer part (bits 24:17) and a fractional part (bit 16). The position value is the amount of cycles between the trigger's start point and the waveform's rising edge.

37.5.250 DI1 Sync Assistance Gen Register (IPUx_DI1_SYNC_AS_GEN)

Address: Base address + 4_8054h offset



IPUx_DI1_SYNC_AS_GEN field descriptions

Field	Description
31–29 -	Reserve
28 di1_sync_start_en	di1_sync_start_en
27–16 Reserved	This read-only field is reserved and always has the value 0.
15–13 di1_vsync_sel	VSYNC select This field defines which of the counters functions as VSYNC signal 000 VSYNC is coming from counter #1 001 VSYNC is coming from counter #2 111 VSYNC is coming from counter #8
12 Reserved	This read-only field is reserved and always has the value 0.
di1_sync_start	DI1 Sync start This field defines the number of low (including blanking rows) on the which the DI1 starts preparing the data for the next frame.

37.5.251 DI1 Data Wave Gen <i> Register (IPUx_DI1_DW_GEN_i)

The DI1_DW_GEN_<i> register holds pointers for the waveform generators.

These registers have different bit arrangements for parallel and serial display. When using a parallel display VDI Plane Size Register 4 is applicable. When using a serial interface VDI Plane Size Register 4 is applicable.

Table 37-704. Register Field Descriptions for Serial display

Field	Description
31-24 di1_serial_period_<i>	DI1 Serial Period <i> This field defines the period of the time base serial display clock. The units are the internal DI clock
23-16 di1_start_period_<i>	DI1 start period This field defines the amount of cycles between the point where the access is ready to be launched to the actual point where the time base serial display clock restarts. The units are the internal DI clock.
15-14 di1_cst_<i>	DI1 Chip Select pointer for waveform <i> This field points to a register that defines the waveform of the CS pin. For serial displays the down value as defined on DI1_DW_SET*_<i> is measured from the assertion of the last serial display time base clock. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i>

Table continues on the next page...

Table 37-704. Register Field Descriptions for Serial display (continued)

Field	Description
	10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
13-9	Reserved
8-4 di1_serial_valid_bits_<i>	DI1 Serial valid bits. This field defines the amount of valid bits to be transmitted within the 32 bits internal word aligned to bit[0]. The actual amount of valid bits is di1_serial_valid_bits_<i> + 1
3-2 di1_serial_rs_<i>	DI1 Serial RS This field points to a register that defines the waveform of the RS pin. For serial displays the down value as defined on DI1_DW_SET*_<i> is measured from the assertion of the last serial display time base clock. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
1-0 di1_serial_clk_<i>	DI1 serial clock<i> This field points to a register that defines the waveform of the Serial clock pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>

Address: Base address + 4_8058h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	di1_access_size_i								di1_componnent_size_i								di1_cst_i		di1_pt_6_i		di1_pt_5_i		di1_pt_4_i		di1_pt_3_i		di1_pt_2_i		di1_pt_1_i		di1_pt_0_i	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_DW_GEN_i field descriptions

Field	Description
31–24 di1_access_size_i	DI1 Access Size <i> This field defines the amount of IPU cycles between any 2 accesses (an access may be a pixel or generic data that may have more one component)
23–16 di1_componnent_size_i	DI1 component Size This field defines the amount of IPU cycles between any 2 components
15–14 di1_cst_i	DI1 Chip Select pointer for waveform <i> This field points to a register that defines the waveform of the CS pin. The CS is automatically mapped to a specific display

Table continues on the next page...

IPUx_DI1_DW_GEN_i field descriptions (continued)

Field	Description
	00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
13–12 di1_pt_6_i	DI1 PIN_17 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_17 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
11–10 di1_pt_5_i	DI1 PIN_16 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_16 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
9–8 di1_pt_4_i	DI1 PIN_15 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_15 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
7–6 di1_pt_3_i	DI1 PIN_14 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_14 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
5–4 di1_pt_2_i	DI1 PIN_13 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_13 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>
3–2 di1_pt_1_i	DI1 PIN_12 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_12 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 The waveform is defined according to the settings on DI1_DW_SET3_<i>

Table continues on the next page...

IPUx_DI1_DW_GEN_i field descriptions (continued)

Field	Description
di1_pt_0_i	DI1 PIN_11 pointer for waveform <i> This field points to a register that defines the waveform of the PIN_11 pin. 00 The waveform is defined according to the settings on DI1_DW_SET0_<i> 01 The waveform is defined according to the settings on DI1_DW_SET1_<i> 10 The waveform is defined according to the settings on DI1_DW_SET2_<i> 11 - The waveform is defined according to the settings on DI1_DW_SET3_<i>

37.5.252 DI1 Data Wave Set 0 <i> Register (IPUx_DI1_DW_SET0_i)

Address: Base address + 4_8088h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di1_data_cnt_down0_i									0							di1_data_cnt_up0_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_DW_SET0_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_data_cnt_down0_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di1_data_cnt_up0_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>

37.5.253 DI1 Data Wave Set 1 <i> Register (IPUx_DI1_DW_SET1_i)

Address: Base address + 4_80B8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di1_data_cnt_down1_i									0							di1_data_cnt_up1_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_DW_SET1_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_data_cnt_down1_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di1_data_cnt_up1_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>

37.5.254 DI1 Data Wave Set 2 <i> Register (IPUx_DI1_DW_SET2_i)

Address: Base address + 4_80E8h offset

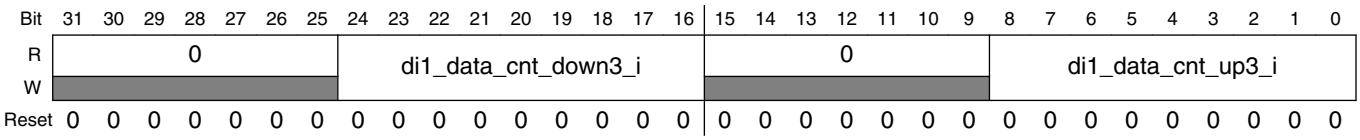
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0							di1_data_cnt_down2_i									0							di1_data_cnt_up2_i								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_DW_SET2_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_data_cnt_down2_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di1_data_cnt_up2_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>

37.5.255 DI1 Data Wave Set 3 <i> Register (IPUx_DI1_DW_SET3_i)

Address: Base address + 4_8118h offset

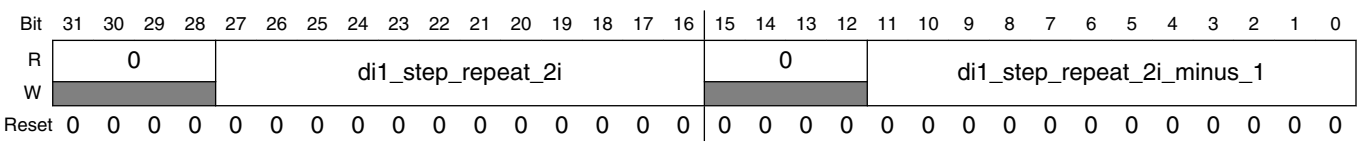


IPUx_DI1_DW_SET3_i field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24–16 di1_data_cnt_down3_i	Waveform's falling edge position. This field defines the Waveform's falling edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>
15–9 Reserved	This read-only field is reserved and always has the value 0.
di1_data_cnt_up3_i	Waveform's rising edge position. This field defines the Waveform's rising edge position. The Waveform is mapped to a pint according to the corresponding di1_pt_*_<i>

37.5.256 DI1 Step Repeat <i> Registers (IPUx_D1_STP_REP_i)

Address: Base address + 4_8148h offset



IPUx_D1_STP_REP_i field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 di1_step_repeat_2i	Step Repeat <i> This fields defines the amount of repetitions that will be performed by the counter <i>
15–12 Reserved	This read-only field is reserved and always has the value 0.
di1_step_repeat_2i_minus_1	Step Repeat <i> This fields defines the amount of repetitions that will be performed by the counter <i>

37.5.257 DI1Step Repeat 9 Registers (IPUx_DI1_STP_REP_9)

Address: Base address + 4_8158h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																				di1_step_repeat_9											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_STP_REP_9 field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
di1_step_repeat_9	Step Repeat 9 This fields defines the amount of repetitions that will be performed by the counter 9.

37.5.258 DI1 Serial Display Control Register (IPUx_DI1_SER_CONF)

Address: Base address + 4_815Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	DI1_SERIAL_LLA_PNTR_RS_R_1				DI1_SERIAL_LLA_PNTR_RS_R_0				DI1_SERIAL_LLA_PNTR_RS_W_1				DI1_SERIAL_LLA_PNTR_RS_W_0			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	DI1_SERIAL_LATCH								0		DI1_LLA_SER_ACCESS	DI1_SER_CLK_POLARITY	DI1_SERIAL_DATA_POLARITY	DI1_SERIAL_RS_POLARITY	DI1_SERIAL_CS_POLARITY	DI1_WAIT4SERIAL
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_SER_CONF field descriptions

Field	Description
31–28 DI1_SERIAL_LLA_PNTR_RS_R_1	RS 3 waveform pointer for read low level access This pointer defines which waveform set will be chosen when the read low level access is targeted to RS group 1. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
27–24 DI1_SERIAL_LLA_PNTR_RS_R_0	RS 2 waveform pointer for low level access This pointer defines which waveform set will be chosen when the read low level access is targeted to RS group 0. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
23–20 DI1_SERIAL_LLA_PNTR_RS_W_1	RS 1 waveform pointer for write low level access This pointer defines which waveform set will be chosen when the write low level access is targeted to RS group 1. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
19–16 DI1_SERIAL_LLA_PNTR_RS_W_0	RS 0 waveform pointer for write low level access This pointer defines which waveform set will be chosen when the write low level access is targeted to RS group 0. 0000 Waveform set #1 0001 Waveform set #2 1011 Waveform set #12 1100 Reserved 1101 Reserved 1100 Reserved
15–8 DI1_SERIAL_LATCH	DI1 Serial Latch This field defines how many cycles to insert between serial read accesses start to data sampling in the
7–6 Reserved	This read-only field is reserved and always has the value 0.

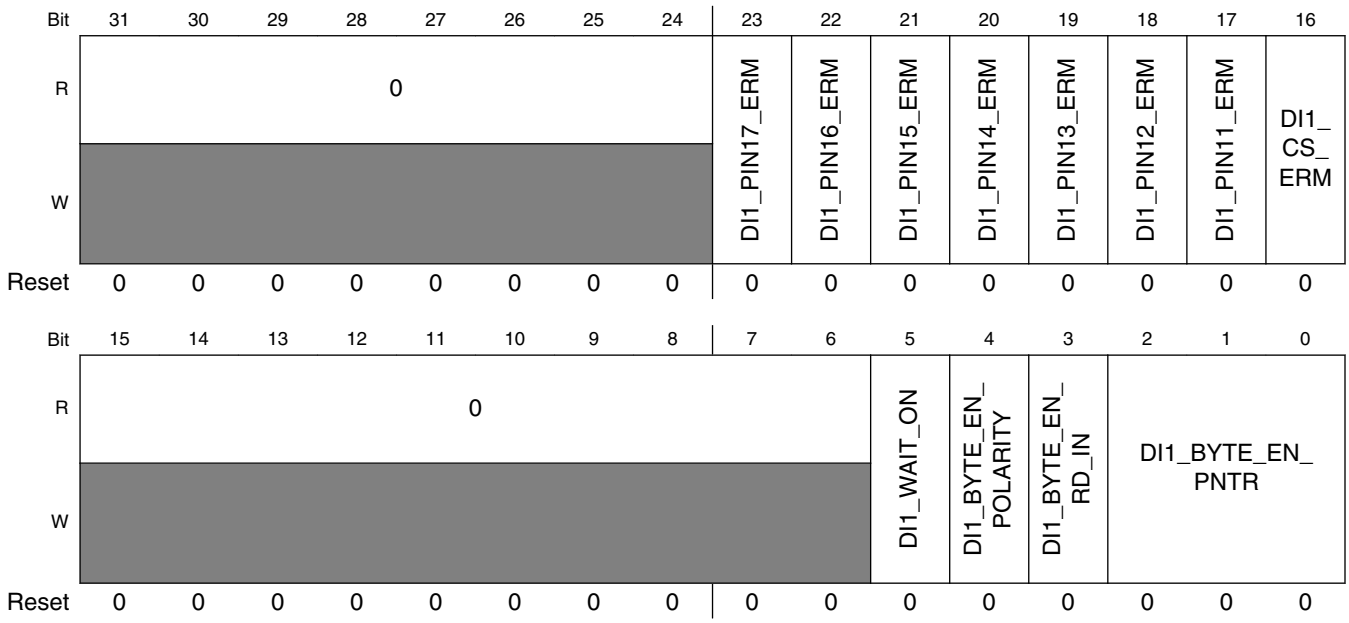
Table continues on the next page...

IPUx_DI1_SER_CONF field descriptions (continued)

Field	Description
5 DI1_LLA_SER_ACCESS	Direct Low Level Access to Serial display 1 ARM platform access is performed via a direct path to the serial display in LLA mode, in this mode only the ARM platform in LLA mode can access the serial port 0 ARM platform access to the serial display port is not done directly, hence other source are allowed to access the serial port. The arbitration is done automatically
4 DI1_SER_CLK_POLARITY	Serial Clock Polarity The output polarity of the SER_CLK pin 1 The clock is inverted 0 The clock is not inverted
3 DI1_SERIAL_DATA_POLARITY	Serial Data Polarity The output polarity of the SER_DATA pin 1 The data is inverted 0 The data is not inverted
2 DI1_SERIAL_RS_POLARITY	Serial RS Polarity The output polarity of the SER_RS pin 1 The RS is inverted 0 The RS is not inverted
1 DI1_SERIAL_CS_POLARITY	Serial Chip Select Polarity The output polarity of the SER_CS pin 1 The CS is inverted 0 The CS is not inverted
0 DI1_WAIT4SERIAL	Wait for Serial When the parallel display share pins with the serial port. Accessing the parallel port is not allowed till the serial port completes its access. 1 The parallel port should wait to the serial port as the pins are shared 0 The parallel port should not wait to the serial port as the pins are not shared

37.5.259 DI1 Special Signals Control Register (IPUx_DI1_SSC)

Address: Base address + 4_8160h offset



IPUx_DI1_SSC field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23 DI1_PIN17_ERM	DI1 PIN17 error recovery mode. This bit defines the error recovery mode of the PIN17 pin. 1 The PIN17 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN17 pin following a display error detection.
22 DI1_PIN16_ERM	DI1 PIN16 error recovery mode. This bit defines the error recovery mode of the PIN16 pin. 1 The PIN16 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN16 pin following a display error detection.
21 DI1_PIN15_ERM	DI1 PIN15 error recovery mode. This bit defines the error recovery mode of the PIN15 pin. 1 The PIN15 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN15 pin following a display error detection.
20 DI1_PIN14_ERM	DI1 PIN14 error recovery mode. This bit defines the error recovery mode of the PIN14 pin.

Table continues on the next page...

IPUx_DI1_SSC field descriptions (continued)

Field	Description
	1 The PIN14 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN14 pin following a display error detection.
19 DI1_PIN13_ERM	DI1 PIN13 error recovery mode. This bit defines the error recovery mode of the PIN13 pin. 1 The PIN13 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN13 pin following a display error detection.
18 DI1_PIN12_ERM	DI1 PIN12 error recovery mode. This bit defines the error recovery mode of the PIN12 pin. 1 The PIN12 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN12 pin following a display error detection.
17 DI1_PIN11_ERM	DI1 PIN11 error recovery mode. This bit defines the error recovery mode of the PIN11 pin. 1 The PIN11 pin is cleared in case of a synchronous display error. The clear will be done on the next VSYNC. 0 Nothing is done to the PIN11 pin following a display error detection.
16 DI1_CS_ERM	DI1 GLUELOGIC error recovery mode. This bit defines the error recovery mode of the GLUELOGIC. 1 The GLUELOGIC is release in case of a synchronous display error. The release will be done on the next VSYNC. 0 Nothing is done to the GLUELOGIC following a display error detection.
15–6 Reserved	This read-only field is reserved and always has the value 0.
5 DI1_WAIT_ON	Wait On This field defines the DC's response to WAIT signal 1 The DC holds the flow as long as WAIT is asserted. 0 The DC continues the flow regardless the WAIT signal.
4 DI1_BYTE_EN_POLARITY	Byte Enable polarity This bit defines the polarity of the byte enable signals to the display. 1 active high. 0 active low.
3 DI1_BYTE_EN_RD_IN	Byte Enable Read In This bit selects the source of the byte enable pins 1 The write byte enable signals are routed via bits [17:16] of the display's data, The read byte enable signals are routed via bits [19:18 of the display's data 0 The byte enable signals are routed via bits [17:16] of the display's data for both read and write
DI1_BYTE_EN_PNTR	Byte Enable Pointer This pointer selects the pin asserted along with the byte enables signals

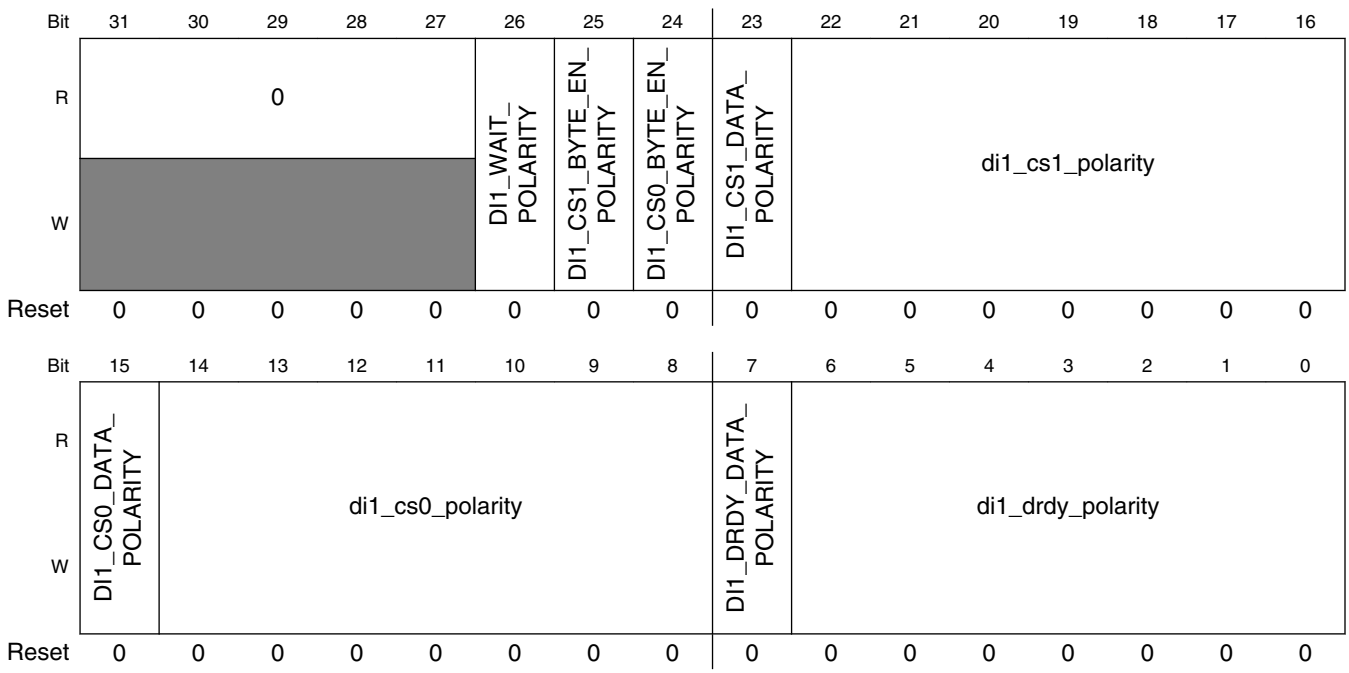
Table continues on the next page...

IPUx_DI1_SSC field descriptions (continued)

Field	Description
	000 wave form of byte enable as pin_11 001 wave form of byte enable as pin_12 111 wave form of byte enable as suitable CS pin

37.5.260 DI1 Polarity Register (IPUx_DI1_POL)

Address: Base address + 4_8164h offset



IPUx_DI1_POL field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26 DI1_WAIT_POLARITY	WAIT polarity This bit defines the polarity of the wait signal input coming from the displa1 1 active high 0 active low
25 DI1_CS1_BYTE_EN_POLARITY	Byte Enable associated with CS1 polarity This bit defines the polarity of the byte enable signals to the display 1 active high 0 active low

Table continues on the next page...

IPUx_DI1_POL field descriptions (continued)

Field	Description
24 DI1_CS0_BYTE_EN_POLARITY	Byte Enable associated with CS0 polarity This bit defines the polarity of the byte enable signals to the display 1 active high 0 active low
23 DI1_CS1_DATA_POLARITY	Data Polarity associated with CS1
22–16 di1_cs1_polarity	DI1 output pin's polarity for CS1 This bits define the polarity of each of the DI's outputs when CS1 is asserted 1 The output pin is active high 0 The output pin is active low
15 DI1_CS0_DATA_POLARITY	Data Polarity associated with CS0
14–8 di1_cs0_polarity	DI1 output pin's polarity for CS1 This bits define the polarity of each of the DI's outputs when CS1 is asserted 1 The output pin is active high 0 The output pin is active low
7 DI1_DRDY_DATA_POLARITY	Data Polarity associated with DRDY
di1_drdy_polarity	DI1 output dynamic pin's polarity for synchronous access This bits define the polarity of each of the DI's outputs when synchronous display access is asserted The pins' default polarity is the same as defined in the di0_drdy_polarity bits 1 The output pin is active high 0 The output pin is active low

37.5.261 DI1Active Window 0 Register (IPUx_DI1_AW0)

Address: Base address + 4_8168h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI1_AW_TRIG_SEL				DI1_AW_HEND												DI1_AW_HCOUNT_SEL				DI1_AW_HSTART											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_AW0 field descriptions

Field	Description
31–28 DI1_AW_TRIG_SEL	This field selects the trigger for sending data during the display's active window 000 disabled 001 The trigger is the same trigger that triggers the displays clock. 010 The trigger is counter #1 011 The trigger is counter #2 100 The trigger is counter #3 101 The trigger is counter #4 110 The trigger is counter #5 111 The trigger is always on.
27–16 DI1_AW_HEND	This field defines the horizontal end of the active window
15–12 DI1_AW_HCOUNT_SEL	This field selects the counter that counts the horizontal position of the display's active window 0000 disabled 0001 reserved 0010 The counter is counter #1 0011 The counter is counter #2 0100 The counter is counter #3 0101 The counter is counter #4 0110 The counter is counter #5 1001 The counter is counter #8
DI1_AW_HSTART	This field defines the horizontal start of the active window DI1_AW_HSTART < DI1_AW_HEND

37.5.262 DI1 Active Window 1 Register (IPUx_DI1_AW1)

Address: Base address + 4_816Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				DI1_AW_VEND												DI1_AW_VCOUNT_SEL				DI1_AW_VSTART											
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DI1_AW1 field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–16 DI1_AW_VEND	This field defines the vertical end of the active window
15–12 DI1_AW_VCOUNT_SEL	This field selects the counter that counts the vertical position of the display's active window 0000 disabled 0001 reserved

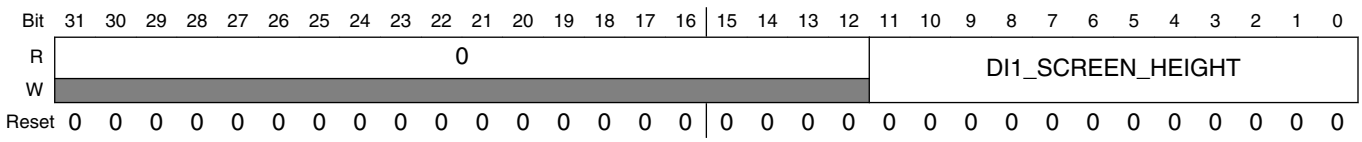
Table continues on the next page...

IPUx_DI1_AW1 field descriptions (continued)

Field	Description
	0010 The counter is counter #1 0011 The counter is counter #2 0100 The counter is counter #3 0101 The counter is counter #4 0110 The counter is counter #5 1001 The counter is counter #8
DI1_AW_ VSTART	This field defines the vertical start of the active window DI1_AW_VSTART < DI1_AW_VEND

37.5.263 DI1 Screen Configuration Register (IPUx_DI1_SCR_CONF)

Address: Base address + 4_8170h offset



IPUx_DI1_SCR_CONF field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
DI1_SCREEN_ HEIGHT	This field defines the number of display rows (Number_of_ROWS = DI1_SCREEN_HEIGHT+1) This field is used for VSYNC calculation and for anti-tearing

37.5.264 DI1 Status Register (IPUx_DI1_STAT)

Address: Base address + 4_8174h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0												DI1_CNTR_FIFO_FULL	DI1_CNTR_FIFO_EMPTY	DI1_READ_FIFO_FULL	DI1_READ_FIFO_EMPTY
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

IPUx_DI1_STAT field descriptions

Field	Description
31–4 Reserved	This read-only field is reserved and always has the value 0.
3 DI1_CNTR_ FIFO_FULL	This bit indicates a full state of the DI1 FIFO. This FIFO is part of the DI1 synchronizer.
2 DI1_CNTR_ FIFO_EMPTY	This bit indicates an empty state of the DI1 FIFO. This FIFO is part of the DI1 synchronizer.
1 DI1_READ_ FIFO_FULL	This bit indicates a full state of the DI1 FIFO when performing a read. This FIFO is part of the DI1 synchronizer.
0 DI1_READ_ FIFO_EMPTY	This bit indicates an empty state of the DI1 FIFO when performing a read. This FIFO is part of the DI1 synchronizer.

37.5.265 SMFC Mapping Register (IPUx_SMFC_MAP)

The purpose of this register is to map CSI frames to IDMAC channels.

Address: Base address + 5_0000h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																				MAP_ CH3			MAP_ CH2			MAP_ CH1			MAP_ CH0		
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SMFC_MAP field descriptions

Field	Description
31–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 MAP_CH3	DMASMFC channel 3mapping bits. 000 CSI0, ID=0 mapped to DMASMFC channel 3. 001 CSI0, ID=1 mapped to DMASMFC channel 3. 010 CSI0, ID=2 mapped to DMASMFC channel 3. 011 CSI0, ID=3 mapped to DMASMFC channel 3. 100 CSI1, ID=0 mapped to DMASMFC channel 3. 101 CSI1, ID=1 mapped to DMASMFC channel 3. 110 CSI1, ID=2 mapped to DMASMFC channel 3. 111 CSI1, ID=3 mapped to DMASMFC channel 3.
8–6 MAP_CH2	DMASMFC channel 2 mapping bits. 000 CSI0, ID=0 mapped to DMASMFC channel 2.

Table continues on the next page...

IPUx_SMFC_MAP field descriptions (continued)

Field	Description
	001 CSI0, ID=1 mapped to DMASMFC channel 2. 010 CSI0, ID=2 mapped to DMASMFC channel 2. 011 CSI0, ID=3 mapped to DMASMFC channel 2. 100 CSI1, ID=0 mapped to DMASMFC channel 2. 101 CSI1, ID=1 mapped to DMASMFC channel 2. 110 CSI1, ID=2 mapped to DMASMFC channel 2. 111 CSI1, ID=3 mapped to DMASMFC channel 2.
5–3 MAP_CH1	DMASMFC channel 1 mapping bits. 000 CSI0, ID=0 mapped to DMASMFC channel 1. 001 CSI0, ID=1 mapped to DMASMFC channel 1. 010 CSI0, ID=2 mapped to DMASMFC channel 1. 011 CSI0, ID=3 mapped to DMASMFC channel 1. 100 CSI1, ID=0 mapped to DMASMFC channel 1. 101 CSI1, ID=1 mapped to DMASMFC channel 1. 110 CSI1, ID=2 mapped to DMASMFC channel 1. 111 CSI1, ID=3 mapped to DMASMFC channel 1.
MAP_CH0	DMASMFC channel 0 mapping bits. 000 CSI0, ID=0 mapped to DMASMFC channel 0. 001 CSI0, ID=1 mapped to DMASMFC channel 0. 010 CSI0, ID=2 mapped to DMASMFC channel 0. 011 CSI0, ID=3 mapped to DMASMFC channel 0. 100 CSI1, ID=0 mapped to DMASMFC channel 0. 101 CSI1, ID=1 mapped to DMASMFC channel 0. 110 CSI1, ID=2 mapped to DMASMFC channel 0. 111 CSI1, ID=3 mapped to DMASMFC channel 0.

37.5.266 SMFC Watermark Control Register (IPUx_SMFC_WMC)

The purpose of this register is to control watermarks levels of DMA channels. The bit setting given relative to FIFO size and not in number of words since FIFO size depend from number of enabled DMA channels.

Address: Base address + 5_0004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0				WM3_			WM3_			WM2_			WM2_			0				WM1_			WM1_			WM0_			WM0_		
W					CLR			SET			CLR			SET							CLR			SET			CLR			SET		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	1	0

IPUx_SMFC_WMC field descriptions

Field	Description
31–28 Reserved	This read-only field is reserved and always has the value 0.
27–25 WM3_CLR	Watermark "clear" level of DMASMFC channel 3. 000 clear watermark level when FIFO is full on 1/8 of their size. 001 clear watermark level when FIFO is full on 2/8 of their size. 110 clear watermark level when FIFO is full on 7/8 of their size. 111 clear watermark level when FIFO is full.
24–22 WM3_SET	Watermark "set" level of DMASMFC channel 3 000 set watermark level when FIFO is full on 1/8 of their size. 001 set watermark level when FIFO is full on 2/8 of their size. 110 set watermark level when FIFO is full on 7/8 of their size. 111 set watermark level when FIFO is full.
21–19 WM2_CLR	Watermark "clear" level of DMASMFC channel 2. 000 clear watermark level when FIFO is full on 1/8 of their size. 001 clear watermark level when FIFO is full on 2/8 of their size. 110 clear watermark level when FIFO is full on 7/8 of their size. 111 clear watermark level when FIFO is full.
18–16 WM2_SET	Watermark "set" level of DMASMFC channel 2. 000 set watermark level when FIFO is full on 1/8 of their size. 001 set watermark level when FIFO is full on 2/8 of their size. 110 set watermark level when FIFO is full on 7/8 of their size. 111 set watermark level when FIFO is full.
15–12 Reserved	This read-only field is reserved and always has the value 0.
11–9 WM1_CLR	Watermark "clear" level of DMASMFC channel 1. 000 clear watermark level when FIFO is full on 1/8 of their size. 001 clear watermark level when FIFO is full on 2/8 of their size. 110 clear watermark level when FIFO is full on 7/8 of their size. 111 clear watermark level when FIFO is full.
8–6 WM1_SET	Watermark "set" level of DMASMFC channel 1. 000 set watermark level when FIFO is full on 1/8 of their size. 001 set watermark level when FIFO is full on 2/8 of their size. 110 set watermark level when FIFO is full on 7/8 of their size. 111 set watermark level when FIFO is full.
5–3 WM0_CLR	Watermark "clear" level of DMASMFC channel 0. 000 clear watermark level when FIFO is full on 1/8 of their size. 001 clear watermark level when FIFO is full on 2/8 of their size. 110 clear watermark level when FIFO is full on 7/8 of their size. 111 clear watermark level when FIFO is full.
WM0_SET	Watermark "set" level of DMASMFC channel 0.

Table continues on the next page...

IPUx_SMFC_WMC field descriptions (continued)

Field	Description
	000 set watermark level when FIFO is full on 1/8 of their size. 001 set watermark level when FIFO is full on 2/8 of their size. 110 set watermark level when FIFO is full on 7/8 of their size. 111 set watermark level when FIFO is full.

37.5.267 SMFC Burst Size Register (IPUx_SMFC_BS)

This register holds the burst size value for each DMASMFC channel. The burst size is the number of IDMAC's active accesses that will done for each IDMAC's burst. This number is a function of PFS, BPP & NPB parameters in the IDMAC's CPMEM. These are the parameters corresponding to the IDMAC's channel used. The table below describes what should be the burst size according to PFS, BPP & NPB settings

Table 37-721. SMFC Burst Size

BPP	PFS	BURST_SIZE
8	6	NPB[6:4]
16	6	NPB[6:3]
All other	All other	NPB[6:2]

Address: Base address + 5_0008h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																BURST3_SIZE				BURST2_SIZE				BURST1_SIZE				BURST0_SIZE			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_SMFC_BS field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–12 BURST3_SIZE	Burst Size of SMFCDMA channel 3. The value programed here should be BURST_SIZE-1 (for example: for a burst size of 1, write 0 to these bits)
11–8 BURST2_SIZE	Burst Size of SMFCDMA channel 2. The value programed here should be BURST_SIZE-1 (for example: for a burst size of 1, write 0 to these bits)
7–4 BURST1_SIZE	Burst Size of SMFCDMA channel 1. The value programed here should be BURST_SIZE-1 (for example: for a burst size of 1, write 0 to these bits)
BURST0_SIZE	Burst Size of SMFCDMA channel 0.

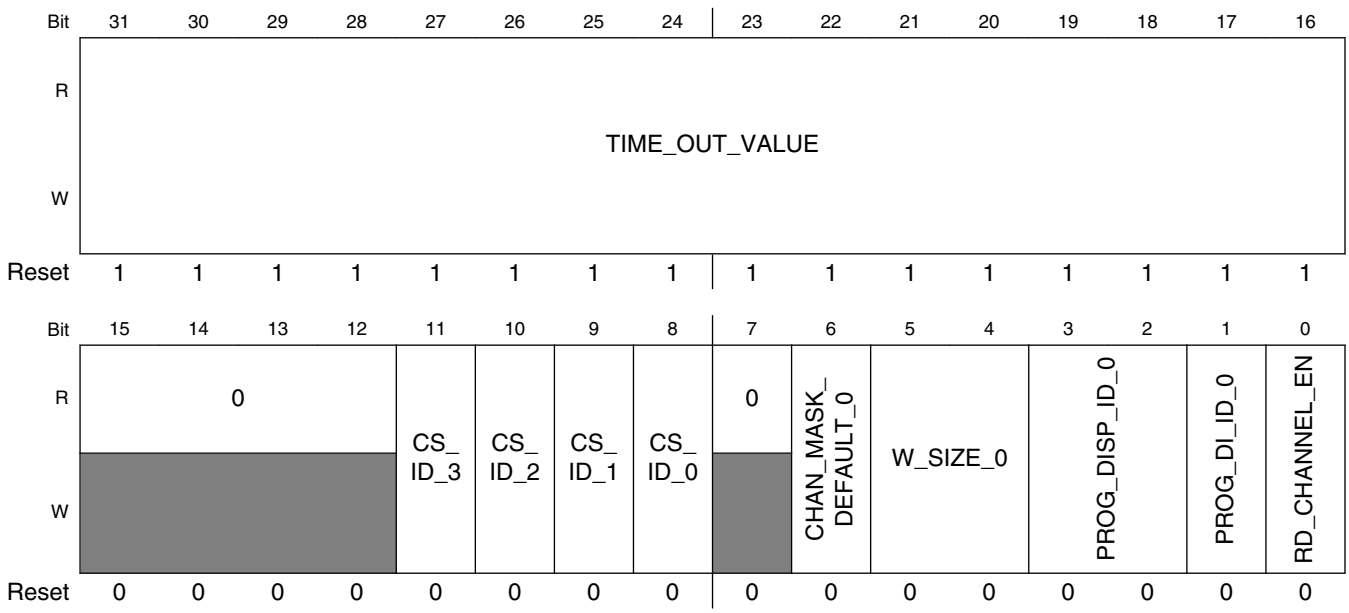
Table continues on the next page...

IPUx_SMFC_BS field descriptions (continued)

Field	Description
	The value programed here should be BURST_SIZE-1 (for example: for a burst size of 1, write 0 to these bits)

37.5.268 DC Read Channel Configuration Register (IPUx_DC_READ_CH_CONF)

Address: Base address + 5_8000h offset



IPUx_DC_READ_CH_CONF field descriptions

Field	Description
31–16 TIME_OUT_ VALUE	Time out value. In case of a error during read accesses to the display, where no response from the display was received. A time-out counter will terminate the current access and perform the next commend defined in the microcode. This field defines the amount of the hsp_clk cycles counted before the time-out event is issued. This event is tied to the interrupt controller and can generate an error interrupt.
15–12 Reserved	This read-only field is reserved and always has the value 0.
11 CS_ID_3	This bit maps an asynchronous display to a chip select 1 display #3 is connected to CS1 0 display #3 is connected to CS0
10 CS_ID_2	This bit maps an asynchronous display to a chip select 1 display #2 is connected to CS1 0 display #2 is connected to CS0

Table continues on the next page...

IPUx_DC_READ_CH_CONF field descriptions (continued)

Field	Description
9 CS_ID_1	This bit maps an asynchronous display to a chip select 1 display #1 is connected to CS1 0 display #1 is connected to CS0
8 CS_ID_0	This bit maps an asynchronous display to a chip select 1 display #0 is connected to CS1 0 display #0 is connected to CS0
7 Reserved	This read-only field is reserved and always has the value 0.
6 CHAN_MASK_DEFAULT_0	Event mask bit for the read channel When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
5–4 W_SIZE_0	Word Size The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used
3–2 PROG_DISP_ID_0	The field defines which one of the 4 displays can be read. 00 display #0 01 display #1 10 display #2 11 display #3
1 PROG_DI_ID_0	This bit select the DI which a read transaction can be performed through 1 DI #1 0 DI #0
0 RD_CHANNEL_EN	This bit enables the read channel. 1 The Read channel is enabled 0 The Read channel is disabled

37.5.269 DC Read Channel Start Address Register (IPUx_DC_READ_SH_ADDR)

Address: Base address + 5_8004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																ST_ADDR_0															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_READ_SH_ADDR field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_0	This field defines the start address within the display's memory space where the read transactions will be done from.

37.5.270 DC Routine Link Register 0 Channel 0 (IPUx_DC_RL0_CH_0)

Address: Base address + 5_8008h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NL_START_CHAN_0								0				COD_NL_PRIORITY_CHAN_0				COD_NF_START_CHAN_0								0				COD_NF_PRIORITY_CHAN_0			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL0_CH_0 field descriptions

Field	Description
31–24 COD_NL_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new line event (NL) resides
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NL_PRIORITY_CHAN_0	This field defines the priority of the new line (NL) event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NF_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new Frame event (NF) resides
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NF_PRIORITY_CHAN_0	This field defines the priority of the new frame (NF) event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2

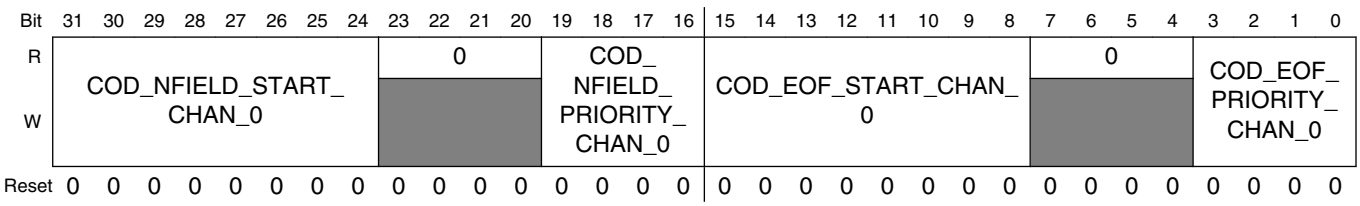
Table continues on the next page...

IPUx_DC_RL0_CH_0 field descriptions (continued)

Field	Description
	1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.271 DC Routine Link Register 1 Channel 0 (IPUx_DC_RL1_CH_0)

Address: Base address + 5_800Ch offset



IPUx_DC_RL1_CH_0 field descriptions

Field	Description
31–24 COD_NFIELD_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new field event resides
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NFIELD_PRIORITY_CHAN_0	This field defines the priority of the new field event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOF_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the end-of-frame event (EOF) resides
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOF_PRIORITY_CHAN_0	This field defines the priority of the end-of-frame event (EOF) event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2

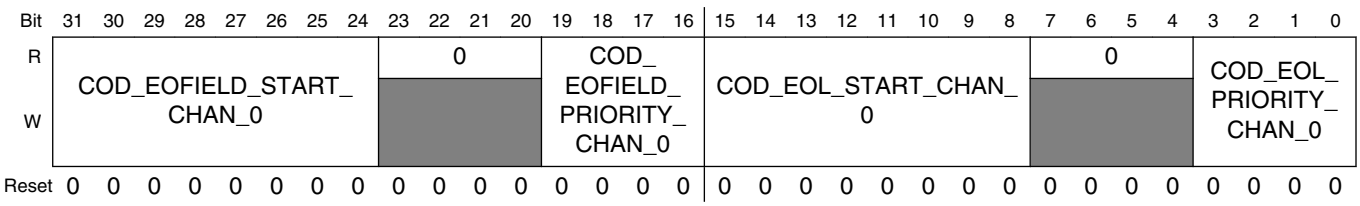
Table continues on the next page...

IPUx_DC_RL1_CH_0 field descriptions (continued)

Field	Description
	1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.272 DC Routine Link Register2 Channel 0 (IPUx_DC_RL2_CH_0)

Address: Base address + 5_8010h offset



IPUx_DC_RL2_CH_0 field descriptions

Field	Description
31–24 COD_EOFIELD_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the end-of-field event resides
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_EOFIELD_PRIORITY_CHAN_0	This field defines the priority of the end-of-field event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOL_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the end-of-line event (EOL) resides
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOL_PRIORITY_CHAN_0	This field defines the priority of the end-of-line event (EOL) event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2

Table continues on the next page...

IPUx_DC_RL2_CH_0 field descriptions (continued)

Field	Description
	1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.273 DC Routine Link Registe3 Channel 0 (IPUx_DC_RL3_CH_0)

Address: Base address + 5_8014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_0								0				COD_NEW_CHAN_PRIORITY_CHAN_0				COD_NEW_ADDR_START_CHAN_0								0				COD_NEW_ADDR_PRIORITY_CHAN_0			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL3_CH_0 field descriptions

Field	Description
31–24 COD_NEW_CHAN_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NEW_CHAN_PRIORITY_CHAN_0	This field defines the priority of the new channel event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NEW_ADDR_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_0	This field defines the priority of the new address event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable

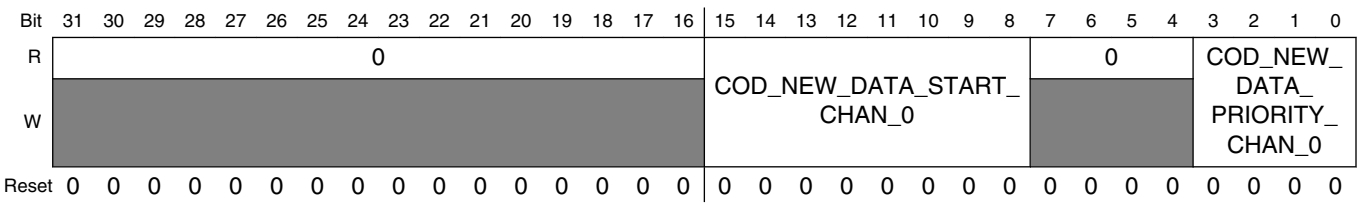
Table continues on the next page...

IPUx_DC_RL3_CH_0 field descriptions (continued)

Field	Description
	0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.274 DC Routine Link Register 4 Channel 0 (IPUx_DC_RL4_CH_0)

Address: Base address + 5_8018h offset

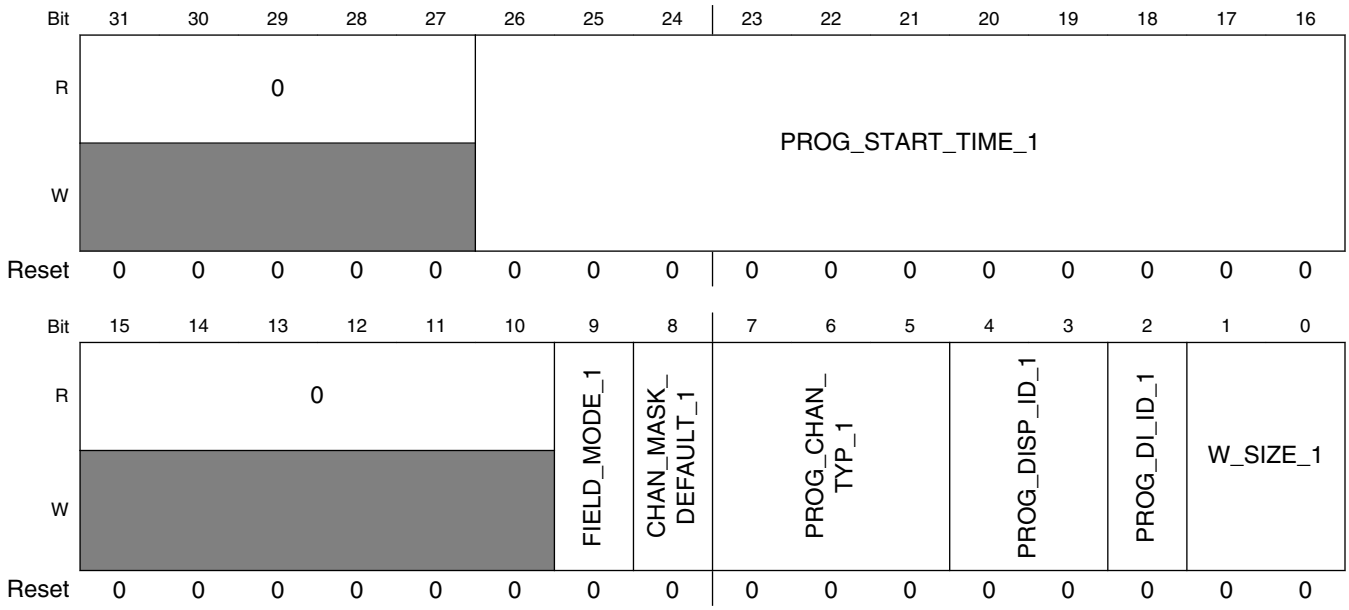


IPUx_DC_RL4_CH_0 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_0	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_0	This field defines the priority of the new data event The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.275 DC Write Channel 1 Configuration Register (IPUx_DC_WR_CH_CONF_1)

Address: Base address + 5_801Ch offset



IPUx_DC_WR_CH_CONF_1 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 PROG_START_TIME_1	This field defines the delay between display 's vertical synchronization pulse and the start time point of DC's channel 1 window. The delay is defined in pairs of rows. It is used for tearing elimination
15–10 Reserved	This read-only field is reserved and always has the value 0.
9 FIELD_MODE_1	Field mode bit for channel #1 This bit defines if the channel works in field mode or frame mode; This bit is relevant if the flow is sync flow 1 Field mode 0 Frame mode
8 CHAN_MASK_DEFAULT_1	Event mask bit for channel #1 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
7–5 PROG_CHAN_TYP_1	This field define the mode of operation of channel #1 000 Disable 001 Reserved

Table continues on the next page...

IPUx_DC_WR_CH_CONF_1 field descriptions (continued)

Field	Description
	010 Reserved 100 Normal mode without anti-tearing. For sync display this is the only mode allowed 101 Normal mode with anti-tearing 110 Reserved 111 Additional command channel is added to the flow handled by DC channel #1
4–3 PROG_DISP_ID_1	The field defines which one of the 4 displays is associated with channel #1. 00 display #0 01 display #1 10 display #2 11 display #3
2 PROG_DI_ID_1	This bit select the DI which a transaction associated with channel #1 can be performed to 1 DI #1 0 DI #0
W_SIZE_1	Word Size associated with channel #1 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.276 DC Write Channel 1 Address Configuration Register (IPUx_DC_WR_CH_ADDR_1)

Address: Base address + 5_8020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																ST_ADDR_1																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_ADDR_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_1	This field defines the start address within the display's memory space where the write transactions will be done to for channel #1.

37.5.277 DC Routine Link Register 0 Channel 1 (IPUx_DC_RL0_CH_1)

Address: Base address + 5_8024h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NL_START_CHAN_1								0				COD_NL_PRIORITY_CHAN_1				COD_NF_START_CHAN_1								0				COD_NF_PRIORITY_CHAN_1			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL0_CH_1 field descriptions

Field	Description
31–24 COD_NL_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new line event resides (associated with channel #1)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NL_PRIORITY_CHAN_1	This field defines the priority of the new line event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NF_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new frame event resides (associated with channel #1)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NF_PRIORITY_CHAN_1	This field defines the priority of the new frame event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.278 DC Routine Link Register 1 Channel 1 (IPUx_DC_RL1_CH_1)

Address: Base address + 5_8028h offset

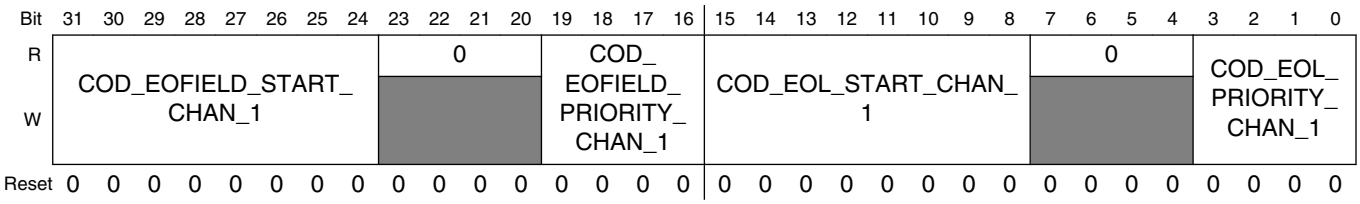
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NFIELD_START_CHAN_1								0				COD_NFIELD_PRIORITY_CHAN_1				COD_EOF_START_CHAN_1								0				COD_EOF_PRIORITY_CHAN_1			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL1_CH_1 field descriptions

Field	Description
31–24 COD_NFIELD_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new field event resides (associated with channel #1)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NFIELD_PRIORITY_CHAN_1	This field defines the priority of the new field event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOF_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the end of frame event resides (associated with channel #1)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOF_PRIORITY_CHAN_1	This field defines the priority of the end of frame event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.279 DC Routine Link Register 2 Channel 1 (IPUx_DC_RL2_CH_1)

Address: Base address + 5_8030h offset



IPUx_DC_RL2_CH_1 field descriptions

Field	Description
31–24 COD_EOFIELD_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the end of field event resides (associated with channel #1)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_EOFIELD_PRIORITY_CHAN_1	This field defines the priority of the end of field event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOL_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the end of line event resides (associated with channel #1)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOL_PRIORITY_CHAN_1	This field defines the priority of the end of line event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.280 DC Routine Link Register 3 Channel 1 (IPUx_DC_RL3_CH_1)

Address: Base address + 5_8032h offset

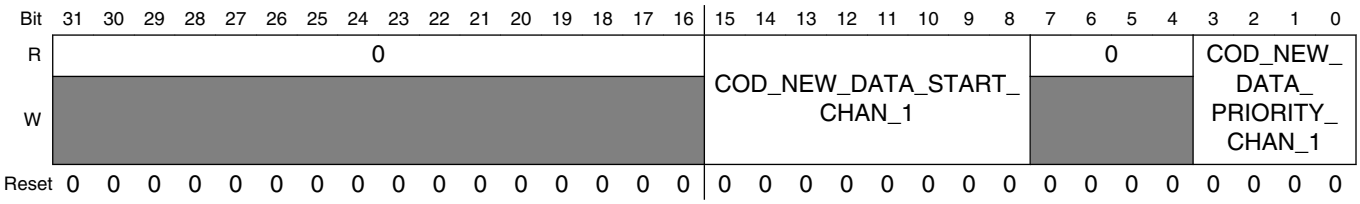
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_1								0				COD_NEW_CHAN_PRIORITY_CHAN_1				COD_NEW_ADDR_START_CHAN_1								0				COD_NEW_ADDR_PRIORITY_CHAN_1			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL3_CH_1 field descriptions

Field	Description
31–24 COD_NEW_CHAN_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #1)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NEW_CHAN_PRIORITY_CHAN_1	This field defines the priority of the new channel event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NEW_ADDR_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #1)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_1	This field defines the priority of the new address event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.281 DC Routine Link Register 4 Channel 1 (IPUx_DC_RL4_CH_1)

Address: Base address + 5_8034h offset

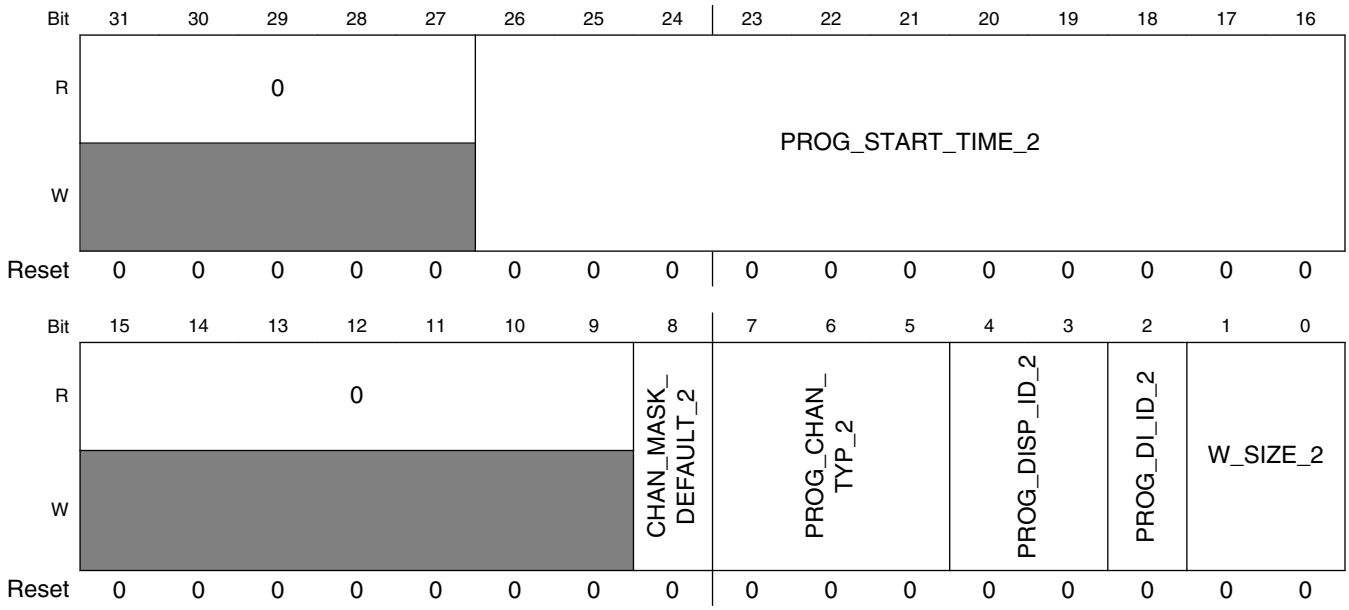


IPUx_DC_RL4_CH_1 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_1	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #1)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_1	This field defines the priority of the new data event (associated with channel #1) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.282 DC Write Channel 2 Configuration Register (IPUx_DC_WR_CH_CONF_2)

Address: Base address + 5_8038h offset



IPUx_DC_WR_CH_CONF_2 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 PROG_START_TIME_2	This field defines the delay between display 's vertical synchronization pulse and the start time point of DC's channel 2 window. The delay is defined in pairs of rows. It is used for tearing elimination
15–9 Reserved	This read-only field is reserved and always has the value 0.
8 CHAN_MASK_DEFAULT_2	Event mask bit for channel #2 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
7–5 PROG_CHAN_TYP_2	This field define the mode of operation of channel #2 000 Disable 001 Reserved 010 Reserved 100 Normal mode without anti-tearing. For sync display this is the only mode allowed 101 Normal mode with anti-tearing 110 Reserved 111 Additional command channel is added to the flow handled by DC channel #2

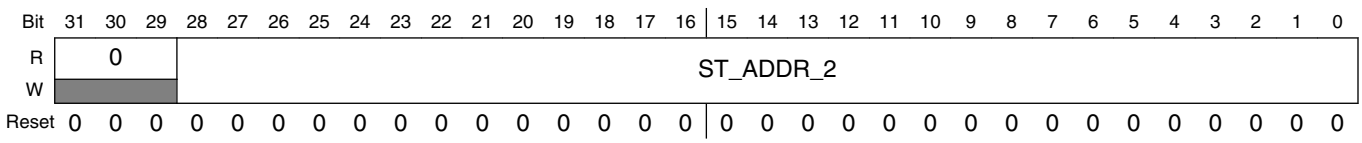
Table continues on the next page...

IPUx_DC_WR_CH_CONF_2 field descriptions (continued)

Field	Description
4–3 PROG_DISP_ID_2	The field defines which one of the 4 displays is associated with channel #2. 00 display #0 01 display #1 10 display #2 11 display #3
2 PROG_DI_ID_2	This bit select the DI which a transaction associated with channel #2 can be performed to 1 DI #1 0 DI #0
W_SIZE_2	Word Size The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.283 DC Write Channel 2 Address Configuration Register (IPUx_DC_WR_CH_ADDR_2)

Address: Base address + 5_803Ch offset



IPUx_DC_WR_CH_ADDR_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_2	This field defines the start address within the display's memory space where the write transactions will be done to for channel #2.

37.5.284 DC Routine Link Register 0 Channel 2 (IPUx_DC_RL0_CH_2)

Address: Base address + 5_8040h offset

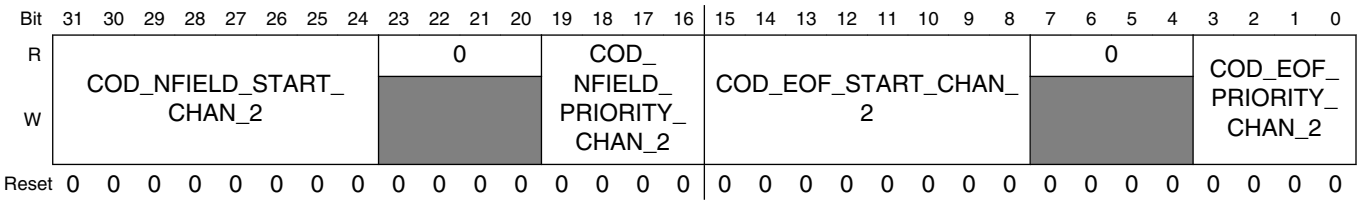
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NL_START_CHAN_2								0				COD_NL_PRIORITY_CHAN_2				COD_NF_START_CHAN_2								0				COD_NF_PRIORITY_CHAN_2			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL0_CH_2 field descriptions

Field	Description
31–24 COD_NL_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new line event resides (associated with channel #2)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NL_PRIORITY_CHAN_2	This field defines the priority of the new line event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NF_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new frame event resides (associated with channel #2)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NF_PRIORITY_CHAN_2	This field defines the priority of the new frame event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.285 DC Routine Link Register 1 Channel 2 (IPUx_DC_RL1_CH_2)

Address: Base address + 5_8044h offset



IPUx_DC_RL1_CH_2 field descriptions

Field	Description
31–24 COD_NFIELD_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new field event resides (associated with channel #2)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NFIELD_PRIORITY_CHAN_2	This field defines the priority of the new field event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOF_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the end of frame event resides (associated with channel #2)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOF_PRIORITY_CHAN_2	This field defines the priority of the end of frame event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.286 DC Routine Link Register 2 Channel 2 (IPUx_DC_RL2_CH_2)

Address: Base address + 5_8048h offset

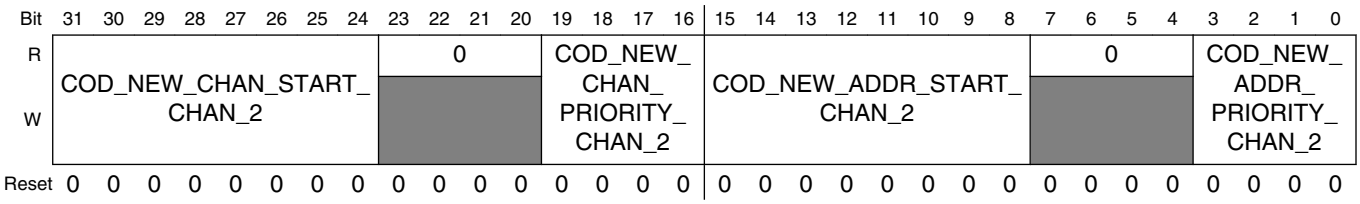
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_EOFIELD_START_CHAN_2								0				COD_EOFIELD_PRIORITY_CHAN_2				COD_EOL_START_CHAN_2								0				COD_EOL_PRIORITY_CHAN_2			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL2_CH_2 field descriptions

Field	Description
31–24 COD_EOFIELD_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the end of field event resides (associated with channel #2)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_EOFIELD_PRIORITY_CHAN_2	This field defines the priority of the end of field event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOL_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the end of line event resides (associated with channel #2)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOL_PRIORITY_CHAN_2	This field defines the priority of the end of line event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.287 DC Routine Link Register 3 Channel 2 (IPUx_DC_RL3_CH_2)

Address: Base address + 5_804Ch offset

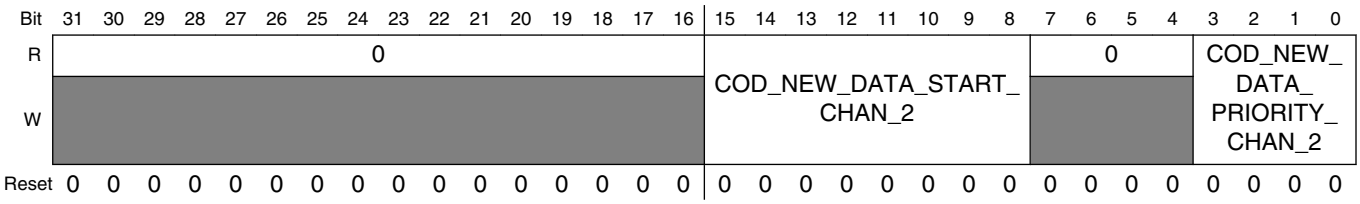


IPUx_DC_RL3_CH_2 field descriptions

Field	Description
31–24 COD_NEW_CHAN_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #2)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NEW_CHAN_PRIORITY_CHAN_2	This field defines the priority of the end of line event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NEW_ADDR_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #2)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_2	This field defines the priority of the end of line event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.288 DC Routine Link Register 4 Channel 2 (IPUx_DC_RL4_CH_2)

Address: Base address + 5_8050h offset

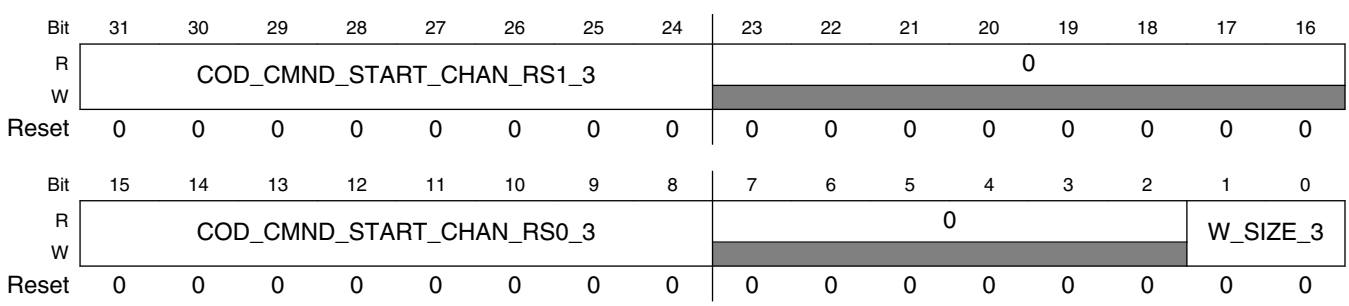


IPUx_DC_RL4_CH_2 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_2	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #2)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_2	This field defines the priority of the end of line event (associated with channel #2) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.289 DC Command Channel 3 Configuration Register (IPUx_DC_CMD_CH_CONF_3)

Address: Base address + 5_8054h offset



IPUx_DC_CMD_CH_CONF_3 field descriptions

Field	Description
31–24 COD_CMND_START_CHAN_RS1_3	This field is a pointer to the address within the microcode memory where the routine that handles the command start event resides (associated with channel #3); This field is relevant when RS is equal to 1
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_CMND_START_CHAN_RS0_3	This field is a pointer to the address within the microcode memory where the routine that handles the command start event resides (associated with channel #3); This field is relevant when RS is equal to 0
7–2 Reserved	This read-only field is reserved and always has the value 0.
W_SIZE_3	Word Size associated with channel #3 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.290 DC Command Channel 4 Configuration Register (IPUx_DC_CMD_CH_CONF_4)

Address: Base address + 5_8058h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	COD_CMND_START_CHAN_RS1_4								0							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_CMND_START_CHAN_RS0_4								0						W_SIZE_4	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_CMD_CH_CONF_4 field descriptions

Field	Description
31–24 COD_CMND_START_CHAN_RS1_4	This field is a pointer to the address within the microcode memory where the routine that handles the command start event resides (associated with channel #4); This field is relevant when RS is equal to 1
23–16 Reserved	This read-only field is reserved and always has the value 0.

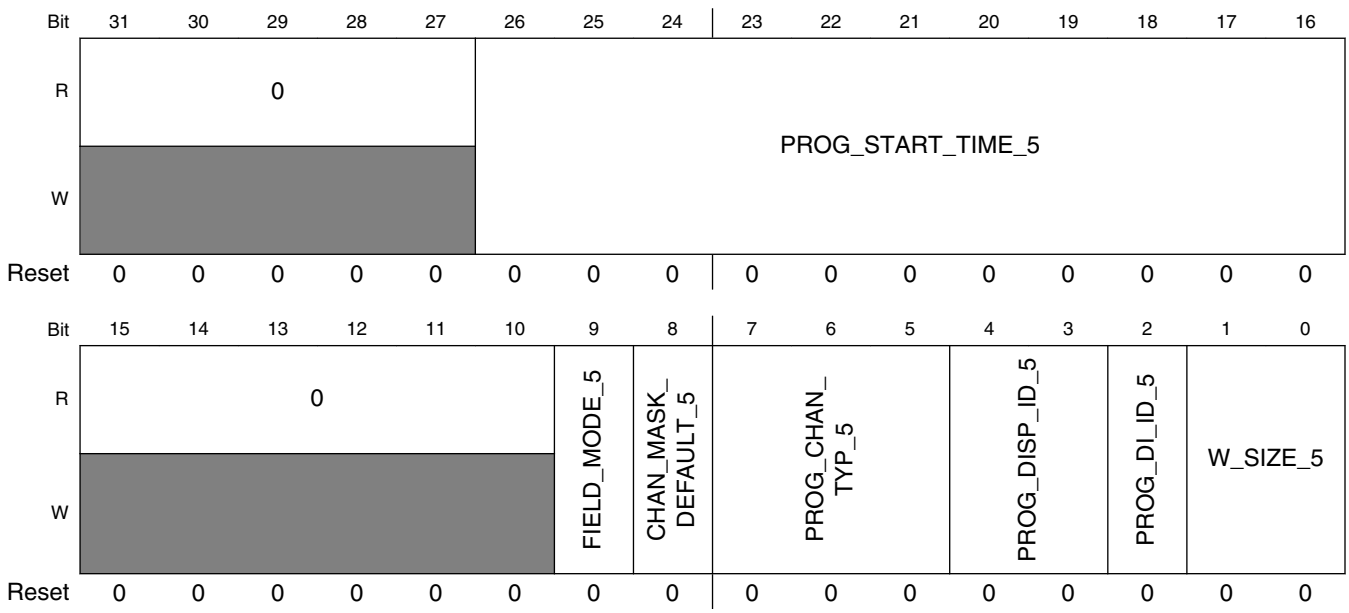
Table continues on the next page...

IPUx_DC_CMD_CH_CONF_4 field descriptions (continued)

Field	Description
15–8 COD_CMND_START_CHAN_RS0_4	This field is a pointer to the address within the microcode memory where the routine that handles the command start event resides (associated with channel #4); This field is relevant when RS is equal to 0
7–2 Reserved	This read-only field is reserved and always has the value 0.
W_SIZE_4	Word Size associated with channel #4 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.291 DC Write Channel 5Configuration Register (IPUx_DC_WR_CH_CONF_5)

Address: Base address + 5_805Ch offset



IPUx_DC_WR_CH_CONF_5 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 PROG_START_TIME_5	This field defines the delay between display 's vertical synchronization pulse and the start time point of DC's channel 5 window. The delay is defined in pairs of rows. It is used for tearing elimination

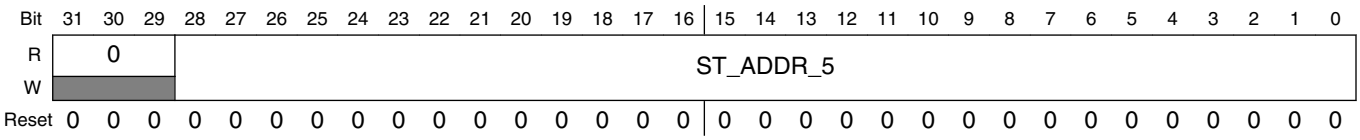
Table continues on the next page...

IPUx_DC_WR_CH_CONF_5 field descriptions (continued)

Field	Description
15–10 Reserved	This read-only field is reserved and always has the value 0.
9 FIELD_MODE_5	Field mode bit for channel #5 This bit defines if the channel works in field mode or frame mode; This bit is relevant if the flow is sync flow 1 Field mode 0 Frame mode
8 CHAN_MASK_DEFAULT_5	Event mask bit for channel #5 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
7–5 PROG_CHAN_TYP_5	This field define the mode of operation of channel #5 000 Disable 001 Reserved 010 Reserved 100 Normal mode without anti-tearing. For sync display this is the only mode allowed 101 Normal mode with anti-tearing 110 Reserved 111 Additional command channel is added to the flow handled by DC channel #5
4–3 PROG_DISP_ID_5	The field defines which one of the 4 displays is associated with channel #5. 00 display #0 01 display #1 10 display #2 11 display #3
2 PROG_DI_ID_5	This bit select the DI which a transaction associated with channel #5 can be performed to. When channel 28 is connected to DI0, channel 23 must be connected to DI1 even if ch23 is not used. This is done by writing 1 to this bit. 1 DI #1 0 DI #0
W_SIZE_5	Word Size associated with channel #5 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.292 DC Write Channel 5Address Configuration Register (IPUx_DC_WR_CH_ADDR_5)

Address: Base address + 5_8060h offset

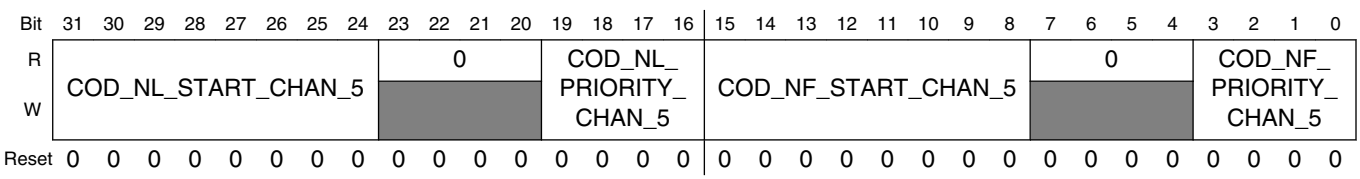


IPUx_DC_WR_CH_ADDR_5 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_5	This field defines the start address within the display's memory space where the write transactions will be done to for channel #5.

37.5.293 DC Routine Link Register 0 Channel 5 (IPUx_DC_RL0_CH_5)

Address: Base address + 5_8064h offset



IPUx_DC_RL0_CH_5 field descriptions

Field	Description
31–24 COD_NL_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new line event resides (associated with channel #5)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NL_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

Table continues on the next page...

IPUx_DC_RL0_CH_5 field descriptions (continued)

Field	Description
15–8 COD_NF_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new frame event resides (associated with channel #5)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NF_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.294 DC Routine Link Register 1 Channel 5 (IPUx_DC_RL1_CH_5)

Address: Base address + 5_8068h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NFIELD_START_CHAN_5								0				COD_NFIELD_PRIORITY_CHAN_5				COD_EOF_START_CHAN_5								0				COD_EOF_PRIORITY_CHAN_5			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL1_CH_5 field descriptions

Field	Description
31–24 COD_NFIELD_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new field event resides (associated with channel #5)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NFIELD_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

Table continues on the next page...

IPUx_DC_RL1_CH_5 field descriptions (continued)

Field	Description
15–8 COD_EOF_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the end of frame event resides (associated with channel #5)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOF_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.295 DC Routine Link Register 2 Channel 5 (IPUx_DC_RL2_CH_5)

Address: Base address + 5_806Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_EOFIELD_START_CHAN_5								0				COD_EOFIELD_PRIORITY_CHAN_5				COD_EOL_START_CHAN_5								0				COD_EOL_PRIORITY_CHAN_5			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL2_CH_5 field descriptions

Field	Description
31–24 COD_EOFIELD_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the end of field event resides (associated with channel #5)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_EOFIELD_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

Table continues on the next page...

IPUx_DC_RL2_CH_5 field descriptions (continued)

Field	Description
15–8 COD_EOL_ START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the end of line event resides (associated with channel #5)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOL_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.296 DC Routine Link Register3 Channel 5 (IPUx_DC_RL3_CH_5)

Address: Base address + 5_8070h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_5								0				COD_NEW_CHAN_PRIORITY_CHAN_5				COD_NEW_ADDR_START_CHAN_5								0				COD_NEW_ADDR_PRIORITY_CHAN_5			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL3_CH_5 field descriptions

Field	Description
31–24 COD_NEW_ CHAN_START_ CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #5)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NEW_ CHAN_ PRIORITY_ CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest)

Table continues on the next page...

IPUx_DC_RL3_CH_5 field descriptions (continued)

Field	Description
	1110 Reserved 1111 Reserved
15–8 COD_NEW_ADDR_START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #5)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.297 DC Routine Link Register 4 Channel 5 (IPUx_DC_RL4_CH_5)

Address: Base address + 5_8074h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																COD_NEW_DATA_START_CHAN_5								0				COD_NEW_DATA_PRIORITY_CHAN_5			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL4_CH_5 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_5	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #5)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_5	This field defines the priority of the new line event (associated with channel #5) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable)

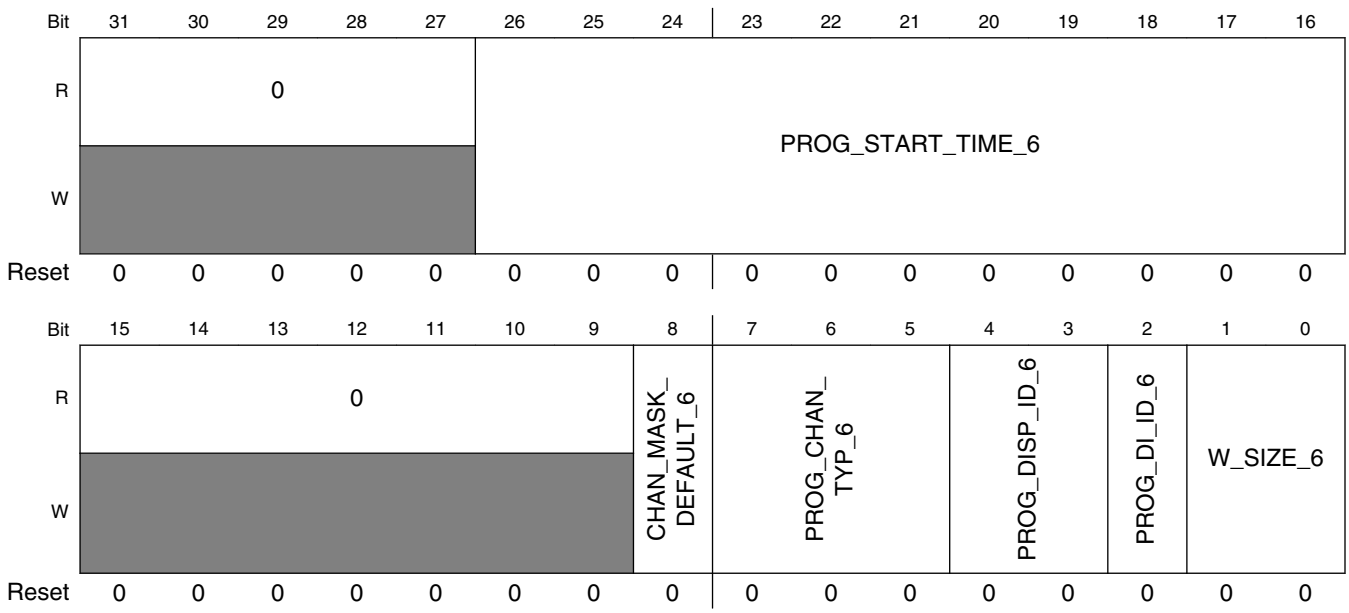
Table continues on the next page...

IPUx_DC_RL4_CH_5 field descriptions (continued)

Field	Description
	0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.298 DC Write Channel 6 Configuration Register (IPUx_DC_WR_CH_CONF_6)

Address: Base address + 5_8078h offset



IPUx_DC_WR_CH_CONF_6 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 PROG_START_TIME_6	This field defines the delay between display 's vertical synchronization pulse and the start time point of DC's channel 6 window. The delay is defined in pairs of rows. It is used for tearing elimination
15–9 Reserved	This read-only field is reserved and always has the value 0.
8 CHAN_MASK_DEFAULT_6	Event mask bit for channel #6 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event

Table continues on the next page...

IPUx_DC_WR_CH_CONF_6 field descriptions (continued)

Field	Description
	1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
7–5 PROG_CHAN_TYP_6	This field define the mode of operation of channel #6 000 Disable 001 Reserved 010 Reserved 100 Normal mode without anti-tearing. For sync display this is the only mode allowed 101 Normal mode with anti-tearing 110 Reserved 111 Additional command channel is added to the flow handled by DC channel #6
4–3 PROG_DISP_ID_6	The field defines which one of the 4 displays is associated with channel #6. 00 display #0 01 display #1 10 display #2 11 display #3
2 PROG_DI_ID_6	This bit select the DI which a transaction associated with channel #6 can be performed to 1 DI #1 0 DI #0
W_SIZE_6	Word Size associated with channel #6 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.299 DC Write Channel 6 Address Configuration Register (IPUx_DC_WR_CH_ADDR_6)

Address: Base address + 5_807Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																ST_ADDR_6															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_ADDR_6 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_6	This field defines the start address within the display's memory space where the write transactions will be done to for channel #6.

37.5.300 DC Routine Link Register 0Channel 6 (IPUx_DC_RL0_CH_6)

Address: Base address + 5_8080h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NL_START_CHAN_6								0				COD_NL_PRIORITY_CHAN_6				COD_NF_START_CHAN_6								0				COD_NF_PRIORITY_CHAN_6			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL0_CH_6 field descriptions

Field	Description
31–24 COD_NL_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new line event resides (associated with channel #6)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NL_PRIORITY_CHAN_6	This field defines the priority of the new line event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NF_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new frame event resides (associated with channel #6)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NF_PRIORITY_CHAN_6	This field defines the priority of the new frame event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.301 DC Routine Link Register 1 Channel 6 (IPUx_DC_RL1_CH_6)

Address: Base address + 5_8084h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NFIELD_START_CHAN_6								0				COD_NFIELD_PRIORITY_CHAN_6				COD_EOF_START_CHAN_6								0				COD_EOF_PRIORITY_CHAN_6			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL1_CH_6 field descriptions

Field	Description
31–24 COD_NFIELD_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new field event resides (associated with channel #6)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NFIELD_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOF_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the end of frame event resides (associated with channel #6)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOF_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.302 DC Routine Link Register 2 Channel 6 (IPUx_DC_RL2_CH_6)

Address: Base address + 5_8088h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_EOFIELD_START_CHAN_6								0				COD_EOFIELD_PRIORITY_CHAN_6				COD_EOL_START_CHAN_6								0				COD_EOL_PRIORITY_CHAN_6			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL2_CH_6 field descriptions

Field	Description
31–24 COD_EOFIELD_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the end of field event resides (associated with channel #6)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_EOFIELD_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_EOL_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the end of line event resides (associated with channel #6)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_EOL_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.303 DC Routine Link Register 3 Channel 6 (IPUx_DC_RL3_CH_6)

Address: Base address + 5_808Ch offset

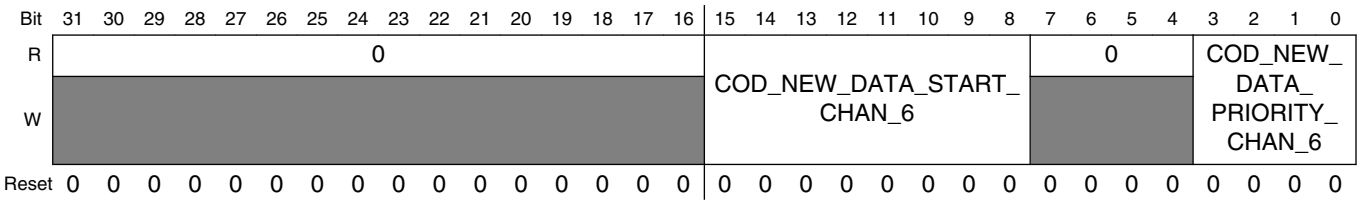
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_6								0				COD_NEW_CHAN_PRIORITY_CHAN_6				COD_NEW_ADDR_START_CHAN_6								0				COD_NEW_ADDR_PRIORITY_CHAN_6			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL3_CH_6 field descriptions

Field	Description
31–24 COD_NEW_CHAN_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #6)
23–20 Reserved	This read-only field is reserved and always has the value 0.
19–16 COD_NEW_CHAN_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
15–8 COD_NEW_ADDR_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #6)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.304 DC Routine Link Register 4 Channel 6 (IPUx_DC_RL4_CH_6)

Address: Base address + 5_8090h offset

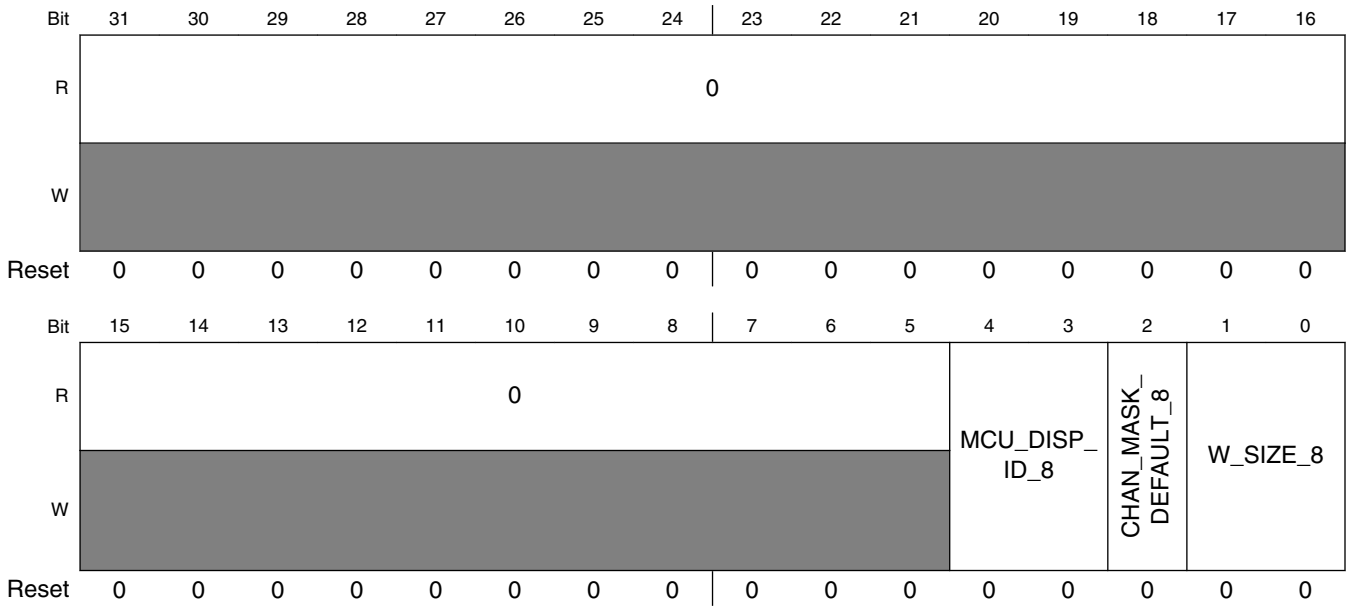


IPUx_DC_RL4_CH_6 field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_6	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #6)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_6	This field defines the priority of the new field event (associated with channel #6) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.305 DC Write Channel 8 Configuration 1Register (IPUx_DC_WR_CH_CONF1_8)

Address: Base address + 5_8094h offset



IPUx_DC_WR_CH_CONF1_8 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4–3 MCU_DISP_ID_8	The field defines which one of the 4 displays is associated with channel #8. 00 display #0 01 display #1 10 display #2 11 display #3
2 CHAN_MASK_DEFAULT_8	Event mask bit for channel #8 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
W_SIZE_8	Word Size associated with channel #8 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.306 DC Write Channel 8 Configuration 2 Register (IPUx_DC_WR_CH_CONF2_8)

Address: Base address + 5_8098h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			NEW_ADDR_SPACE_SA_8																												
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_CONF2_8 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
NEW_ADDR_ SPACE_SA_8	Channel #8 is used for ARM platform direct access to the display. This field defines the base address of the second region accessible on the display

37.5.307 DC Routine Link Register 1 Channel 8 (IPUx_DC_RL1_CH_8)

Address: Base address + 5_809Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_ADDR_START_CHAN_W_8_1								0								COD_NEW_ADDR_START_CHAN_W_8_0								0				COD_NEW_ADDR_PRIORITY_CHAN_8			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL1_CH_8 field descriptions

Field	Description
31–24 COD_NEW_ ADDR_START_ CHAN_W_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #8, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ ADDR_START_ CHAN_W_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #8, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ ADDR_	This field defines the priority of the new address event (associated with channel #8, both regions)

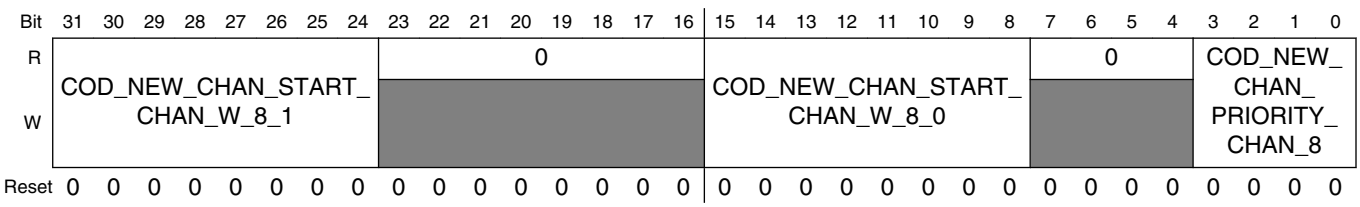
Table continues on the next page...

IPUx_DC_RL1_CH_8 field descriptions (continued)

Field	Description
PRIORITY_ CHAN_8	The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.308 DC Routine Link Register 2 Channel 8 (IPUx_DC_RL2_CH_8)

Address: Base address + 5_80A0h offset

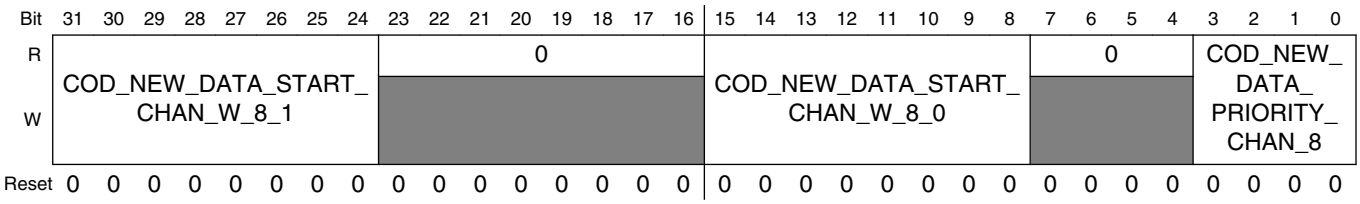


IPUx_DC_RL2_CH_8 field descriptions

Field	Description
31–24 COD_NEW_ CHAN_START_ CHAN_W_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #8, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ CHAN_START_ CHAN_W_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #8, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ CHAN_ PRIORITY_ CHAN_8	This field defines the priority of the new address event (associated with channel #8, both regions) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.309 DC Routine Link Register 3 Channel 8 (IPUx_DC_RL3_CH_8)

Address: Base address + 5_80A4h offset

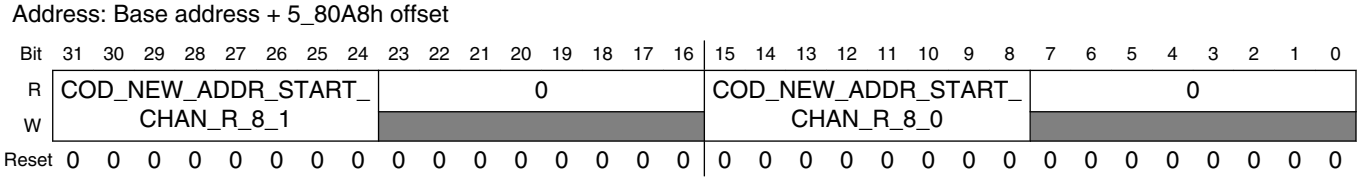


IPUx_DC_RL3_CH_8 field descriptions

Field	Description
31–24 COD_NEW_DATA_START_CHAN_W_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #8,second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_W_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #8, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_DATA_PRIORITY_CHAN_8	This field defines the priority of the new address event (associated with channel #8, both regions) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.310 DC Routine Link Register 4 Channel 8 (IPUx_DC_RL4_CH_8)

Address: Base address + 5_80A8h offset



IPUx_DC_RL4_CH_8 field descriptions

Field	Description
31–24 COD_NEW_ ADDR_START_ CHAN_R_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #8, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ ADDR_START_ CHAN_R_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #8, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.311 DC Routine Link Register 5 Channel 8 (IPUx_DC_RL5_CH_8)

Address: Base address + 5_80ACh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_R_8_1								0								COD_NEW_CHAN_START_CHAN_R_8_0								0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL5_CH_8 field descriptions

Field	Description
31–24 COD_NEW_ CHAN_START_ CHAN_R_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #8, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ CHAN_START_ CHAN_R_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #8, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.312 DC Routine Link Register 6 Channel 8 (IPUx_DC_RL6_CH_8)

Address: Base address + 5_80B0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_DATA_START_CHAN_R_8_1								0								COD_NEW_DATA_START_CHAN_R_8_0								0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL6_CH_8 field descriptions

Field	Description
31–24 COD_NEW_DATA_START_CHAN_R_8_1	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #8, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_DATA_START_CHAN_R_8_0	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #8, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.313 DC Write Channel 9 Configuration 1Register (IPUx_DC_WR_CH_CONF1_9)

Address: Base address + 5_80B4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											MCU_DISP_ID_9		CHAN_MASK_DEFAULT_9	W_SIZE_9	
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_CONF1_9 field descriptions

Field	Description
31–5 Reserved	This read-only field is reserved and always has the value 0.
4–3 MCU_DISP_ID_9	The field defines which one of the 4 displays is associated with channel #9. 00 display #0 01 display #1 10 display #2 11 display #3
2 CHAN_MASK_DEFAULT_9	Event mask bit for channel #9 When more then one event is used during a flow (EOF, EOL, NL, NF, EOFIELD, etc.) masks all the event besides the event that is defined as the highest priority event 1 All the events are used - no mask 0 Only the highest priority event is used, the rest are masked
W_SIZE_9	Word Size associated with channel #9 The data coming from the IDMAC is 32bit wide. This field defines the size of the word used by the DC 00 8 bits are used - a 32 bit words includes 4X8bit valid words. 01 16 LSB bits - a 32 bit words includes 2 X16bit valid words. 10 24 MSB bits are used (RGB) - 8 LSB bits are ignored by the DC 11 32 bits are used

37.5.314 DC Write Channel 9Configuration 2Register (IPUx_DC_WR_CH_CONF2_9)

Address: Base address + 5_80B8h offset

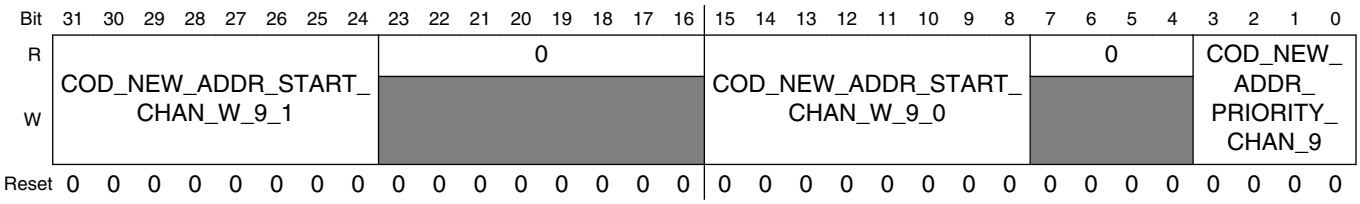
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																NEW_ADDR_SPACE_SA_9															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_CONF2_9 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
NEW_ADDR_SPACE_SA_9	Channel #8 is used for ARM platform direct access to the display. This field defines the base address of the second region accessible on the display

37.5.315 DC Routine Link Register 1 Channel 9 (IPUx_DC_RL1_CH_9)

Address: Base address + 5_80BCh offset

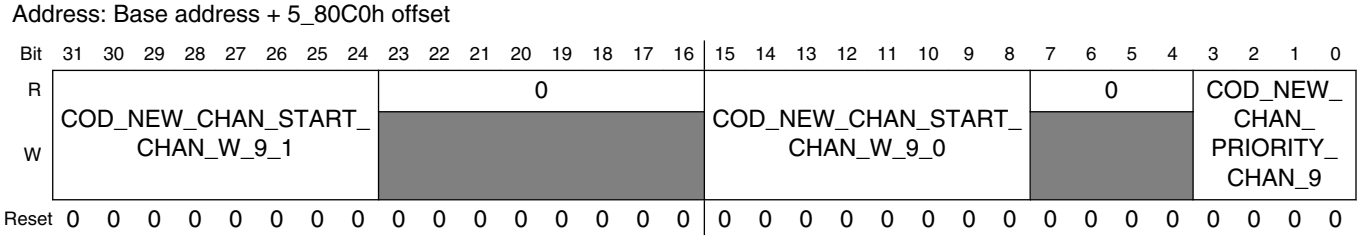


IPUx_DC_RL1_CH_9 field descriptions

Field	Description
31–24 COD_NEW_ADDR_START_CHAN_W_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ADDR_START_CHAN_W_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #9, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ADDR_PRIORITY_CHAN_9	This field defines the priority of the new address event (associated with channel #9, both regions) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.316 DC Routine Link Register 2 Channel 9 (IPUx_DC_RL2_CH_9)

Address: Base address + 5_80C0h offset



IPUx_DC_RL2_CH_9 field descriptions

Field	Description
31–24 COD_NEW_ CHAN_START_ CHAN_W_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ CHAN_START_ CHAN_W_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #9, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ CHAN_ PRIORITY_ CHAN_9	This field defines the priority of the new address event (associated with channel #9, both regions) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.317 DC Routine Link Register 3Channel 9 (IPUx_DC_RL3_CH_9)

Address: Base address + 5_80C4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_DATA_START_CHAN_W_9_1								0								COD_NEW_DATA_START_CHAN_W_9_0								0				COD_NEW_DATA_PRIORITY_CHAN_9			
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL3_CH_9 field descriptions

Field	Description
31–24 COD_NEW_ DATA_START_ CHAN_W_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DC_RL3_CH_9 field descriptions (continued)

Field	Description
15–8 COD_NEW_ DATA_START_ CHAN_W_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #9, first region)
7–4 Reserved	This read-only field is reserved and always has the value 0.
COD_NEW_ DATA_ PRIORITY_ CHAN_9	This field defines the priority of the new address event (associated with channel #9, both regions) The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved

37.5.318 DC Routine Link Register 4 Channel 9 (IPUx_DC_RL4_CH_9)

Address: Base address + 5_80C8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_ADDR_START_CHAN_R_9_1								0								COD_NEW_ADDR_START_CHAN_R_9_0								0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL4_CH_9 field descriptions

Field	Description
31–24 COD_NEW_ ADDR_START_ CHAN_R_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_ ADDR_START_ CHAN_R_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new address event resides (associated with channel #9, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.319 DC Routine Link Register 5 Channel 9 (IPUx_DC_RL5_CH_9)

Address: Base address + 5_80CCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_CHAN_START_CHAN_R_9_1								0								COD_NEW_CHAN_START_CHAN_R_9_0								0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL5_CH_9 field descriptions

Field	Description
31–24 COD_NEW_CHAN_START_CHAN_R_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.
15–8 COD_NEW_CHAN_START_CHAN_R_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new channel event resides (associated with channel #9, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.320 DC Routine Link Register 6 Channel 9 (IPUx_DC_RL6_CH_9)

Address: Base address + 5_80D0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	COD_NEW_DATA_START_CHAN_R_9_1								0								COD_NEW_DATA_START_CHAN_R_9_0								0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_RL6_CH_9 field descriptions

Field	Description
31–24 COD_NEW_DATA_START_CHAN_R_9_1	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #9, second region)
23–16 Reserved	This read-only field is reserved and always has the value 0.

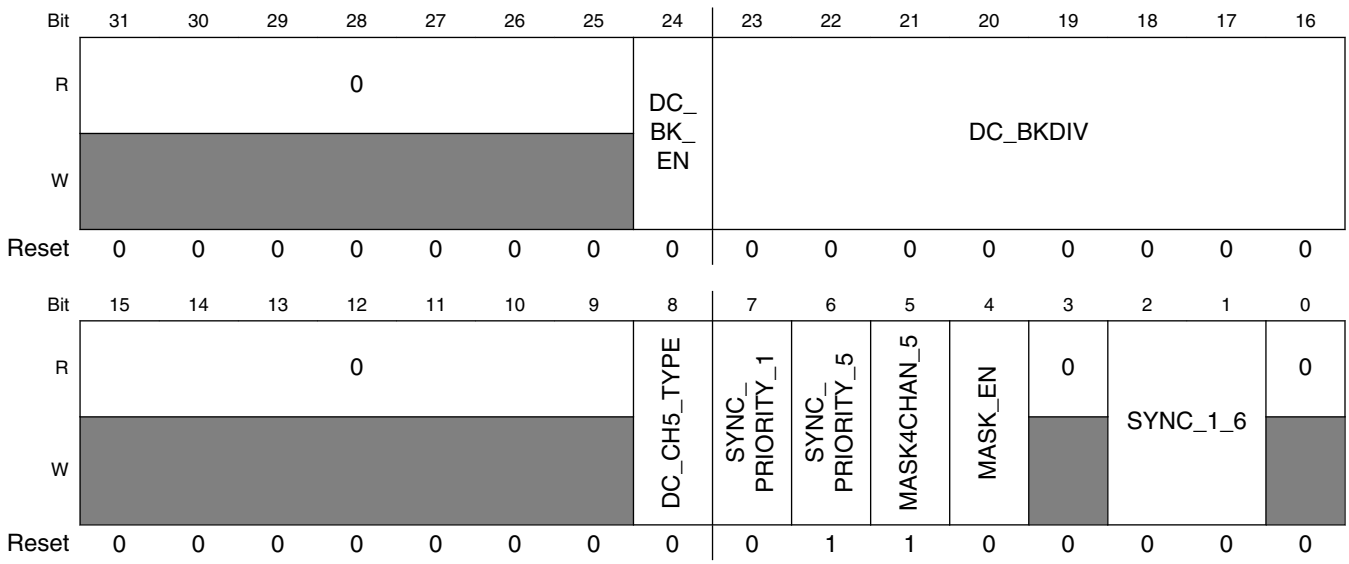
Table continues on the next page...

IPUx_DC_RL6_CH_9 field descriptions (continued)

Field	Description
15–8 COD_NEW_DATA_START_CHAN_R_9_0	This field is a pointer to the address within the microcode memory where the routine that handles the new data event resides (associated with channel #9, first region)
Reserved	This read-only field is reserved and always has the value 0.

37.5.321 DC General Register (IPUx_DC_GEN)

Address: Base address + 5_80D4h offset



IPUx_DC_GEN field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 DC_BK_EN	Cursor blinking enable 1 blinking is enabled 0 blinking is disabled
23–16 DC_BKDIV	Blinking Rate This field defines the blinking rate. The blinking occurs every N-th frame While N is defined by DC_BKDIV
15–9 Reserved	This read-only field is reserved and always has the value 0.
8 DC_CH5_TYPE	Channel 5 is used for synchronous flow. When this channel is used for accessing asynchronous display that is activated in a synchronous way

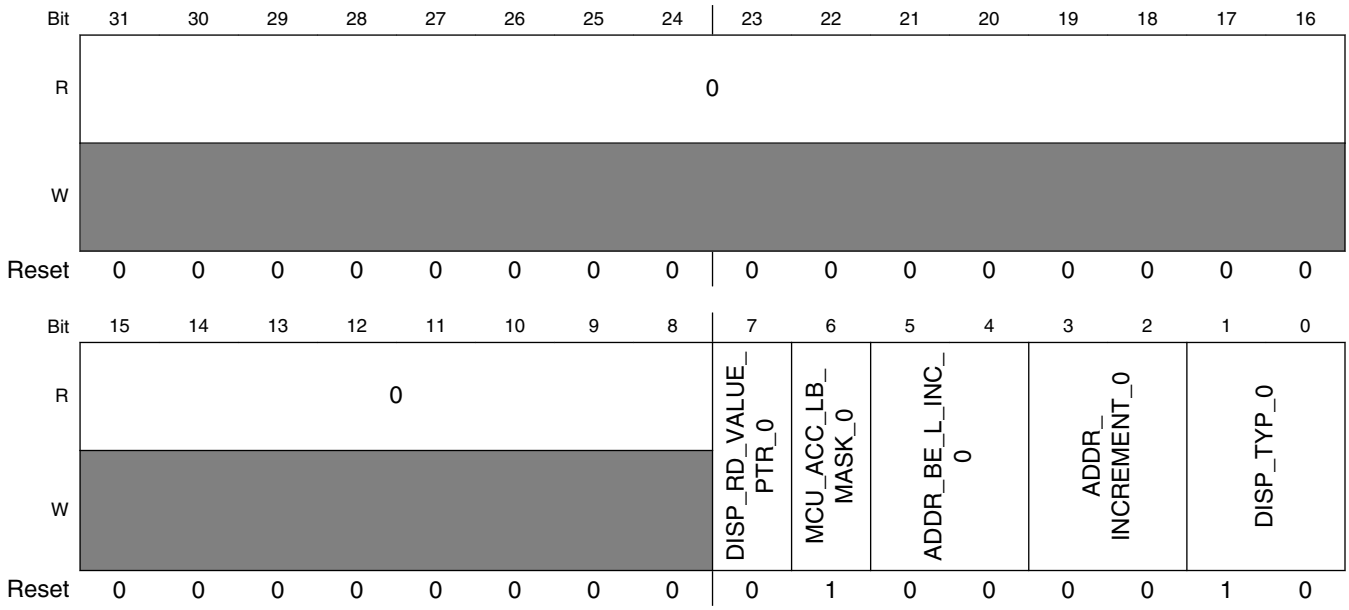
Table continues on the next page...

IPUx_DC_GEN field descriptions (continued)

Field	Description
	1 Enable the asynchronous interface via channel 5 0 normal mode, synchronous flow via channel 5
7 SYNC_ PRIORITY_1	When 2 sync flows are running, this bit sets the priority of channel #1. both SYNC_PRIORITY_5 and SYNC_PRIORITY_1 should not have the value of 0 This bit should be 1 high Priority 0 low Priority
6 SYNC_ PRIORITY_5	When 2 sync flows are running, this bit sets the priority of channel #5. both SYNC_PRIORITY_5 and SYNC_PRIORITY_1 should not have the value of 0 This bit should be 1 high priority 0 low Priority
5 MASK4CHAN_5	Sync flow can be associated with a mask channel. Only one sync flow can have a mask. This bit is ignored if MASK_EN is clear 1 mask channel is associated to the sync flow via DP 0 mask channel is associated to the sync flow via DC (without DP)
4 MASK_EN	Enable of the mask channel 1 mask channel is enabled 0 mask channel is disabled
3 Reserved	This read-only field is reserved and always has the value 0.
2–1 SYNC_1_6	This field 00 Channel 1 of the DC handles async flow 01 Illegal 10 Channel 1 of the DC handles sync flow 11 illegal
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.322 DC Display Configuration 1 Register 0 (IPUx_DC_DISP_CONF1_0)

Address: Base address + 5_80D8h offset



IPUx_DC_DISP_CONF1_0 field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 DISP_RD_VALUE_PTR_0	When the display works in wait for status mode. The IPU polls the display and compare the value to the value stored on the status DI_READ_DATA_ACK_VALUE and mask it with DI_READ_DATA_MASK. The DC holds 2 sets of mask and value. This field holds a pointer to the corresponding set of mask and value. 1 DI_READ_DATA_ACK_VALUE_1 & DI_READ_DATA_MASK_1 are used for display 0 0 DI_READ_DATA_ACK_VALUE_0 & DI_READ_DATA_MASK_0 are used for display 0
6 MCU_ACC_LB_MASK_0	The DC compares between the current access to the a calculated address. The calculated address is the next consecutive address following the last address. This bit defines the comparing mode 1 The 2 addresses are fully compared 0 The 2 addresses are compared, but the ADDR[0] bit of the new address is ignored
5–4 ADDR_BE_L_INC_0	This bits define the increment mode when the latest access was done with some of the byte enable signals are low, in that case different increment should be done instead of the normal auto increment of the address IF MCU_ACC_LB_MASK_0 is 0 then only 00 and 10 values are allowed. 00 No increment 01 Increment by 1 10 Increment by 2 11 Increment by 3

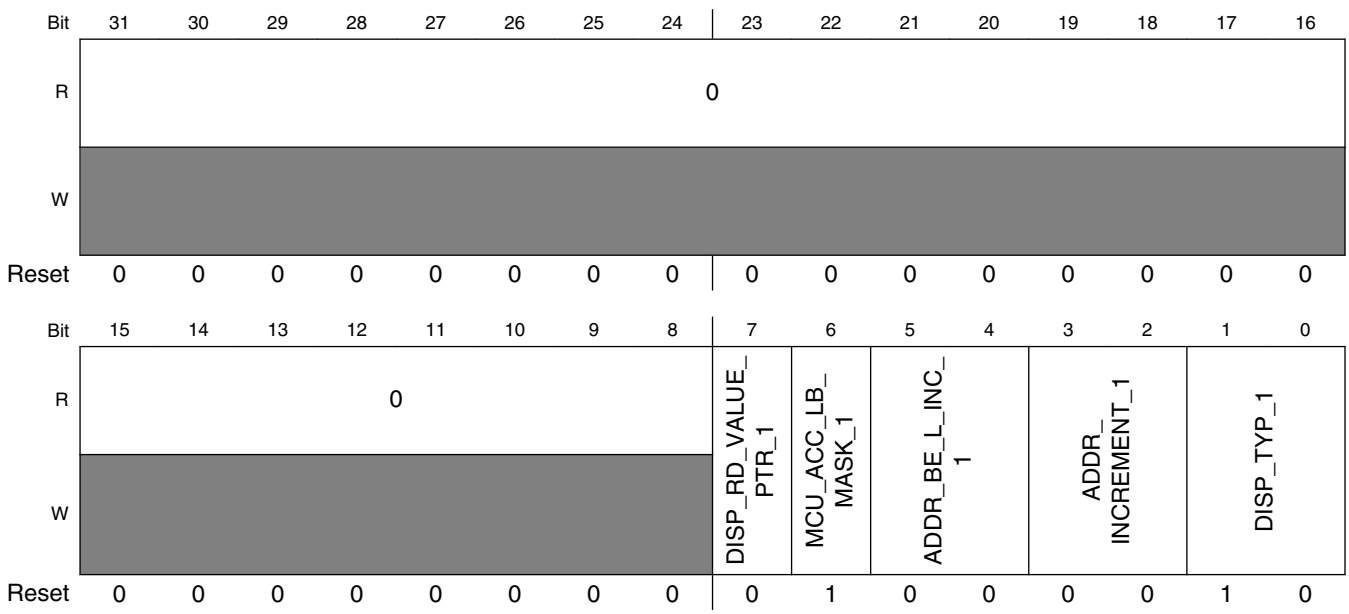
Table continues on the next page...

IPUx_DC_DISP_CONF1_0 field descriptions (continued)

Field	Description
3–2 ADDR_ INCREMENT_0	This field is the increment step for auto increment mode 00 Increment the address by 1 01 Increment the address by 2 10 Increment the address by 3 11 Increment the address by 4
DISP_TYP_0	This field defines the type of the display 00 Serial accesses display 01 Reserved 10 parallel display, without byte_enable support 11 parallel display, with byte_enable support

37.5.323 DC Display Configuration 1 Register 1 (IPUx_DC_DISP_CONF1_1)

Address: Base address + 5_80DCh offset



IPUx_DC_DISP_CONF1_1 field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 DISP_RD_ VALUE_PTR_1	When the display works in wait for status mode. The IPU polls the display and compare the value to the value stored on the status DI_READ_DATA_ACK_VALUE and mask it with DI_READ_DATA_MASK. The DC holds 2 sets of mask and value. This field holds a pointer to the corresponding set of mask and value.

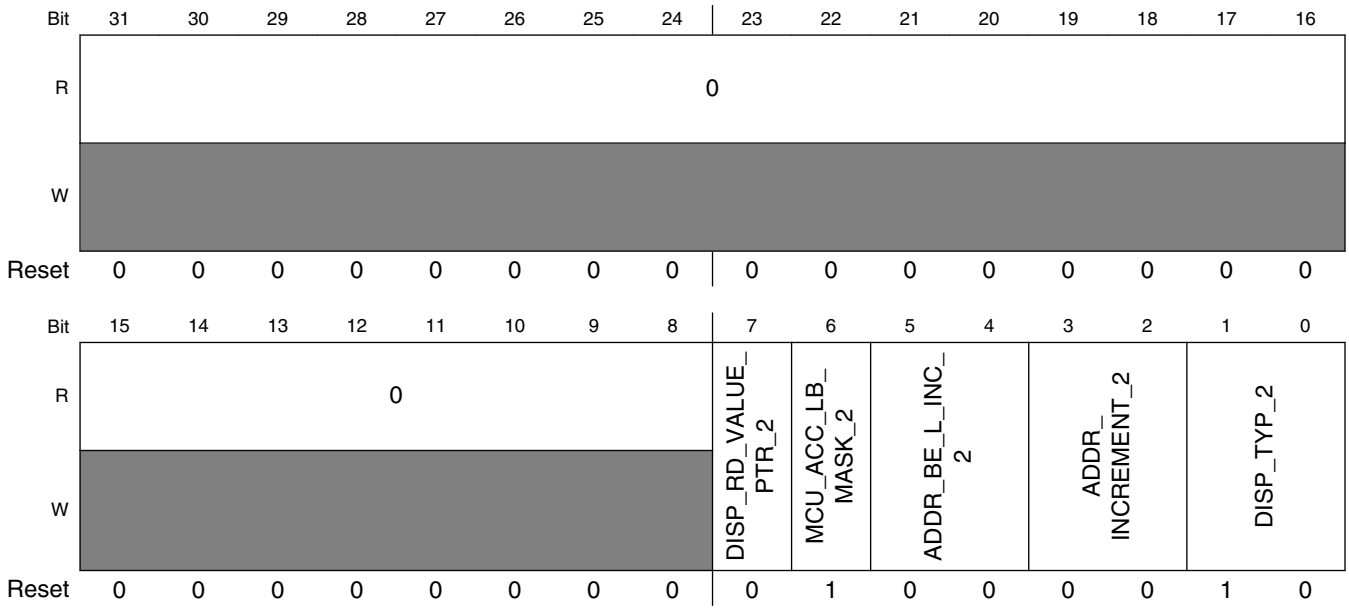
Table continues on the next page...

IPUx_DC_DISP_CONF1_1 field descriptions (continued)

Field	Description
	1 DI_READ_DATA_ACK_VALUE_1 & DI_READ_DATA_MASK_1 are used for display 1 0 DI_READ_DATA_ACK_VALUE_0 & DI_READ_DATA_MASK_0 are used for display 1
6 MCU_ACC_LB_MASK_1	The DC compares between the current access to the a calculated address. The calculated address is the next consecutive address following the last address. This bit defines the comparing mode 1 The 2 addresses are fully compared 0 The 2 addresses are compared, but the ADDR[0] bit of the new address is ignored
5–4 ADDR_BE_L_INC_1	This bits define the increment mode when the latest access was done with some of the byte enable signals are low, in that case different increment should be done instead of the normal auto increment of the address IF MCU_ACC_LB_MASK_1 is 0 then only 00 and 10 values are allowed. 00 No increment 01 Increment by 1 10 Increment by 2 11 Increment by 3
3–2 ADDR_INCREMENT_1	This field is the increment step for auto increment mode 00 Increment the address by 1 01 Increment the address by 2 10 Increment the address by 3 11 Increment the address by 4
DISP_TYP_1	This field defines the type of the display 00 Serial accesses display 01 Reserved 10 parallel display, without byte_enable support 11 parallel display, with byte_enable support

37.5.324 DC Display Configuration 1 Register 2 (IPUx_DC_DISP_CONF1_2)

Address: Base address + 5_80E0h offset



IPUx_DC_DISP_CONF1_2 field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 DISP_RD_VALUE_PTR_2	When the display works in wait for status mode. The IPU polls the display and compare the value to the value stored on the status DI_READ_DATA_ACK_VALUE and mask it with DI_READ_DATA_MASK. The DC holds 2 sets of mask and value. This field holds a pointer to the corresponding set of mask and value. 1 DI_READ_DATA_ACK_VALUE_1 & DI_READ_DATA_MASK_1 are used for display 2 0 DI_READ_DATA_ACK_VALUE_0 & DI_READ_DATA_MASK_0 are used for display 2
6 MCU_ACC_LB_MASK_2	The DC compares between the current access to the a calculated address. The calculated address is the next consecutive address following the last address. This bit defines the comparing mode 1 The 2 addresses are fully compared 0 The 2 addresses are compared, but the ADDR[0] bit of the new address is ignored
5–4 ADDR_BE_L_INC_2	This bits define the increment mode when the latest access was done with some of the byte enable signals are low, in that case different increment should be done instead of the normal auto increment of the address IF MCU_ACC_LB_MASK_2 is 0 then only 00 and 10 values are allowed. 00 No increment 01 Increment by 1 10 Increment by 2 11 Increment by 3

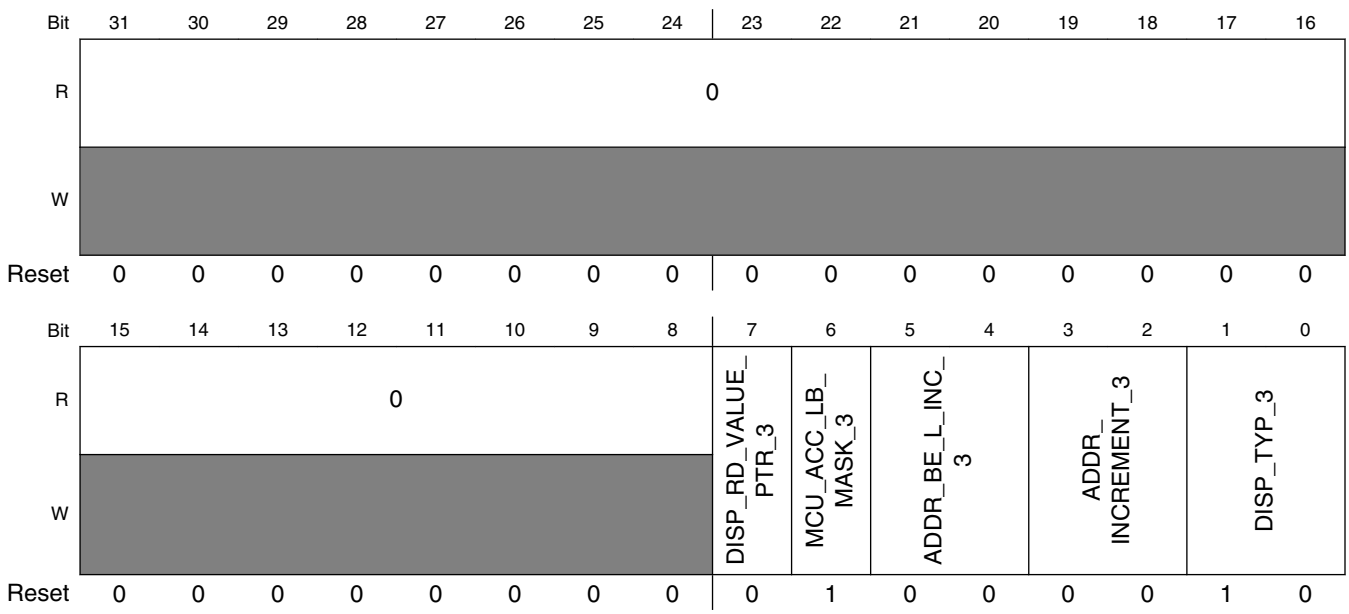
Table continues on the next page...

IPUx_DC_DISP_CONF1_2 field descriptions (continued)

Field	Description
3–2 ADDR_ INCREMENT_2	This field is the increment step for auto increment mode 00 Increment the address by 1 01 Increment the address by 2 10 Increment the address by 3 11 Increment the address by 4
DISP_TYP_2	This field defines the type of the display 00 Serial accesses display 01 Reserved 10 parallel display, without byte_enable support 11 parallel display, with byte_enable support

37.5.325 DC Display Configuration 1 Register 3 (IPUx_DC_DISP_CONF1_3)

Address: Base address + 5_80E4h offset



IPUx_DC_DISP_CONF1_3 field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 DISP_RD_ VALUE_PTR_3	When the display works in wait for status mode. The IPU polls the display and compare the value to the value stored on the status DI_READ_DATA_ACK_VALUE and mask it with DI_READ_DATA_MASK. The DC holds 2 sets of mask and value. This field holds a pointer to the corresponding set of mask and value.

Table continues on the next page...

IPUx_DC_DISP_CONF1_3 field descriptions (continued)

Field	Description
	1 DI_READ_DATA_ACK_VALUE_1 & DI_READ_DATA_MASK_1 are used for display 3 0 DI_READ_DATA_ACK_VALUE_0 & DI_READ_DATA_MASK_0 are used for display 3
6 MCU_ACC_LB_MASK_3	The DC compares between the current access to the a calculated address. The calculated address is the next consecutive address following the last address. This bit defines the comparing mode 1 The 2 addresses are fully compared 0 The 2 addresses are compared, but the ADDR[0] bit of the new address is ignored
5–4 ADDR_BE_L_INC_3	This bits define the increment mode when the latest access was done with some of the byte enable signals are low, in that case different increment should be done instead of the normal auto increment of the address IF MCU_ACC_LB_MASK_3 is 0 then only 00 and 10 values are allowed. 00 No increment 01 Increment by 1 10 Increment by 2 11 Increment by 3
3–2 ADDR_INCREMENT_3	This field is the increment step for auto increment mode 00 Increment the address by 1 01 Increment the address by 2 10 Increment the address by 3 11 Increment the address by 4
DISP_TYP_3	This field defines the type of the display 00 Serial accesses display 01 Reserved 10 parallel display, without byte_enable support 11 parallel display, with byte_enable support

37.5.326 DC Display Configuration 2 Register 0 (IPUx_DC_DISP_CONF2_0)

Address: Base address + 5_80E8h offset

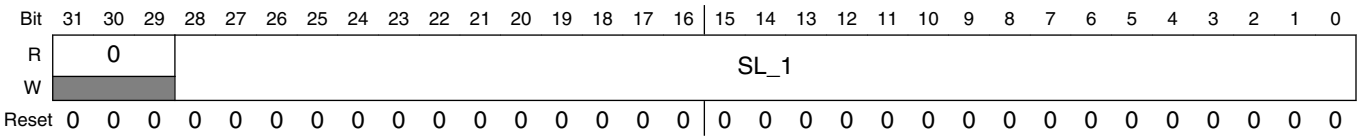
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																SL_0															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_DISP_CONF2_0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
SL_0	Stride line of display 0

37.5.327 DC Display Configuration 2 Register 1 (IPUx_DC_DISP_CONF2_1)

Address: Base address + 5_80ECh offset

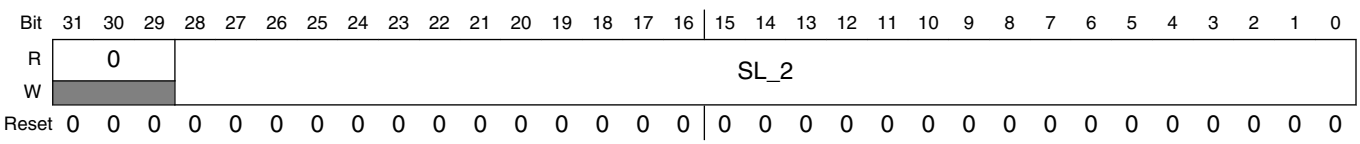


IPUx_DC_DISP_CONF2_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
SL_1	Stride line of display 1

37.5.328 DC Display Configuration 2 Register 2 (IPUx_DC_DISP_CONF2_2)

Address: Base address + 5_80F0h offset

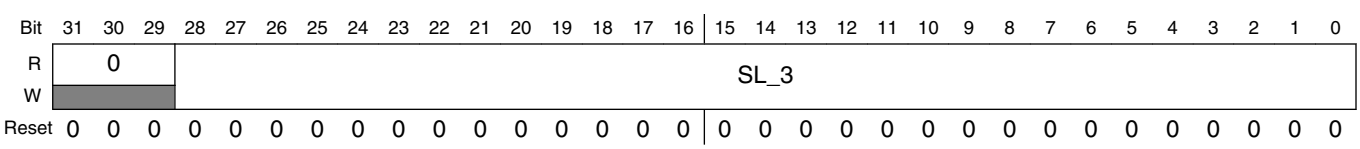


IPUx_DC_DISP_CONF2_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
SL_2	Stride line of display 2

37.5.329 DC Display Configuration 2 Register 3 (IPUx_DC_DISP_CONF2_3)

Address: Base address + 5_80F4h offset



IPUx_DC_DISP_CONF2_3 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
SL_3	Stride line of display 3

37.5.330 DC DI0Configuration Register 1 (IPUx_DC_DI0_CONF_1)

Address: Base address + 5_80F8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI_READ_DATA_MASK_0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_DI0_CONF_1 field descriptions

Field	Description
DI_READ_ DATA_MASK_0	This field defines the mask value of the data read from the display.

37.5.331 DC DI0Configuration Register 2 (IPUx_DC_DI0_CONF_2)

Address: Base address + 5_80FCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI_READ_DATA_ACK_VALUE_0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_DI0_CONF_2 field descriptions

Field	Description
DI_READ_ DATA_ACK_ VALUE_0	This is the expected data to be read from the display. The value reads from the display is anded with the DI_READ_DATA_MASK_0 and compared with the DI_READ_DATA_ACK_VALUE_0. This field is used for the READ_STATUS task of the DC

37.5.332 DC DI1Configuration Register 1 (IPUx_DC_DI1_CONF_1)

Address: Base address + 5_8100h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI_READ_DATA_MASK_1																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_DI1_CONF_1 field descriptions

Field	Description
DI_READ_ DATA_MASK_1	This field defines the mask value of the data read from the display.

37.5.333 DC DI1Configuration Register 2 (IPUx_DC_DI1_CONF_2)

Address: Base address + 5_8104h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DI_READ_DATA_ACK_VALUE_1																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_DI1_CONF_2 field descriptions

Field	Description
DI_READ_ DATA_ACK_ VALUE_1	This is the expected data to be read from the display. The value reads from the display is anded with the DI_READ_DATA_MASK_1 and compared with the DI_READ_DATA_ACK_VALUE_1 This field is used for the READ_STATUS task of the DC

37.5.334 DC Mapping Configuration Register 0 (IPUx_DC_MAP_CONF_0)

Address: Base address + 5_8108h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_1					MAPPING_PNTR_BYTE1_1					MAPPING_PNTR_BYTE0_1				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_0					MAPPING_PNTR_BYTE1_0					MAPPING_PNTR_BYTE0_0				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_0 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_1	Mapping pointer #1 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_1	Mapping pointer #1 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1

Table continues on the next page...

IPUx_DC_MAP_CONF_0 field descriptions (continued)

Field	Description
20–16 MAPPING_ PNTR_BYTE0_1	Mapping pointer #1 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_0	Mapping pointer #0 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_0	Mapping pointer #0 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_0	Mapping pointer #0 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.335 DC Mapping Configuration Register 1 (IPUx_DC_MAP_CONF_1)

Address: Base address + 5_810Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_3							MAPPING_PNTR_BYTE1_3			MAPPING_PNTR_BYTE0_3				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_2							MAPPING_PNTR_BYTE1_2			MAPPING_PNTR_BYTE0_2				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_1 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_3	Mapping pointer #3 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_3	Mapping pointer #3 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_3	Mapping pointer #3 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.

Table continues on the next page...

IPUx_DC_MAP_CONF_1 field descriptions (continued)

Field	Description
14–10 MAPPING_ PNTR_BYTE2_2	Mapping pointer #1 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_2	Mapping pointer #1 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_2	Mapping pointer #1 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.336 DC Mapping Configuration Register 2 (IPUx_DC_MAP_CONF_2)

Address: Base address + 5_8110h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_5					MAPPING_PNTR_BYTE1_5					MAPPING_PNTR_BYTE0_5				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_4					MAPPING_PNTR_BYTE1_4					MAPPING_PNTR_BYTE0_4				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_2 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_5	Mapping pointer #5 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_5	Mapping pointer #5 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_5	Mapping pointer #5 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_4	Mapping pointer #4 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_4	Mapping pointer #4 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1

Table continues on the next page...

IPUx_DC_MAP_CONF_2 field descriptions (continued)

Field	Description
MAPPING_ PNTR_BYTE0_4	Mapping pointer #4 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.337 DC Mapping Configuration Register 3 (IPUx_DC_MAP_CONF_3)

Address: Base address + 5_8114h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_7							MAPPING_PNTR_BYTE1_7			MAPPING_PNTR_BYTE0_7				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_6							MAPPING_PNTR_BYTE1_6			MAPPING_PNTR_BYTE0_6				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_3 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_7	Mapping pointer #7 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_7	Mapping pointer #7 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_7	Mapping pointer #7 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_6	Mapping pointer #6 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_6	Mapping pointer #6 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_6	Mapping pointer #6 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.338 DC Mapping Configuration Register 4 (IPUx_DC_MAP_CONF_4)

Address: Base address + 5_8118h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_9					MAPPING_PNTR_BYTE1_1					MAPPING_PNTR_BYTE0_9				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_8					MAPPING_PNTR_BYTE1_8					MAPPING_PNTR_BYTE0_8				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_4 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_9	Mapping pointer #9 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_1	Mapping pointer #9 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_9	Mapping pointer #9 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_8	Mapping pointer #8 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_8	Mapping pointer #8 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_8	Mapping pointer #8 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.339 DC Mapping Configuration Register 5 (IPUx_DC_MAP_CONF_5)

Address: Base address + 5_811Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_11							MAPPING_PNTR_BYTE1_11			MAPPING_PNTR_BYTE0_11				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_10							MAPPING_PNTR_BYTE1_10			MAPPING_PNTR_BYTE0_10				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_5 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 11	Mapping pointer #11 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 11	Mapping pointer #11 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 11	Mapping pointer #11 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 10	Mapping pointer #10 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 10	Mapping pointer #10 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 10	Mapping pointer #10 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.340 DC Mapping Configuration Register 6 (IPUx_DC_MAP_CONF_6)

Address: Base address + 5_8120h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_13							MAPPING_PNTR_BYTE1_13			MAPPING_PNTR_BYTE0_13				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_12							MAPPING_PNTR_BYTE1_12			MAPPING_PNTR_BYTE0_12				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_6 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 13	Mapping pointer #13 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 13	Mapping pointer #13 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 13	Mapping pointer #13 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 12	Mapping pointer #12 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 12	Mapping pointer #12 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 12	Mapping pointer #12 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.341 DC Mapping Configuration Register 7 (IPUx_DC_MAP_CONF_7)

Address: Base address + 5_8124h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_15							MAPPING_PNTR_BYTE1_15			MAPPING_PNTR_BYTE0_15				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_14							MAPPING_PNTR_BYTE1_14			MAPPING_PNTR_BYTE0_14				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_7 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 15	Mapping pointer #15 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 15	Mapping pointer #15 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 15	Mapping pointer #15 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 14	Mapping pointer #14 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 14	Mapping pointer #14 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 14	Mapping pointer #14 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.342 DC Mapping Configuration Register 8 (IPUx_DC_MAP_CONF_8)

Address: Base address + 5_8128h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_17							MAPPING_PNTR_BYTE1_17			MAPPING_PNTR_BYTE0_17				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_16							MAPPING_PNTR_BYTE1_16			MAPPING_PNTR_BYTE0_16				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_8 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 17	Mapping pointer #17 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 17	Mapping pointer #17 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 17	Mapping pointer #17 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 16	Mapping pointer #16 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 16	Mapping pointer #16 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 16	Mapping pointer #16 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.343 DC Mapping Configuration Register 9 (IPUx_DC_MAP_CONF_9)

Address: Base address + 5_812Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_19							MAPPING_PNTR_BYTE1_19			MAPPING_PNTR_BYTE0_19				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_18							MAPPING_PNTR_BYTE1_18			MAPPING_PNTR_BYTE0_18				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_9 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 19	Mapping pointer #19 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 19	Mapping pointer #19 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 19	Mapping pointer #19 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 18	Mapping pointer #18 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 18	Mapping pointer #18 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 18	Mapping pointer #18 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.344 DC Mapping Configuration Register 10 (IPUx_DC_MAP_CONF_10)

Address: Base address + 5_8130h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_21							MAPPING_PNTR_BYTE1_21			MAPPING_PNTR_BYTE0_21				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_20							MAPPING_PNTR_BYTE1_20			MAPPING_PNTR_BYTE0_20				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_10 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 21	Mapping pointer #21 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 21	Mapping pointer #21 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 21	Mapping pointer #21 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 20	Mapping pointer #20 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 20	Mapping pointer #20 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 20	Mapping pointer #20 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.345 DC Mapping Configuration Register 11 (IPUx_DC_MAP_CONF_11)

Address: Base address + 5_8134h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_23							MAPPING_PNTR_BYTE1_23			MAPPING_PNTR_BYTE0_23				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_22							MAPPING_PNTR_BYTE1_22			MAPPING_PNTR_BYTE0_22				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_11 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 23	Mapping pointer #23 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 23	Mapping pointer #23 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 23	Mapping pointer #23 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 22	Mapping pointer #22 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 22	Mapping pointer #22 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 22	Mapping pointer #22 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.346 DC Mapping Configuration Register 12 (IPUx_DC_MAP_CONF_12)

Address: Base address + 5_8138h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_25							MAPPING_PNTR_BYTE1_25			MAPPING_PNTR_BYTE0_25				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_24							MAPPING_PNTR_BYTE1_24			MAPPING_PNTR_BYTE0_24				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_12 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 25	Mapping pointer #25 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 25	Mapping pointer #25 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 25	Mapping pointer #25 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 24	Mapping pointer #24 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 24	Mapping pointer #24 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 24	Mapping pointer #24 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.347 DC Mapping Configuration Register 13 (IPUx_DC_MAP_CONF_13)

Address: Base address + 5_813Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_27							MAPPING_PNTR_BYTE1_27			MAPPING_PNTR_BYTE0_27				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_26							MAPPING_PNTR_BYTE1_26			MAPPING_PNTR_BYTE0_26				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_13 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 27	Mapping pointer #27 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 27	Mapping pointer #27 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 27	Mapping pointer #27 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 26	Mapping pointer #26 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 26	Mapping pointer #26 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE0_ 26	Mapping pointer #26 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.348 DC Mapping Configuration Register 14 (IPUx_DC_MAP_CONF_14)

Address: Base address + 5_8140h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0	MAPPING_PNTR_BYTE2_29							MAPPING_PNTR_BYTE1_29			MAPPING_PNTR_BYTE0_29				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	MAPPING_PNTR_BYTE2_28f							MAPPING_PNTR_BYTE1_28			MAPPING_PNTR_BYTE2_28				
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_14 field descriptions

Field	Description
31 Reserved	This read-only field is reserved and always has the value 0.
30–26 MAPPING_ PNTR_BYTE2_ 29	Mapping pointer #29 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
25–21 MAPPING_ PNTR_BYTE1_ 29	Mapping pointer #29 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
20–16 MAPPING_ PNTR_BYTE0_ 29	Mapping pointer #29 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0
15 Reserved	This read-only field is reserved and always has the value 0.
14–10 MAPPING_ PNTR_BYTE2_ 28f	Mapping pointer #28 for Byte 2 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #2
9–5 MAPPING_ PNTR_BYTE1_ 28	Mapping pointer #28 for Byte 1 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #1
MAPPING_ PNTR_BYTE2_ 28	Mapping pointer #28 for Byte 0 This field is a pointer to the set of MD_OFFSET_<i> and MD_MASK_<i> used for mapping byte #0

37.5.349 DC Mapping Configuration Register 15 (IPUx_DC_MAP_CONF_15)

Address: Base address + 5_8144h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_1					MD_MASK_1								0			MD_OFFSET_0					MD_MASK_0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_15 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_1	Mapping unit's offset parameter #1 This field defines the offset parameter #1 within the 24bit word coming from the DC.
23–16 MD_MASK_1	Mapping unit's mask value #1 This field defines the mask value #1 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_0	Mapping unit's offset parameter #0 This field defines the offset parameter #0 within the 24bit word coming from the DC.
MD_MASK_0	Mapping unit's mask value #0 This field defines the mask value #0 within the 8bit word coming from the DC.

37.5.350 DC Mapping Configuration Register 16 (IPUx_DC_MAP_CONF_16)

Address: Base address + 5_8148h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_3					MD_MASK_3								0			MD_OFFSET_2					MD_MASK_0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_16 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_3	Mapping unit's offset parameter #3 This field defines the offset parameter #3 within the 24bit word coming from the DC.

Table continues on the next page...

IPUx_DC_MAP_CONF_16 field descriptions (continued)

Field	Description
23–16 MD_MASK_3	Mapping unit's mask value #3 This field defines the mask value #3 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_2	Mapping unit's offset parameter #2 This field defines the offset parameter #2 within the 24bit word coming from the DC.
MD_MASK_0	Mapping unit's mask value #2 This field defines the mask value #2 within the 8bit word coming from the DC.

37.5.351 DC Mapping Configuration Register 17 (IPUx_DC_MAP_CONF_17)

Address: Base address + 5_814Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_5					MD_MASK_5								0			MD_OFFSET_4					MD_MASK_4							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_17 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_5	Mapping unit's offset parameter #5 This field defines the offset parameter #5 within the 24bit word coming from the DC.
23–16 MD_MASK_5	Mapping unit's mask value #5 This field defines the mask value #5 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_4	Mapping unit's offset parameter #4 This field defines the offset parameter #4 within the 24bit word coming from the DC.
MD_MASK_4	Mapping unit's mask value #4 This field defines the mask value #4 within the 8bit word coming from the DC.

37.5.352 DC Mapping Configuration Register 18 (IPUx_DC_MAP_CONF_18)

Address: Base address + 5_8150h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_7					MD_MASK_7								0			MD_OFFSET_6					MD_MASK_6							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_18 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_7	Mapping unit's offset parameter #7 This field defines the offset parameter #7 within the 24bit word coming from the DC.
23–16 MD_MASK_7	Mapping unit's mask value #7 This field defines the mask value #7 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_6	Mapping unit's offset parameter #6 This field defines the offset parameter #6 within the 24bit word coming from the DC.
MD_MASK_6	Mapping unit's mask value #6 This field defines the mask value #6 within the 8bit word coming from the DC.

37.5.353 DC Mapping Configuration Register 19 (IPUx_DC_MAP_CONF_19)

Address: Base address + 5_8154h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_9					MD_MASK_9								0			MD_OFFSET_8					MD_MASK_8							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_19 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_9	Mapping unit's offset parameter #9 This field defines the offset parameter #9 within the 24bit word coming from the DC.

Table continues on the next page...

IPUx_DC_MAP_CONF_19 field descriptions (continued)

Field	Description
23–16 MD_MASK_9	Mapping unit's mask value #9 This field defines the mask value #9 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_8	Mapping unit's offset parameter #8 This field defines the offset parameter #8 within the 24bit word coming from the DC.
MD_MASK_8	Mapping unit's mask value #8 This field defines the mask value #8 within the 8bit word coming from the DC.

37.5.354 DC Mapping Configuration Register 20 (IPUx_DC_MAP_CONF_20)

Address: Base address + 5_8158h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_11					MD_MASK_11								0			MD_OFFSET_10					MD_MASK_10							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_20 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_11	Mapping unit's offset parameter #11 This field defines the offset parameter #11 within the 24bit word coming from the DC.
23–16 MD_MASK_11	Mapping unit's mask value #11 This field defines the mask value #11 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_10	Mapping unit's offset parameter #10 This field defines the offset parameter #10 within the 24bit word coming from the DC.
MD_MASK_10	Mapping unit's mask value #10 This field defines the mask value #10 within the 8bit word coming from the DC.

37.5.355 DC Mapping Configuration Register 21 (IPUx_DC_MAP_CONF_21)

Address: Base address + 5_815Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_13					MD_MASK_13								0			MD_OFFSET_12					MD_MASK_12							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_21 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_13	Mapping unit's offset parameter #13 This field defines the offset parameter #13 within the 24bit word coming from the DC.
23–16 MD_MASK_13	Mapping unit's mask value #13 This field defines the mask value #13 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_12	Mapping unit's offset parameter #12 This field defines the offset parameter #12 within the 24bit word coming from the DC.
MD_MASK_12	Mapping unit's mask value #12 This field defines the mask value #12 within the 8bit word coming from the DC.

37.5.356 DC Mapping Configuration Register 22 (IPUx_DC_MAP_CONF_22)

Address: Base address + 5_8160h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_15					MD_MASK_15								0			MD_OFFSET_14					MD_MASK_14							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_22 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_15	Mapping unit's offset parameter #15 This field defines the offset parameter #15 within the 24bit word coming from the DC.

Table continues on the next page...

IPUx_DC_MAP_CONF_22 field descriptions (continued)

Field	Description
23–16 MD_MASK_15	Mapping unit's mask value #15 This field defines the mask value #15 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_14	Mapping unit's offset parameter #14 This field defines the offset parameter #14 within the 24bit word coming from the DC.
MD_MASK_14	Mapping unit's mask value #14 This field defines the mask value #14 within the 8bit word coming from the DC.

37.5.357 DC Mapping Configuration Register 23 (IPUx_DC_MAP_CONF_23)

Address: Base address + 5_8164h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_17					MD_MASK_17								0			MD_OFFSET_16					MD_MASK_16							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_23 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_17	Mapping unit's offset parameter #17 This field defines the offset parameter #17 within the 24bit word coming from the DC.
23–16 MD_MASK_17	Mapping unit's mask value #17 This field defines the mask value #17 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_16	Mapping unit's offset parameter #16 This field defines the offset parameter #16 within the 24bit word coming from the DC.
MD_MASK_16	Mapping unit's mask value #16 This field defines the mask value #16 within the 8bit word coming from the DC.

37.5.358 DC Mapping Configuration Register 24 (IPUx_DC_MAP_CONF_24)

Address: Base address + 5_8168h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_19					MD_MASK_19								0			MD_OFFSET_18					MD_MASK_18							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_24 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_19	Mapping unit's offset parameter #19 This field defines the offset parameter #19 within the 24bit word coming from the DC.
23–16 MD_MASK_19	Mapping unit's mask value #19 This field defines the mask value #19 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_18	Mapping unit's offset parameter #18 This field defines the offset parameter #18 within the 24bit word coming from the DC.
MD_MASK_18	Mapping unit's mask value #18 This field defines the mask value #18 within the 8bit word coming from the DC.

37.5.359 DC Mapping Configuration Register 25 (IPUx_DC_MAP_CONF_25)

Address: Base address + 5_816Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_21					MD_MASK_21								0			MD_OFFSET_20					MD_MASK_20							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_25 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_21	Mapping unit's offset parameter #21 This field defines the offset parameter #21 within the 24bit word coming from the DC.

Table continues on the next page...

IPUx_DC_MAP_CONF_25 field descriptions (continued)

Field	Description
23–16 MD_MASK_21	Mapping unit's mask value #21 This field defines the mask value #21 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_20	Mapping unit's offset parameter #20 This field defines the offset parameter #20 within the 24bit word coming from the DC.
MD_MASK_20	Mapping unit's mask value #20 This field defines the mask value #20 within the 8bit word coming from the DC.

37.5.360 DC Mapping Configuration Register 26 (IPUx_DC_MAP_CONF_26)

Address: Base address + 5_8170h offset

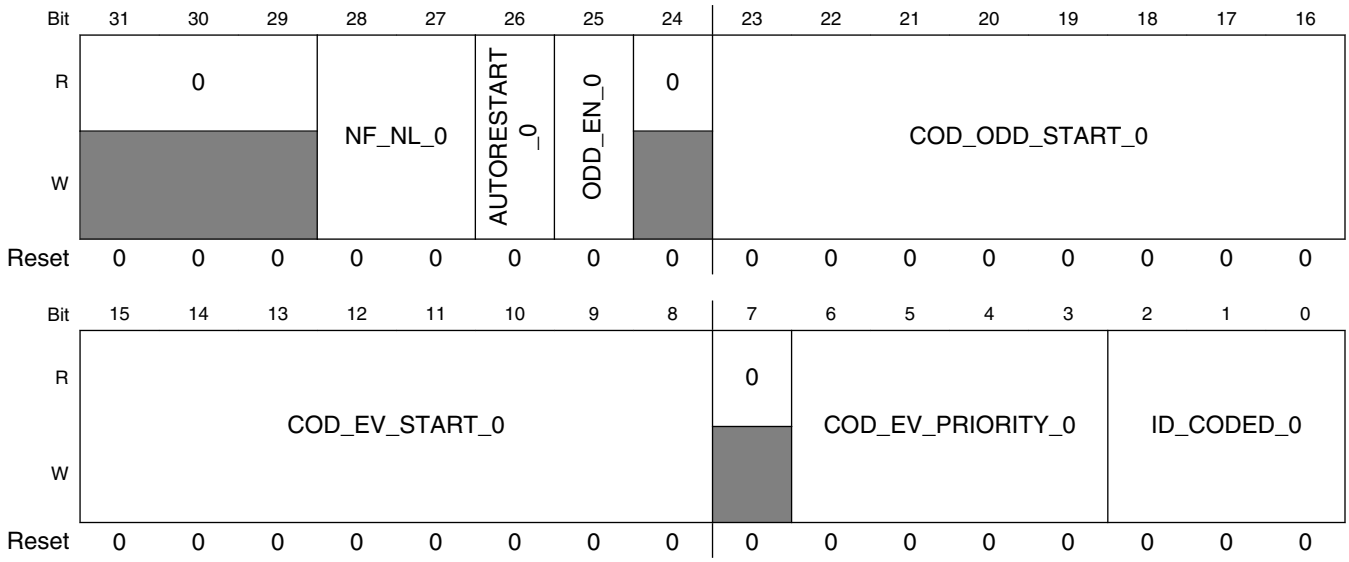
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			MD_OFFSET_23					MD_MASK_23								0			MD_OFFSET_22					MD_MASK_22							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_MAP_CONF_26 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–24 MD_OFFSET_23	Mapping unit's offset parameter #23 This field defines the offset parameter #23 within the 24bit word coming from the DC.
23–16 MD_MASK_23	Mapping unit's mask value #23 This field defines the mask value #23 within the 8bit word coming from the DC.
15–13 Reserved	This read-only field is reserved and always has the value 0.
12–8 MD_OFFSET_22	Mapping unit's offset parameter #22 This field defines the offset parameter #22 within the 24bit word coming from the DC.
MD_MASK_22	Mapping unit's mask value #22 This field defines the mask value #22 within the 8bit word coming from the DC.

37.5.361 DC User General Data Event 0 Register 0 (IPUx_DC_UGDE0_0)

Address: Base address + 5_8174h offset



IPUx_DC_UGDE0_0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–27 NF_NL_0	the user may attach his general event #0 to New-line New-Frame and New-field events. One of these event triggers the user's general event #0's counter. The actual internal trigger is the pixel following the occurrence of the selected event. 00 New Line 01 New Frame 10 New Field 11 Reserved
26 AUTORESTART_0	User's general event #0 auto restart mode 0 disable 1 User's general event #0's counter is automatically restarted.
25 ODD_EN_0	The user's general event #0 may be split into 2 internal signals. One one mode all the events are sent on one signal, on the second mode odd events are sent over one signal while even events are sent over the other signal (ODD_MODE) 1 ODD_MODE is enabled 0 ODD_MODE is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–16 COD_ODD_START_0	This field holds a pointer in the microcode holding the routine to be performed following the user general event #0. When ODD_MODE is enabled, only the odd events will use this pointer

Table continues on the next page...

IPUx_DC_UGDE0_0 field descriptions (continued)

Field	Description
	When ODD_MODE is disabled this field is ignored
15–8 COD_EV_ START_0	This field holds a pointer in the microcode holding the routine to be performed following the user general event #0. When ODD_MODE is enabled, only the even events will use this pointer When ODD_MODE is disabled, all the events will use this pointer
7 Reserved	This read-only field is reserved and always has the value 0.
6–3 COD_EV_ PRIORITY_0	This field defines the priority of the user general event #0 The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
ID_CODED_0	This field defines the number of DC channel number that user's general event #0 will be associated to. 000 DC channel_0 001 DC channel_1 010 DC channel_2 011 DC channel_5 (DP_SYNC) 100 DC channel_6 (DP ASYNC) 101 Reserved 110 Reserved. 111 Reserved.

37.5.362 DC User General Data Event 0 Register 1 (IPUx_DC_UGDE0_1)

Address: Base address + 5_8178h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_0															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE0_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_0	This field holds the pre defined value that the counter counts too.

37.5.363 DC User General Data Event 0 Register2 (IPUx_DC_UGDE0_2)

Address: Base address + 5_817Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																OFFSET_DT_0															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE0_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
OFFSET_DT_0	This field defines the offset value from which the counter of user general event #0 will start counting from

37.5.364 DC User General Data Event 0 Register 3 (IPUx_DC_UGDE0_3)

Address: Base address + 5_8180h offset

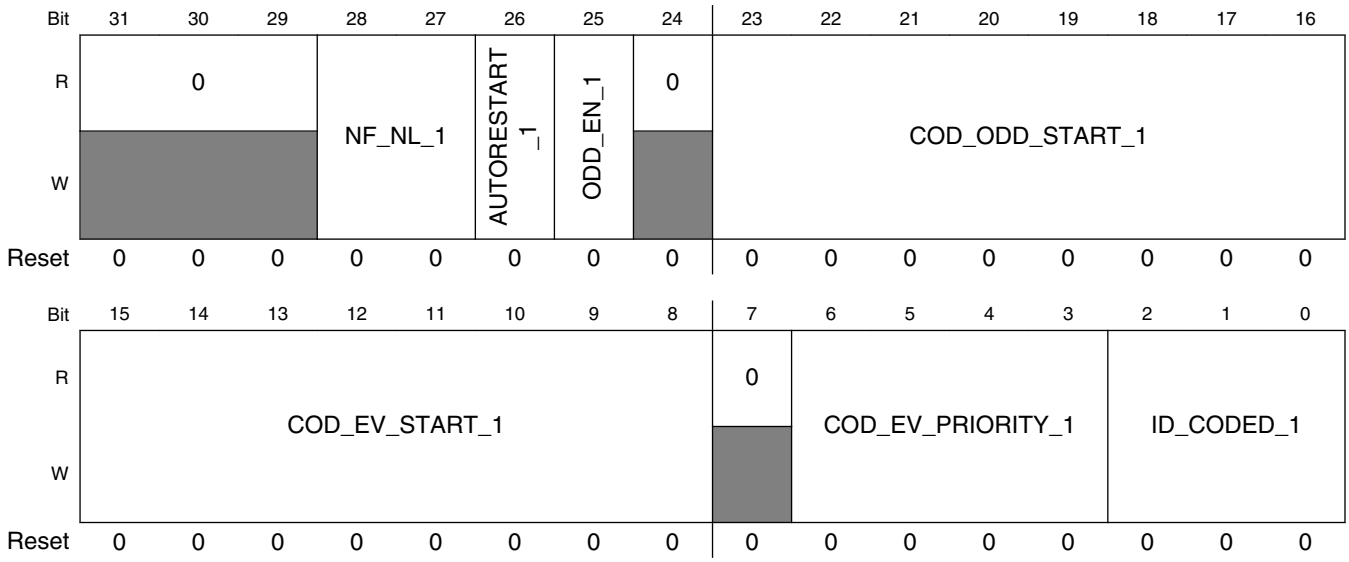
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_REPEAT_0															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE0_3 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_REPEAT_0	When auto reload mode is disabled this field defines the number of events that will be generated by the user general event #0 mechanism

37.5.365 DC User General Data Event 1Register0 (IPUx_DC_UGDE1_0)

Address: Base address + 5_8184h offset



IPUx_DC_UGDE1_0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–27 NF_NL_1	the user may attach his general event #1 to New-line New-Frame and New-field events. One of these event triggers the user's general event #1's counter. The actual internal trigger is the pixel following the occurrence of the selected event. 00 New Line 01 New Frame 10 New Field 11 Reserved
26 AUTORESTART_1	User's general event #1 auto restart mode 0 disable 1 User's general event #1's counter is automatically restarted.
25 ODD_EN_1	The user's general event #1 may be split into 2 internal signals. One one mode all the events are sent on one signal, on the second mode odd events are sent over one signal while even events are sent over the other signal (ODD_MODE) 1 ODD_MODE is enabled 0 ODD_MODE is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–16 COD_ODD_START_1	This field holds a pointer in the microcode holding the routine to be performed following the user general event #1. When ODD_MODE is enabled, only the odd events will use this pointer

Table continues on the next page...

IPUx_DC_UGDE1_0 field descriptions (continued)

Field	Description
	When ODD_MODE is disabled this field is ignored
15–8 COD_EV_ START_1	This field holds a pointer in the microcode holding the routine to be performed following the user general event #1. When ODD_MODE is enabled, only the even events will use this pointer When ODD_MODE is disabled, all the events will use this pointer
7 Reserved	This read-only field is reserved and always has the value 0.
6–3 COD_EV_ PRIORITY_1	This field defines the priority of the user general event #1 The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
ID_CODED_1	This field defines the number of DC channel number that user's general event #1 will be associated to 000 DC channel_0 001 DC channel_1 010 DC channel_2 011 DC channel_5 (DP_SYNC) 100 DC channel_6 (DP ASYNC) 101 Reserved 110 Reserved 111 Reserved

37.5.366 DC User General Data Event 1 Register 1 (IPUx_DC_UGDE1_1)

Address: Base address + 5_8188h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_1																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE1_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_1	This field hold the pre defined value that the counter counts too

37.5.367 DC User General Data Event 1Register 2 (IPUx_DC_UGDE1_2)

Address: Base address + 5_818Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																OFFSET_DT_1																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE1_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
OFFSET_DT_1	This field defines the offset value from which the counter of user general event #1 will start counting from

37.5.368 DC User General Data Event 1Register 3 (IPUx_DC_UGDE1_3)

Address: Base address + 5_8190h offset

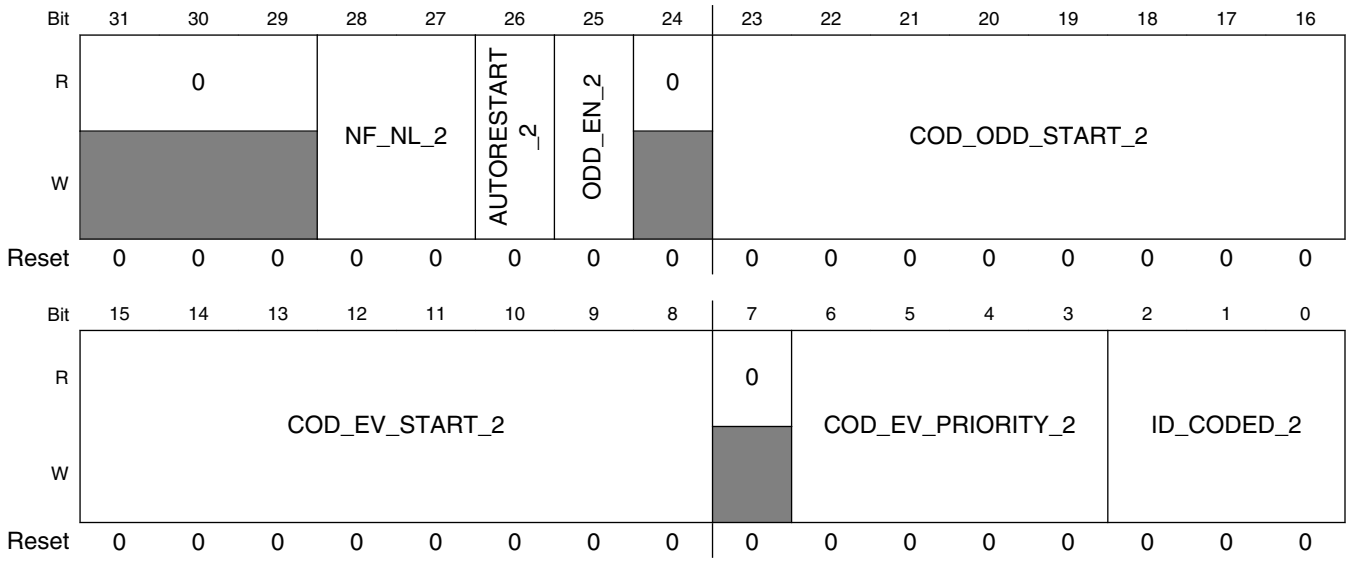
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_REPEAT_1																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE1_3 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_REPEAT_1	When auto reload mode is disabled this field defines the number of events that will be generated by the user general event #1 mechanism

37.5.369 DC User General Data Event 2 Register 0 (IPUx_DC_UGDE2_0)

Address: Base address + 5_8194h offset



IPUx_DC_UGDE2_0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–27 NF_NL_2	the user may attach his general event #2 to New-line New-Frame and New-field events. One of these event triggers the user's general event #2's counter. The actual internal trigger is the pixel following the occurrence of the selected event. 00 New Line 01 New Frame 10 New Field 11 Reserved
26 AUTORESTART_2	User's general event #2 auto restart mode 0 disable 1 User's general event #2's counter is automatically restarted.
25 ODD_EN_2	The user's general event #2 may be split into 2 internal signals. One one mode all the events are sent on one signal, on the second mode odd events are sent over one signal while even events are sent over the other signal (ODD_MODE) 1 ODD_MODE is enabled 0 ODD_MODE is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–16 COD_ODD_START_2	This field holds a pointer in the microcode holding the routine to be performed following the user general event #2 When ODD_MODE is enabled, only the odd events will use this pointer

Table continues on the next page...

IPUx_DC_UGDE2_0 field descriptions (continued)

Field	Description
	When ODD_MODE is disabled this field is ignored
15–8 COD_EV_ START_2	This field holds a pointer in the microcode holding the routine to be performed following the user general event #2. When ODD_MODE is enabled, only the even events will use this pointer When ODD_MODE is disabled, all the events will use this pointer
7 Reserved	This read-only field is reserved and always has the value 0.
6–3 COD_EV_ PRIORITY_2	This field defines the priority of the user general event #2 The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0000 disable 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
ID_CODED_2	This field defines the number of DC channel number that user's general event #2 will be associated to 000 DC channel_0 001 DC channel_1 010 DC channel_2 011 DC channel_5 (DP_SYNC) 100 DC channel_6 (DP ASYNC) 101 Reserved 110 Reserved 111 Reserved

37.5.370 DC User General Data Event 2 Register 1 (IPUx_DC_UGDE2_1)

Address: Base address + 5_8198h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_2															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE2_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_2	This field hold the pre defined value that the counter counts too

37.5.371 DC User General Data Event 2Register 2 (IPUx_DC_UGDE2_2)

Address: Base address + 5_819Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																OFFSET_DT_2																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE2_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
OFFSET_DT_2	This field defines the offset value from which the counter of user general event #2 will start counting from

37.5.372 DC User General Data Event 2Register 3 (IPUx_DC_UGDE2_3)

Address: Base address + 5_81A0h offset

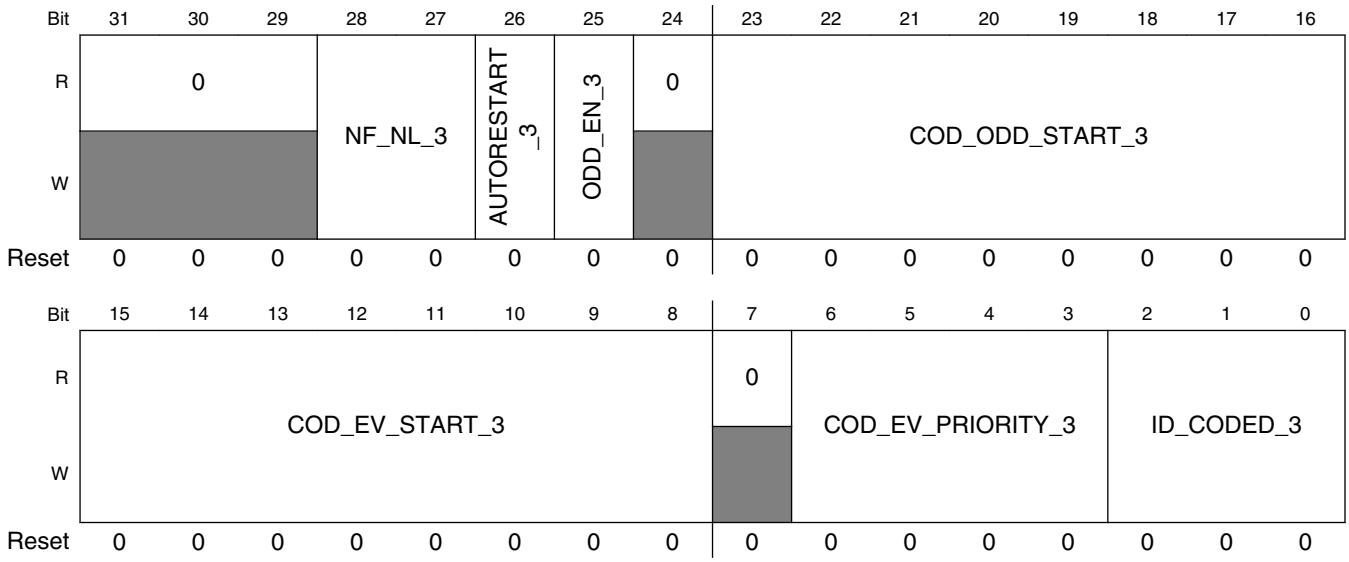
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_REPEAT_2																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE2_3 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_REPEAT_2	When auto reload mode is disabled this field defines the number of events that will be generated by the user general event #2 mechanism

37.5.373 DC User General Data Event 3Register 0 (IPUx_DC_UGDE3_0)

Address: Base address + 5_81A4h offset



IPUx_DC_UGDE3_0 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–27 NF_NL_3	the user may attach his general event #3 to New-line New-Frame and New-field events. One of these event triggers the user's general event #3's counter. The actual internal trigger is the pixel following the occurrence of the selected event. 00 New Line 01 New Frame 10 New Field 11 Reserved
26 AUTORESTART_3	User's general event #3 auto restart mode 0 disable 1 User's general event #3's counter is automatically restarted.
25 ODD_EN_3	The user's general event #3 may be split into 2 internal signals. One one mode all the events are sent on one signal, on the second mode odd events are sent over one signal while even events are sent over the other signal (ODD_MODE) 1 ODD_MODE is enabled 0 ODD_MODE is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–16 COD_ODD_START_3	This field holds a pointer in the microcode holding the routine to be performed following the user general event #3. When ODD_MODE is enabled, only the odd events will use this pointer

Table continues on the next page...

IPUx_DC_UGDE3_0 field descriptions (continued)

Field	Description
	When ODD_MODE is disabled this field is ignored
15–8 COD_EV_ START_3	This field holds a pointer in the microcode holding the routine to be performed following the user general event #3. When ODD_MODE is enabled, only the even events will use this pointer When ODD_MODE is disabled, all the events will use this pointer
7 Reserved	This read-only field is reserved and always has the value 0.
6–3 COD_EV_ PRIORITY_3	This field defines the priority of the user general event #3 0000 disable The priority between the events should be set to a unique value. i.e. two events must not have the same priority (except 0000 - disable) 0001 Priority #1 (lowest) 0010 Priority #2 1101 Priority #13 (highest) 1110 Reserved 1111 Reserved
ID_CODED_3	This field defines the number of DC channel number that user's general event #3 will be associated to 000 DC channel_0 001 DC channel_1 010 DC channel_2 011 DC channel_5 (DP_SYNC) 100 DC channel_6 (DP ASYNC) 101 Reserved 110 Reserved 111 Reserved.

37.5.374 DC User General Data Event 3Register 1 (IPUx_DC_UGDE3_1)

Address: Base address + 5_81A8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																STEP_3															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE3_1 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_3	This field hold the pre defined value that the counter counts too

37.5.375 DC User General Data Event 3Register 2 (IPUx_DC_UGDE3_2)

Address: Base address + 5_81ACh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			OFFSET_DT_3																												
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE3_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
OFFSET_DT_3	This field defines the offset value from which the counter of user general event #3 will start counting from

37.5.376 DC User General Data Event 3Register 2 (IPUx_DC_UGDE3_3)

Address: Base address + 5_81B0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			STEP_REPEAT_3																												
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_UGDE3_3 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
STEP_REPEAT_3	When auto reload mode is disabled this field defines the number of events that will be generated by the user general event #3 mechanism

37.5.377 DC Low Level Access Control Register 0 (IPUx_DC_LLA0)

Address: Base address + 5_81B4h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	MCU_RS_3_0								MCU_RS_2_0								MCU_RS_1_0								MCU_RS_0_0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_LLA0 field descriptions

Field	Description
31–24 MCU_RS_3_0	This field holds a pointer in the microcode handling the RS_3 routine for the display defined at DISP_ID_8, when in Low level access mode,
23–16 MCU_RS_2_0	This field holds a pointer in the microcode handling the RS_2 routine for the display defined at DISP_ID_8, when in Low level access mode,
15–8 MCU_RS_1_0	This field holds a pointer in the microcode handling the RS_1 routine for the display defined at DISP_ID_8, when in Low level access mode,
MCU_RS_0_0	This field holds a pointer in the microcode handling the RS_0 routine for the display defined at DISP_ID_8, when in Low level access mode,

37.5.378 DC Low Level Access Control Register 1 (IPUx_DC_LLA1)

Address: Base address + 5_81B8h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MCU_RS_3_1								MCU_RS_2_1								MCU_RS_1_1								MCU_RS_0_1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_LLA1 field descriptions

Field	Description
31–24 MCU_RS_3_1	This field holds a pointer in the microcode handling the RS_3 routine for the display defined at DISP_ID_9, when in Low level access mode,
23–16 MCU_RS_2_1	This field holds a pointer in the microcode handling the RS_2 routine for the display defined at DISP_ID_9, when in Low level access mode,
15–8 MCU_RS_1_1	This field holds a pointer in the microcode handling the RS_1 routine for the display defined at DISP_ID_9, when in Low level access mode,
MCU_RS_0_1	This field holds a pointer in the microcode handling the RS_0 routine for the display defined at DISP_ID_9, when in Low level access mode,

37.5.379 DC Read Low Level Read Access Control Register 0 (IPUx_DC_R_LLA0)

Address: Base address + 5_81BCh offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MCU_RS_3_0								MCU_RS_2_0								MCU_RS_R_1_0								MCU_RS_R_0_0							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_R_LLA0 field descriptions

Field	Description
31–24 MCU_RS_3_0	This field holds a pointer in the microcode handling the RS_3 routine for the display defined at DISP_ID_8, when in Read Low level access mode,
23–16 MCU_RS_2_0	This field holds a pointer in the microcode handling the RS_2 routine for the display defined at DISP_ID_8, when in Read Low level access mode,
15–8 MCU_RS_R_1_0	This field holds a pointer in the microcode handling the RS_1 routine for the display defined at DISP_ID_8, when in Read Low level access mode,
MCU_RS_R_0_0	This field holds a pointer in the microcode handling the RS_0 routine for the display defined at DISP_ID_8, when in Read Low level access mode,

37.5.380 DC Read Low Level Read Access Control Register1 (IPUx_DC_R_LLA1)

Address: Base address + 5_81C0h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MCU_RS_R_3_1								MCU_RS_R_2_1								MCU_RS_R_1_1								MCU_RS_R_0_1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_R_LLA1 field descriptions

Field	Description
31–24 MCU_RS_R_3_1	This field holds a pointer in the microcode handling the RS_3 routine for the display defined at DISP_ID_9, when in Read Low level access mode,
23–16 MCU_RS_R_2_1	This field holds a pointer in the microcode handling the RS_2 routine for the display defined at DISP_ID_9, when in Read Low level access mode,
15–8 MCU_RS_R_1_1	This field holds a pointer in the microcode handling the RS_1 routine for the display defined at DISP_ID_9, when in Read Low level access mode,
MCU_RS_R_0_1	This field holds a pointer in the microcode handling the RS_0 routine for the display defined at DISP_ID_9, when in Read Low level access mode,

37.5.381 DC Write Channel 5 Configuration Register (IPUx_DC_WR_CH_ADDR_5_ALT)

Address: Base address + 5_81C4h offset

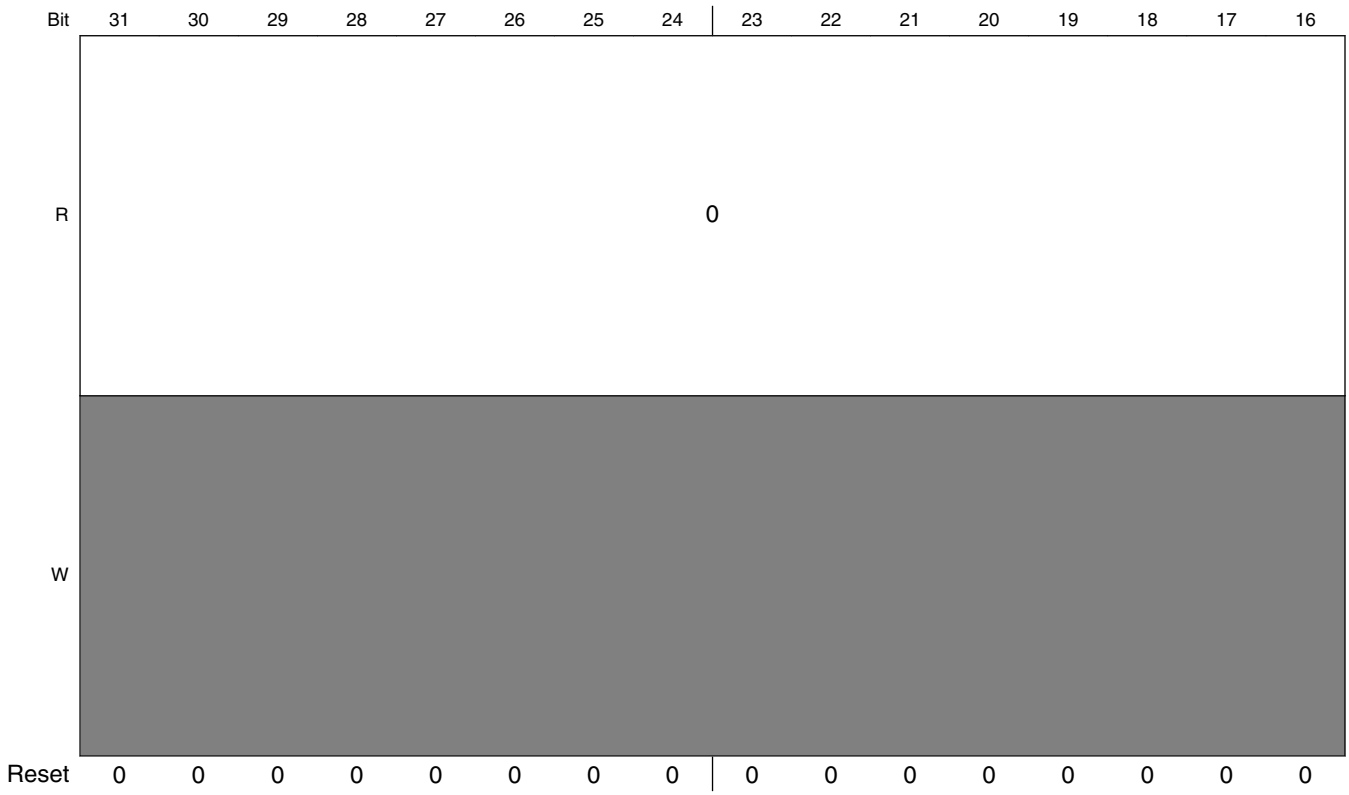
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0																ST_ADDR_5_ALT															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DC_WR_CH_ADDR_5_ALT field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
ST_ADDR_5_ ALT	This field defines the start address within the display's memory space where the write transactions will be done to for channel #5, when alternate flow is performed via channel #5

37.5.382 DC Status Register (IPUx_DC_STAT)

Address: Base address + 5_81C8h offset



Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								DC_TRIPLE_BUF_DATA_EMPTY_1	DC_TRIPLE_BUF_DATA_FULL_1	DC_TRIPLE_BUF_CNT_EMPTY_1	DC_TRIPLE_BUF_CNT_FULL_1	DC_TRIPLE_BUF_DATA_EMPTY_0	DC_TRIPLE_BUF_DATA_FULL_0	DC_TRIPLE_BUF_CNT_EMPTY_0	DC_TRIPLE_BUF_CNT_FULL_0
W																
Reset	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0

IPUx_DC_STAT field descriptions

Field	Description
31–8 Reserved	This read-only field is reserved and always has the value 0.
7 DC_TRIPLE_BUF_DATA_EMPTY_1	This bit indicates a FIFO full state on the DC FIFO accessing DI1 when write to the display flow is used
6 DC_TRIPLE_BUF_DATA_FULL_1	This bit indicates a FIFO empty state on the DC FIFO accessing DI1 when read from the display flow is used

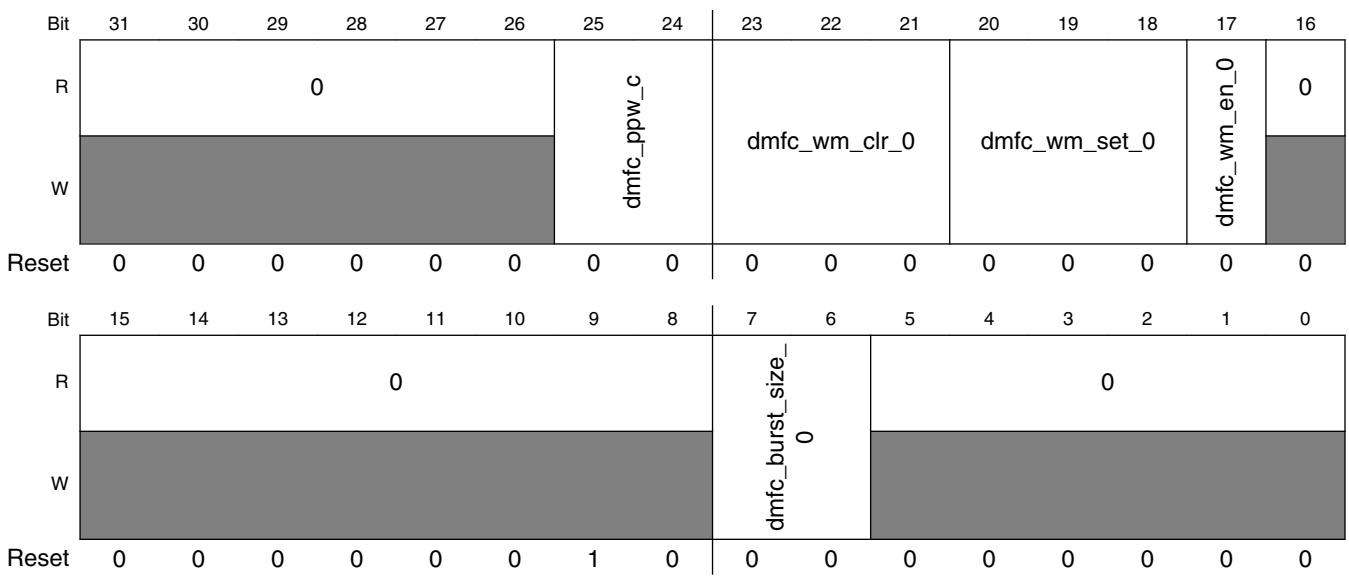
Table continues on the next page...

IPUx_DC_STAT field descriptions (continued)

Field	Description
5 DC_TRIPLE_BUF_CNT_EMPTY_1	This bit indicates a FIFO empty state on the DC FIFO accessing DI1 when write to the display flow is used
4 DC_TRIPLE_BUF_CNT_FULL_1	This bit indicates a FIFO full state on the DC FIFO accessing DI1 when write to the display flow is used
3 DC_TRIPLE_BUF_DATA_EMPTY_0	This bit indicates a FIFO empty state on the DC FIFO accessing DI0 when read from the display flow is used
2 DC_TRIPLE_BUF_DATA_FULL_0	This bit indicates a FIFO full state on the DC FIFO accessing DI0 when read from the display flow is used
1 DC_TRIPLE_BUF_CNT_EMPTY_0	This bit indicates a FIFO empty state on the DC FIFO accessing DI0 when write to the display flow is used
0 DC_TRIPLE_BUF_CNT_FULL_0	This bit indicates a FIFO full state on the DC FIFO accessing DI0 when write to the display flow is used

37.5.383 DMFC Read Channel Register (IPUx_DMFC_RD_CHAN)

Address: Base address + 6_0000h offset



IPUx_DMFC_RD_CHAN field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25–24 dmfc_ppw_c	Pixel Per Word coded. This field defines the size of the read data from the display. 00 8 bit per pixel 01 16 bit per pixel 10 24 (rgb) bit per pixel or 32 bit per pixel 11 Reserved
23–21 dmfc_wm_clr_0	Watermark Clear This field defines the watermark's level of the DMFC read FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of free bursts at the FIFO (dmfc_wm_clr_0 > dmfc_wm_set_0)
20–18 dmfc_wm_set_0	Watermark Set This field defines the watermark's level of the DMFC read FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of free bursts at the FIFO (dmfc_wm_clr_0 > dmfc_wm_set_0)
17 dmfc_wm_en_0	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
16–8 Reserved	This read-only field is reserved and always has the value 0.
7–6 dmfc_burst_size_0	Read burst Size This field defines the burst size of the DMFC's read accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, going to the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
Reserved	This read-only field is reserved and always has the value 0.

37.5.384 DMFC Write Channel Register (IPUx_DMFC_WR_CHAN)

Address: Base address + 6_0004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	dmfc_burst_size_2c		dmfc_fifo_size_2c			dmfc_st_addr_2c			dmfc_burst_size_1c		dmfc_fifo_size_1c			dmfc_st_addr_1c		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	dmfc_burst_size_2		dmfc_fifo_size_2			dmfc_st_addr_2			dmfc_burst_size_1		dmfc_fifo_size_1			dmfc_st_addr_1		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DMFC_WR_CHAN field descriptions

Field	Description
31–30 dmfc_burst_size_2c	Burst size of IDMAC's channel 43 This field defines the burst size of the IDMAC's channel 43 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
29–27 dmfc_fifo_size_2c	DMFC FIFO size for IDMAC's channel 43 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 43 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
26–24 dmfc_st_addr_2c	DMFC Start Address for IDMAC's channel 43 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 43. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
23–22 dmfc_burst_size_1c	Burst size of IDMAC's channel 42 This field defines the burst size of the IDMAC's channel 42 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings.

Table continues on the next page...

IPUx_DMFC_WR_CHAN field descriptions (continued)

Field	Description
	00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
21–19 dmfc_fifo_size_1c	DMFC FIFO size for IDMAC's channel 42 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 42 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
18–16 dmfc_st_addr_1c	DMFC Start Address for IDMAC's channel 42 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 42. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
15–14 dmfc_burst_size_2	Burst size of IDMAC's channel 41 This field defines the burst size of the IDMAC's channel 41 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
13–11 dmfc_fifo_size_2	DMFC FIFO size for IDMAC's channel 41 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 41 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
10–8 dmfc_st_addr_2	DMFC Start Address for IDMAC's channel 41 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 41. The FIFO is partitioned to 8 equal segments.

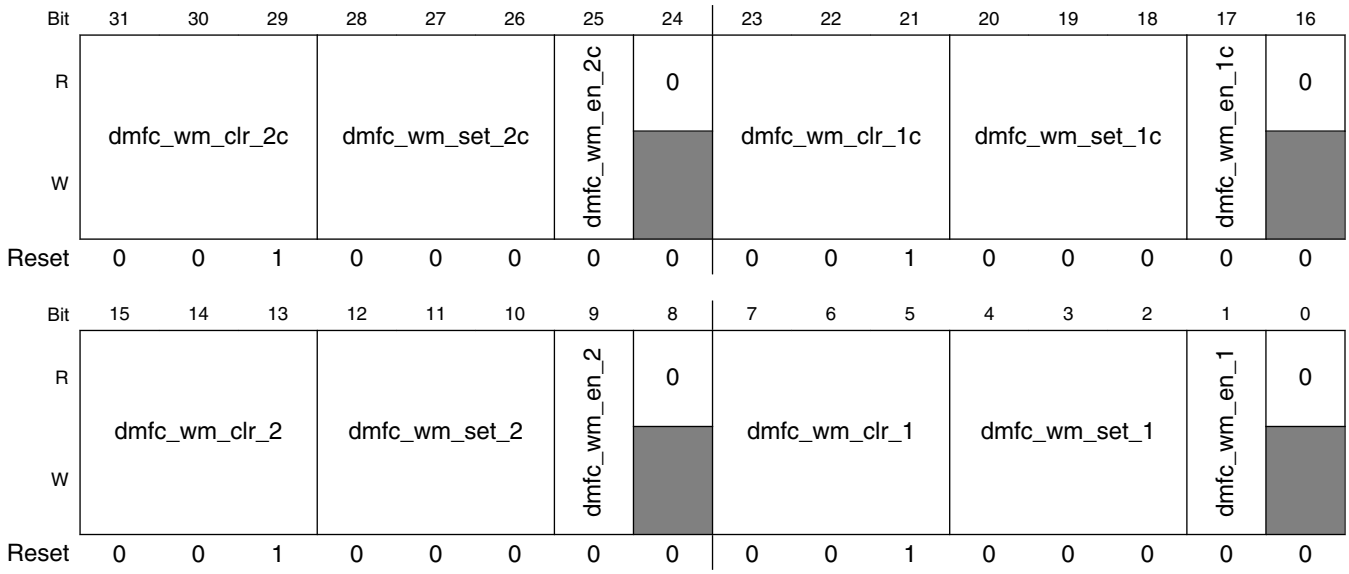
Table continues on the next page...

IPUx_DMFC_WR_CHAN field descriptions (continued)

Field	Description
	Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
7–6 dmfc_burst_size_1	Burst size of IDMAC's channel 28 This field defines the burst size of the IDMAC's channel 28 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 1hbit
5–3 dmfc_fifo_size_1	DMFC FIFO size for IDMAC's channel 28 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 28 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
dmfc_st_addr_1	DMFC Start Address for IDMAC's channel 28 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 28. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7

37.5.385 DMFC Write Channel Definition Register (IPUx_DMFC_WR_CHAN_DEF)

Address: Base address + 6_0008h offset



IPUx_DMFC_WR_CHAN_DEF field descriptions

Field	Description
31–29 dmfc_wm_clr_2c	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
28–26 dmfc_wm_set_2c	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
25 dmfc_wm_en_2c	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–21 dmfc_wm_clr_1c	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
20–18 dmfc_wm_set_1c	Watermark Set

Table continues on the next page...

IPUx_DMFC_WR_CHAN_DEF field descriptions (continued)

Field	Description
	This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
17 dmfc_wm_en_1c	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
16 Reserved	This read-only field is reserved and always has the value 0.
15–13 dmfc_wm_clr_2	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
12–10 dmfc_wm_set_2	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
9 dmfc_wm_en_2	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
8 Reserved	This read-only field is reserved and always has the value 0.
7–5 dmfc_wm_clr_1	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
4–2 dmfc_wm_set_1	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
1 dmfc_wm_en_1	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.386 DMFC Display Processor Channel Register (IPUx_DMFC_DP_CHAN)

Address: Base address + 6_000Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	dmfc_burst_size_6f		dmfc_fifo_size_6f			dmfc_st_addr_6f			dmfc_burst_size_6b		dmfc_fifo_size_6b			dmfc_st_addr_6b		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	dmfc_burst_size_5f		dmfc_fifo_size_5f			dmfc_st_addr_5f			dmfc_burst_size_5b		dmfc_fifo_size_5b			dmfc_st_addr_5b		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DMFC_DP_CHAN field descriptions

Field	Description
31–30 dmfc_burst_size_6f	Burst size of IDMAC's channel 29 This field defines the burst size of the IDMAC's channel 29 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
29–27 dmfc_fifo_size_6f	DMFC FIFO size for IDMAC's channel 29 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 29 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
26–24 dmfc_st_addr_6f	DMFC Start Address for IDMAC's channel 29 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 29. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7

Table continues on the next page...

IPUx_DMFC_DP_CHAN field descriptions (continued)

Field	Description
23–22 dmfc_burst_size_6b	Burst size of IDMAC's channel 24 This field defines the burst size of the IDMAC's channel 24 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
21–19 dmfc_fifo_size_6b	DMFC FIFO size for IDMAC's channel 24 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 24 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
18–16 dmfc_st_addr_6b	DMFC Start Address for IDMAC's channel 24 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 24. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
15–14 dmfc_burst_size_5f	Burst size of IDMAC's channel 27 This field defines the burst size of the IDMAC's channel 27 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
13–11 dmfc_fifo_size_5f	DMFC FIFO size for IDMAC's channel 27 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 27 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel

Table continues on the next page...

IPUx_DMFC_DP_CHAN field descriptions (continued)

Field	Description
10–8 dmfc_st_addr_5f	DMFC Start Address for IDMAC's channel 27 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 27. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
7–6 dmfc_burst_size_5b	Burst size of IDMAC's channel 23 This field defines the burst size of the IDMAC's channel 23 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
5–3 dmfc_fifo_size_5b	DMFC FIFO size for IDMAC's channel 23 This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 23 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
dmfc_st_addr_5b	DMFC Start Address for IDMAC's channel 23 This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 23. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7

37.5.387 DMFC Display Processor Channel Definition Register (IPUx_DMFC_DP_CHAN_DEF)

Address: Base address + 6_0010h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	dmfc_wm_clr_6f			dmfc_wm_set_6f			dmfc_wm_en_6f	0	dmfc_wm_clr_6b			dmfc_wm_set_6b			dmfc_wm_en_6b	0
W																
Reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	dmfc_wm_clr_5f			dmfc_wm_set_5f			dmfc_wm_en_5f	0	dmfc_wm_clr_5b			dmfc_wm_set_5b			dmfc_wm_en_5b	0
W																
Reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

IPUx_DMFC_DP_CHAN_DEF field descriptions

Field	Description
31–29 dmfc_wm_clr_6f	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
28–26 dmfc_wm_set_6f	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
25 dmfc_wm_en_6f	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–21 dmfc_wm_clr_6b	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
20–18 dmfc_wm_set_6b	Watermark Set

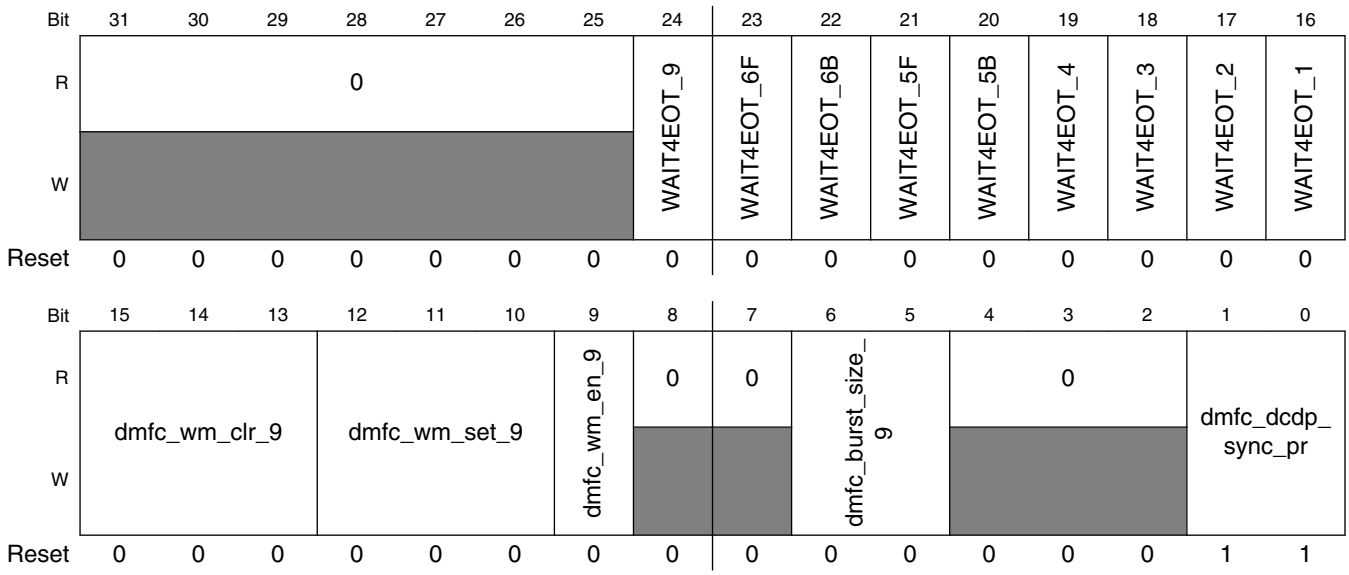
Table continues on the next page...

IPUx_DMFC_DP_CHAN_DEF field descriptions (continued)

Field	Description
	This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
17 dmfc_wm_en_6b	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
16 Reserved	This read-only field is reserved and always has the value 0.
15–13 dmfc_wm_clr_5f	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
12–10 dmfc_wm_set_5f	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
9 dmfc_wm_en_5f	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
8 Reserved	This read-only field is reserved and always has the value 0.
7–5 dmfc_wm_clr_5b	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
4–2 dmfc_wm_set_5b	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
1 dmfc_wm_en_5b	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.388 DMFC General 1 Register (IPUx_DMFC_GENERAL_1)

Address: Base address + 6_0014h offset



IPUx_DMFC_GENERAL_1 field descriptions

Field	Description
31–25 Reserved	This read-only field is reserved and always has the value 0.
24 WAIT4EOT_9	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #9 is in wait4eot mode 0 FIFO #9 is in normal mode
23 WAIT4EOT_6F	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #6F is in wait4eot mode 0 FIFO #6F is in normal mode
22 WAIT4EOT_6B	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #6B is in wait4eot mode 0 FIFO #6B is in normal mode

Table continues on the next page...

IPUx_DMFC_GENERAL_1 field descriptions (continued)

Field	Description
21 WAIT4EOT_5F	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #5F is in wait4eot mode 0 FIFO #5F is in normal mode
20 WAIT4EOT_5B	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #5B is in wait4eot mode 0 FIFO #5B is in normal mode
19 WAIT4EOT_4	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #4 is in wait4eot mode 0 FIFO #4 is in normal mode
18 WAIT4EOT_3	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #3 is in wait4eot mode 0 FIFO #3 is in normal mode
17 WAIT4EOT_2	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #2 is in wait4eot mode 0 FIFO #2 is in normal mode
16 WAIT4EOT_1	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #1 is in wait4eot mode 0 FIFO #1 is in normal mode
15–13 dmfc_wm_clr_9	Watermark Clear This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)

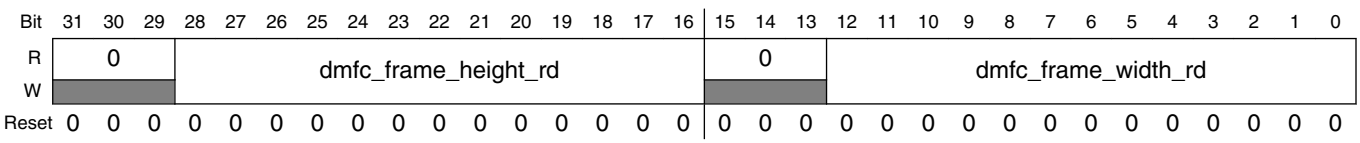
Table continues on the next page...

IPUx_DMFC_GENERAL_1 field descriptions (continued)

Field	Description
12–10 dmfc_wm_set_9	Watermark Set This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
9 dmfc_wm_en_9	Watermark enable. This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
8 Reserved	This read-only field is reserved and always has the value 0.
7 Reserved	This read-only field is reserved and always has the value 0.
6–5 dmfc_burst_size_9	Burst size of IDMAC's channel 44 This field defines the burst size of the IDMAC's channel 44 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. This channel is targeted for MASK - the FIFO size is always 32X128; The base address is always the upper half of the 8th segment 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
4–2 Reserved	This read-only field is reserved and always has the value 0.
dmfc_dcdp_sync_pr	DMFC's memory access priority settings for simultaneous synchronous flows from DC & DP 00 Forbidden - should not be used. 01 DC has higher priority over DP 10 DP has higher priority over DC 11 Round Robin

37.5.389 DMFC General 2 Register (IPUx_DMFC_GENERAL_2)

Address: Base address + 6_0018h offset



IPUx_DMFC_GENERAL_2 field descriptions

Field	Description
31–29 Reserved	This read-only field is reserved and always has the value 0.
28–16 dmfc_frame_ height_rd	Frame height for read channel from the display to the IDMAC; Units are pixels
15–13 Reserved	This read-only field is reserved and always has the value 0.
dmfc_frame_ width_rd	Frame width for read channel from the display to the IDMAC; Units are pixels

37.5.390 DMFC IC Interface Control Register (IPUx_DMFC_IC_CTRL)

Address: Base address + 6_001Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	dmfc_ic_frame_height_rd													dmfc_ic_frame_ width_rd		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	dmfc_ic_frame_width_rd										dmfc_ic_ ppw_c		0	dmfc_ic_in_port		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

IPUx_DMFC_IC_CTRL field descriptions

Field	Description
31–19 dmfc_ic_frame_ height_rd	Frame's height for the channel coming from IC. Units are lines
18–6 dmfc_ic_frame_ width_rd	Frame's width for the channel coming from IC. Units are pixels
5–4 dmfc_ic_ppw_c	Pixel Per Word coded from IC. This field defines the size of the data coming from the IC. 00 8 bit per pixel 01 16 bit per pixel 10 24 bit per pixel 11 Reserved
3 Reserved	This read-only field is reserved and always has the value 0.
dmfc_ic_in_port	DMFC input port

Table continues on the next page...

IPUx_DMFC_IC_CTRL field descriptions (continued)

Field	Description
	When data is coming from the IC, the IC channel replaces one of the IDMAC's channels connected to the DMFC. This field defines which IDMAC's channel is replaced by the IC channel. 000 CH28 001 CH41 010 Reserved, IC channel is disabled 011 Reserved, IC channel is disabled 100 CH23 101 CH27 110 CH24 111 CH29

37.5.391 DMFC Write Channel Alternate Register (IPUx_DMFC_WR_CHAN_ALT)

Address: Base address + 6_0020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	dmfc_burst_size_2_alt		dmfc_fifo_size_2_alt			dmfc_st_addr_2_alt			0							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_DMFC_WR_CHAN_ALT field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.
15–14 dmfc_burst_size_2_alt	Burst size of IDMAC's channel 41 (for alternate flow) This field defines the burst size of the IDMAC's channel 41 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
13–11 dmfc_fifo_size_2_alt	DMFC FIFO size for IDMAC's channel 41 (for alternate flow) This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 41 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel

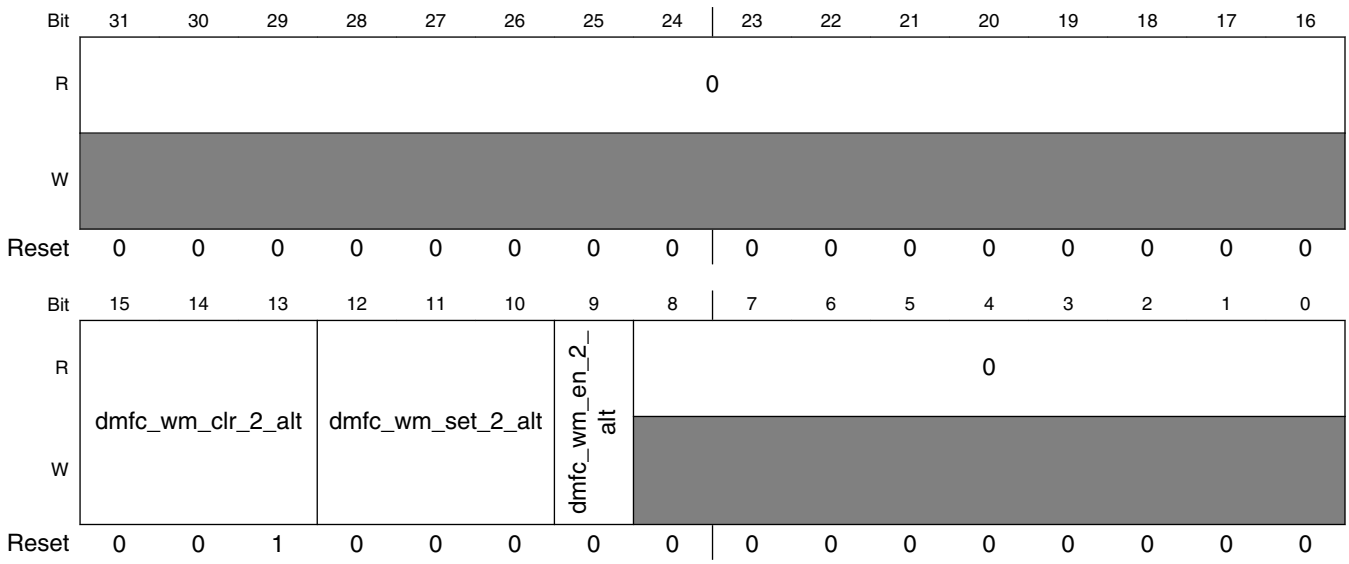
Table continues on the next page...

IPUx_DMFC_WR_CHAN_ALT field descriptions (continued)

Field	Description
	011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
10–8 dmfc_st_addr_2_alt	DMFC Start Address for IDMAC's channel 41 (for alternate flow) This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 41. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
Reserved	This read-only field is reserved and always has the value 0.

37.5.392 DMFC Write Channel Definition Alternate Register (IPUx_DMFC_WR_CHAN_DEF_ALT)

Address: Base address + 6_0024h offset



IPUx_DMFC_WR_CHAN_DEF_ALT field descriptions

Field	Description
31–16 Reserved	This read-only field is reserved and always has the value 0.

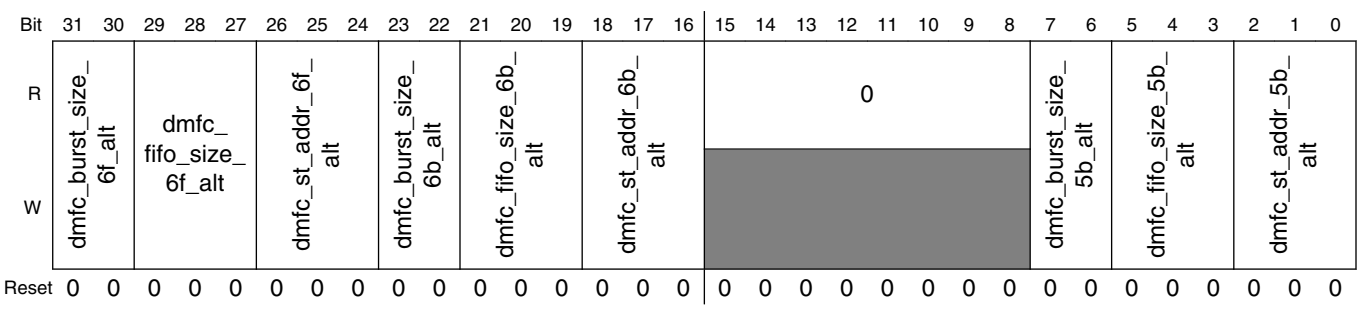
Table continues on the next page...

IPUx_DMFC_WR_CHAN_DEF_ALT field descriptions (continued)

Field	Description
15–13 dmfc_wm_clr_2_alt	Watermark Clear (for alternate flow) This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
12–10 dmfc_wm_set_2_alt	Watermark Set (for alternate flow) This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
9 dmfc_wm_en_2_alt	Watermark enable. (for alternate flow) This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
Reserved	This read-only field is reserved and always has the value 0.

37.5.393 DMFC MFC Display Processor Channel Alternate Register (IPUx_DMFC_DP_CHAN_ALT)

Address: Base address + 6_0028h offset



IPUx_DMFC_DP_CHAN_ALT field descriptions

Field	Description
31–30 dmfc_burst_size_6f_alt	Burst size of IDMAC's channel 29 (for alternate flow) This field defines the burst size of the IDMAC's channel 29 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
29–27 dmfc_fifo_size_6f_alt	DMFC FIFO size for IDMAC's channel 29 (for alternate flow) This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 29 000 All (512X128 words) the DMFC's FIFO is allocated to this channel

Table continues on the next page...

IPUx_DMFC_DP_CHAN_ALT field descriptions (continued)

Field	Description
	001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
26–24 dmfc_st_addr_6f_alt	DMFC Start Address for IDMAC's channel 29 (for alternate flow) This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 29. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
23–22 dmfc_burst_size_6b_alt	Burst size of IDMAC's channel 24 (for alternate flow) This field defines the burst size of the IDMAC's channel 24 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
21–19 dmfc_fifo_size_6b_alt	DMFC FIFO size for IDMAC's channel 24 (for alternate flow) This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 24 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
18–16 dmfc_st_addr_6b_alt	DMFC Start Address for IDMAC's channel 24 (for alternate flow) This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 24. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7
15–8 Reserved	This read-only field is reserved and always has the value 0.

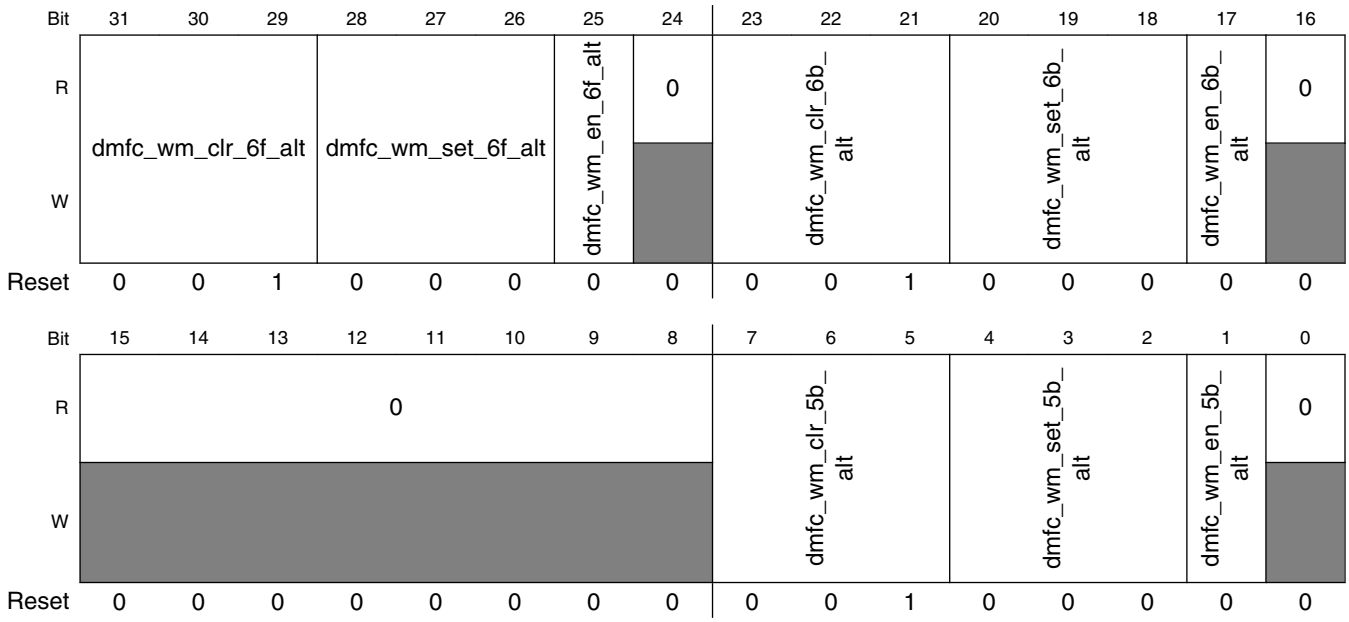
Table continues on the next page...

IPUx_DMFC_DP_CHAN_ALT field descriptions (continued)

Field	Description
7–6 dmfc_burst_size_5b_alt	Burst size of IDMAC's channel 23 (for alternate flow) This field defines the burst size of the IDMAC's channel 23 write accesses. This settings must match the settings in the corresponding IDMAC channel's settings. 00 32 words of 128 bit (4 pixels of 32 bit each, coming from the IDMAC) 01 16 words of 128 bit 10 8 words of 128 bit 11 4 words of 128 bit
5–3 dmfc_fifo_size_5b_alt	DMFC FIFO size for IDMAC's channel 23 (for alternate flow) This field defines the FIFO partition for the DMFC channel connected to IDMAC's channel 23 000 All (512X128 words) the DMFC's FIFO is allocated to this channel 001 256X128 words are allocated to this channel 010 128X128 words are allocated to this channel 011 64X128 words are allocated to this channel 100 32X128 words are allocated to this channel 101 16X128 words are allocated to this channel 110 8X128 words are allocated to this channel 111 4X128 words are allocated to this channel
dmfc_st_addr_5b_alt	DMFC Start Address for IDMAC's channel 23 (for alternate flow) This field defines the base address at the DMFC's FIFO of the partition allocated to the channel connected to IDMAC's channel 23. The FIFO is partitioned to 8 equal segments. Each segment is 64X128 words The value of this field is the number of the segment 000 Segment 0 001 Segment 1 111 Segment 7

37.5.394 DMFC Display Channel Definition Alternate Register (IPUx_DMFC_DP_CHAN_DEF_ALT)

Address: Base address + 6_002Ch offset



IPUx_DMFC_DP_CHAN_DEF_ALT field descriptions

Field	Description
31–29 dmfc_wm_clr_6f_alt	Watermark Clear (for alternate flow) This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
28–26 dmfc_wm_set_6f_alt	Watermark Set (for alternate flow) This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
25 dmfc_wm_en_6f_alt	Watermark enable. (for alternate flow) This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
24 Reserved	This read-only field is reserved and always has the value 0.
23–21 dmfc_wm_clr_6b_alt	Watermark Clear (for alternate flow) This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)

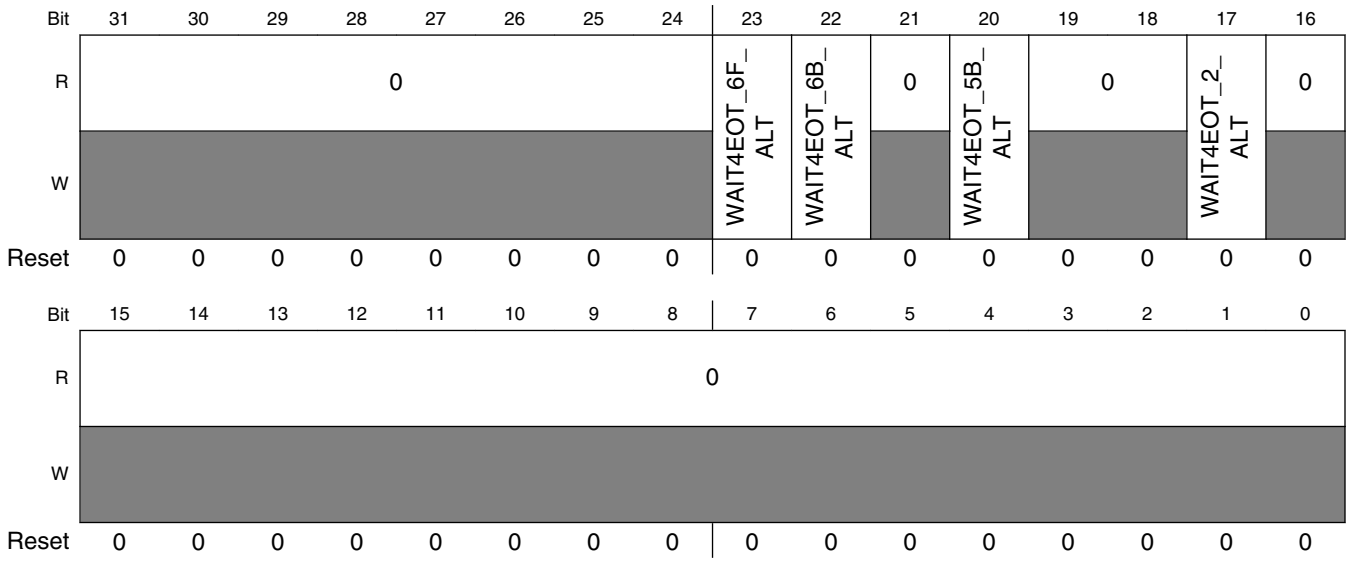
Table continues on the next page...

IPUx_DMFC_DP_CHAN_DEF_ALT field descriptions (continued)

Field	Description
20–18 dmfc_wm_set_ 6b_alt	Watermark Set (for alternate flow) This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
17 dmfc_wm_en_ 6b_alt	Watermark enable. (for alternate flow) This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
16–8 Reserved	This read-only field is reserved and always has the value 0.
7–5 dmfc_wm_clr_ 5b_alt	Watermark Clear (for alternate flow) This field defines the watermark's level of the DMFC's write FIFO. Crossing this level will clear the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set)
4–2 dmfc_wm_set_ 5b_alt	Watermark Set (for alternate flow) This field defines the watermark's level of the DMFC write FIFO. Crossing this level will send the watermark signal to the IDMAC. The WM level is the amount of occupied bursts at the FIFO (dmfc_wm_clr > dmfc_wm_set
1 dmfc_wm_en_ 5b_alt	Watermark enable. (for alternate flow) This bit enables the watermark feature of the FIFO 1 WM feature is enabled 0 WM feature is disabled
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.395 DMFC General 1 Alternate Register (IPUx_DMFC_GENERAL1_ALT)

Address: Base address + 6_0030h offset



IPUx_DMFC_GENERAL1_ALT field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23 WAIT4EOT_6F_ALT	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #6F is in wait4eot mode (for alternate flow) 0 FIFO #6F is in normal mode (for alternate flow)
22 WAIT4EOT_6B_ALT	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #6B is in wait4eot mode (for alternate flow) 0 FIFO #6B is in normal mode (for alternate flow)
21 Reserved	This read-only field is reserved and always has the value 0.
20 WAIT4EOT_5B_ALT	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode.

Table continues on the next page...

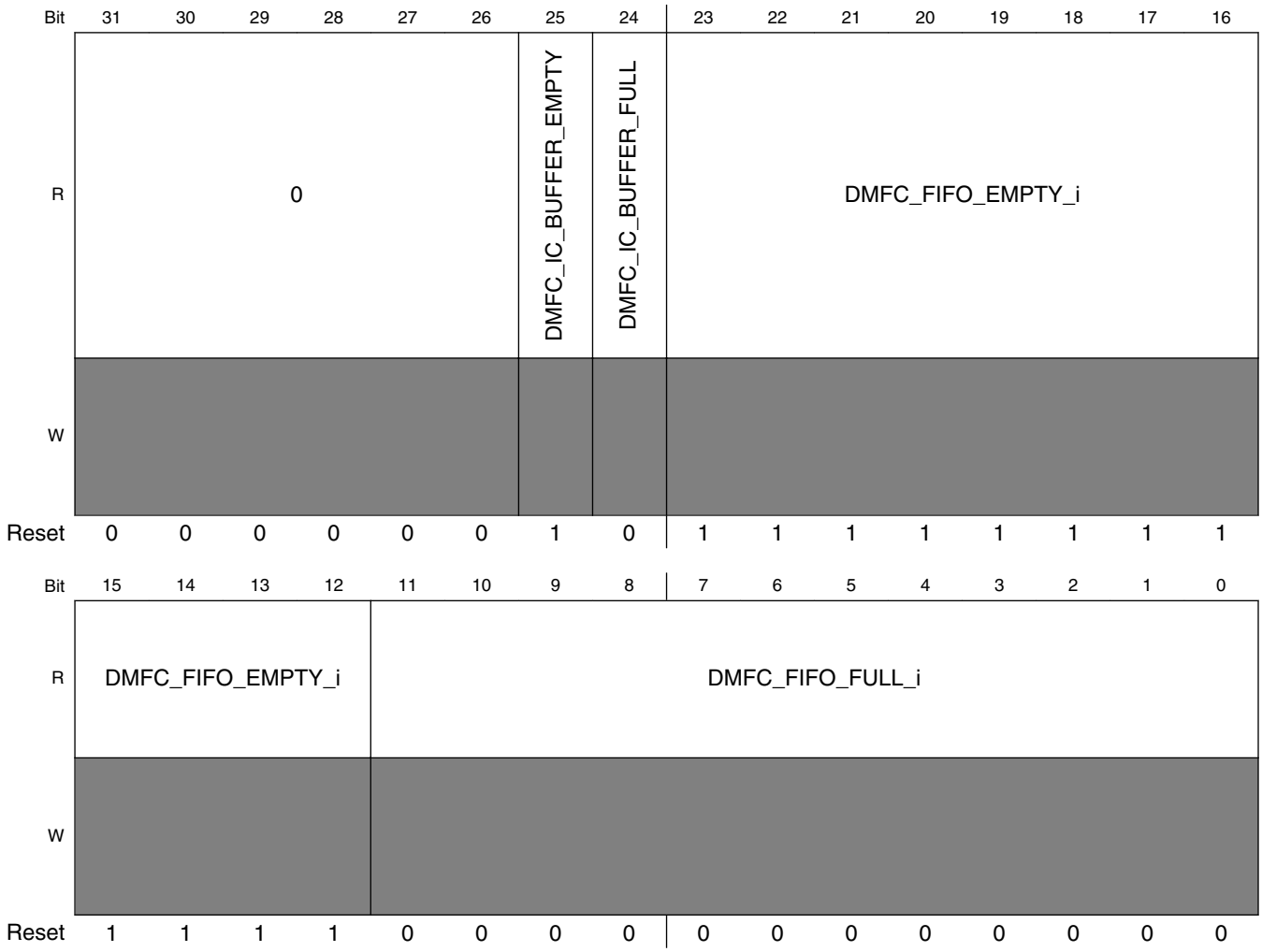
IPUx_DMFC_GENERAL1_ALT field descriptions (continued)

Field	Description
	1 FIFO #5B is in wait4eot mode (for alternate flow) 0 FIFO #5B is in normal mode (for alternate flow)
19–18 Reserved	This read-only field is reserved and always has the value 0.
17 WAIT4EOT_2_ALT	In normal operation the DMFC sends requests to the IDMAC whenever there is room in the FIFO. When this bit is set the DMFC sends the request only after the current transfer is terminated, (wait4eot mode) When the FIFO is larger then the size of the line, the user should work in wait4eot mode. 1 FIFO #2 is in wait4eot mode (for alternate flow) 0 FIFO #2 is in normal mode (for alternate flow)
Reserved	This read-only field is reserved and always has the value 0.

37.5.396 DMFC Status Register (IPUx_DMFC_STAT)

This register contains DMFC's status bits. All the bits in this register are read-only.

Address: Base address + 6_0034h offset



IPUx_DMFC_STAT field descriptions

Field	Description
31–26 Reserved	This read-only field is reserved and always has the value 0.
25 DMFC_IC_ BUFFER_ EMPTY	This bit indicates on a IC FIFO, inside the DMFC, empty condition. 0 IC FIFO not empty 1 IC FIFO is empty

Table continues on the next page...

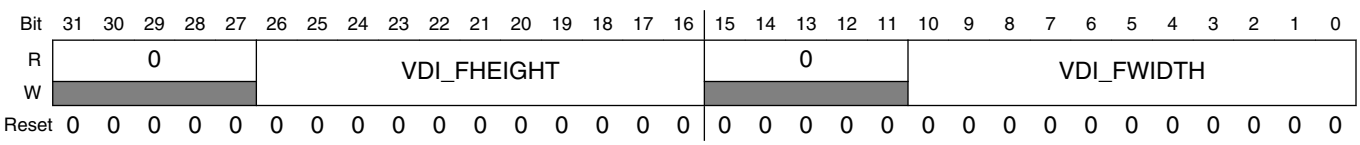
IPUx_DMFC_STAT field descriptions (continued)

Field	Description
24 DMFC_IC_ BUFFER_FULL	This bit indicates on a IC FIFO, inside the DMFC, full condition. 0 IC FIFO not full 1 IC FIFO is full
23–12 DMFC_FIFO_ EMPTY_i	This bit indicates on a DMFC FIFO#<i> empty condition. Mapping of these bit to an actual FIFO number is as follows: bit 0 => 0 bit 1=> 1 bit 2 => 2 bit 3 =>1c bit 4 => 2c bit 5 => 5b bit 6 => 5f bit 7 => 6b bit 8 => 6f bit 9 => 9 bit 10 => 10 (ARM platform access) bit 11 => 11 (ARM platform access) 0 FIFO #<i> is not empty 1 FIFO #<i> is empty
DMFC_FIFO_ FULL_i	This bit indicates on a DMFC FIFO#<i> full condition. Mapping of these bit to an actual FIFO number is as follows: bit 0 => 0 bit 1=> 1 bit 2 => 2 bit 3 =>1c bit 4 => 2c bit 5 => 5b bit 6 => 5f bit 7 => 6b bit 8 => 6f bit 9 => 9 bit 10 => 10 (ARM platform access) bit 11 => 11 (ARM platform access) 0 FIFO #<i> is not full 1 FIFO #<i> is full

37.5.397 VDI Field Size Register (IPUx_VDI_FSIZE)

The register used to control size of VDIC input fields.

Address: Base address + 6_8000h offset



IPUx_VDI_FSIZE field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 VDI_FHEIGHT	Frame height The value to be written to this register is the frame's height minus 1. The frame height should not be smaller than 16. When VDI_CMB_EN bit is clear: • The frame height should not be greater than 1080. • The frame's height must be even (which means that both fields have the same height) • The frame's height in 4:2:0 format, must be multiple of 4 (which means that both chroma fields have the same height)

Table continues on the next page...

IPUx_VDI_FSIZE field descriptions (continued)

Field	Description
	When VDI_CMB_EN bit is set: • The frame height should not be greater than 1200.
15–11 Reserved	This read-only field is reserved and always has the value 0.
VDI_FWIDTH	Frame width. The value to be written to this register is the frame's width minus 1. The Frame width should not be smaller than 16. The width must be even. When VDI_CMB_EN bit is clear • The Frame width should not be greater than 720968. When VDI_CMB_EN bit is set: • The Frame width should not be greater than 1920.

37.5.398 VDI Control Register (IPUx_VDI_C)

The register used to control modes of operations of VDIC module.

Address: Base address + 6_8004h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	-	-	0		VDI_VWM3_CLR				VDI_VWM3_SET		VDI_VWM1_CLR			VDI_VWM1_SET		
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	VDI_BURST_SIZE3				VDI_BURST_SIZE2				VDI_BURST_SIZE1				VDI_MOT_SEL		VDI_CH_422	0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_C field descriptions

Field	Description
31 -	VDIC top filed (automatic) This defines what would be the top field to be processed when the data is coming from the CSI 0 top field is field 0 1 top field is field 1
30 -	VDIC top filed (manual) This defines what would be the next top field to be processed when the data is coming from the memory

Table continues on the next page...

IPUx_VDI_C field descriptions (continued)

Field	Description
	0 top field is field 0 1 top field is field 1
29–28 Reserved	This read-only field is reserved and always has the value 0.
27–25 VDI_VWM3_CLR	VDIC WaterMark "clear" level for channel 3. 0 clear watermark level when FIFO3 is full on 1/8 of their size. 1 clear watermark level when FIFO3 is full on 2/8 of their size. 7 clear watermark level when FIFO3 is full.
24–22 VDI_VWM3_SET	VDIC WaterMark "set" level for channel 3. 0 set watermark level when FIFO3 is full on 1/8 of their size. 1 set watermark level when FIFO3 is full on 2/8 of their size. 7 set watermark level when FIFO3 is full.
21–19 VDI_VWM1_CLR	VDIC WaterMark "clear" level for channel 1 or channel 4 (channels 1 and 4 are not working simultaneously). 0 clear watermark level when FIFO1 is full on 1/8 of their size. 1 clear watermark level when FIFO1 is full on 2/8 of their size. 7 clear watermark level when FIFO1 is full.
18–16 VDI_VWM1_SET	VDIC WaterMark "set" level for channel 1 or channel 2 (channels 1 and 4 are not working simultaneously). 0 set watermark level when FIFO1 is full on 1/8 of their size. 1 set watermark level when FIFO1 is full on 2/8 of their size. 7 set watermark level when FIFO1 is full.
15–12 VDI_BURST_ SIZE3	Burst Size for channel 3. The VDIC's burst size is restrict by the IDMAC's restriction - This value must match the IDMAC settings and follow the IDMAC's restrictions 0 Burst size is 1 access. 1 Burst size is 2 accesses. 15 Burst size is 16 accesses.
11–8 VDI_BURST_ SIZE2	Burst Size for channel 2. The VDIC's burst size is restrict by the IDMAC's restriction - This value must match the IDMAC settings and follow the IDMAC's restrictions 0 Burst size is 1 access. 1 Burst size is 2 accesses. 15 Burst size is 16 accesses.
7–4 VDI_BURST_ SIZE1	Burst Size for channels 1 or 4 (channels 1 and 4 are not working simultaneously). The VDIC's burst size is restrict by the IDMAC's restriction - This value must match the IDMAC settings and follow the IDMAC's restrictions 0 Burst size is 1 access. 1 Burst size is 2 accesses. 15 Burst size is 16 accesses.
3–2 VDI_MOT_SEL	Motion select.

Table continues on the next page...

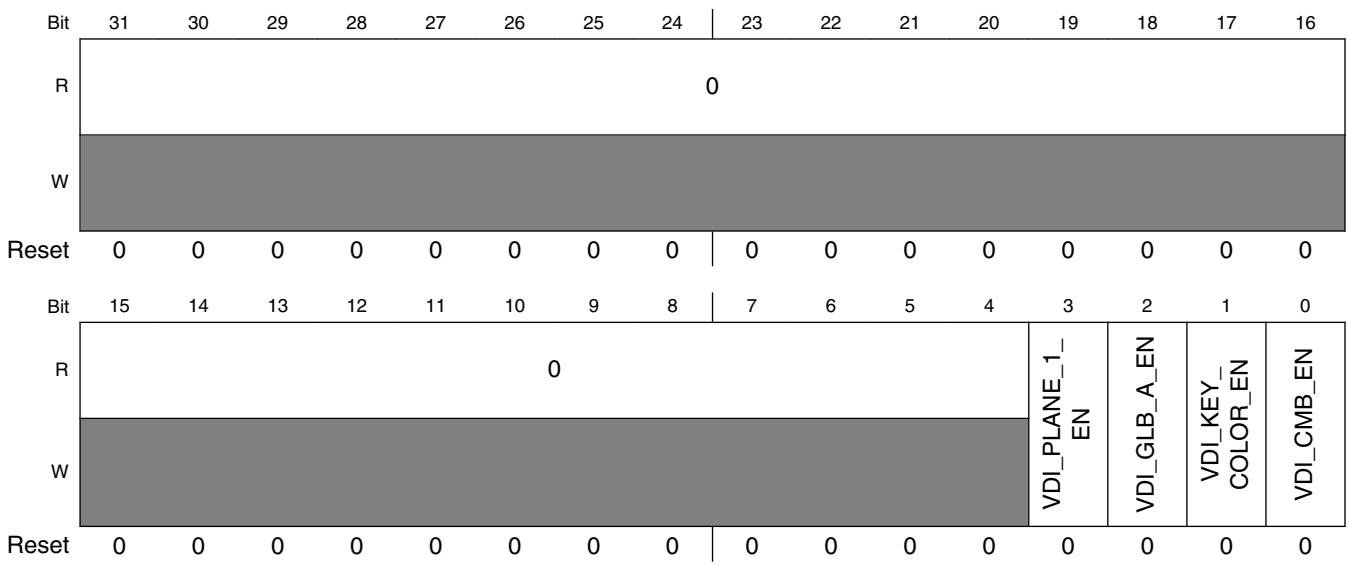
IPUx_VDI_C field descriptions (continued)

Field	Description
	0 Motion determined by ROM "1" (shared toward medium/high motion). 1 Motion determined by ROM "2" (This option will not work well for high motion). 2 Full motion, only vertical filter is used 3 Forbidden.
1 VDI_CH_422	Chroma format at input and output of VDIC. 0 Chroma format is 420. 1 Chroma format is 422.
0 Reserved	This read-only field is reserved and always has the value 0.

37.5.399 VDI Control Register 2 (IPUx_VDI_C2_)

The register used to control modes of operations of VDIC module.

Address: Base address + 6_8008h offset



IPUx_VDI_C2_ field descriptions

Field	Description
31–4 Reserved	This read-only field is reserved and always has the value 0.
3 VDI_PLANE_1_EN	Plane 1 enable 0 plane #1 is disabled 1 plane #1 is enabled

Table continues on the next page...

IPUx_VDI_C2_ field descriptions (continued)

Field	Description
2 VDI_GLB_A_EN	Global alpha enable 0 Alpha is local 1 Alpha is global
1 VDI_KEY_ COLOR_EN	Key Color Enable 0 Key Color disabled. 1 Key color enabled
0 VDI_CMB_EN	Combining enable 0 Combining disabled. The VDIC works in de-interlacing mode 1 Combining enabled. The de-interlacing mode is not functional

37.5.400 VDI Combining Parameters Register 1 (IPUx_VDI_CMDP_1)

The register holds combining paramemters.

Address: Base address + 6_800Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	VDI_ALPHA								VDI_KEY_COLOR_R								VDI_KEY_COLOR_G								VDI_KEY_COLOR_B							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_CMDP_1 field descriptions

Field	Description
31–24 VDI_ALPHA	Global Alpha Actual value of the alpha is VDI_ALPHA + VDI_ALPHA[7]
23–16 VDI_KEY_ COLOR_R	Red component of Key Color
15–8 VDI_KEY_ COLOR_G	Green component of Key Color
VDI_KEY_ COLOR_B	Blue component of Key Color

37.5.401 VDI Combining Parameters Register 2 (IPUx_VDI_CMDP_2)

The register holds combining paramemters.

Address: Base address + 6_8010h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0								VDI_KEY_COLOR_R								VDI_KEY_COLOR_G								VDI_KEY_COLOR_B							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_CMDP_2 field descriptions

Field	Description
31–24 Reserved	This read-only field is reserved and always has the value 0.
23–16 VDI_KEY_ COLOR_R	Red component of background Color
15–8 VDI_KEY_ COLOR_G	Green component of background Color
VDI_KEY_ COLOR_B	Blue component of background Color

37.5.402 VDI Plane Size Register 1 (IPUx_VDI_PS_1)

The register holds the plane size's paramemters.

Address: Base address + 6_8014h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					VDI_FHEIGHT1											0					VDI_FWIDTH1										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_PS_1 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 VDI_FHEIGHT1	Plane 1 height

Table continues on the next page...

IPUx_VDI_PS_1 field descriptions (continued)

Field	Description
	The value to be written to this register is the plane's height minus 1. The Plane height should not be smaller than 16. The Plane height should not be greater than 1200. The Plane's height must be even
15–11 Reserved	This read-only field is reserved and always has the value 0.
VDI_FWIDTH1	Plane 1 width. The value to be written to this register is the plane's width minus 1. The Plane width should not be smaller than 16. The Plane width should not be greater than 1920. The width must be even.

37.5.403 VDI Plane Size Register 2 (IPUx_VDI_PS_2)

The register holds the plane's offset paramemters.

Address: Base address + 6_8018h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					VDI_OFFSET_VER1											0					VDI_OFFSET_HOR1										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_PS_2 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 VDI_OFFSET_ VER1	Vertical offset of plane 1
15–11 Reserved	This read-only field is reserved and always has the value 0.
VDI_OFFSET_ HOR1	Horizontal offset of plane 1

37.5.404 VDI Plane Size Register 3 (IPUx_VDI_PS_3)

The register holds the plane size's paramemters.

Address: Base address + 6_801Ch offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					VDI_FHEIGHT3											0					VDI_FWIDTH3										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_PS_3 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 VDI_FHEIGHT3	Plane 3 height The value to be written to this register is the plane's height minus 1. The Plane height should not be smaller than 16. The Plane height should not be greater than 1200. The Plane's height must be even
15–11 Reserved	This read-only field is reserved and always has the value 0.
VDI_FWIDTH3	Plane 3 width. The value to be written to this register is the plane's width minus 1. The Plane width should not be smaller than 16. The Plane width should not be greater than 1920. The width must be even.

37.5.405 VDI Plane Size Register 4 (IPUx_VDI_PS_4)

The register holds the plane's offset paramemters.

Address: Base address + 6_8020h offset

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0					VDI_OFFSET_VER3											0					VDI_OFFSET_HOR3										
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IPUx_VDI_PS_4 field descriptions

Field	Description
31–27 Reserved	This read-only field is reserved and always has the value 0.
26–16 VDI_OFFSET_VER3	Vertical offset of plane 3
15–11 Reserved	This read-only field is reserved and always has the value 0.
VDI_OFFSET_HOR3	Horizontal offset of plane 3

Chapter 38
Keypad Port (KPP)

38.1 Overview

The Keypad Port (KPP) is a 16-bit peripheral that can be used as a keypad matrix interface or as general purpose input/output (I/O).

The figure below shows the KPP block diagram. The KPP provides interface for the keypad matrix with 2-point contact or 3-point contact keys. The KPP is designed to simplify the software task of scanning a keypad matrix. With appropriate software support, the KPP is capable of detecting, debouncing, and decoding one or multiple keys pressed simultaneously on the keypad.

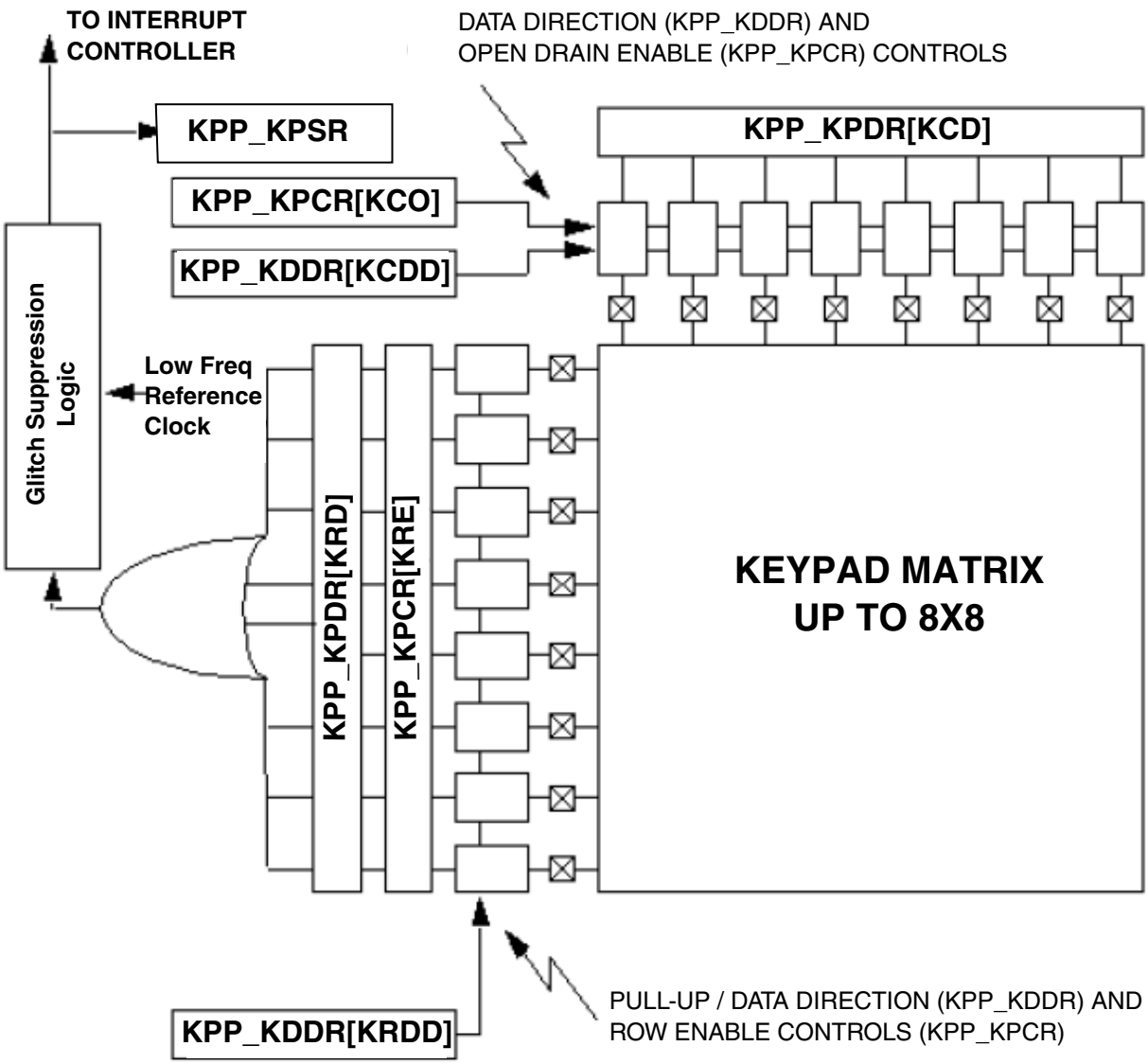


Figure 38-1. KPP Peripheral Block Diagram

38.1.1 Features

The KPP includes these distinctive features:

- Supports up to an 8 x 8 external key pad matrix
- Port pins can be used as general purpose I/O
- Open drain design
- Glitch suppression circuit design
- Multiple-key detection
- Long key-press detection
- Standby key-press detection
- Synchronizer chain clear
- Supports a 2-point and 3-point contact key matrix

38.1.2 Modes and Operations

This block supports the following modes:

- Run Mode-This is the normal functional mode in which the KPP can detect any key press event.
- Low Power Mode-The keypad can detect any key press even in low power modes (when there is no MCU clock).

38.2 Clocks

The table found here describes the clock sources for KPP.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 38-1. KPP Clocks

Clock name	Clock Root	Description
ipg_clk_32k	ckil_sync_clk_root	Low-frequency reference clock (32kHz)
ipg_clk_s	ipg_clk_root	Peripheral access clock

38.3 External Signals

There are several pins dedicated to the KPP. Keypads of any configuration up to eight rows and eight columns are supported through the software configuration of the peripheral pins. Any pins not used for the keypad are available as general purpose I/O. The registers are configured such that the pins can be treated as an I/O port up to 16 bits wide.

See the table below for the list of external signals.

Table 38-2. KPP External Signals

Signal	Description	Pad	Mode	Direction
KEY_COL0	Column input or output pin, from chip	KEY_COL0	ALT3	I/O
KEY_COL1	Column input or output pin, from chip	KEY_COL1	ALT3	I/O
KEY_COL2	Column input or output pin, from chip	KEY_COL2	ALT3	I/O
KEY_COL3	Column input or output pin, from chip	KEY_COL3	ALT3	I/O
KEY_COL4	Column input or output pin, from chip	KEY_COL4	ALT3	I/O
KEY_COL5	Column input or output pin, from chip	CSI0_DAT4	ALT3	I/O
		GPIO_0	ALT2	
		GPIO_19	ALT0	
		SD2_CLK	ALT2	
KEY_COL6	Column input or output pin, from chip	CSI0_DAT6	ALT3	I/O
		GPIO_9	ALT2	
		SD2_DAT3	ALT2	
KEY_COL7	Column input or output pin, from chip	CSI0_DAT8	ALT3	I/O
		GPIO_4	ALT2	
		SD2_DAT1	ALT4	
KEY_ROW0	Row input or output pin, from chip	KEY_ROW0	ALT3	I/O
KEY_ROW1	Row input or output pin, from chip	KEY_ROW1	ALT3	I/O
KEY_ROW2	Row input or output pin, from chip	KEY_ROW2	ALT3	I/O
KEY_ROW3	Row input or output pin, from chip	KEY_ROW3	ALT3	I/O
KEY_ROW4	Row input or output pin, from chip	KEY_ROW4	ALT3	I/O
KEY_ROW5	Row input or output pin, from chip	CSI0_DAT5	ALT3	I/O
		GPIO_1	ALT2	
		SD2_CMD	ALT2	
KEY_ROW6	Row input or output pin, from chip	CSI0_DAT7	ALT3	I/O
		GPIO_2	ALT2	
		SD2_DAT2	ALT4	
KEY_ROW7	Row input or output pin, from chip	CSI0_DAT9	ALT3	I/O

Table continues on the next page...

Table 38-2. KPP External Signals (continued)

Signal	Description	Pad	Mode	Direction
		GPIO_5	ALT2	
		SD2_DAT0	ALT4	

38.3.1 Input Pins

Any of the 16 pins associated with the KPP can be configured as inputs by writing a "0" to the appropriate bits in the KPP_KDDR. Additionally, the least significant 8 bits (ROW inputs) corresponding to KPP_KDDR[KRDD] have internal pull-ups, which are enabled when the pin is used as an input.

38.3.2 Output Pins

Any of the 16 pins associated with the KPP can be configured as outputs by writing the appropriate bits in the KPP_KDDR to a "1". Additionally, the 8 most significant bits (15-8) can be designated as open drain outputs by writing a "1" to the appropriate bits in the KPP_KPCR. The lower 8 bits (7-0) are always in "totem pole" style, driven when configured as outputs.

See the table below.

Table 38-3. Keypad Port Column Modes

KPP_KDDR (15:8)	KPP_KPCR (15:8)	Pin Function
0	x	Input
1	0	Totem-Pole Output
1	1	Open-Drain Output

NOTE

Totem pole capability should be provided for column pins. Totem pole configuration helps for a faster discharge of keypad capacitance when all columns need to be quickly brought to a "1" during the scan routine. With this configuration, delay between the scanning of two subsequent columns is reduced.

38.3.3 Generation of Transfer Error Signal on Peripheral Bus

If there is an access to an address which is not implemented, then the KPP asserts a transfer error signal on Peripheral Bus.

38.4 Functional Description

The Keypad Port (KPP) is designed to simplify the software task of scanning a keypad matrix. With appropriate software support and matrix organization, the KPP is capable of detecting, debouncing, and decoding one or more keys pressed simultaneously on the keypad.

Logic in the KPP is capable of detecting a key press even while the processor is in one of the low power standby modes provided that a low frequency reference clock is on. The KPP may generate an ARM platform interrupt any time a key press or key release is detected. This interrupt is capable of forcing the processor out of a low power mode.

38.4.1 Keypad Matrix Construction

The KPP is designed to interface to a keypad matrix, which shorts the intersecting row and column lines together whenever a key is depressed. The interface is not optimized for any other switch configuration.

38.4.2 Keypad Port Configuration

The software must initialize the KPP for the size of the keypad matrix. Pins connected to the keypad columns should be configured as open-drain outputs. Pins connected to the keypad rows should be configured as inputs. On-chip, pull-up resistors should be implemented for active keypad rows.

In addition to enabled row inputs in the Keypad Control register, corresponding interrupt (depress or/and release) must also be enabled to generate an interrupt.

Discrete switches that are not part of the matrix may be connected to any unused row inputs. The second terminal of the discrete switch is connected to ground. The hardware detects closures of these switches without the need for software polling.

38.4.3 Keypad Matrix Scanning

Keypad scanning is performed by a software loop that walks a zero across each of the keypad columns, reading the value on the rows at each step. The process is repeated several times in succession, with the results of each pass optionally compared to those from the previous pass. When several (3 or 4) consecutive scans yield the same key closures, a valid key press has been detected. Software then can decode exactly which switch was depressed and pass the value up to the next higher software layer.

The basic debouncing period, which must be defined in the software routine, may be controlled with an internal timer. The basic period is the period between the scan of two consecutive columns, so the debouncing time between two consecutive scans of the whole matrix shall be the number of columns multiplied by the basic period.

38.4.4 Keypad Standby

There is no need for the ARM platform to continually scan the keypad. Between key presses, the keypad can be left in a state that requires no software intervention until the next key press is detected. To place the keypad in a standby state, software should write all column outputs low. Row inputs are left enabled. At this point, the ARM platform can attend to other tasks or revert to a low power standby mode. The KPP will interrupt the ARM platform if any key is pressed.

Upon receiving a keypad interrupt, the ARM platform should set all the column strobes high, and begin a normal keypad scanning routine to determine which key was pressed. It is important that open-drain drivers be used when scanning to prevent a possible DC path between power and ground through two or more switches.

38.4.5 Glitch Suppression on Keypad Inputs

A glitch suppression circuit qualifies the keypad inputs to prevent noise from inadvertently interrupting the ARM platform. The circuit is a 4-state synchronizer clocked from a low frequency reference clock source.

This clock must continue to run in any low power mode where the keypad is a wake-up source, as the ARM platform interrupt is generated from the synchronized input. An interrupt is not generated until all four synchronizer stages have latched a valid key assertion. This guarantees the filtering out of any noise less than three clock periods in duration of a low frequency reference clock. Noise filtering of the duration between three to four clock periods cannot be guaranteed. The interrupt output is latched in an S-R latch and remains asserted until cleared by the software. The Set input of the latch is rising-edge clocked. See the figure below.

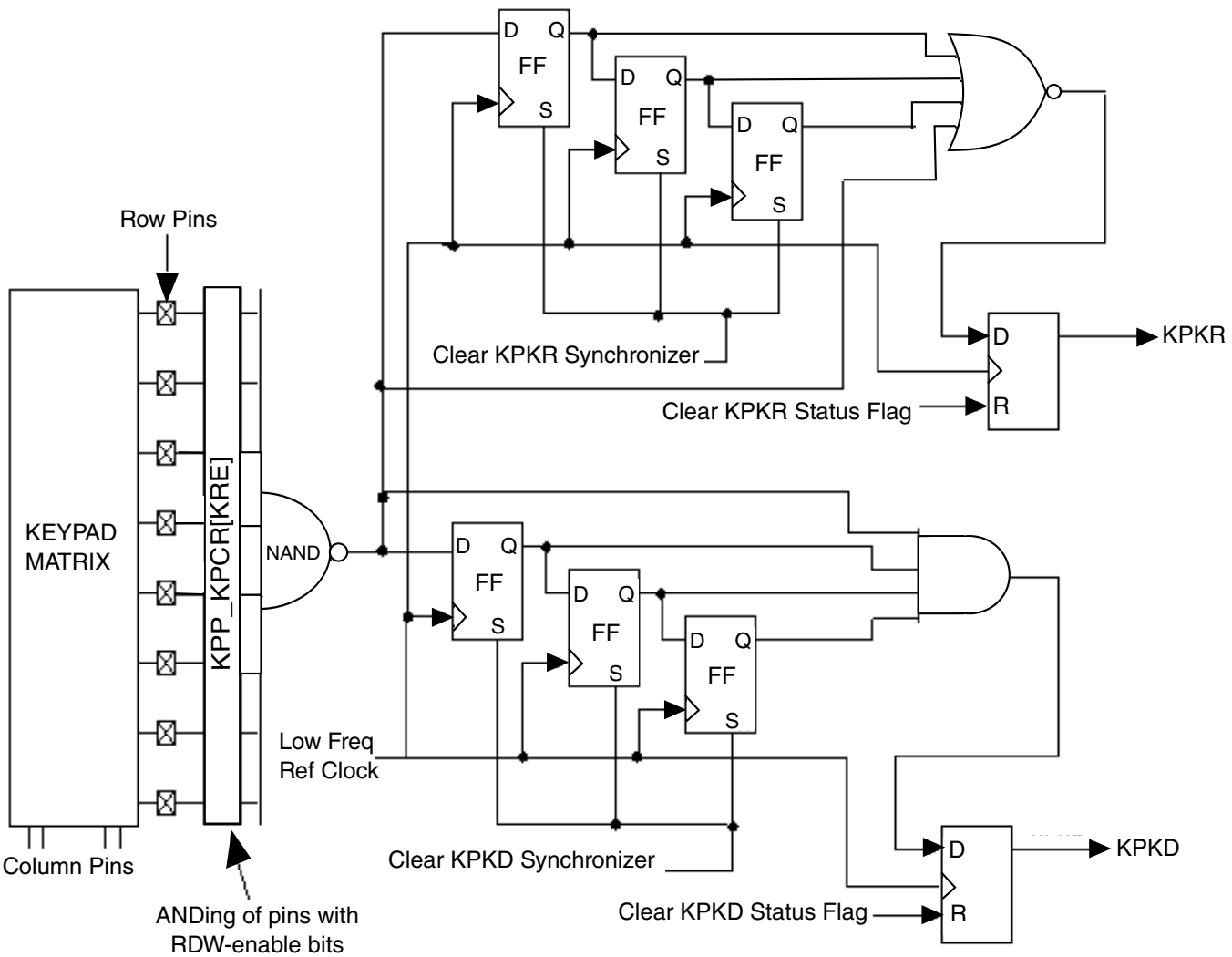


Figure 38-2. Keypad Synchronizer Functional Diagram

38.4.6 Multiple Key Closures

Using the key press and Key release interrupts, the software can detect multiple keys or achieve n key rollover. The key scanning routine can be programmed accordingly.

See the following figures for illustrations of the interfacing of a 2-contact keypad matrix with the KPP controller. With proper enabling of row lines and the performing scan-routine, multiple key presses can be detected. When keys present on the same row are pressed, corresponding row lines (multiple lines) become low when the column is driven low during a scan-routine. By reading the data-register, pressed keys can be detected. Similarly, when keys present on same row line are pressed, the corresponding row line (only one line) becomes low when logic "0" is driven on the column line during a scan-routine.

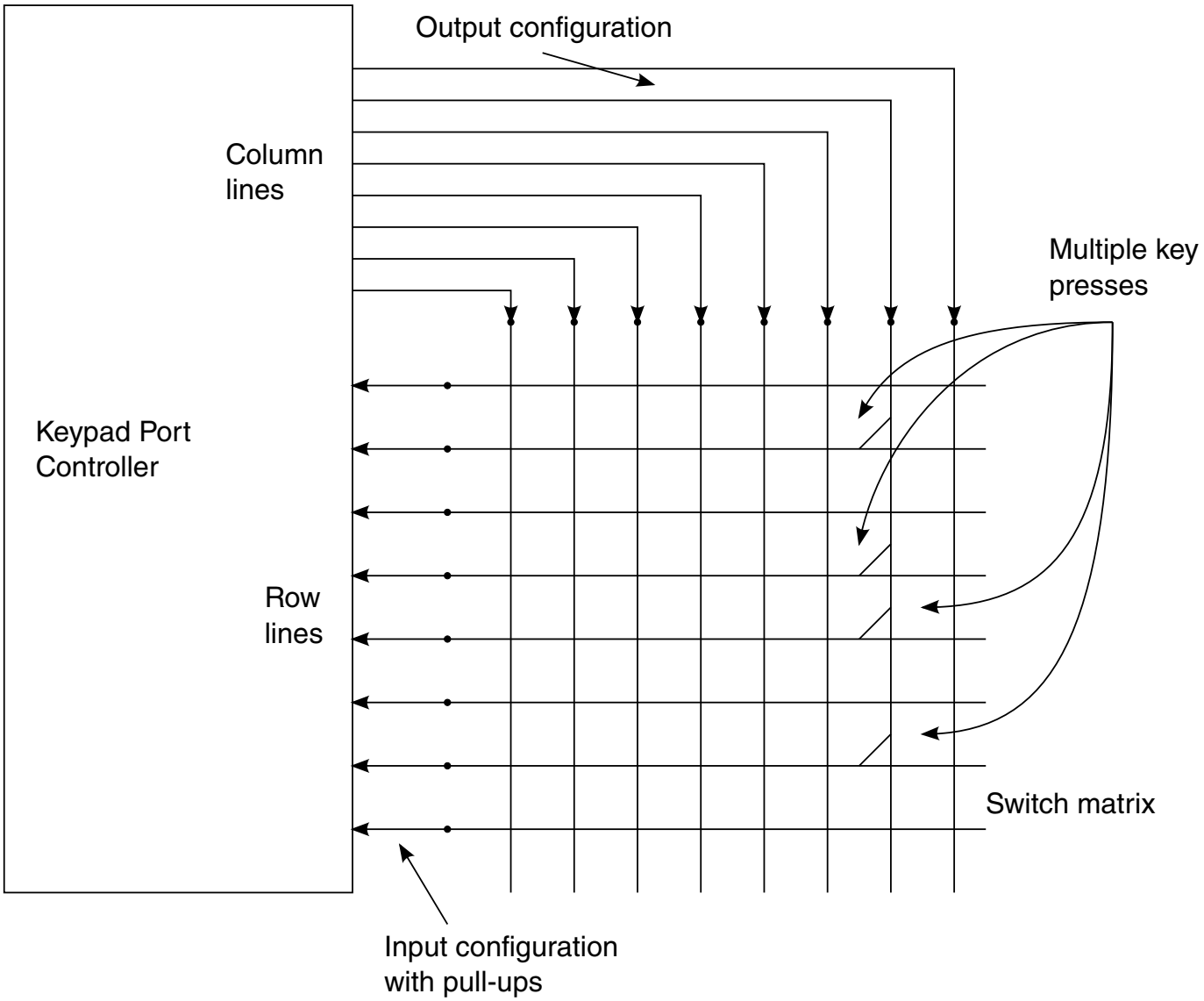


Figure 38-3. Multiple Key Presses on Same Column Line (Simplified View)

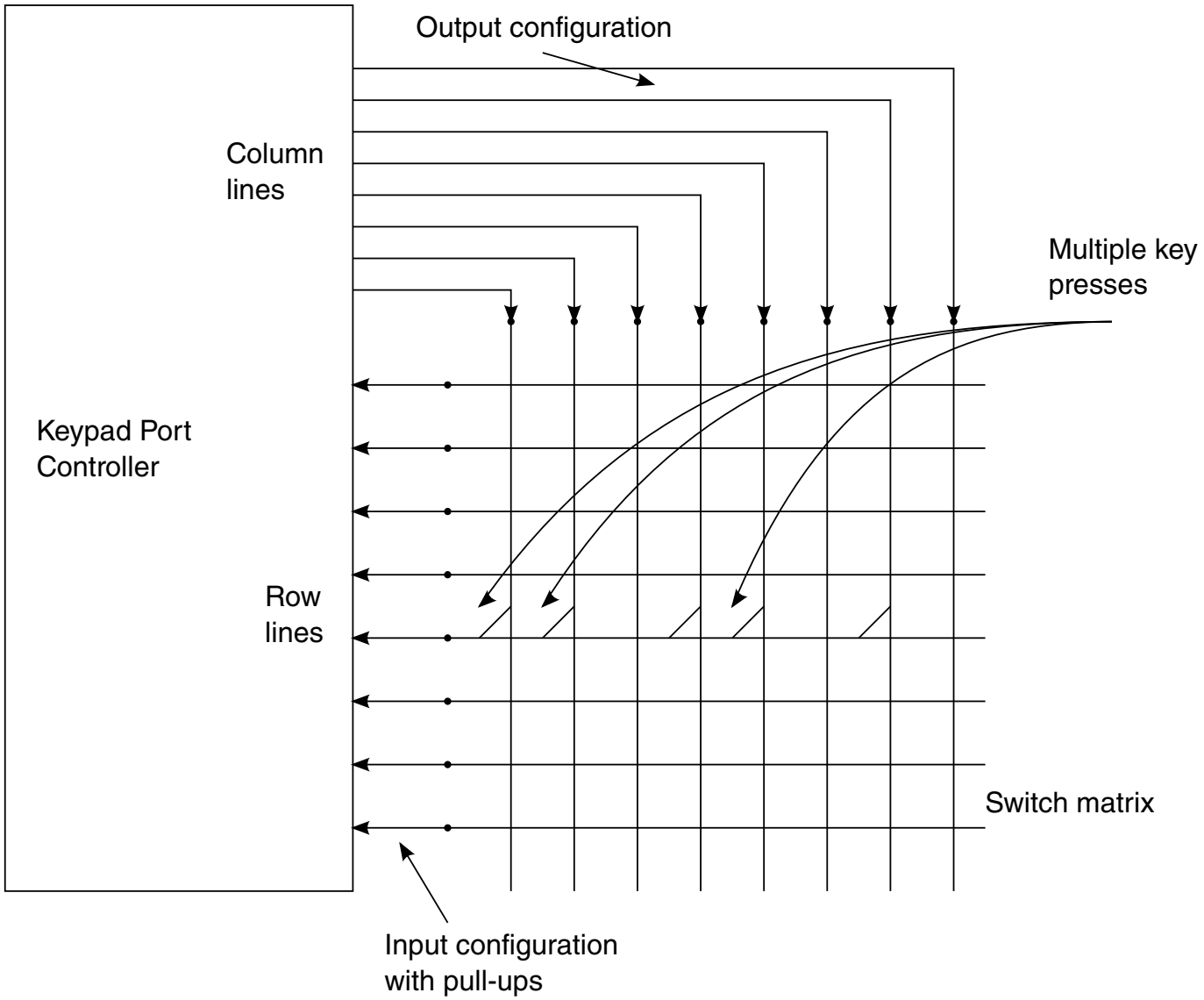


Figure 38-4. Multiple Key Presses on Same Row Line (Simplified View)

NOTE

An n key rollover is a technique with which the system can recognize the order in which keys are pressed.

38.4.6.1 Ghost Key Problem and Correction

The KPP detects if one or multiple keys are pressed or released. In the case where a simple keypad matrix with two-contact switches is used, there is a chance of "ghost" key detection when three or more keys are pressed. This is a limitation imposed by such a keypad matrix.

As can be seen in Figure 38-5, three keys pressed simultaneously can cause a short between the column currently "scanned" by the software and another column. Depending on the location of the third key pressed, a "ghost" key press may be detected.

However, this can be corrected by using a keypad matrix that provides "ghost" key protection. Such a matrix implements a one-way "diode" at all keypad points between rows and columns. This way, the multiple pressing of three keys will not cause a short at a fourth key (see Figure 38-6).

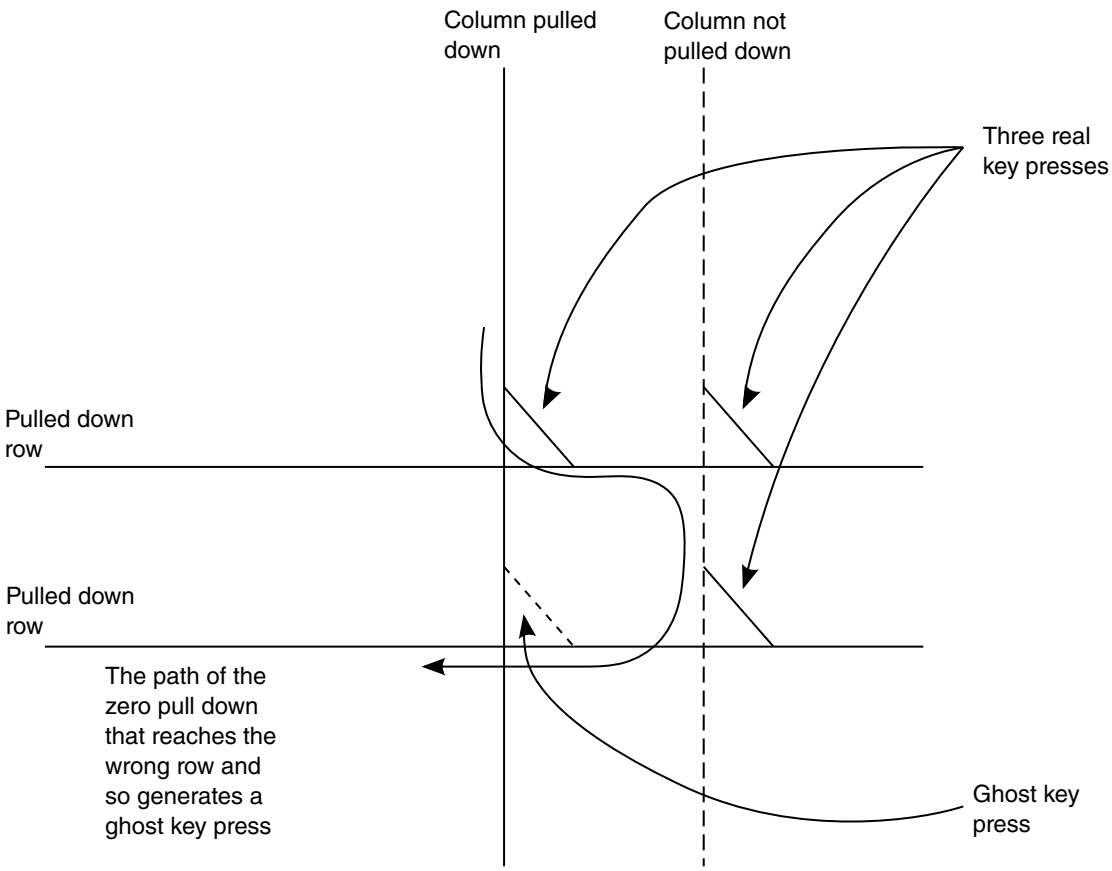


Figure 38-5. Decoding Wrong Three- Key-Presses

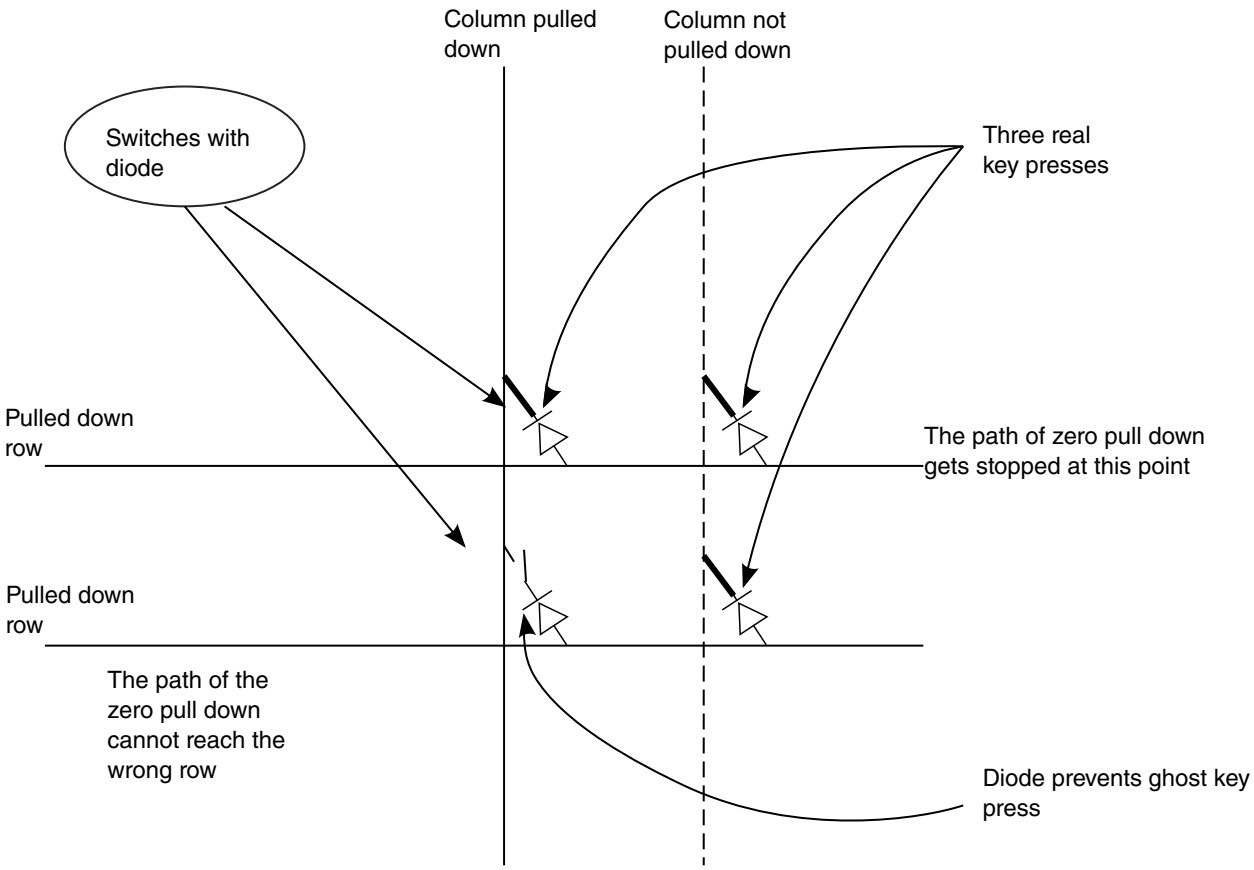


Figure 38-6. Matrix with "Ghost" Key Protections

38.4.7 3-Point Contact Keys Support

The KPP supports interfacing to a matrix consisting of 3-point contact keys. As shown in Figure 38-7, two points of such a key are connected to keypad lines, while a third point is connected to ground (low logic).

The keypad lines should be configured as input and a pull-up should to be present on these lines. When such a key is pressed, corresponding keypad lines go low and an interrupt is generated. There is no need to perform a scanning routine for identification of pressed key as it can be done by reading the keypad data-register. A limitation with such a matrix is that for every key at least one keypad row line should be used.

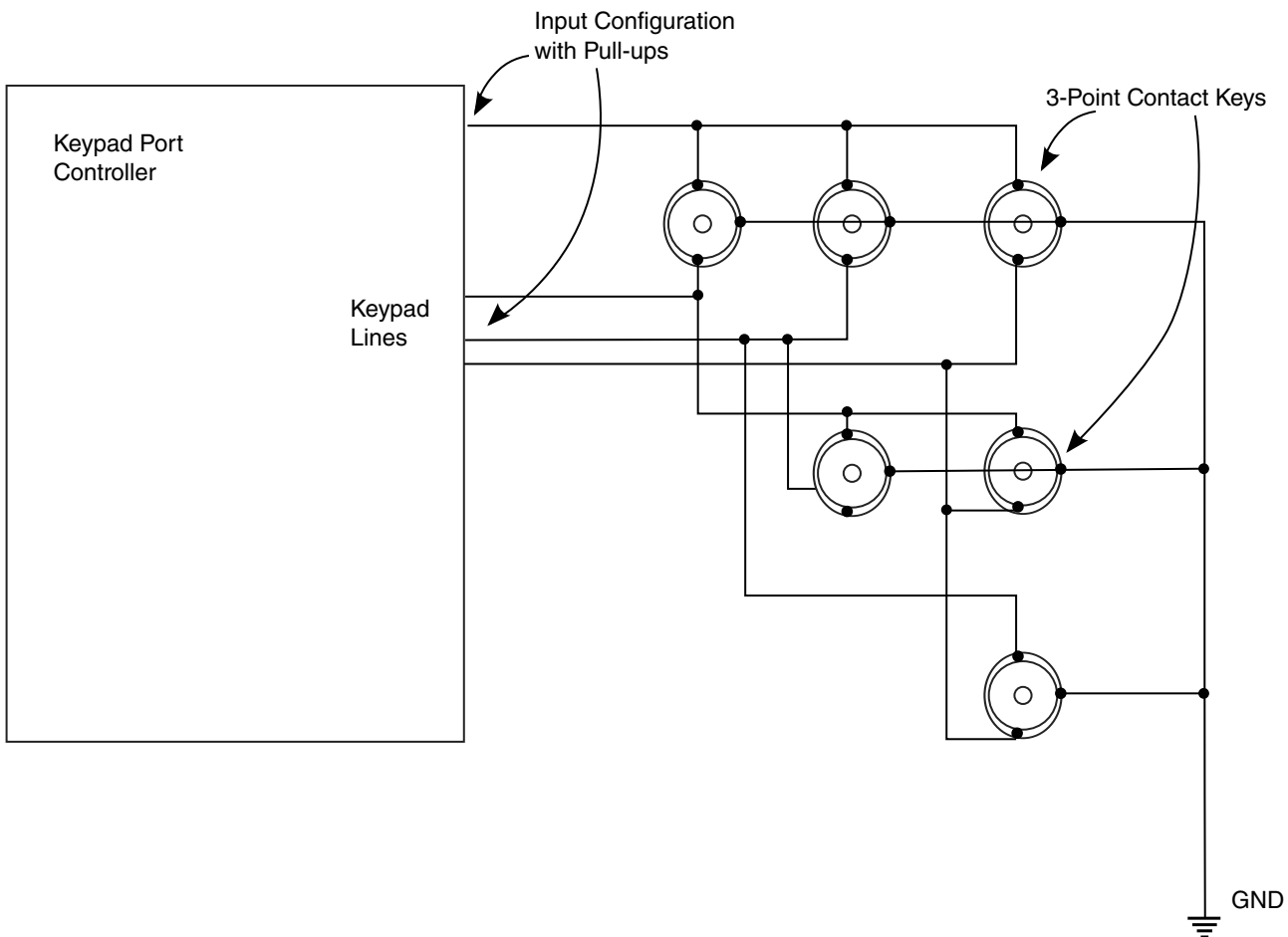


Figure 38-7. KPP Interface with 3-point Contact Key Matrix (Simplified View)

38.5 Initialization/Application Information

38.5.1 Typical Keypad Configuration and Scanning Sequence

Perform the following steps to configure the keypad:

1. Enable the number of rows in the keypad (KPP_KPCR[KRE]).
2. Write 0s to KPP_KPDR[KCD].
3. Configure the keypad columns as open-drain (KPP_KPCR[KCO]).
4. Configure columns as output (KPP_KDDR[KCDD]) and rows as input (KPP_KDDR[KRDD]).
5. Clear the KPKD Status Flag and Synchronizer chain.
6. Set the KDIE control bit, and clear the KRIE control bit (avoid false release events).
7. (The system is now in standby mode, and awaiting a key press.)

38.5.2 Key Press Interrupt Scanning Sequence

Perform the following steps to perform a keypad scanning routine:

1. Disable both (depress and release) keypad interrupts.
2. Write 1s to KPP_KPDR[KCD], setting column data to 1s.
3. Configure columns as totem pole outputs (for quick discharging of keypad capacitance).
4. Configure columns as open-drain.
5. Write a single column to 0, and other columns to 1.
6. Sample row inputs and save data. Multiple key presses can be detected on a single column.
7. Repeat Steps 2-6 for remaining columns.
8. Return all columns to 0 in preparation for standby mode.
9. Clear KPKD and KPKR status bit(s) by writing to a "1"; set the KPKR synchronizer chain by writing a "1" to the KPP_KRSS register; and clear the KPKD synchronizer chain by writing a "1" to the KDSC register.
10. Re-enable the appropriate keypad interrupt(s) so that the KDIE detects a key hold condition, or the KRIE detects a key-release event.

38.5.3 Additional Comments

The order of key press detection can be done in software only. Therefore, the software may need to run the scan routines at very short intervals of time per the application's demands. The reason that such functionality cannot be put in the KPP is that the block is limited by the number of external pins.

For the keys that require a very precise order (such as game keys), individual GPIO pins may be more useful.

38.6 KPP Memory Map/Register Definition

The KPP contains four registers.

KPP memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20B_8000	Keypad Control Register (KPP_KPCR)	16	R/W	0000h	38.6.1/3554
20B_8002	Keypad Status Register (KPP_KPSR)	16	R/W	0400h	38.6.2/3555
20B_8004	Keypad Data Direction Register (KPP_KDDR)	16	R/W	0000h	38.6.3/3557
20B_8006	Keypad Data Register (KPP_KPDR)	16	R/W	0000h	38.6.4/3557

38.6.1 Keypad Control Register (KPP_KPCR)

The Keypad Control Register determines which of the eight possible column strobes are to be open drain when configured as outputs, and which of the eight row sense lines are considered in generating an interrupt to the core.

It is up to the programmer to ensure that pins being used for functions other than the keypad are properly disabled. The KPP_KPCR register is byte- or half-word-addressable.

Address: 20B_8000h base + 0h offset = 20B_8000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	KCO								KRE							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KPP_KPCR field descriptions

Field	Description
15–8 KCO	Keypad Column Strobe Open-Drain Enable. Setting a column open-drain enable bit (KCO7-KCO0) disables the pull-up driver on that pin. Clearing the bit allows the pin to drive to the high state. This bit has no effect when the pin is configured as an input. **NOTE:** Configuration of external port control logic (for example, IOMUX) should be done properly so that the KPP controls an open-drain enable of the pin. 0 **TOTEM_POLE** — Column strobe output is totem pole drive. 1 **OPEN_DRAIN** — Column strobe output is open drain.

Table continues on the next page...

KPP_KPCR field descriptions (continued)

Field	Description
KRE	Keypad Row Enable. Setting a row enable control bit in this register enables the corresponding row line to participate in interrupt generation. Likewise, clearing a bit disables that row from being used to generate an interrupt. This register is cleared by a reset, disabling all rows. The row-enable logic is independent of the programmed direction of the pin. Writing a "0" to the data register of the pins configured as outputs will cause a keypad interrupt to be generated if the row enable associated with that bit is set. 0 Row is not included in the keypad key press detect. 1 Row is included in the keypad key press detect.

38.6.2 Keypad Status Register (KPP_KPSR)

The Keypad Status Register reflects the state of the key press detect circuit. The KPP_KPSR register is byte- or half-word-addressable.

Address: 20B_8000h base + 2h offset = 20B_8002h

Bit	15	14	13	12	11	10	9	8
Read	0						KRIE	KDIE
Write								
Reset	0	0	0	0	0	1	0	0

Bit	7	6	5	4	3	2	1	0
Read	0				0	0	KPKR	KPKD
Write					KRSS	KDSC	w1c	w1c
Reset	0	0	0	0	0	0	0	0

KPP_KPSR field descriptions

Field	Description
15–10 Reserved	This read-only field is reserved and always has the value 0.
9 KRIE	Keypad Release Interrupt Enable. The software should ensure that the interrupt for a Key Release event is masked until it has entered the key pressed state, and vice versa, unless this activity is desired (as might be the case when a repeated interrupt is to be generated). The synchronizer chains are capable of being initialized to detect repeated key presses or releases. If they are not initialized when the corresponding event flag is cleared, false interrupts may be generated for depress (or release) events shorter than the length of the corresponding chain. 0 No interrupt request is generated when KPKR is set. 1 An interrupt request is generated when KPKR is set.
8 KDIE	Keypad Key Depress Interrupt Enable. Software should ensure that the interrupt for a Key Release event is masked until it has entered the key pressed state, and vice-versa, unless this activity is desired (as might be the case when a repeated interrupt is to be generated). The synchronizer chains are capable of being initialized to detect repeated key presses or releases. If they are not initialized when the corresponding event flag is cleared, false interrupts may be generated for depress (or release) events shorter than the length of the corresponding chain.

Table continues on the next page...

KPP_KPSR field descriptions (continued)

Field	Description
	0 No interrupt request is generated when KPKD is set. 1 An interrupt request is generated when KPKD is set.
7–4 Reserved	This read-only field is reserved and always has the value 0.
3 KRSS	Key Release Synchronizer Set. Self-clear bit. The Key release synchronizer is set by writing a logic one into this bit. Reads return a value of "0". 0 No effect 1 Set bits which sets keypad release synchronizer chain
2 KDSC	Key Depress Synchronizer Clear. Self-clear bit. The Key depress synchronizer is cleared by writing a logic "1" into this bit. Reads return a value of "0". 0 No effect 1 Set bits that clear the keypad depress synchronizer chain
1 KPKR	Keypad Key Release. The keypad key release (KPKR) status bit is set when all enabled rows are detected high after synchronization (the KPKR status bit will be set when cleared by a reset). The KPKR bit may be used to generate a maskable key release interrupt. The key release synchronizer may be set high by software after scanning the keypad to ensure a known state. Due to the logic function of the release and depress synchronizer chains, it is possible to see the re-assertion of a status flag (KPKD or KPKR) if it is cleared by software prior to the system exiting the state it represents. Reset value of register is "0" as long as reset is asserted. However when reset is de-asserted, the value of the register depends upon the external row pins and can become "1". 0 No key release detected 1 All keys have been released
0 KPKD	Keypad Key Depress. The keypad key depress (KPKD) status bit is set when one or more enabled rows are detected low after synchronization. The KPKD status bit remains set until cleared by the software. The KPKD bit may be used to generate a maskable key depress interrupt. If desired, the software may clear the key press synchronizer chain to allow a repeated interrupt to be generated while a key remains pressed. In this case, a new interrupt will be generated after the synchronizer delay (4 cycles of the low frequency reference clock elapses if a key remains pressed. This functionality can be used to detect a long key press. This allows detection of additional key presses of the same key or other keys. Due to the logic function of the release and depress synchronizer chains, it is possible to see the re-assertion of a status flag (KPKD or KPKR) if it is cleared by the software prior to the system exiting the state it represents. 0 No key presses detected 1 A key has been depressed

38.6.3 Keypad Data Direction Register (KPP_KDDR)

The bits in the KPP_KDDR control the direction of the keypad port pins. The upper eight bits in the register affect the pins designated as column strobes, while the lower eight bits affect the row sense pins. Setting any bit in this register configures the corresponding pin as an output. Clearing any bit in this register configures the corresponding port pin as an input. For the Keypad Row DDR, an internal pull-up is enabled if the corresponding bit is clear. This register is cleared by a reset, configuring all pins as inputs. The KPP_KDDR register is byte- or half-word addressable.

NOTE

When a pin is used as row pin for keypad purposes, all corresponding pull-ups should be enabled at the upper level (for example, IOMUX) when the bit in KRDD is cleared.

Address: 20B_8000h base + 4h offset = 20B_8004h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	KCDD								KRDD							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KPP_KDDR field descriptions

Field	Description
15–8 KCDD	Keypad Column Data Direction Register. Setting a bit configures the corresponding COL*n* pin as an output (where *n* = 7 through 0). 0 **INPUT** — COLn pin is configured as an input. 1 **OUTPUT** — COLn pin is configured as an output.
KRDD	Keypad Row Data Direction. Setting a bit configures the corresponding ROW*n* pin as an output (where *n* = 7 through 0). 0 **INPUT** — ROWn pin configured as an input. 1 **OUTPUT** — ROWn pin configured as an output.

38.6.4 Keypad Data Register (KPP_KPDR)

This 16-bit register is used to access the column and row data. Data written to this register is stored in an internal latch, and for each pin configured as an output, the stored data is driven onto the pin. A read of this register returns the value on the pin for those bits configured as inputs. Otherwise, the value read is the value stored in the register.

The KPP_KPDR register is byte- or half-word addressable. This register is not initialized by a reset. Valid data should be written to this register before any bits are configured as outputs.

Address: 20B_8000h base + 6h offset = 20B_8006h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	KCD								KRD							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KPP_KPDR field descriptions

Field	Description
15–8 KCD	Keypad Column Data. A read of these bits returns the value on the pin for those bits configured as inputs. Otherwise, the value read is the value stored in the register. 0 Read/Write "0" from/to column ports 1 Read/Write "1" from/to column ports
KRD	Keypad Row Data. A read of these bits returns the value on the pin for those bits configured as inputs. Otherwise, the value read is the value stored in the register. 0 Read/Write "0" from/to row ports 1 Read/Write "1" from/to row ports

Chapter 39
LVDS Display Bridge (LDB)

39.1 Overview

The LVDS Display Bridge (LDB) connects the IPU (Image Processing Unit) to an External LVDS Display Interface.

The purpose of the LDB is to support flow of synchronous RGB data from the IPU to external display devices through the LVDS interface. This support covers all aspects of these activities:

- Connectivity to relevant devices - Displays with LVDS receivers.
- Arranging the data as required by the external display receiver and by LVDS display standards.
- Synchronization and control capabilities.

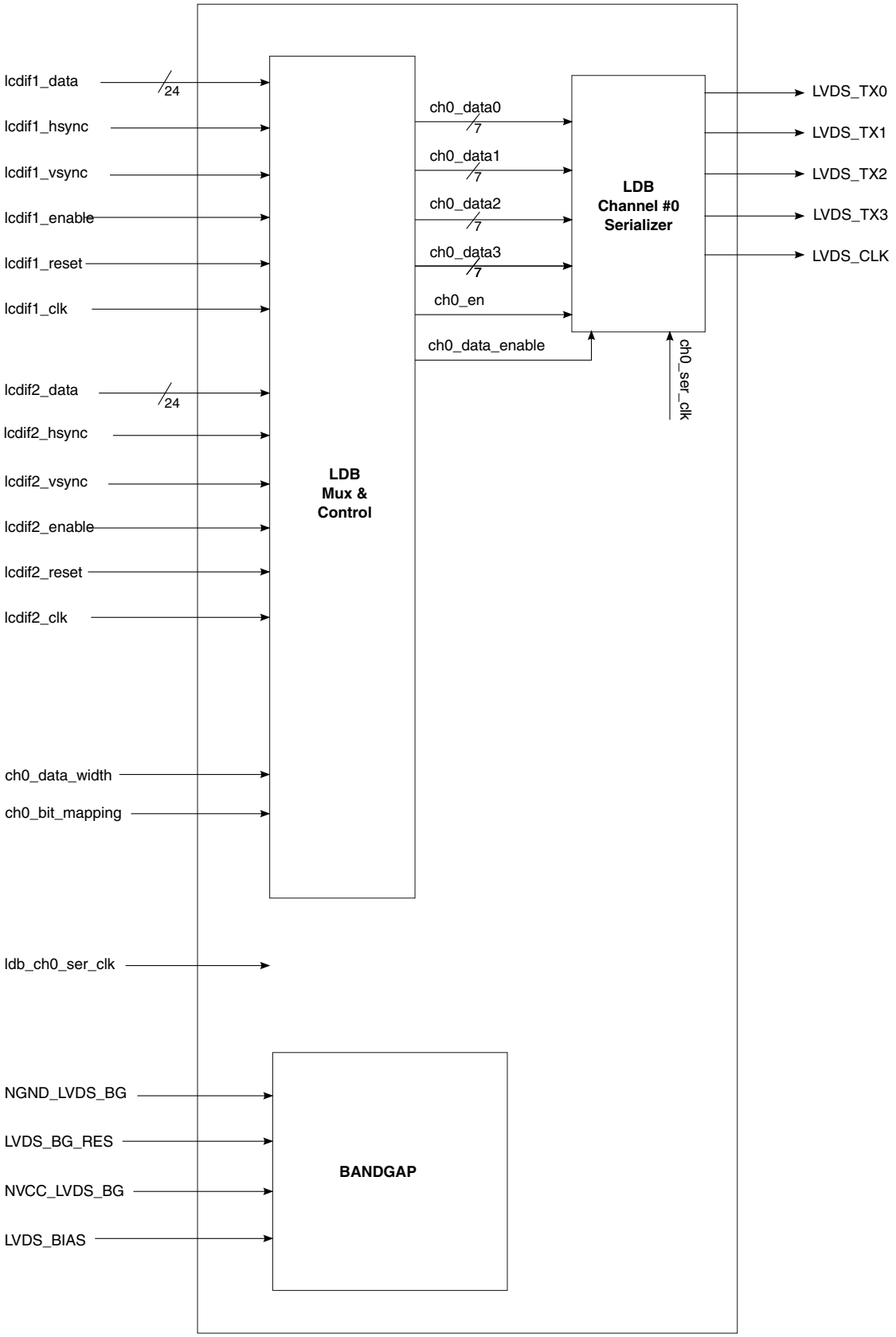


Figure 39-1. LDB Block Diagram

Table 39-1. LDB - Block Description

Block	Description
LDB Mux and Control	Gets control signals from SoC and determines parameters of LDB
Channel Serializers	LDB has 2 channel serializers. The serializer does parallel to serial conversion 7:1 to 3 or 4 data lines
Bandgap	Provides reference current to the LVDS I/O pads.

Table 39-2. LDB IP Parametric Table

Name	IPU
Function	Connectivity to displays with LVDS interface
External I/O Pins Those are LVDS I/O pads	LVDS Display port: 2 channels, consists of: • 1 clock pair • 4 data pairs Each pair contains - LVDS special differential pad (PadP, PadM). total of 20 I/O pads.
SoC Buses	None. Only configuration signals.
Interrupts	None
DMA Requests	None
Number of instantiations	1
Clock sources and range	IPU_DI0_CLK, IPU_DI1_CLK- Display interface clock: 20-170 MHz DI0_SERIAL_CLK, DI1_SERIAL_CLK - Serializer clock: 140-595 MHz

39.1.1 Relevant Standards

Below are the relevant standards to LDB:

1. ANSI EIA-644-A. Electrical Characteristics of Low Voltage Differential Signaling (LVDS) Interface Circuits.
2. SPWG Notebook Panel Specification (V3.8 from 03/2007) http://www.spwg.org/specifications.htm.
3. PSWG standards (Panel Standardization Working Group) - set of standards for panels using LVDS. All are available from http://www.vesa.org.
4. DISM Standard JEIDA-59-1999

39.2 External Signals

The following table describes the external signals of LDB:

Table 39-3. LDB External Signals

Signal	Description	Pad	Mode	Direction
LVDS0_CLK_N	LVDS0 Negative Clock Signal	LVDS0_CLK_N	No Muxing	I/O
LVDS0_CLK_P	LVDS0 Positive Clock Signal	LVDS0_CLK_P	No Muxing	I/O
LVDS0_DATA[3:0]_N	LVDS0 Negative Data Signals	LVDS0_TX[3:0]_N	No Muxing	I/O
LVDS0_DATA[3:0]_P	LVDS0 Positive Data Signals	LVDS0_TX[3:0]_P	No Muxing	I/O
LVDS1_CLK_N	LVDS1 Negative Clock Signal	LVDS1_CLK_N	No Muxing	I/O
LVDS1_CLK_P	LVDS1 Positive Clock Signal	LVDS1_CLK_P	No Muxing	I/O
LVDS1_DATA[3:0]_N	LVDS1 Negative Data Signals	LVDS1_TX[3:0]_N	No Muxing	I/O
LVDS1_DATA[3:0]_P	LVDS1 Positive Data Signals	LVDS1_TX[3:0]_P	No Muxing	I/O

39.3 Clocks

A table with the LDB Clock Sources can be found here.

Table 39-4. LDB Clock Sources

Name	Symbol	Source	Rate	Comments
IPU DI0 interface pixel clock	IPU_DI0_CLK	Clock control Module	Up to 170 MHz	See note below [1]
IPU DI1 interface pixel clock	IPU_DI1_CLK	Clock control Module	Up to 170 MHz	This input also goes to IPU DI1 as input. See note below
CH0 interface serializer clock	DI0_SERIAL_CLK	Clock control Module	Up to 595 MHz	This is x7 the rate of the DI0 interface pixel clock. See note below.
CH1 interface serializer clock	DI1_SERIAL_CLK	Clock control Module	Up to 595 MHz	This is x7 the rate of the DI1 interface pixel clock. See note below

1. In case of single-channel or separate-channels use-case, the IPU DI_CLK is identical to the LVDS DI_CLK. In case of dual-channel use-case, the IPU DI_CLK has x2 higher frequency than that of the LVDS DI_CLK. Still both need to be synchronized

39.4 Input and Output Ports

39.4.1 Input Parallel Display Ports

One or Two parallel RGB input ports are supported (configurable). Only synchronous access mode is supported.

Each RGB data interface contains the following:

- RGB Data of 18 or 24 bits
- Pixel clock
- Control signals: HSYNC, VSYNC, DE, and 1 additional optional general purpose control.

Total of up to 28 bits per data interface are transferred per pixel clock cycle.

Rates supported:

- Overall: LDB supports rates needed by WUXGA 16:10 aspect ratio (1920 x 1200 @ 60 frames per second, data rate supported up to 170 MHz)
- For single input data interface case: Up to 170 MHz pixel clock (WUXGA 1920x1200)
- For dual input data interface case: Up to 85 MHz per interface. (WXGA 1366x768 @ 60 frames per second, 35% blanking).

39.4.2 Output LVDS Ports

There is 2 LVDS channels. The output of each is used to communicate RGB data and controls to external LCD displays.

The LVDS ports may be used as follows:

- Single channel output
- Dual channel output (one input source, two channels outputs for two displays)
- Split channel output (one input source, splitted to 2 channels on output)
- Separate 2 channel output (2 input sources from IPU).

The output LVDS port complies to the EIA-644-A standard.

39.5 Processing

LDB data processing stages are as follows:

- Receive input data from 1 or 2 (configurable) parallel input interfaces. 18/24 RGB data + up to 4 controls and map them to LVDS channels.
  - If needed (dual-channel) split the input bus to two half-rate busses.
- Re-arrange the input data according to channel configuration, and muxing scheme.
- Serialize the 22/28 bit input bus (per channel) on 3-4 output serial data lines (7:1)

39.5.1 Mapping of Input Data Busses

Mapping of Parallel input interfaces (DI0, DI1) to output LVDS channels (Channel 0, Channel 1). See Table 39-5.

Table 39-5. Channel Mapping

Use Case	LVDS Channel 0	LVDS Channel 1
Single Channel DI0	DI0	Disabled
Single Channel DI1 on Channel 1	Disabled	DI1
Separate Channels	DI0	DI1
Dual Channels DI0	DI0	DI0
Dual Channels DI1	DI1	DI1
Split Channel DI0	DI0 (first pixel)	DI0 (second pixel)
Split Channel DI1	DI1 (first pixel)	DI1 (second pixel)

39.5.2 Bit Mapping

LDB supports two mapping standards:

- SPWG mapping
- JEIDA mapping

Table 39-6. SPWG/PSWG/VESA 18/24 bpp Data Mapping

Serializer input	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
LVDSn_DATA0	G0	R5	R4	R3	R2	R1	R0
LVDSn_DATA1	B1	B0	G5	G4	G3	G2	G1
LVDSn_DATA2	DE	VS	HS	B5	B4	B3	B2
LVDSn_DATA3 (for 24 bpp only)	CTL	B7	B6	G7	G6	R7	R6

Table 39-7. JEIDA 24bpp Data Mapping

Serializer input	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
LVDSn_DATA0	G2	R7	R6	R5	R4	R3	R2
LVDSn_DATA1	B3	B2	G7	G6	G5	G4	G3
LVDSn_DATA2	DE	VS	HS	B7	B6	B5	B4
LVDSn_DATA3	CTL	B1	B0	G1	G0	R1	R0

NOTE

Several options of control usage can be available. some display devices use only DE, some others use all 3 controls, some use only HS, VS. "CTL" is an optional general purpose control which is usually unused by display.

39.6 LDB Memory Map/Register Definition

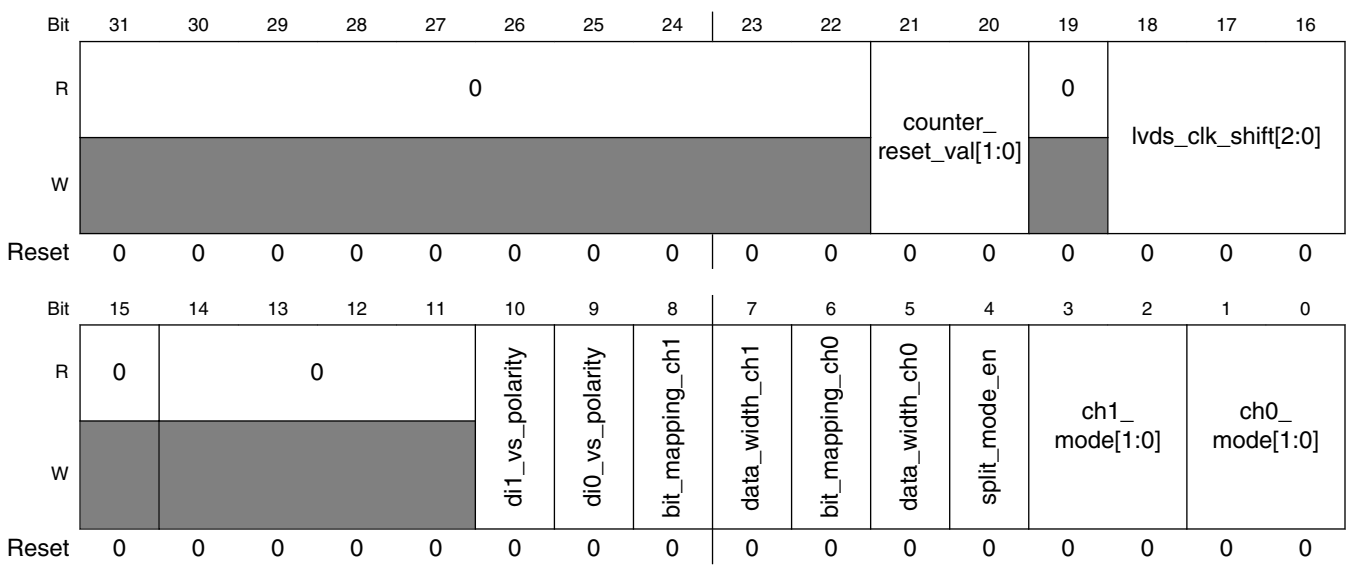
LDB memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
20E_0008	LDB Control Register (LDB_CTRL)	32	R/W	0000_0000h	39.6.1/3565

39.6.1 LDB Control Register (LDB_CTRL)

The register is implemented in the IOMUX Controller block (IOMUXC), as the register IOMUXC_GPR2.

Address: 20E_0008h base + 0h offset = 20E_0008h



LDB_CTRL field descriptions

Field	Description
31–22 Reserved	This read-only field is reserved and always has the value 0.
21–20 counter_reset_val[1:0]	Reset value for the LDB counter which determines when the shift registers are loaded with data. **NOTE:** Used for debug purposes only. In normal functional operation must be '00' 00 Reset value is 5 01 Reset value is 3 10 Reset value is 4 11 Reset value is 6
19 Reserved	This read-only field is reserved and always has the value 0.
18–16 lvds_clk_shift[2:0]	Shifts the LVDS output clock in relation to the data. **NOTE:** Used for debug purposes only. In normal functional operation must be '000' 000 Output clock is '1100011' (normal operation) 001 Output clock is '1110001' 010 Output clock is '1111000' 011 Output clock is '1000111' 100 Output clock is '0001111' 101 Output clock is '0011111' 110 Output clock is '0111100' 111 Output clock is '1100011'
15 Reserved	This read-only field is reserved and always has the value 0.
14–11 Reserved	This read-only field is reserved and always has the value 0.
10 di1_vs_polarity	Vsync polarity for IPU's DI1 interface. 0 ipu_di1_vsync is active high. 1 ipu_di1_vsync is active low.
9 di0_vs_polarity	Vsync polarity for IPU's DI0 interface. 0 ipu_di0_vsync is active high. 1 ipu_di0_vsync is active low.
8 bit_mapping_ch1	Data mapping for LVDS channel 1. 0 Use SPWG standard. 1 Use JEIDA standard.
7 data_width_ch1	Data width for LVDS channel 1. **NOTE:** This bit must be set when using JEIDA standard (bit_mapping_ch1 is set) 0 Data width is 18 bits wide (lvds1_tx3 is not used) 1 Data width is 24 bits wide.
6 bit_mapping_ch0	Data mapping for LVDS channel 0. 0 Use SPWG standard. 1 Use JEIDA standard.

Table continues on the next page...

LDB_CTRL field descriptions (continued)

Field	Description
5 data_width_ch0	Data width for LVDS channel 0. **NOTE:** This bit must be set when using JEIDA standard (bit_mapping_ch0 is set) 0 Data width is 18 bits wide (lvds0_tx3 is not used) 1 Data width is 24 bits wide.
4 split_mode_en	Enable split mode. **NOTE:** In this mode both channels should be enabled and working with the same DI (ch0_mode and ch1_mode should both be either '01' or '11') 0 Split mode is disabled. 1 Split mode is enabled.
3–2 ch1_mode[1:0]	LVDS channel 1 operation mode 00 Channel disabled. 01 Channel enabled, routed to DI0 10 Channel disabled. 11 Channel enabled, routed to DI1.
ch0_mode[1:0]	LVDS channel 0 operation mode 00 Channel disabled. 01 Channel enabled, routed to DI0 10 Channel disabled. 11 Channel enabled, routed to DI1.

Chapter 40
MIPI - Camera Serial Interface Host Controller (MIPI_CSI)

40.1 Overview

The CSI-2 Host Controller is a digital core that implements all protocol functions defined in the MIPI CSI-2 Specification, providing an interface between the System and the MIPI D-PHY, allowing the communication with a MIPI CSI-2 compliant Camera Sensor.

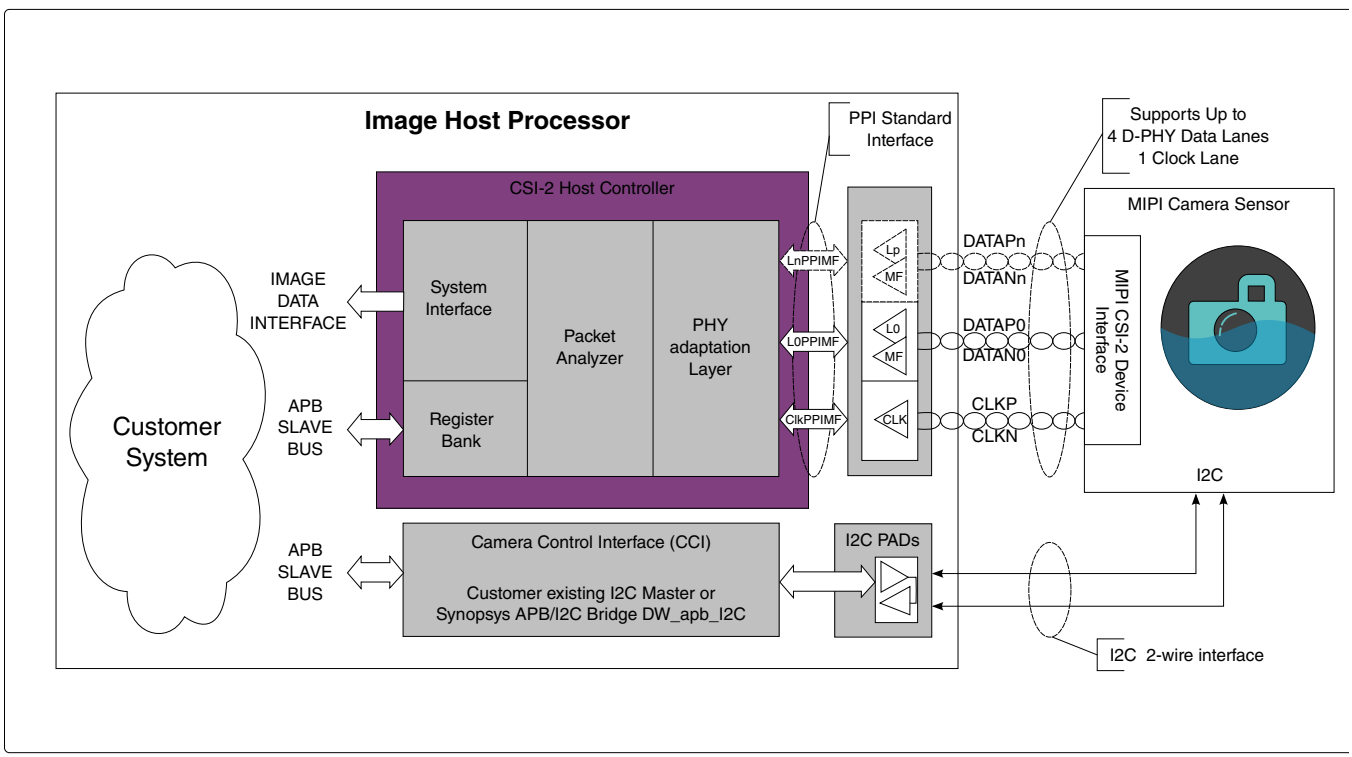


Figure 40-1. Block Diagram presenting the CSI-2 Host Controller function.

40.2 Features

The MIPI CSI-2 Host Controller supports the following features:

- Compliant with MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2), Version 1.00 - 29 November 2005
- Interface with MIPI D-PHY following PHY Protocol Interface (PPI), as defined in MIPI Alliance Specification for D-PHY, Version 1.00.00 - 14 May 2009
- Supports up to 4 D-PHY Rx Data Lanes
- Dynamically configurable multi-lane merging
- Long and Short packet decoding
- Timing accurate signaling of Frame and Line synchronization packets
- Support for several frame formats such as:
  - General Frame or Digital Interlaced Video with or without accurate sync timing
  - Data type (Packet or Frame level) and Virtual Channel interleaving
- 32-bit Image Data Interface delivering data formatted as recommended in CSI-2 Specification;
- Supports all primary and secondary data formats:
  - RGB, YUV and RAW color space definitions
  - From 24-bit down to 6-bit per pixel
  - Generic or user-defined byte-based data types
- Error detection and correction:
  - PHY level
  - Packet level
  - Line level
  - Frame level

40.3 Architecture

The following figure presents the overall architecture of the CSI-2 Host Controller. The main blocks are the following:

- PHY Adaptation Layer, which is responsible for managing the D-PHY interface, including PHY error handling;
- Packet Analyzer, where data lane merging is implemented if required, together with header decoding, error detection and correction, frame size error detection and CRC error detection;
- Image Data Interface. This block separates CSI-2 packet header information and reorders data according to memory storage format, and it also generates timing

accurate video synchronization signals. Several error detections are also performed at frame-level and line-level;

- Register Bank, accessible through a standard AMBA-APB slave interface, providing access to the CSI-2 Host Controller registers for configuration and control. There is also a fully programmable interrupt generator to inform the system upon certain events;

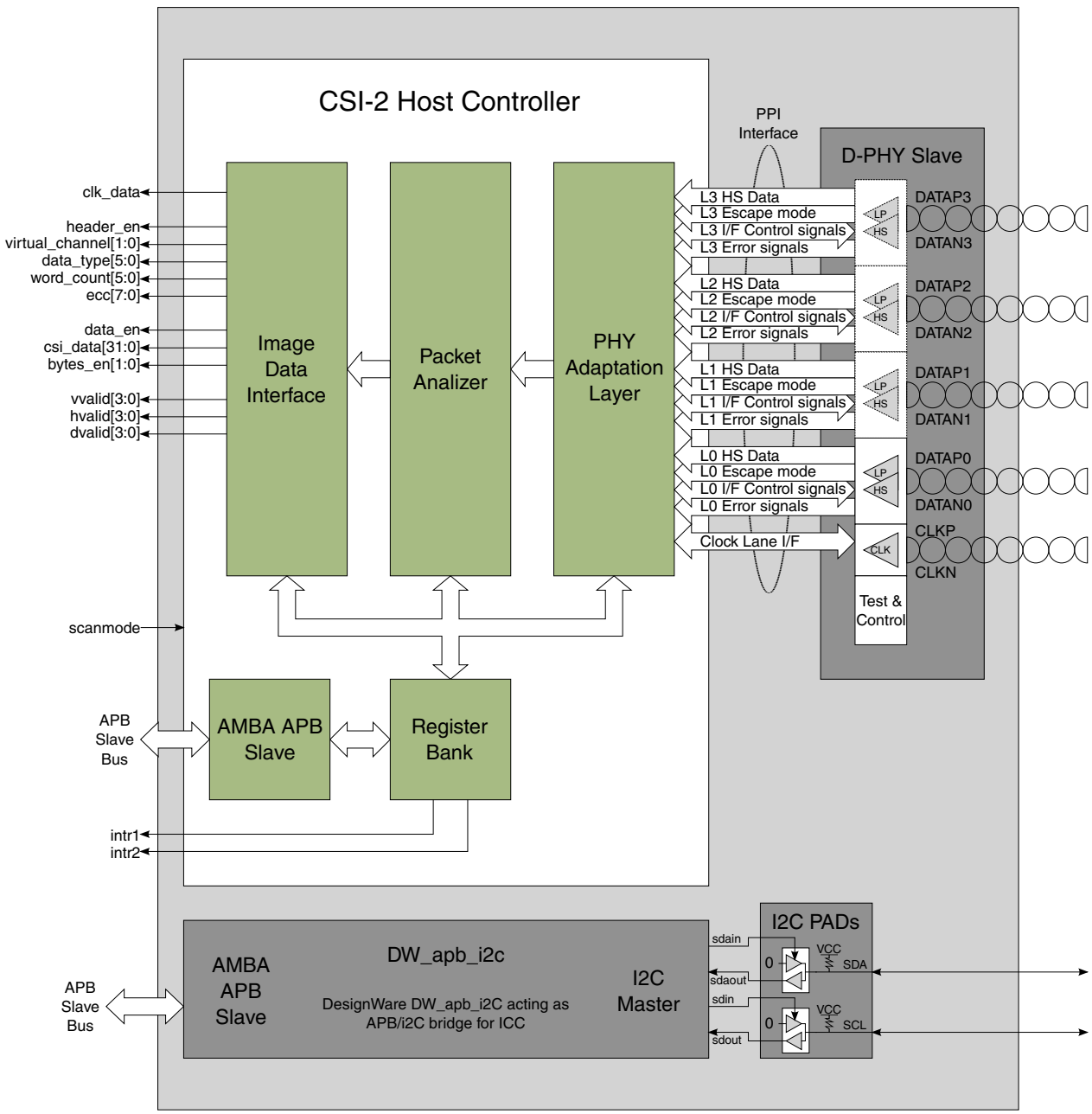


Figure 40-2. Architecture of the CSI-2 Host Controller.

40.3.1 Startup Sequence

The following information is provided as a guideline to allow a safe startup of the system operation.

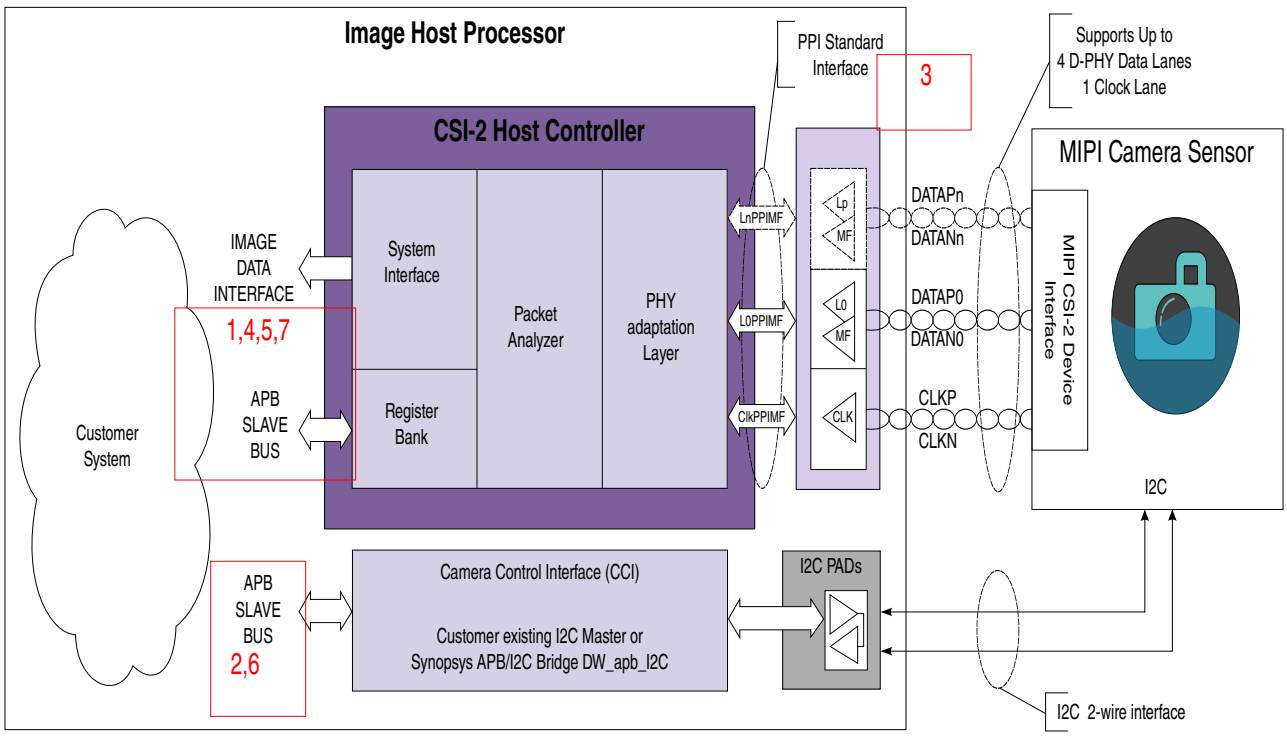


Figure 40-3. Block diagram presenting the recommended Startup Sequence.

1. Deassert CSI2 presetn signal (global reset);
2. **Configure MIPI Camera Sensor to have all Tx lanes in PL-11 state (STOPSTATE) if required** - According to D-PHY Specification the D-PHY master should be initialized at LP-11 state (STOPSTATE), nevertheless a CCI command may be required to switch-on the MIPI interface;
3. **D-PHY initialization** - Access the D-PHY programming interface to initialize and program the D-PHY according to the selected operating mode of the D-PHY. This is D-PHY dependent, and this programming should be carried out according to the D-PHY databook;
4. **CSI2 Controller programming** - Program the CSI2 Host Controller registers according to the operating mode required:
   - **Number of Lanes (register N_LANES);**
   - **Deassert PHY shutdown (register PHY_SHUTDOWNZ);**
   - **Deassert PHY reset (register PHY_RSTZ);**
   - **Deassert CSI reset (register CSI2_RESETN);**

- **(Optional) Program Data IDs for matching error reporting (registers DATA_IDS_1 and DATA_IDS_2);**
- **(Optional) Program the interrupt masks (registers MASK1 and MASK2);**

5. **CSI2 Controller programming** - Read the PHY status register (PHY_STATE) to confirm that all data and clock lanes of the D-PHY are in Stop State (i.e. ready to receive data);
6. **Configure the MIPI Camera Sensor** - Access Camera Sensor using CCI interface to initialize and configure the Camera Sensor to start transmitting a clock on the D-PHY clock lane;
7. **CSI2 Controller programming** - Read the PHY status register (PHY_STATE) to confirm that the D-PHY is receiving a clock on the D-PHY clock lane;

NOTE

Additional steps may be required to correctly configure the specific D-PHY and MIPI Camera Sensor(s) that are part of the system, as well as any other requirements that is integral part of the relevant MIPI Specifications.

40.3.2 Interrupt mechanism

The CSI-2 Host Controller provides an interrupt mechanism that is usable mostly for monitoring errors and debugging.

There are two interrupt signals, *intr1* and *intr2*, which are synchronous with the AMBA-APB clock signal. Registers MASK1 and MASK2 are used to select which bits of registers ERR1 and ERR2 are able to generate interrupts by asserting signals *intr1* and *intr2*, respectively. Both ERR1 and ERR2 will always contain the information of events, irrespective of the state of MASK1 and MASK2. Registers ERR1 and ERR2 will self-clear after a read access. Interrupt signals *intr1* and *intr2* will be de-asserted upon read access of register ERR1 and ERR2, respectively.

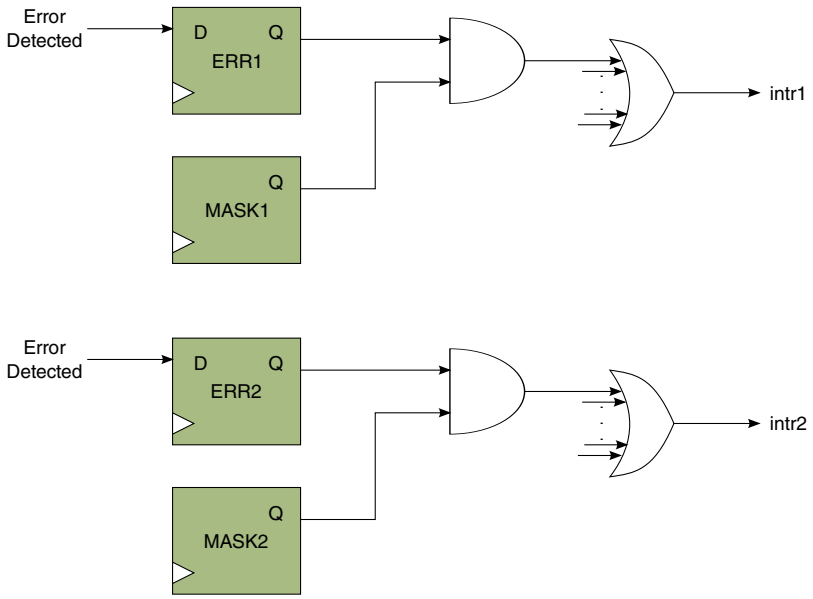


Figure 40-4. Block Diagram presenting the Interrupt mechanism.

40.3.3 Signals

Table 40-1. D-PHY External interface signals

Pin Name	Width	Direction	Description
AVDD	1	Input	D-PHY Analog power supply.
VDD	1	Input	D-PHY Digital power supply.
AVDDREF	1	Input	D-PHY Analog supply for reference generator.
AGND	1	Input	D-PHY Analog supply ground return.
VSS	1	Input	D-PHY Digital supply ground return.
AGNDREF	1	Input	D-PHY Analog supply ground return for reference generator.
REXT	1	Input	D-PHY External resistor connection (REXT and REXTV should be shorted at the chip pad connection).
REXTV	1	Input	D-PHY External resistor connection (REXT and REXTV should be shorted at the chip pad connection)
CSI2_CLKP	1	Input	D-PHY Positive D-Phy differential clock line Receiver input
CSI2_CLKN	1	Input	D-PHY Negative D-Phy differential clock line Receiver input
CSI2_DATAP0	1	Input	D-PHY Positive D-Phy differential data line Receiver input , Lane 0
CSI2_DATAN0	1	Input	D-PHY Negative D-Phy differential data line Receiver input , Lane 0

Table continues on the next page...

Table 40-1. D-PHY External interface signals (continued)

CSI2_DATAP1	1	Input	D-PHY Positive D-Phy differential data line Receiver input , Lane 1
CSI2_DATAN1	1	Input	D-PHY Negative D-Phy differential data line Receiver input , Lane 1
CSI2_DATAP2	1	Input	D-PHY Positive D-Phy differential data line Receiver input , Lane 2
CSI2_DATAN2	1	Input	D-PHY Negative D-Phy differential data line Receiver input , Lane 2
CSI2_DATAP3	1	Input	D-PHY Positive D-Phy differential data line Receiver input , Lane 3
CSI2_DATAN3	1	Input	D-PHY Negative D-Phy differential data line Receiver input , Lane 3

40.4 Timing Interfaces

40.4.1 Image Data Interface

At the Image Data Interface, signal *header_en* is used to indicate that new data is being transferred. It rises when a new packet becomes available at the interface and falls as soon as a packet finishes. Between two consecutive packets, there is always a fall and a rise of *header_en*, since the CSI-2 transmitter must enter Low-Power State between an End of Transmission and the following Start of Transmission.

An example of transferring two short packets is presented in the following figure. The fields of the header packet become available simultaneously to the rise of *header_en*, which falls after one clock cycle, as no new data is to be transferred. This is the behavior of the circuit, independently of the number of active lanes.

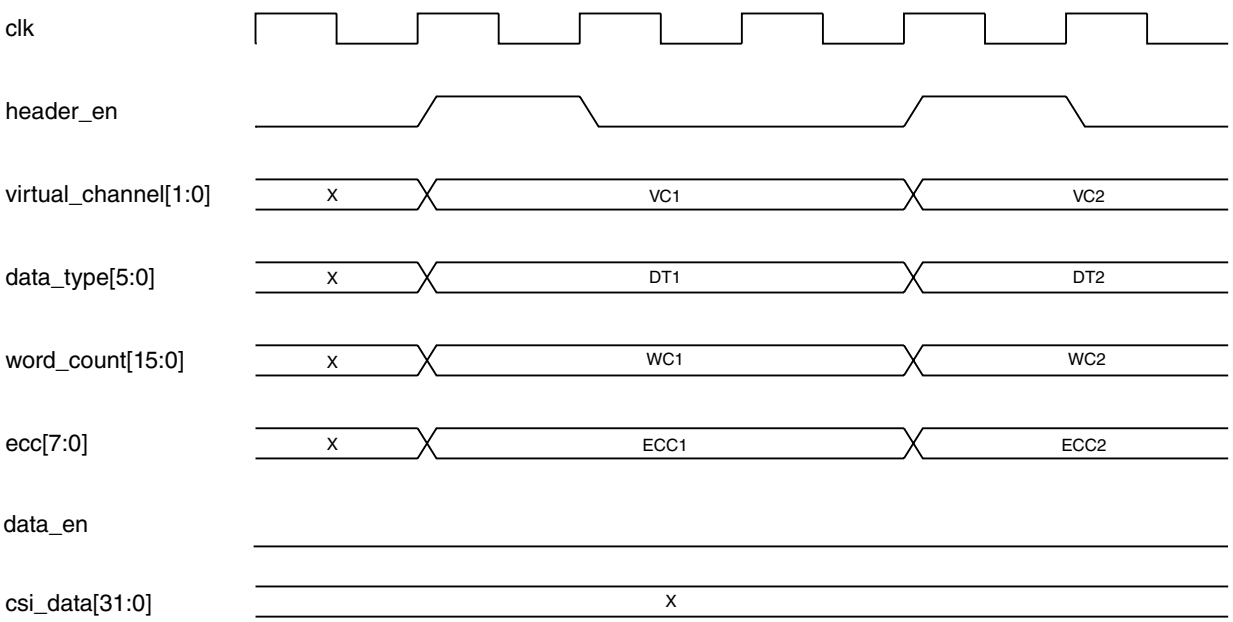


Figure 40-5. Image Data Interface example with two short packets.

When transferring long packets, the data provided by the payload on all lanes is transferred separately on *csi_data* bus, while the header fields remain stable until the transfer of the packet has completed.

Signal *data_en* is used to indicate that a new 32-bit word is available in *csi_data*, and it can only be set if *header_en* is also set. The following Figure shows an example of transferring a long packet received from 4 data lanes. Since a new 32-bit word is transferred at each clock cycle, signal *data_en* remains set until all data is transferred.

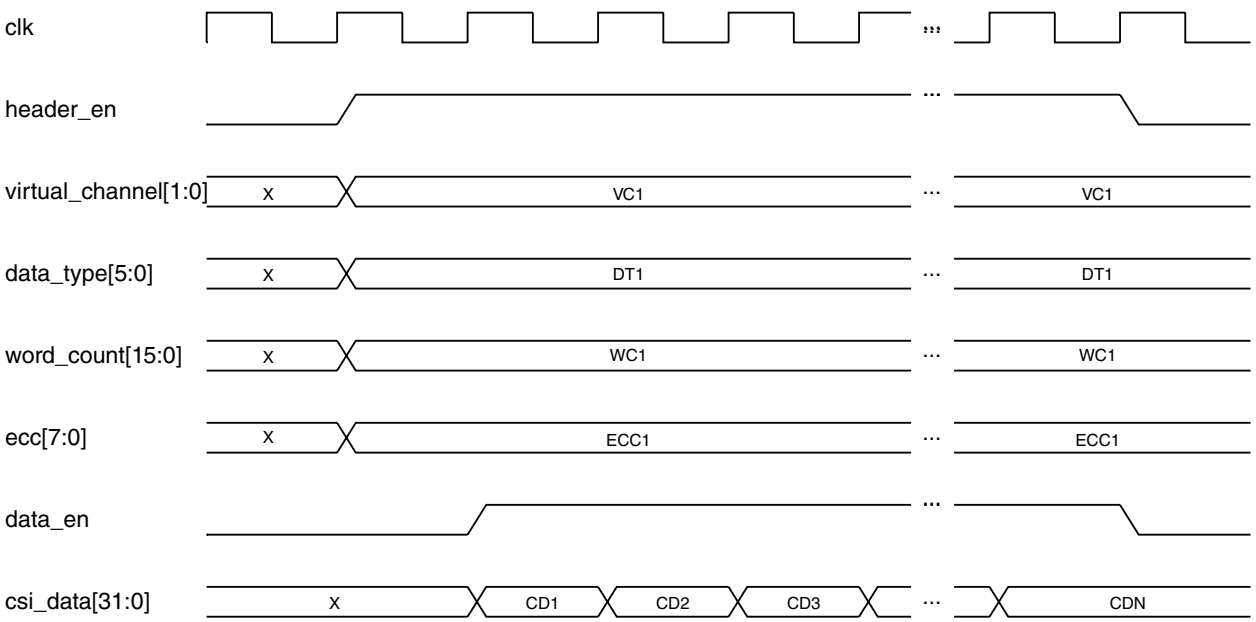


Figure 40-6. Timing interface for a long packet, receiving data from 4 lanes.

If less than 4 lanes are used, a 32-bit word might take more than one clock cycle to be received. In this case, *data_en* is set only when a new word has been fully received and becomes available at the interface. As shown in the following figure, when data is received from two lanes a new word is released every two clock cycles. Both *header_en* and *data_en* return to 0 as soon as all the data has been transferred.

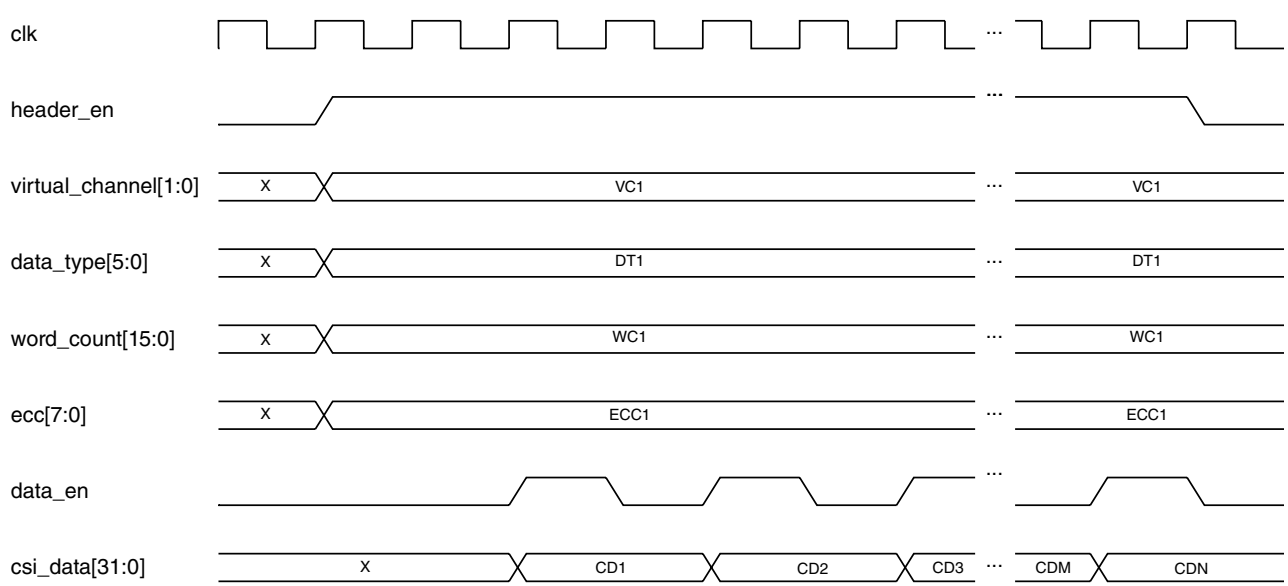


Figure 40-7. Timing interface for a long packet, receiving data from 2 lanes.

If data is transferred using only 1 lane, a new 32-bit word becomes available every four clock cycles. In the case of 3 lanes, a new word is available at clock cycle except every fourth.

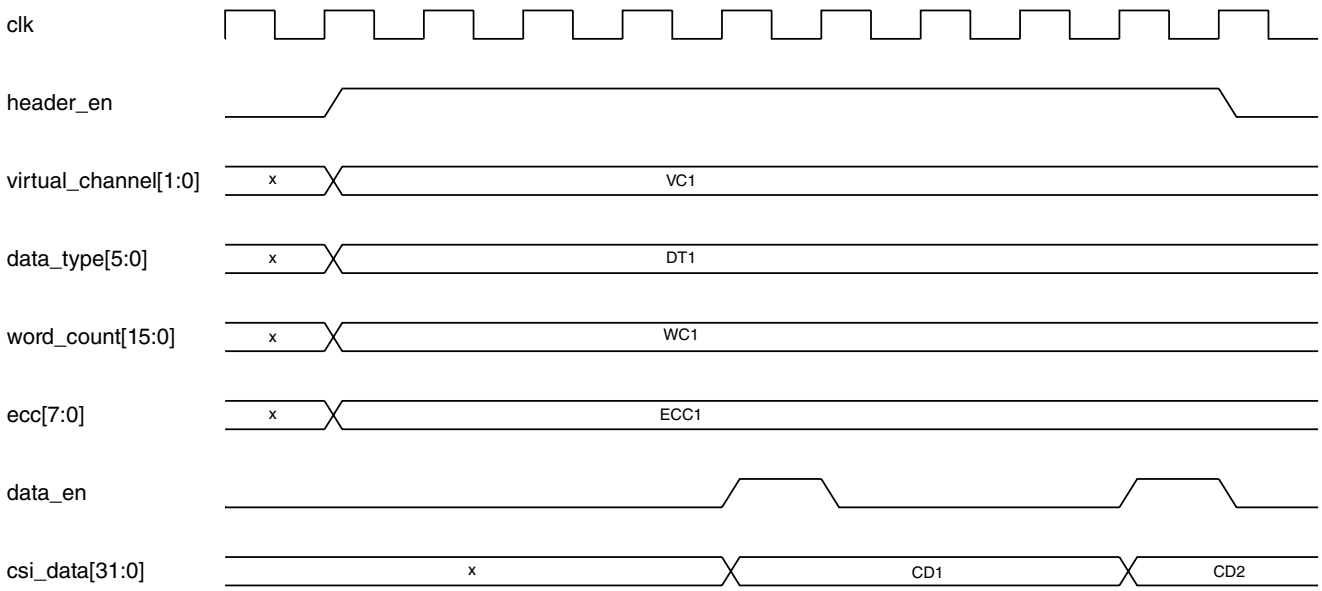


Figure 40-8. Timing interface for a long packet, receiving data from 1 lane.

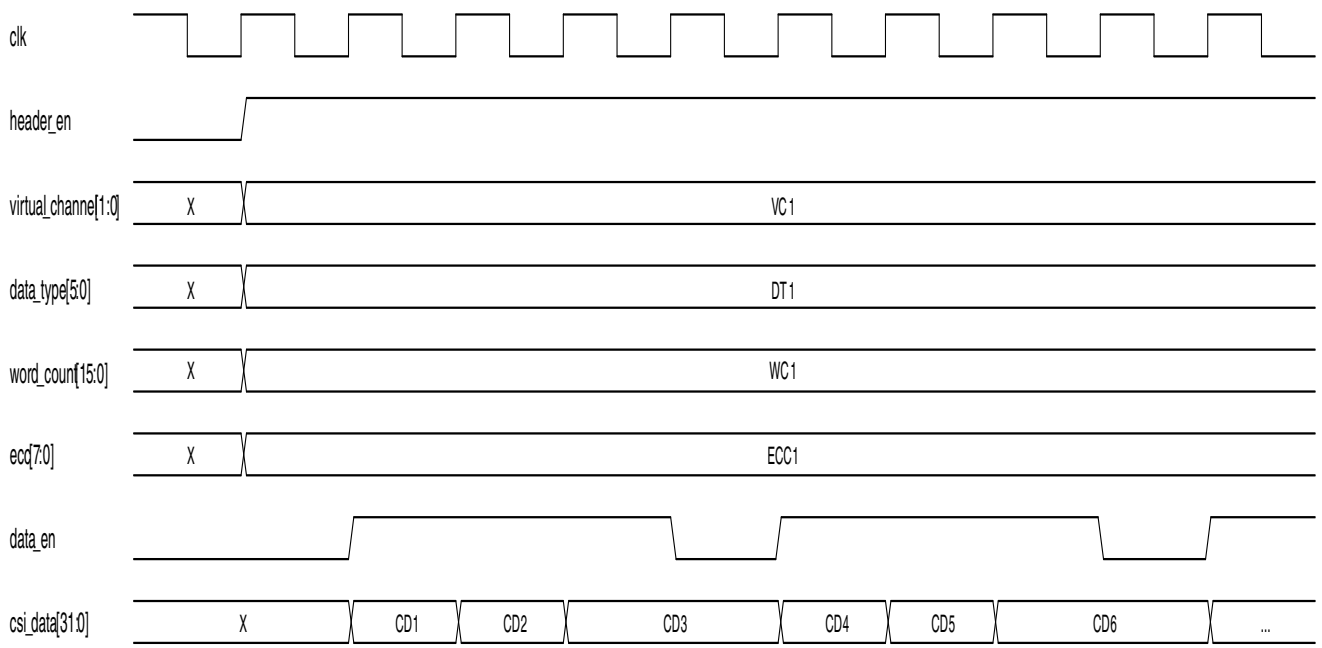


Figure 40-9. Timing interface for a long packet, receiving data from 3 lanes.

Signals *dvalid*, *hvalid* and *vvalid*, are used provide information about Frame Start, Frame End, Line Start and Line End packets. These signals are included in the interface for synchronization purposes.

dvalid is used to indicate when data is being transferred, excluding header information. Data sent through Blank or Null packets do not activate *dvalid*.

hvalid is set on detection of Line Start packets and unset by Line End ones. These packets can be used as a reference for synchronization, even though relevant data can be more or less delayed between them, as it might be surrounded by Blanking Periods, before and after the data. Because Line Start and Line End packets are optional, in case they are not available, *hvalid* adopts the same behavior of *dvalid*, and is not activated by Blank or Null packets. In the case a new Line Start packet is received without a Line End packet being received, a pulse will be generated in *hvalid* to signal that a new Line Start packet was received.

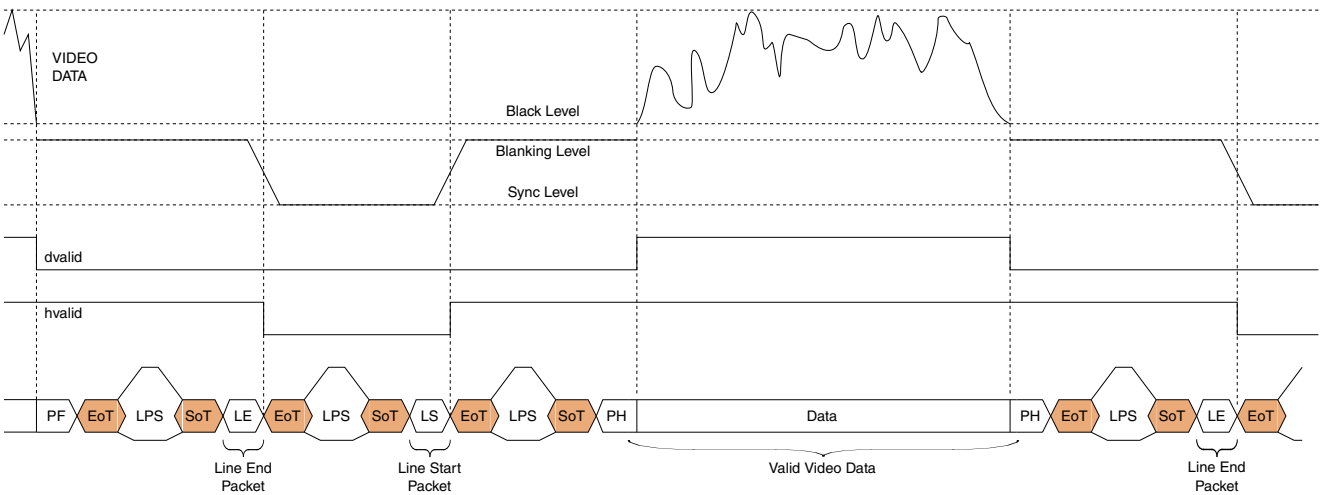


Figure 40-10. HVALID synchronization signal.

vvalid is set on detection of Frame Start packets and unset by Frame End ones. As these packets are mandatory, *vvalid* is always reliable for video applications to synchronize frame updating. In the case a new Frame Start packet is received without a Frame End packet being received, a pulse will be generated in *hvalid* to signal that a new Frame Start packet was received.

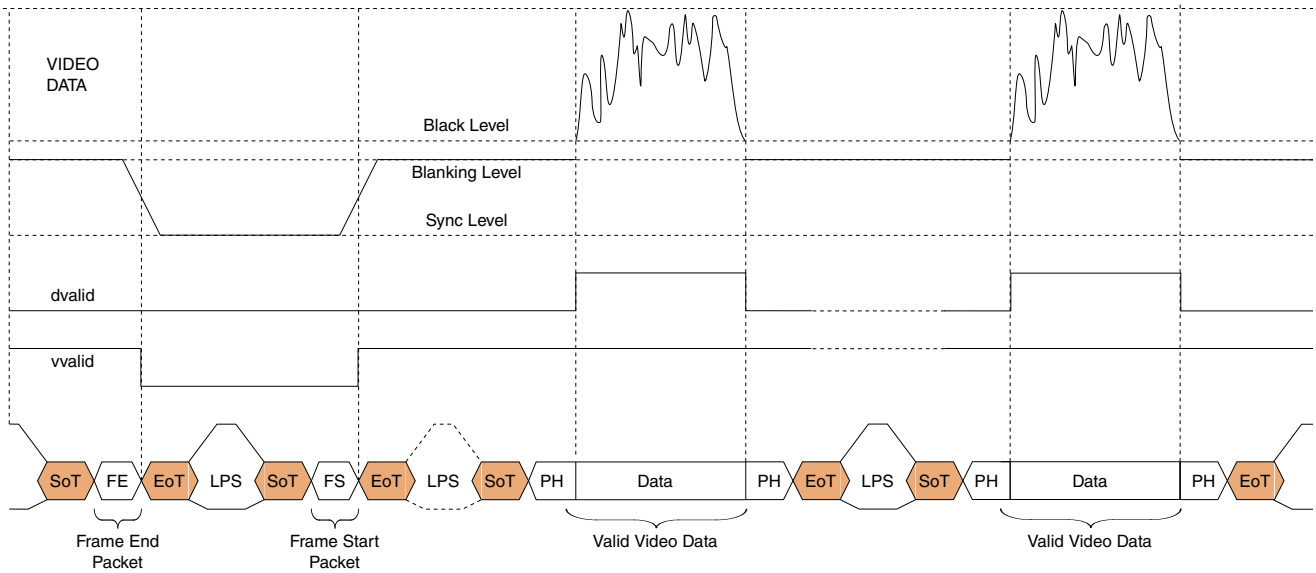


Figure 40-11. VVALID synchronization signal.

40.5 Payload Data Output Format

The Image Data Interface delivers payload data in a common data storage format, as suggested in CSI-2 Specification and described below.

The following sections describe how different data formats are transferred in output bus *csi_data*.

40.5.1 General/Arbitrary Data Reception

In the generic case and for arbitrary data, the first byte of payload data transmitted maps the least significant byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the most significant byte of the 32-bit memory word.

The following Figure illustrates the generic CSI-2 byte to 32-bit memory word mapping rule.

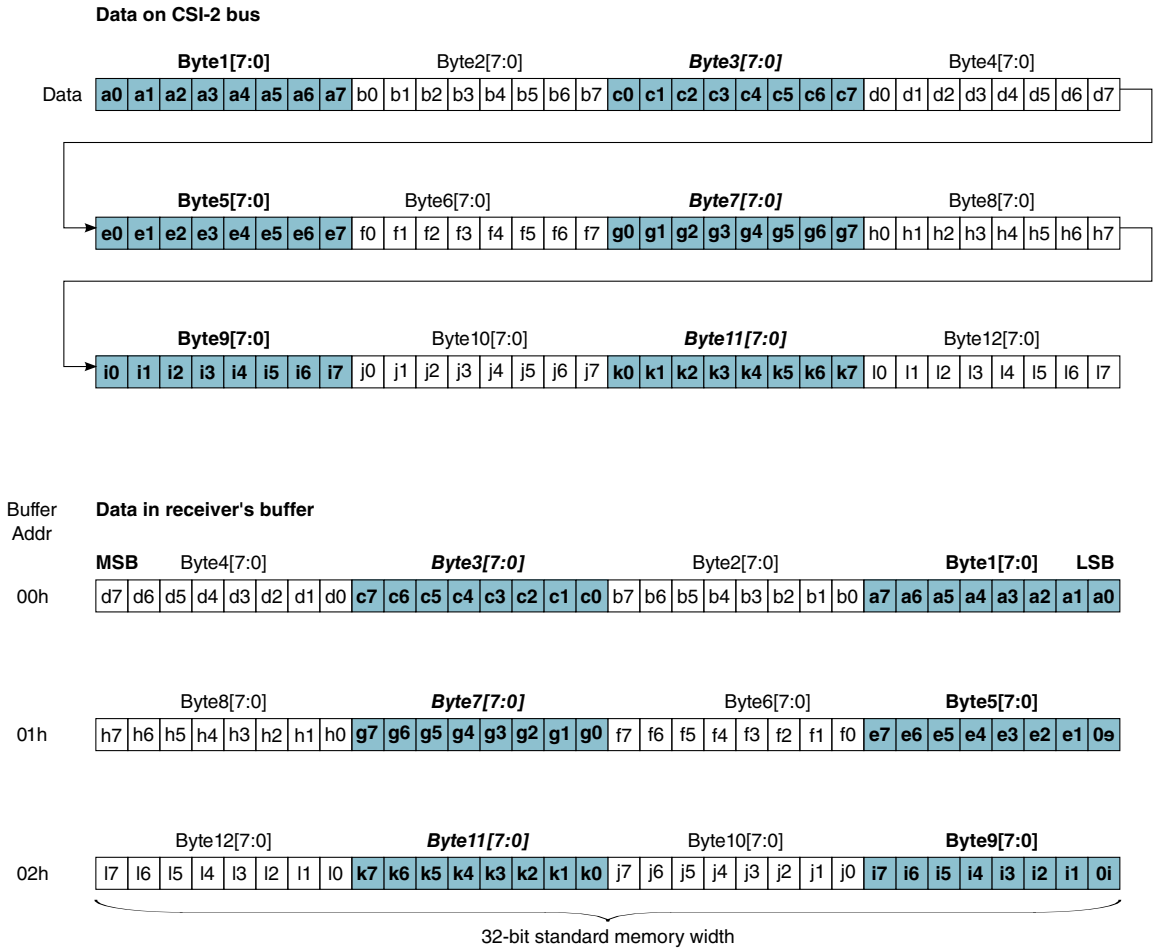


Figure 40-12. General/Arbitrary Data Reception.

40.5.2 RGB888 Data Reception

The RGB888 data format byte to 32-bit memory word mapping follows the generic CSI-2 rule.

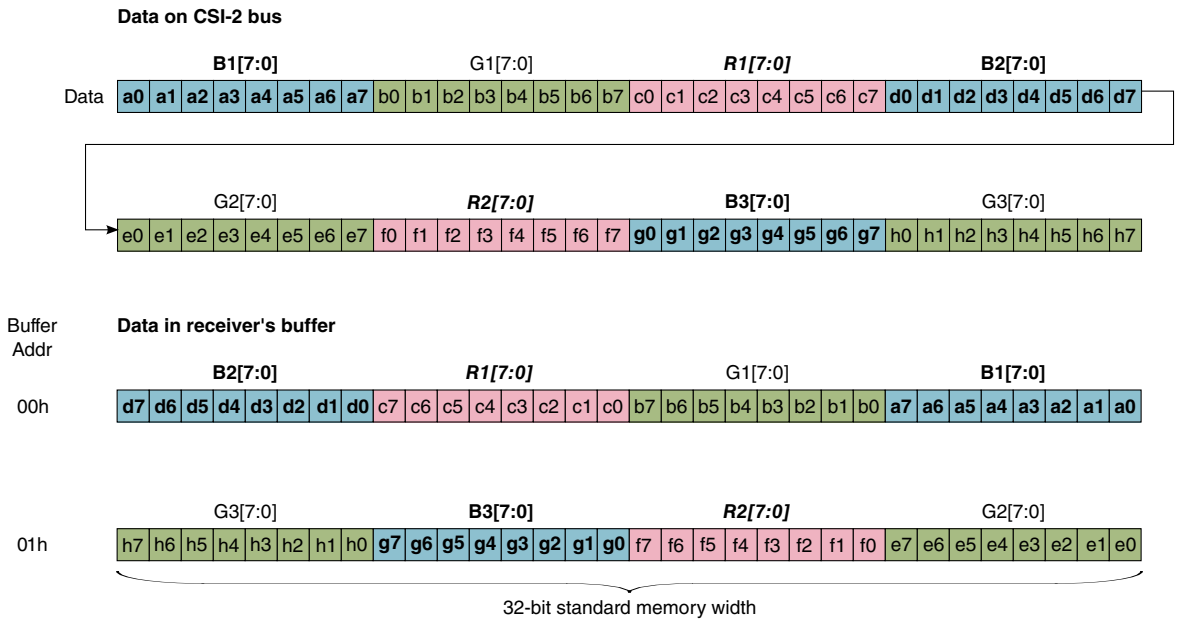


Figure 40-13. RGB888 Data Format Reception.

40.5.3 RGB666 Data Reception

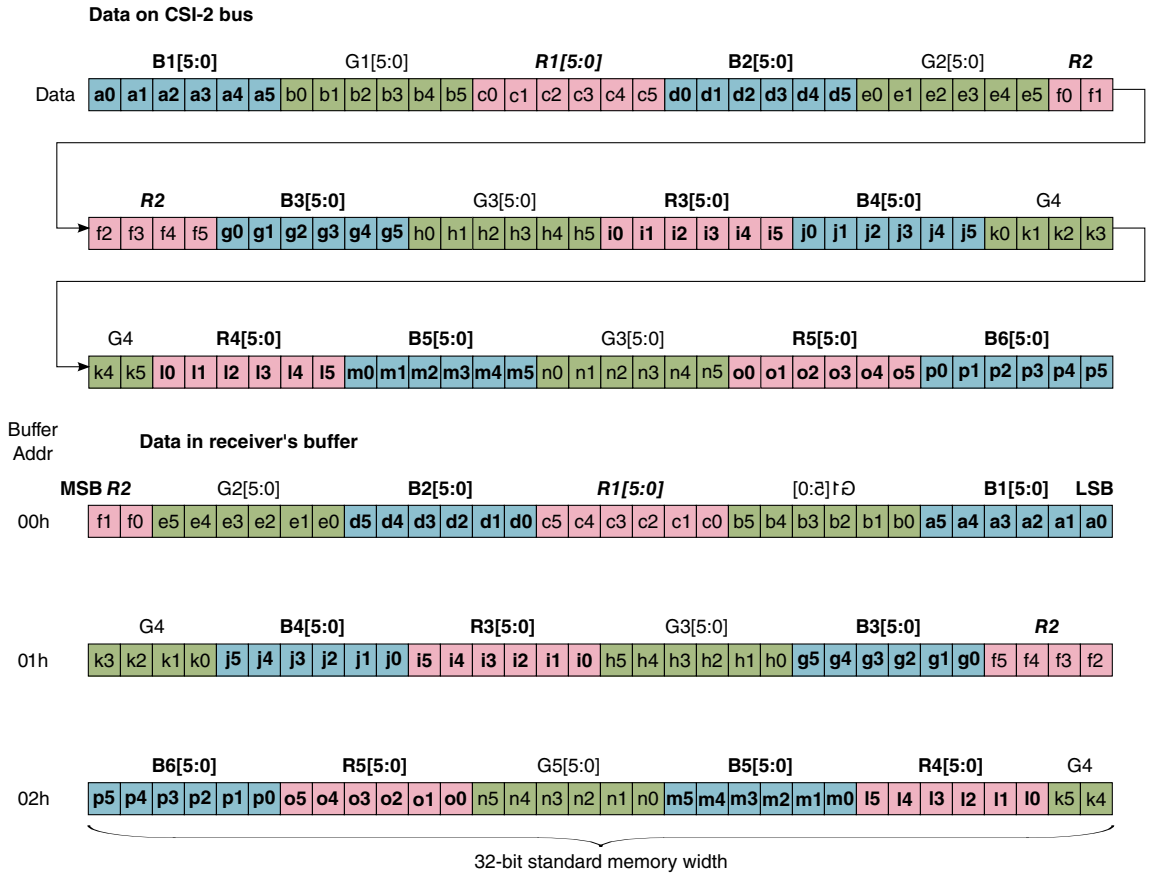


Figure 40-14. RGB666 Data Format Reception.

40.5.4 RGB565 Data Reception

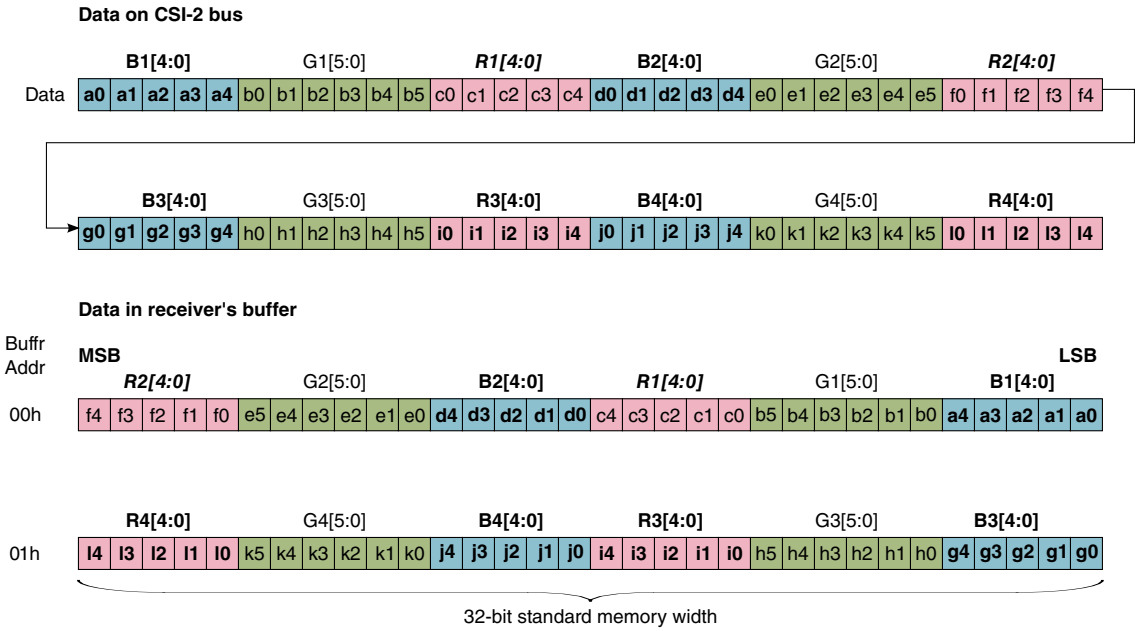


Figure 40-15. RGB565 Data Format Reception.

40.5.5 RGB555 Data Reception

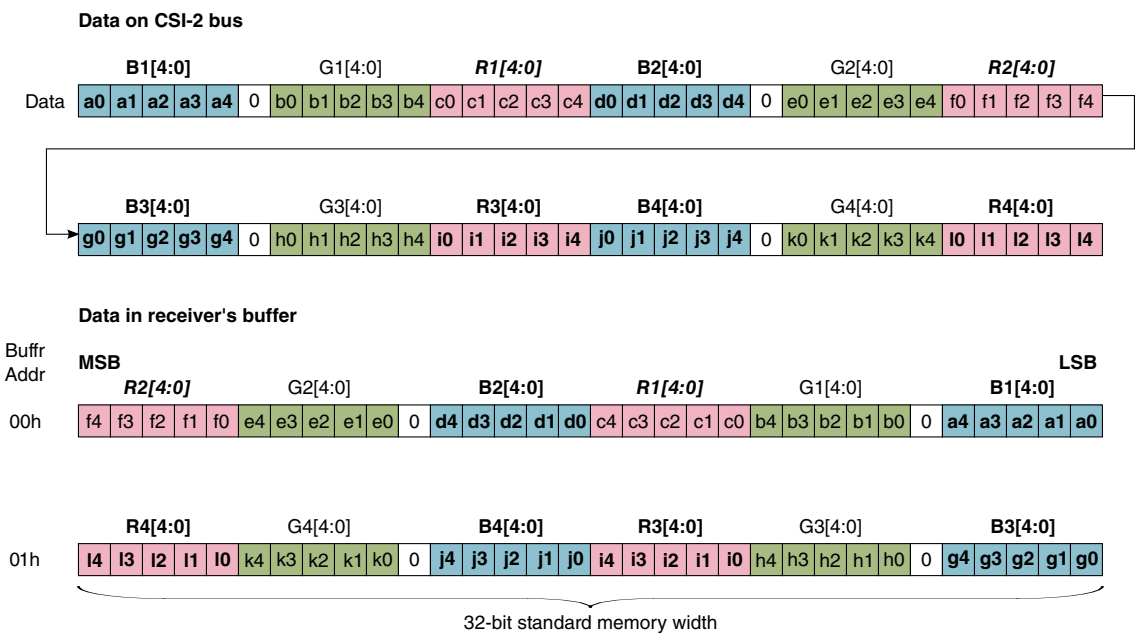


Figure 40-16. RGB555 Data Format Reception.

40.5.6 RGB444 Data Reception

The RGB444 data format byte to 32-bit memory word mapping has a special transform as shown in the following Figure:

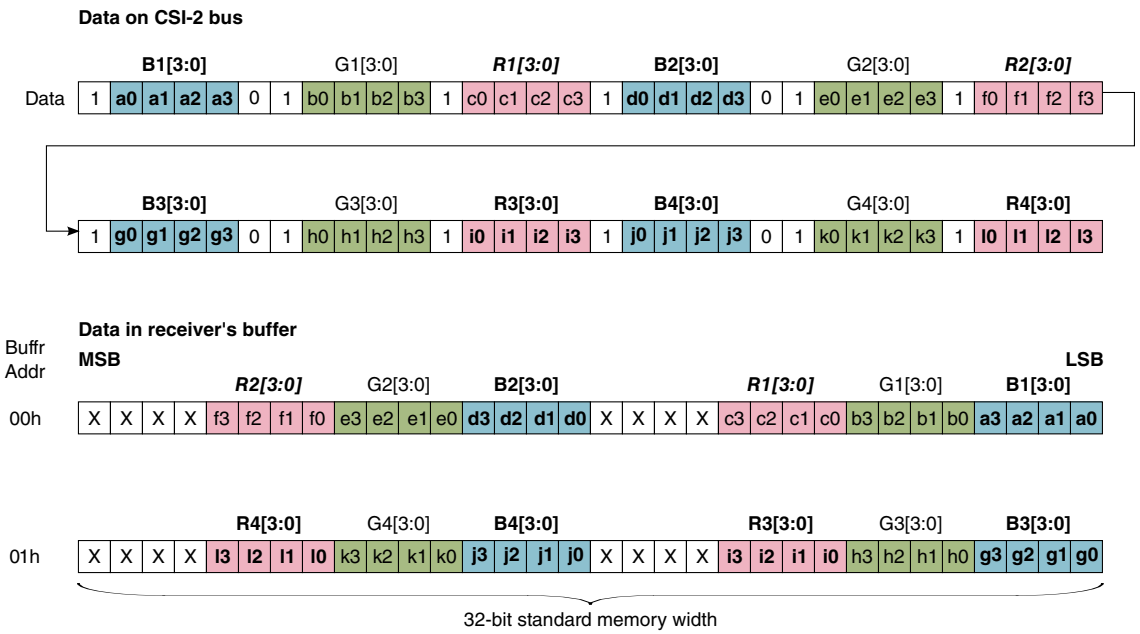


Figure 40-17. RGB444 Data Format Reception.

40.5.7 YUV422 8-bit Data Reception

The YUV422 8-bit data format the byte to 32-bit memory word mapping does not follow the generic CSI-2 rule.

For YUV422 8-bit data format the first byte of payload data transmitted maps the most significant byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the least significant byte of the 32-bit memory word.

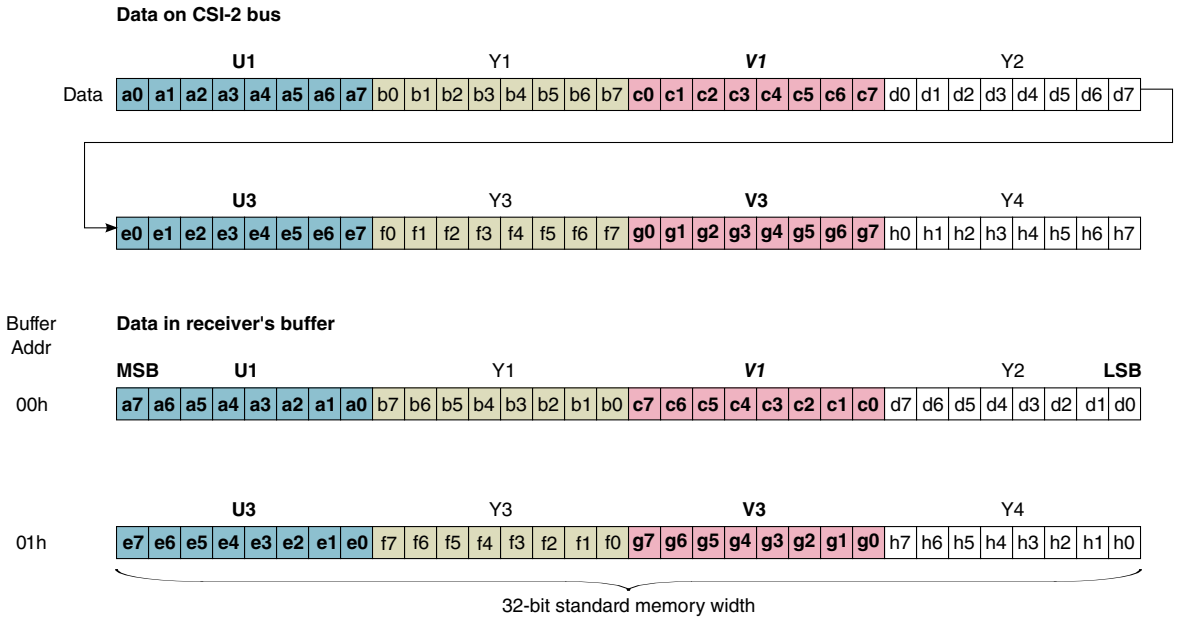


Figure 40-18. YUV422 8-bit Data Format Reception.

40.5.8 YUV422 10-bit Data Reception

The YUV422 10-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

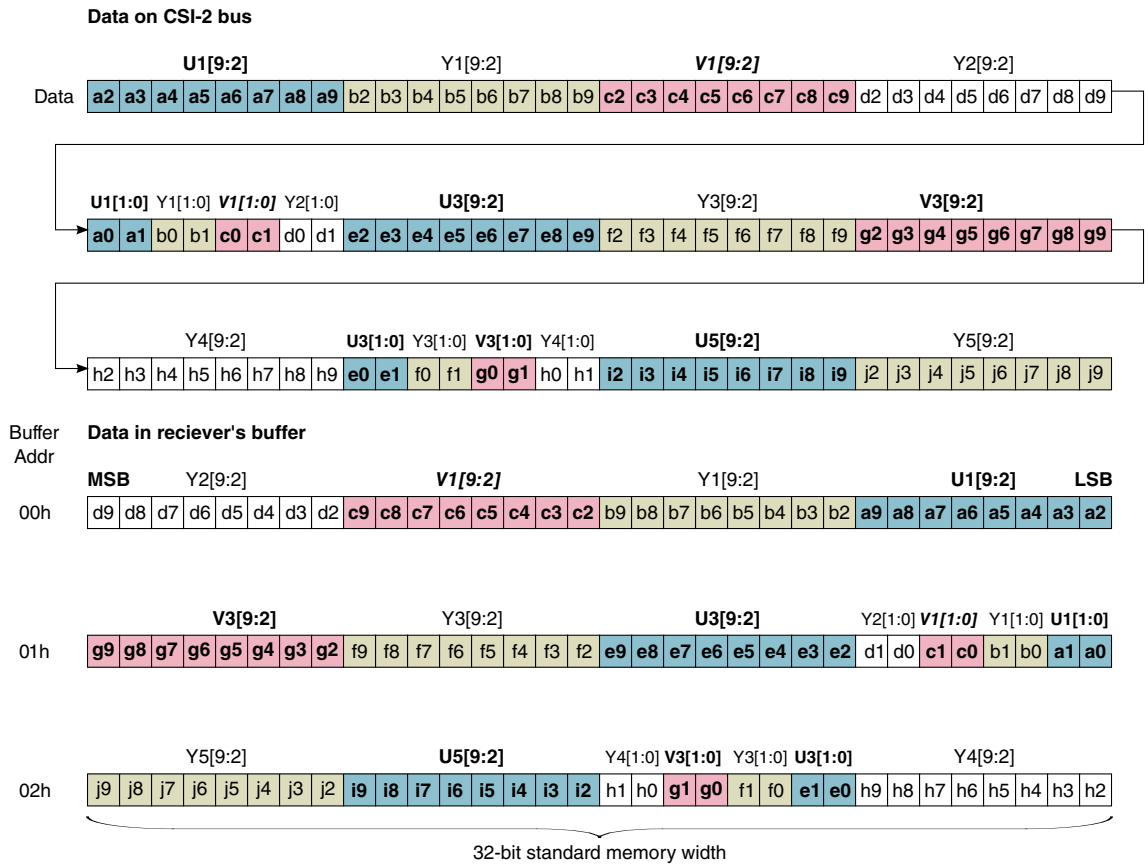


Figure 40-19. YUV422 10-bit Data Format Reception.

40.5.9 YUV420 8-bit (Legacy) Data Reception

The YUV420 8-bit (legacy) data format the byte to 32-bit memory word mapping does not follow the generic CSI-2 rule.

For YUV422 8-bit (legacy) data format the first byte of payload data transmitted maps the MS byte of the 32-bit memory word and the fourth byte of payload data transmitted maps to the LS byte of the 32-bit memory word.

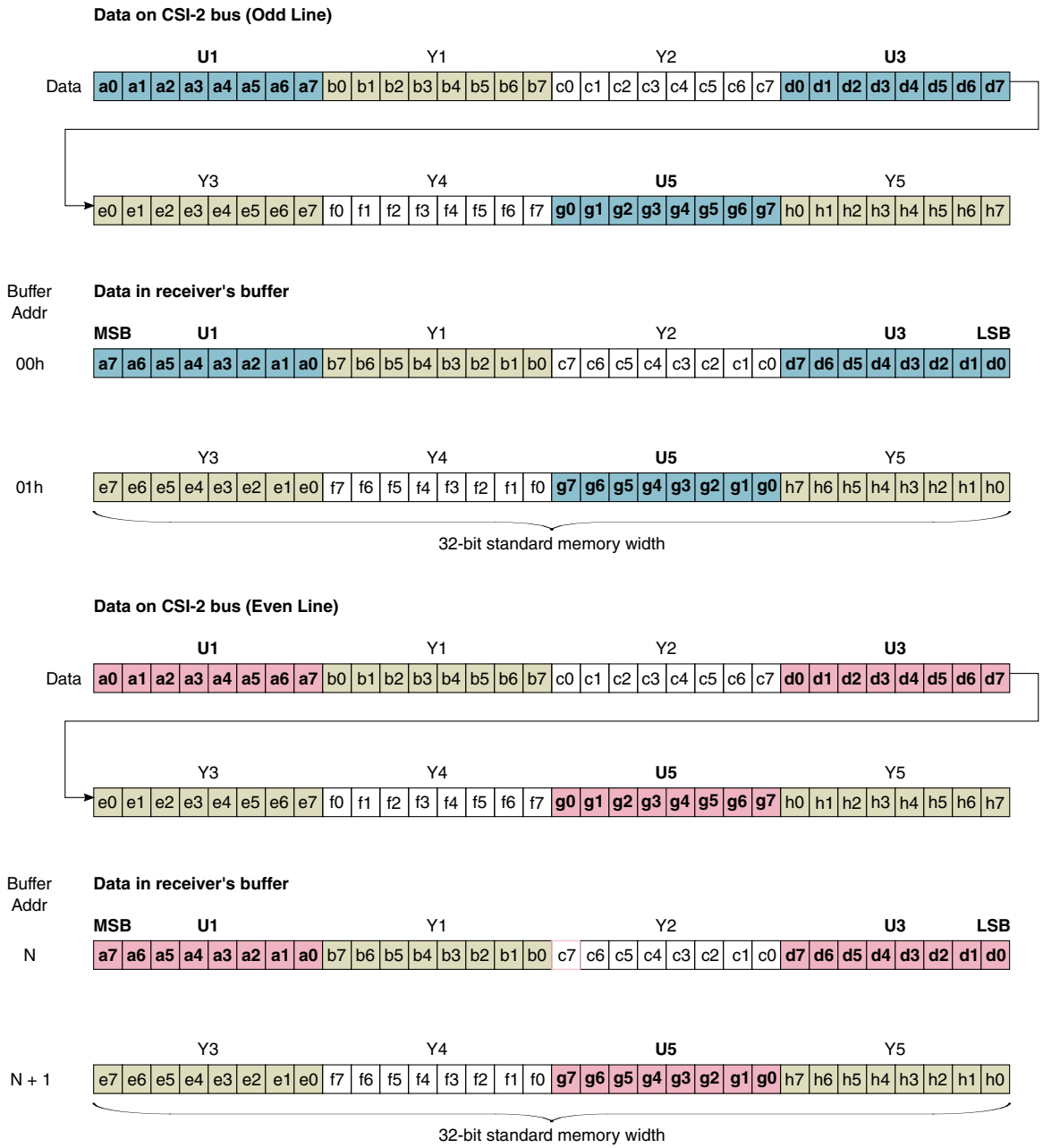


Figure 40-20. YUV420 8-bit Legacy Data Format Reception.

40.5.10 YUV420 8-bit Data Reception

The YUV420 8-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

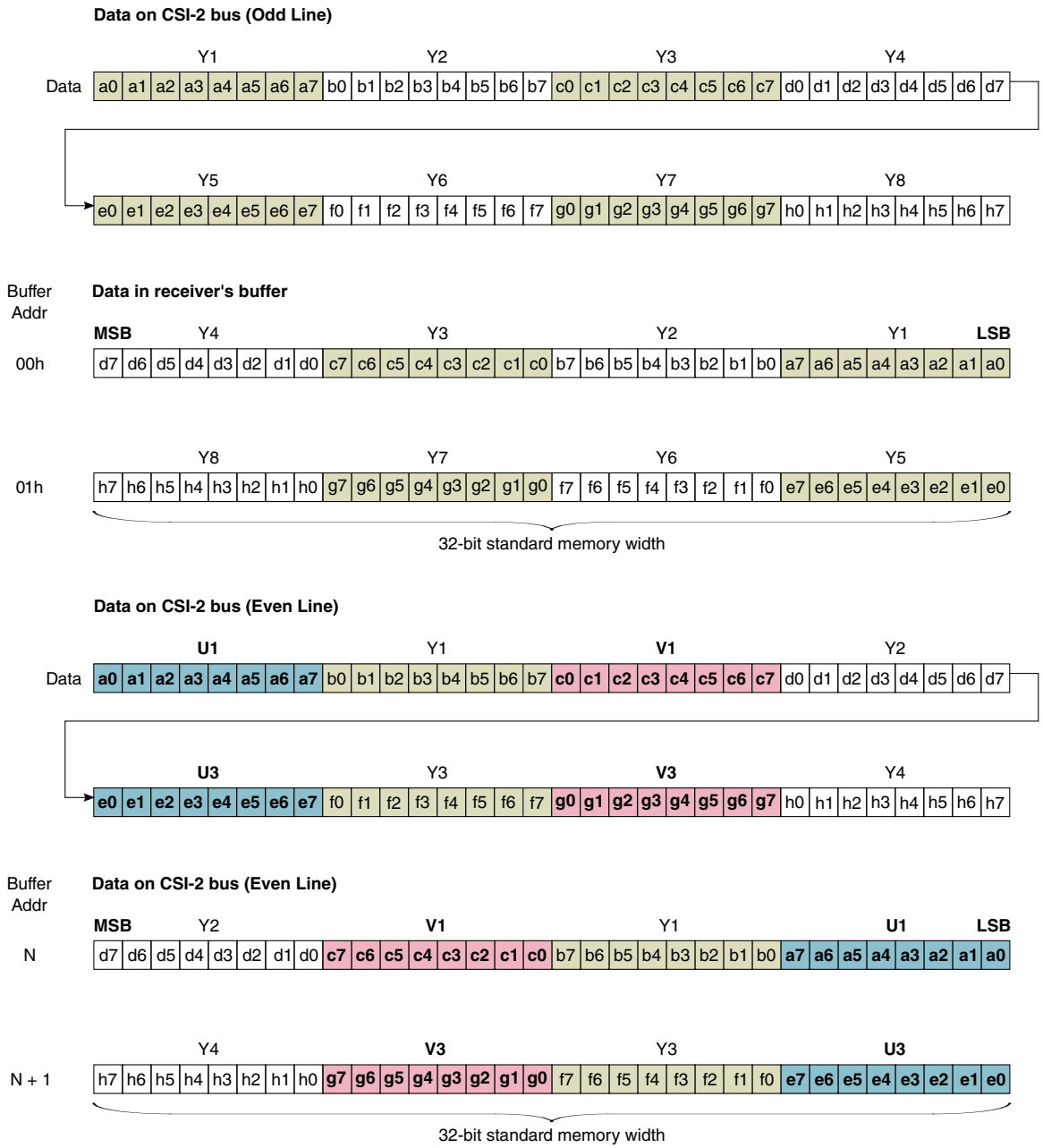


Figure 40-21. YUV420 8-bit Data Format Reception.

40.5.11 YUV420 10-bit Data Reception

The YUV420 10-bit data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

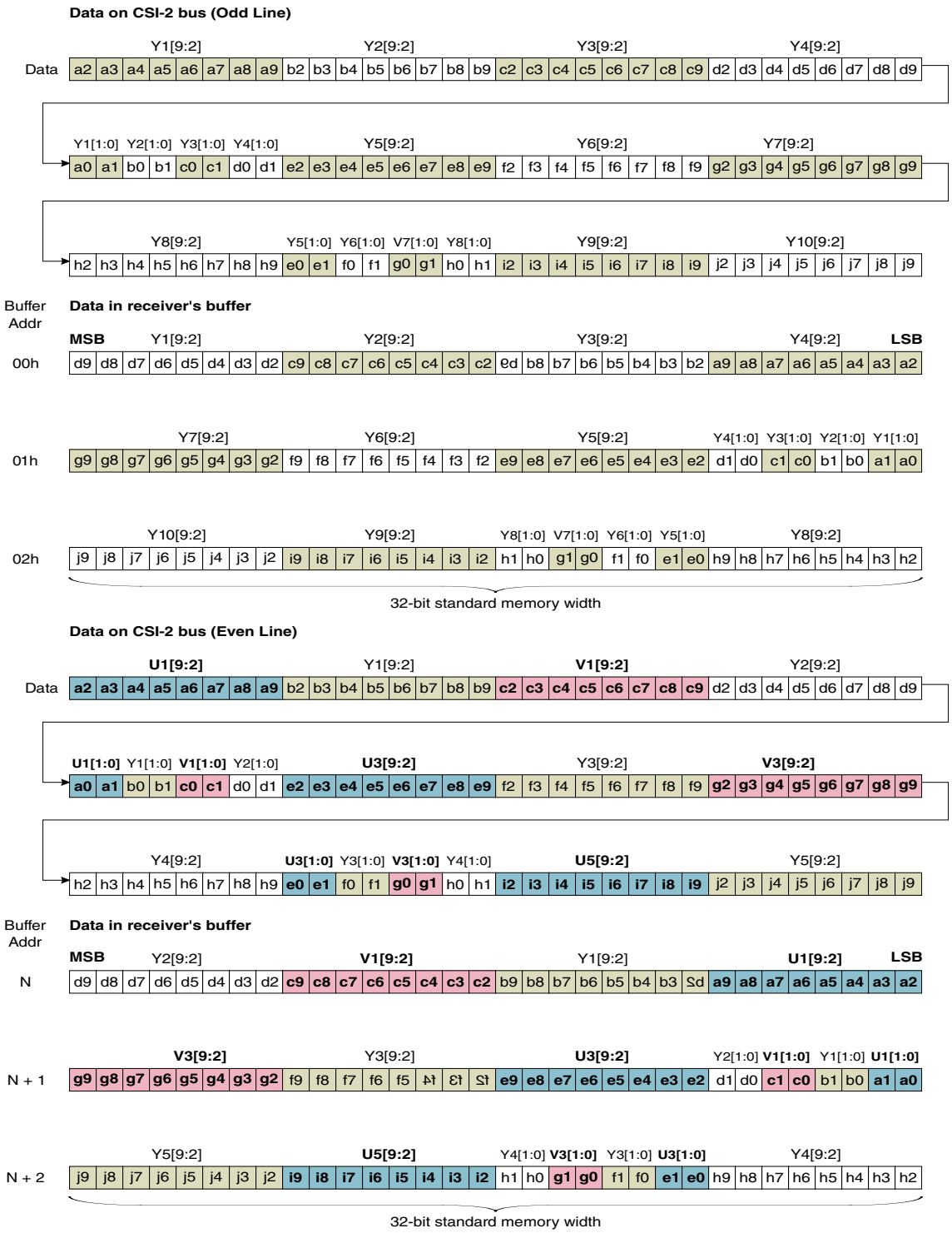


Figure 40-22. YUV420 10-bit Data Format Reception.

40.5.12 RAW6 Data Reception

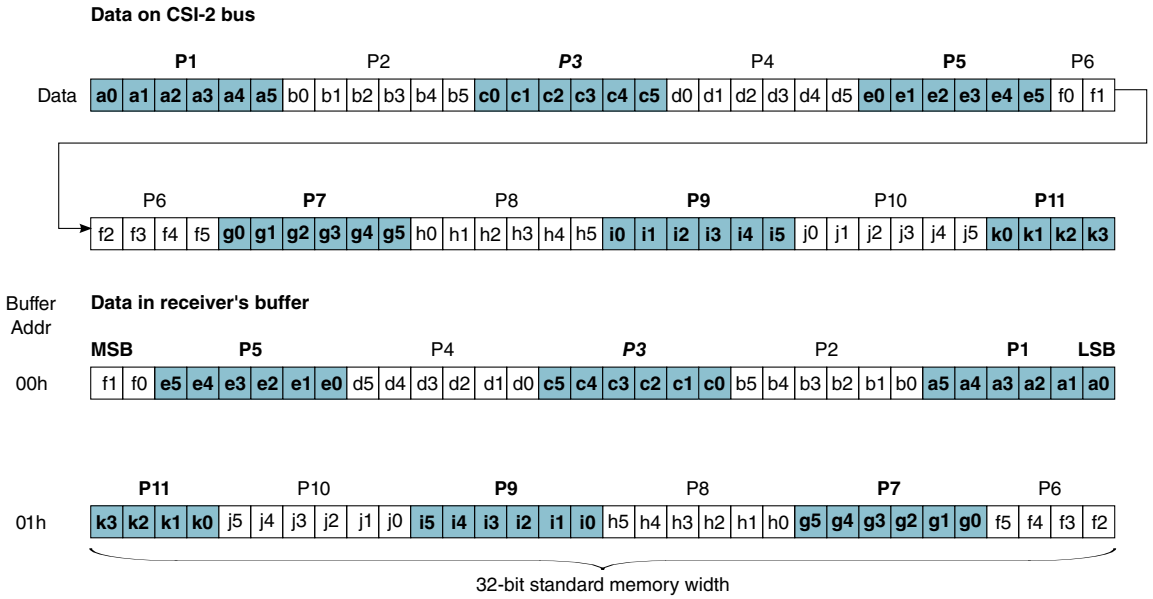


Figure 40-23. RAW6 Data Format Reception.

40.5.13 RAW7 Data Reception

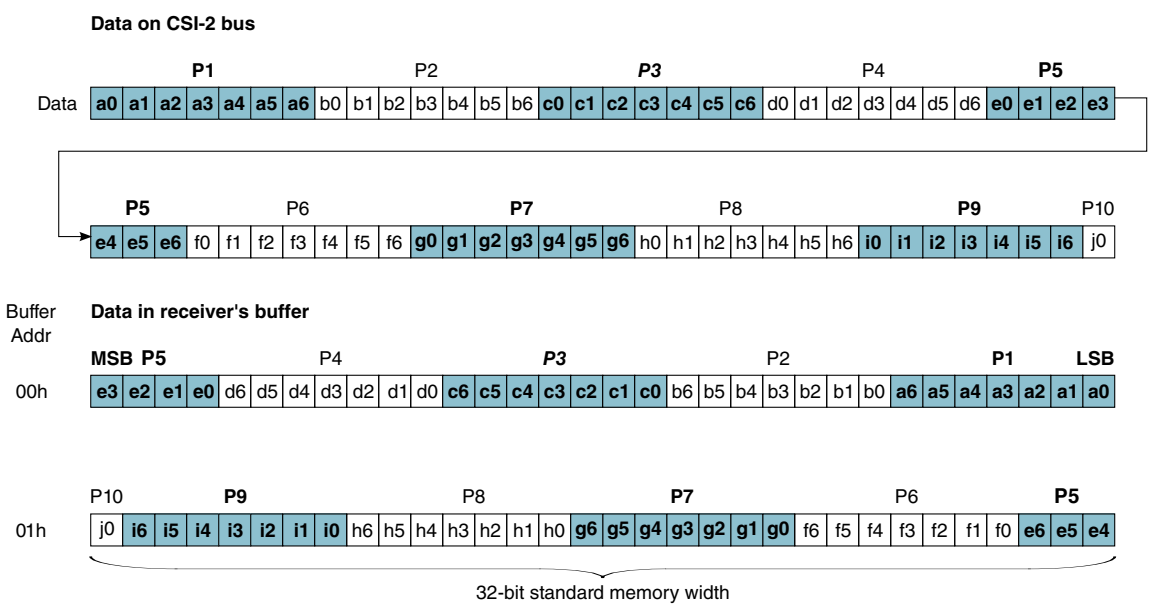


Figure 40-24. RAW7 Data Format Reception.

40.5.14 RAW8 Data Reception

The RAW8 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

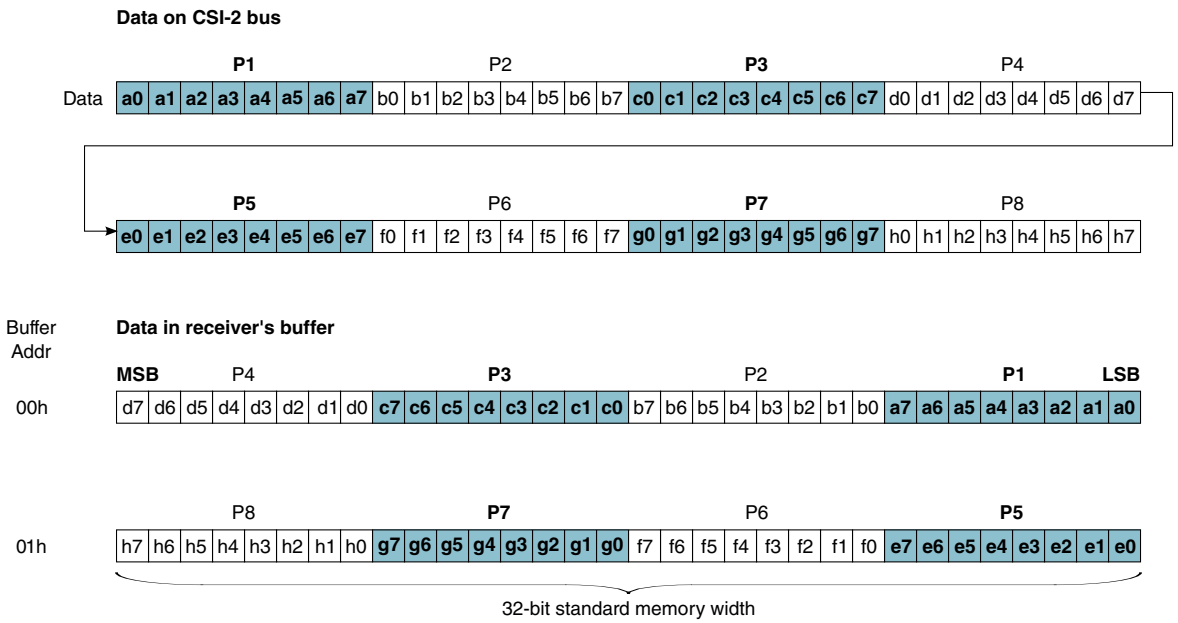


Figure 40-25. RAW8 Data Format Reception.

40.5.15 RAW10 Data Reception

The RAW10 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

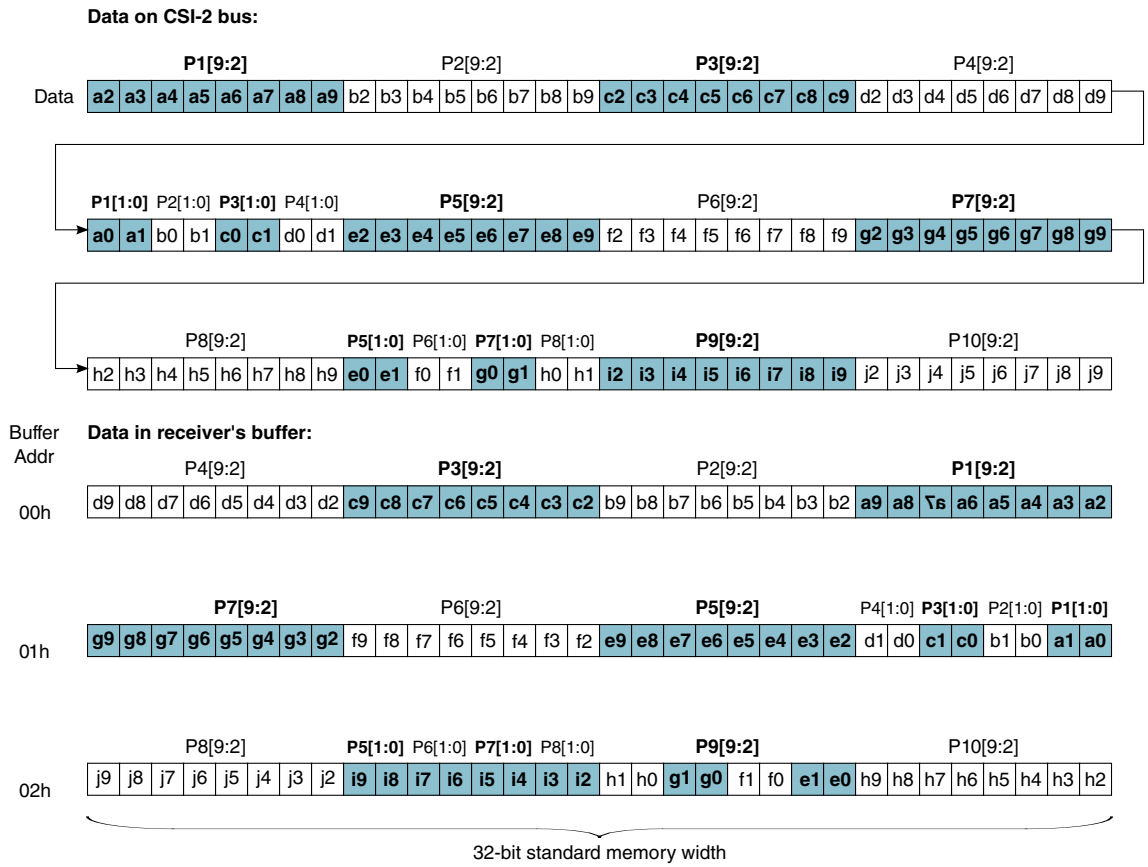


Figure 40-26. RAW10 Data Format Reception.

40.5.16 RAW12 Data Reception

The RAW12 data format the byte to 32-bit memory word mapping follows the generic CSI-2 rule.

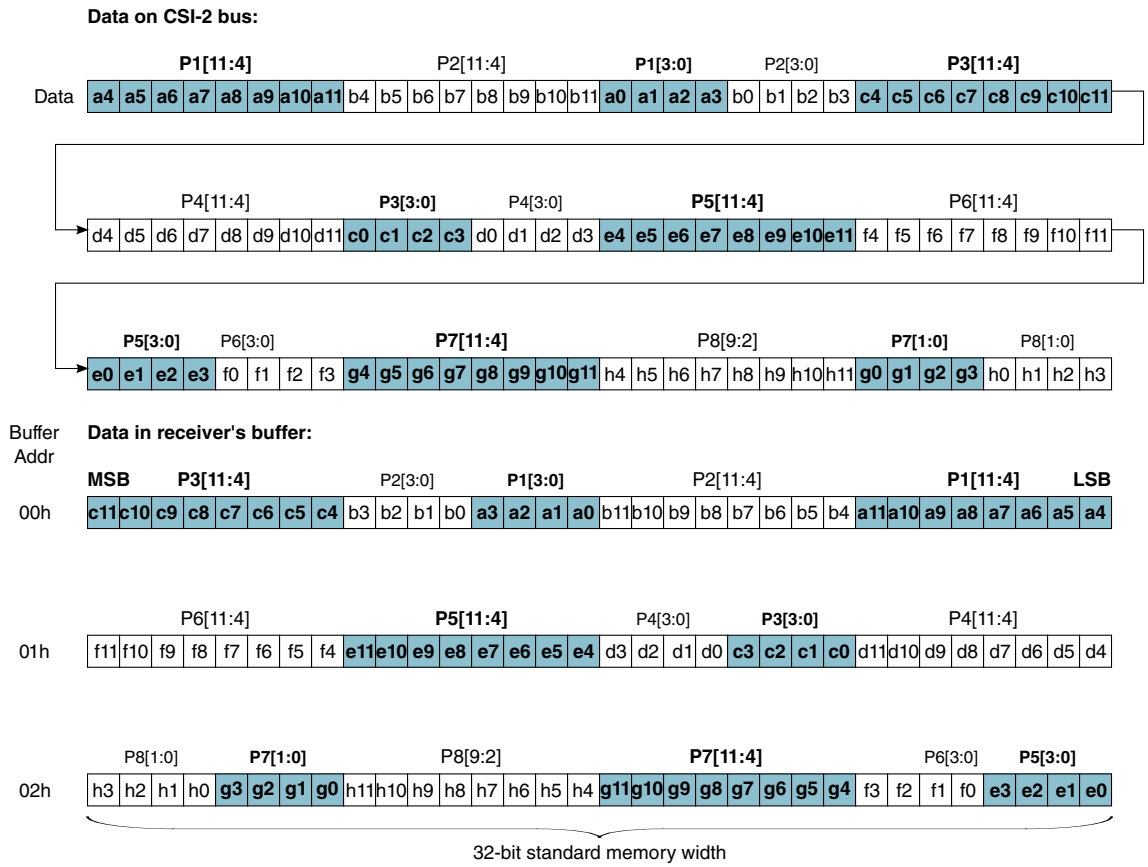


Figure 40-27. RAW12 Data Format Reception.

40.5.17 RAW14 Data Reception

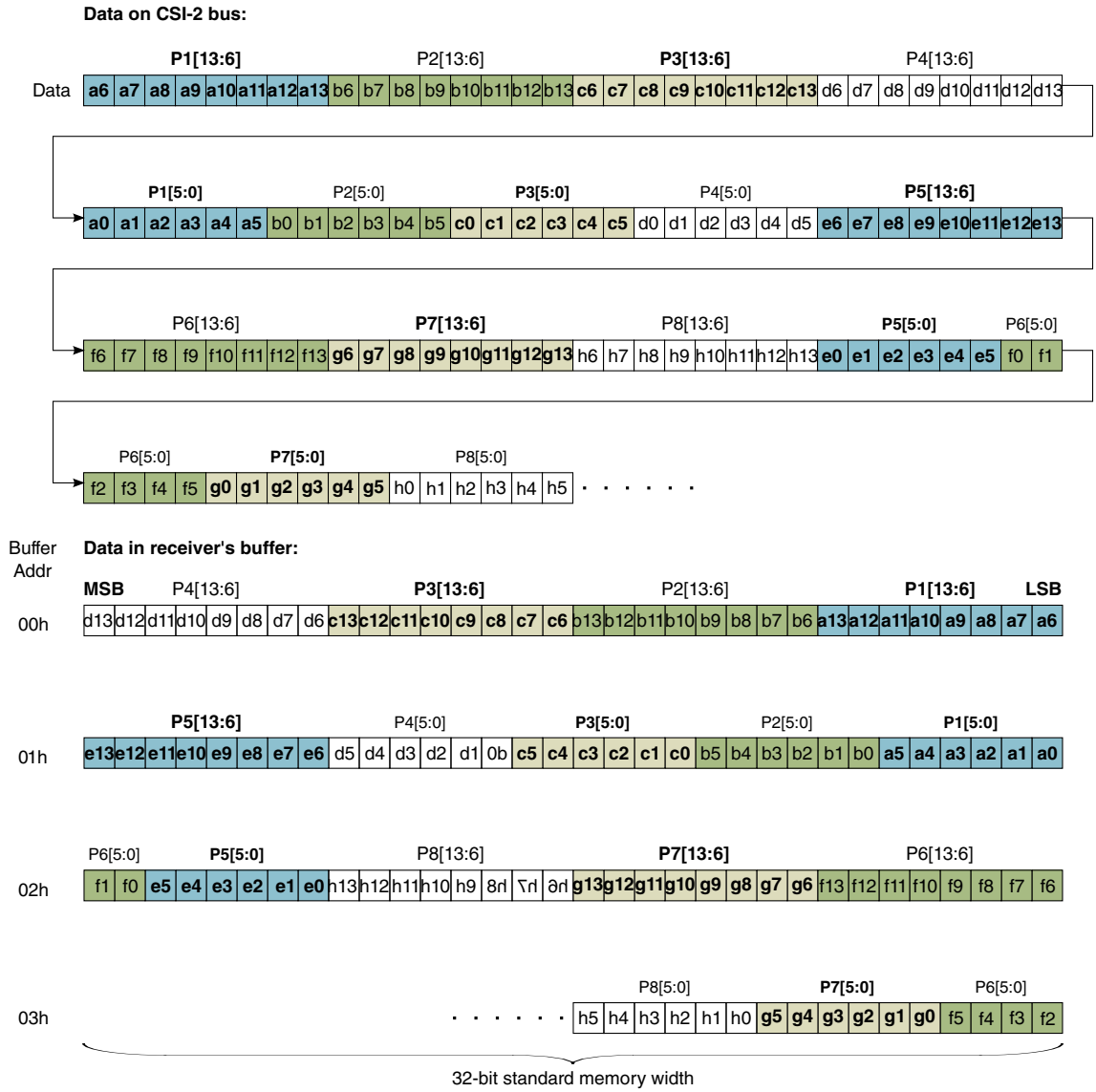


Figure 40-28. RAW 14 Data Format Reception.

40.6 MIPI_CSI Memory Map/Register Definition

MIPI_CSI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21D_C000	Controller Version Identification Register (MIPI_CSI_VERSION)	32	R	0000_0000h	40.6.1/3596

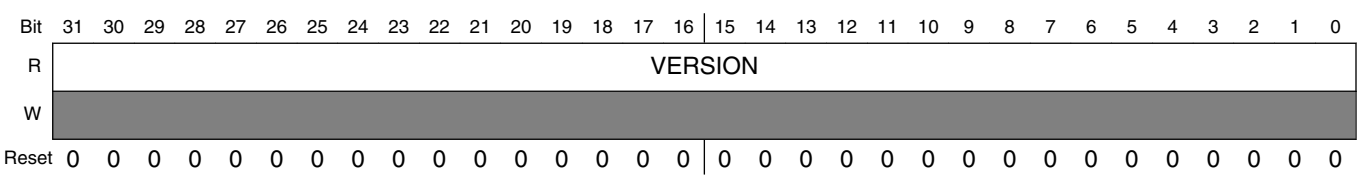
Table continues on the next page...

MIPI_CSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21D_C004	Number of Active Data Lanes (MIPI_CSI_N_LANES)	32	R/W	0000_0000h	40.6.2/3597
21D_C008	Phy shutdown control (MIPI_CSI_PHY_SHUTDOWNZ)	32	R/W	0000_0000h	40.6.3/3597
21D_C00C	Phy reset control (MIPI_CSI_DPHY_RSTZ)	32	R/W	0000_0000h	40.6.4/3598
21D_C010	CSI2 controller reset (MIPI_CSI_CSI2_RESETN)	32	R/W	0000_0000h	40.6.5/3599
21D_C014	General settings for all blocks (MIPI_CSI_PHY_STATE)	32	R	0000_0000h	40.6.6/3600
21D_C018	Data IDs for which IDI reports line boundary matching errors (MIPI_CSI_DATA_IDS_1)	32	R/W	0000_0000h	40.6.7/3601
21D_C01C	Data IDs for which IDI reports line boundary matching errors (MIPI_CSI_DATA_IDS_2)	32	R/W	0000_0000h	40.6.8/3602
21D_C020	Error state register 1 (MIPI_CSI_ERR1)	32	R	0000_0000h	40.6.9/3604
21D_C024	Error state register 2 (MIPI_CSI_ERR2)	32	R	0000_0000h	40.6.10/ 3608
21D_C028	Masks for errors 1 (MIPI_CSI_MASK1)	32	R/W	0000_0000h	40.6.11/ 3611
21D_C02C	Masks for errors 2 (MIPI_CSI_MASK2)	32	R/W	0000_0000h	40.6.12/ 3613
21D_C030	D-PHY Test interface control 0 (MIPI_CSI_PHY_TST_CRTL0)	32	R/W	0000_0000h	40.6.13/ 3615
21D_C034	D-PHY Test interface control 1 (MIPI_CSI_PHY_TST_CTRL1)	32	R/W	0000_0000h	40.6.14/ 3616

40.6.1 Controller Version Identification Register (MIPI_CSI_VERSION)

Address: 21D_C000h base + 0h offset = 21D_C000h



MIPI_CSI_VERSION field descriptions

Field	Description
VERSION	Version of the CSI-2 Host Controller **Default Value:** CSI_VERSION_ID

40.6.2 Number of Active Data Lanes (MIPI_CSI_N_LANES)

Address: 21D_C000h base + 4h offset = 21D_C004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved														N_LANES	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_N_LANES field descriptions

Field	Description
31–2 -	This field is reserved. Reserved
N_LANES	Number of Active Data Lanes Can only be updated when the PHY lane is in stop state. **Default Value:** CSI_N_LANES 00 1 Data Lane (Lane 0) 01 2 Data Lanes (Lane 0, and 1) 10 3 Data Lanes (Lane 0,1 and 2) 11 4 Data Lanes (All)

40.6.3 Phy shutdown control (MIPI_CSI_PHY_SHUTDOWNZ)

Address: 21D_C000h base + 8h offset = 21D_C008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

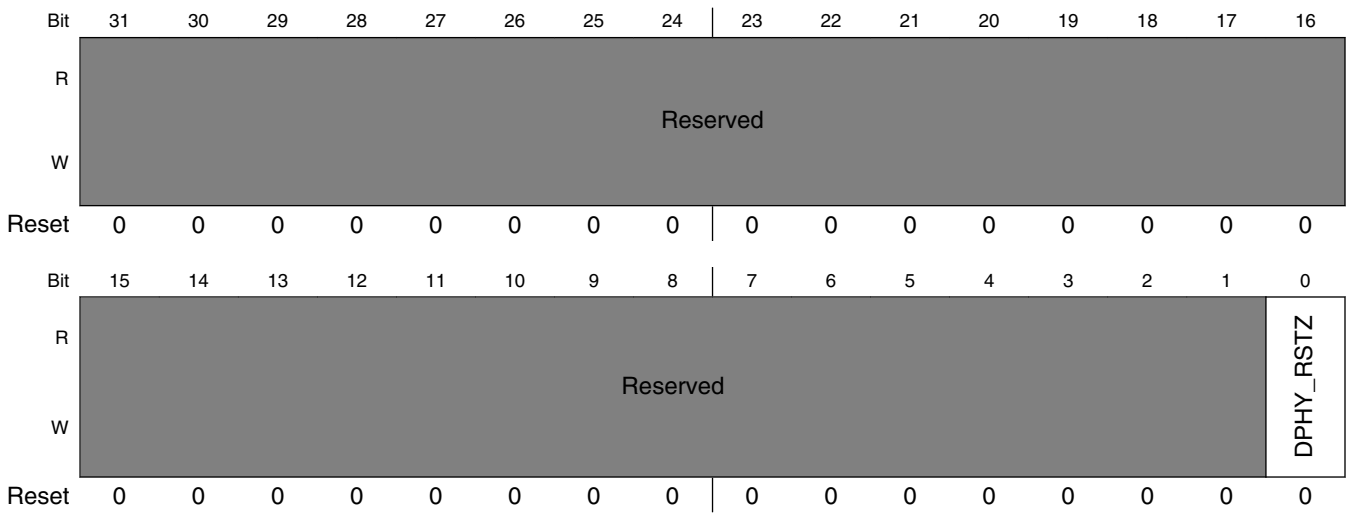
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved															PHY_ SHUTDOWNZ
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_PHY_SHUTDOWNZ field descriptions

Field	Description
31–1 -	This field is reserved. Reserved
0 PHY_ SHUTDOWNZ	Shutdown input. This line is used to place the complete module in power down. All analog blocks are in power down mode and digital logic is cleared. Active Low **Default Value:** 0

40.6.4 Phy reset control (MIPI_CSI_DPHY_RSTZ)

Address: 21D_C000h base + Ch offset = 21D_C00Ch

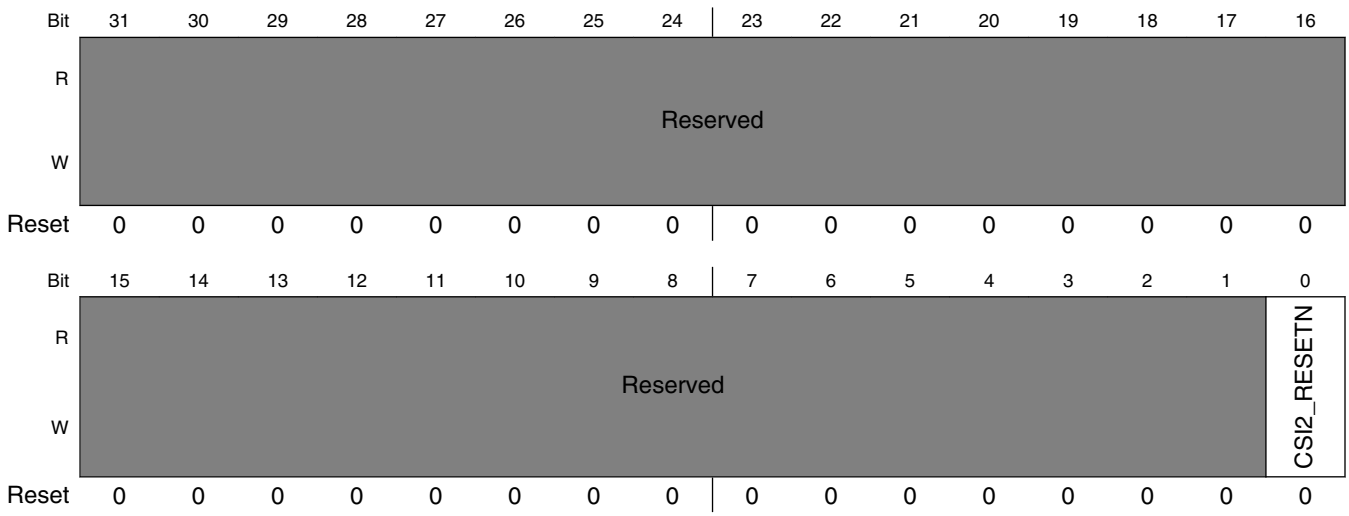


MIPI_CSI_DPHY_RSTZ field descriptions

Field	Description
31–1 -	This field is reserved. Reserved
0 DPHY_RSTZ	DPHY reset output. Active Low **Default Value:** 0

40.6.5 CSI2 controller reset (MIPI_CSI_CSI2_RESETN)

Address: 21D_C000h base + 10h offset = 21D_C010h

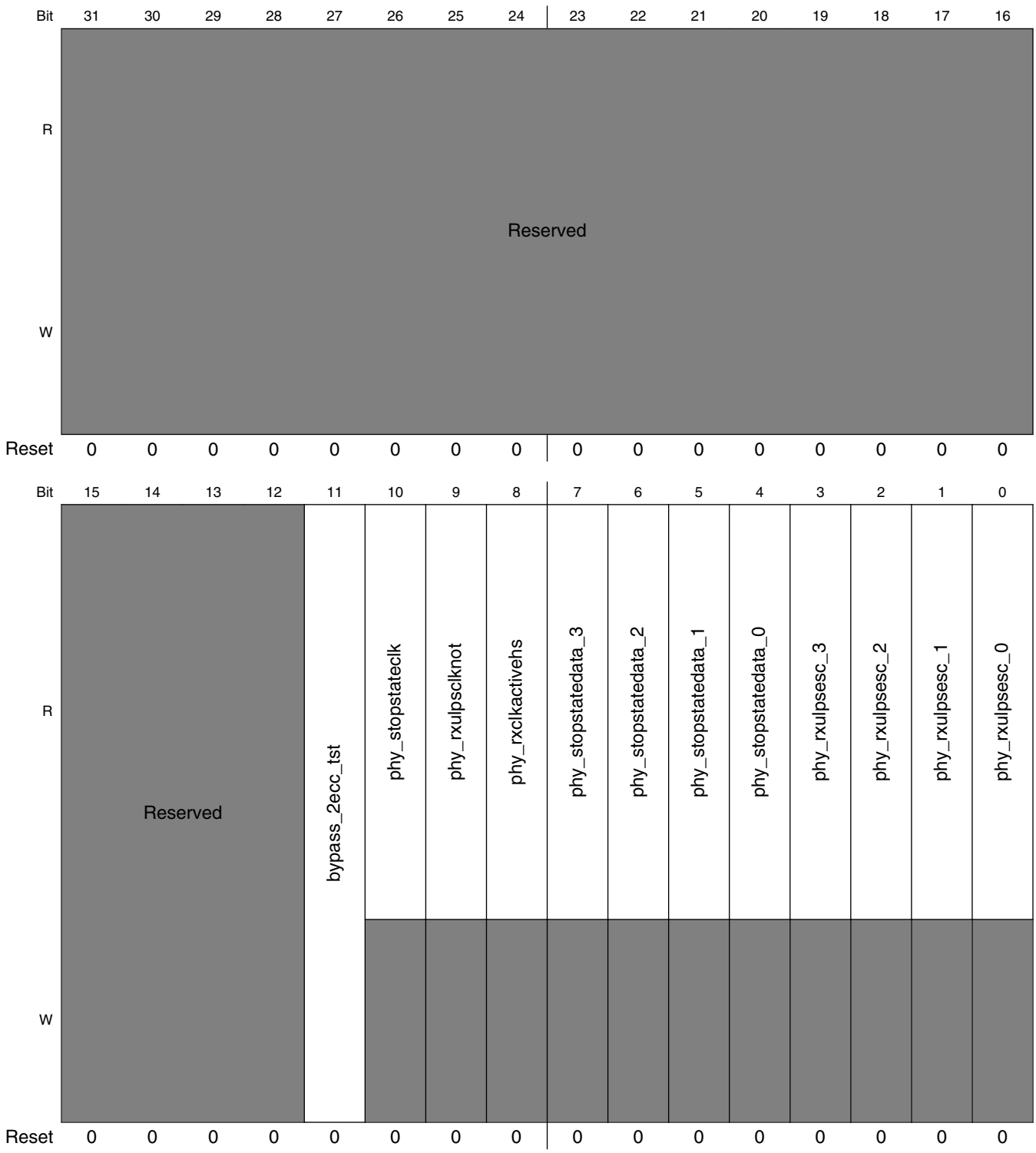


MIPI_CSI_CSI2_RESETN field descriptions

Field	Description
31–1 -	This field is reserved. Reserved
0 CSI2_RESETN	CSI-2 controller reset output. Active Low **Default Value:** 0

40.6.6 General settings for all blocks (MIPI_CSI_PHY_STATE)

Address: 21D_C000h base + 14h offset = 21D_C014h



MIPI_CSI_PHY_STATE field descriptions

Field	Description
31–12 -	This field is reserved. Reserved
11 bypass_2ecc_tst	Payload Bypass test mode for double ECC errors **Default Value:** 0
10 phy_stopstateclk	Clock Lane in Stop state **Default Value:** 0
9 phy_rxulpsclknot	Active Low. This signal indicates that the Clock Lane module has entered the Ultra Low Power state **Default Value:** 0
8 phy_rxclkactivehs	Indicates that the clock lane is actively receiving a DDR clock **Default Value:** 0
7 phy_ stopstatedata_3	Data Lane 3 in Stop state **Default Value:** 0
6 phy_ stopstatedata_2	Data Lane 2 in Stop state **Default Value:** 0
5 phy_ stopstatedata_1	Data Lane 1 in Stop state **Default Value:** 0
4 phy_ stopstatedata_0	Data Lane 0 in Stop state **Default Value:** 0
3 phy_rxulpsesc_3	Lane module 3 has entered the Ultra Low Power mode **Default Value:** 0
2 phy_rxulpsesc_2	Lane module 2 has entered the Ultra Low Power mode **Default Value:** 0
1 phy_rxulpsesc_1	Lane module 1 has entered the Ultra Low Power mode **Default Value:** 0
0 phy_rxulpsesc_0	Lane module 0 has entered the Ultra Low Power mode **Default Value:** 0

40.6.7 Data IDs for which IDI reports line boundary matching errors (MIPI_CSI_DATA_IDS_1)

Address: 21D_C000h base + 18h offset = 21D_C018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	di3_vc				di3_dt						di2_vc				di2_dt				di1_vc				di1_dt						di0_vc				di0_dt					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_DATA_IDS_1 field descriptions

Field	Description
31–30 di3_vc	Data ID 3 Virtual channel **Default Value:** 0
29–24 di3_dt	Data ID 3 Data Type **Default Value:** 0
23–22 di2_vc	DATA ID 2 Virtual channel **Default Value:** 0
21–16 di2_dt	DATA ID 2 Data Type **Default Value:** 0
15–14 di1_vc	Data ID 1 Virtual channel **Default Value:** 0
13–8 di1_dt	Data ID 1 Data Type **Default Value:** 0
7–6 di0_vc	Data ID 0 Virtual channel **Default Value:** 0
di0_dt	Data ID 0 Data Type **Default Value:** 0

40.6.8 Data IDs for which IDI reports line boundary matching errors (MIPI_CSI_DATA_IDS_2)

Address: 21D_C000h base + 1Ch offset = 21D_C01Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	di7_vc		di7_dt						di6_vc		di6_dt						di5_vc		di5_dt						di4_vc		di4_dt					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_DATA_IDS_2 field descriptions

Field	Description
31–30 di7_vc	Data ID 7 Virtual channel **Default Value:** 0
29–24 di7_dt	Data ID 7 Data Type **Default Value:** 0
23–22 di6_vc	Data ID 6 Virtual channel **Default Value:** 0
21–16 di6_dt	Data ID 6 Data Type **Default Value:** 0
15–14 di5_vc	Data ID 5 Virtual channel **Default Value:** 0

Table continues on the next page...

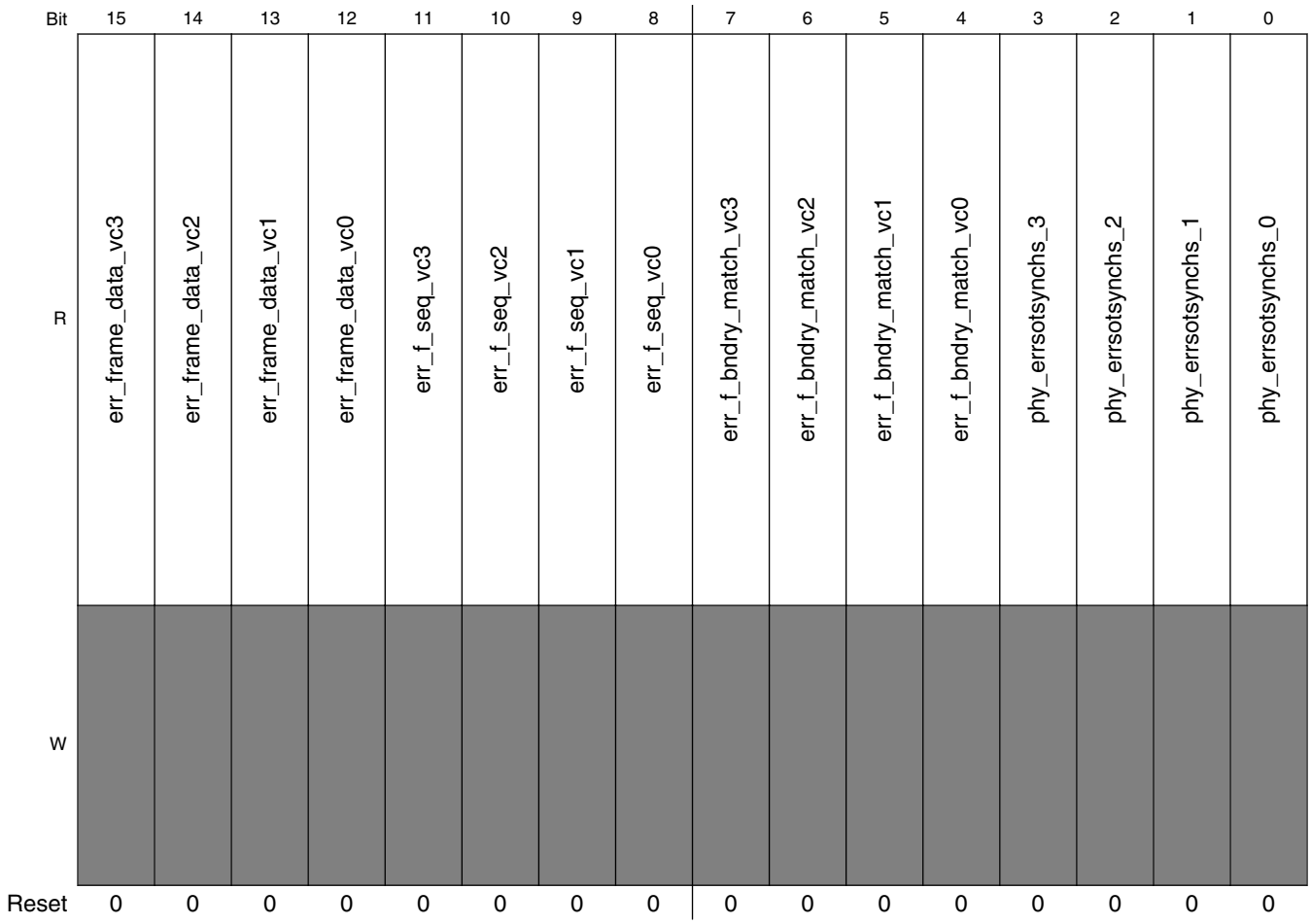
MIPI_CSI_DATA_IDS_2 field descriptions (continued)

Field	Description
13–8 di5_dt	Data ID 5 Data Type **Default Value:** 0
7–6 di4_vc	Data ID 4 Virtual channel **Default Value:** 0
di4_dt	Data ID 4 Data Type **Default Value:** 0

40.6.9 Error state register 1 (MIPI_CSI_ERR1)

Address: 21D_C000h base + 20h offset = 21D_C020h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved			err_ecc_double	vc3_err_crc	vc2_err_crc	vc1_err_crc	vc0_err_crc	err_l_seq_di3	err_l_seq_di2	err_l_seq_di1	err_l_seq_di0	err_l_bndry_match_di3	err_l_bndry_match_di2	err_l_bndry_match_di1	err_l_bndry_match_di0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



MIPI_CSI_ERR1 field descriptions

Field	Description
31–29 -	This field is reserved. Reserved
28 err_ecc_double	Header ECC contains 2 errors. Unrecoverable. **Default Value:** 0
27 vc3_err_crc	Checksum Error detected on Virtual Channel 3 **Default Value:** 0
26 vc2_err_crc	Checksum Error detected on Virtual Channel 2 **Default Value:** 0
25 vc1_err_crc	Checksum Error detected on Virtual Channel 1 **Default Value:** 0
24 vc0_err_crc	Checksum Error detected on Virtual Channel 0 **Default Value:** 0
23 err_l_seq_di3	Error in the sequence of lines for vc3 and dt3 **Default Value:** 0

Table continues on the next page...

MIPI_CSI_ERR1 field descriptions (continued)

Field	Description
22 err_l_seq_di2	Error in the sequence of lines for vc2 and dt2 **Default Value:** 0
21 err_l_seq_di1	Error in the sequence of lines for vc1 and dt1 **Default Value:** 0
20 err_l_seq_di0	Error in the sequence of lines for vc0 and dt0 **Default Value:** 0
19 err_l_bndry_match_di3	Error matching Line Start with Line End for vc3 and dt3 **Default Value:** 0
18 err_l_bndry_match_di2	Error matching Line Start with Line End for vc2 and dt2 **Default Value:** 0
17 err_l_bndry_match_di1	Error matching Line Start with Line End for vc1 and dt1 **Default Value:** 0
16 err_l_bndry_match_di0	Error matching Line Start with Line End for vc0 and dt0 **Default Value:** 0
15 err_frame_data_vc3	Last received frame, in Virtual Channel 3, had at least one CRC error **Default Value:** 0
14 err_frame_data_vc2	Last received frame, in Virtual Channel 2, had at least one CRC error **Default Value:** 0
13 err_frame_data_vc1	Last received frame, in Virtual Channel 1, had at least one CRC error **Default Value:** 0
12 err_frame_data_vc0	Last received frame, in Virtual Channel 0, had at least one CRC error **Default Value:** 0
11 err_f_seq_vc3	Incorrect Frame Sequence detected in Virtual Channel 3 **Default Value:** 0
10 err_f_seq_vc2	Incorrect Frame Sequence detected in Virtual Channel 2 **Default Value:** 0
9 err_f_seq_vc1	Incorrect Frame Sequence detected in Virtual Channel 1 **Default Value:** 0
8 err_f_seq_vc0	Incorrect Frame Sequence detected in Virtual Channel 0 **Default Value:** 0
7 err_f_bndry_match_vc3	Error matching Frame Start with Frame End for Virtual Channel 3 **Default Value:** 0
6 err_f_bndry_match_vc2	Error matching Frame Start with Frame End for Virtual Channel 2 **Default Value:** 0

Table continues on the next page...

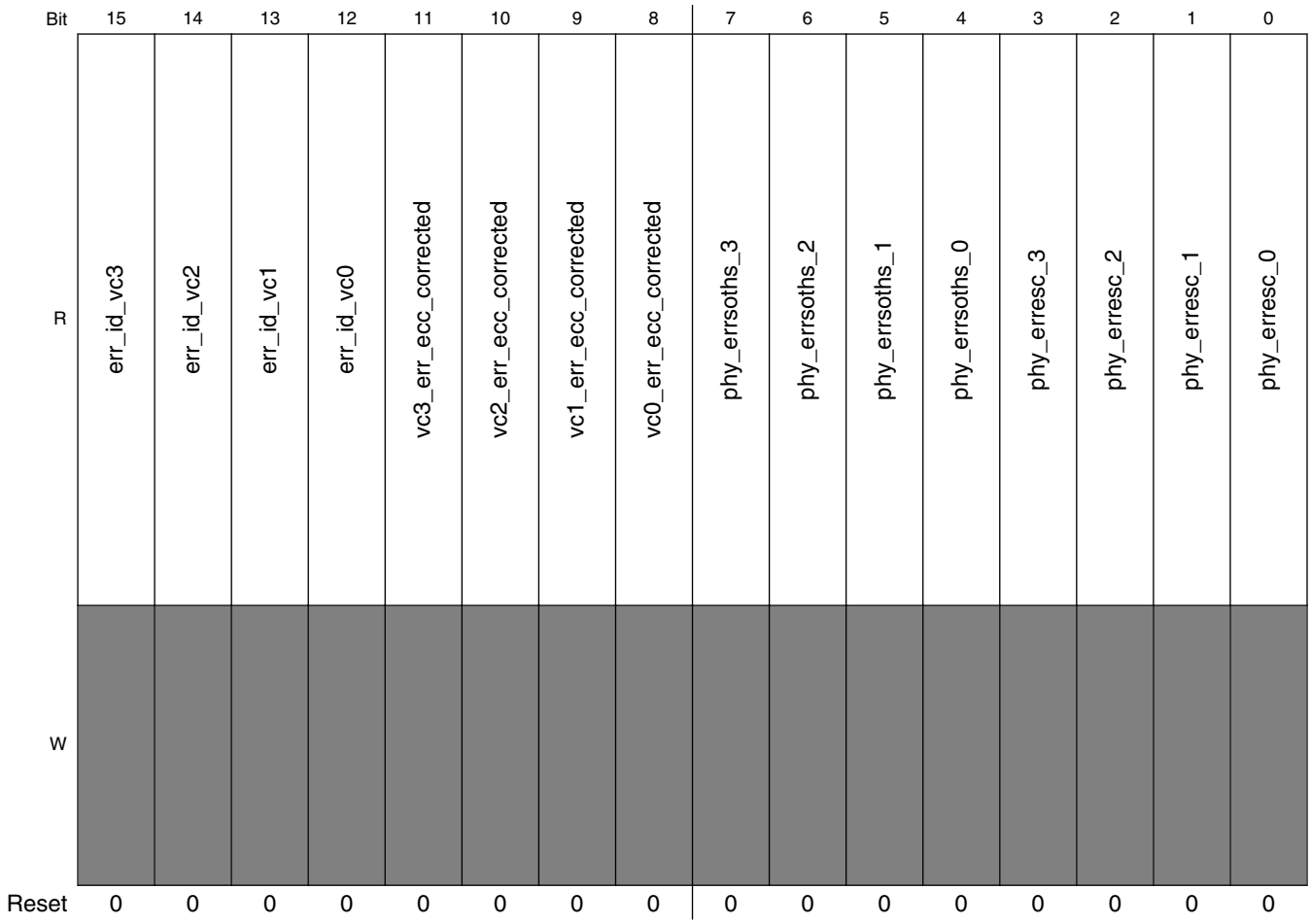
MIPI_CSI_ERR1 field descriptions (continued)

Field	Description
5 err_f_bndry_ match_vc1	Error matching Frame Start with Frame End for Virtual Channel 1 **Default Value:** 0
4 err_f_bndry_ match_vc0	Error matching Frame Start with Frame End for Virtual Channel 0 **Default Value:** 0
3 phy_ errsotsynchs_3	Start of Transmission Error on data lane 3 (no synchronization achieved) **Default Value:** 0
2 phy_ errsotsynchs_2	Start of Transmission Error on data lane 2 (no synchronization achieved) **Default Value:** 0
1 phy_ errsotsynchs_1	Start of Transmission Error on data lane 1 (no synchronization achieved) **Default Value:** 0
0 phy_ errsotsynchs_0	Start of Transmission Error on data lane 0 (no synchronization achieved) **Default Value:** 0

40.6.10 Error state register 2 (MIPI_CSI_ERR2)

Address: 21D_C000h base + 24h offset = 21D_C024h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved								err_l_seq_di7	err_l_seq_di6	err_l_seq_di5	err_l_seq_di4	err_l_bndry_match_di7	err_l_bndry_match_di6	err_l_bndry_match_di5	err_l_bndry_match_di4
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



MIPI_CSI_ERR2 field descriptions

Field	Description
31–24 -	This field is reserved. Reserved
23 err_l_seq_di7	Error in the sequence of lines for vc7 and dt7 **Default Value:** 0
22 err_l_seq_di6	Error in the sequence of lines for vc6 and dt6 **Default Value:** 0
21 err_l_seq_di5	Error in the sequence of lines for vc5 and dt5 **Default Value:** 0
20 err_l_seq_di4	Error in the sequence of lines for vc4 and dt4 **Default Value:** 0
19 err_l_bndry_ match_di7	Error matching Line Start with Line End for vc7 and dt7 **Default Value:** 0
18 err_l_bndry_ match_di6	Error matching Line Start with Line End for vc6 and dt6 **Default Value:** 0

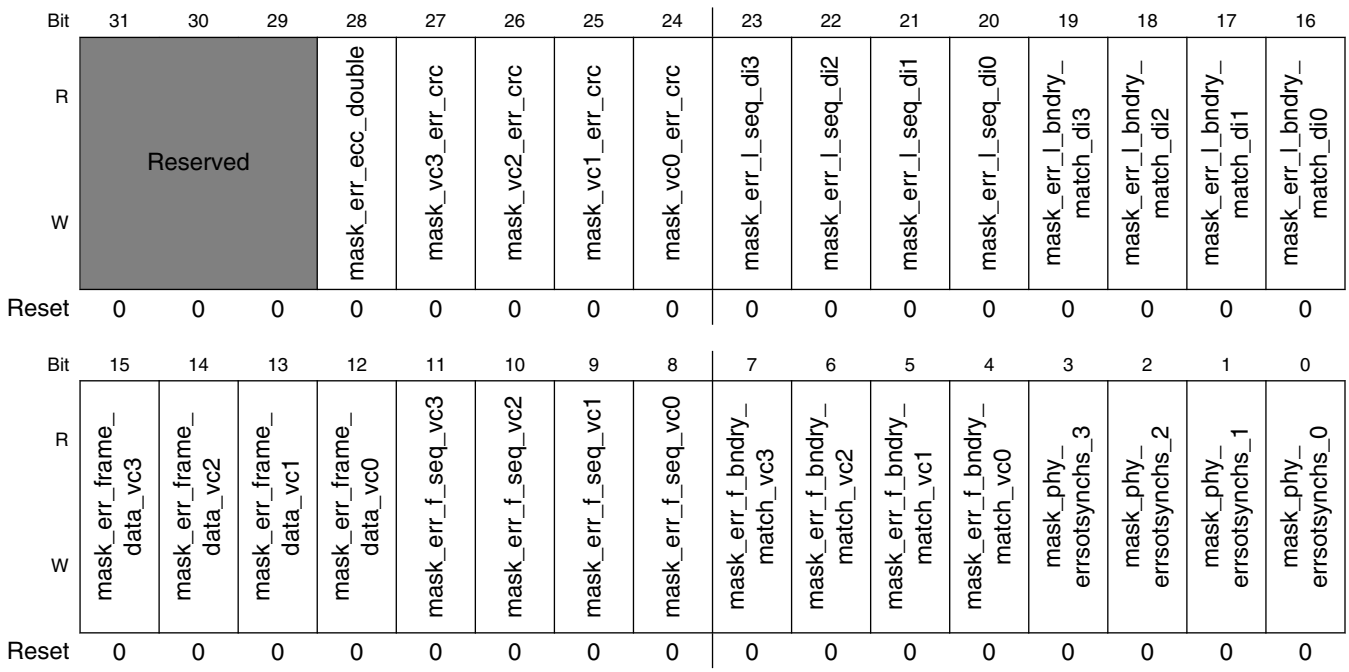
Table continues on the next page...

MIPI_CSI_ERR2 field descriptions (continued)

Field	Description
17 err_l_bndry_match_di5	Error matching Line Start with Line End for vc5 and dt5 **Default Value:** 0
16 err_l_bndry_match_di4	Error matching Line Start with Line End for vc4 and dt4 **Default Value:** 0
15 err_id_vc3	Unrecognized or unimplemented data type detected in Virtual Channel 3 **Default Value:** 0
14 err_id_vc2	Unrecognized or unimplemented data type detected in Virtual Channel 2 **Default Value:** 0
13 err_id_vc1	Unrecognized or unimplemented data type detected in Virtual Channel 1 **Default Value:** 0
12 err_id_vc0	Unrecognized or unimplemented data type detected in Virtual Channel 0 **Default Value:** 0
11 vc3_err_ecc_corrected	Header error detected and corrected on Virtual Channel 3 **Default Value:** 0
10 vc2_err_ecc_corrected	Header error detected and corrected on Virtual Channel 2 **Default Value:** 0
9 vc1_err_ecc_corrected	Header error detected and corrected on Virtual Channel 1 **Default Value:** 0
8 vc0_err_ecc_corrected	Header error detected and corrected on Virtual Channel 0 **Default Value:** 0
7 phy_errsoths_3	Start of Transmission Error on data lane 3 (synchronization can still be achieved) **Default Value:** 0
6 phy_errsoths_2	Start of Transmission Error on data lane 2 (synchronization can still be achieved) **Default Value:** 0
5 phy_errsoths_1	Start of Transmission Error on data lane 1 (synchronization can still be achieved) **Default Value:** 0
4 phy_errsoths_0	Start of Transmission Error on data lane 0 (synchronization can still be achieved) **Default Value:** 0
3 phy_erresc_3	Escape Entry Error (ULPM) on data lane 3 **Default Value:** 0
2 phy_erresc_2	Escape Entry Error (ULPM) on data lane 2 **Default Value:** 0
1 phy_erresc_1	Escape Entry Error (ULPM) on data lane 1 **Default Value:** 0
0 phy_erresc_0	Escape Entry Error (ULPM) on data lane 0 **Default Value:** 0

40.6.11 Masks for errors 1 (MIPI_CSI_MASK1)

Address: 21D_C000h base + 28h offset = 21D_C028h



MIPI_CSI_MASK1 field descriptions

Field	Description
31–29 -	This field is reserved. Reserved
28 mask_err_ecc_double	Mask for err_ecc_double. **Default Value:** 0
27 mask_vc3_err_crc	Mask for vc3_err_crc. **Default Value:** 0
26 mask_vc2_err_crc	Mask for vc2_err_crc. **Default Value:** 0
25 mask_vc1_err_crc	Mask for vc1_err_crc. **Default Value:** 0
24 mask_vc0_err_crc	Mask for vc0_err_crc. **Default Value:** 0
23 mask_err_l_seq_di3	Mask for err_l_seq_di3. **Default Value:** 0

Table continues on the next page...

MIPI_CSI_MASK1 field descriptions (continued)

Field	Description
22 mask_err_l_seq_di2	Mask for err_l_seq_di2. **Default Value:** 0
21 mask_err_l_seq_di1	Mask for err_l_seq_di1. **Default Value:** 0
20 mask_err_l_seq_di0	Mask for err_l_seq_di0. **Default Value:** 0
19 mask_err_l_bndry_match_di3	Mask for err_l_bndry_match_di3. **Default Value:** 0
18 mask_err_l_bndry_match_di2	Mask for err_l_bndry_match_di2. **Default Value:** 0
17 mask_err_l_bndry_match_di1	Mask for err_l_bndry_match_di1. **Default Value:** 0
16 mask_err_l_bndry_match_di0	Mask for err_l_bndry_match_di0. **Default Value:** 0
15 mask_err_frame_data_vc3	Mask for err_frame_data_vc3. **Default Value:** 0
14 mask_err_frame_data_vc2	Mask for err_frame_data_vc2. **Default Value:** 0
13 mask_err_frame_data_vc1	Mask for err_frame_data_vc1. **Default Value:** 0
12 mask_err_frame_data_vc0	Mask for err_frame_data_vc0. **Default Value:** 0
11 mask_err_f_seq_vc3	Mask for err_f_seq_vc3. **Default Value:** 0
10 mask_err_f_seq_vc2	Mask for err_f_seq_vc2. **Default Value:** 0
9 mask_err_f_seq_vc1	Mask for err_f_seq_vc1. **Default Value:** 0
8 mask_err_f_seq_vc0	Mask for err_f_seq_vc0. **Default Value:** 0
7 mask_err_f_bndry_match_vc3	Mask for err_f_bndry_match_vc3. **Default Value:** 0

Table continues on the next page...

MIPI_CSI_MASK1 field descriptions (continued)

Field	Description
6 mask_err_f_ bndry_match_vc2	Mask for err_f_bndry_match_vc2. **Default Value:** 0
5 mask_err_f_ bndry_match_vc1	Mask for err_f_bndry_match_vc1. **Default Value:** 0
4 mask_err_f_ bndry_match_vc0	Mask for err_f_bndry_match_vc0. **Default Value:** 0
3 mask_phy_ errsotsynchs_3	Mask for phy_errsotsynchs_3. **Default Value:** 0
2 mask_phy_ errsotsynchs_2	Mask for phy_errsotsynchs_2. **Default Value:** 0
1 mask_phy_ errsotsynchs_1	Mask for phy_errsotsynchs_1. **Default Value:** 0
0 mask_phy_ errsotsynchs_0	Mask for phy_errsotsynchs_0. **Default Value:** 0

40.6.12 Masks for errors 2 (MIPI_CSI_MASK2)

Address: 21D_C000h base + 2Ch offset = 21D_C02Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved								mask_err_l_seq_di7	mask_err_l_seq_di6	mask_err_l_seq_di5	mask_err_l_seq_di4	mask_err_l_bndry_match_di7	mask_err_l_bndry_match_di6	mask_err_l_bndry_match_di5	mask_err_l_bndry_match_di4
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	mask_err_id_vc3	mask_err_id_vc2	mask_err_id_vc1	mask_err_id_vc0	mask_vc3_err_ecc_corrected	mask_vc2_err_ecc_corrected	mask_vc1_err_ecc_corrected	mask_vc0_err_ecc_corrected	mask_phy_errsoths_3	mask_phy_errsoths_2	mask_phy_errsoths_1	mask_phy_errsoths_0	mask_phy_erresc_3	mask_phy_erresc_2	mask_phy_erresc_1	mask_phy_erresc_0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_MASK2 field descriptions

Field	Description
31–24 -	This field is reserved. Reserved
23 mask_err_l_seq_di7	Mask for err_l_seq_di7. **Default Value:** 0
22 mask_err_l_seq_di6	Mask for err_l_seq_di6. **Default Value:** 0
21 mask_err_l_seq_di5	Mask for err_l_seq_di5. **Default Value:** 0
20 mask_err_l_seq_di4	Mask for err_l_seq_di4. **Default Value:** 0
19 mask_err_l_bndry_match_di7	Mask for err_l_bndry_match_di7. **Default Value:** 0
18 mask_err_l_bndry_match_di6	Mask for err_l_bndry_match_di6. **Default Value:** 0
17 mask_err_l_bndry_match_di5	Mask for err_l_bndry_match_di5. **Default Value:** 0
16 mask_err_l_bndry_match_di4	Mask for err_l_bndry_match_di4. **Default Value:** 0
15 mask_err_id_vc3	Mask for err_id_vc3. **Default Value:** 0
14 mask_err_id_vc2	Mask for err_id_vc2. **Default Value:** 0
13 mask_err_id_vc1	Mask for err_id_vc1. **Default Value:** 0
12 mask_err_id_vc0	Mask for err_id_vc0. **Default Value:** 0
11 mask_vc3_err_ecc_corrected	Mask for vc3_err_ecc_corrected. **Default Value:** 0
10 mask_vc2_err_ecc_corrected	Mask for vc2_err_ecc_corrected. **Default Value:** 0
9 mask_vc1_err_ecc_corrected	Mask for vc1_err_ecc_corrected. **Default Value:** 0
8 mask_vc0_err_ecc_corrected	Mask for vc0_err_ecc_corrected. **Default Value:** 0

Table continues on the next page...

MIPI_CSI_MASK2 field descriptions (continued)

Field	Description
7 mask_phy_ errsoths_3	Mask for phy_errsoths_3. **Default Value:** 0
6 mask_phy_ errsoths_2	Mask for phy_errsoths_2. **Default Value:** 0
5 mask_phy_ errsoths_1	Mask for phy_errsoths_1. **Default Value:** 0
4 mask_phy_ errsoths_0	Mask for phy_errsoths_0. **Default Value:** 0
3 mask_phy_ erresc_3	Mask for phy_erresc_3. **Default Value:** 0
2 mask_phy_ erresc_2	Mask for phy_erresc_2. **Default Value:** 0
1 mask_phy_ erresc_1	Mask for phy_erresc_1. **Default Value:** 0
0 mask_phy_ erresc_0	Mask for phy_erresc_0. **Default Value:** 0

40.6.13 D-PHY Test interface control 0 (MIPI_CSI_PHY_TST_CRTL0)

Address: 21D_C000h base + 30h offset = 21D_C030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

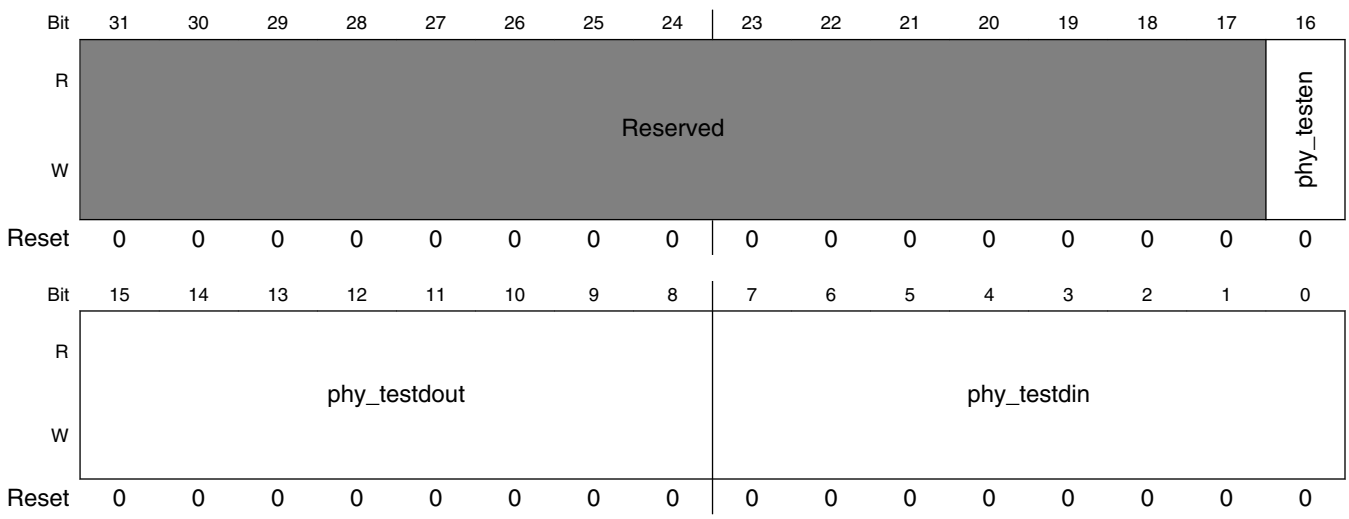
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved														phy_testclk	phy_testclr
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_CSI_PHY_TST_CRTL0 field descriptions

Field	Description
31–2 -	This field is reserved. Reserved
1 phy_testclk	PHY test interface strobe signal. Used to clock TESTDIN bus into the D-PHY. In conjunction with TESTEN signal controls the operation selection. **Default Value:** 0
0 phy_testclr	PHY test interface clear. Used when active performs vendor specific interface initialization(Active High). **Default Value:** 0

40.6.14 D-PHY Test interface control 1 (MIPI_CSI_PHY_TST_CTRL1)

Address: 21D_C000h base + 34h offset = 21D_C034h



MIPI_CSI_PHY_TST_CTRL1 field descriptions

Field	Description
31–17 -	This field is reserved. Reserved
16 phy_testen	PHY test interface operation selector: 1 configures address write operation on the falling edge of TESTCLK 0 configures a data write operation on the rising edge of TESTCLK
15–8 phy_testdout	PHY output 8-bit data bus for read-back and internal probing functionalities. **Default Value:** 0
phy_testdin	PHY test interface input 8-bit data bus for internal register programming and test functionalities access **Default Value:** 0

MIPI_CSI_PHY_TST_CTRL1 field descriptions (continued)

Field	Description

Chapter 41
MIPI DSI Host Controller (MIPI_DSI)

41.1 Overview

The DSI Host Controller is a digital core that implements all protocol functions defined in the MIPI DSI Specification, providing an interface between the System and the MIPI D-PHY, and allowing communication with a MIPI DSI-compliant Display.

41.2 Features

The MIPI DSI Host Controller supports the following features:

Compliant with MIPI Alliance Specification for Display Serial Interface (DSI), Version 1.01.00 - 21 February 2008;

- Fully Compliant with MIPI Alliance Standard for Display Pixel Interface (DPI-2), Version 2.00 15 September 2005 with Pixel Data bus width up to 24bits
- Compliant with MIPI Alliance Standard for Display Bus Interface (DBI-2) Version 2.00 - 29 November 2005. Supported DBI types are:
  - Type B
  - 16bit, 9bit and 8bit Data bus width
- DBI and DPI interface can coexist but only one is operational
- Support all commands defined in MIPI Alliance Specification for Display Command Set (DCS), Version 1.02.00 - 23 July 2009

Interface with MIPI D-PHY following PHY Protocol Interface (PPI), as defined in MIPI Alliance Specification for D-PHY, Version 1.00.00 - 14 May 2009;

Supports up to 2 D-PHY Data Lanes:

- Bidirectional Communication and Escape Mode Support through Data Lane 0.
- Programmable display resolutions, from 160x120(QQVGA) to 1280x720(XVGA).
- Multiple Peripheral Support capability, configurable Virtual Channels.

- Video Mode Pixel Formats, 16bpp(RGB565), 18 bpp(RGB666) packed, 18 bpp(RGB666) loosely, 24 bpp(RGB888).
- Supports the transmission of all generic commands;
- ECC and Checksum capabilities;
- End of Transmission Packet (EoTp) support;
- Supports ultra low power mode
- Schemes for fault recovery.

41.2.1 System Overview

The DSI Host Controller is a digital core that implements all protocol functions defined in the MIPI DSI Specification, providing an interface between the System and the MIPI D-PHY, allowing the communication with a MIPI DSI compliant Display.

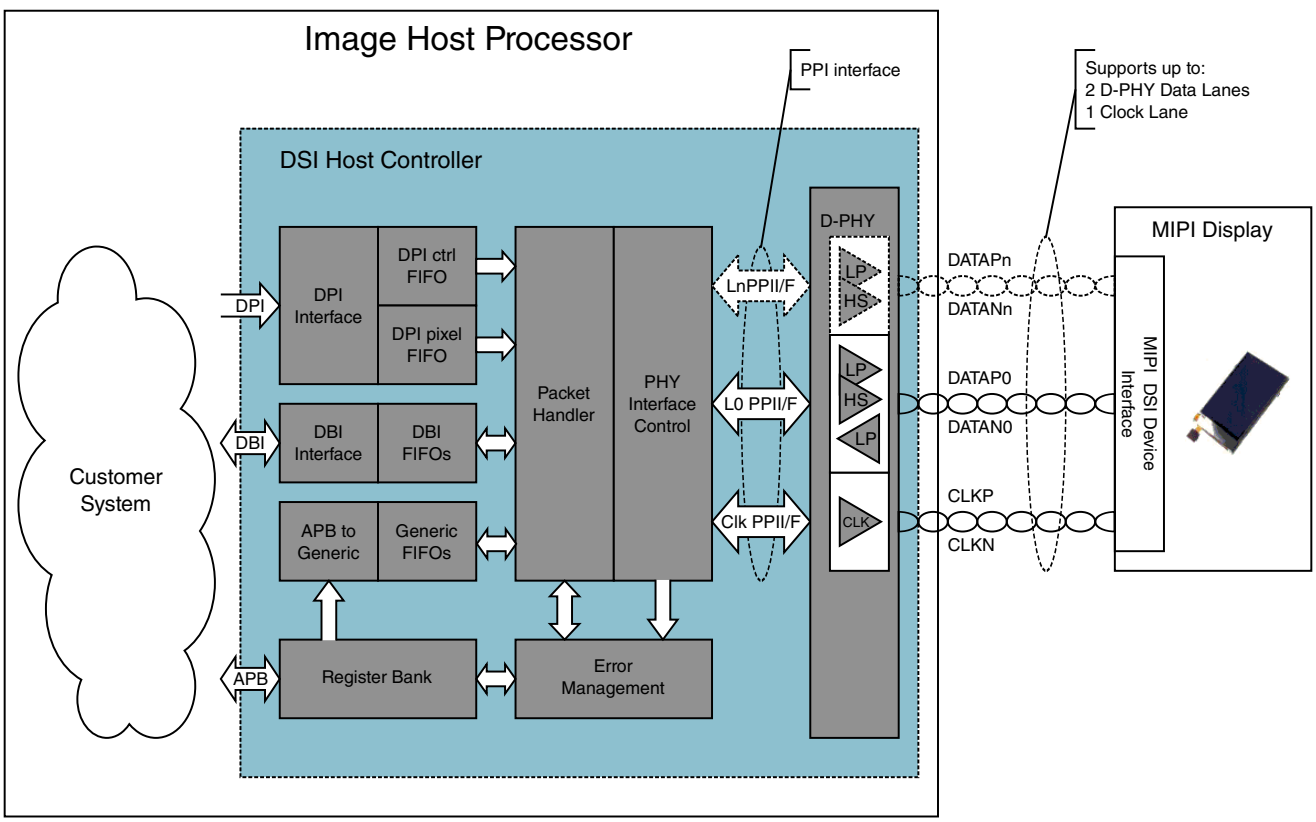


Figure 41-1. Block Diagram.

41.3 Clocks

The table found here describes the clock sources for MIPI.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 41-1. MIPI Clocks

Clock name	Clock Root	Description
ac_clk_125m	ahb_clk_root	Bus clock
pixel_clk	axi_clk_root	Pixel clock
cfg_clk	video_27m_clk_root	Configuration clock
ips_clk	ipg_clk_root	Peripheral clock
ips_clk_s	ipg_clk_root	Peripheral access clock
pll_refclk	video_27m_clk_root	Video reference clock (27MHz)

41.4 Architecture

This section describes the DSI Host Controller block interfaces, protocols, functionality, and implementation.

41.4.1 Architecture Overview

The following figure presents the overall architecture of the DSI Host Controller. The main blocks are the following:

DPI interface captures data and control signals from the DPI interface and conveys them in a FIFO for video control signals and another one for the pixel data.

DBI interface encapsulates DCS commands in DSI packets that are then conveyed into command and payload FIFOs. For commands that require a response from the device, the block uses an incoming data FIFO to acquire data from peripheral.

Register Bank is accessible through a standard AMBA-APB slave interface, providing access to the DSI Host Controller registers for configuration and control. There is also a fully programmable interrupt generator to inform the system upon certain events.

PHY Interface Control is responsible for managing the D-PHY PPI interface. It acknowledges current operation and enables LP transmission/reception or a High Speed transmission. It also performs data splitting between available D-PHY Lanes for High Speed transmission.

Packet Handler schedules activity inside the link. It performs several functions based on the interfaces that are currently operational (DPI and/or DBI) and the video transmission mode that is used (burst mode or non-burst mode with sync pulses or sync events). It

builds long or short packet generating correspondent ECC and CRC codes. This block also handles packet reception: Validates packet header by checking ECC, performs header correction for single bit erros and notifies and abort reception for multiple header errors. Depending on the packet type, generic read response or DCS command, route the output data to the respective port (generic or DBI)

APB to Generic block bridges APB operations into FIFOs holding the Generic commands. The block interfaces with 3 FIFOS, a command FIFO, a write payload FIFO and a read payload FIFO.

Error Management notifies and monitors error conditions on the DSI link. It controls timers used to determine if a time out condition occurred and triggers interruption for errors.

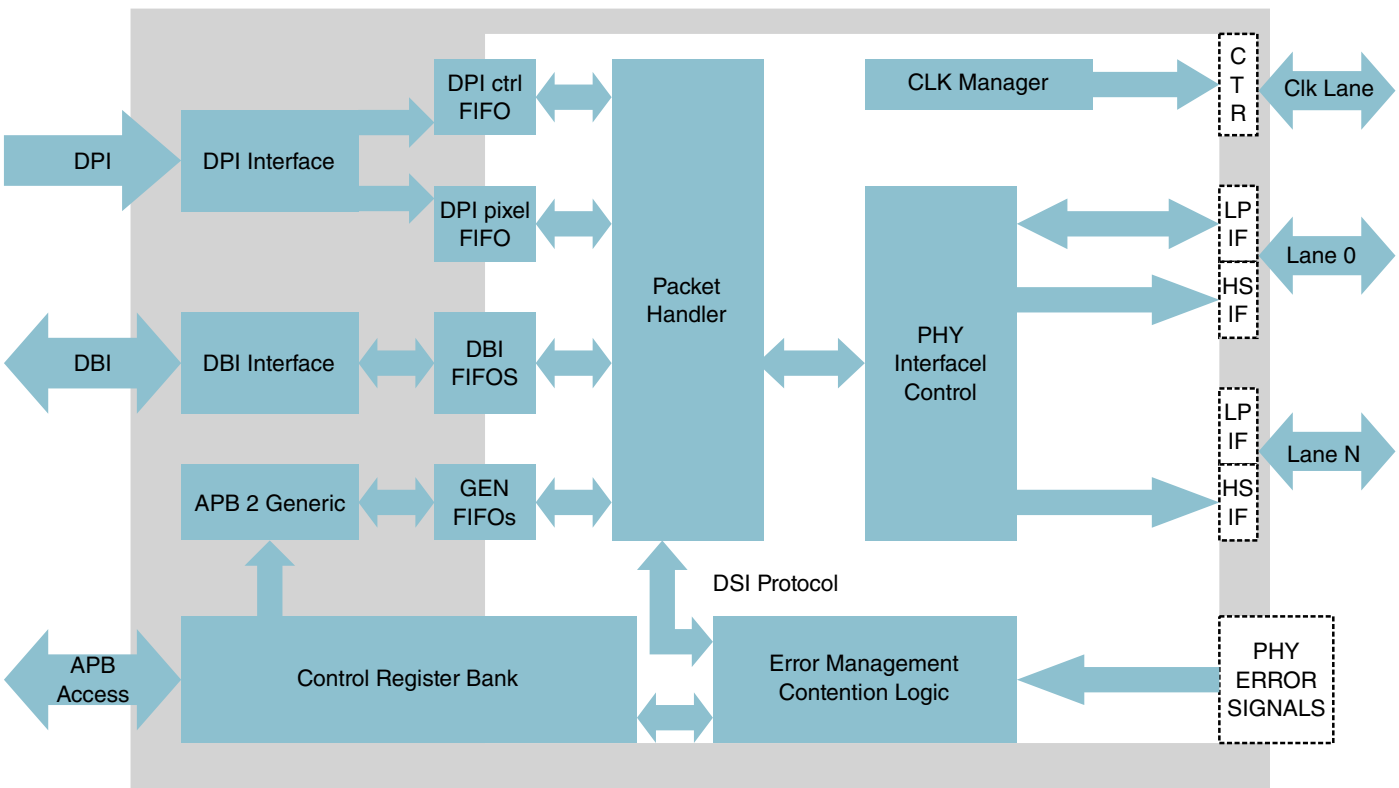


Figure 41-2. DSI Host Controller Architecture

41.4.2 DBI-2 interface

The DBI-2 interface encapsulates DCS commands in DSI packets to be transmitted through the D-PHY link. Some commands require a response from the device and the interface provides read data from the device/peripheral.

The support DBI types is Type B interface

The selection type maps the related pins on the DSI core pinout. DBI interface used RGB additive color mixing method and the pixel data format is selected on register DBI_CFG register. The field **in_dbi_conf** defines the input pixel data format and field **out_dbi_conf** defines the output pixel data that is determined by the display device. This interface needs to be associated with a particular DSI Virtual Channel that is programmed on field **dbi_vid**.

The following figures show how the pixel to byte conversion is done for each mode. All the diagrams are representing the case were the interface is working as type B.

The DBI interface type B signals list .

DBICSX: Chip select, it is low active .

DBIRDX: Read signal, the read data should be captured at the DBIRDX rising edge .

DBIWRX: Write signal, the write data will be switch at DBIWRX falling edge.

DBIDATIN[15:0]: Write data

DBIDCX: Data commnad select, Data is indicated when high and command is indicated when low .

DBITE: Tearing effect.

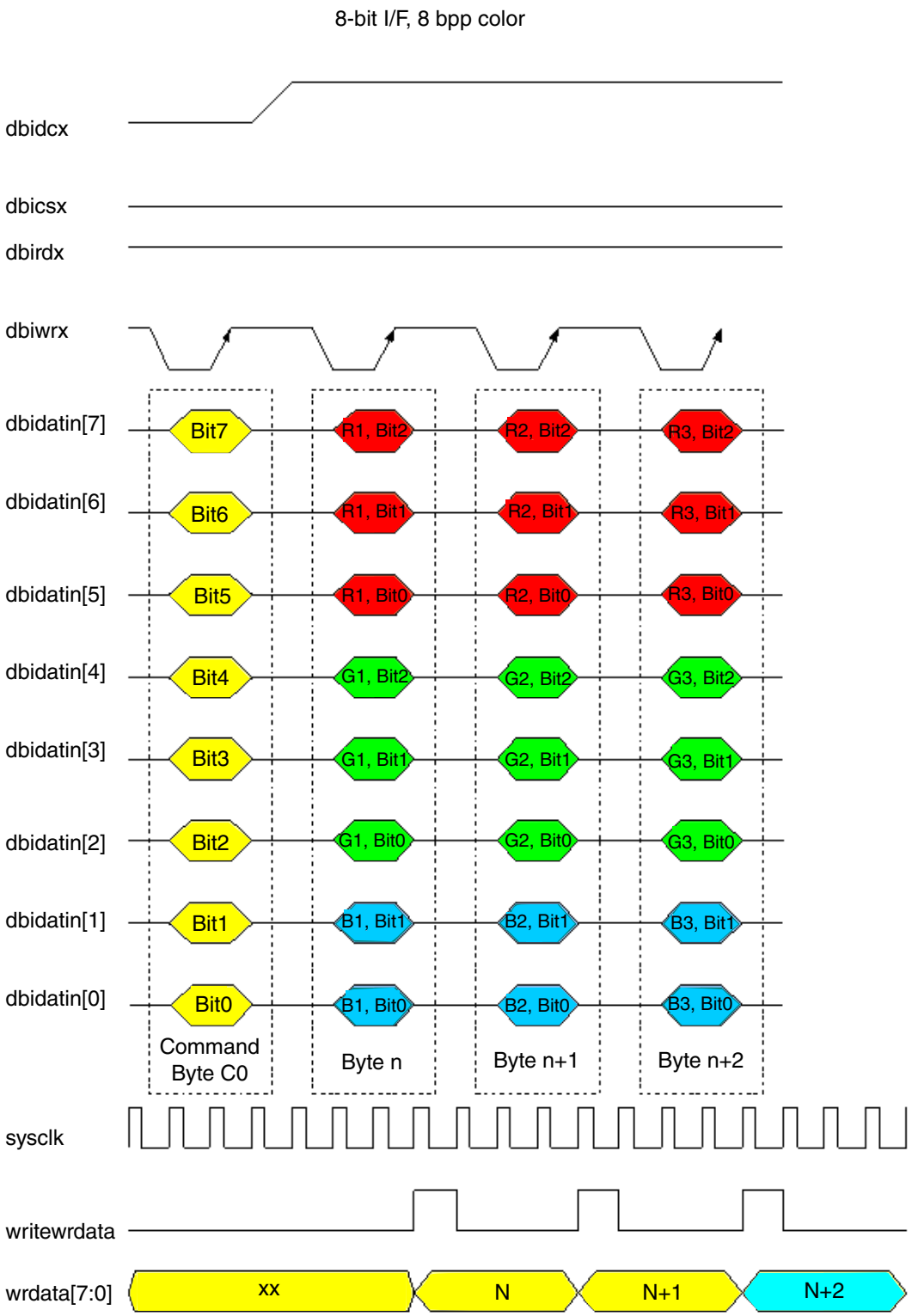


Figure 41-3. DSI 8 bit /8bpp byte write

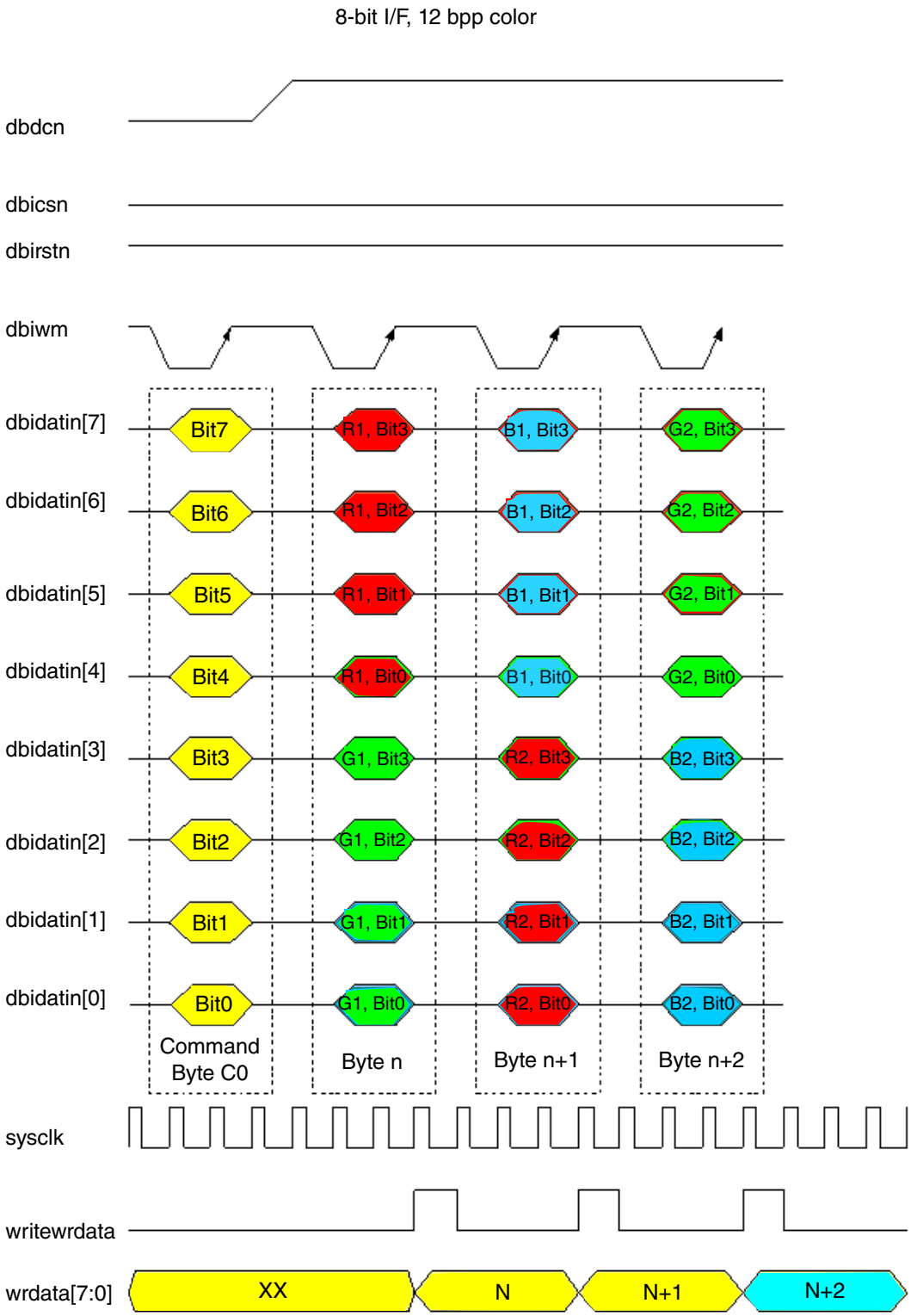


Figure 41-4. DSI 8 bit/12bpp byte write

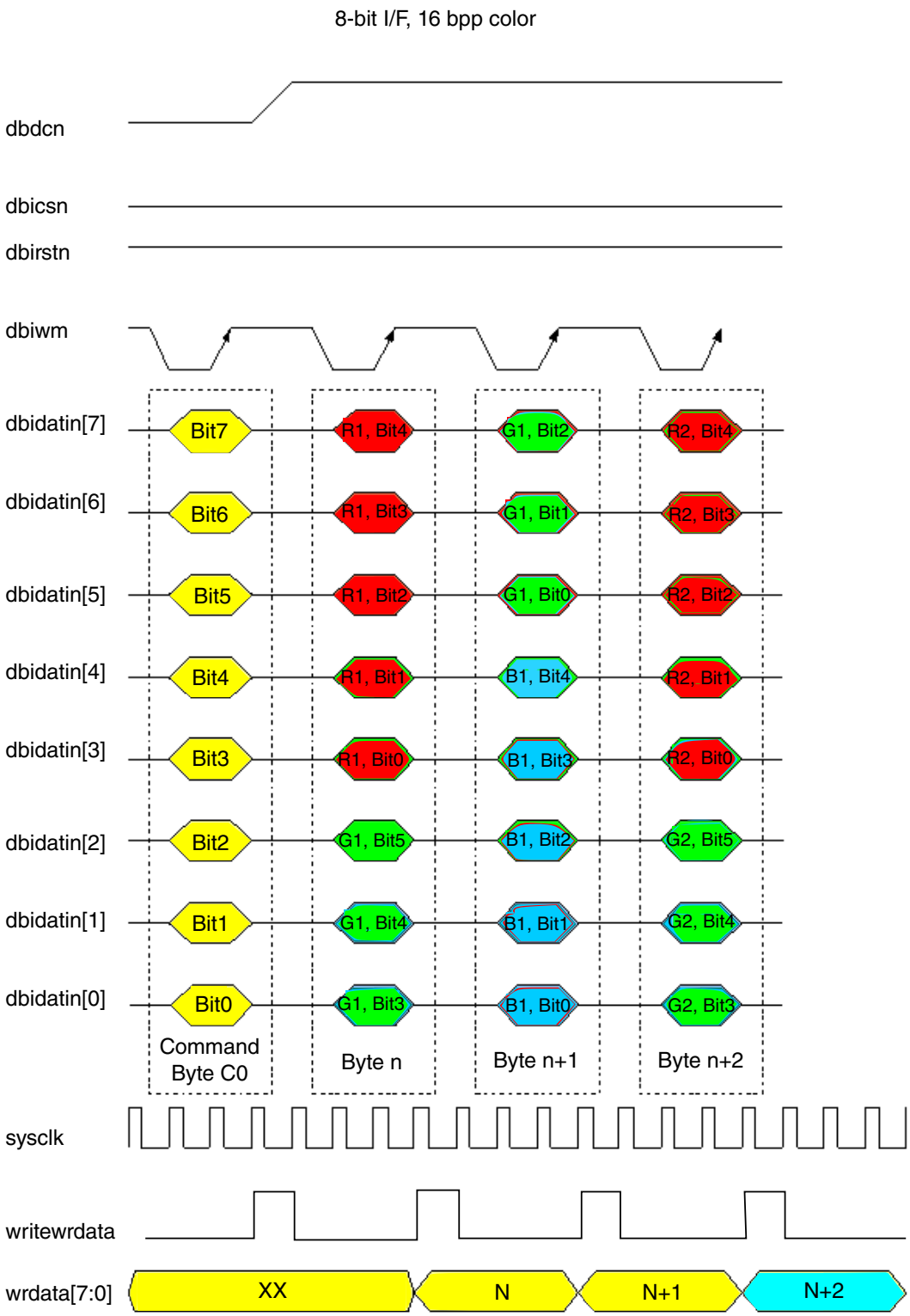


Figure 41-5. DSI 8 bit/16bpp byte write

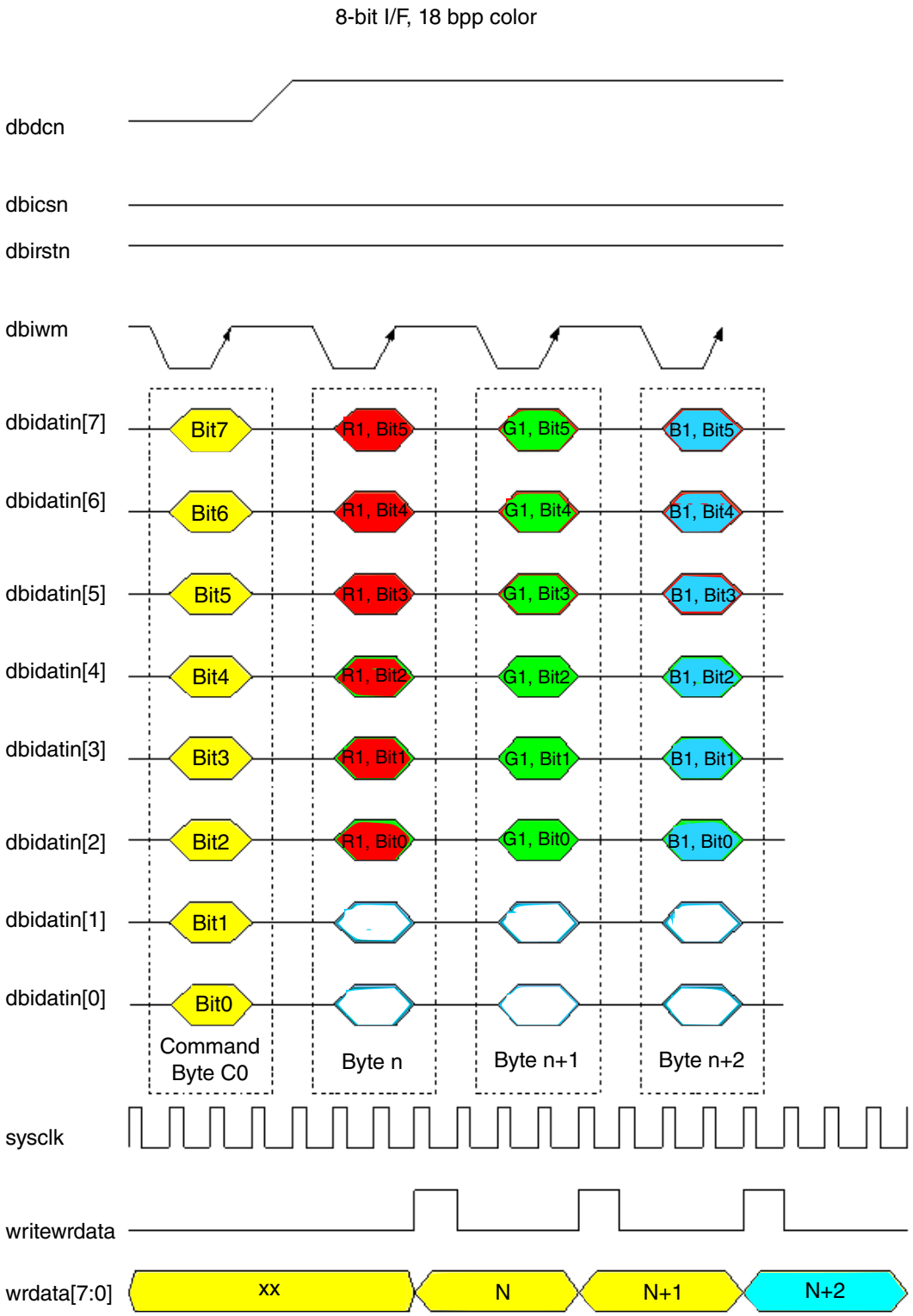


Figure 41-6. DSI 8 bit/18bpp byte write

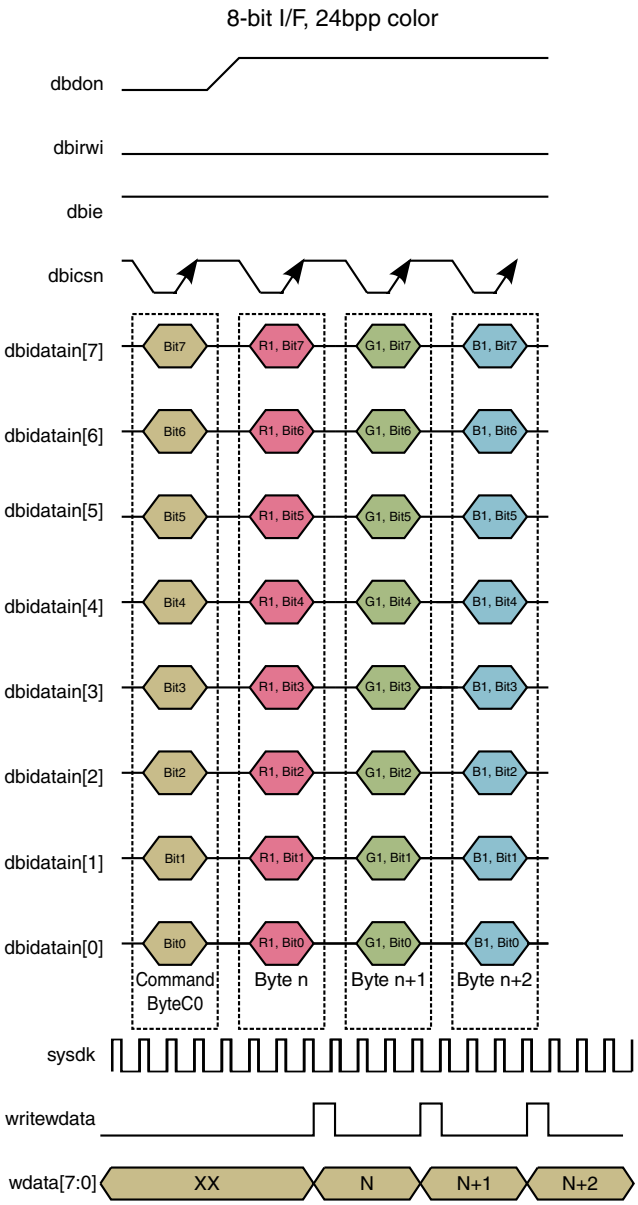


Figure 41-7. DSI 8 bit/24bpp byte write

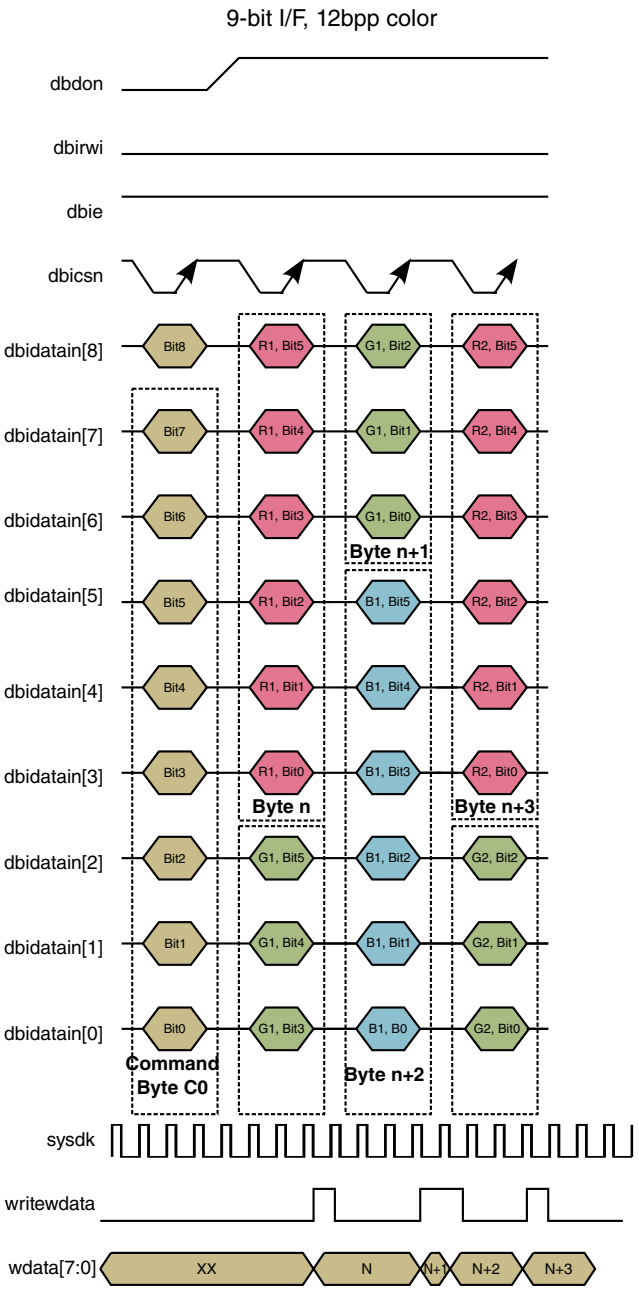


Figure 41-8. DSI 9 bit/12 byte write

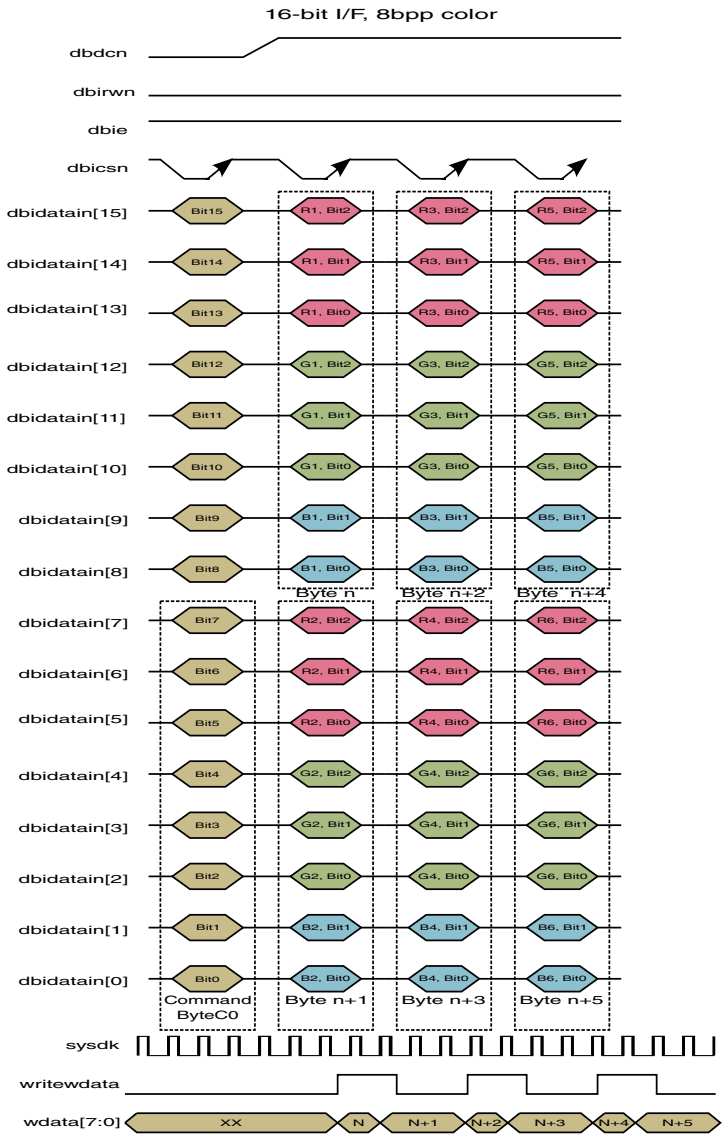


Figure 41-9. DSI 16 bit/8bpp byte write

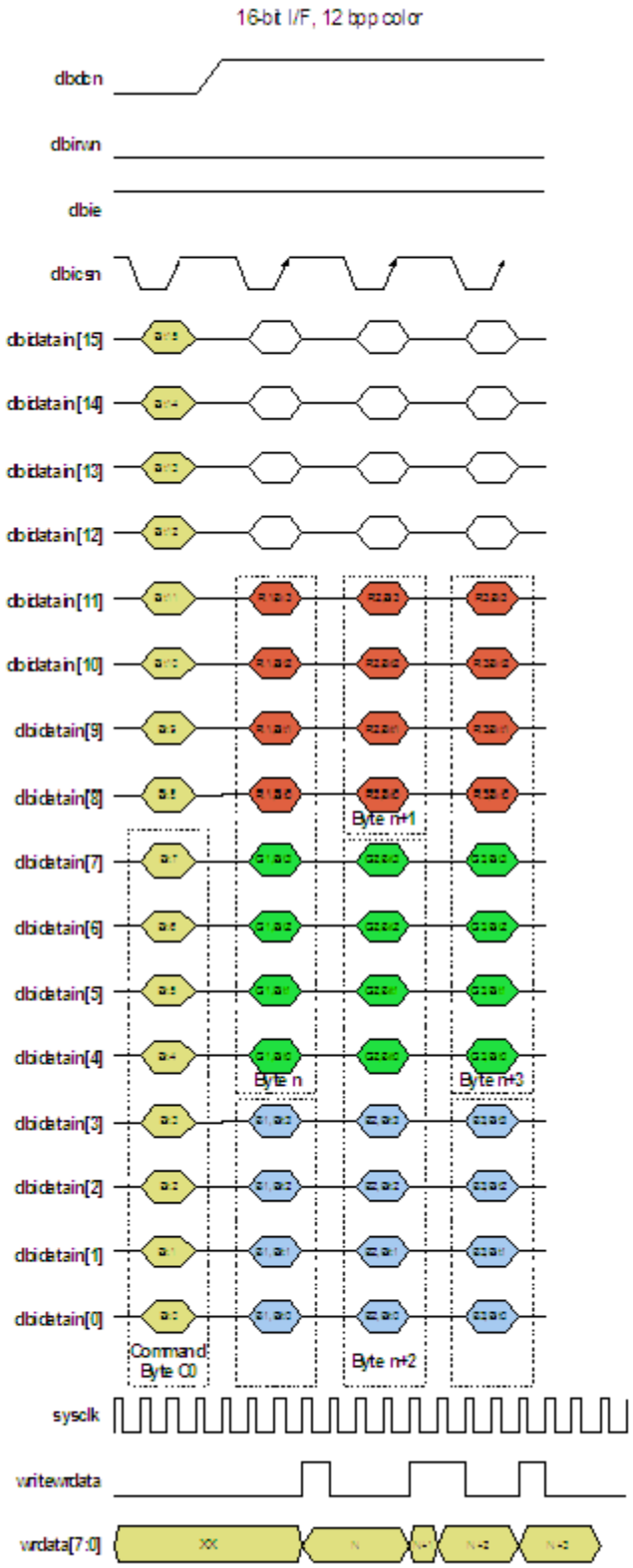


Figure 41-10. DSI 16 bit/12bpp byte write

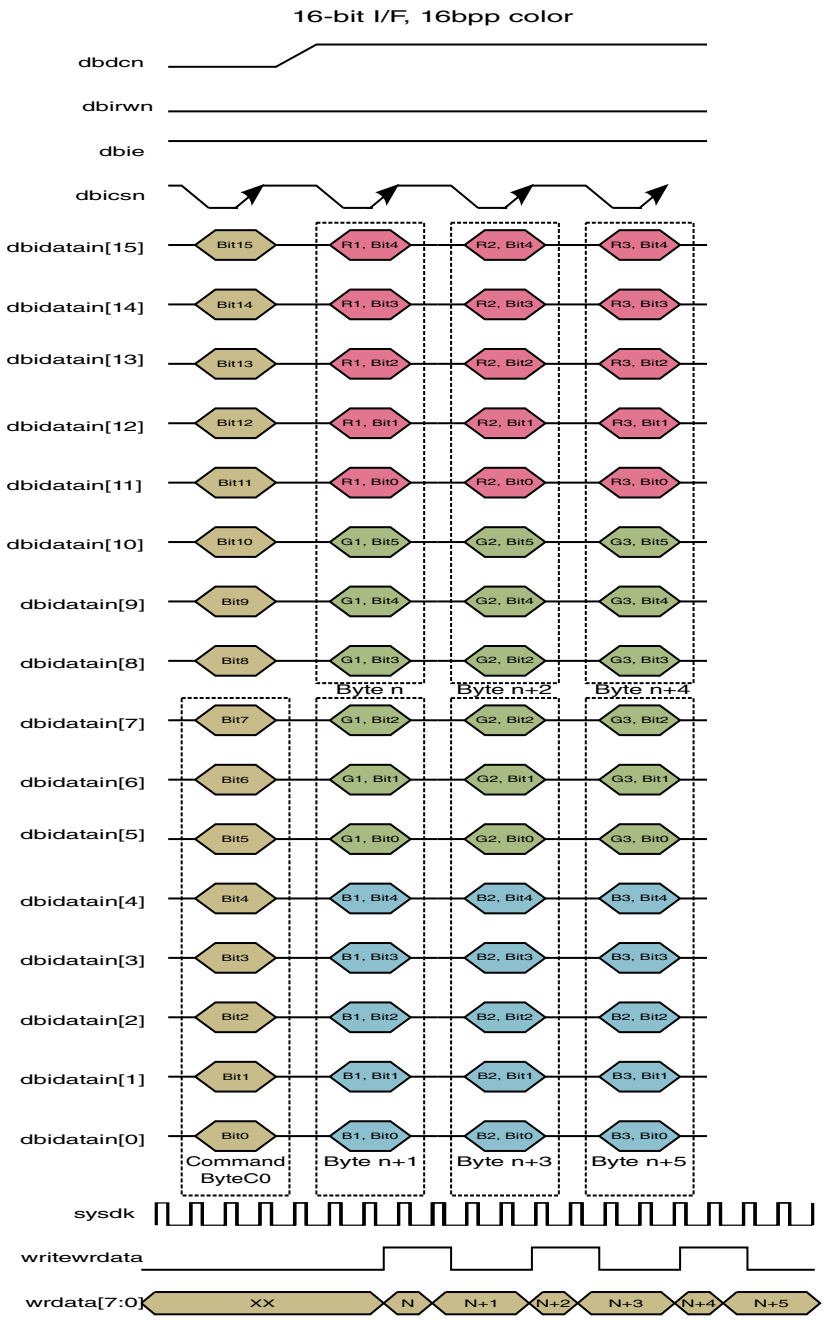


Figure 41-11. DSI 16 bit/16bpp byte write

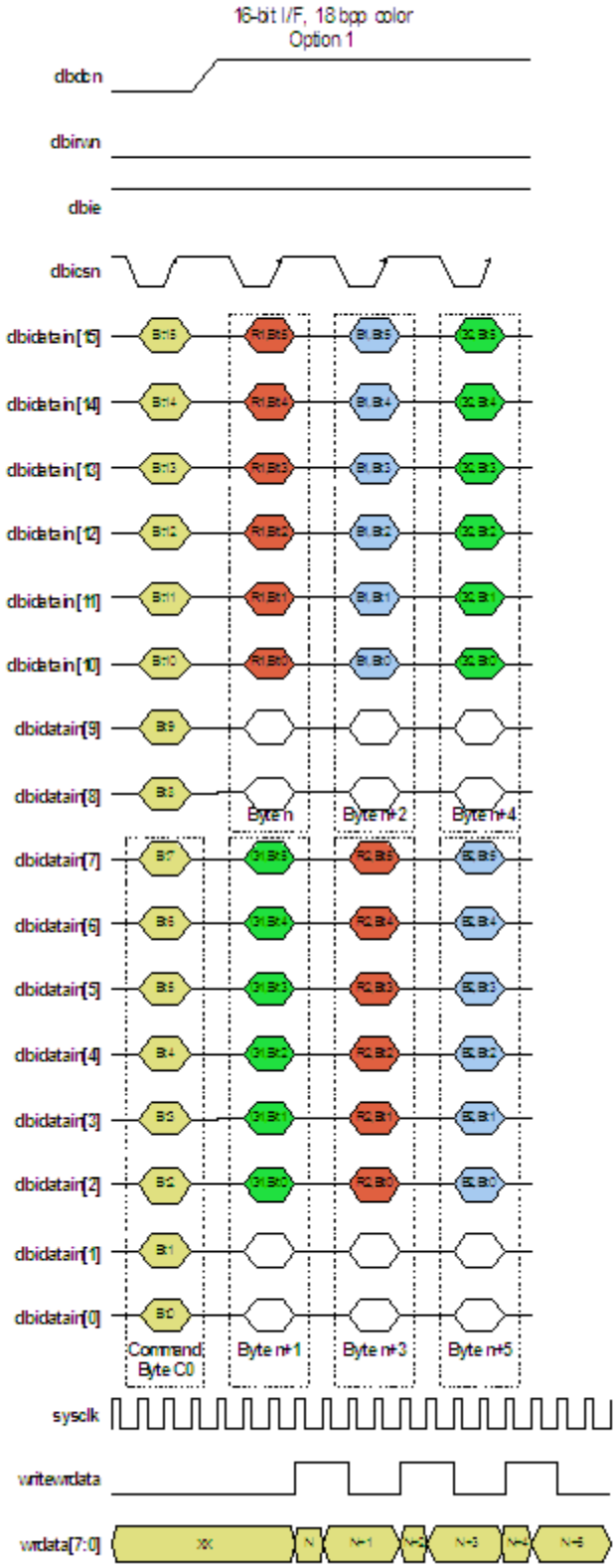


Figure 41-12. DSI 16 bit/18bpp option 1 byte write

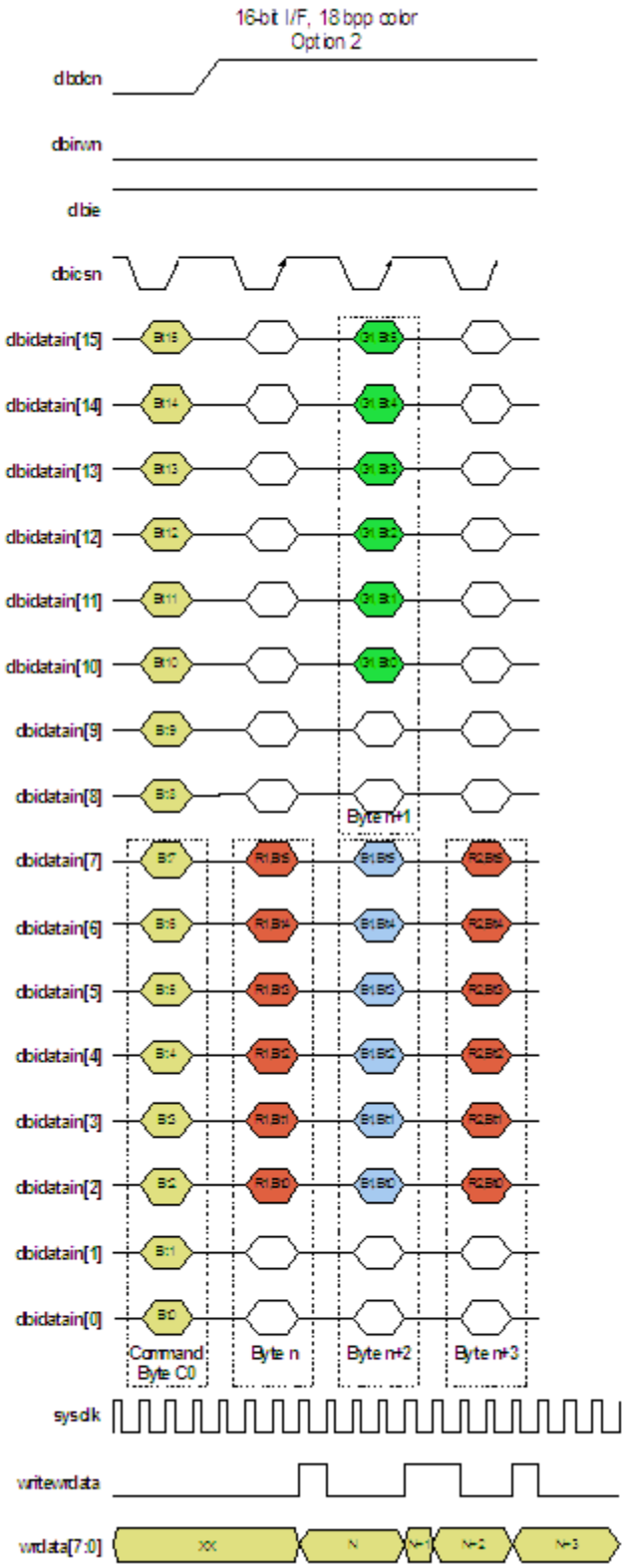


Figure 41-13. DSI 16 bit/18bpp option 2 byte write

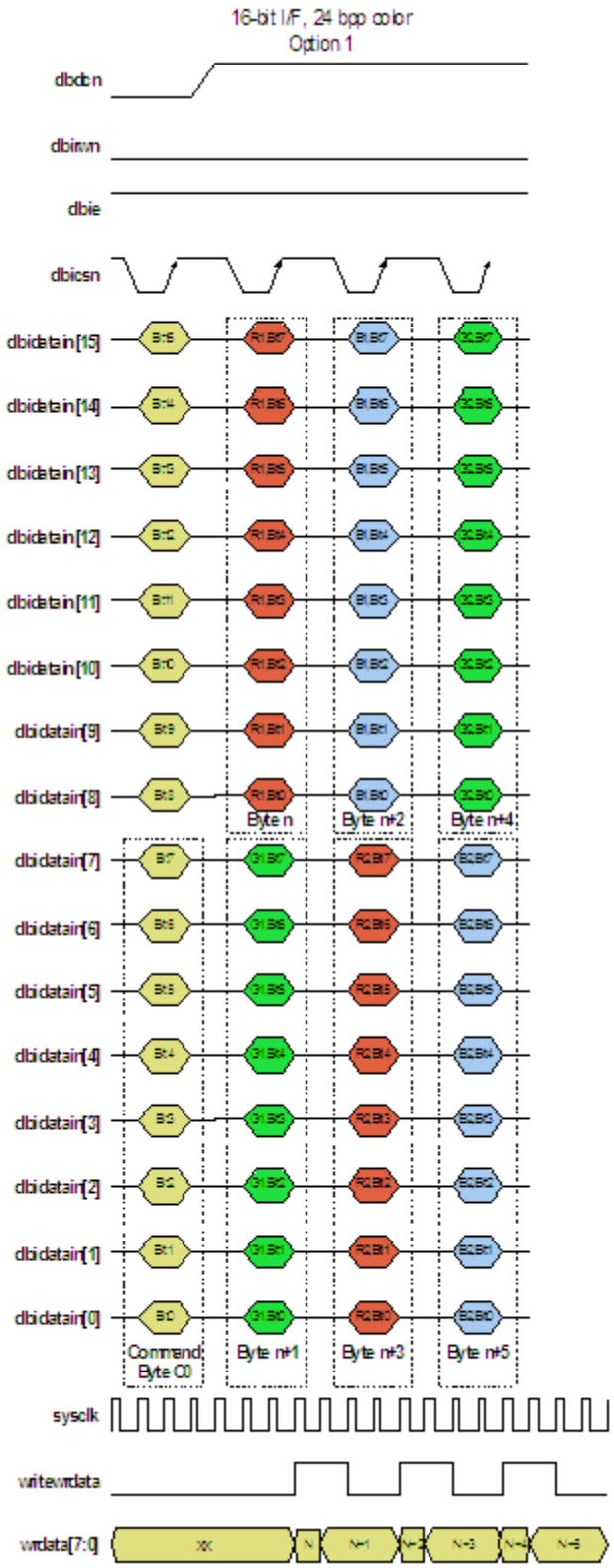


Figure 41-14. DSI 16 bit/24bpp option 1 byte write

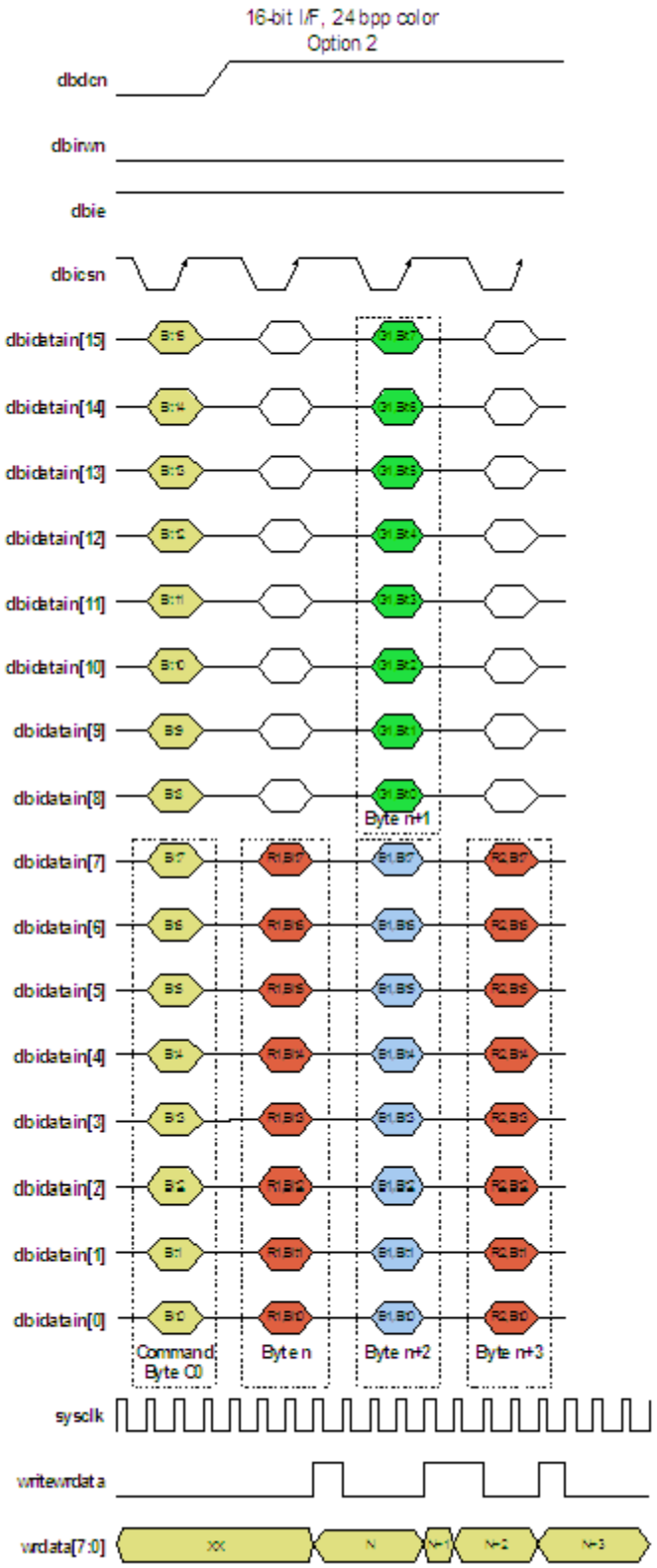


Figure 41-15. DSI 16 bit/24bpp option 2 byte write

DBI interface receives commands and decodes information for the command packetizing format. Depending on the command type, DSI core output the related DSI packet type (DCS short write packet; DCS read packet or DCS long write). If the DCS commands have variable size this needs to be configured on the DBI_CMDSIZE register. The following table presents DCS packet types:

Table 41-2. DSI packet types for DCS commands

DCS command	Command ID	DCS parameter	DSI packet type
enter_idle_mode	0x39h	0	6'h05
enter_invert_mode	0x21h	0	6'h05
enter_normal_mode	0x13h	0	6'h05
enter_partial_mode	0x12h	0	6'h05
enter_sleep_mode	0x10h	0	6'h05
exit_idle_mode	0x38h	0	6'h05
exit_invert_mode	0x20h	0	6'h05
exit_sleep_mode	0x11h	0	6'h05
nop	0x00h	0	6'h05
set_display_off	0x28h	0	6'h05
set_display_on	0x29h	0	6'h05
set_tear_off	0x34h	0	6'h05
soft_reset	0x01h	0	6'h05
set_address_mode	0x36h	1	6'h15
set_gamma_curve	0x26h	1	6'h15
set_pixel_format	0x3Ah	1	6'h15
set_tear_on	0x35h	1	6'h15
set_scroll_start	0x37h	2	6'h39
set_tear_scan_line	0x44h	2	6'h39
set_column_address	0x2Ah	4	6'h39
set_page_address	0x2Bh	4	6'h39
set_partial_area	0x30h	4	6'h39
set_scroll_area	0x33h	6	6'h39
write_LUT	0x2Dh	N	6'h39
write_memory_continue	0x3Ch	N	6'h39
write_memory_start	0x2Ch	N	6'h39
get_address_mode	0x0Bh	1(read)	6'h06
get_blue_channel	0x08h	1(read)	6'h06
get_diagnostic_result	0x0Fh	1(read)	6'h06
get_display_mode	0x0Dh	1(read)	6'h06
get_green_channel	0x07h	1(read)	6'h06
get_pixel_format	0x0Ch	1(read)	6'h06
get_power_mode	0x0Ah	1(read)	6'h06
get_red_channel	0x06h	1(read)	6'h06

Table continues on the next page...

Table 41-2. DSI packet types for DCS commands (continued)

DCS command	Command ID	DCS parameter	DSI packet type
get_scan_line	0x45h	2(read)	6'h06
get_signal_mode	0x0Eh	1(read)	6'h06
read_DDB_continue	0xA8h	n(read)	6'h06
read_DDB_start	0xA1h	n(read)	6'h06
read_memory_continue	0x3Eh	n(read)	6'h06
read_memory_start	0x2Eh	n(read)	6'h06

All packets that have 1 or 0 parameters index to the command are considered short packet and all other shall be considered long packets.

There are several register fields to configure DBI input pixel data and DBI pixel data retrieval. It is required to have prior knowledge of display supported feature to avoid incorrectly core configuration. Some guidelines are presented below to clarify how core uses the configuration for its operation.

Field **in_dbi_conf** in register DBI_CFG configures the color depth and the pixel format used by the input interface. Field **out_dbi_conf** configures the colour depth and the pixels format used by the output interface. Here the colour depth is directly related with formats supported by the peripheral. If the peripheral returns pixel data with a specific colour depth this colour depth should be consistent with the register configuration. If this is not respected, the byte to pixel converter will use an incorrect pixel mapping and it will corrupt output data.

Field **out_dbi_conf** should not be programmed for a higher number of interface pins than **in_dbi_conf**. As an example an incorrect pin configuration happens if **in_dbi_conf** defines 9 bit interface and **out_dbi_conf** defines 16 bit since pins 16 to 10 will not be available.

Field **lut_size_conf** configures the size of the DCS write_LUT command; this command will program the peripheral's colour space conversion table. This register is directly related with the capabilities of the peripheral. If the peripheral supports a colour depth of 16/18/24 bits per pixel it is expected that this register is configured for 48/128/192 bytes respectively. These coefficients should be loaded to the peripheral before normal operation of pixel data transmission is started. The DCS specification only refers to the conversion of formats equal or above 12bpp. This means that any format outside this scope might not be convertible to a higher colour depth mode by the peripheral. Taking this in consideration, when **in_dbi_conf** selects a colour depth below 12bpp, the **out_dbi_conf** should be configured such that colour depth has not a higher value. As an example, if **in_dbi_conf** is configured for 8bits/8bpp interface and **out_dbi_conf** is configured for 8bits/12bpp, this results in an incorrect configuration because the DCS

specification does not contemplates conversions from 8bpp to 12bpp. This configuration is possible inside the core but might lead into incorrect results. Nevertheless it is left for the system to configure these registers according with the desired operation.

For Read back commands the DBI specification states that the reading process does not use any acknowledge mechanism to know that the requested data is ready. There are mechanisms for the system to get this information by polling the Command Status Register CMD_PKT_STATUS. This register, besides other information, contains the status on the DBI read back payload FIFO and also contains a special busy flag **dbi_rd_cmd_busy** that can be used to understand if the read command is complete. By using the DBI read back FIFO status bits that report if the FIFO is full or empty the processor can read these bits to understand if it can read another byte/sample from the interface (probing the empty flag) or when it should stop reading bytes/samples from the interface. In addition, the Read Command Busy flag is asserted when a read command is issued by the interface and cleared once the reading process ends and the read data is fully stored inside the read back FIFO. The system needs to issue the read command and then periodically probe this status bit to acknowledge if the read back operation is complete. Once this information is available the system can assume that the entire data is stored inside the read back FIFO and the interface can deliver all the desired data.

41.4.3 DPI-2 Interface

The DPI-2 interface captures data and control signals and conveys them to FIFO interfaces that will transmit them to the DSI link. Two different streams of data are presented at the interface: video control signals and pixel data. Depending on the interface color coding, the pixel data is handled differently throughout the dpipixdata bus.

Interface pixel color coding is presented in the table below.

Table 41-3. DPI interface pixel color coding mapping.

Signal Line	16-bit			18-bit		24-bit
	Config1	Config2	Config3	Config1	Config2	
D23	Not used	Not used	Not used	Not used	Not used	R7
D22	Not used	Not used	Not used	Not used	Not used	R6
D21	Not used	Not used	R4	Not used	R5	R5
D20	Not used	R4	R3	Not used	R4	R4
D19	Not used	R3	R2	Not used	R3	R3
D18	Not used	R2	R1	Not used	R2	R2
D17	Not used	R1	R0	R5	R1	R1
D16	Not used	R0	Not used	R4	R0	R0
D15	R4	Not used	Not used	R3	Not used	G7

Table continues on the next page...

Table 41-3. DPI interface pixel color coding mapping. (continued)

Signal Line	16-bit			18-bit		24-bit
	Config1	Config2	Config3	Config1	Config2	
D14	R3	Not used	Not used	R2	Not used	G6
D13	R2	G5	G5	R1	G5	G5
D12	R1	G4	G4	R0	G4	G4
D11	R0	G3	G3	G5	G3	G3
D10	G5	G2	G2	G4	G2	G2
D9	G4	G1	G1	G3	G1	G1
D8	G3	G0	G0	G2	G0	G0
D7	G2	Not used	Not used	G1	Not used	B7
D6	G1	Not used	Not used	G0	Not used	B6
D5	G0	Not used	B4	B5	B5	B5
D4	B4	B4	B3	B4	B4	B4
D3	B3	B3	B2	B3	B3	B3
D2	B2	B2	B1	B2	B2	B2
D1	B1	B1	B0	B1	B1	B1
D0	B0	B0	Not used	B0	B0	B0

The DPI interface can be configured to increase flexibility and promote correct usage of this interface for several systems. These configuration options are described below:

- Polarity control. All the control signals are programmable to change polarity according to system requirements.
- After core reset, DPI waits for the first VSYNC active transition to start signals sampling, including pixel data and preventing image transmission in the middle of a frame.
- If interface pixel color coding is 18 bits and the 18 bit loosely packet stream is enabled, the number of lines programmed in the pixels per lines configuration is a multiple of four. This means that in this mode the 2 LSB in the configuration will be always inferred as zero. Specification states that in this mode the pixel line size should be a multiple of 4.
- To avoid pixel underflows and overflows the configured number of pixel will originate the respective memory spaces. This will happen even if the dpidataen pin is active during more or less time than necessary.
- In order to keep the memory organized in respect to the packet scheduling the number of pixels per packet parameter will be used to separate the memory space of different video packets.

For the **dpishutd** and **dpicolorm** sampling and transmission the DPI video signalling must be active. This means that if video signals like VSYNC and HSYNC are not being actively generated the DPI commands these are not transmitted through the DSI link. The

DSI commands packets have the highest priority in the class of Non Video packets. Because of such constraint and in order for commands to be correctly transmitted, the first VSYNC active pulse needs to happen for the commands sampling and transmission. When shutting down the display, it is necessary for DPI video to keep active for one frame after the command being issued. This guarantees that the commands are correctly transmitted before actually disabling the Video generation in the DPI interface.

41.4.4 Error control and timeout timers

The DSI host controller implements a set of timers and conditions to notify errors.

The core has a set of registers to control the timers that will be used to determine if a time out has occurred. It also contains a set of interruption status registers that are cleared upon read operation. These registers also trigger an interruption pin that can be used by the system to act upon an error within the DSI connection.

41.4.5 Timeout timers

Bit **to_hs_tx** on register ERROR_ST1 is set when a High Speed Transmission Time Out occurs rising an interrupt pin.

The time out is configured in **hstx_to_cnt** field of register TO_CNT_CFG. A 16 bit counter measures the time when High Speed is active. If that counter reaches the value defined by register **hstx_to_cnt**, bit **to_hs_tx** is asserted.

Bit to_lp_rx on register ERROR_ST1 is set signaling contention detection when a Low Power Reception Time Out occurs rising an interrupt pin. The time out is configured in **lprx_to_cnt** field of register TO_CNT_CFG. A 16 bit counter measures the time when Low Power reception is active. If that counter reaches the value defined by register **lprx_to_cnt**,bit **to_lp_rx** is asserted.

Time units for this 16bit counters are configured in cycles defined in **TO_CLK_DIVISION** field in register CLKMGR_CFG. The value written to **TO_CLK_DIVISION** defines the time unit for the Time out limits using Lane byte clock as input. This mechanism increases the range to define these limits.

41.4.6 Error control

Two registers ERROR_ST0 and ERROR_ST1 are associated with error condition reporting. This resister can trigger interrupt pins to inform the system about the occurrence of errors.

The coreConsultant GUI allows different interrupt pin configuration that can support:

- No interrupt pin on DSI core for notification of error although system can check for error by reading error status registers ERROR_ST0 and ERROR_ST0.
- DSI core has one interrupt pin interrupt that is set High when an error occurs either in register ERROR_ST1 or ERROR_ST2.
- DSI core has 2 interrupts associated with each error status register. Pins interrupt0 and interrupt1 are respectively associated with ERROR_ST1 and ERROR_ST2. This mechanism allows faster association of interrupts with the status register that generated the error.

The triggering of the interrupt pins, when defined, can be masked by programming the mask registers ERROR_MSK0 and ERROR_MSK1. When any bit of this registers is set to 1 it will inhibit the interrupt for that specific error. Nevertheless the error bit will always be set on the respective ERROR_ST register. Registers ERROR_ST1 and ERROR_ST2 are always cleared when read.

The figure below illustrates the location of some of the errors.

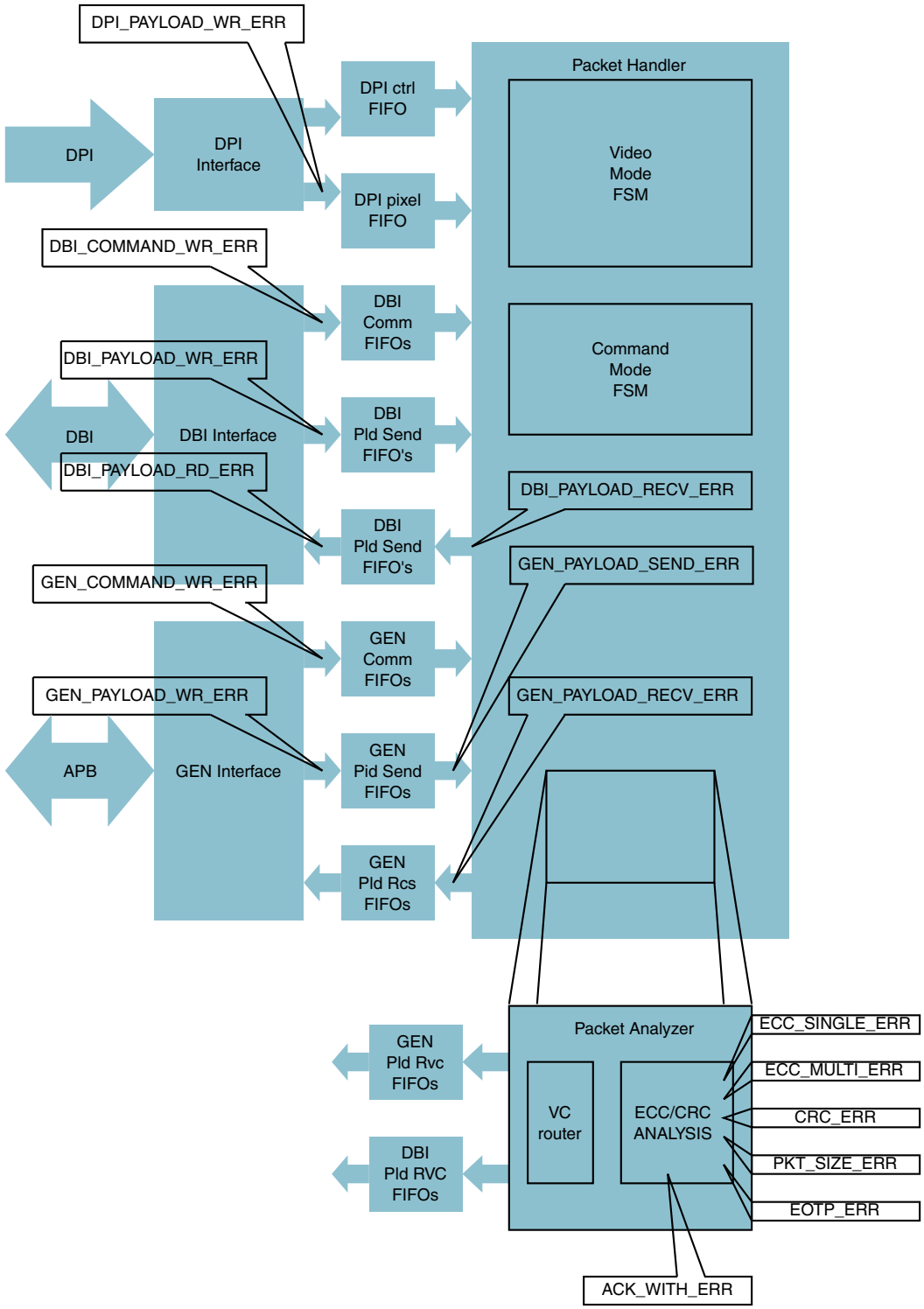


Figure 41-16. Error sources diagram

41.4.7 Generic packets

DSI host Core support generic packets transmission and reception as described in the DSI-2 specification. These packets are channeled through the APB register bank.

Register GEN_PLD_DATA is used to write payload of generic packets and when read it returns the payload of a read back operation. Register GEN_HDR contains the generic packet header type and header data. Writing to this register triggers the transmission of the packet implying that for long generic packets data needs to be written in advance on register GEN_PLD_DATA.

The valid packets available to be transmitted through the Generic interface are:

Table 41-4. Generic interface available packets

Generic Write Short Packet 0 Parameters	Generic Write Short Packet 1 Parameters	Generic Write Short Packet 2 Parameters	Generic Read Short Packet 0 Parameters
Generic Read Short Packet 1 Parameters	Generic Read Short Packet 2 Parameters	Maximum Read Packet Configuration	Generic Long Write Packet
DCS Write Short Packet 0 Parameters	DCS Write Short Packet 1 Parameters	DCS Read Short Packet 0 Parameters	DCS Write Long Packet

A set of bits in register CMD_PKT_STATUS report the status of the FIFOs associated with generic packet support.

Generic packets are always transported using one of the DSI transmission modes: Video Mode or Command Mode. If none of these modes are selected the packets will not be transmitted through the link and the related FIFOs will eventually get overflowed.

If system does not requires Generic packets support this functionality can be disabled in coreConsultant GUI and the interfaces with the 2-port RAMs that buffer command, payload and read data are removed.

41.4.8 Signals

A Clock Lane and a Data Lane (Lane0) with TX HS features are always required. Data Lanes can be up to a maximum of 2.

Table 41-5. D-PHY external interface

Pin Name	Width	Direction	Description
DPHY Supply			
AVDD	1	Input	D-PHY Analog power supply.
VDD	1	Input	D-PHY Digital power supply.
AVDDREF	1	Input	D-PHY Analog supply for reference generator.

Table continues on the next page...

Table 41-5. D-PHY external interface (continued)

Pin Name	Width	Direction	Description
AGND	1	Input	D-PHY Analog supply ground return.
VSS	1	Input	D-PHY Digital supply ground return
AGNDREF	1	Input	D-PHY Analog supply ground return for reference generator.
DPHY external pins			
REFCLK	1	Input	D-PHY Reference clock used for Master-side serial clock generation in Clock Multiplying Unit (PLL)
CFG_CLK	1	Input	D-PHY Configuration clock used for the initialization of the PHY. It is also used for exiting ULPS state.
REXT	1	Input	D-PHY External resistor connection (REXT and REXTV should be shorted at the chip pad connection)
REXTV	1	Input	D-PHY External resistor connection (REXT and REXTV should be shorted at the chip pad connection)
CLKP	1	Input	D-PHY Positive D-Phy differential clock line transceiver output.
CLKN	1	Input	D-PHY Negative D-Phy differential clock line transceiver output.
DATAP0	1	Input	D-PHY Positive D-Phy differential data line transceiver output, Lane 0
DATAN0	1	Input	D-PHY Negative D-Phy differential data line transceiver output, Lane 0
DATAP1	1	Input	D-PHY Positive D-Phy differential data line transceiver output, Lane 1
DATAN1	1	Input	D-PHY Negative D-Phy differential data line transceiver output, Lane 1

41.5 Programming

41.5.1 DSI and D-PHY initialization sequence

This chapter describes the procedure for DSI and D-PHY initialization. This process is based on APB register interface access.

- By default register PHY_RSTZ is activating the PHY resets **physhutdownz**, **phyrstz** and disabling **enableclk** and register PHY_TEST_CTRL0 is by default asserting the **testclr** pin. All the PHY reset pins are being activated by default.
- Configure Register PHY_IF_CFG with correct the number of lanes to be used by the controller.
- Configure the TX_ESC clock frequency to a frequency lower than 20MHz that is the maximum allowed frequency for D-PHY ESCAPE mode. This is done by writing in Register CLKMGR_CFG, field TX_ESC_CLK_DIVISION.
  TX_ESC_CLK_DIVISION divides Byte Clock and generates a TX_ESC clock for the D-PHY. (Note: Byte clock is limited to 125MHz (1GHz/8bits) and by writing TX_ESC_CLK_DIVISION=0x07 TX_ESC clock will always be lower than 20MHz)

- Configure the DPHY PLL clock frequency through the TEST Interface to operate at 1GHz, assuming that the REF_CLK is provided with a frequency of 27MHz
  - Write @ PHY_TST_CTRL0 - 32'h00000000 this disables the **testclr** pin enabling the interface to write new values to the DPHY internal registers.
  - Write @ PHY_TST_CTRL1 - 32'h00010044 this enables the **testen** pin bit 17 of this Core register and configures the **testdatain** to 8'h44. This operation initiate the configuration process of the test code number 0x44.
  - Write @ PHY_TEST_CTRL0 - 32'h0000002 followed by a new write to PHY_TEST_CTRL0 - 32'h00000000. This operation toggles the **testclk** (bit 2) and the **testdin** will be sampled on the falling edge of **testclk** latching a new test code.
  - Write @ PHY_TEST_CTRL1 - 32'h00000074 disabling the testen pin and configuring **testdatain** to 8'h74. This operation prepares the interface to load in test code 0x44 the 0x74 value.
  - Write @ PHY_TEST_CTRL0 - 32'h00000002 followed by a new write to PHY_TEST_CTRL0 - 32'h00000000. This operation toggles the **testclk** and the **testdin** will be sampled on the rising edge of **testclk** latching a new content data to the configured test code.
- Write @ PHY_RSTZ - 32'h00000007. This operation asserts **physhutdownz**, **phyrstz** and **enableclk** releasing the PHY from power down. The PHY will startup the PLL locking procedure to 1GHz operation.
- Read @ PHY_STATUS - 32'hxxxxxxx1, until bit 0 **phylock** is detected at 1 signaling that PLL is locked and that a stable byte clock is being provided to the DSI host controller.
- Read @ PHY_STATUS - 32'hxxxxx1x1, until bit 2 **phystopstateclklane** is read '1' identifying that Clock Lane is in Stop State. Clock lane need to be in Stop state so that the D-PHY can switch to other operational states such as the High Speed mode.
- Write register PHY_IF_CTRL bit 0 to generate High Speed clock (**txrequestHSclk**).
- Only after: 1) PLL locked and 2) Clock lane in Stop-State; the PHY will drive the correct LP sequence to configure the receiver end for HS.
- D-PHY starts transmitting HS clock on the Clock Lane.

41.6 MIPI_DSI Memory Map/Register Definition

MIPI_DSI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21E_0000	Version of the DSI host ctrl (MIPI_DSI_VERSION)	32	R	0000_0000h	41.6.1/3648

Table continues on the next page...

MIPI_DSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21E_0004	Core power up (MIPI_DSI_PWR_UP)	32	R/W	0000_0000h	41.6.2/3648
21E_0008	Number of active data lanes (MIPI_DSI_CLKMGR_CFG)	32	R/W	0000_0000h	41.6.3/3649
21E_000C	DPI interface configuration (MIPI_DSI_DPI_CFG)	32	R/W	0000_0000h	41.6.4/3649
21E_0010	DBI interface configuration (MIPI_DSI_DBI_CFG)	32	R/W	0000_0000h	41.6.5/3651
21E_0014	DBI command size configuration (MIPI_DSI_DBIS_CMDSIZE)	32	R/W	0000_0000h	41.6.6/3652
21E_0018	Packet handler configuration (MIPI_DSI_PCKHDL_CFG)	32	R/W	0000_0000h	41.6.7/3653
21E_001C	Video Mode Configuration (MIPI_DSI_VID_MODE_CFG)	32	R/W	0000_0000h	41.6.8/3654
21E_0020	Video packet configuration (MIPI_DSI_VID_PKT_CFG)	32	R/W	0000_0000h	41.6.9/3655
21E_0024	Command mode configuration (MIPI_DSI_CMD_MODE_CFG)	32	R/W	0000_0000h	41.6.10/ 3656
21E_0028	Line timer configuration (MIPI_DSI_TMR_LINE_CFG)	32	R/W	0000_0000h	41.6.11/ 3658
21E_002C	Vertical timing configuration (MIPI_DSI_VTIMING_CFG)	32	R/W	0000_0000h	41.6.12/ 3658
21E_0030	D-PHY timing configuration (MIPI_DSI_PHY_TMR_CFG)	32	R/W	0000_0000h	41.6.13/ 3659
21E_0034	Generic packet Header configuration (MIPI_DSI_GEN_HDR)	32	R/W	0000_0000h	41.6.14/ 3659
21E_0038	Generic payload data in/out (MIPI_DSI_GEN_PLD_DATA)	32	R/W	0000_0000h	41.6.15/ 3660
21E_003C	Command packet status (MIPI_DSI_CMD_PKT_STATUS)	32	R/W	0000_0000h	41.6.16/ 3660
21E_0040	Time Out timers configuration (MIPI_DSI_TO_CNT_CFG0)	32	R/W	0000_0000h	41.6.17/ 3662
21E_0044	Interrupt status register 0 (MIPI_DSI_ERROR_ST0)	32	R/W	0000_0000h	41.6.18/ 3662
21E_0048	Interrupt status register 1 (MIPI_DSI_ERROR_ST1)	32	R/W	0000_0000h	41.6.19/ 3664
21E_004C	Masks Interrupt generation trigged by ERROR_ST0 register (MIPI_DSI_ERROR_MSK0)	32	R/W	0000_0000h	41.6.20/ 3665
21E_0050	Masks Interrupt generation trigged by ERROR_ST1 register (MIPI_DSI_ERROR_MSK1)	32	R/W	0000_0000h	41.6.21/ 3667
21E_0054	D-PHY reset control (MIPI_DSI_PHY_RSTZ)	32	R/W	0000_0000h	41.6.22/ 3669
21E_0058	D-PHY interface configuration (MIPI_DSI_PHY_IF_CFG_)	32	R/W	0000_0000h	41.6.23/ 3670
21E_005C	D-PHY PPI interface control (MIPI_DSI_PHY_IF_CTRL)	32	R/W	0000_0000h	41.6.24/ 3670
21E_0060	D-PHY PPI status interface (MIPI_DSI_PHY_STATUS)	32	R/W	0000_0000h	41.6.25/ 3671
21E_0064	D-PHY Test interface control 0 (MIPI_DSI_PHY_TST_CTRL0)	32	R/W	0000_0001h	41.6.26/ 3673

Table continues on the next page...

MIPI_DSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21E_0068	D-PHY Test interface control 1 (MIPI_DSI_PHY_TST_CTRL1)	32	R/W	0000_0000h	41.6.27/ 3673

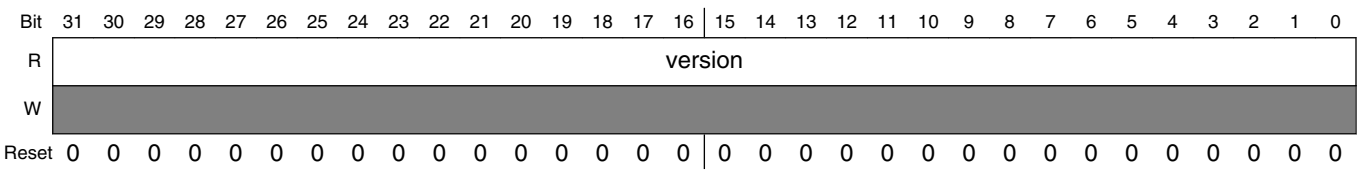
41.6.1 Version of the DSI host ctrl (MIPI_DSI_VERSION)

Size: 32 bits

Offset: 0x0

Memory Access: R

Address: 21E_0000h base + 0h offset = 21E_0000h



MIPI_DSI_VERSION field descriptions

Field	Description
version	Version of the DSI host controller

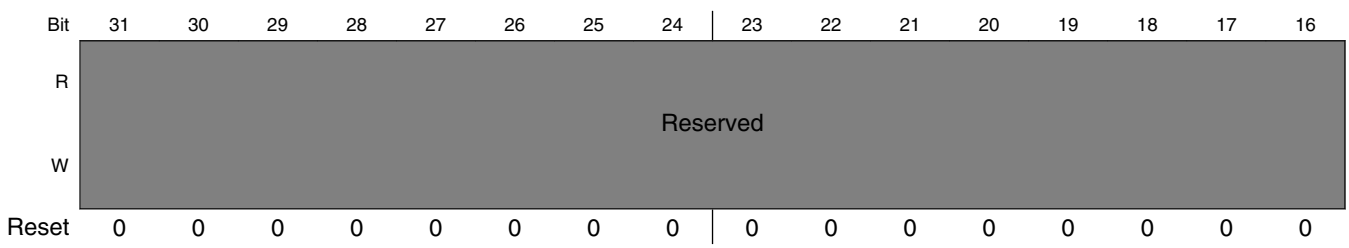
41.6.2 Core power up (MIPI_DSI_PWR_UP)

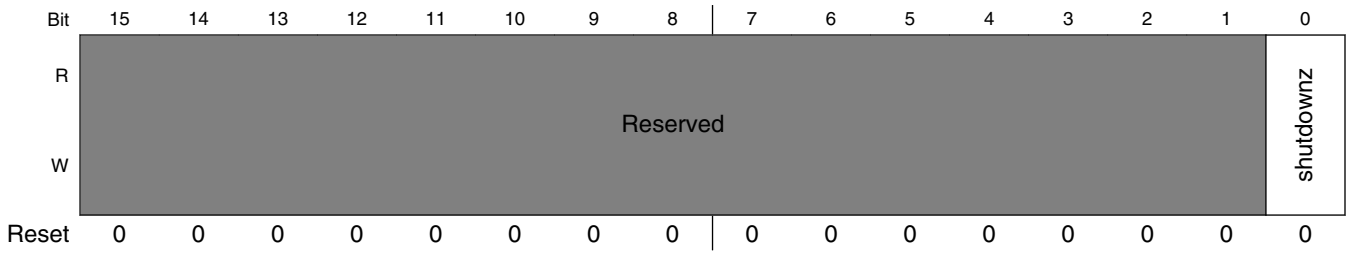
Size: 32 bits

Offset: 0x4

Memory Access: R/W

Address: 21E_0000h base + 4h offset = 21E_0004h





MIPI_DSI_PWR_UP field descriptions

Field	Description
31–1 -	This field is reserved. Reserved
0 shutdownz	Core power up 0 reset; 1 power up)

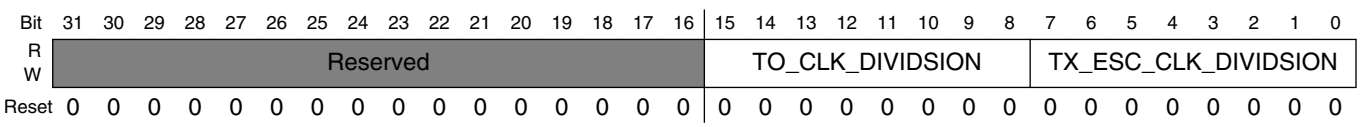
41.6.3 Number of active data lanes (MIPI_DSI_CLKMGR_CFG)

Size: 32 bits

Offset: 0x8

Memory Access: R/W

Address: 21E_0000h base + 8h offset = 21E_0008h



MIPI_DSI_CLKMGR_CFG field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
15–8 TO_CLK_ DIVIDSION	Division factor for Time Out clock used as timing unit in the configuration of HS to LP and LP to HS transition error.
TX_ESC_CLK_ DIVIDSION	Division factor for TX ESCAPE clock source (lanebyteclk pin), values 0 and 1 stop TX_ESC clock generation.

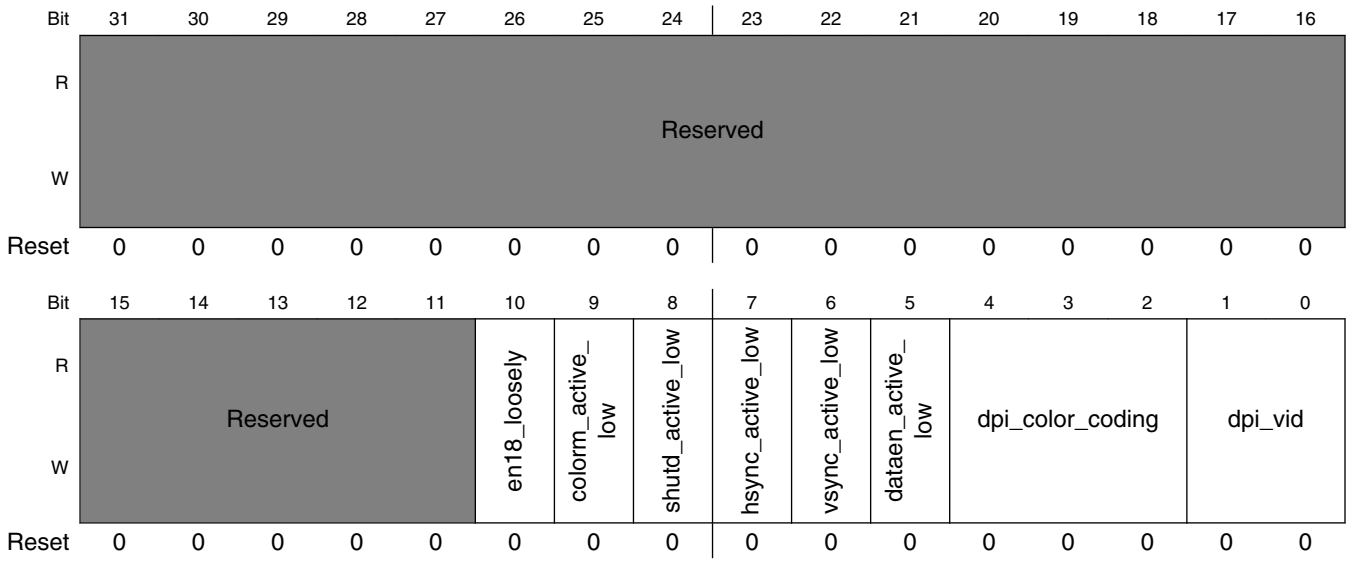
41.6.4 DPI interface configuration (MIPI_DSI_DPI_CFG)

Size: 32 bits

Offset: 0xc

Memory Access: R/W

Address: 21E_0000h base + Ch offset = 21E_000Ch



MIPI_DSI_DPI_CFG field descriptions

Field	Description
31–11 -	This field is reserved. Reserved
10 en18_loosely	Enable 18 loosely packet pixel stream.
9 colorm_active_low	Set to configure Color Mode pin (dpicolcorm) as Active low
8 shutd_active_low	Set to configure Shut Down pin (dpishutdn) as Active low
7 hsync_active_low	Set to configure Horizontal Synchronism pin (dpihsync) as Active low
6 vsync_active_low	Set to configure Vertical Synchronism pin (dpivsync) as Active low
5 dataen_active_low	Set to configure Data enable pin (dpidaten) as Active low
4–2 dpi_color_coding	DPI color coding. 0 16bit config1 1 16bit config2; 2 16bit config3; 3 18bit config1; 4 18bit config2; 5 to 7 24 bit.
dpi_vid	Configures the DPI Virtual Channel ID that will be indexed to the Video mode packets.

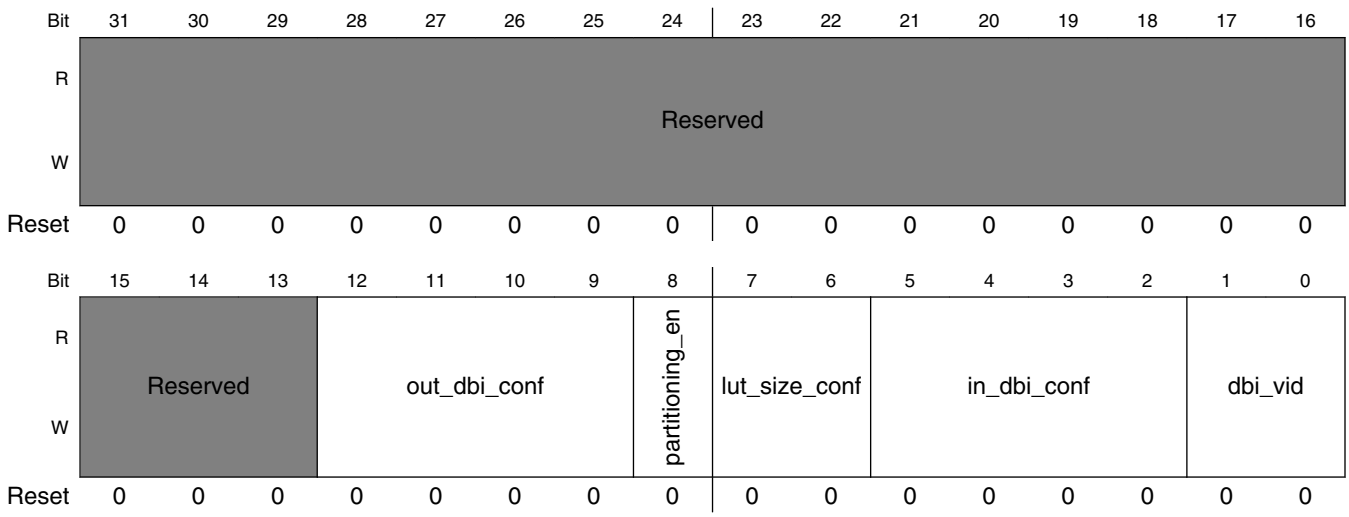
41.6.5 DBI interface configuration (MIPI_DSI_DBI_CFG)

Size: 32 bits

Offset: 0x10

Memory Access: R/W

Address: 21E_0000h base + 10h offset = 21E_0010h



MIPI_DSI_DBI_CFG field descriptions

Field	Description
31–13 -	This field is reserved. Reserved
12–9 out_dbi_conf	Configures the DBI output pixel data configuration; 0 8bit 8bpp; 1 8bit 12bpp; 2 8bit 16bpp; 3 8bit 18bpp; 4 8bit 24bpp; 5 9bit 18bpp; 6 16bit 8bpp; 7 16bit 12bpp; 8 16bit 16bpp; 9 16bit 18bpp, option1; 10 16bit 18bpp, option2; 11 16bit 24bpp, option1; 12 16bit 24bpp, option2
8 partitioning_en	Enables write memory continue through input command (system needs to ensure correct partitioning of Long Write commands)

Table continues on the next page...

MIPI_DSI_DBI_CFG field descriptions (continued)

Field	Description
7–6 lut_size_conf	Configures the size used to transport Write Lut commands; 0 16-bit color display; 1 18-bit color display; 2 24-bit color display; 3 16-bit color display
5–2 in_dbi_conf	Configures DBI input pixel data configuration; 0 8bit 8bpp; 1 8bit 12bpp; 2 8bit 16bpp; 3 8bit 18bpp; 4 8bit 24bpp; 5 9bit 18bpp; 6 16bit 8bpp; 7 16bit 12bpp; 8 16bit 16bpp; 9 16bit 18bpp, option1; 10 16bit 18bpp, option2; 11 16bit 24bpp, option1; 12 16bit 24bpp, option2
dbi_vid	Configures the DBI Virtual Channel ID that will be indexed to the DCS packets.

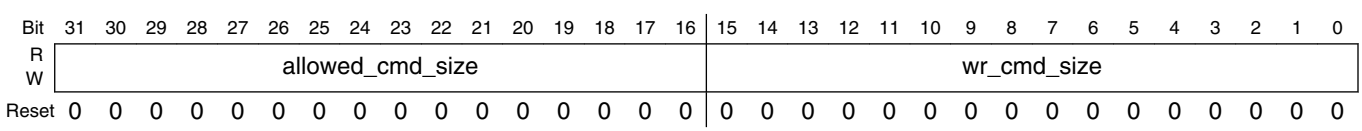
41.6.6 DBI command size configuration (MIPI_DSI_DBIS_CMDSIZE)

Size: 32 bits

Offset: 0x14

Memory Access: R/W

Address: 21E_0000h base + 14h offset = 21E_0014h



MIPI_DSI_DBIS_CMDSIZE field descriptions

Field	Description
31–16 allowed_cmd_size	Configures the maximum allowed size of a DCS write memory command. This register is used to partition a write memory command into several write memory continues. It is only used if bit 'partitioning_en' is disabled. Size of DSI packet payload is the actual payload size minus 1 since the DCS command is in the DSI packet payload.

Table continues on the next page...

MIPI_DSI_DBIS_CMDSIZE field descriptions (continued)

Field	Description
wr_cmd_size	Configures the size of the DCS write memory commands. Size of DSI packet payload is the actual payload size minus 1 since the DCS command is in the DSI packet payload.

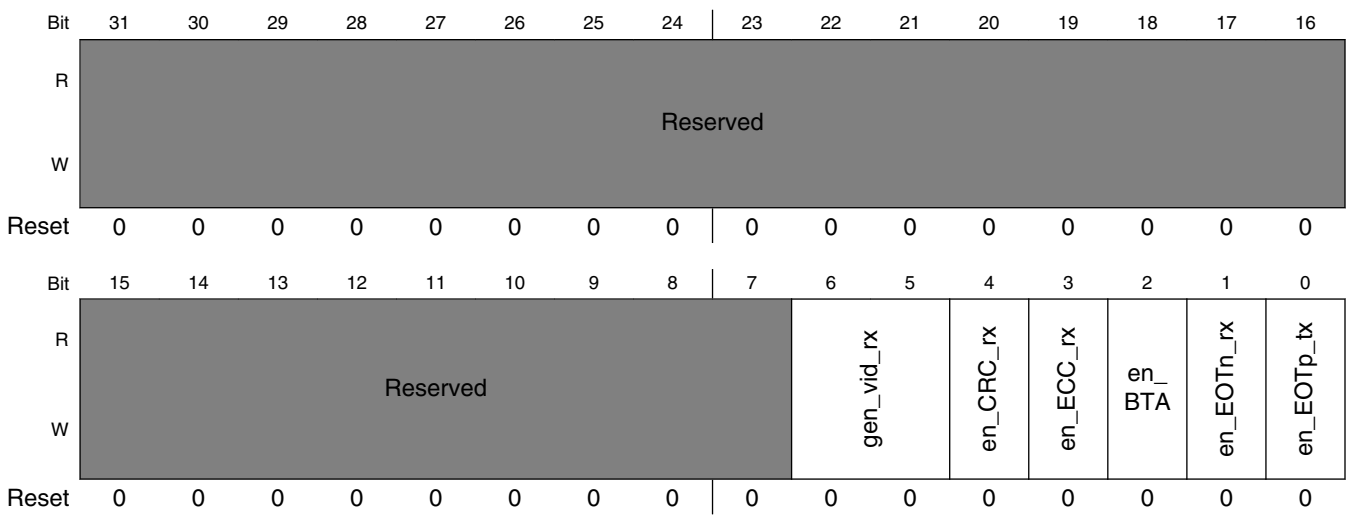
41.6.7 Packet handler configuration (MIPI_DSI_PCKHDL_CFG)

Size: 32 bits

Offset: 0x18

Memory Access: R/W

Address: 21E_0000h base + 18h offset = 21E_0018h



MIPI_DSI_PCKHDL_CFG field descriptions

Field	Description
31–7 -	This field is reserved. Reserved
6–5 gen_vid_rx	Generic interface read-back Virtual Channel identification
4 en_CRC_rx	Enables CRC reception and error reporting
3 en_ECC_rx	Enables ECC reception, error correction and reporting
2 en_BTA	Enables Bus Turn-Around request
1 en_EOTn_rx	Enables EOTp reception
0 en_EOTp_tx	Enables EOTp transmission

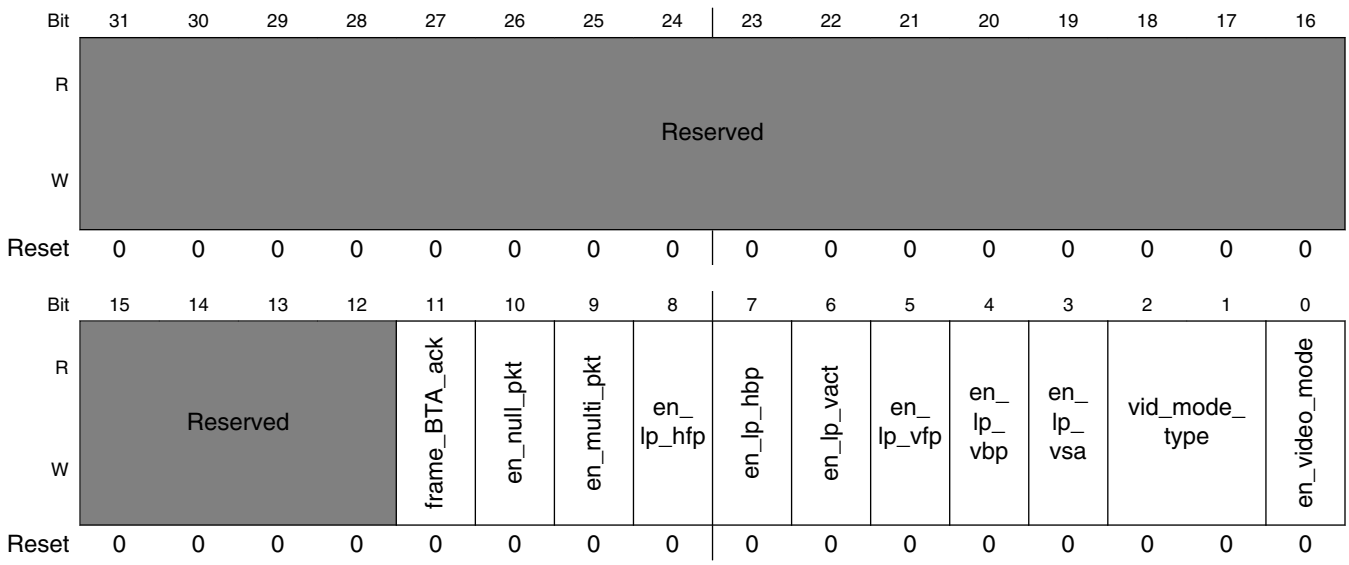
41.6.8 Video Mode Configuration (MIPI_DSI_VID_MODE_CFG)

Size: 32 bits

Offset: 0x1c

Memory Access: R/W

Address: 21E_0000h base + 1Ch offset = 21E_001Ch



MIPI_DSI_VID_MODE_CFG field descriptions

Field	Description
31–12 -	This field is reserved. Reserved
11 frame_BTA_ack	Enables the request for an acknowledge response at the end of a frame
10 en_null_pkt	Enables the tranmission of null packets in the HACT period
9 en_multi_pkt	Enables the tranmission of multi video packets in the HACT period
8 en_lp_hfp	Enables return to Low Power inside HFP period when timing allows
7 en_lp_hbp	Enables return to Low Power inside HBP period when timing allows
6 en_lp_vact	Enables return to Low Power inside VACT period when timing allows
5 en_lp_vfp	Enables return to Low Power inside VFP period when timing allows
4 en_lp_vbp	Enables return to Low Power inside VBP period when timing allows

Table continues on the next page...

MIPI_DSI_VID_MODE_CFG field descriptions (continued)

Field	Description
3 en_lp_vsa	Enables return to Low Power inside VSA period when timing allows
2–1 vid_mode_type	Selects video mode transmission type. 0: Non-burst with Sync pulses; 1: Non-burst with Sync events; 2-3: Burst with Sync pulses.
0 en_video_mode	Enables DPI Video mode transmission

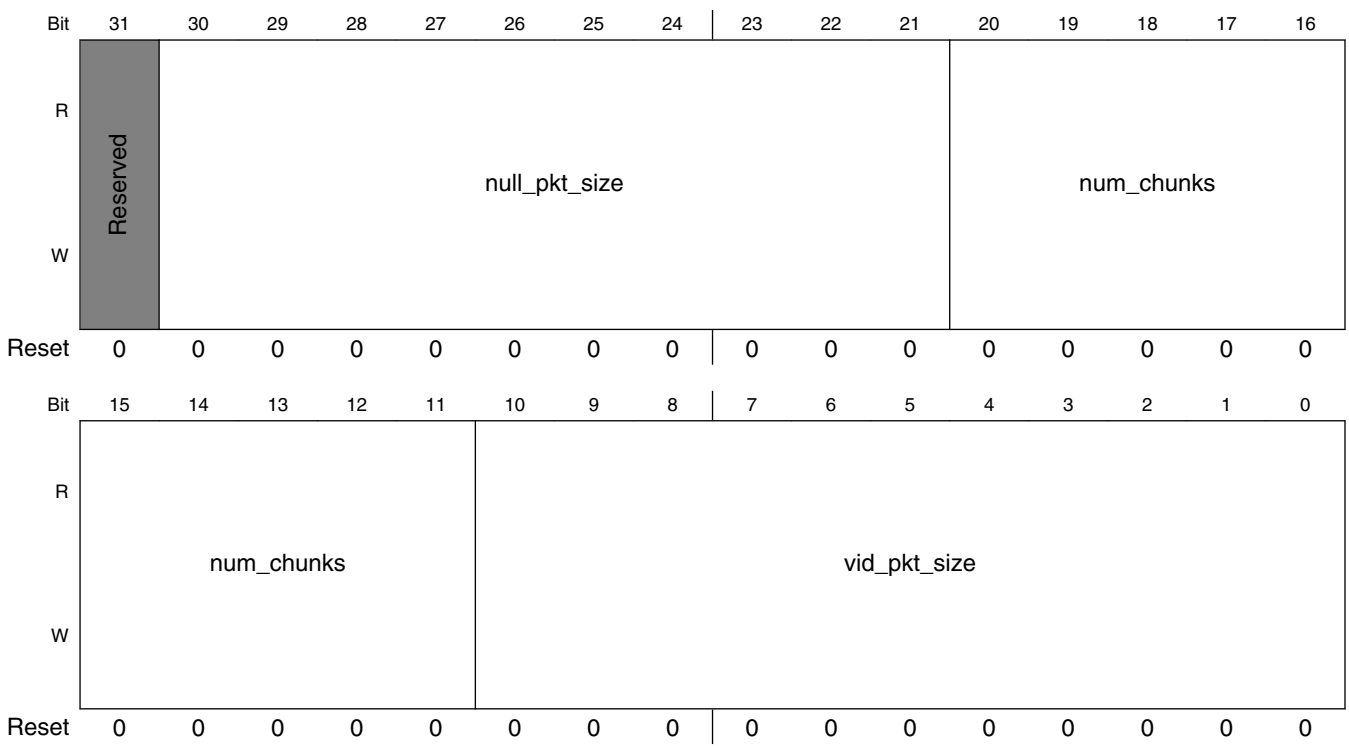
41.6.9 Video packet configuration (MIPI_DSI_VID_PKT_CFG)

Size: 32 bits

Offset: 0x20

Memory Access: R/W

Address: 21E_0000h base + 20h offset = 21E_0020h



MIPI_DSI_VID_PKT_CFG field descriptions

Field	Description
31 -	This field is reserved. Reserved

Table continues on the next page...

MIPI_DSI_VID_PKT_CFG field descriptions (continued)

Field	Description
30–21 null_pkt_size	Configures the number of bytes in a null packet
20–11 num_chunks	Configures the number of chunks to be transmitted during a Line period. (A chunk is a video packet or a null packet)
vid_pkt_size	Configures the number of pixel on a single video packet. (If using 18 bit mode and not enabling loosely packet stream this value must be a multiple of 4)

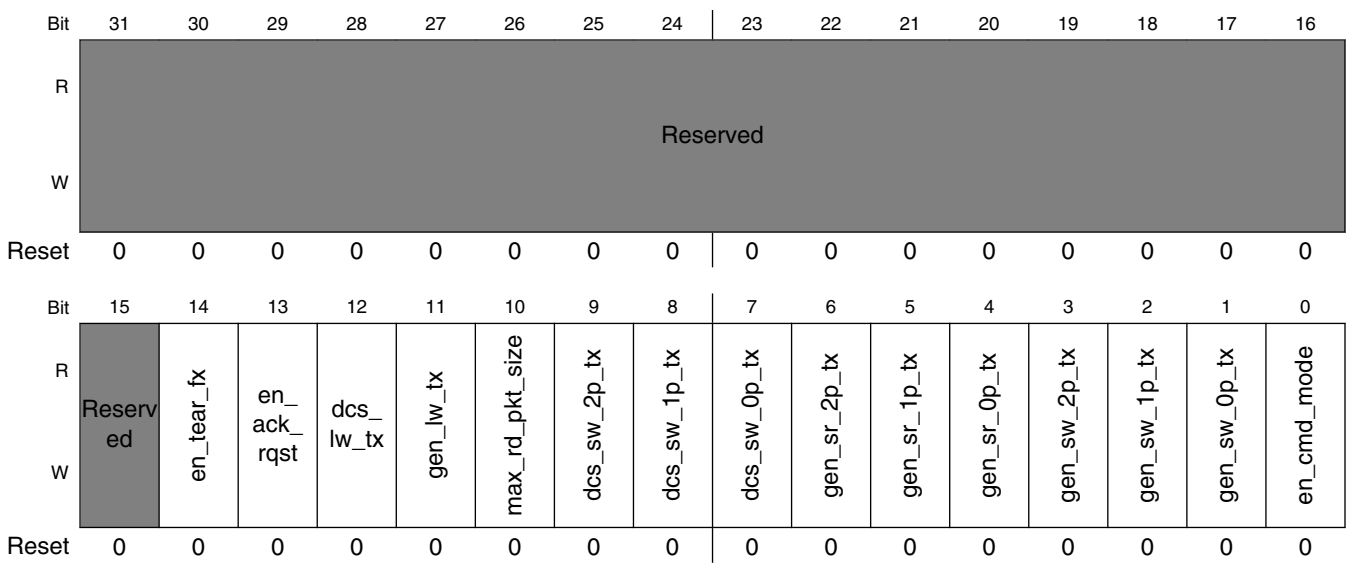
41.6.10 Command mode configuration (MIPI_DSI_CMD_MODE_CFG)

Size: 32 bits

Offset: 0x24

Memory Access: R/W

Address: 21E_0000h base + 24h offset = 21E_0024h



MIPI_DSI_CMD_MODE_CFG field descriptions

Field	Description
31–15 -	This field is reserved. Reserved
14 en_tear_fx	Enables the tearing effect acknowledge request
13 en_ack_rqst	Enables the acknowledge request after each packet transmission

Table continues on the next page...

MIPI_DSI_CMD_MODE_CFG field descriptions (continued)

Field	Description
12 dcs_lw_tx	Configures the DCS Long Write Packet command transmission type. 0 High Speed; 1 Low Power
11 gen_lw_tx	Configures the Generic Long Write Packet command transmission type. 0 High Speed; 1 Low Power
10 max_rd_pkt_size	Configures the Maximum Read Packet Size command transmission type. 0 High Speed; 1 Low Power
9 dcs_sw_2p_tx	Configures the DCS Short Write Packet with 2 Parameters command transmission type. 0 High Speed; 1 Low Power
8 dcs_sw_1p_tx	Configures the DCS Short Write Packet with 1 Parameters command transmission type. 0 High Speed; 1 Low Power
7 dcs_sw_0p_tx	Configures the DCS Short Write Packet with 0 Parameters command transmission type. 0 High Speed; 1 Low Power
6 gen_sr_2p_tx	Configures the Generic Short Read Packet with 2 Parameters command transmission type. 0 High Speed; 1 Low Power
5 gen_sr_1p_tx	Configures the Generic Short Read Packet with 1 Parameters command transmission type. 0 High Speed; 1 Low Power
4 gen_sr_0p_tx	Configures the Generic Short Read Packet with 0 Parameters command transmission type. 0 High Speed; 1 Low Power
3 gen_sw_2p_tx	Configures the Generic Short Write Packet with 2 Parameters command transmission type. 0 High Speed; 1 Low Power
2 gen_sw_1p_tx	Configures the Generic Short Write Packet with 1 Parameters command transmission type. 0 High Speed; 1 Low Power
1 gen_sw_0p_tx	Configures the Generic Short Write Packet with 0 Parameters command transmission type. 0 High Speed; 1 Low Power

Table continues on the next page...

MIPI_DSI_CMD_MODE_CFG field descriptions (continued)

Field	Description
0 en_cmd_mode	Enables the Command Mode Protocol for transmissions.

41.6.11 Line timer configuration (MIPI_DSI_TMR_LINE_CFG)

Size: 32 bits

Offset: 0x28

Memory Access: R/W

Address: 21E_0000h base + 28h offset = 21E_0028h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	hline_time																hbp_time								hsa_time							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_TMR_LINE_CFG field descriptions

Field	Description
31–18 hline_time	Configures the size of the total line counted in lane byte cycles
17–9 hbp_time	Configures the Horizontal Back Porch period in lane byte clock cycles
hsa_time	Configures the Horizontal Synchronism Active period in lane byte clock cycles

41.6.12 Vertical timing configuration (MIPI_DSI_VTIMING_CFG)

Size: 32 bits

Offset: 0x2c

Memory Access: R/W

Address: 21E_0000h base + 2Ch offset = 21E_002Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved					v_active_lines											vfp_lines						vbp_lines						vsa_lines			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_VTIMING_CFG field descriptions

Field	Description
31–27 -	This field is reserved. Reserved
26–16 v_active_lines	Configures the Vertical Active period measured in horizontal lines
15–10 vfp_lines	Configures the Vertical Front Porch period measured in horizontal lines
9–4 vbp_lines	Configures the Vertical Back Porch period measured in horizontal lines
vsa_lines	Configures the Vertical Synchronism Active period measured in horizontal lines

41.6.13 D-PHY timing configuration (MIPI_DSI_PHY_TMR_CFG)

Size: 32 bits

Offset: 0x30

Memory Access: R/W

Address: 21E_0000h base + 30h offset = 21E_0030h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				phy_hs2lp_time								phy_lp2hs_time				bta_time															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_PHY_TMR_CFG field descriptions

Field	Description
31–28 -	This field is reserved. Reserved
27–20 phy_hs2lp_time	Configures the maximum time that the PHY takes to go from High Speed to Low Power transmission measured in lane byte clock cycles
19–12 phy_lp2hs_time	Configures the maximum time that the PHY takes to go from Low Power to High Speed transmission measured in lane byte clock cycles
bta_time	Configures the maximum time required to perform the Bus Turn Around operation measured in lane byte clock cycles

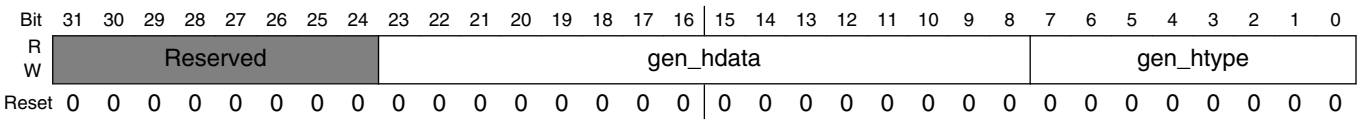
41.6.14 Generic packet Header configuration (MIPI_DSI_GEN_HDR)

Size: 32 bits

Offset: 0x34

Memory Access: R/W

Address: 21E_0000h base + 34h offset = 21E_0034h



MIPI_DSI_GEN_HDR field descriptions

Field	Description
31–24 -	This field is reserved. Reserved
23–8 gen_hdata	Configures the packet data to be transmitted through the generic interface
gen_htype	Configures the packet type to be transmitted through the generic interface. Writing to this register triggers packet transmission (Payload must be written in advance)

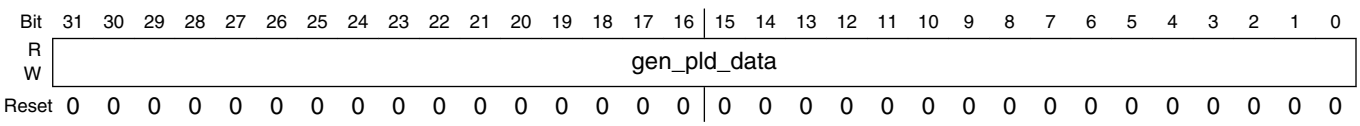
41.6.15 Generic payload data in/out (MIPI_DSI_GEN_PLD_DATA)

Size: 32 bits

Offset: 0x38

Memory Access: R/W

Address: 21E_0000h base + 38h offset = 21E_0038h



MIPI_DSI_GEN_PLD_DATA field descriptions

Field	Description
gen_pld_data	This register contains the input/output generic packet data. Write access to it writes the content of the packet payload. Read access reads the incoming generic read data

41.6.16 Command packet status (MIPI_DSI_CMD_PKT_STATUS)

Size: 32 bits

Offset: 0x3c

Memory Access: R/W

Address: 21E_0000h base + 3Ch offset = 21E_003Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	reserved2	dbi_rd_cmd_busy	dbi_pld_r_full	dbi_pld_r_empty	dbi_pld_w_full	dbi_pld_w_empty	dbi_cmd_full	dbi_cmd_empty	reserved1	gen_rd_cmd_busy	gen_pld_r_full	gen_pld_r_empty	gen_pld_w_full	gen_pld_w_empty	gen_cmd_full	gen_cmd_empty
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_CMD_PKT_STATUS field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
15 reserved2	This field is reserved. Reserved
14 dbi_rd_cmd_busy	Set when a read command is issued and cleared once the entire response is stored in the FIFO
13 dbi_pld_r_full	Reports the full status of the DBI read payload FIFO
12 dbi_pld_r_empty	Reports the empty status of the DBI read payload FIFO
11 dbi_pld_w_full	Reports the full status of the DBI write payload FIFO
10 dbi_pld_w_empty	Reports the empty status of the DBI write payload FIFO
9 dbi_cmd_full	Reports the full status of the DBI command FIFO
8 dbi_cmd_empty	Reports the empty status of the DBI command FIFO
7 reserved1	This field is reserved. Reserved
6 gen_rd_cmd_busy	Set when a read command is issued and cleared once the entire response is stored in the FIFO
5 gen_pld_r_full	Reports the full status of the generic read payload FIFO
4 gen_pld_r_empty	Reports the empty status of the generic read payload FIFO

Table continues on the next page...

MIPI_DSI_CMD_PKT_STATUS field descriptions (continued)

Field	Description
3 gen_pld_w_full	Reports the full status of the generic write payload FIFO
2 gen_pld_w_ empty	Reports the empty status of the generic write payload FIFO
1 gen_cmd_full	Reports the full status of the generic command FIFO
0 gen_cmd_empty	Reports the empty status of the generic command FIFO

41.6.17 Time Out timers configuration (MIPI_DSI_TO_CNT_CFG0)

Size: 32 bits

Offset: 0x40

Memory Access: R/W

Address: 21E_0000h base + 40h offset = 21E_0040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	lprx_to_cnt																hstx_to_cnt															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_TO_CNT_CFG0 field descriptions

Field	Description
31–16 lprx_to_cnt	Configures the time out counter that will trigger a Low Power Reception Time Out Contention Detection. (Measured in TO_CLK_DIVISION cycles)
hstx_to_cnt	Configures the time out counter that will trigger a High Speed Transmission Time Out Contention Detection (Measured in TO_CLK_DIVISION cycles)

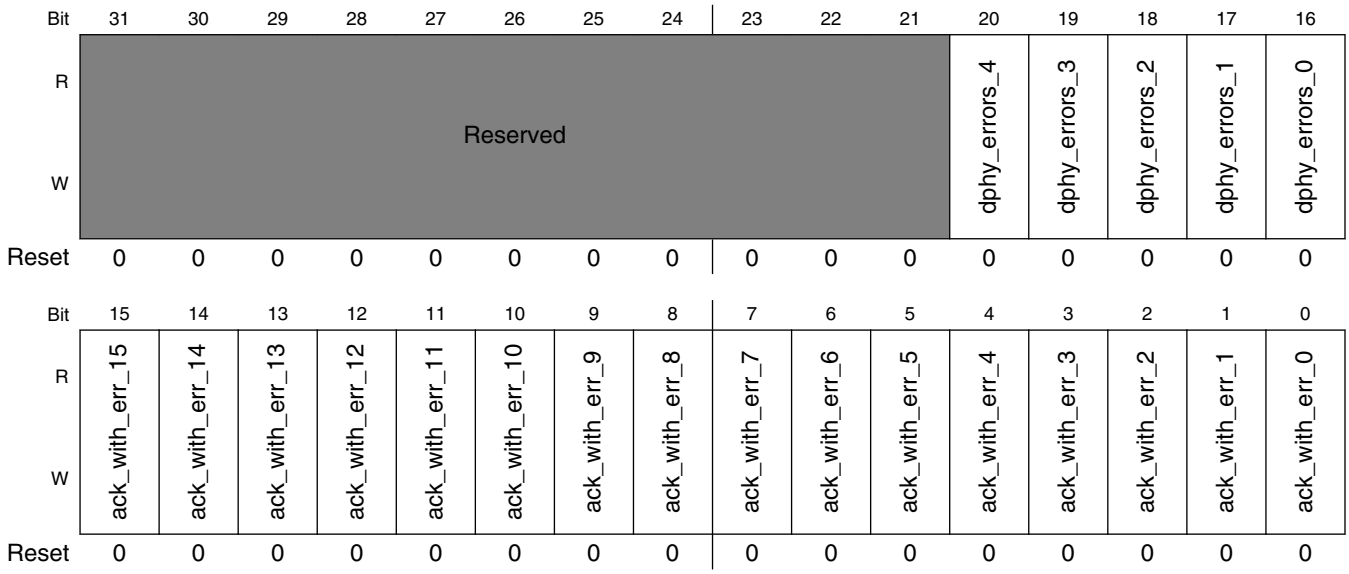
41.6.18 Interrupt status register 0 (MIPI_DSI_ERROR_ST0)

Size: 32 bits

Offset: 0x44

Memory Access: R/W

Address: 21E_0000h base + 44h offset = 21E_0044h



MIPI_DSI_ERROR_ST0 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved
20 dphy_errors_4	ErrContentionLP1 LP1 Contention Error from Lane 0
19 dphy_errors_3	ErrContentionLP0 LP0 Contention Error from Lane 0
18 dphy_errors_2	ErrControl Control Error from Lane 0
17 dphy_errors_1	ErrSyncEsc Low-Power Data Transmission Synchronization Error from Lane 0
16 dphy_errors_0	ErrEsc Escape Entry Error from Lane 0
15 ack_with_err_15	Retrieves DSI Protocol Violation from Display Acknowledge Error Report
14 ack_with_err_14	Retrieves Reserved (specific to device) from Display Acknowledge Error Report
13 ack_with_err_13	Retrieves Invalid Transmission Length from Display Acknowledge Error Report
12 ack_with_err_12	Retrieves DSI VC ID Invalid from Display Acknowledge Error Report
11 ack_with_err_11	Retrieves DSI Data Type Not Recognized from Display Acknowledge Error Report
10 ack_with_err_10	Retrieves Checksum Error (Long packet only) from Display Acknowledge Error Report
9 ack_with_err_9	Retrieves ECC Error, multi-bit (detected, not corrected) from Display Acknowledge Error Report
8 ack_with_err_8	Retrieves ECC Error, single-bit (detected and corrected) from Display Acknowledge Error Report

Table continues on the next page...

MIPI_DSI_ERROR_ST0 field descriptions (continued)

Field	Description
7 ack_with_err_7	Retrieves Reserved (specific to device) from Display Acknowledge Error Report
6 ack_with_err_6	Retrieves False Control Error from Display Acknowledge Error Report
5 ack_with_err_5	Retrieves HS Receive Timeout Error from Display Acknowledge Error Report
4 ack_with_err_4	Retrieves Low-Power Transmit Sync Error from Display Acknowledge Error Report
3 ack_with_err_3	Retrieves Escape Mode Entry Command Error from Display Acknowledge Error Report
2 ack_with_err_2	Retrieves EoT Sync Error from Display Acknowledge Error Report
1 ack_with_err_1	Retrieves SoT Sync Error from Display Acknowledge Error Report
0 ack_with_err_0	Retrieves SoT Error from Display Acknowledge Error Report

41.6.19 Interrupt status register 1 (MIPI_DSI_ERROR_ST1)

Size: 32 bits

Offset: 0x48

Memory Access: R/W

Address: 21E_0000h base + 48h offset = 21E_0048h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved														dbi_illegal_comm_err	dbi_pld_recv_err
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	dbi_pld_rd_err	dbi_pld_wr_err	dbi_cmd_wr_err	gen_pld_recv_err	gen_pld_rd_err	gen_pld_send_err	gen_pld_wr_err	gen_cmd_wr_err	dpi_pld_wr_err	eopt_err	pkt_size_err	crc_err	ecc_multi_err	ecc_sinlge_err	to_lp_rx	to_hs_tx
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_ERROR_ST1 field descriptions

Field	Description
31–18 -	This field is reserved. Reserved
17 dbi_illegal_comm_err	Attempt to write an illegal command on the DPI interface and core blocked by transmission
16 dbi_pld_recv_err	During a DBI read back packet, the payload FIFO went full and received data was corrupted
15 dbi_pld_rd_err	During a DCS read data, the payload FIFO went empty and data was send to the interface corrupted
14 dbi_pld_wr_err	System tried to write payload data through the DBI interface and the FIFO was full, therefore the Command was not written
13 dbi_cmd_wr_err	System tried to write a command through the DBI but the command FIFO was full, therefore the command was not written
12 gen_pld_recv_err	During a generic interface packet read back, the payload FIFO went full and received data was corrupted
11 gen_pld_rd_err	During a DCS read data, the payload FIFO went empty and data was send to the interface corrupted
10 gen_pld_send_err	During a generic interface packet build, the payload FIFO went empty and data was sent corrupted
9 gen_pld_wr_err	System tried to write a payload data through the generic interface and FIFO was full, therefore the payload was not written
8 gen_cmd_wr_err	System tried to write a command through the generic interface and FIFO was full, therefore the command was not written
7 dpi_pld_wr_err	During a DPI pixel line storage the payload FIFO went full and data stored is corrupted
6 eopt_err	EOTp Packet not received at the end of the incoming peripheral transmission
5 pkt_size_err	Packet size error was detected during packet reception
4 crc_err	CRC error was detected in the received packet payload
3 ecc_multi_err	ECC multiple error was detected in a received packet
2 ecc_sinlge_err	ECC single error was detected and corrected in a received packet
1 to_lp_rx	Low Power Reception Time Out Counter reached the end and Contention Detection as been detected
0 to_hs_tx	High Speed Transmission Time Out Counter reached the end and Contention Detection as been detected

41.6.20 Masks Interrupt generation trigged by ERROR_ST0 register (MIPI_DSI_ERROR_MSK0)

Size: 32 bits

Offset: 0x4c

Memory Access: R/W

Address: 21E_0000h base + 4Ch offset = 21E_004Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved											dphy_errors_4	dphy_errors_3	dphy_errors_2	dphy_errors_1	dphy_errors_0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	ack_with_err_15	ack_with_err_14	ack_with_err_13	ack_with_err_12	ack_with_err_11	ack_with_err_10	ack_with_err_9	ack_with_err_8	ack_with_err_7	ack_with_err_6	ack_with_err_5	ack_with_err_4	ack_with_err_3	ack_with_err_2	ack_with_err_1	ack_with_err_0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_ERROR_MSK0 field descriptions

Field	Description
31–21 -	This field is reserved. Reserved
20 dphy_errors_4	ErrContentionLP1 LP1 Contention Error from Lane 0
19 dphy_errors_3	ErrContentionLP0 LP0 Contention Error from Lane 0
18 dphy_errors_2	ErrControl Control Error from Lane 0
17 dphy_errors_1	ErrSyncEsc Low-Power Data Transmission Synchronization Error from Lane 0
16 dphy_errors_0	ErrEsc Escape Entry Error from Lane 0
15 ack_with_err_15	Masks DSI Protocol Violation from Display Acknowledge Error Report
14 ack_with_err_14	Masks Reserved (specific to device) from Display Acknowledge Error Report
13 ack_with_err_13	Masks Invalid Transmission Length from Display Acknowledge Error Report
12 ack_with_err_12	Masks DSI VC ID Invalid from Display Acknowledge Error Report
11 ack_with_err_11	Masks DSI Data Type Not Recognized from Display Acknowledge Error Report
10 ack_with_err_10	Masks Checksum Error (Long packet only) from Display Acknowledge Error Report

Table continues on the next page...

MIPI_DSI_ERROR_MSK0 field descriptions (continued)

Field	Description
9 ack_with_err_9	Masks ECC Error, multi-bit (detected, not corrected) from Display Acknowledge Error Report
8 ack_with_err_8	Masks ECC Error, single-bit (detected and corrected) from Display Acknowledge Error Report
7 ack_with_err_7	Masks Reserved (specific to device) from Display Acknowledge Error Report
6 ack_with_err_6	Masks False Control Error from Display Acknowledge Error Report
5 ack_with_err_5	Masks HS Receive Timeout Error from Display Acknowledge Error Report
4 ack_with_err_4	Masks Low-Power Transmit Sync Error from Display Acknowledge Error Report
3 ack_with_err_3	Masks Escape Mode Entry Command Error from Display Acknowledge Error Report
2 ack_with_err_2	Masks EoT Sync Error from Display Acknowledge Error Report
1 ack_with_err_1	Masks SoT Sync Error from Display Acknowledge Error Report
0 ack_with_err_0	Masks SoT Error from Display Acknowledge Error Report

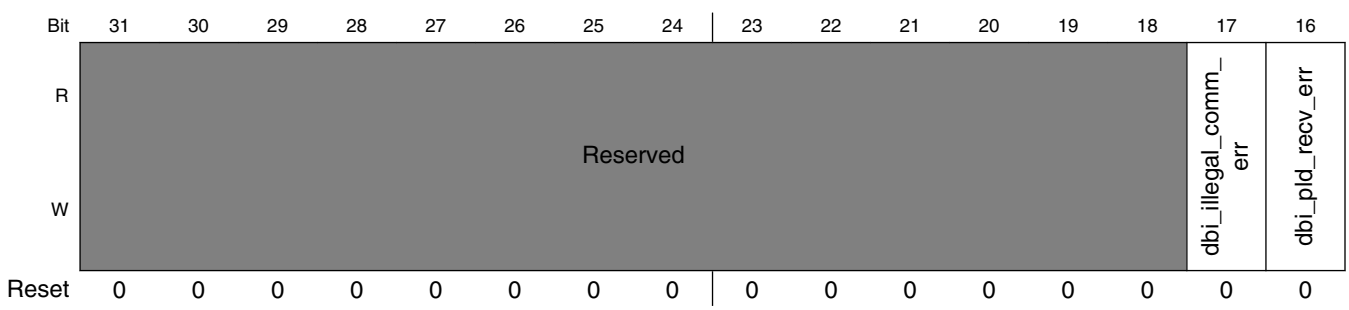
41.6.21 Masks Interrupt generation trigged by ERROR_ST1 register (MIPI_DSI_ERROR_MSK1)

Size: 32 bits

Offset: 0x50

Memory Access: R/W

Address: 21E_0000h base + 50h offset = 21E_0050h



Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	dbi_pld_rd_err	dbi_pld_wr_err	dbi_cmd_wr_err	gen_pld_recv_err	gen_pld_rd_err	gen_pld_send_err	gen_pld_wr_err	gen_cmd_wr_err	dpi_pld_wr_err	eopt_err	pkt_size_err	crc_err	ecc_multi_err	ecc_sinlge_err	to_lp_rx	to_ hs_tx
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_ERROR_MSK1 field descriptions

Field	Description
31–18 -	This field is reserved. Reserved
17 dbi_illegal_ comm_err	Masks error attempt to write an illegal command on DPI
16 dbi_pld_recv_err	Masks DBI read back packet payload FIFO full error
15 dbi_pld_rd_err	Masks payload DBI FIFO empty error
14 dbi_pld_wr_err	Masks write payload data DBI FIFO full error
13 dbi_cmd_wr_err	Masks DBI command FIFO full error
12 gen_pld_recv_err	Masks generic interface packet read back FIFO full error
11 gen_pld_rd_err	Masks DCS read data payload FIFO empty error
10 gen_pld_send_ err	Masks generic interface packet build FIFO empty error
9 gen_pld_wr_err	Masks payload data FIFO of generic interface full error
8 gen_cmd_wr_err	Masks command FIFO of generic interface full error
7 dpi_pld_wr_err	Masks DPI pixel line payload FIFO full error
6 eopt_err	Masks EOTp Packet not received error
5 pkt_size_err	Masks Packet size error
4 crc_err	Masks CRC error
3 ecc_multi_err	Masks ECC multiple error
2 ecc_sinlge_err	Masks ECC single error

Table continues on the next page...

MIPI_DSI_ERROR_MSK1 field descriptions (continued)

Field	Description
1 to_lp_rx	Masks Low Power Reception Time Out Counter error
0 to_hs_tx	Masks High Speed Transmission Time Out Counter error

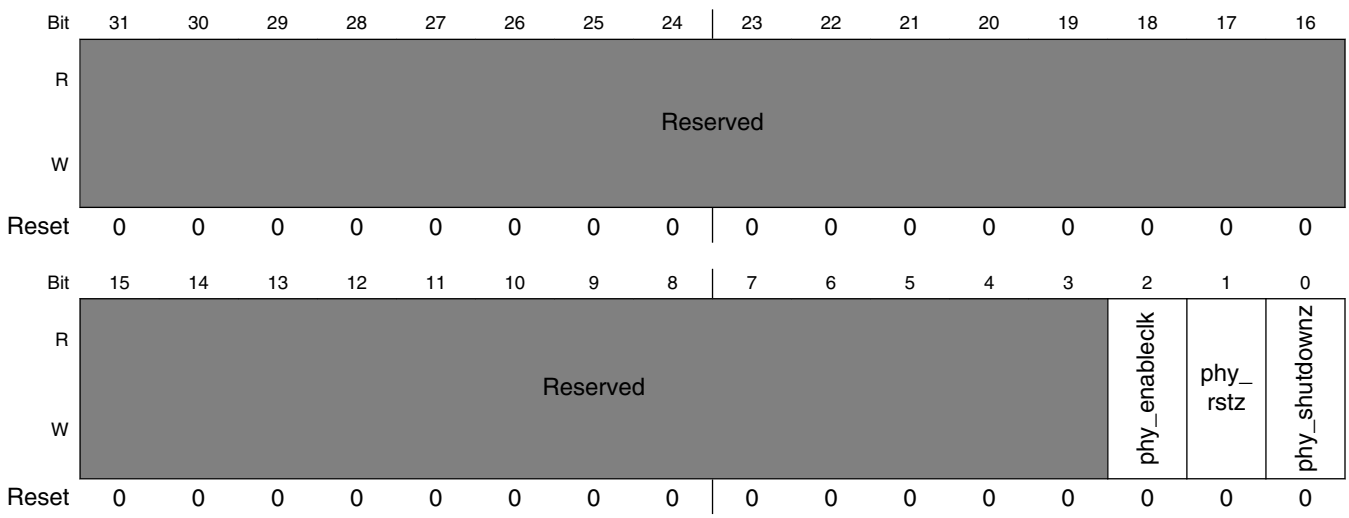
41.6.22 D-PHY reset control (MIPI_DSI_PHY_RSTZ)

Size: 32 bits

Offset: 0x54

Memory Access: R/W

Address: 21E_0000h base + 54h offset = 21E_0054h



MIPI_DSI_PHY_RSTZ field descriptions

Field	Description
31–3 -	This field is reserved. Reserved
2 phy_enableclk	Enables D-PHY Clock Lane Module when 1
1 phy_rstz	D-PHY Reset disable when 1, used to place the digital section of D-PHY in reset state
0 phy_shutdownz	D-PHY Shutdown disable when 1, used to place the complete D-PHY macro in power down

41.6.23 D-PHY interface configuration (MIPI_DSI_PHY_IF_CFG_)

Size: 32 bits

Offset: 0x58

Memory Access: R/W

Address: 21E_0000h base + 58h offset = 21E_0058h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved						phy_stop_wait_time								n_lanes	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_PHY_IF_CFG_ field descriptions

Field	Description
31–10 -	This field is reserved. Reserved
9–2 phy_stop_wait_time	Configures minimum wait period to request an HS transmission after the stop state accounted in clock lane cycles
n_lanes	Number of active data lanes. 00 1 Data Lane (Lane 0) 01 2 Data Lanes (Lane 0, and Lane 1) 10 Reserved 11 Reserved

41.6.24 D-PHY PPI interface control (MIPI_DSI_PHY_IF_CTRL)

Size: 32 bits

Offset: 0x5c

Memory Access: R/W

Address: 21E_0000h base + 5Ch offset = 21E_005Ch

MIPI_DSI_PHY_IF_CTRL field descriptions

Field	Description
31–9 -	This field is reserved. Reserved
8–5 phy_tx_triggers	Controls the trigger transmissions
4 phy_txexitulpslan	ULPS mode Exit on on all active data lanes
3 phy_txrequlpslan	ULPS mode Request on all active data lanes
2 phy_txexitulpsclk	ULPS mode Exit on Clock Lane
1 phy_txrequlpsclk	ULPS mode Request on Clock Lane
0 phy_ txrequestclkhs	Controls D-PHY PPI txrequestclkHS

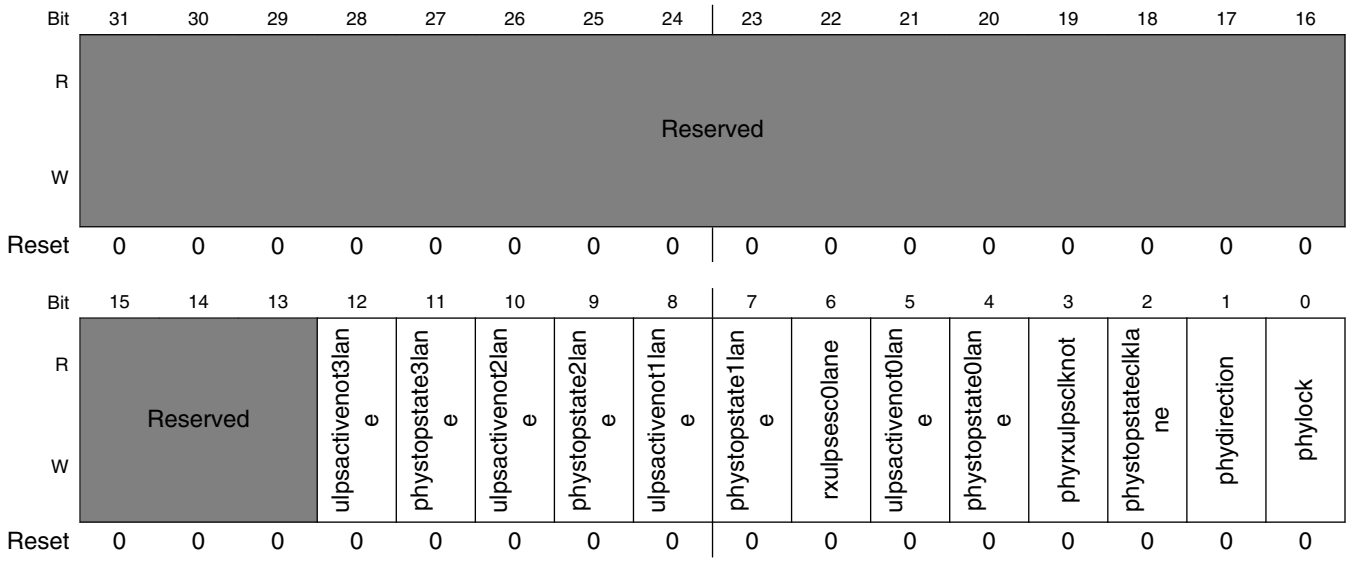
41.6.25 D-PHY PPI status interface (MIPI_DSI_PHY_STATUS)

Size: 32 bits

Offset: 0x60

Memory Access: R/W

Address: 21E_0000h base + 60h offset = 21E_0060h



MIPI_DSI_PHY_STATUS field descriptions

Field	Description
31–13 -	This field is reserved. Reserved
12 ulpsactivenot3lane	Reports status of ulpsactivenot3lane D-PHY pin
11 phystopstate3lane	Reports status of phystopstate3lane D-PHY pin
10 ulpsactivenot2lane	Reports status of ulpsactivenot2lane D-PHY pin
9 phystopstate2lane	Reports status of phystopstate2lane D-PHY pin
8 ulpsactivenot1lane	Reports status of ulpsactivenot1lane D-PHY pin
7 phystopstate1lane	Reports status of phystopstate1lane D-PHY pin
6 rxulpsesc0lane	Reports status of rxulpsEsc0lane D-PHY pin
5 ulpsactivenot0lane	Reports status of ulpsactivenot0lane D-PHY pin
4 phystopstate0lane	Reports status of phystopstate0lane D-PHY pin
3 phyrxulpsclknot	Reports status of phyrxulpsclknot D-PHY pin
2 phystopstateclklane	Reports status of phystopstateclklane D-PHY pin
1 phydirection	Reports status of phydirection D-PHY pin
0 phylock	Reports status of phylock D-PHY pin

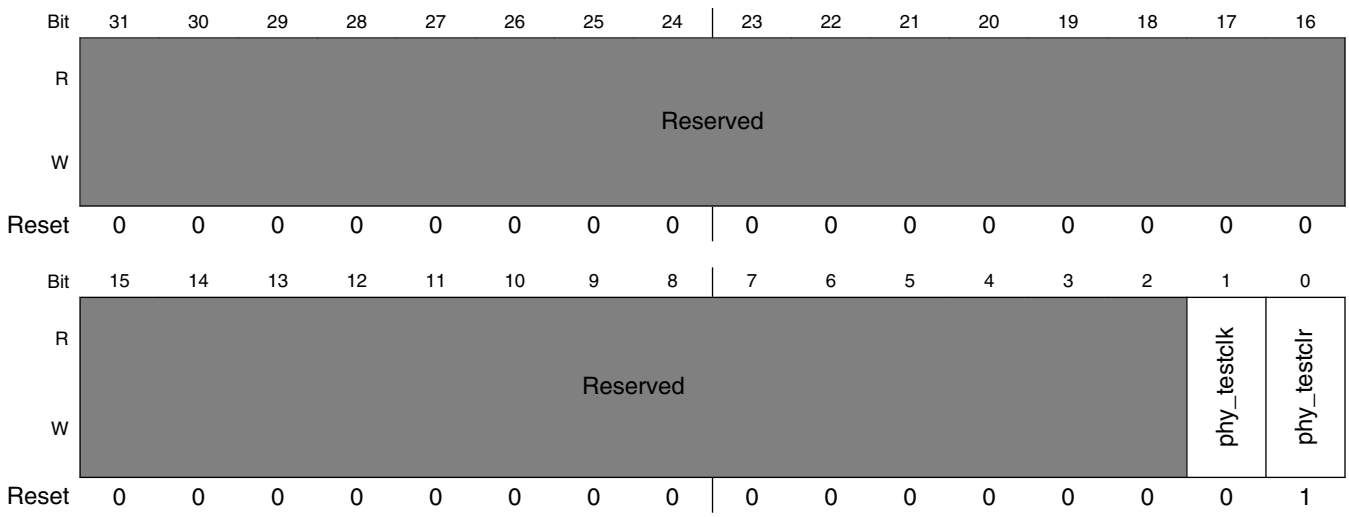
41.6.26 D-PHY Test interface control 0 (MIPI_DSI_PHY_TST_CTRL0)

Size: 32 bits

Offset: 0x64

Memory Access: R/W

Address: 21E_0000h base + 64h offset = 21E_0064h



MIPI_DSI_PHY_TST_CTRL0 field descriptions

Field	Description
31–2 -	This field is reserved. Reserved
1 phy_testclk	PHY test interface strobe signal. Used to clock TESTDIN bus into the D-PHY. In conjunction with TESTEN signal controls the operation selection
0 phy_testclr	PHY test interface clear. When active performs vendor specific interface initialization (Active High)

41.6.27 D-PHY Test interface control 1 (MIPI_DSI_PHY_TST_CTRL1)

Size: 32 bits

Offset: 0x68

Memory Access: R/W

Address: 21E_0000h base + 68h offset = 21E_0068h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															phy_testen
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	phy_testdout								phy_testdin							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_DSI_PHY_TST_CTRL1 field descriptions

Field	Description
31–17 -	This field is reserved. Reserved
16 phy_testen	PHY test interface operation selector: when 1 configures address write operation on the falling edge of TESTCLK; when 0 configures a data write operation on the rising edge of TESTCLK
15–8 phy_testdout	PHY output 8-bit data bus for read-back and internal probing functionalities
phy_testdin	PHY test interface input 8-bit data bus for internal register programming and test functionalities access

Chapter 42
MIPI HSI Host Controller (MIPI_HSI)

42.1 Overview

MIPI_HSI is an interface intended to connect an application processor to a cellular modem controller in cellular handsets, but it can be used in other applications.

This document describes the micro-architecture for the MIPI_HSI host controller and the block level functionality and implementation in detail.

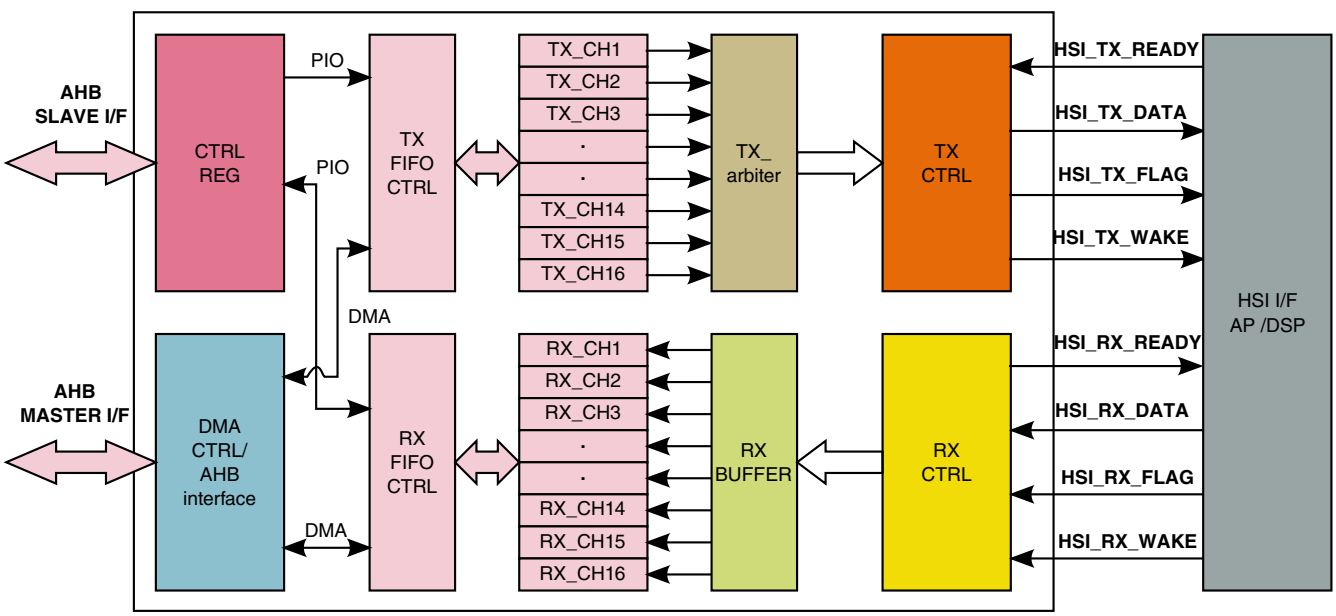


Figure 42-1. Top-Level Architecture

42.1.1 Features

The features of MIPI_HSI include the following:

- AHB slave interface support for PIO and configure.
- Supports PIO and DMA access mode
- Supports AHB Master interface to SOC interconnect
- Compatible with the MIPI_HSI specification version1.0 & MIPI_HSI Physical Layer v1.01.00 specification
- Full-Duplex High Speed Serial interface
- Supports 16 logical channels for both transmit and receive operations
- Each channel is configurable and can be enabled and disabled
- Bandwidth for each channel is programmable
- Supports both stream mode and frame mode for data transmission and reception
- Maximum bandwidth up to 100Mbs in both transmit and receive directions
- Configurable transmit and receive FIFO
- Programmable transmission bit rate
- Support round-robin arbitration for transmission mode
- Run-time configurability of channel ID bits
- Data time out feature supported for receive operation

42.2 External Signals

42.2.1 External Signals Overview

MIPI_HSI has 8 associate I/O signals.

- HSI_RX_DATA: Serial data bus
- HSI_RX_FLAG: Flag signal. If data on HSI_RX_DATA is toggling then flag stays constant and if data is constant then flag should toggle. It can be used with HSI_RX_DATA.
- HSI_RX_READY: Link data flow control signal. It indicates HSI RX is ready to receive.
- HSI_RX_WAKE: Link Control signal used to wake up the receiver.
- HSI_TX_DATA: Serial data out
- HSI_TX_FLAG: Flag signal. If data on HSI_TX_DATA is toggling, then flag stays constant and if data is constant, then flag should toggle.
- HSI_TX_READY: Link data flow control signal. It indicates HSI RX of other die is ready.
- HSI_TX_WAKE: Assertion of this signal shows that HSI TX wants to transfer some data. This signal is used to wake up HSI RX of other die.

42.2.2 Ports Table

See the following table for the signal properties of the I/Os.

Table 42-1. Properties of I/O Signals

Name	Port	Function	Reset State	Pull up
HSI_RX_DATA	I	Serial data bus	N/A	N/A
HSI_RX_FLAG	I	Flag signal. If data on HSI_RX_DATA is toggling then flag stays constant and if data is constant then flag should toggle. It can be used with HSI_RX_DATA.	N/A	N/A
HSI_RX_READY	O	Link data flow control signal. It indicates HSI RX is ready to receive	N/A	N/A
HSI_RX_WAKE	I	Link Control signal used to wake up the receiver so that transmitter starts a transmission	N/A	N/A
HSI_TX_DATA	O	Serial data out	0	N/A
HSI_TX_FLAG	O	Flag signal. If data on tx_data is toggling, then flag stays constant and if data is constant, then flag should toggle.	0	N/A
HSI_TX_READY	I	Link data flow control signal. It indicates HSI RX of other die is ready	N/A	N/A
HSI_TX_WAKE	O	Assertion of this signal shows that HSI TX wants to transfer some data. This signal is used to wake up HSI RX of other die	0	N/A

42.3 Clocks

The table found here describes the clock sources for MIPI HSI.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 42-2. MIPI HSI Clocks

Clock name	Clock Root	Description
h_clk	ahb_clk_root	AHB clock
tx_ref_clk	hsi_tx_clk_root	Tx reference clock

42.4 Functional Description

The following sections provide a brief functional description of the major system blocks, including the Data Channel Buffer, DMA AHB interface, register bank as well as AHB slave Bus interface, dual-port memory wrapper, clock & reset manager and clock generator.

42.4.1 DMA AHB /PIO AHB slave and Data Channel Buffer

The HSI uses one configurable data buffer so data can be transferred between the system bus (AHB slave Bus or AHB Master Bus) and the HSI controller.

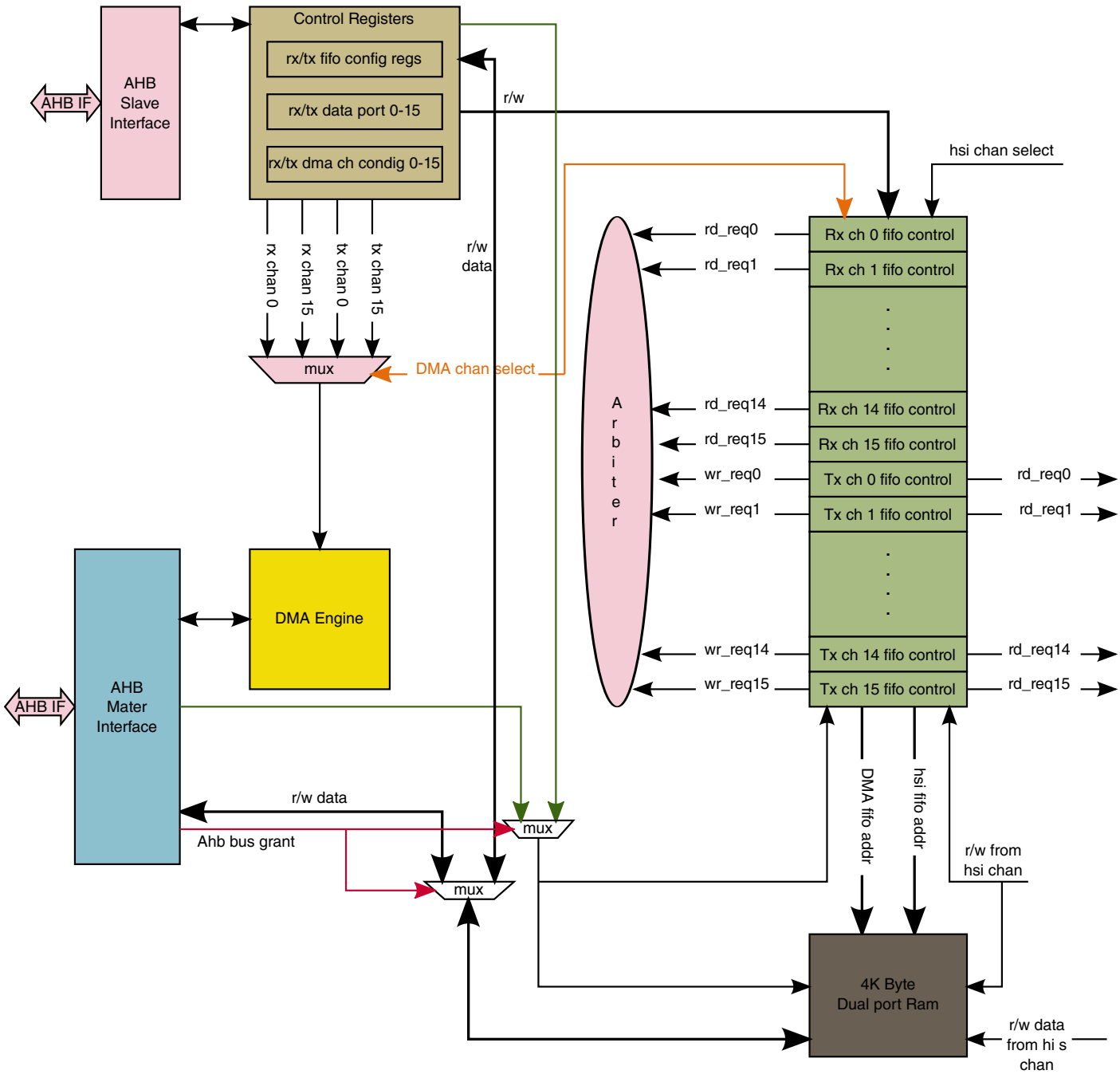


Figure 42-2. HSI DMA AHB/PIO AHB slave and data channel Buffer Scheme

42.4.1.1 AHB Slave Interface

This interface is compatible with AMBA AHB2.0 interface.

42.4.1.2 AHB Master Interface

This block access bus data by burst 8, burst 4 or single. AHB bus error is support.

42.4.1.3 Control Registers

This block has all the control registers of HSI Controller.

42.4.1.4 DMA Engine

This block generate DMA commands and send it to AHB master interface. These DMA commands are generated according current DMA channel configuration registers. AHB Master Interface should access AHB bus according the commands.

42.4.1.5 Rx/Tx FIFO controller

There are 32 sets of Rx/Tx FIFO Controller. Each FIFO Controller is corresponding to a Rx/Tx channel and a DMA channel. The depth for Each FIFO Controller is configurable. Once the number of data in the FIFO is less (for Tx) or more (for Rx) than the burst length of the corresponding DMA channel, it will send wr_req/rd_req signal to the Arbiter.

42.4.1.6 Arbiter

The arbiter arbitrates the requests from all 32 FIFO controllers according round robin priority.

42.4.1.7 4K Bytes Dual Port RAM

This block is used to save all FIFO data.

42.4.1.8 Typical Software Operation Flow for DMA

1. Configure Tx/Rx FIFO Size Configuration Registers. The total size of the FIFOs should not more than 1K DWords.

2. Enable RDMA/TDMA channel Complete Irq and Error Irq.

3. Configure RDMA/TDMA Start Address, Burst Size and Trans_length for each DMA channel. After that, set the RDMA/TDMA Enable bit. Once the Enable bit is set, the DMA channel will start to work right away.

4. Wait HSI Controller Irq and check RDMA/TDMA Complete Irq Stat bit.

42.4.1.9 Typical Software Operation Flow 1 for Data Port

1. Read corresponding Tx/Rx FIFO Status Register to make sure FIFO isn't empty/full.

2. Read/write Tx/Rx channel Data Port registers to access channel FIFO data.

42.4.1.10 Typical Software Operation Flow 2 for Data Port

1. Configure corresponding Tx/Rx FIFO Threshold Register.

2. Enable Tx/Rx FIFO Threshold interrupt.

3. Wait Tx/Rx FIFO Threshold interrupt.

4. Read/write Tx/Rx channel Data Port registers to access channel FIFO data.

42.4.2 HSI Reset

There are two kinds of reset signals within HSI:

1. Hardware reset
2. Software reset for all

These signals are fed into this module and stable signals are generated inside the module to reset all other modules. The module also gates off all the inside signals.

This module controls all the reset signals within the HSI.

42.5 HSI Memory Map/Register Definition

This section includes the module memory map and detailed descriptions of all registers.

NOTE

The HSI registers are 32 bits wide and only support 32-bit access.

NOTE

The term reset refers to power-on-reset. It will be specified whenever the software reset value differs from the power-on-reset.

MIPI_HSI memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_8000	HSI Control Register (MIPI_HSI_CTRL)	32	R/W	C800_0000h	42.5.1/3688
220_8004	HSI Tx Config Register (MIPI_HSI_TX_CONF)	32	R/W	0000_0000h	42.5.2/3690
220_8008	HSI Rx Config Register (MIPI_HSI_RX_CONF)	32	R/W	0000_0000h	42.5.3/3692
220_800C	HSI Capability Register (MIPI_HSI_CAP)	32	R	0007_1FFFh	42.5.4/3695
220_8010	HSI Tx Water Mark Level 0 Register (MIPI_HSI_TX_WML0)	32	R/W	0000_0000h	42.5.5/3696
220_8014	HSI Tx Water Mark Level 1 Register (MIPI_HSI_TX_TML1)	32	R/W	0000_0000h	42.5.6/3698
220_8018	HSI Tx Arbiter Priority 0 Register (MIPI_HSI_TX_ARB_PRI0)	32	R/W	0000_0000h	42.5.7/3700
220_801C	HSI Tx Arbiter Priority 1 Register (MIPI_HSI_TX_ARB_PRI1)	32	R/W	0000_0000h	42.5.8/3702
220_8020	HSI Line Status Register (MIPI_HSI_LINE_ST)	32	R	0000_0000h	42.5.9/3705
220_8024	HSI ID Bits Register (MIPI_HSI_ID_BIT)	32	R/W	0000_0000h	42.5.10/ 3707
220_8028	Tx and Rx Fif0 Threshold Configuration Register (MIPI_HSI_FIFO_THR_CONF)	32	R/W	0000_0000h	42.5.11/ 3708
220_802C	Tx and Rx Channel Soft Reset Register (MIPI_HSI_CH_SFTRST)	32	W	0000_0000h	42.5.12/ 3711
220_8030	HSI Interrupt Status Register (MIPI_HSI_IRQSTAT)	32	R/W	0000_0000h	42.5.13/ 3713
220_8034	HSI Interrupt Status Enable Register (MIPI_HSI_IRQSTAT_EN)	32	R/W	0000_0000h	42.5.14/ 3715
220_8038	HSI Interrupt Signal Enable Register (MIPI_HSI_IRQSIG_EN)	32	R/W	0000_0000h	42.5.15/ 3717
220_803C	HSI FIFO Threshold Interrupt Status Register (MIPI_HSI_FIFO_THR_IRQSTAT)	32	R	FFFF_0000h	42.5.16/ 3719
220_8040	HSI FIFO Threshold Interrupt Status Enable Register (MIPI_HSI_FIFO_THR_IRQSTAT_EN)	32	R/W	0000_0000h	42.5.17/ 3722

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_8044	HSI FIFO Threshold Interrupt Signal Enable Register (MIPI_HSI_FIFO_THR_IRQSIG_EN)	32	R/W	0000_0000h	42.5.18/ 3725
220_8050	Tx Channel n Data Port Register (MIPI_HSI_TX_CH0_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8054	Tx Channel n Data Port Register (MIPI_HSI_TX_CH1_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8058	Tx Channel n Data Port Register (MIPI_HSI_TX_CH2_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_805C	Tx Channel n Data Port Register (MIPI_HSI_TX_CH3_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8060	Tx Channel n Data Port Register (MIPI_HSI_TX_CH4_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8064	Tx Channel n Data Port Register (MIPI_HSI_TX_CH5_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8068	Tx Channel n Data Port Register (MIPI_HSI_TX_CH6_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_806C	Tx Channel n Data Port Register (MIPI_HSI_TX_CH7_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8070	Tx Channel n Data Port Register (MIPI_HSI_TX_CH8_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8074	Tx Channel n Data Port Register (MIPI_HSI_TX_CH9_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8078	Tx Channel n Data Port Register (MIPI_HSI_TX_CH10_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_807C	Tx Channel n Data Port Register (MIPI_HSI_TX_CH11_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8080	Tx Channel n Data Port Register (MIPI_HSI_TX_CH12_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8084	Tx Channel n Data Port Register (MIPI_HSI_TX_CH13_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8088	Tx Channel n Data Port Register (MIPI_HSI_TX_CH14_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_808C	Tx Channel n Data Port Register (MIPI_HSI_TX_CH15_DP)	32	R/W	0000_0000h	42.5.19/ 3728
220_8090	Rx Channel n Data Port Register (MIPI_HSI_RX_CH0_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_8094	Rx Channel n Data Port Register (MIPI_HSI_RX_CH1_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_8098	Rx Channel n Data Port Register (MIPI_HSI_RX_CH2_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_809C	Rx Channel n Data Port Register (MIPI_HSI_RX_CH3_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80A0	Rx Channel n Data Port Register (MIPI_HSI_RX_CH4_DP)	32	R/W	0000_0000h	42.5.20/ 3729

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_80A4	Rx Channel n Data Port Register (MIPI_HSI_RX_CH5_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80A8	Rx Channel n Data Port Register (MIPI_HSI_RX_CH6_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80AC	Rx Channel n Data Port Register (MIPI_HSI_RX_CH7_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80B0	Rx Channel n Data Port Register (MIPI_HSI_RX_CH8_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80B4	Rx Channel n Data Port Register (MIPI_HSI_RX_CH9_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80B8	Rx Channel n Data Port Register (MIPI_HSI_RX_CH10_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80BC	Rx Channel n Data Port Register (MIPI_HSI_RX_CH11_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80C0	Rx Channel n Data Port Register (MIPI_HSI_RX_CH12_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80C4	Rx Channel n Data Port Register (MIPI_HSI_RX_CH13_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80C8	Rx Channel n Data Port Register (MIPI_HSI_RX_CH14_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80CC	Rx Channel n Data Port Register (MIPI_HSI_RX_CH15_DP)	32	R/W	0000_0000h	42.5.20/ 3729
220_80D0	HSI Error Interrupt Status Register (MIPI_HSI_ERR_IRQSTAT)	32	R	0000_0000h	42.5.21/ 3730
220_80D4	HSI Error Interrupt Status Enable Register (MIPI_HSI_ERR_IRQSTAT_EN)	32	R/W	0000_0000h	42.5.22/ 3733
220_80D8	HSI Error Interrupt Signal Enable Register (MIPI_HSI_ERR_IRQSIG_EN)	32	R/W	0000_0000h	42.5.23/ 3735
220_80DC	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA0_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80E0	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA1_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80E4	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA2_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80E8	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA3_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80EC	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA4_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80F0	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA5_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80F4	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA6_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_80F8	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA7_CONF)	32	R/W	0000_0000h	42.5.24/ 3737

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_80FC	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA8_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8100	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA9_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8104	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA10_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8108	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA11_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_810C	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA12_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8110	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA13_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8114	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA14_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_8118	Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA15_CONF)	32	R/W	0000_0000h	42.5.24/ 3737
220_811C	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA0_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8120	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA1_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8124	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA2_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8128	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA3_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_812C	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA4_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8130	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA5_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8134	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA6_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8138	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA7_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_813C	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA8_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8140	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA9_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8144	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA10_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8148	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA11_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_814C	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA12_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8150	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA13_CONF)	32	R/W	0000_0000h	42.5.25/ 3738

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_8154	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA14_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_8158	Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA15_CONF)	32	R/W	0000_0000h	42.5.25/ 3738
220_815C	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA0_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8160	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA1_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8164	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA2_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8168	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA3_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_816C	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA4_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8170	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA5_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8174	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA6_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8178	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA7_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_817C	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA8_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8180	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA9_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8184	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA10_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8188	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA11_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_818C	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA12_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8190	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA13_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8194	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA14_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_8198	Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA15_STA_ADDR)	32	R/W	0000_0000h	42.5.26/ 3739
220_819C	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA0_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81A0	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA1_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81A4	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA2_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81A8	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA3_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_81AC	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA4_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81B0	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA5_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81B4	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA6_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81B8	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA7_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81BC	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA8_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81C0	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA9_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81C4	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA10_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81C8	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA11_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81CC	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA12_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81D0	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA13_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81D4	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA14_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81D8	Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA15_STA_ADDR)	32	R/W	0000_0000h	42.5.27/ 3739
220_81DC	DMA Interrupt Status Register (MIPI_HSI_DMA_IRQSTAT)	32	R	0000_0000h	42.5.28/ 3740
220_81E0	DMA Interrupt Enable Register (MIPI_HSI_DMA_IRQSTAT_EN)	32	R/W	0000_0000h	42.5.29/ 3742
220_81E4	DMA Interrupt Status Signal Enable Register (MIPI_HSI_DMA_IRQSIG_EN)	32	R/W	0000_0000h	42.5.30/ 3744
220_81E8	DMA Error Interrupt Status Register (MIPI_HSI_DMA_ERR_IRQSTAT)	32	R	0000_0000h	42.5.31/ 3747
220_81EC	DMA Error Interrupt Enable Register (MIPI_HSI_DMA_ERR_IRQSTAT_EN)	32	R/W	0000_0000h	42.5.32/ 3749
220_81F0	DMA Error Interrupt Signal Enable Register (MIPI_HSI_DMA_ERR_IRQSIG_EN)	32	R/W	0000_0000h	42.5.33/ 3751
220_81F4	DMA Single Request Enable Register (MIPI_HSI_DMA_SINGLE_REQ_EN)	32	R	0000_0000h	42.5.34/ 3754
220_8200	Tx Fifo Size Configuration Register 0 (MIPI_HSI_TX_FIFO_SIZE_CONF0)	32	R/W	5555_5555h	42.5.35/ 3756
220_8204	Tx Fifo Size Configuration Register 1 (MIPI_HSI_TX_FIFO_SIZE_CONF1)	32	R/W	5555_5555h	42.5.36/ 3759
220_8208	Rx Fifo Size Configuration Register 0 (MIPI_HSI_RX_FIFO_SIZE_CONF0)	32	R/W	5555_5555h	42.5.37/ 3762

Table continues on the next page...

MIPI_HSI memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
220_820C	Rx Fifo Size Configuration Register 1 (MIPI_HSI_RX_FIFO_SIZE_CONF1)	32	R/W	5555_5555h	42.5.38/ 3765
220_8210	Tx Fifo Status Register (MIPI_HSI_TX_FIFO_STAT)	32	R	5555_5555h	42.5.39/ 3768
220_8214	Rx Fifo Status Register (MIPI_HSI_RX_FIFO_STAT)	32	R	5555_5555h	42.5.40/ 3770
220_8228	Ahb Master Config Register (MIPI_HSI_AHB_MASTER_CONF)	32	R/W	0000_0180h	42.5.41/ 3772
220_822C	TX Break Length Register (MIPI_HSI_TX_BREAK_LEN)	32	R/W	0000_0025h	42.5.42/ 3773

42.5.1 HSI Control Register (MIPI_HSI_CTRL)

This register contains module soft reset, clock gating, clock divisor and so on.

Address: 220_8000h base + 0h offset = 220_8000h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	SFTRST	CLKGATE	Reserved		DMA_ DISABLE	RX_DLY_SEL			RX_FRAME_BRST_CNT							
W																
Reset	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Reserved		RX_TAIL_ BIT_CNT		DATA_TIMEOUT_CNT				Reserved			TX_BREAK	TX_CLK_DIVISOR			
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_CTRL field descriptions

Field	Description
31 SFTRST	Set this bit to zero to enable normal HSI operation. Set this bit to one (default) to disable clocking with the HSI and hold it in its reset (lowest power) state. This bit can be turned on and then off to reset the HSI block to its default state.
30 CLKGATE	This bit must be set to zero for normal operation. When set to one it gates off the clocks to the block.

Table continues on the next page...

MIPI_HSI_CTRL field descriptions (continued)

Field	Description
29–28 Reserved	This field is reserved. Reserved, always set to zero.
27 DMA_DISABLE	This bit must be set to zero for any DMA operation. When set to one it disabel all the DMA channels.
26–24 RX_DLY_SEL	These values denote the tap delay values for reception of data and flag. 000 0ns ; 001 1ns ; 010 2ns ; 011 3ns ; 100 4ns ; 101 5ns ; 110 6ns ; 111 7ns ;
23–16 RX_FRAME_ BRST_CNT	This value is to limit the continous Frame transmission count in Pipelined Data flow. The Receiver Frame Burst counter shall be able to support upto 256 frames of continous transfer. 7'h00 256 frames transmission count is set. 7'h01 1 frames transmission count is set. 7'h02 2 frames transmission count is set. 7'hff 255 frames transmission count is set.
15–14 Reserved	This field is reserved. Reserved.
13–12 RX_TAIL_BIT_ CNT	The value determines the length of the Tailing bit counter. The receiver shall start Receiver Tailing bit counter after the nth frame programmed in Rx Frame Burst counter is received. The receiver shall then drive ready to logic one if the receiver Tailing-bit counter has completed with no errors detected, and the receiver has enough room for at least one new frame. 00 800-> tx_refclk 01 400-> tx_refclk 10 200-> tx_refclk 11 100-> tx_refclk
11–8 DATA_ TIMEOUT_CNT	This value determines the interval by which DATA timeouts are detected. This data timeout counter logic is used only for Receive operations. The counter should start counting when data in any of the RX channel fifo is less than the threshold value and resets to zero when there is a threshold reached interrupt from any of the RX buffers. The counter value should be zero, when RX fifo is empty. An interrupt will be asserted to the host driver, when the counter value reaches the data timeout counter value. 1110 HSI Tx Clock x 2 ^ 27 0001 HSI Tx Clock x 2 ^ 14 0000 HSI Tx Clock x 2 ^ 13

Table continues on the next page...

MIPI_HSI_CTRL field descriptions (continued)

Field	Description
7–5 Reserved	This field is reserved. Reserved.
4 TX_BREAK	Seeting this bit to one trigger a transmission break at HSI Tx. Once this bit is set to one, the HSI controller will send a serise zeros on HSI_TX_DATA port according to the tx break co unt. It will be automatically cleared, when the send is finished.
TX_CLK_ DIVISOR	This register holds the divisor of the base clock (tx_refclk) frequency for HSI Tx clock (internal clock used to drive Transmitter interface). 1000 tx_refclk divided by 256 0111 tx_refclk divided by 128 0110 tx_refclk divided by 64 0101 tx_refclkdivided by 32 0100 tx_refclk divided by 16 0011 tx_refclk divided by 8 0010 tx_refclk divided by 4 0001 tx_refclk divided by 2 0000 tx_refclk divided by 1

42.5.2 HSI Tx Config Register (MIPI_HSI_TX_CONF)

This register contains the configurations of tx channel enable/disable, tx wakup and tx trans mode.

Address: 220_8000h base + 4h offset = 220_8004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CH15_EN	CH14_EN	CH13_EN	CH12_EN	CH11_EN	CH10_EN	CH9_EN	CH8_EN	CH7_EN	CH6_EN	CH5_EN	CH4_EN	CH3_EN	CH2_EN	CH1_EN	CH0_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved				TIMEOUT_CNT				Reserved						WAKEUP	TRANS_MODE
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_CONF field descriptions

Field	Description
31 CH15_EN	1 Tx Ch15 is Enabled. 0 Tx Ch15 is Disabled.
30 CH14_EN	1 Tx Ch14 is Enabled. 0 Tx Ch14 is Disabled.
29 CH13_EN	1 Tx Ch13 is Enabled. 0 Tx Ch13 is Disabled.
28 CH12_EN	1 Tx Ch12 is Enabled. 0 Tx Ch12 is Disabled.
27 CH11_EN	1 Tx Ch11 is Enabled. 0 Tx Ch11 is Disabled.
26 CH10_EN	1 Tx Ch10 is Enabled. 0 Tx Ch10 is Disabled.
25 CH9_EN	1 Tx Ch9 is Enabled. 0 Tx Ch9 is Disabled.
24 CH8_EN	1 Tx Ch8 is Enabled. 0 Tx Ch8 is Disabled.
23 CH7_EN	1 Tx Ch7 is Enabled. 0 Tx Ch7 is Disabled.
22 CH6_EN	1 Tx Ch6 is Enabled. 0 Tx Ch6 is Disabled.
21 CH5_EN	1 Tx Ch5 is Enabled. 0 Tx Ch5 is Disabled.
20 CH4_EN	1 Tx Ch4 is Enabled. 0 Tx Ch4 is Disabled.
19 CH3_EN	1 Tx Ch3 is Enabled. 0 Tx Ch3 is Disabled.
18 CH2_EN	1 Tx Ch2 is Enabled. 0 Tx Ch2 is Disabled.
17 CH1_EN	1 Tx Ch1 is Enabled. 0 Tx Ch1 is Disabled.
16 CH0_EN	1 Tx Ch0 is Enabled. 0 Tx Ch0 is Disabled.
15–12 Reserved	This field is reserved. Reserved.
11–8 TIMEOUT_CNT	0000 tx timeout value 2^14 tx_refclk 0001 tx timeout value 2^15 tx_refclk 0010 tx timeout value 2^16 tx_refclk 0011 tx timeout value 2^17 tx_refclk 1110 tx timeout value 2^28 tx_refclk 1111 tx timeout value 2^29 tx_refclk
7–2 Reserved	This field is reserved. Reserved.
1 WAKEUP	When this bit gets set to one, HSI transmitter sends HSI_TX_WAKE signal to Rx of other device. For a transmit operation this bit should be one.

Table continues on the next page...

MIPI_HSI_TX_CONF field descriptions (continued)

Field	Description
	0 Transmitter is in Sleep State 1 Transmitter is in Wakeup State.
0 TRANS_MODE	0 Stream Transmission Mode 1 Frame Transmission Mode

42.5.3 HSI Rx Config Register (MIPI_HSI_RX_CONF)

This register contains the configurations of rx channel enable/disable, rx wakup and rx trans mode, rx data flow.

Address: 220_8000h base + 8h offset = 220_8008h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	CH15_EN	CH14_EN	CH13_EN	CH12_EN	CH11_EN	CH10_EN	CH9_EN	CH8_EN	CH7_EN	CH6_EN	CH5_EN	CH4_EN	CH3_EN	CH2_EN	CH1_EN	CH0_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved	TIMEOUT_CNT							Reserved			TAIL_BIT_CNT_EN	REC_MODE	WAKE	DATA_FLOW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_RX_CONF field descriptions

Field	Description
31 CH15_EN	1 Rx Ch15 is Enabled. 0 Rx Ch15 is Disabled.
30 CH14_EN	1 Rx Ch14 is Enabled. 0 Rx Ch14 is Disabled.

Table continues on the next page...

MIPI_HSI_RX_CONF field descriptions (continued)

Field	Description
29 CH13_EN	1 Rx Ch13 is Enabled. 0 Rx Ch13 is Disabled.
28 CH12_EN	1 Rx Ch12 is Enabled. 0 Rx Ch12 is Disabled.
27 CH11_EN	1 Rx Ch11 is Enabled. 0 Rx Ch11 is Disabled.
26 CH10_EN	1 Rx Ch10 is Enabled. 0 Rx Ch10 is Disabled.
25 CH9_EN	1 Rx Ch9 is Enabled. 0 Rx Ch9 is Disabled.
24 CH8_EN	1 Rx Ch8 is Enabled. 0 Rx Ch8 is Disabled.
23 CH7_EN	1 Rx Ch7 is Enabled. 0 Rx Ch7 is Disabled.
22 CH6_EN	1 Rx Ch6 is Enabled. 0 Rx Ch6 is Disabled.
21 CH5_EN	1 Rx Ch5 is Enabled. 0 Rx Ch5 is Disabled.
20 CH4_EN	1 Rx Ch4 is Enabled. 0 Rx Ch4 is Disabled.
19 CH3_EN	1 Rx Ch3 is Enabled. 0 Rx Ch3 is Disabled.
18 CH2_EN	1 Rx Ch2 is Enabled. 0 Rx Ch2 is Disabled.
17 CH1_EN	1 Rx Ch1 is Enabled. 0 Rx Ch1 is Disabled.
16 CH0_EN	1 Rx Ch0 is Enabled. 0 Rx Ch0 is Disabled.
15 Reserved	This field is reserved. Reserved.
14–8 TIMEOUT_CNT	Receive Frame Timeout Counter: The counter shall be started when the first bit of the Frame has been found. The counter shall be stopped once the receiver has received the correct number of bits for a Frame. If the counter expires before Frame reception is completed, the receiver will signal to the protocol layer that it has found an incomplete Frame and asserts Rx Error Interrupt. 7'h0 14800 ---> tx_refclk 7'h1 16400 ---> tx_refclk 7'h2 18000 ---> tx_refclk 7'h4 19600 ---> tx_refclk 7'h8 21200 ---> tx_refclk 7'h10 22800 ---> tx_refclk 7'h20 24400 ---> tx_refclk

Table continues on the next page...

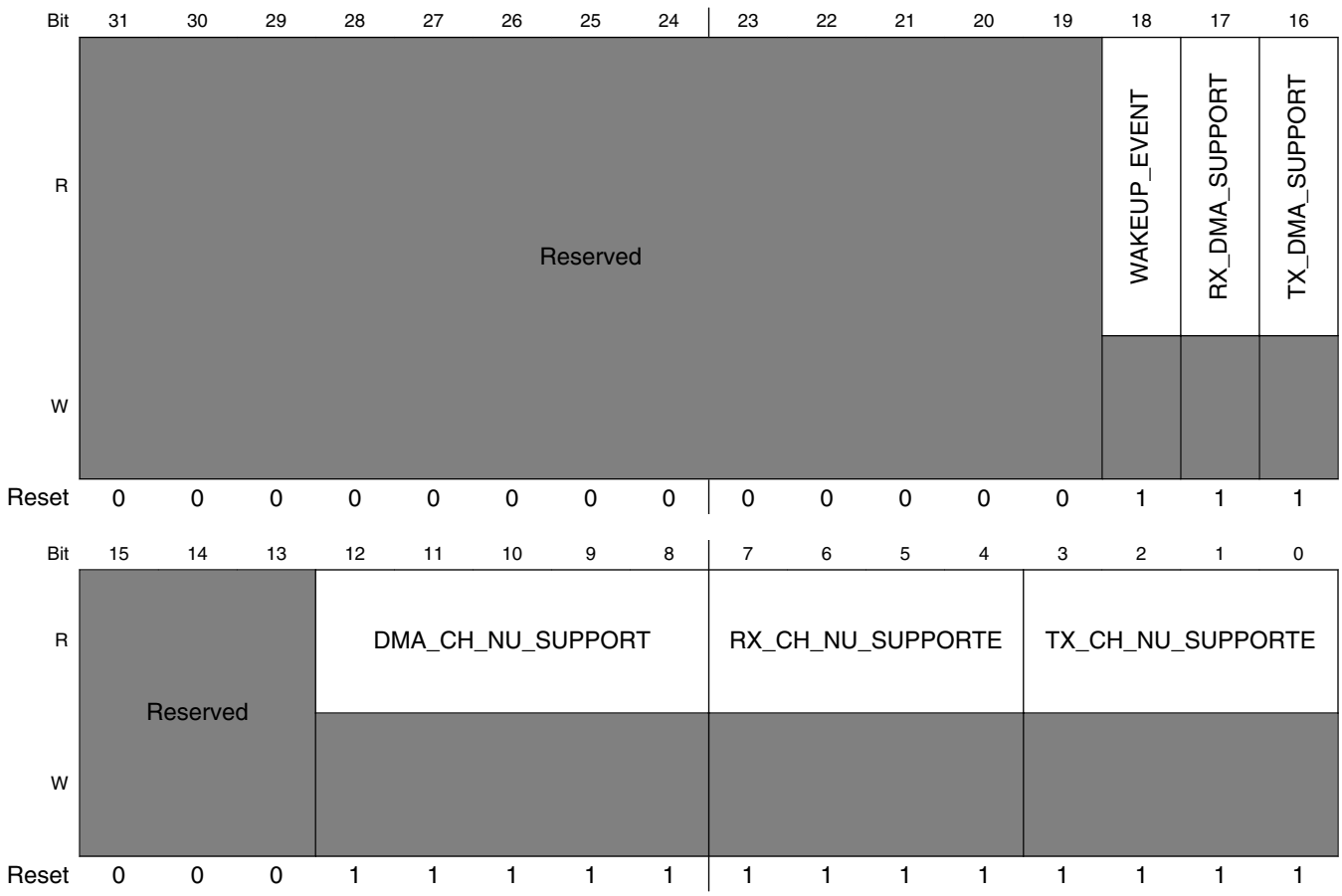
MIPI_HSI_RX_CONF field descriptions (continued)

Field	Description
	7'h40 26000 ---> tx_refclk
7–5 Reserved	This field is reserved. Reserved.
4 TAIL_BIT_CNT_EN	0 Tailing bit counter disable 1 Tailing bit counter Enable
3 REC_MODE	0 Stream Receive Mode 1 Frame Receive Mode
2 WAKE	0 Receiver is in Sleep State 1 Receiver is in Wakeup State
DATA_FLOW	00 Synchronized Data Flow 01 Pipelined Data Flow 10 Receiver Real-time Data Flow 11 Reserved

42.5.4 HSI Capability Register (MIPI_HSI_CAP)

This register contains the HSI controller Capability information.

Address: 220_8000h base + Ch offset = 220_800Ch



MIPI_HSI_CAP field descriptions

Field	Description
31–19 Reserved	This field is reserved. Reserved.
18 WAKEUP_ EVENT	0 Wakeup Event is supported 1 Wakeup Event is not supported.
17 RX_DMA_ SUPPORT	1 DMA is supported. 0 Not supported
16 TX_DMA_ SUPPORT	1 DMA is supported. 0 Not supported

Table continues on the next page...

MIPI_HSI_CAP field descriptions (continued)

Field	Description
15–13 Reserved	This field is reserved. Reserved.
12–8 DMA_CH_NU_ SUPPORT	00000 1 DMA supported 00001 2 DMA supported 00010 3 DMA supported 00011 4 DMA supported 00100 5 DMA supported 00101 6 DMA supported 11110 31 DMA supported 11111 32 DMA supported
7–4 RX_CH_NU_ SUPPORTE	0000 1 Rx channel supported 0001 2 Rx channels supported 1111 16 Rx channels supported
TX_CH_NU_ SUPPORTE	0000 1 Tx channel supported 0001 2 Tx channels supported 1111 16 Tx channels supported

42.5.5 HSI Tx Water Mark Level 0 Register (MIPI_HSI_TX_WML0)

This register contains HSI controller Tx channel Water Mark Level information.

Address: 220_8000h base + 10h offset = 220_8010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH15				CH14				CH13				CH12				CH11				CH10				CH9				CH8			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_WML0 field descriptions

Field	Description
31–28 CH15	This value denotes the WML of Tx Channel 15. When > 1010 Reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
27–24 CH14	This value denotes the WML of Tx Channel 14. When > 1010 reserved

Table continues on the next page...

MIPI_HSI_TX_WML0 field descriptions (continued)

Field	Description
	0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
23–20 CH13	This value denotes the WML of Tx Channel 13. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
19–16 CH12	This value denotes the WML of Tx Channel 12. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
15–12 CH11	This value denotes the WML of Tx Channel 11. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
11–8 CH10	This value denotes the WML of Tx Channel 10. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
7–4 CH9	This value denotes the WML of Tx Channel 9. When > 1010 reserved 0000 1 0001 2

Table continues on the next page...

MIPI_HSI_TX_WML0 field descriptions (continued)

Field	Description
	0010 4 1000 256 1001 512 1010 1024
CH8	This value denotes the WML of Tx Channel 8. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024

42.5.6 HSI Tx Water Mark Level 1 Register (MIPI_HSI_TX_TML1)

This register contains HSI controller Tx channel Water Mark Level information.

This register contains HSI controller Tx channel bandwidth information.

Address: 220_8000h base + 14h offset = 220_8014h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH7				CH6				CH5				CH4				CH3				CH2				CH1				CH0			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_TML1 field descriptions

Field	Description
31–28 CH7	This value denotes the WML of Tx Channel 7. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
27–24 CH6	This value denotes the WML of Tx Channel 6. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256

Table continues on the next page...

MIPI_HSI_TX_TML1 field descriptions (continued)

Field	Description
	1001 512 1010 1024
23–20 CH5	This value denotes the WML of Tx Channel 5. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
19–16 CH4	This value denotes the WML of Tx Channel 4. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
15–12 CH3	This value denotes the WML of Tx Channel 3. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
11–8 CH2	This value denotes the WML of Tx Channel 2. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024
7–4 CH1	This value denotes the WML of Tx Channel 1. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256

Table continues on the next page...

MIPI_HSI_TX_TML1 field descriptions (continued)

Field	Description
	1001 512 1010 1024
CH0	This value denotes the WML of Tx Channel 0. When > 1010 reserved 0000 1 0001 2 0010 4 1000 256 1001 512 1010 1024

42.5.7 HSI Tx Arbiter Priority 0 Register (MIPI_HSI_TX_ARB_PRI0)

This is HSI Tx Arbiter Priority Register.

Address: 220_8000h base + 18h offset = 220_8018h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH7				CH6				CH5				CH4				CH3				CH2				CH1				CH0			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_ARB_PRI0 field descriptions

Field	Description
31–28 CH7	This value denotes the priority of Tx Channel 7. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
27–24 CH6	This value denotes the priority of Tx Channel 6. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority

Table continues on the next page...

MIPI_HSI_TX_ARB_PRI0 field descriptions (continued)

Field	Description
	1110 15th priority 1111 16th priority
23–20 CH5	This value denotes the priority of Tx Channel 5. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
19–16 CH4	This value denotes the priority of Tx Channel 4. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
15–12 CH3	This value denotes the priority of Tx Channel 3. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
11–8 CH2	This value denotes the priority of Tx Channel 2. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
7–4 CH1	This value denotes the priority of Tx Channel 1. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority

Table continues on the next page...

MIPI_HSI_TX_ARB_PRI0 field descriptions (continued)

Field	Description
	1110 15th priority 1111 16th priority
CH0	This value denotes the priority of Tx Channel 0. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority

42.5.8 HSI Tx Arbiter Priority 1 Register (MIPI_HSI_TX_ARB_PRI1)

Address: 220_8000h base + 1Ch offset = 220_801Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH15				CH14				CH13				CH12				CH11				CH10				CH9				CH8			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_ARB_PRI1 field descriptions

Field	Description
31–28 CH15	This value denotes the priority of Tx Channel 15. When > 1010 Reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
27–24 CH14	This value denotes the priority of Tx Channel 14. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
23–20 CH13	This value denotes the priority of Tx Channel 13. When > 1010 reserved

Table continues on the next page...

MIPI_HSI_TX_ARB_PRI1 field descriptions (continued)

Field	Description
	0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
19–16 CH12	This value denotes the priority of Tx Channel 12. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
15–12 CH11	This value denotes the priority of Tx Channel 11. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
11–8 CH10	This value denotes the priority of Tx Channel 10. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
7–4 CH9	This value denotes the priority of Tx Channel 9. When > 1010 reserved 0000 1st prioirity 0001 2nd priority 0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority
CH8	This value denotes the priority of Tx Channel 8. When > 1010 reserved 0000 1st prioirity 0001 2nd priority

Table continues on the next page...

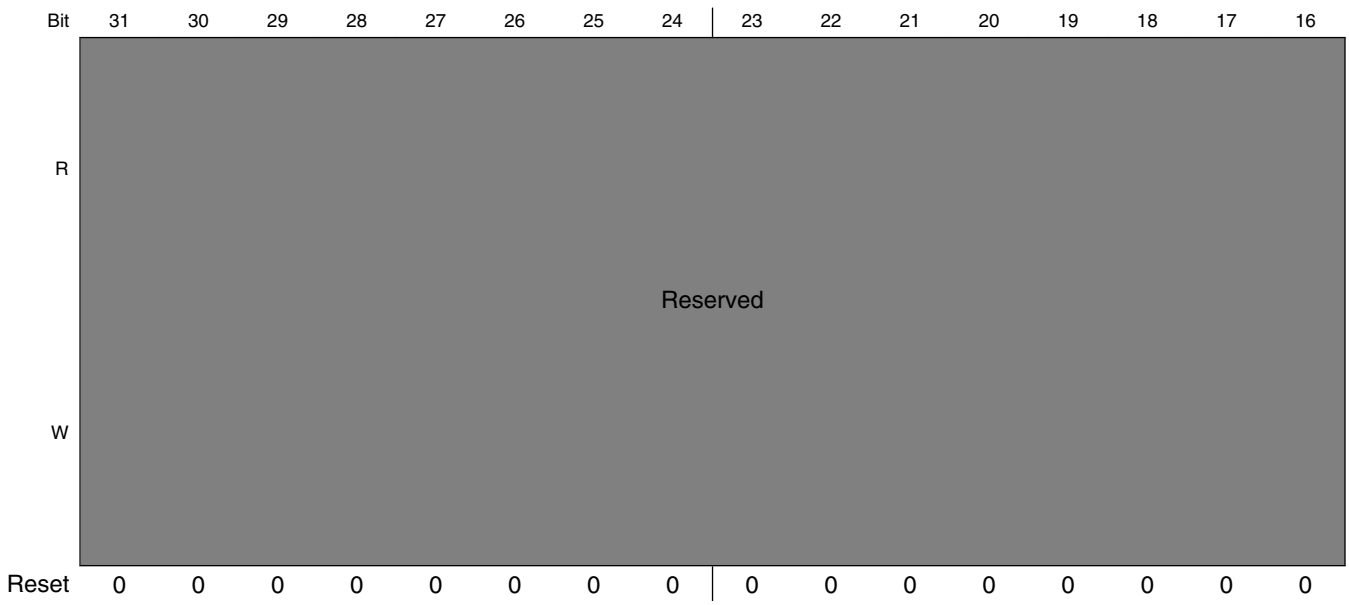
MIPI_HSI_TX_ARB_PRI1 field descriptions (continued)

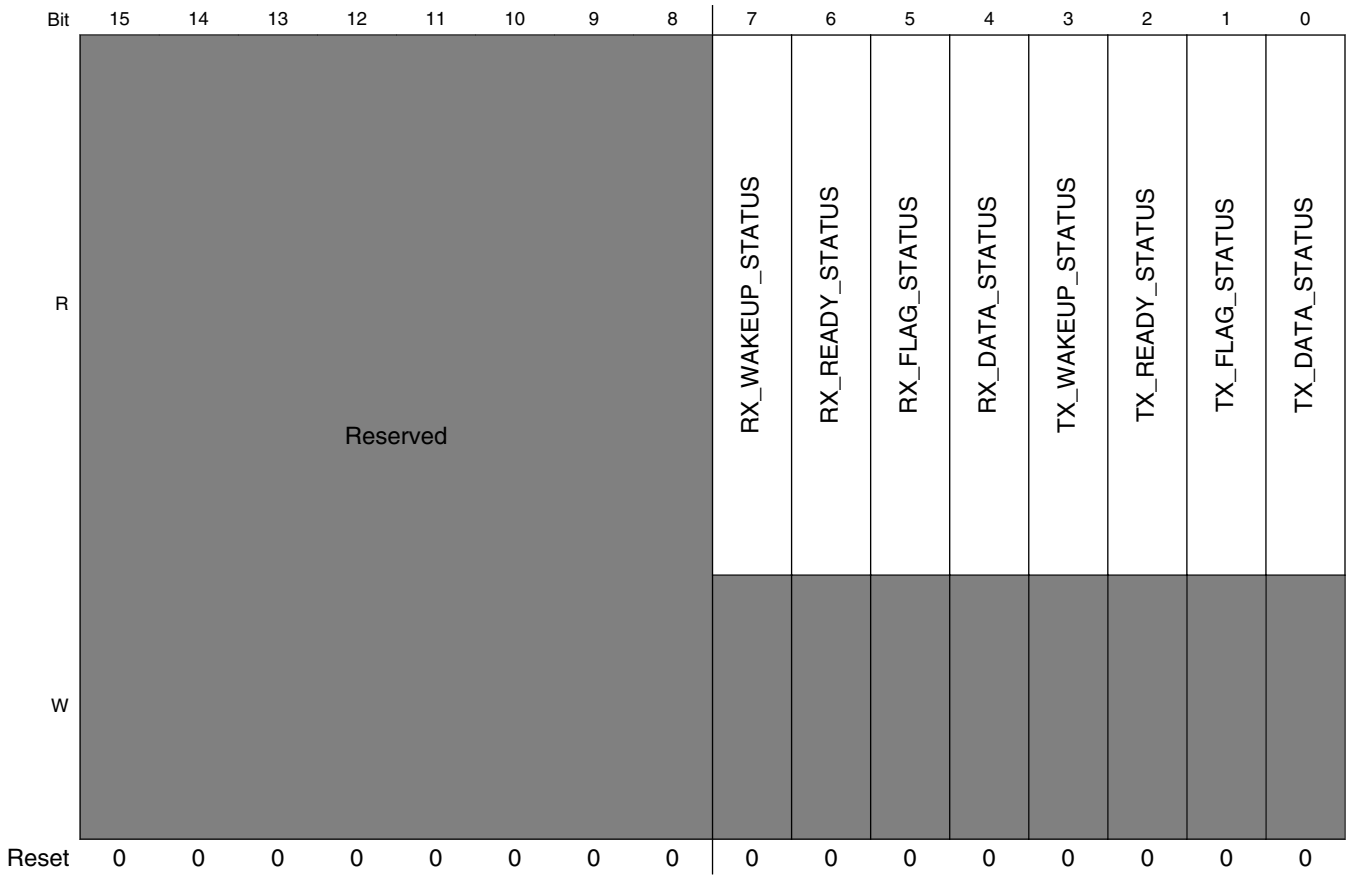
Field	Description
	0010 3rd priority 1101 14th priority 1110 15th priority 1111 16th priority

42.5.9 HSI Line Status Register (MIPI_HSI_LINE_ST)

This register contains the HSI controller line status for debug.

Address: 220_8000h base + 20h offset = 220_8020h





MIPI_HSI_LINE_ST field descriptions

Field	Description
31–8 Reserved	This field is reserved. Reserved, always set to zero.
7 RX_WAKEUP_ STATUS	This field reflects the rx_wake pin(only for debug).
6 RX_READY_ STATUS	This field reflects the rx_rdy pin(only for debug).
5 RX_FLAG_ STATUS	This field reflects the rx_flag pin(only for debug).
4 RX_DATA_ STATUS	This field reflects the rx_data pin(only for debug).
3 TX_WAKEUP_ STATUS	This field reflects the tx_wake pin(only for debug).
2 TX_READY_ STATUS	This field reflects the tx_ready pin(only for debug).

Table continues on the next page...

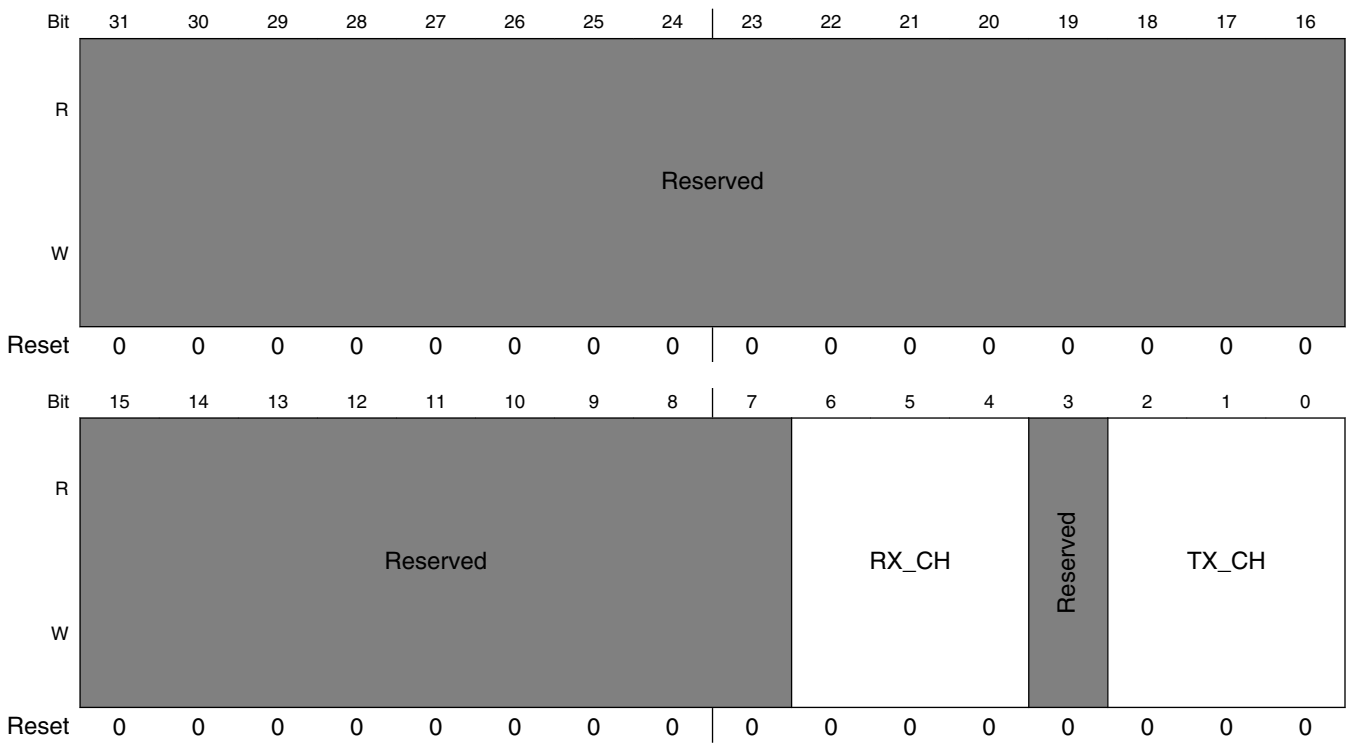
MIPI_HSI_LINE_ST field descriptions (continued)

Field	Description
1 TX_FLAG_ STATUS	This field reflects the tx_flag pin(only for debug).
0 TX_DATA_ STATUS	This field reflects the tx_data pin(only for debug).

42.5.10 HSI ID Bits Register (MIPI_HSI_ID_BIT)

This register contains the configurations of tx channel enable/disable, , tx wakup and tx trans mode.

Address: 220_8000h base + 24h offset = 220_8024h



MIPI_HSI_ID_BIT field descriptions

Field	Description
31–7 Reserved	This field is reserved. Reserved, always set to zero.
6–4 RX_CH	This bit sets the number of channel ID bits per frame or stream for a Receive operation.

Table continues on the next page...

MIPI_HSI_ID_BIT field descriptions (continued)

Field	Description
	0 0 bit 1 1 bit 2 2 bits 3 3 bits 4 4 bits
3 Reserved	This field is reserved. Reserved, always set to zero.
TX_CH	This bit sets the number of channel ID bits per frame or stream for a transmit operation. 0 0 bit 1 1 bit 2 2 bits 3 3 bits 4 4 bits

42.5.11 Tx and Rx Fif0 Threshold Configuration Register (MIPI_HSI_FIFO_THR_CONF)

This register sets the threshold level for each Tx and Rx channel fifo

Address: 220_8000h base + 28h offset = 220_8028h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TX_CH15	TX_CH14	TX_CH13	TX_CH12	TX_CH11	TX_CH10	TX_CH9	TX_CH8	TX_CH7	TX_CH6	TX_CH5	TX_CH4	TX_CH3	TX_CH2	TX_CH1	TX_CH0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RX_CH15	RX_CH14	RX_CH13	RX_CH12	RX_CH11	RX_CH10	RX_CH9	RX_CH8	RX_CH7	RX_CH6	RX_CH5	RX_CH4	RX_CH3	RX_CH2	RX_CH1	RX_CH0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_FIFO_THR_CONF field descriptions

Field	Description
31 TX_CH15	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)

Table continues on the next page...

MIPI_HSI_FIFO_THR_CONF field descriptions (continued)

Field	Description
30 TX_CH14	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
29 TX_CH13	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
28 TX_CH12	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
27 TX_CH11	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
26 TX_CH10	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
25 TX_CH9	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
24 TX_CH8	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
23 TX_CH7	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
22 TX_CH6	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
21 TX_CH5	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
20 TX_CH4	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
19 TX_CH3	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
18 TX_CH2	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
17 TX_CH1	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
16 TX_CH0	0 Half Empty (fifo size / 2) 1 Almost Empty (fifo size / 4)
15 RX_CH15	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
14 RX_CH14	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
13 RX_CH13	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
12 RX_CH12	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
11 RX_CH11	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
10 RX_CH10	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)

Table continues on the next page...

MIPI_HSI_FIFO_THR_CONF field descriptions (continued)

Field	Description
9 RX_CH9	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
8 RX_CH8	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
7 RX_CH7	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
6 RX_CH6	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
5 RX_CH5	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
4 RX_CH4	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
3 RX_CH3	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
2 RX_CH2	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
1 RX_CH1	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)
0 RX_CH0	0 Half Full (fifo size / 2) 1 Almost Full (3/4th of fifo size)

42.5.12 Tx and Rx Channel Soft Reset Register (MIPI_HSI_CH_SFTRST)

This register is used to reset each Tx and Rx Channel

Address: 220_8000h base + 2Ch offset = 220_802Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R																
W	TX_CH15	TX_CH14	TX_CH13	TX_CH12	TX_CH11	TX_CH10	TX_CH9	TX_CH8	TX_CH7	TX_CH6	TX_CH5	TX_CH4	TX_CH3	TX_CH2	TX_CH1	TX_CH0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R																
W	RX_CH15	RX_CH14	RX_CH13	RX_CH12	RX_CH11	RX_CH10	RX_CH9	RX_CH8	RX_CH7	RX_CH6	RX_CH5	RX_CH4	RX_CH3	RX_CH2	RX_CH1	RX_CH0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_CH_SFTRST field descriptions

Field	Description
31 TX_CH15	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 15 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
30 TX_CH14	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 14 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
29 TX_CH13	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 13 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
28 TX_CH12	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 12 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
27 TX_CH11	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 11 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
26 TX_CH10	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 10 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
25 TX_CH9	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 9 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.

Table continues on the next page...

MIPI_HSI_CH_SFTRST field descriptions (continued)

Field	Description
24 TX_CH8	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 8 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
23 TX_CH7	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 7 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
22 TX_CH6	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 6 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
21 TX_CH5	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 5 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
20 TX_CH4	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 4 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
19 TX_CH3	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 3 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
18 TX_CH2	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 2 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
17 TX_CH1	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 1 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
16 TX_CH0	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Tx Channel 0 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
15 RX_CH15	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 15 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
14 RX_CH14	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 14 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
13 RX_CH13	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 13 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
12 RX_CH12	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 12 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
11 RX_CH11	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 11 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
10 RX_CH10	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 10 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
9 RX_CH9	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 9 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
8 RX_CH8	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 8 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
7 RX_CH7	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 7 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
6 RX_CH6	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 6 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
5 RX_CH5	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 5 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
4 RX_CH4	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 4 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
3 RX_CH3	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 3 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
2 RX_CH2	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 2 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.

Table continues on the next page...

MIPI_HSI_CH_SFTRST field descriptions (continued)

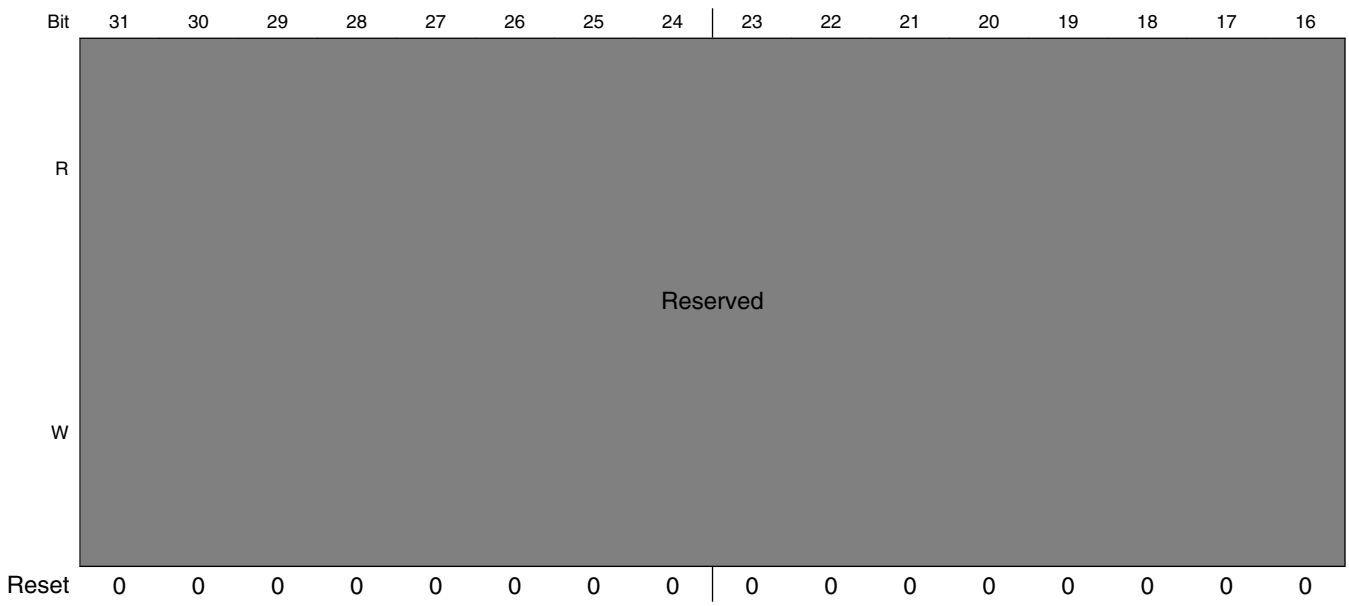
Field	Description
1 RX_CH1	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 1 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.
0 RX_CH0	Set this bit to zero to enable normal HSI operation. Set this bit to one to reset Rx Channel 0 (DMA and Fifo). When the reset operation complete, this bit will turn to zero automatically.

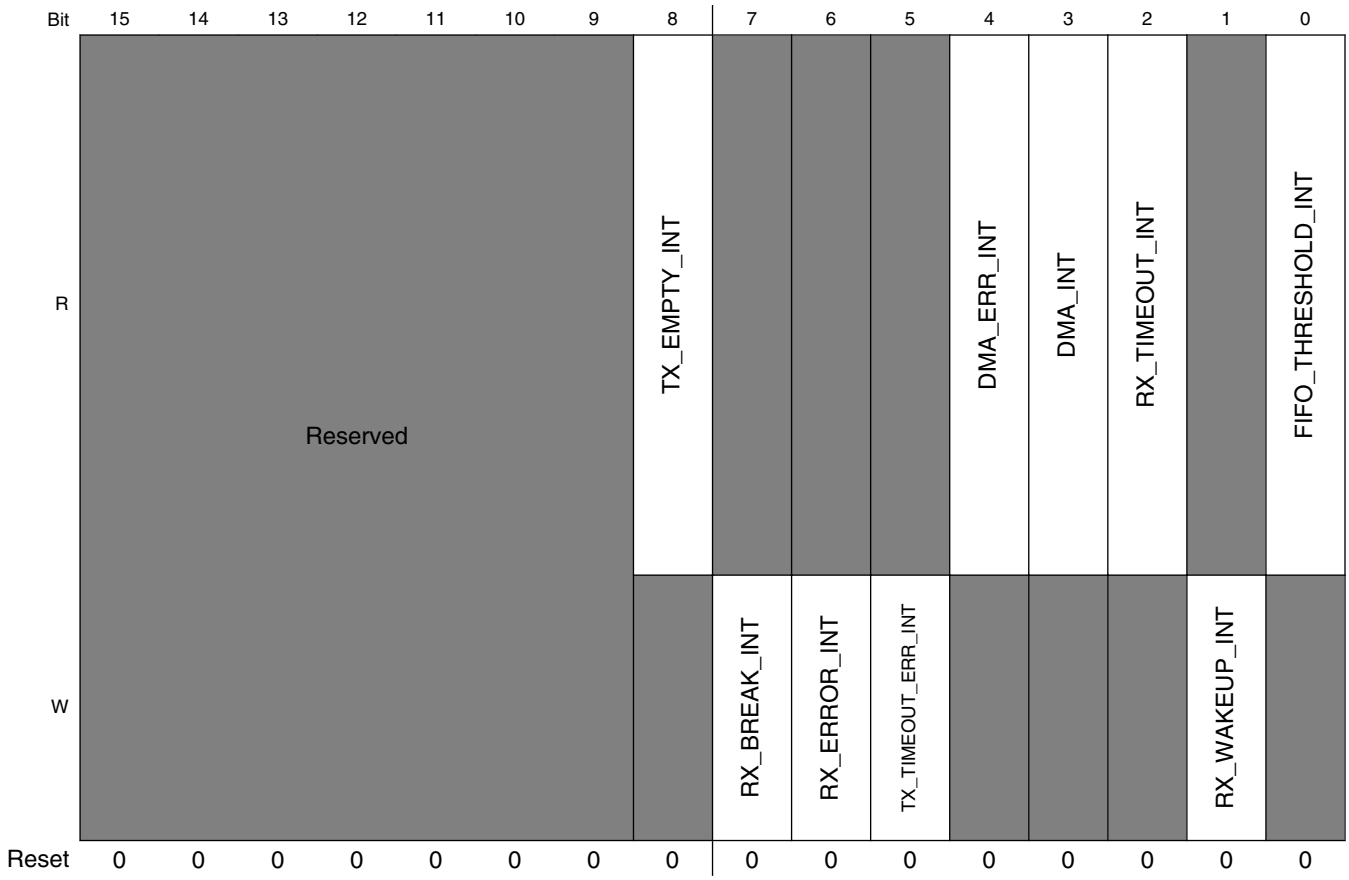
42.5.13 HSI Interrupt Status Register (MIPI_HSI_IRQSTAT)

This is HSI controller Interrupt Status Register.

This register contains the HSI controller Interrupt Status.

Address: 220_8000h base + 30h offset = 220_8030h





MIPI_HSI_IRQSTAT field descriptions

Field	Description
31–9 Reserved	This field is reserved. Reserved, always set to zero.
8 TX_EMPTY_INT	1 All tx channel empty and tx state IDLE Interrupt Status 0 not All tx channel empty and tx state IDLE Interrupt Status
7 RX_BREAK_INT	0 No Error. 1 Error.
6 RX_ERROR_INT	0 No Error. 1 Error.
5 TX_TIMEOUT_ERR_INT	0 No Error. 1 Error.
4 DMA_ERR_INT	If any bit in the DMA Error Interrupt Status Register is set, then this bit is set. on seeing this bit set, the ocp driver will read the Error Interrupt Staus Register. 0 No Error. 1 Error.
3 DMA_INT	This bit is set when a Transmit or Receive Operation is completed for DMA.

Table continues on the next page...

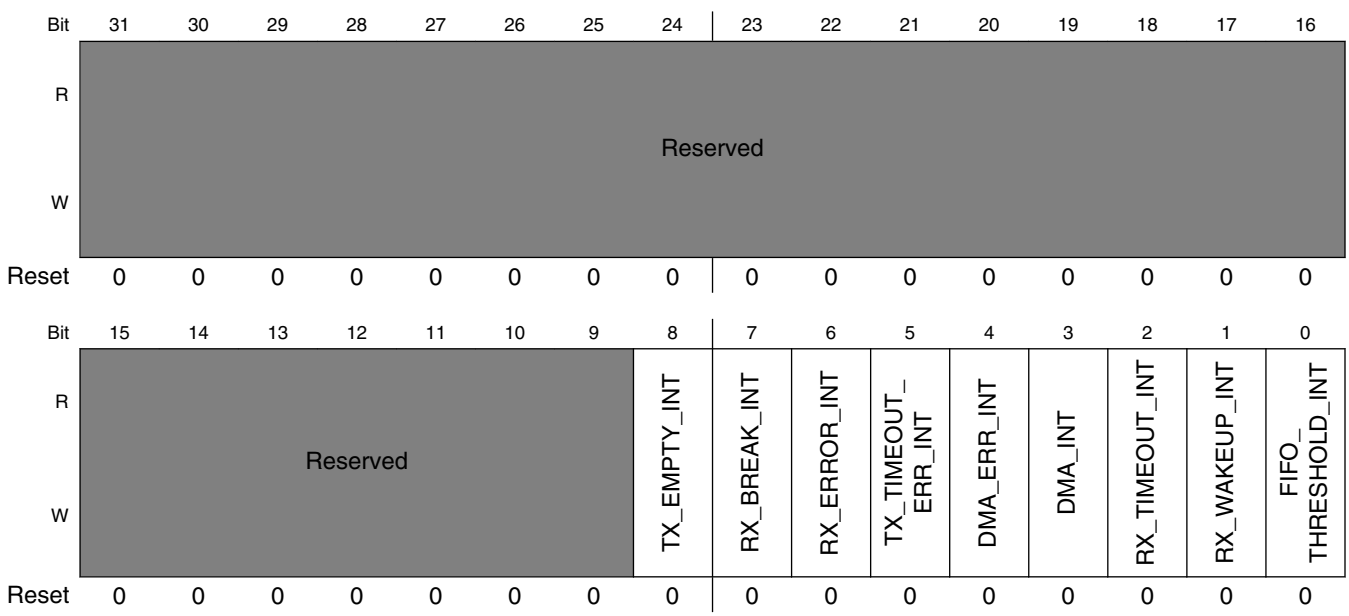
MIPI_HSI_IRQSTAT field descriptions (continued)

Field	Description
2 RX_TIMEOUT_INT	If any bit in the HSI Error Interrupt Status Register is set, then this bit is set. on seeing this bit set, the ocp driver will read the Error Interrupt Staus Register. 0 No Error. 1 Error.
1 RX_WAKEUP_INT	1 Receiver Wakeup event is occurred 0 Receiver Wakeup event is not occurred
0 FIFO_THRESHOLD_INT	1 Threshold amount of data reached in TX/Rx FIFO Interrupt Status 0 Threshold amount of data not reached in TX/Rx FIFO Interrupt Status

42.5.14 HSI Interrupt Status Enable Register (MIPI_HSI_IRQSTAT_EN)

This register contains the HSI controller Interrupt Status Enable.

Address: 220_8000h base + 34h offset = 220_8034h



MIPI_HSI_IRQSTAT_EN field descriptions

Field	Description
31–9 Reserved	This field is reserved. Reserved, always set to zero.

Table continues on the next page...

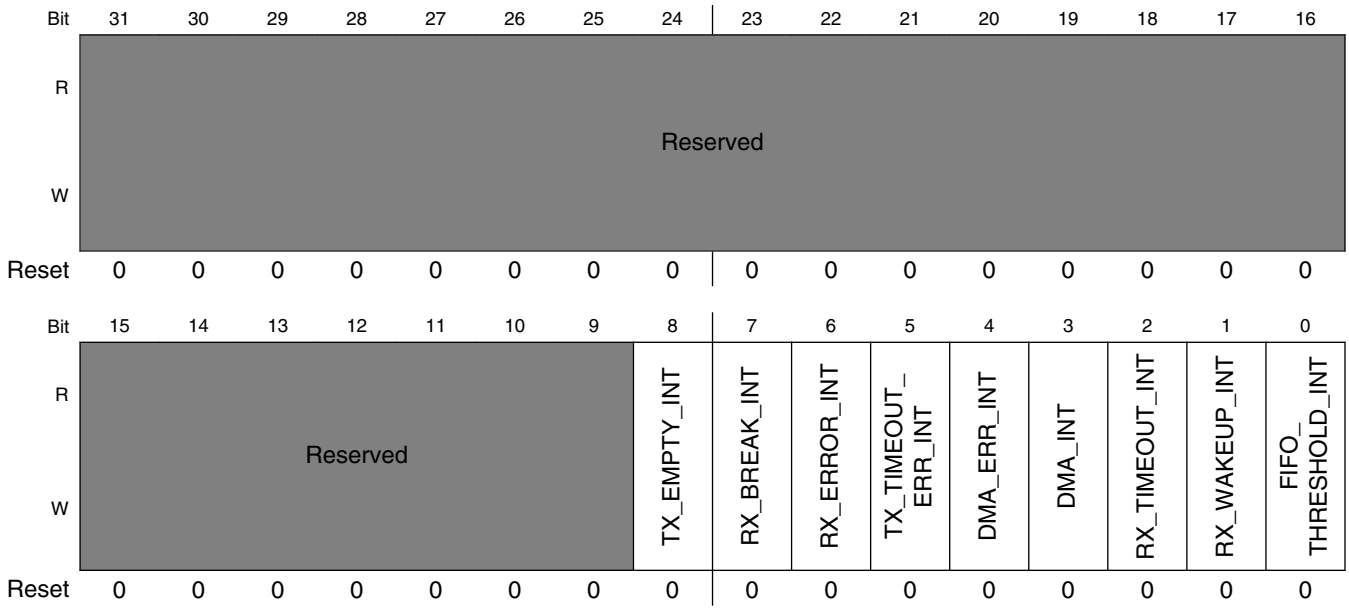
MIPI_HSI_IRQSTAT_EN field descriptions (continued)

Field	Description
8 TX_EMPTY_INT	1 Interrupt status enabled for TX_EMPTY_INT_STATUS interrupt. 0 Interrupt status masked TX_EMPTY_INT_STATUS interrupt.
7 RX_BREAK_INT	1 Interrupt status enabled for RX_BREAK status interrupt. 0 Interrupt status masked RX_BREAK status interrupt.
6 RX_ERROR_INT	1 Interrupt status enabled for RX_ERROR status interrupt. 0 Interrupt status masked RX_ERROR status interrupt.
5 TX_TIMEOUT_ERR_INT	1 Interrupt status enabled for TX_TIMEOUT_ERR status interrupt. 0 Interrupt status masked TX_TIMEOUT_ERR status interrupt.
4 DMA_ERR_INT	1 Interrupt status enabled for DMA_ERROR_INT_STATUS interrupt. 0 Interrupt status masked DMA_ERROR_INT_STATUS interrupt.
3 DMA_INT	1 Interrupt status enabled for DMA_INT_STATUS interrupt. 0 Interrupt status masked DMA_INT_STATUS interrupt.
2 RX_TIMEOUT_INT	1 Interrupt status enabled for RX_TIMEOUT_INT_STATUS interrupt. 0 Interrupt status masked RX_TIMEOUT_INT_STATUS interrupt.
1 RX_WAKEUP_INT	1 Interrupt status enabled for RX_WAKEUP_INT_STATUS interrupt. 0 Interrupt status masked RX_WAKEUP_INT_STATUS interrupt.
0 FIFO_THRESHOLD_INT	1 Interrupt status enabled for FIFO_THRESHOLD_INT_STATUS interrupt. 0 Interrupt status masked FIFO_THRESHOLD_INT_STATUS interrupt.

42.5.15 HSI Interrupt Signal Enable Register (MIPI_HSI_IRQSIG_EN)

This register contains the HSI controller Interrupt Signal Enable.

Address: 220_8000h base + 38h offset = 220_8038h



MIPI_HSI_IRQSIG_EN field descriptions

Field	Description
31–9 Reserved	This field is reserved. Reserved, always set to zero.
8 TX_EMPTY_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for HSI TX_EMPTY interrupt. 0 Interrupt signal masked for HSI TX_EMPTY interrupt.
7 RX_BREAK_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for RX_BREAK interrupt. 0 Interrupt signal masked for RX_BREAK interrupt.
6 RX_ERROR_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for RX Error interrupt. 0 Interrupt signal masked for RX Error interrupt.
5 TX_TIMEOUT_ERR_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for TX Timeout Error interrupt. 0 Interrupt signal masked for TX Timeout Error interrupt.

Table continues on the next page...

MIPI_HSI_IRQSIG_EN field descriptions (continued)

Field	Description
4 DMA_ERR_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for DMA Error interrupt. 0 Interrupt signal masked for DMA Error interrupt.
3 DMA_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for DMA Completed interrupt. 0 Interrupt signal masked for DMA Completed interrupt.
2 RX_TIMEOUT_ INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for RX TIMEOUT interrupt. 0 Interrupt signal masked for RX TIMEOUT interrupt.
1 RX_WAKEUP_ INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for HSI RX Wakeup interrupt. 0 Interrupt signal masked for HSI RX Wakeup interrupt.
0 FIFO_ THRESHOLD_ INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for HSI FIFO_THRESHOLD interrupt. 0 Interrupt signal masked for HSI FIFO_THRESHOLD interrupt.

42.5.16 HSI FIFO Threshold Interrupt Status Register (MIPI_HSI_FIFO_THR_IRQSTAT)

This register contains the HSI controller FIFO Threshold Interrupt Status.

Address: 220_8000h base + 3Ch offset = 220_803Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	TX_CH15_INT	TX_CH14_INT	TX_CH13_INT	TX_CH12_INT	TX_CH11_INT	TX_CH10_INT	TX_CH9_INT	TX_CH8_INT	TX_CH7_INT	TX_CH6_INT	TX_CH5_INT	TX_CH4_INT	TX_CH3_INT	TX_CH2_INT	TX_CH1_INT	TX_CH0_INT
W																
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RX_CH15_INT	RX_CH14_INT	RX_CH13_INT	RX_CH12_INT	RX_CH11_INT	RX_CH10_INT	RX_CH9_INT	RX_CH8_INT	RX_CH7_INT	RX_CH6_INT	RX_CH5_INT	RX_CH4_INT	RX_CH3_INT	RX_CH2_INT	RX_CH1_INT	RX_CH0_INT
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_FIFO_THR_IRQSTAT field descriptions

Field	Description
31 TX_CH15_INT	1 Threshold amount of data reached in Tx Channel 15 FIFO 0 Threshold amount of data not reached in Tx Channel 15 FIFO
30 TX_CH14_INT	1 Threshold amount of data reached in Tx Channel 14 FIFO 0 Threshold amount of data not reached in Tx Channel 14 FIFO
29 TX_CH13_INT	1 Threshold amount of data reached in Tx Channel 13 FIFO 0 Threshold amount of data not reached in Tx Channel 13 FIFO
28 TX_CH12_INT	1 Threshold amount of data reached in Tx Channel 12 FIFO 0 Threshold amount of data not reached in Tx Channel 12 FIFO
27 TX_CH11_INT	1 Threshold amount of data reached in Tx Channel 11 FIFO 0 Threshold amount of data not reached in Tx Channel 11 FIFO
26 TX_CH10_INT	1 Threshold amount of data reached in Tx Channel 10 FIFO 0 Threshold amount of data not reached in Tx Channel 10 FIFO
25 TX_CH9_INT	1 Threshold amount of data reached in Tx Channel 9 FIFO 0 Threshold amount of data not reached in Tx Channel 9 FIFO
24 TX_CH8_INT	1 Threshold amount of data reached in Tx Channel 8 FIFO 0 Threshold amount of data not reached in Tx Channel 8 FIFO
23 TX_CH7_INT	1 Threshold amount of data reached in Tx Channel 7 FIFO 0 Threshold amount of data not reached in Tx Channel 7 FIFO
22 TX_CH6_INT	1 Threshold amount of data reached in Tx Channel 6 FIFO 0 Threshold amount of data not reached in Tx Channel 6 FIFO
21 TX_CH5_INT	1 Threshold amount of data reached in Tx Channel 5 FIFO 0 Threshold amount of data not reached in Tx Channel 5 FIFO
20 TX_CH4_INT	1 Threshold amount of data reached in Tx Channel 4 FIFO 0 Threshold amount of data not reached in Tx Channel 4 FIFO
19 TX_CH3_INT	1 Threshold amount of data reached in Tx Channel 3 FIFO 0 Threshold amount of data not reached in Tx Channel 3 FIFO
18 TX_CH2_INT	1 Threshold amount of data reached in Tx Channel 2 FIFO 0 Threshold amount of data not reached in Tx Channel 2 FIFO
17 TX_CH1_INT	1 Threshold amount of data reached in Tx Channel 1 FIFO 0 Threshold amount of data not reached in Tx Channel 1 FIFO
16 TX_CH0_INT	1 Threshold amount of data reached in Tx Channel 0 FIFO 0 Threshold amount of data not reached in Tx Channel 0 FIFO
15 RX_CH15_INT	1 Threshold amount of data reached in Rx Channel 15 FIFO 0 Threshold amount of data not reached in Rx Channel 15 FIFO
14 RX_CH14_INT	1 Threshold amount of data reached in Rx Channel 14 FIFO 0 Threshold amount of data not reached in Rx Channel 14 FIFO
13 RX_CH13_INT	1 Threshold amount of data reached in Rx Channel 13 FIFO 0 Threshold amount of data not reached in Rx Channel 13 FIFO
12 RX_CH12_INT	1 Threshold amount of data reached in Rx Channel 12 FIFO 0 Threshold amount of data not reached in Rx Channel 12 FIFO
11 RX_CH11_INT	1 Threshold amount of data reached in Rx Channel 11 FIFO 0 Threshold amount of data not reached in Rx Channel 11 FIFO

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSTAT field descriptions (continued)

Field	Description
10 RX_CH10_INT	1 Threshold amount of data reached in Rx Channel 10 FIFO 0 Threshold amount of data not reached in Rx Channel 10 FIFO
9 RX_CH9_INT	1 Threshold amount of data reached in Rx Channel 9 FIFO 0 Threshold amount of data not reached in Rx Channel 9 FIFO
8 RX_CH8_INT	1 Threshold amount of data reached in Rx Channel 8 FIFO 0 Threshold amount of data not reached in Rx Channel 8 FIFO
7 RX_CH7_INT	1 Threshold amount of data reached in Rx Channel 7 FIFO 0 Threshold amount of data not reached in Rx Channel 7 FIFO
6 RX_CH6_INT	1 Threshold amount of data reached in Rx Channel 6 FIFO 0 Threshold amount of data not reached in Rx Channel 6 FIFO
5 RX_CH5_INT	1 Threshold amount of data reached in Rx Channel 5 FIFO 0 Threshold amount of data not reached in Rx Channel 5 FIFO
4 RX_CH4_INT	1 Threshold amount of data reached in Rx Channel 4 FIFO 0 Threshold amount of data not reached in Rx Channel 4 FIFO
3 RX_CH3_INT	1 Threshold amount of data reached in Rx Channel 3 FIFO 0 Threshold amount of data not reached in Rx Channel 3 FIFO
2 RX_CH2_INT	1 Threshold amount of data reached in Rx Channel 2 FIFO 0 Threshold amount of data not reached in Rx Channel 2 FIFO
1 RX_CH1_INT	1 Threshold amount of data reached in Rx Channel 1 FIFO 0 Threshold amount of data not reached in Rx Channel 1 FIFO
0 RX_CH0_INT	1 Threshold amount of data reached in Rx Channel 0 FIFO 0 Threshold amount of data not reached in Rx Channel 0 FIFO

42.5.17 HSI FIFO Threshold Interrupt Status Enable Register (MIPI_HSI_FIFO_THR_IRQSTAT_EN)

This register contains the HSI controller FIFO Threshold Interrupt Status Enable.

Address: 220_8000h base + 40h offset = 220_8040h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TX_CH15_INT	TX_CH14_INT	TX_CH13_INT	TX_CH12_INT	TX_CH11_INT	TX_CH10_INT	TX_CH9_INT	TX_CH8_INT	TX_CH7_INT	TX_CH6_INT	TX_CH5_INT	TX_CH4_INT	TX_CH3_INT	TX_CH2_INT	TX_CH1_INT	TX_CH0_INT
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RX_CH15_INT	RX_CH14_INT	RX_CH13_INT	RX_CH12_INT	RX_CH11_INT	RX_CH10_INT	RX_CH9_INT	RX_CH8_INT	RX_CH7_INT	RX_CH6_INT	RX_CH5_INT	RX_CH4_INT	RX_CH3_INT	RX_CH2_INT	RX_CH1_INT	RX_CH0_INT
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_FIFO_THR_IRQSTAT_EN field descriptions

Field	Description
31 TX_CH15_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch15 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch15 threshold Reached interrupt.
30 TX_CH14_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch14 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch14 threshold Reached interrupt.
29 TX_CH13_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch13 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch13 threshold Reached interrupt.
28 TX_CH12_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch12 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch12 threshold Reached interrupt.
27 TX_CH11_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch11 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch11 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSTAT_EN field descriptions (continued)

Field	Description
26 TX_CH10_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch10 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch10 threshold Reached interrupt.
25 TX_CH9_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch9 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch9 threshold Reached interrupt.
24 TX_CH8_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch8 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch8 threshold Reached interrupt.
23 TX_CH7_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch7 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch7 threshold Reached interrupt.
22 TX_CH6_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch6 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch6 threshold Reached interrupt.
21 TX_CH5_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch5 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch5 threshold Reached interrupt.
20 TX_CH4_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch4 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch4 threshold Reached interrupt.
19 TX_CH3_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch3 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch3 threshold Reached interrupt.
18 TX_CH2_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch2 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch2 threshold Reached interrupt.
17 TX_CH1_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch1 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch1 threshold Reached interrupt.
16 TX_CH0_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch0 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch0 threshold Reached interrupt.
15 RX_CH15_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch15 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch15 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSTAT_EN field descriptions (continued)

Field	Description
14 RX_CH14_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch14 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch14 threshold Reached interrupt.
13 RX_CH13_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch13 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch13 threshold Reached interrupt.
12 RX_CH12_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch12 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch12 threshold Reached interrupt.
11 RX_CH11_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch11 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch11 threshold Reached interrupt.
10 RX_CH10_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch10 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch10 threshold Reached interrupt.
9 RX_CH9_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch9 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch9 threshold Reached interrupt.
8 RX_CH8_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch8 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch8 threshold Reached interrupt.
7 RX_CH7_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch7 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch7 threshold Reached interrupt.
6 RX_CH6_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch6 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch6 threshold Reached interrupt.
5 RX_CH5_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch5 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch5 threshold Reached interrupt.
4 RX_CH4_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch4 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch4 threshold Reached interrupt.
3 RX_CH3_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch3 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch3 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSTAT_EN field descriptions (continued)

Field	Description
2 RX_CH2_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch2 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch2 threshold Reached interrupt.
1 RX_CH1_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch1 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch1 threshold Reached interrupt.
0 RX_CH0_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch0 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch0 threshold Reached interrupt.

42.5.18 HSI FIFO Threshold Interrupt Signal Enable Register (MIPI_HSI_FIFO_THR_IRQSIG_EN)

This register contains the HSI controller FIFO Threshold Interrupt Enable.

Address: 220_8000h base + 44h offset = 220_8044h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TX_CH15_INT	TX_CH14_INT	TX_CH13_INT	TX_CH12_INT	TX_CH11_INT	TX_CH10_INT	TX_CH9_INT	TX_CH8_INT	TX_CH7_INT	TX_CH6_INT	TX_CH5_INT	TX_CH4_INT	TX_CH3_INT	TX_CH2_INT	TX_CH1_INT	TX_CH0_INT
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RX_CH15_INT	RX_CH14_INT	RX_CH13_INT	RX_CH12_INT	RX_CH11_INT	RX_CH10_INT	RX_CH9_INT	RX_CH8_INT	RX_CH7_INT	RX_CH6_INT	RX_CH5_INT	RX_CH4_INT	RX_CH3_INT	RX_CH2_INT	RX_CH1_INT	RX_CH0_INT
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_FIFO_THR_IRQSIG_EN field descriptions

Field	Description
31 TX_CH15_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch15 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch15 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSIG_EN field descriptions (continued)

Field	Description
30 TX_CH14_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch14 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch14 threshold Reached interrupt.
29 TX_CH13_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch13 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch13 threshold Reached interrupt.
28 TX_CH12_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch12 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch12 threshold Reached interrupt.
27 TX_CH11_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch11 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch11 threshold Reached interrupt.
26 TX_CH10_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch10 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch10 threshold Reached interrupt.
25 TX_CH9_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch9 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch9 threshold Reached interrupt.
24 TX_CH8_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch8 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch8 threshold Reached interrupt.
23 TX_CH7_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch7 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch7 threshold Reached interrupt.
22 TX_CH6_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch6 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch6 threshold Reached interrupt.
21 TX_CH5_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch5 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch5 threshold Reached interrupt.
20 TX_CH4_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch4 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch4 threshold Reached interrupt.
19 TX_CH3_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch3 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch3 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSIG_EN field descriptions (continued)

Field	Description
18 TX_CH2_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch2 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch2 threshold Reached interrupt.
17 TX_CH1_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch1 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch1 threshold Reached interrupt.
16 TX_CH0_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Tx Ch0 threshold Reached interrupt. 0 Interrupt signal masked for Tx Ch0 threshold Reached interrupt.
15 RX_CH15_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch15 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch15 threshold Reached interrupt.
14 RX_CH14_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch14 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch14 threshold Reached interrupt.
13 RX_CH13_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch13 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch13 threshold Reached interrupt.
12 RX_CH12_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch12 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch12 threshold Reached interrupt.
11 RX_CH11_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch11 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch11 threshold Reached interrupt.
10 RX_CH10_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch10 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch10 threshold Reached interrupt.
9 RX_CH9_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch9 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch9 threshold Reached interrupt.
8 RX_CH8_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch8 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch8 threshold Reached interrupt.
7 RX_CH7_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch7 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch7 threshold Reached interrupt.

Table continues on the next page...

MIPI_HSI_FIFO_THR_IRQSIG_EN field descriptions (continued)

Field	Description
6 RX_CH6_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch6 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch6 threshold Reached interrupt.
5 RX_CH5_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch5 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch5 threshold Reached interrupt.
4 RX_CH4_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch4 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch4 threshold Reached interrupt.
3 RX_CH3_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch3 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch3 threshold Reached interrupt.
2 RX_CH2_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch2 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch2 threshold Reached interrupt.
1 RX_CH1_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch1 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch1 threshold Reached interrupt.
0 RX_CH0_INT	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for Rx Ch0 threshold Reached interrupt. 0 Interrupt signal masked for Rx Ch0 threshold Reached interrupt.

42.5.19 Tx Channel n Data Port Register (MIPI_HSI_TX_CH*n*_DP)

This Register is connected to fifo data port for Tx Channel n.

Address: 220_8000h base + 50h offset + (4d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TX_CH*n*_DP field descriptions

Field	Description
DATA	Software could Write/Read this bits to access Tx Channel n.

42.5.20 Rx Channel n Data Port Register (MIPI_HSI_RX_CH*n*_DP)

This Register is connected to fifo data port for Rx Channel n.

Address: 220_8000h base + 90h offset + (4d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

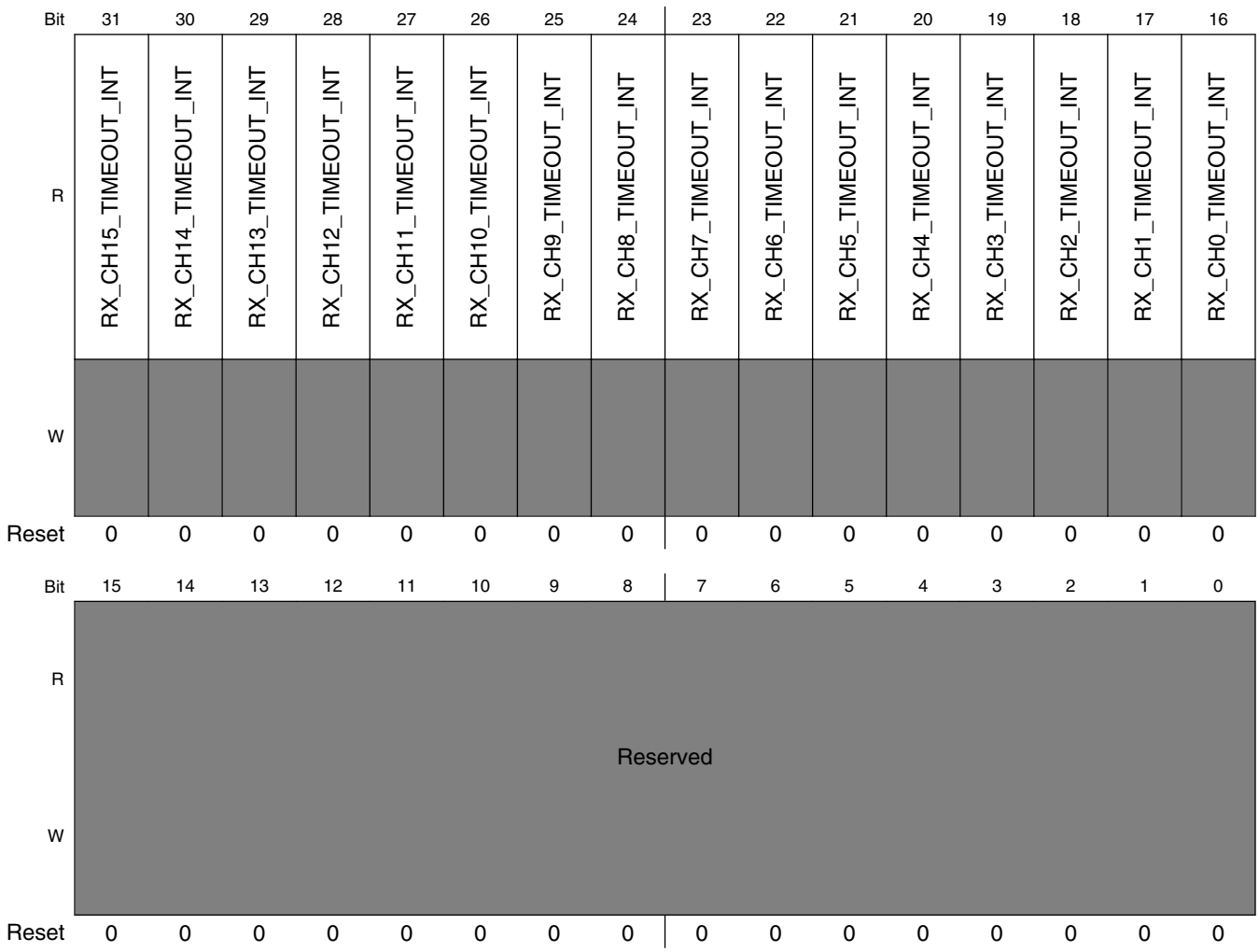
MIPI_HSI_RX_CH*n*_DP field descriptions

Field	Description
DATA	Software could Write/Read this bits to access Rx Channel n.

42.5.21 HSI Error Interrupt Status Register (MIPI_HSI_ERR_IRQSTAT)

This register contains the HSI controller Error Interrupt Status.

Address: 220_8000h base + D0h offset = 220_80D0h



MIPI_HSI_ERR_IRQSTAT field descriptions

Field	Description
31 RX_CH15_ TIMEOUT_INT	This status bit is set when data timeout counter for ch15 reaches thedata timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch15 buffer and then read HSI Status register to find the further status of the Rx ch15 Buffer. The host driver has to read the Rx ch15 fifo on Dword basis, till the fifo is completely empty.

Table continues on the next page...

MIPI_HSI_ERR_IRQSTAT field descriptions (continued)

Field	Description
30 RX_CH14_ TIMEOUT_INT	This status bit is set when data timeout counter for ch14 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch14 buffer and then read HSI Status register to find the further status of the Rx ch14 Buffer. The host driver has to read the Rx ch14 fifo on Dword basis, till the fifo is completely empty.
29 RX_CH13_ TIMEOUT_INT	This status bit is set when data timeout counter for ch13 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch13 buffer and then read HSI Status register to find the further status of the Rx ch13 Buffer. The host driver has to read the Rx ch13 fifo on Dword basis, till thefifo is completely empty.
28 RX_CH12_ TIMEOUT_INT	This status bit is set when data timeout counter for ch12 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch12 buffer and then read HSI Status register to find the further status of the Rx ch12 Buffer. The host driver has to read the Rx ch12 fifo on Dword basis, till the fifo is completely empty.
27 RX_CH11_ TIMEOUT_INT	This status bit is set when data timeout counter for ch11 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch11 buffer and then read HSI Status register to find the further status of the Rx ch11 Buffer. The host driver has to read the Rx ch11 fifo on Dword basis, till the fifo is completely empty.
26 RX_CH10_ TIMEOUT_INT	This status bit is set when data timeout counter for ch10 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch10 buffer and then read HSI Status register to find the further status of the Rx ch10 Buffer. The host driver has to read the Rx ch10 fifo on Dword basis, till the fifo is completely empty.
25 RX_CH9_ TIMEOUT_INT	This status bit is set when data timeout counter for ch9 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch9 buffer and then read HSI Status register to find the further status of the Rx ch9 Buffer. The host driver has to read the Rx ch9 fifo on Dword basis, till the fifo is completely empty.
24 RX_CH8_ TIMEOUT_INT	This status bit is set when data timeout counter for ch8 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch8 buffer and then read HSI Status register to find the further status of the Rx ch8 Buffer. The host driver has to read the Rx ch8 fifo on Dword basis, till the fifo is completely empty.
23 RX_CH7_ TIMEOUT_INT	This status bit is set when data timeout counter for ch7 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch7 buffer and then read HSI Status register to find the further status of the Rx ch7 Buffer. The host driver has to read the Rx ch7 fifo on Dword basis, till the fifo is completely empty.
22 RX_CH6_ TIMEOUT_INT	This status bit is set when data timeout counter for ch6 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch6 buffer and then read HSI Status register to find the further status of the Rx ch6 Buffer. The host driver has to read the Rx ch6 fifo on Dword basis, till the fifo is completely empty.
21 RX_CH5_ TIMEOUT_INT	This status bit is set when data timeout counter for ch5 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch5 buffer and then read HSI Status register to find the further status of the Rx ch5 Buffer. The host driver has to read the Rx ch5 fifo on Dword basis, till the fifo is completely empty.
20 RX_CH4_ TIMEOUT_INT	This status bit is set when data timeout counter for ch4 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch4 buffer and then read HSI Status register to find the further status of the Rx ch4 Buffer. The host driver has to read the Rx ch4 fifo on Dword basis, till the fifo is completely empty.

Table continues on the next page...

MIPI_HSI_ERR_IRQSTAT field descriptions (continued)

Field	Description
19 RX_CH3_ TIMEOUT_INT	This status bit is set when data timeout counter for ch3 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch3 buffer and then read HSI Status register to find the further status of the Rx ch3 Buffer. The host driver has to read the Rx ch3 fifo on Dword basis, till the fifo is completely empty.
18 RX_CH2_ TIMEOUT_INT	This status bit is set when data timeout counter for ch2 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch2 buffer and then read HSI Status register to find the further status of the Rx ch2 Buffer. The host driver has to read the Rx ch2 fifo on Dword basis, till the fifo is completely empty.
17 RX_CH1_ TIMEOUT_INT	This status bit is set when data timeout counter for ch1 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch1 buffer and then read HSI Status register to find the further status of the Rx ch1 Buffer. The host driver has to read the Rx ch1 fifo on Dword basis, till the fifo is completely empty.
16 RX_CH0_ TIMEOUT_INT	This status bit is set when data timeout counter for ch0 reaches the data timeout counter value. On receiving the interrupt the host driver should read 1Dword of data from Rx ch0 buffer and then read HSI Status register to find the further status of the Rx ch0 Buffer. The host driver has to read the Rx ch0 fifo on Dword basis, till the fifo is completely empty.
Reserved	This field is reserved. Reserved, always set to zero.

42.5.22 HSI Error Interrupt Status Enable Register (MIPI_HSI_ERR_IRQSTAT_EN)

This register contains the HSI controller Error Interrupt Status Enable.

Address: 220_8000h base + D4h offset = 220_80D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	RX_CH15_TIMEOUT_INT_EN	RX_CH14_TIMEOUT_INT_EN	RX_CH13_TIMEOUT_INT_EN	RX_CH12_TIMEOUT_INT_EN	RX_CH11_TIMEOUT_INT_EN	RX_CH10_TIMEOUT_INT_EN	RX_CH9_TIMEOUT_INT_EN	RX_CH8_TIMEOUT_INT_EN	RX_CH7_TIMEOUT_INT_EN	RX_CH6_TIMEOUT_INT_EN	RX_CH5_TIMEOUT_INT_EN	RX_CH4_TIMEOUT_INT_EN	RX_CH3_TIMEOUT_INT_EN	RX_CH2_TIMEOUT_INT_EN	RX_CH1_TIMEOUT_INT_EN	RX_CH0_TIMEOUT_INT_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_ERR_IRQSTAT_EN field descriptions

Field	Description
31 RX_CH15_TIMEOUT_INT_EN	1 Interrupt status enabled for data timeout for ch15 interrupt. 0 Interrupt status masked for data timeout for ch15 interrupt.
30 RX_CH14_TIMEOUT_INT_EN	1 Interrupt status enabled for data timeout for ch14 interrupt. 0 Interrupt status masked for data timeout for ch14 interrupt.
29 RX_CH13_TIMEOUT_INT_EN	1 Interrupt status enabled for data timeout for ch13 interrupt. 0 Interrupt status masked for data timeout for ch13 interrupt.
28 RX_CH12_TIMEOUT_INT_EN	1 Interrupt status enabled for data timeout for ch12 interrupt. 0 Interrupt status masked for data timeout for ch12 interrupt.

Table continues on the next page...

MIPI_HSI_ERR_IRQSTAT_EN field descriptions (continued)

Field	Description
27 RX_CH11_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch11 interrupt. 0 Interrupt status masked for data timeout for ch11 interrupt.
26 RX_CH10_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch10 interrupt. 0 Interrupt status masked for data timeout for ch10 interrupt.
25 RX_CH9_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch9 interrupt. 0 Interrupt status masked for data timeout for ch9 interrupt.
24 RX_CH8_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch8 interrupt. 0 Interrupt status masked for data timeout for ch8 interrupt.
23 RX_CH7_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch7 interrupt. 0 Interrupt status masked for data timeout for ch7 interrupt.
22 RX_CH6_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch6 interrupt. 0 Interrupt status masked for data timeout for ch6 interrupt.
21 RX_CH5_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch5 interrupt. 0 Interrupt status masked for data timeout for ch5 interrupt.
20 RX_CH4_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch4 interrupt. 0 Interrupt status masked for data timeout for ch4 interrupt.
19 RX_CH3_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch3 interrupt. 0 Interrupt status masked for data timeout for ch3 interrupt.
18 RX_CH2_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch2 interrupt. 0 Interrupt status masked for data timeout for ch2 interrupt.
17 RX_CH1_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch1 interrupt. 0 Interrupt status masked for data timeout for ch1 interrupt.
16 RX_CH0_ TIMEOUT_INT_ EN	1 Interrupt status enabled for data timeout for ch0 interrupt. 0 Interrupt status masked for data timeout for ch0 interrupt.
Reserved	This field is reserved. Reserved, always set to zero.

42.5.23 HSI Error Interrupt Signal Enable Register (MIPI_HSI_ERR_IRQSIG_EN)

This register contains the HSI controller Error Interrupt Signal Enable.

Address: 220_8000h base + D8h offset = 220_80D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	RX_CH15_TIMEOUT_INT_EN	RX_CH14_TIMEOUT_INT_EN	RX_CH13_TIMEOUT_INT_EN	RX_CH12_TIMEOUT_INT_EN	RX_CH11_TIMEOUT_INT_EN	RX_CH10_TIMEOUT_INT_EN	RX_CH9_TIMEOUT_INT_EN	RX_CH8_TIMEOUT_INT_EN	RX_CH7_TIMEOUT_INT_EN	RX_CH6_TIMEOUT_INT_EN	RX_CH5_TIMEOUT_INT_EN	RX_CH4_TIMEOUT_INT_EN	RX_CH3_TIMEOUT_INT_EN	RX_CH2_TIMEOUT_INT_EN	RX_CH1_TIMEOUT_INT_EN	RX_CH0_TIMEOUT_INT_EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_ERR_IRQSIG_EN field descriptions

Field	Description
31 RX_CH15_TIMEOUT_INT_EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch15 interrupt. 0 Interrupt signal masked for data timeout for ch15 interrupt.
30 RX_CH14_TIMEOUT_INT_EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch14 interrupt. 0 Interrupt signal masked for data timeout for ch14 interrupt.
29 RX_CH13_TIMEOUT_INT_EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch13 interrupt. 0 Interrupt signal masked for data timeout for ch13 interrupt.
28 RX_CH12_TIMEOUT_INT_EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch12 interrupt. 0 Interrupt signal masked for data timeout for ch12 interrupt.

Table continues on the next page...

MIPI_HSI_ERR_IRQSIG_EN field descriptions (continued)

Field	Description
27 RX_CH11_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch11 interrupt. 0 Interrupt signal masked for data timeout for ch11 interrupt.
26 RX_CH10_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch10 interrupt. 0 Interrupt signal masked for data timeout for ch10 interrupt.
25 RX_CH9_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch9 interrupt. 0 Interrupt signal masked for data timeout for ch9 interrupt.
24 RX_CH8_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch8 interrupt. 0 Interrupt signal masked for data timeout for ch8 interrupt.
23 RX_CH7_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch7 interrupt. 0 Interrupt signal masked for data timeout for ch7 interrupt.
22 RX_CH6_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch6 interrupt. 0 Interrupt signal masked for data timeout for ch6 interrupt.
21 RX_CH5_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch5 interrupt. 0 Interrupt signal masked for data timeout for ch5 interrupt.
20 RX_CH4_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch4 interrupt. 0 Interrupt signal masked for data timeout for ch4 interrupt.
19 RX_CH3_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch3 interrupt. 0 Interrupt signal masked for data timeout for ch3 interrupt.
18 RX_CH2_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch2 interrupt. 0 Interrupt signal masked for data timeout for ch2 interrupt.
17 RX_CH1_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch1 interrupt. 0 Interrupt signal masked for data timeout for ch1interrupt.
16 RX_CH0_ TIMEOUT_INT_ EN	Setting this bit will enable interrupt generation on interrupt line. 1 Interrupt signal enabled for data timeout for ch0 interrupt. 0 Interrupt signal masked for data timeout for ch0 interrupt.
Reserved	This field is reserved.

Table continues on the next page...

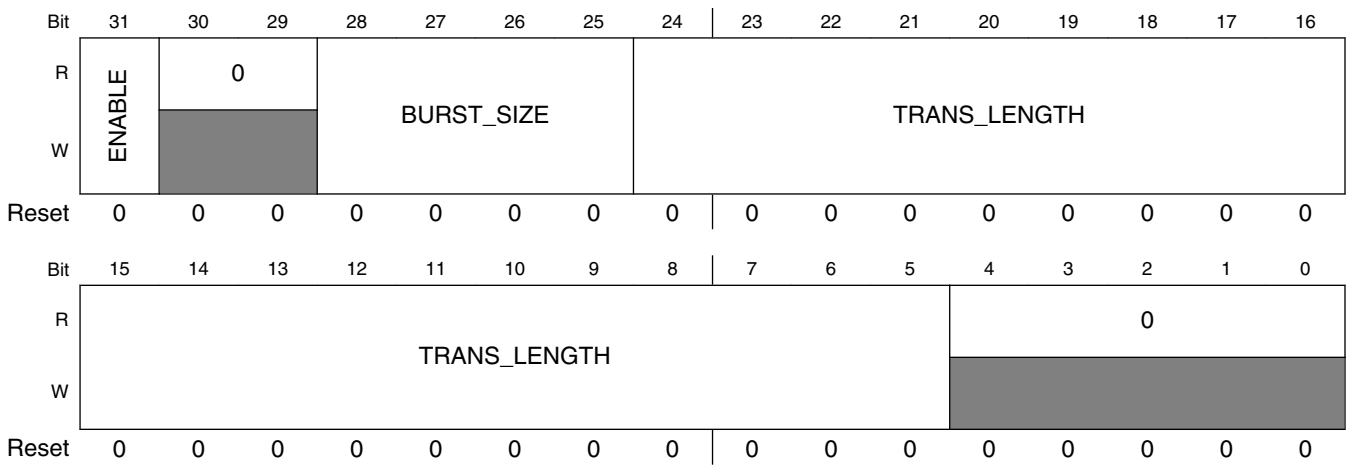
MIPI_HSI_ERR_IRQSIG_EN field descriptions (continued)

Field	Description
	Reserved, always set to zero.

42.5.24 Tx DMA Channel n Configuration Register (MIPI_HSI_TDMA*n*_CONF)

This register contains the configurations of enable/disable, burst size and transfer count for Tx DMA channel n.

Address: 220_8000h base + DCh offset + (4d × i), where i=0d to 15d



MIPI_HSI_TDMA*n*_CONF field descriptions

Field	Description
31 ENABLE	Setting this bit enables the inernal Tx DMA channel n.
30–29 Reserved	This read-only field is reserved and always has the value 0.
28–25 BURST_SIZE	Burst size for Tx DMA channel n. The unit is Dword. The burst size should not be larger than relevant TRANS_LENGTH and FIFO_SIZE. h0 1Dword to transfer for each burst h1 2Dword to transfer for each burst h2 4Dword to transfer for each burst h10 1024Dword to transfer for each burst
24–5 TRANS_ LENGTH	Transfer data length for Tx DMA channel n. The unit is Dword. h1 1Dword to transfer h2 2Dwords to transfer

Table continues on the next page...

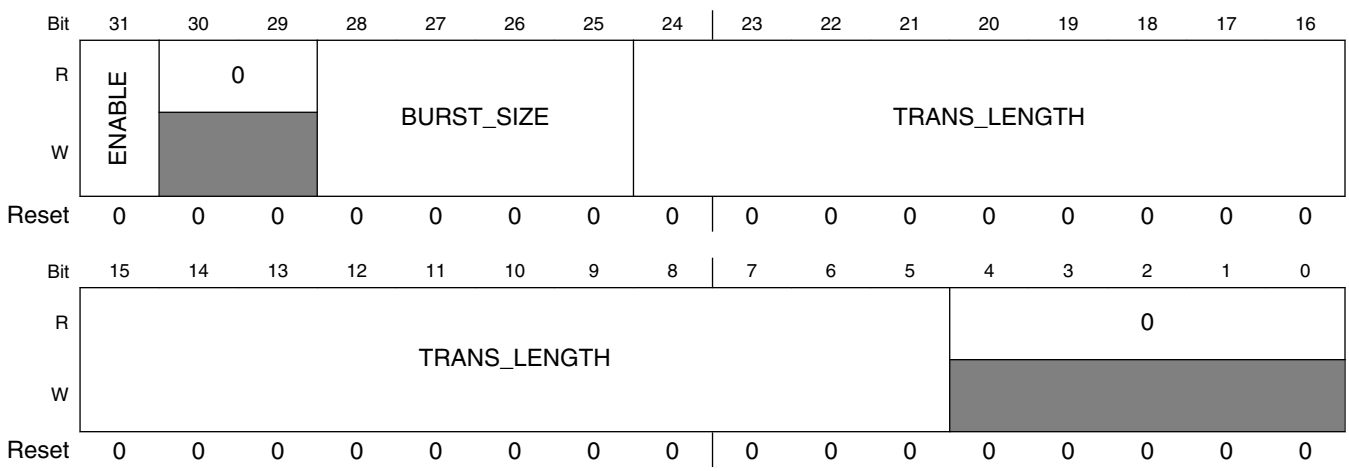
MIPI_HSI_TDMA*n*_CONF field descriptions (continued)

Field	Description
	hffffff 1048575Dwords to transfer
Reserved	This read-only field is reserved and always has the value 0.

42.5.25 Rx DMA Channel n Configuration Register (MIPI_HSI_RDMA*n*_CONF)

This register contains the configurations of enable/disable, burst size and transfer count for Rx DMA channel n.

Address: 220_8000h base + 11Ch offset + (4d × i), where i=0d to 15d



MIPI_HSI_RDMA*n*_CONF field descriptions

Field	Description
31 ENABLE	Setting this bit enables the inernal Rx DMA channel n.
30–29 Reserved	This read-only field is reserved and always has the value 0.
28–25 BURST_SIZE	Burst size for Rx DMA channel n. The unit is Dword. The burst size should not be larger than relevant TRANS_LENGTH and FIFO_SIZE. h0 1Dword to transfer for each burst h1 2Dword to transfer for each burst h2 4Dword to transfer for each burst h10 1024Dword to transfer for each burst
24–5 TRANS_ LENGTH	Transfer data length for Rx DMA channel 0. The unit is Dword. h1 1Dword to transfer

Table continues on the next page...

MIPI_HSI_RDMA*n*_CONF field descriptions (continued)

Field	Description
	h2 2Dwords to transfer hffffff 1048575Dwords to transfer
Reserved	This read-only field is reserved and always has the value 0.

42.5.26 Tx DMA Channel n Start Address Register (MIPI_HSI_TDMA*n*_STA_ADDR)

This Register containts the physical Start Address HSI for Tx DMA Channel n.

Address: 220_8000h base + 15Ch offset + (4d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DS_ADDR																														0	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_TDMA*n*_STA_ADDR field descriptions

Field	Description
31–2 DS_ADDR	The Physical Start Address for Tx DMA Channel n. DWord aligned
Reserved	This read-only field is reserved and always has the value 0.

42.5.27 Rx DMA Channel n Start Address Register (MIPI_HSI_RDMA*n*_STA_ADDR)

This Register containts the physical Start Address HSI for Rx DMA Channel n.

Address: 220_8000h base + 19Ch offset + (4d × i), where i=0d to 15d

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DS_ADDR																														0	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_RDMA*n*_STA_ADDR field descriptions

Field	Description
31–2 DS_ADDR	The Physical Start Address for Rx DMA Channel n. DWord aligned
Reserved	This read-only field is reserved and always has the value 0.

42.5.28 DMA Interrupt Status Register (MIPI_HSI_DMA_IRQSTAT)

This register contains all the interrupt status for HSI internal DMA

Address: 220_8000h base + 1DCh offset = 220_81DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_IRQSTAT field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 interrupt status
30 TDMA14	TDMA Channel 14 interrupt status
29 TDMA13	TDMA Channel 13 interrupt status
28 TDMA12	TDMA Channel 12 interrupt status
27 TDMA11	TDMA Channel 11 interrupt status
26 TDMA10	TDMA Channel 10 interrupt status
25 TDMA9	TDMA Channel 9 interrupt status
24 TDMA8	TDMA Channel 8 interrupt status
23 TDMA7	TDMA Channel 7 interrupt status
22 TDMA6	TDMA Channel 6 interrupt status
21 TDMA5	TDMA Channel 5 interrupt status
20 TDMA4	TDMA Channel 4 interrupt status
19 TDMA3	TDMA Channel 3 interrupt status
18 TDMA2	TDMA Channel 2 interrupt status
17 TDMA1	TDMA Channel 1 interrupt status
16 TDMA0	TDMA Channel 0 interrupt status
15 RDMA15	RDMA Channel 15 interrupt status
14 RDMA14	RDMA Channel 14 interrupt status
13 RDMA13	RDMA Channel 13 interrupt status
12 RDMA12	RDMA Channel 12 interrupt status
11 RDMA11	RDMA Channel 11 interrupt status
10 RDMA10	RDMA Channel 10 interrupt status
9 RDMA9	RDMA Channel 9 interrupt status

Table continues on the next page...

MIPI_HSI_DMA_IRQSTAT field descriptions (continued)

Field	Description
8 RDMA8	RDMA Channel 8 interrupt status
7 RDMA7	RDMA Channel 7 interrupt status
6 RDMA6	RDMA Channel 6 interrupt status
5 RDMA5	RDMA Channel 5 interrupt status
4 RDMA4	RDMA Channel 4 interrupt status
3 RDMA3	RDMA Channel 3 interrupt status
2 RDMA2	RDMA Channel 2 interrupt status
1 RDMA1	RDMA Channel 1 interrupt status
0 RDMA0	RDMA Channel 0 interrupt status

42.5.29 DMA Interrupt Enable Register (MIPI_HSI_DMA_IRQSTAT_EN)

This Register is used to select which DMA interrupt could send to HIS Interrupt Status Register

Address: 220_8000h base + 1E0h offset = 220_81E0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_IRQSTAT_EN field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 interrupt Enable
30 TDMA14	TDMA Channel 14 interrupt Enable
29 TDMA13	TDMA Channel 13 interrupt Enable
28 TDMA12	TDMA Channel 12 interrupt Enable
27 TDMA11	TDMA Channel 11 interrupt Enable
26 TDMA10	TDMA Channel 10 interrupt Enable
25 TDMA9	TDMA Channel 9 interrupt Enable
24 TDMA8	TDMA Channel 8 interrupt Enable
23 TDMA7	TDMA Channel 7 interrupt Enable
22 TDMA6	TDMA Channel 6 interrupt Enable
21 TDMA5	TDMA Channel 5 interrupt Enable
20 TDMA4	TDMA Channel 4 interrupt Enable
19 TDMA3	TDMA Channel 3 interrupt Enable
18 TDMA2	TDMA Channel 2 interrupt Enable
17 TDMA1	TDMA Channel 1 interrupt Enable
16 TDMA0	TDMA Channel 0 interrupt Enable
15 RDMA15	RDMA Channel 15 interrupt Enable
14 RDMA14	RDMA Channel 14 interrupt Enable
13 RDMA13	RDMA Channel 13 interrupt Enable
12 RDMA12	RDMA Channel 12 interrupt Enable
11 RDMA11	RDMA Channel 11 interrupt Enable
10 RDMA10	RDMA Channel 10 interrupt Enable
9 RDMA9	RDMA Channel 9 interrupt Enable

Table continues on the next page...

MIPI_HSI_DMA_IRQSTAT_EN field descriptions (continued)

Field	Description
8 RDMA8	RDMA Channel 8 interrupt Enable
7 RDMA7	RDMA Channel 7 interrupt Enable
6 RDMA6	RDMA Channel 6 interrupt Enable
5 RDMA5	RDMA Channel 5 interrupt Enable
4 RDMA4	RDMA Channel 4 interrupt Enable
3 RDMA3	RDMA Channel 3 interrupt Enable
2 RDMA2	RDMA Channel 2 interrupt Enable
1 RDMA1	RDMA Channel 1 interrupt Enable
0 RDMA0	RDMA Channel 0 interrupt Enable

42.5.30 DMA Interrupt Status Signal Enable Register (MIPI_HSI_DMA_IRQSIG_EN)

This Register is used to select which DMA interrupt status could send to HIS Interrupt Status Register

Address: 220_8000h base + 1E4h offset = 220_81E4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_IRQSIG_EN field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 interrupt status enable
30 TDMA14	TDMA Channel 14 interrupt status enable
29 TDMA13	TDMA Channel 13 interrupt status enable
28 TDMA12	TDMA Channel 12 interrupt status enable
27 TDMA11	TDMA Channel 11 interrupt status enable
26 TDMA10	TDMA Channel 10 interrupt status enable
25 TDMA9	TDMA Channel 9 interrupt status enable
24 TDMA8	TDMA Channel 8 interrupt status enable
23 TDMA7	TDMA Channel 7 interrupt status enable
22 TDMA6	TDMA Channel 6 interrupt status enable
21 TDMA5	TDMA Channel 5 interrupt status enable
20 TDMA4	TDMA Channel 4 interrupt status enable
19 TDMA3	TDMA Channel 3 interrupt status enable
18 TDMA2	TDMA Channel 2 interrupt status enable
17 TDMA1	TDMA Channel 1 interrupt status enable
16 TDMA0	TDMA Channel 0 interrupt status enable
15 RDMA15	RDMA Channel 15 interrupt status enable
14 RDMA14	RDMA Channel 14 interrupt status enable
13 RDMA13	RDMA Channel 13 interrupt status enable
12 RDMA12	RDMA Channel 12 interrupt status enable
11 RDMA11	RDMA Channel 11 interrupt status enable
10 RDMA10	RDMA Channel 10 interrupt status enable
9 RDMA9	RDMA Channel 9 interrupt status enable

Table continues on the next page...

MIPI_HSI_DMA_IRQSIG_EN field descriptions (continued)

Field	Description
8 RDMA8	RDMA Channel 8 interrupt status enable
7 RDMA7	RDMA Channel 7 interrupt status enable
6 RDMA6	RDMA Channel 6 interrupt status enable
5 RDMA5	RDMA Channel 5 interrupt status enable
4 RDMA4	RDMA Channel 4 interrupt status enable
3 RDMA3	RDMA Channel 3 interrupt status enable
2 RDMA2	RDMA Channel 2 interrupt status enable
1 RDMA1	RDMA Channel 1 interrupt status enable
0 RDMA0	RDMA Channel 0 interrupt status enable

42.5.31 DMA Error Interrupt Status Register (MIPI_HSI_DMA_ERR_IRQSTAT)

This register contains all the error interrupt status for HSI internal DMA

Address: 220_8000h base + 1E8h offset = 220_81E8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_ERR_IRQSTAT field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 error interrupt status
30 TDMA14	TDMA Channel 14 error interrupt status

Table continues on the next page...

MIPI_HSI_DMA_ERR_IRQSTAT field descriptions (continued)

Field	Description
29 TDMA13	TDMA Channel 13 error interrupt status
28 TDMA12	TDMA Channel 12 error interrupt status
27 TDMA11	TDMA Channel 11 error interrupt status
26 TDMA10	TDMA Channel 10 error interrupt status
25 TDMA9	TDMA Channel 9 error interrupt status
24 TDMA8	TDMA Channel 8 error interrupt status
23 TDMA7	TDMA Channel 7 error interrupt status
22 TDMA6	TDMA Channel 6 error interrupt status
21 TDMA5	TDMA Channel 5 error interrupt status
20 TDMA4	TDMA Channel 4 error interrupt status
19 TDMA3	TDMA Channel 3 error interrupt status
18 TDMA2	TDMA Channel 2 error interrupt status
17 TDMA1	TDMA Channel 1 error interrupt status
16 TDMA0	TDMA Channel 0 error interrupt status
15 RDMA15	RDMA Channel 15 error interrupt status
14 RDMA14	RDMA Channel 14 error interrupt status
13 RDMA13	RDMA Channel 13 error interrupt status
12 RDMA12	RDMA Channel 12 error interrupt status
11 RDMA11	RDMA Channel 11 error interrupt status
10 RDMA10	RDMA Channel 10 error interrupt status
9 RDMA9	RDMA Channel 9 error interrupt status
8 RDMA8	RDMA Channel 8 error interrupt status
7 RDMA7	RDMA Channel 7 error interrupt status

Table continues on the next page...

MIPI_HSI_DMA_ERR_IRQSTAT field descriptions (continued)

Field	Description
6 RDMA6	RDMA Channel 6 error interrupt status
5 RDMA5	RDMA Channel 5 error interrupt status
4 RDMA4	RDMA Channel 4 error interrupt status
3 RDMA3	RDMA Channel 3 error interrupt status
2 RDMA2	RDMA Channel 2 error interrupt status
1 RDMA1	RDMA Channel 1 error interrupt status
0 RDMA0	RDMA Channel 0 error interrupt status

42.5.32 DMA Error Interrupt Enable Register (MIPI_HSI_DMA_ERR_IRQSTAT_EN)

This register is used to select which DMA error interrupt could send to HIS Interrupt Status Register

Address: 220_8000h base + 1ECh offset = 220_81ECh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_ERR_IRQSTAT_EN field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 error interrupt enable

Table continues on the next page...

MIPI_HSI_DMA_ERR_IRQSTAT_EN field descriptions (continued)

Field	Description
30 TDMA14	TDMA Channel 14 error interrupt enable
29 TDMA13	TDMA Channel 13 error interrupt enable
28 TDMA12	TDMA Channel 12 error interrupt enable
27 TDMA11	TDMA Channel 11 error interrupt enable
26 TDMA10	TDMA Channel 10 error interrupt enable
25 TDMA9	TDMA Channel 9 error interrupt enable
24 TDMA8	TDMA Channel 8 error interrupt enable
23 TDMA7	TDMA Channel 7 error interrupt enable
22 TDMA6	TDMA Channel 6 error interrupt enable
21 TDMA5	TDMA Channel 5 error interrupt enable
20 TDMA4	TDMA Channel 4 error interrupt enable
19 TDMA3	TDMA Channel 3 error interrupt enable
18 TDMA2	TDMA Channel 2 error interrupt enable
17 TDMA1	TDMA Channel 1 error interrupt enable
16 TDMA0	TDMA Channel 0 error interrupt enable
15 RDMA15	RDMA Channel 15 error interrupt enable
14 RDMA14	RDMA Channel 14 error interrupt enable
13 RDMA13	RDMA Channel 13 error interrupt enable
12 RDMA12	RDMA Channel 12 error interrupt enable
11 RDMA11	RDMA Channel 11 error interrupt enable
10 RDMA10	RDMA Channel 10 error interrupt enable
9 RDMA9	RDMA Channel 9 error interrupt enable
8 RDMA8	RDMA Channel 8 error interrupt enable

Table continues on the next page...

MIPI_HSI_DMA_ERR_IRQSTAT_EN field descriptions (continued)

Field	Description
7 RDMA7	RDMA Channel 7 error interrupt enable
6 RDMA6	RDMA Channel 6 error interrupt enable
5 RDMA5	RDMA Channel 5 error interrupt enable
4 RDMA4	RDMA Channel 4 error interrupt enable
3 RDMA3	RDMA Channel 3 error interrupt enable
2 RDMA2	RDMA Channel 2 error interrupt enable
1 RDMA1	RDMA Channel 1 error interrupt enable
0 RDMA0	RDMA Channel 0 error interrupt enable

42.5.33 DMA Error Interrupt Signal Enable Register (MIPI_HSI_DMA_ERR_IRQSIG_EN)

This Register is used to select which DMA error interrupt status could send to HIS Interrupt Status Register

Address: 220_8000h base + 1F0h offset = 220_81F0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_ERR_IRQSIG_EN field descriptions

Field	Description
31 TDMA15	TDMA Channel 15 error interrupt status enable
30 TDMA14	TDMA Channel 14 error interrupt status enable
29 TDMA13	TDMA Channel 13 error interrupt status enable
28 TDMA12	TDMA Channel 12 error interrupt status enable
27 TDMA11	TDMA Channel 11 error interrupt status enable
26 TDMA10	TDMA Channel 10 error interrupt status enable
25 TDMA9	TDMA Channel 9 error interrupt status enable
24 TDMA8	TDMA Channel 8 error interrupt status enable
23 TDMA7	TDMA Channel 7 error interrupt status enable
22 TDMA6	TDMA Channel 6 error interrupt status enable
21 TDMA5	TDMA Channel 5 error interrupt status enable
20 TDMA4	TDMA Channel 4 error interrupt status enable
19 TDMA3	TDMA Channel 3 error interrupt status enable
18 TDMA2	TDMA Channel 2 error interrupt status enable
17 TDMA1	TDMA Channel 1 error interrupt status enable
16 TDMA0	TDMA Channel 0 error interrupt status enable
15 RDMA15	RDMA Channel 15 error interrupt status enable
14 RDMA14	RDMA Channel 14 error interrupt status enable
13 RDMA13	RDMA Channel 13 error interrupt status enable
12 RDMA12	RDMA Channel 12 error interrupt status enable
11 RDMA11	RDMA Channel 11 error interrupt status enable
10 RDMA10	RDMA Channel 10 error interrupt status enable
9 RDMA9	RDMA Channel 9 error interrupt status enable

Table continues on the next page...

MIPI_HSI_DMA_ERR_IRQSIG_EN field descriptions (continued)

Field	Description
8 RDMA8	RDMA Channel 8 error interrupt status enable
7 RDMA7	RDMA Channel 7 error interrupt status enable
6 RDMA6	RDMA Channel 6 error interrupt status enable
5 RDMA5	RDMA Channel 5 error interrupt status enable
4 RDMA4	RDMA Channel 4 error interrupt status enable
3 RDMA3	RDMA Channel 3 error interrupt status enable
2 RDMA2	RDMA Channel 2 error interrupt status enable
1 RDMA1	RDMA Channel 1 error interrupt status enable
0 RDMA0	RDMA Channel 0 error interrupt status enable

42.5.34 DMA Single Request Enable Register (MIPI_HSI_DMA_SINGLE_REQ_EN)

This Register is used to debug

Address: 220_8000h base + 1F4h offset = 220_81F4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	TDMA15	TDMA14	TDMA13	TDMA12	TDMA11	TDMA10	TDMA9	TDMA8	TDMA7	TDMA6	TDMA5	TDMA4	TDMA3	TDMA2	TDMA1	TDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	RDMA15	RDMA14	RDMA13	RDMA12	RDMA11	RDMA10	RDMA9	RDMA8	RDMA7	RDMA6	RDMA5	RDMA4	RDMA3	RDMA2	RDMA1	RDMA0
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MIPI_HSI_DMA_SINGLE_REQ_EN field descriptions

Field	Description
31 TDMA15	When the remain DMA data less than one DMA burst size in Tx Dma Channle 15, this bit will be set automatically
30 TDMA14	When the remain DMA data less than one DMA burst size in Tx Dma Channle 14, this bit will be set automatically

Table continues on the next page...

MIPI_HSI_DMA_SINGLE_REQ_EN field descriptions (continued)

Field	Description
29 TDMA13	When the remain DMA data less than one DMA burst size in Tx Dma Channle 13, this bit will be set automatically
28 TDMA12	When the remain DMA data less than one DMA burst size in Tx Dma Channle 12, this bit will be set automatically
27 TDMA11	When the remain DMA data less than one DMA burst size in Tx Dma Channle 11, this bit will be set automatically
26 TDMA10	When the remain DMA data less than one DMA burst size in Tx Dma Channle 10, this bit will be set automatically
25 TDMA9	When the remain DMA data less than one DMA burst size in Tx Dma Channle 9, this bit will be set automatically
24 TDMA8	When the remain DMA data less than one DMA burst size in Tx Dma Channle 8, this bit will be set automatically
23 TDMA7	When the remain DMA data less than one DMA burst size in Tx Dma Channle 7, this bit will be set automatically
22 TDMA6	When the remain DMA data less than one DMA burst size in Tx Dma Channle 6, this bit will be set automatically
21 TDMA5	When the remain DMA data less than one DMA burst size in Tx Dma Channle 5, this bit will be set automatically
20 TDMA4	When the remain DMA data less than one DMA burst size in Tx Dma Channle 4, this bit will be set automatically
19 TDMA3	When the remain DMA data less than one DMA burst size in Tx Dma Channle 3, this bit will be set automatically
18 TDMA2	When the remain DMA data less than one DMA burst size in Tx Dma Channle 2, this bit will be set automatically
17 TDMA1	When the remain DMA data less than one DMA burst size in Tx Dma Channle 1, this bit will be set automatically
16 TDMA0	When the remain DMA data less than one DMA burst size in Tx Dma Channle 0, this bit will be set automatically
15 RDMA15	When the remain DMA data less than one DMA burst size in Rx Dma Channle 15, this bit will be set automatically
14 RDMA14	When the remain DMA data less than one DMA burst size in Rx Dma Channle 14, this bit will be set automatically
13 RDMA13	When the remain DMA data less than one DMA burst size in Rx Dma Channle 13, this bit will be set automatically
12 RDMA12	When the remain DMA data less than one DMA burst size in Rx Dma Channle 12, this bit will be set automatically
11 RDMA11	When the remain DMA data less than one DMA burst size in Rx Dma Channle 11, this bit will be set automatically
10 RDMA10	When the remain DMA data less than one DMA burst size in Rx Dma Channle 10, this bit will be set automatically
9 RDMA9	When the remain DMA data less than one DMA burst size in Rx Dma Channle 9, this bit will be set automatically
8 RDMA8	When the remain DMA data less than one DMA burst size in Rx Dma Channle 8, this bit will be set automatically
7 RDMA7	When the remain DMA data less than one DMA burst size in Rx Dma Channle 7, this bit will be set automatically

Table continues on the next page...

MIPI_HSI_DMA_SINGLE_REQ_EN field descriptions (continued)

Field	Description
6 RDMA6	When the remain DMA data less than one DMA burst size in Rx Dma Channle 6, this bit will be set automatically
5 RDMA5	When the remain DMA data less than one DMA burst size in Rx Dma Channle 5, this bit will be set automatically
4 RDMA4	When the remain DMA data less than one DMA burst size in Rx Dma Channle 4, this bit will be set automatically
3 RDMA3	When the remain DMA data less than one DMA burst size in Rx Dma Channle 3, this bit will be set automatically
2 RDMA2	When the remain DMA data less than one DMA burst size in Rx Dma Channle 2, this bit will be set automatically
1 RDMA1	When the remain DMA data less than one DMA burst size in Rx Dma Channle 1, this bit will be set automatically
0 RDMA0	When the remain DMA data less than one DMA burst size in Rx Dma Channle 0, this bit will be set automatically

42.5.35 Tx Fifo Size Configuration Register 0 (MIPI_HSI_TX_FIFO_SIZE_CONF0)

This register is used to config each Tx fifo size

Address: 220_8000h base + 200h offset = 220_8200h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH15				CH14				CH13				CH12				CH11				CH10				CH9				CH8			
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_TX_FIFO_SIZE_CONF0 field descriptions

Field	Description
31–28 CH15	This field is used to set the buffer size for channel 15. All the allowed combinations of bit setting are listed here . 0000 channel 15 buffer size is 1Dword 0001 channel 15 buffer size is 2Dwords 0010 channel 15 buffer size is 4Dwords 0011 channel 15 buffer size is 8Dwords 0100 channel 15 buffer size is 16Dwords 0101 channel 15 buffer size is 32Dwords 0110 channel 15 buffer size is 64Dwords 0111 channel 15 buffer size is 128Dwords 1000 channel 15 buffer size is 256Dwords 1001 channel 15 buffer size is 512Dwords

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
	1010 channel 15 buffer size is 1024Dwords 1111-1011 Reserved
27–24 CH14	This field is used to set the buffer size for channel 14. All the allowed combinations of bit setting are listed here . 0000 channel 14 buffer size is 1Dword 0001 channel 14 buffer size is 2Dwords 0010 channel 14 buffer size is 4Dwords 0011 channel 14 buffer size is 8Dwords 0100 channel 14 buffer size is 16Dwords 0101 channel 14 buffer size is 32Dwords 0110 channel 14 buffer size is 64Dwords 0111 channel 14 buffer size is 128Dwords 1000 channel 14 buffer size is 256Dwords 1001 channel 14 buffer size is 512Dwords 1010 channel 14 buffer size is 1024Dwords 1111-1011 Reserved
23–20 CH13	This field is used to set the buffer size for channel 13. All the allowed combinations of bit setting are listed here . 0000 channel 13 buffer size is 1Dword 0001 channel 13 buffer size is 2Dwords 0010 channel 13 buffer size is 4Dwords 0011 channel 13 buffer size is 8Dwords 0100 channel 13 buffer size is 16Dwords 0101 channel 13 buffer size is 32Dwords 0110 channel 13 buffer size is 64Dwords 0111 channel 13 buffer size is 128Dwords 1000 channel 13 buffer size is 256Dwords 1001 channel 13 buffer size is 512Dwords 1010 channel 13 buffer size is 1024Dwords 1111-1011 Reserved
19–16 CH12	This field is used to set the buffer size for channel 12. All the allowed combinations of bit setting are listed here . 0000 channel 12 buffer size is 1Dword 0001 channel 12 buffer size is 2Dwords 0010 channel 12 buffer size is 4Dwords 0011 channel 12 buffer size is 8Dwords 0100 channel 12 buffer size is 16Dwords 0101 channel 12 buffer size is 32Dwords 0110 channel 12 buffer size is 64Dwords 0111 channel 12 buffer size is 128Dwords 1000 channel 12 buffer size is 256Dwords 1001 channel 12 buffer size is 512Dwords 1010 channel 12 buffer size is 1024Dwords 1111-1011 Reserved

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
15–12 CH11	This field is used to set the buffer size for channel 11. All the allowed combinations of bit setting are listed here . 0000 channel 11 buffer size is 1Dword 0001 channel 11 buffer size is 2Dwords 0010 channel 11 buffer size is 4Dwords 0011 channel 11 buffer size is 8Dwords 0100 channel 11 buffer size is 16Dwords 0101 channel 11 buffer size is 32Dwords 0110 channel 11 buffer size is 64Dwords 0111 channel 11 buffer size is 128Dwords 1000 channel 11 buffer size is 256Dwords 1001 channel 11 buffer size is 512Dwords 1010 channel 11 buffer size is 1024Dwords 1111-1011 Reserved
11–8 CH10	This field is used to set the buffer size for channel 10. All the allowed combinations of bit setting are listed here . 0000 channel 10 buffer size is 1Dword 0001 channel 10 buffer size is 2Dwords 0010 channel 10 buffer size is 4Dwords 0011 channel 10 buffer size is 8Dwords 0100 channel 10 buffer size is 16Dwords 0101 channel 10 buffer size is 32Dwords 0110 channel 10 buffer size is 64Dwords 0111 channel 10 buffer size is 128Dwords 1000 channel 10 buffer size is 256Dwords 1001 channel 10 buffer size is 512Dwords 1010 channel 10 buffer size is 1024Dwords 1111-1011 Reserved
7–4 CH9	This field is used to set the buffer size for channel 9. All the allowed combinations of bit setting are listed here . 0000 channel 9 buffer size is 1Dword 0001 channel 9 buffer size is 2Dwords 0010 channel 9 buffer size is 4Dwords 0011 channel 9 buffer size is 8Dwords 0100 channel 9 buffer size is 16Dwords 0101 channel 9 buffer size is 32Dwords 0110 channel 9 buffer size is 64Dwords 0111 channel 9 buffer size is 128Dwords 1000 channel 9 buffer size is 256Dwords 1001 channel 9 buffer size is 512Dwords 1010 channel 9 buffer size is 1024Dwords 1111-1011 Reserved
CH8	This field is used to set the buffer size for channel 8.

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
	All the allowed combinations of bit setting are listed here . 0000 channel 8 buffer size is 1Dword 0001 channel 8 buffer size is 2Dwords 0010 channel 8 buffer size is 4Dwords 0011 channel 8 buffer size is 8Dwords 0100 channel 8 buffer size is 16Dwords 0101 channel 8 buffer size is 32Dwords 0110 channel 8 buffer size is 64Dwords 0111 channel 8 buffer size is 128Dwords 1000 channel 8 buffer size is 256Dwords 1001 channel 8 buffer size is 512Dwords 1010 channel 8 buffer size is 1024Dwords 1111-1011 Reserved

42.5.36 Tx Fifo Size Configuration Register 1 (MIPI_HSI_TX_FIFO_SIZE_CONF1)

This register is used to config each Tx fifo size

Address: 220_8000h base + 204h offset = 220_8204h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH7				CH6				CH5				CH4				CH3				CH2				CH1				CH0			
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_TX_FIFO_SIZE_CONF1 field descriptions

Field	Description
31–28 CH7	This field is used to set the buffer size for channel 7. All the allowed combinations of bit setting are listed here . 0000 channel 7 buffer size is 1Dword 0001 channel 7 buffer size is 2Dwords 0010 channel 7 buffer size is 4Dwords 0011 channel 7 buffer size is 8Dwords 0100 channel 7 buffer size is 16Dwords 0101 channel 7 buffer size is 32Dwords 0110 channel 7 buffer size is 64Dwords 0111 channel 7 buffer size is 128Dwords 1000 channel 7 buffer size is 256Dwords 1001 channel 7 buffer size is 512Dwords

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
	1010 channel 7 buffer size is 1024Dwords 1111-1011 Reserved
27–24 CH6	This field is used to set the buffer size for channel 6. All the allowed combinations of bit setting are listed here . 0000 channel 6 buffer size is 1Dword 0001 channel 6 buffer size is 2Dwords 0010 channel 6 buffer size is 4Dwords 0011 channel 6 buffer size is 8Dwords 0100 channel 6 buffer size is 16Dwords 0101 channel 6 buffer size is 32Dwords 0110 channel 6 buffer size is 64Dwords 0111 channel 6 buffer size is 128Dwords 1000 channel 6 buffer size is 256Dwords 1001 channel 6 buffer size is 512Dwords 1010 channel 6 buffer size is 1024Dwords 1111-1011 Reserved
23–20 CH5	This field is used to set the buffer size for channel 5. All the allowed combinations of bit setting are listed here . 0000 channel 5 buffer size is 1Dword 0001 channel 5 buffer size is 2Dwords 0010 channel 5 buffer size is 4Dwords 0011 channel 5 buffer size is 8Dwords 0100 channel 5 buffer size is 16Dwords 0101 channel 5 buffer size is 32Dwords 0110 channel 5 buffer size is 64Dwords 0111 channel 5 buffer size is 128Dwords 1000 channel 5 buffer size is 256Dwords 1001 channel 5 buffer size is 512Dwords 1010 channel 5 buffer size is 1024Dwords 1111-1011 Reserved
19–16 CH4	This field is used to set the buffer size for channel 4. All the allowed combinations of bit setting are listed here . 0000 channel 4 buffer size is 1Dword 0001 channel 4 buffer size is 2Dwords 0010 channel 4 buffer size is 4Dwords 0011 channel 4 buffer size is 8Dwords 0100 channel 4 buffer size is 16Dwords 0101 channel 4 buffer size is 32Dwords 0110 channel 4 buffer size is 64Dwords 0111 channel 4 buffer size is 128Dwords 1000 channel 4 buffer size is 256Dwords 1001 channel 4 buffer size is 512Dwords 1010 channel 4 buffer size is 1024Dwords 1111-1011 Reserved

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
15–12 CH3	This field is used to set the buffer size for channel 3. All the allowed combinations of bit setting are listed here . 0000 channel 3 buffer size is 1Dword 0001 channel 3 buffer size is 2Dwords 0010 channel 3 buffer size is 4Dwords 0011 channel 3 buffer size is 8Dwords 0100 channel 3 buffer size is 16Dwords 0101 channel 3 buffer size is 32Dwords 0110 channel 3 buffer size is 64Dwords 0111 channel 3 buffer size is 128Dwords 1000 channel 3 buffer size is 256Dwords 1001 channel 3 buffer size is 512Dwords 1010 channel 3 buffer size is 1024Dwords 1111-1011 Reserved
11–8 CH2	This field is used to set the buffer size for channel 2. All the allowed combinations of bit setting are listed here . 0000 channel 2 buffer size is 1Dword 0001 channel 2 buffer size is 2Dwords 0010 channel 2 buffer size is 4Dwords 0011 channel 2 buffer size is 8Dwords 0100 channel 2 buffer size is 16Dwords 0101 channel 2 buffer size is 32Dwords 0110 channel 2 buffer size is 64Dwords 0111 channel 2 buffer size is 128Dwords 1000 channel 2 buffer size is 256Dwords 1001 channel 2 buffer size is 512Dwords 1010 channel 2 buffer size is 1024Dwords 1111-1011 Reserved
7–4 CH1	This field is used to set the buffer size for channel 1. All the allowed combinations of bit setting are listed here . 0000 channel 1 buffer size is 1Dword 0001 channel 1 buffer size is 2Dwords 0010 channel 1 buffer size is 4Dwords 0011 channel 1 buffer size is 8Dwords 0100 channel 1 buffer size is 16Dwords 0101 channel 1 buffer size is 32Dwords 0110 channel 1 buffer size is 64Dwords 0111 channel 1 buffer size is 128Dwords 1000 channel 1 buffer size is 256Dwords 1001 channel 1 buffer size is 512Dwords 1010 channel 1 buffer size is 1024Dwords 1111-1011 Reserved
CH0	This field is used to set the buffer size for channel 0.

Table continues on the next page...

MIPI_HSI_TX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
	All the allowed combinations of bit setting are listed here . 0000 channel 0 buffer size is 1Dword 0001 channel 0 buffer size is 2Dwords 0010 channel 0 buffer size is 4Dwords 0011 channel 0 buffer size is 8Dwords 0100 channel 0 buffer size is 16Dwords 0101 channel 0 buffer size is 32Dwords 0110 channel 0 buffer size is 64Dwords 0111 channel 0 buffer size is 128Dwords 1000 channel 0 buffer size is 256Dwords 1001 channel 0 buffer size is 512Dwords 1010 channel 0 buffer size is 1024Dwords 1111-1011 Reserved

42.5.37 Rx Fifo Size Configuration Register 0 (MIPI_HSI_RX_FIFO_SIZE_CONF0)

This register is used to config each Rx fifo size

Address: 220_8000h base + 208h offset = 220_8208h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH15				CH14				CH13				CH12				CH11				CH10				CH9				CH8			
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_RX_FIFO_SIZE_CONF0 field descriptions

Field	Description
31–28 CH15	This field is used to set the buffer size for channel 15. All the allowed combinations of bit setting are listed here . 0000 channel 15 buffer size is 1Dword 0001 channel 15 buffer size is 2Dwords 0010 channel 15 buffer size is 4Dwords 0011 channel 15 buffer size is 8Dwords 0100 channel 15 buffer size is 16Dwords 0101 channel 15 buffer size is 32Dwords 0110 channel 15 buffer size is 64Dwords 0111 channel 15 buffer size is 128Dwords 1000 channel 15 buffer size is 256Dwords 1001 channel 15 buffer size is 512Dwords

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
	1010 channel 15 buffer size is 1024Dwords 1111-1011 Reserved
27–24 CH14	This field is used to set the buffer size for channel 14. All the allowed combinations of bit setting are listed here . 0000 channel 14 buffer size is 1Dword 0001 channel 14 buffer size is 2Dwords 0010 channel 14 buffer size is 4Dwords 0011 channel 14 buffer size is 8Dwords 0100 channel 14 buffer size is 16Dwords 0101 channel 14 buffer size is 32Dwords 0110 channel 14 buffer size is 64Dwords 0111 channel 14 buffer size is 128Dwords 1000 channel 14 buffer size is 256Dwords 1001 channel 14 buffer size is 512Dwords 1010 channel 14 buffer size is 1024Dwords 1111-1011 Reserved
23–20 CH13	This field is used to set the buffer size for channel 13. All the allowed combinations of bit setting are listed here . 0000 channel 13 buffer size is 1Dword 0001 channel 13 buffer size is 2Dwords 0010 channel 13 buffer size is 4Dwords 0011 channel 13 buffer size is 8Dwords 0100 channel 13 buffer size is 16Dwords 0101 channel 13 buffer size is 32Dwords 0110 channel 13 buffer size is 64Dwords 0111 channel 13 buffer size is 128Dwords 1000 channel 13 buffer size is 256Dwords 1001 channel 13 buffer size is 512Dwords 1010 channel 13 buffer size is 1024Dwords 1111 b1011 Reserved
19–16 CH12	This field is used to set the buffer size for channel 12. All the allowed combinations of bit setting are listed here . 0000 channel 12 buffer size is 1Dword 0001 channel 12 buffer size is 2Dwords 0010 channel 12 buffer size is 4Dwords 0011 channel 12 buffer size is 8Dwords 0100 channel 12 buffer size is 16Dwords 0101 channel 12 buffer size is 32Dwords 0110 channel 12 buffer size is 64Dwords 0111 channel 12 buffer size is 128Dwords 1000 channel 12 buffer size is 256Dwords 1001 channel 12 buffer size is 512Dwords 1010 channel 12 buffer size is 1024Dwords 1111-1011 Reserved

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
15–12 CH11	This field is used to set the buffer size for channel 11. All the allowed combinations of bit setting are listed here . 0000 channel 11 buffer size is 1Dword 0001 channel 11 buffer size is 2Dwords 0010 channel 11 buffer size is 4Dwords 0011 channel 11 buffer size is 8Dwords 0100 channel 11 buffer size is 16Dwords 0101 channel 11 buffer size is 32Dwords 0110 channel 11 buffer size is 64Dwords 0111 channel 11 buffer size is 128Dwords 1000 channel 11 buffer size is 256Dwords 1001 channel 11 buffer size is 512Dwords 1010 channel 11 buffer size is 1024Dwords 1111-1011 Reserved
11–8 CH10	This field is used to set the buffer size for channel 10. All the allowed combinations of bit setting are listed here . 0000 channel 10 buffer size is 1Dword 0001 channel 10 buffer size is 2Dwords 0010 channel 10 buffer size is 4Dwords 0011 channel 10 buffer size is 8Dwords 0100 channel 10 buffer size is 16Dwords 0101 channel 10 buffer size is 32Dwords 0110 channel 10 buffer size is 64Dwords 0111 channel 10 buffer size is 128Dwords 1000 channel 10 buffer size is 256Dwords 1001 channel 10 buffer size is 512Dwords 1010 channel 10 buffer size is 1024Dwords 1111-1011 Reserved
7–4 CH9	This field is used to set the buffer size for channel 9. All the allowed combinations of bit setting are listed here . 0000 channel 9 buffer size is 1Dword 0001 channel 9 buffer size is 2Dwords 0010 channel 9 buffer size is 4Dwords 0011 channel 9 buffer size is 8Dwords 0100 channel 9 buffer size is 16Dwords 0101 channel 9 buffer size is 32Dwords 0110 channel 9 buffer size is 64Dwords 0111 channel 9 buffer size is 128Dwords 1000 channel 9 buffer size is 256Dwords 1001 channel 9 buffer size is 512Dwords 1010 channel 9 buffer size is 1024Dwords 1111-1011 Reserved
CH8	This field is used to set the buffer size for channel 8.

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF0 field descriptions (continued)

Field	Description
	All the allowed combinations of bit setting are listed here . 0000 channel 8 buffer size is 1Dword 0001 channel 8 buffer size is 2Dwords 0010 channel 8 buffer size is 4Dwords 0011 channel 8 buffer size is 8Dwords 0100 channel 8 buffer size is 16Dwords 0101 channel 8 buffer size is 32Dwords 0110 channel 8 buffer size is 64Dwords 0111 channel 8 buffer size is 128Dwords 1000 channel 8 buffer size is 256Dwords 1001 channel 8 buffer size is 512Dwords 1010 channel 8 buffer size is 1024Dwords 1111-1011 Reserved

42.5.38 Rx Fifo Size Configuration Register 1 (MIPI_HSI_RX_FIFO_SIZE_CONF1)

This register is used to config each Rx fifo size

Address: 220_8000h base + 20Ch offset = 220_820Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CH7				CH6				CH5				CH4				CH3				CH2				CH1				CH0			
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_RX_FIFO_SIZE_CONF1 field descriptions

Field	Description
31–28 CH7	This field is used to set the buffer size for channel 7. All the allowed combinations of bit setting are listed here . 0000 channel 7 buffer size is 1Dword 0001 channel 7 buffer size is 2Dwords 0010 channel 7 buffer size is 4Dwords 0011 channel 7 buffer size is 8Dwords 0100 channel 7 buffer size is 16Dwords 0101 channel 7 buffer size is 32Dwords 0110 channel 7 buffer size is 64Dwords 0111 channel 7 buffer size is 128Dwords 1000 channel 7 buffer size is 256Dwords 1001 channel 7 buffer size is 512Dwords

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
	1010 channel 7 buffer size is 1024Dwords 1111-1011 Reserved
27–24 CH6	This field is used to set the buffer size for channel 6. All the allowed combinations of bit setting are listed here . 0000 channel 6 buffer size is 1Dword 0001 channel 6 buffer size is 2Dwords 0010 channel 6 buffer size is 4Dwords 0011 channel 6 buffer size is 8Dwords 0100 channel 6 buffer size is 16Dwords 0101 channel 6 buffer size is 32Dwords 0110 channel 6 buffer size is 64Dwords 0111 channel 6 buffer size is 128Dwords 1000 channel 6 buffer size is 256Dwords 1001 channel 6 buffer size is 512Dwords 1010 channel 6 buffer size is 1024Dwords 1111-1011 Reserved
23–20 CH5	This field is used to set the buffer size for channel 5. All the allowed combinations of bit setting are listed here . 0000 channel 5 buffer size is 1Dword 0001 channel 5 buffer size is 2Dwords 0010 channel 5 buffer size is 4Dwords 0011 channel 5 buffer size is 8Dwords 0100 channel 5 buffer size is 16Dwords 0101 channel 5 buffer size is 32Dwords 0110 channel 5 buffer size is 64Dwords 0111 channel 5 buffer size is 128Dwords 1000 channel 5 buffer size is 256Dwords 1001 channel 5 buffer size is 512Dwords 1010 channel 5 buffer size is 1024Dwords 1111-1011 Reserved
19–16 CH4	This field is used to set the buffer size for channel 4. All the allowed combinations of bit setting are listed here . 0000 channel 4 buffer size is 1Dword 0001 channel 4 buffer size is 2Dwords 0010 channel 4 buffer size is 4Dwords 0011 channel 4 buffer size is 8Dwords 0100 channel 4 buffer size is 16Dwords 0101 channel 4 buffer size is 32Dwords 0110 channel 4 buffer size is 64Dwords 0111 channel 4 buffer size is 128Dwords 1000 channel 4 buffer size is 256Dwords 1001 channel 4 buffer size is 512Dwords 1010 channel 4 buffer size is 1024Dwords 1111-1011 Reserved

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
15–12 CH3	This field is used to set the buffer size for channel 3. All the allowed combinations of bit setting are listed here . 0000 channel 3 buffer size is 1Dword 0001 channel 3 buffer size is 2Dwords 0010 channel 3 buffer size is 4Dwords 0011 channel 3 buffer size is 8Dwords 0100 channel 3 buffer size is 16Dwords 0101 channel 3 buffer size is 32Dwords 0110 channel 3 buffer size is 64Dwords 0111 channel 3 buffer size is 128Dwords 1000 channel 3 buffer size is 256Dwords 1001 channel 3 buffer size is 512Dwords 1010 channel 3 buffer size is 1024Dwords 1111-1011 Reserved
11–8 CH2	This field is used to set the buffer size for channel 2. All the allowed combinations of bit setting are listed here . 0000 channel 2 buffer size is 1Dword 0001 channel 2 buffer size is 2Dwords 0010 channel 2 buffer size is 4Dwords 0011 channel 2 buffer size is 8Dwords 0100 channel 2 buffer size is 16Dwords 0101 channel 2 buffer size is 32Dwords 0110 channel 2 buffer size is 64Dwords 0111 channel 2 buffer size is 128Dwords 1000 channel 2 buffer size is 256Dwords 1001 channel 2 buffer size is 512Dwords 1010 channel 2 buffer size is 1024Dwords 1111-1011 Reserved
7–4 CH1	This field is used to set the buffer size for channel 1. All the allowed combinations of bit setting are listed here . 0000 channel 1 buffer size is 1Dword 0001 channel 1 buffer size is 2Dwords 0010 channel 1 buffer size is 4Dwords 0011 channel 1 buffer size is 8Dwords 0100 channel 1 buffer size is 16Dwords 0101 channel 1 buffer size is 32Dwords 0110 channel 1 buffer size is 64Dwords 0111 channel 1 buffer size is 128Dwords 1000 channel 1 buffer size is 256Dwords 1001 channel 1 buffer size is 512Dwords 1010 channel 1 buffer size is 1024Dwords 1111-1011 Reserved
CH0	This field is used to set the buffer size for channel 0.

Table continues on the next page...

MIPI_HSI_RX_FIFO_SIZE_CONF1 field descriptions (continued)

Field	Description
	All the allowed combinations of bit setting are listed here . 0000 channel 0 buffer size is 1Dword 0001 channel 0 buffer size is 2Dwords 0010 channel 0 buffer size is 4Dwords 0011 channel 0 buffer size is 8Dwords 0100 channel 0 buffer size is 16Dwords 0101 channel 0 buffer size is 32Dwords 0110 channel 0 buffer size is 64Dwords 0111 channel 0 buffer size is 128Dwords 1000 channel 0 buffer size is 256Dwords 1001 channel 0 buffer size is 512Dwords 1010 channel 0 buffer size is 1024Dwords 1111-1011 Reserved

42.5.39 Tx Fifo Status Register (MIPI_HSI_TX_FIFO_STAT)

This register containts full and empty status for each Tx channel fifo

Address: 220_8000h base + 210h offset = 220_8210h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	CH15		CH14		CH13		CH12		CH11		CH10		CH9		CH8	
W																
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CH7		CH6		CH5		CH4		CH3		CH2		CH1		CH0	
W																
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_TX_FIFO_STAT field descriptions

Field	Description
31–30 CH15	00 Tx channel 15 fifo not Empty and Full; 01 Tx channel 15 fifo Empty; 10 Tx channel 15 fifo Full; 11 Reserved.
29–28 CH14	00 Tx channel 14 fifo not Empty and Full; 01 Tx channel 14 fifo Empty; 10 Tx channel 14 fifo Full; 11 Reserved.

Table continues on the next page...

MIPI_HSI_TX_FIFO_STAT field descriptions (continued)

Field	Description
27–26 CH13	00 Tx channel 13 fifo not Empty and Full; 01 Tx channel 13 fifo Empty; 10 Tx channel 13 fifo Full; 11 Reserved.
25–24 CH12	00 Tx channel 12 fifo not Empty and Full; 01 Tx channel 12 fifo Empty; 10 Tx channel 12 fifo Full; 11 Reserved.
23–22 CH11	00 Tx channel 11 fifo not Empty and Full; 01 Tx channel 11 fifo Empty; 10 Tx channel 11 fifo Full; 11 Reserved.
21–20 CH10	00 Tx channel 10 fifo not Empty and Full; 01 Tx channel 10 fifo Empty; 10 Tx channel 10 fifo Full; 11 Reserved.
19–18 CH9	00 Tx channel 9 fifo not Empty and Full; 01 Tx channel 9 fifo Empty; 10 Tx channel 9 fifo Full; 11 Reserved.
17–16 CH8	00 Tx channel 8 fifo not Empty and Full; 01 Tx channel 8 fifo Empty; 10 Tx channel 8 fifo Full; 11 Reserved.
15–14 CH7	00 Tx channel 7 fifo not Empty and Full; 01 Tx channel 7 fifo Empty; 10 Tx channel 7 fifo Full; 11 Reserved.
13–12 CH6	00 Tx channel 6 fifo not Empty and Full; 01 Tx channel 6 fifo Empty; 10 Tx channel 6 fifo Full; 11 Reserved.
11–10 CH5	00 Tx channel 5 fifo not Empty and Full; 01 Tx channel 5 fifo Empty; 10 Tx channel 5 fifo Full; 11 Reserved.
9–8 CH4	00 Tx channel 4 fifo not Empty and Full; 01 Tx channel 4 fifo Empty; 10 Tx channel 4 fifo Full; 11 Reserved.
7–6 CH3	00 Tx channel 3 fifo not Empty and Full; 01 Tx channel 3 fifo Empty; 10 Tx channel 3 fifo Full; 11 Reserved.

Table continues on the next page...

MIPI_HSI_TX_FIFO_STAT field descriptions (continued)

Field	Description
5–4 CH2	00 Tx channel 2 fifo not Empty and Full; 01 Tx channel 2 fifo Empty; 10 Tx channel 2 fifo Full; 11 Reserved.
3–2 CH1	00 Tx channel 1 fifo not Empty and Full; 01 Tx channel 1 fifo Empty; 10 Tx channel 1 fifo Full; 11 Reserved.
CH0	00 Tx channel 0 fifo not Empty and Full; 01 Tx channel 0 fifo Empty; 10 Tx channel 0 fifo Full; 11 Reserved.

42.5.40 Rx Fifo Status Register (MIPI_HSI_RX_FIFO_STAT)

This register containts full and empty status for each Rx channel fifo

Address: 220_8000h base + 214h offset = 220_8214h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	CH15		CH14		CH13		CH12		CH11		CH10		CH9		CH8	
W																
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CH7		CH6		CH5		CH4		CH3		CH2		CH1		CH0	
W																
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

MIPI_HSI_RX_FIFO_STAT field descriptions

Field	Description
31–30 CH15	00 Rx channel 15 fifo not Empty and Full; 01 Rx channel 15 fifo Empty; 10 Rx channel 15 fifo Full; 11 Reserved.
29–28 CH14	00 Rx channel 14 fifo not Empty and Full; 01 Rx channel 14 fifo Empty; 10 Rx channel 14 fifo Full; 11 Reserved.

Table continues on the next page...

MIPI_HSI_RX_FIFO_STAT field descriptions (continued)

Field	Description
27–26 CH13	00 Rx channel 13 fifo not Empty and Full; 01 Rx channel 13 fifo Empty; 10 Rx channel 13 fifo Full; 11 Reserved.
25–24 CH12	00 Rx channel 12 fifo not Empty and Full; 01 Rx channel 12 fifo Empty; 10 Rx channel 12 fifo Full; 11 Reserved.
23–22 CH11	00 Rx channel 11 fifo not Empty and Full; 01 Rx channel 11 fifo Empty; 10 Rx channel 11 fifo Full; 11 Reserved.
21–20 CH10	00 Rx channel 10 fifo not Empty and Full; 01 Rx channel 10 fifo Empty; 10 Rx channel 10 fifo Full; 11 Reserved.
19–18 CH9	00 Rx channel 9 fifo not Empty and Full; 01 Rx channel 9 fifo Empty; 10 Rx channel 9 fifo Full; 11 Reserved.
17–16 CH8	00 Rx channel 8 fifo not Empty and Full; 01 Rx channel 8 fifo Empty; 10 Rx channel 8 fifo Full; 11 Reserved.
15–14 CH7	00 Rx channel 7 fifo not Empty and Full; 01 Rx channel 7 fifo Empty; 10 Rx channel 7 fifo Full; 11 Reserved.
13–12 CH6	00 Rx channel 6 fifo not Empty and Full; 01 Rx channel 6 fifo Empty; 10 Rx channel 6 fifo Full; 11 Reserved.
11–10 CH5	00 Rx channel 5 fifo not Empty and Full; 01 Rx channel 5 fifo Empty; 10 Rx channel 5 fifo Full; 11 Reserved.
9–8 CH4	00 Rx channel 4 fifo not Empty and Full; 01 Rx channel 4 fifo Empty; 10 Rx channel 4 fifo Full; 11 Reserved.
7–6 CH3	00 Rx channel 3 fifo not Empty and Full; 01 Rx channel 3 fifo Empty; 10 Rx channel 3 fifo Full; 11 Reserved.

Table continues on the next page...

MIPI_HSI_RX_FIFO_STAT field descriptions (continued)

Field	Description
5–4 CH2	00 Rx channel 2 fifo not Empty and Full; 01 Rx channel 2 fifo Empty; 10 Rx channel 2 fifo Full; 11 Reserved.
3–2 CH1	00 Rx channel 1 fifo not Empty and Full; 01 Rx channel 1 fifo Empty; 10 Rx channel 1 fifo Full; 11 Reserved.
CH0	00 Rx channel 0 fifo not Empty and Full; 01 Rx channel 0 fifo Empty; 10 Rx channel 0 fifo Full; 11 Reserved.

42.5.41 Ahb Master Config Register (MIPI_HSI_AHB_MASTER_CONF)

This register used to config hsi internal ahb master

Address: 220_8000h base + 228h offset = 220_8228h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved						DP_HOLD_CYCLE				DMA_MODE		DMA_INSERT_IDLE_NUM			
Reset	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

MIPI_HSI_AHB_MASTER_CONF field descriptions

Field	Description
31–10 Reserved	This field is reserved. Reserved, always set to zero.
9–6 DP_HOLD_ CYCLE	These bits used to set the number of cycles for DP access fifo.
5–4 DMA_MODE	00 Once AHB master get hgrant from bus, it will set htrans "IDLE" for serval ahb cycles.In the serval cycles, once it found dataport is accessing fifo, it will release bus. 01 Once AHB master get hgrant from bus, it will set htrans "IDLE" for serval ahb cycles.After the serval cycles, once it found dataport is accessing fifo, it will keep on sending "IDLE" out untill dataport finish accessing fifo. 1x Once AHB master get hgrant from bus, dataport can not access fifo untill a dma operation done.

Table continues on the next page...

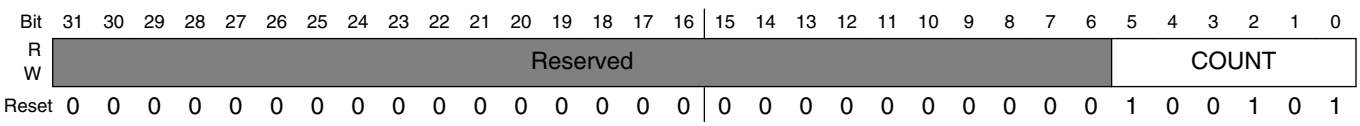
MIPI_HSI_AHB_MASTER_CONF field descriptions (continued)

Field	Description
DMA_INSERT_IDLE_NUM	These bits used to set the number of "IDLE" cycles when DMA_MODE == 2'b0x.

42.5.42 TX Break Length Register (MIPI_HSI_TX_BREAK_LEN)

This register used to set tx break length

Address: 220_8000h base + 22Ch offset = 220_822Ch



MIPI_HSI_TX_BREAK_LEN field descriptions

Field	Description
31–6 Reserved	This field is reserved. Reserved, always set to zero.
COUNT	The tx break length count. 6'h00 64 6'h01 1 6'h3f 63

Chapter 43
MediaLB (MLB)

43.1 Overview

The MLB implements the required functionality of a Media Local Bus (MediaLB) Device. Functionality includes:

- Transmission of commands and data when functioning as the transmitting device associated with a ChannelAddress
- Reception of data and transmission of RxStatus responses when functioning as the receiving device associated with a ChannelAddress
- MediaLB lock detection
- SystemChannel command handling

MediaLB Device functionality is implemented with an MediaLB 3-pin interface . The MediaLB interfaces are capable of exchanging data at speeds up to 1024xFs in 3-pin mode.

A set of physical channels for exchanging data over the MediaLB bus is supported. These physical channels (4 bytes in length, or a quadlet) can be grouped into logical channels, where each logical channel is referenced using a ChannelAddress and represents a unidirectional data path between a specific MediaLB Device transmitting the data and the MediaLB Device(s) receiving the data. The logical channels, configured by system software, can be any combination of channel types (synchronous, asynchronous, isochronous, or control) and direction (transmit or receive).

43.1.1 Block Diagram

The following figure is the top-level block diagram of the MLB behavioral models.

43.1.1.1 Bus Interfaces

The external bus interfaces include:

- MediaLB 3-pin Interface
- MediaLB 6-pin Interface
- MediaLB Analog Configuration Interface
- Channel Table Bus (CTB) Interface
- Data Buffer Bus (DBB) Interface
- System Interface
- AMBA Advanced High-performance Bus (AHB) Interface
- AMBA Advanced Peripheral Bus (APB) Interface

43.2 External Signals

The table found here describes the external signals of MLB.

Table 43-1. MLB External Signals

Signal	Description	Pad	Mode	Direction
MLB_CLK	3-Wire clock signal	ENET_TXD1	ALT0	I
		GPIO_3	ALT7	
MLB_CLK_N	Negative clock signal	MLB_CN	No Muxing	I
MLB_CLK_P	Positive clock signal	MLB_CP	No Muxing	I
MLB_DATA	3-Wire data signal	ENET_MDC	ALT0	I/O
		GPIO_2	ALT7	
MLB_DATA_N	Negative data signal	MLB_DN	No Muxing	I/O
MLB_DATA_P	Positive data signal	MLB_DP	No Muxing	I/O
MLB_SIG	3-Wire signal	ENET_RXD1	ALT0	I/O
		GPIO_6	ALT7	
MLB_SIG_N	Negative signal	MLB_SIG_N	No Muxing	I/O
MLB_SIG_P	Positive signal	MLB_SIG_P	No Muxing	I/O

43.3 Clocks

The table found here describes the clock sources for MLB.

Please see Clock Controller Module (CCM) for clock setting, configuration and gating information.

Table 43-2. MLB150 Clocks

Clock name	Clock Root	Description
hclk	ahb_clk_root	AHB bus clock
ipg_clk_s	ipg_clk_root	Peripheral access clock
sys_clk	axi_clk_root	Module clock
mem_ct_CLK	axi_clk_root	Channel table bus clock
mem_db_CLK	axi_clk_root	Data buffer bus clock

43.4 Functional Description

This section describes the functional architecture of the MLB.

The internal functional blocks of the MLB include:

- MediaLB Block (mlb_top) - Implements the physical and link-layer requirements of MediaLB 3-pin interface. Serial-to-parallel and parallel-to-serial data transformations are implemented, as well as MediaLB frame synchronization
- Host Bus Interface Block (hbi_top) - Provides 16-bit parallel slave access to all MOST channels and data types for the external Host Controller (HC). The HBI supports up to 64 independent channels with a minimum access latency of 40 ns per word and a maximum bandwidth of 400 Mbps
- Routing Fabric Block (rf_top) - Manages the flow of data between the MediaLB block and the HBI block, implementing a bus arbiter and muxing logic to the Channel Table RAM (CTR) and the Data Buffer RAM (DBR)
- Memory Interface Block (mif_top) - Implements a bridge between the I/O bus and the customer-implemented RAMs (i.e. Channel Table and Data Buffer)
- Interrupt Interface Block (intif_top) - Sends notifications to HBI that there are changes to the channel descriptors
- Clocks, Power, and Reset Block (cpr_top) - Implements clock and reset muxing and synchronization
- AMBA AHB Block (ahb_top) - Implements a bus bridge between the AHB master and the HBI slave interfaces
- AMBA APB Block (apb_top) - Implements a bus bridge that translates the two cycle APB interface signals to the single-cycle I/O interface signals

43.4.1 MediaLB Block

The Media Local Bus (MediaLB) block supports MediaLB 3-pin interface. Both MediaLB interfaces provide real-time access to all network data types including streaming, packet, control, and isochronous data.

- MediaLB 3-pin Interface - Supports the MediaLB protocol for single-ended 3-pin mode, with a maximum data rate of 1024xFs (49.152 MHz at Fs=48 kHz).

43.4.1.1 MediaLB Channel Address to Logical Channel Mapping

The MediaLB channel addresses are mapped to the logical channels as follows.

Table 43-3. MediaLB Channel Address to Logical Channel Mapping

Channel Address	Logical Channel
0x0002	1
0x0004	2
0x0006	3
....	....
0x007C	62
0x007E	63
0x01FE	0*
* Logical Channel 0 is the System Channel and is reserved.	

43.4.2 Host Bus Interface Block

The Host Bus Interface (HBI) block provides a 16-bit parallel slave port that provides an external Host Controller (HC) with access to all MOST channels and data types.

Up to 64 independent HBI channels are available to the HC, each configurable for either transmitting or receiving a particular application data type (synchronous, isochronous, asynchronous, or control). The HBI block provides source and sink access to the full network data bandwidth.

43.4.2.1 HBI Physical Addresses

To access a particular HBI DMA channel, hardware must first translate the HBI channel address to a channel allocation table (CAT) physical address. This physical address is then used to retrieve the channel label (CL), which in turn retrieves the channel descriptor.

See Table 43-4 for more information on the mapping between the HBI channel address and physical address.

Table 43-4. HBI Channel Address to Physical Address Mapping

HBI Channel	CAT Address	CAT Offset
0x0	0x88	000
0x1	0x88	001
0x2	0x88	010
0x3	0x88	011
0x4	0x88	100
0x5	0x88	101
0x6	0x88	110
0x7	0x88	111
0x8	0x89	000
...	...	...
0x3E	0x8F	110
0x3F	0x8F	111

43.4.3 Routing Fabric Block

The Routing Fabric (RF) block manages the flow of data between the MediaLB Port and the HBI Port. Bus multiplexers and a bus arbiter are implemented in the RF block for accessing the channel table RAM (CTR) and data buffer RAM (DBR).

Each DMA controller in the routing fabric uses Channel Descriptors (stored in the CTR) to manage access to dynamic buffers in the DBR.

43.4.3.1 Data Buffer RAM

The MLB has an external data buffer RAM (DBR) that is 8-bit x 16k entries deep. The DBR provides dynamic circular buffering between the transmit and receive devices.

The size and location of each data buffer is defined by software in the channel descriptor table (CDT), which is located in the CTR.

Receive devices retain the write address pointer to the associated circular data buffer in the DBR, while transmit devices retain the read address pointer. The DMA controllers in the routing fabric are responsible for ensuring that the circular buffers do not overflow or underflow. Each channel type (e.g. synchronous, isochronous, asynchronous and control) has Full and Empty detection.

43.4.3.1.1 Synchronous Channels

For synchronous channels, two mechanisms prevent overflow and underflow of the data buffer:

- Hardware aligns the read pointer (RPTR) to the write pointer (WPTR) to ensure an offset of two sub-buffers.
- RPTR and WPTR are periodically synchronized to the start of the next sub-buffer (e.g. following a FRAMESYNC).

43.4.3.1.2 Isochronous Channels

For isochronous channels, hardware does not read from an empty data buffer or write to a full data buffer. The conditions used by hardware for detection include:

Data buffer Empty condition: (RPTR = WPTR) AND (BF = 0), and

Data buffer Full condition: (WPTR = RPTR) AND (BF = 1).

43.4.3.1.3 Asynchronous and Control Channels

For asynchronous and control channels, hardware does not read from an empty data buffer or write to a full data buffer. Hardware evaluates the DMA pointers (RPTR, WPTR) and packet count (RPC, WPC) to detect the data buffer condition, where:

- Data buffer Empty condition: (RPTR = WPTR) AND (RPC = WPC), and
- Data buffer Full condition: ((WPTR = RPTR) AND (WPC != RPC)) OR (WPC = (RPC - 1)).

43.4.3.2 Channel Table RAM

The MLB has an external Channel Table RAM (CTR) that is 128-bit x 144-entry. The CTR allows system software to dynamically configure channel routing and allocate data buffers in the DBR.

The CTR is logically divided into three sub-tables:

- Channel Descriptor Table (CDT)
- AHB Descriptor Table (ADT)
- Channel Allocation Table (CAT)

43.4.3.2.1 Address Mapping

Table 43-5. CTR Address Mapping

Label	Address	Bits 127…96	Bits 95…64	Bits 63…32	Bits 31…0
Channel Descriptor Table (CDT):					
CDT	0x00	CDT0[127:0], CL = 0			
	0x01	CDT1[127:0], CL = 1			
	0x02	CDT2[127:0], CL = 2			
	...	...			
	0x3D	CDT61[127:0], CL = 61			
	0x3E	CDT62[127:0], CL = 62			
	0x3F	CDT63[127:0], CL = 63			
AHB Descriptor Table (ADT):					
ADT*	0x40	ADT0[127:0]			
	0x41	ADT1[127:0]			
	0x42	ADT2[127:0]			
	...	...			
	0x7D	ADT61[127:0]			
	0x7E	ADT62[127:0]			
	0x7F	ADT63[127:0]			
Channel Allocation Table (CAT):					
CAT for MediaLB	0x80	CAT7	CAT6	CAT5	CAT4	CAT3	CAT2	CAT1	CAT0
	...	...	...	...	...	...	...	...	...
	0x87	CAT63	CAT62	CAT61	CAT60	CAT59	CAT58	CAT57	CAT56
CAT for HBI*	0x88	CAT71	CAT70	CAT69	CAT68	CAT67	CAT66	CAT65	CAT64
	...	...	...	...	...	...	...	...	...
	0x8F	CAT127	CAT126	CAT125	CAT124	CAT123	CAT122	CAT121	CAT120
* A fixed relationship exists between ADT entries and HBI CAT entries. When using HBI channel 0 (CAT64) on should program ADT0. When using HBI channel 1 (CAT65) on should program ADT1, and so on.									

43.4.3.2.2 Channel Allocation Table

The Channel Allocation Table (CAT) is comprised of 16 CTR entries (addresses 0x80 - 0x8F), as shown in Table 43-5. Each 16-bit CAT entry represents a logical connection to or from a transmit/receive device (e.g. MediaLB or HBI channel). All entries are indexed according to a fixed physical address assigned to every Rx/Tx channel (as shown in Table 43-6). The value stored in a CAT entry includes a 6-bit Connection Label, which

provides a pointer to the CDT. To complete a logical channel and form a routing connection, system software must assign the same Connection Label to both the Rx and Tx channels.

Table 43-6. CAT Entry Map

Peripheral	Tx Channels	Rx Channels	CAT Start Index	CAT End Index	Entries
MediaLB	0 to 64	64 - Tx Channels	0	63	64
HBI	0 to 64	64 - Tx Channels	64	127	64

The format of a full CAT entry is shown in Table 43-7, with field descriptions described in Table 43-8. All reserved bits of a CAT entry field should be written as zero.

Table 43-7. CAT Entry Formats

Channel Type	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Isochronous	rsvd	FCE	rsvd	RNW	CE	CT[2:0] = 3			rsvd		CL[5:0]					
Asynchronous	rsvd		MT	RNW	CE	CT[2:0] = 2			rsvd		CL[5:0]					
Control	rsvd		MT	RNW	CE	CT[2:0] = 1			rsvd		CL[5:0]					
Synchronous	rsvd	MFE	MT	RNW	CE	CT[2:0] = 0			rsvd		CL[5:0]					

Table 43-8. CAT Field Definitions

Field	Description
CL[5:0]	Connection Label (offset into CDT)
CT[2:0]	Channel Type (Others): 111 = Reserved 110 = Reserved 101 = Reserved 100 = Reserved 011 = Isochronous 010 = Asynchronous 001 = Control 000 = Synchronous
CE	Channel Enable: 1 = Enabled 0 = Disabled
RNW	Read Not Write: 1 = Read 0 = Write
MT	Mute Enable: 1 1 = Enabled 0 = Disabled
FCE	Flow Control Enable: 2 1 = Enabled 0 = Disabled
MFE	Multi-Frame per Sub-buffer Enable: 3 1 = Enabled 0 = Disabled
rsvd	Reserved. Software writes a zero to all Reserved bits when the entry is initialized. The Reserved bits are Read-only after initialization.
1. When set for synchronous channels, the MT bit forces Rx channels to write zeros into the channel data buffer, and Tx channels to output zeros on the physical interface. When set for asynchronous and control channels, the MT bit causes DMA to halt at a packet boundary. Not valid for isochronous channels. 2. The FCE bit is used by MediaLB isochronous Rx channels only. 3. The MFE bit is used by MediaLB synchronous channels only.	

43.4.3.2.2.1 Channel Setup

Data direction in the MLB is in reference to the DBR. Therefore, the data direction of CAT entries corresponding to the same channel is reversed for the HBI CAT and the MediaLB CAT.

For a Tx channel (from the HC to the MediaLB interface):

- HBI CAT entry: RNW = 0 (write)
- MediaLB CAT entry: RNW = 1 (read)

Conversely, for a Rx channel (data from MediaLB to HC):

- HBI CAT entry: RNW = 1 (read)
- MediaLB CAT entry: RNW = 0 (write)

The figure below illustrates the directional relationship in the MLB.

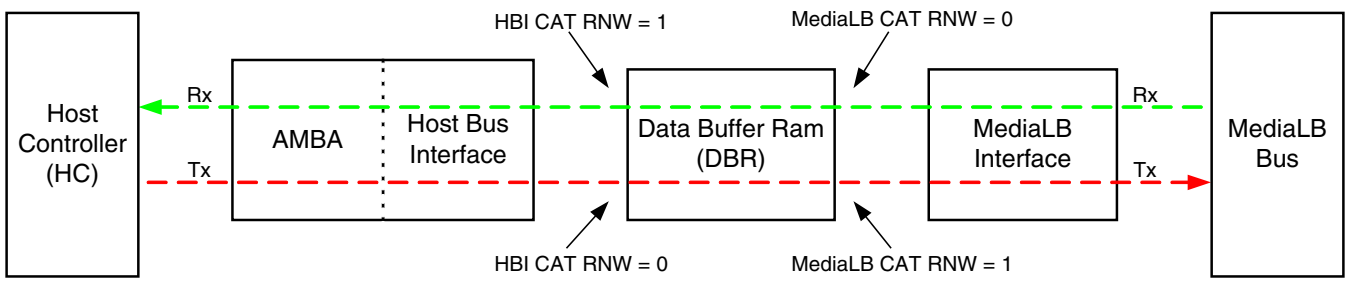


Figure 43-1. MLB DBR Directional Relationship

43.4.3.2.3 Channel Descriptor Table

The *Channel Descriptor Table* (CDT) is comprised of 64 CTR entries (addresses 0x00 - 0x3F), as shown in Table 43-5.

Each 128-bit CDT entry (also referred to as a *Channel Descriptor*) is referenced by a *Connection Label* and contains information about a data buffer in the DBR (e.g. buffer size, address pointers).

The format of each CDT entry (also referred to as a *Channel Descriptor*) is dependent on the channel type (e.g. synchronous, isochronous, asynchronous, or control).

NOTE

All reserved *Channel Descriptor* bits must be written to '0' by software when initialized.

43.4.3.2.3.1 Synchronous Channel Operation

A sample synchronous data buffer is shown in the figure found here.

A sample synchronous data buffer is shown in the figure below. Each data buffer contains four sub-buffers and each sub-buffer contains space for 1 to 64 frames of data, determined by MLBC0.FCNT[2:0].

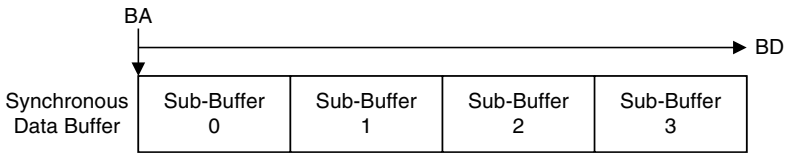


Figure 43-2. Synchronous Data Buffer Structure

43.4.3.2.3.2 Synchronous Channel Descriptors

The format and field definitions for a synchronous CDT entry are shown in the tables below respectively.

Table 43-9. Synchronous CDT Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	WSBC		Reserved													
16	RSBC		Reserved													
32	Reserved															
48	Reserved															
64	WSTS[3:0]				WPTR[11:0]											
80	RSTS[3:0]				RPTR[11:0]											
96	Reserved				BD[11:0]											
112	Reserved		BA[13:0]													

Table 43-10. Synchronous CDT Entry Field Definitions

Field	Description	Details	Accessibility
BA	Buffer Base Address	- BA can start at any byte in the 16k DBR	r,w
BD	Buffer Depth	- BD = size of buffer in bytes - 1 - Buffer end address = BA + BD - BD = 4 x m x bpf - 1, where: m = frames per sub-buffer (for MFE = 0, m = 1) bpf = bytes per frame.	r,w
RPTR	Read Pointer	- Software initializes to zero, hardware updates - Counts the read address offset within a buffer - DMA read address = BA + RPTR	r,w,u
WPTR	Write Pointer	- Software initializes to zero, hardware updates	r,w,u

Table continues on the next page...

Table 43-10. Synchronous CDT Entry Field Definitions (continued)

Field	Description	Details	Accessibility
		- Counts the write address offset within a buffer - DMA write address = BA + WPTR	
RSBC	Read Sub-buffer Counter	- Software initializes to zero, hardware updates - Counts the read sub-buffer offset - DMA uses for pointer management	r,w,u
WSBC	Write Sub-buffer Counter	- Software initializes to zero, hardware updates - Counts the write sub-buffer offset - DMA uses for pointer management	r,w,u
RSTS	Read Status	- Software initializes to zero, hardware updates - RSTS states:* xxx0 = normal operation (no mute) xxx1 = normal operation (mute) xx0x = idle	r,w,u
WSTS	Write Status	- Software initializes to zero, hardware updates - WSTS states:* xxx0 = normal operation (no mute) xxx1 = normal operation (mute) xx0x = idle 1xxx = command protocol error	r,w,u
Reserved	Reserved	- Software writes a zero to all *Reserved* bits when the entry is initialized. The *Reserved* bits are *Read-only* after initialization.	r,w,u
* Only valid for DMA pointers associated with the MediaLB block (Not valid for HBI block related pointers).			

43.4.3.2.3.3 Isochronous Channel Descriptors

The format and field definitions for an isochronous CDT entry are shown in the tables below respectively.

Table 43-11. Isochronous CDT Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Reserved															
16	Reserved															
32	Reserved							BS[8:0]								
48	Reserved															
64	WSTS[2:0]			WPTR[12:0]												

Table continues on the next page...

Table 43-11. Isochronous CDT Entry Format (continued)

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
80	RSTS[2:0]			RPTR[12:0]												
96	Reserved			BD[12:0]												
112	BF	rsvd	BA[13:0]													

Table 43-12. Isochronous CDT Entry Field Definitions

Field	Description	Details	Accessibility
BA	Buffer Base Address	- BA can start at any byte in the 16k DBR	r,w
BD	Buffer Depth	- BD = size of buffer in bytes - 1 - Buffer end address = BA + BD - Isochronous buffers must be large enough to hold at least 3 blocks (packets) of data - Buffer depth must be a integer multiple of blocks	r,w
BF	Buffer Full	- Software initializes to zero, hardware updates - DMA write hardware sets BF when the buffer is full - DMA read hardware clears BF when the buffer is empty - BF is valid only when the buffer is full or empty, otherwise ignore	r,w,u
BS	Block Size	- BS defines when to begin the DMA to the data buffer - BS = buffer block size in bytes - 1 - For Rx channels, the DMA writes start when the number of empty bytes (SPACE) in the data buffer >= the block size - For Tx channels, the DMA reads start when the number of valid bytes (VALID) in the data buffer >= the block size	r,w,u
RPTR	Read Pointer	- Software initializes to zero, hardware updates - Counts the read address offset within a buffer - DMA read address = BA + RPTR	r,w,u
WPTR	Write Pointer	- Software initializes to zero, hardware updates - Counts the write address offset within a buffer - DMA write address = BA + WPTR	r,w,u
RSTS	Read Status	- Software initializes to zero, hardware updates - RSTS states:* xx1 = active xx0 = idle	r,w,u
WSTS	Write Status	- Software initializes to zero, hardware updates - WSTS states:* xx1 = active xx0 = idle x1x = command protocol error	r,w,u

Table continues on the next page...

Table 43-12. Isochronous CDT Entry Field Definitions (continued)

Field	Description	Details	Accessibility
		1xx = buffer overflow (FCE = 0 only)	
Reserved	Reserved	- Software writes a zero to all *Reserved* bits when the entry is initialized. The *Reserved* bits are *Read-only* after initialization.	r,w,u
* Only valid for DMA pointers associated with the MediaLB block (Not valid for HBI block related pointers).			

43.4.3.2.3.4 Asynchronous and Control Channel Descriptors

The format and field definitions for asynchronous and control CDT entries are shown in the tables below respectively.

Table 43-13. Asynchronous/Control CDT Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	WPC[4:0]					Reserved										
16	RPC[4:0]					Reserved										
32	rsvd	WPC[7:5]			Reserved											
48	rsvd	RPC[7:5]			Reserved											
64	WSTS[3:0]				WPTR[11:0]											
80	RSTS[3:0]				RPTR[11:0]											
96	RSTS[4]	WSTS[4]	rsvd		BD[11:0]											
112	Reserved		BA[13:0]													

Table 43-14. Asynchronous/Control CDT Entry Field Definitions

Field	Description	Details	Accessibility
BA	Buffer Base Address	- BA can start at any byte in the 16k DBR	r,w
BD	Buffer Depth	- BD = size of buffer in bytes - 1 - Buffer end address = BA + BD - BD >= max packet length - 1	r,w
RPC	Read Packet Count	- Software initializes to zero, hardware updates - Used in conjunction with WPC, RPTR and WPTR to determine if the buffer is empty or full	r,w,u
WPC	Write Packet Count	- Software initializes to zero, hardware updates - Used in conjunction with RPC, RPTR and WPTR to determine if the buffer is empty or full	r,w,u
RPTR	Read Pointer	- Software initializes to zero, hardware updates - Counts the read address offset within a buffer - DMA read address = BA + RPTR	r,w,u

Table continues on the next page...

Table 43-14. Asynchronous/Control CDT Entry Field Definitions (continued)

Field	Description	Details	Accessibility
WPTR	Write Pointer	- Software initializes to zero, hardware updates - Counts the write address offset within a buffer - DMA read address = BA + WPTR	r,w,u
RSTS	Read Status	- Software initializes to zero, hardware updates - Status states:* x0x00 = idle xx1xx = *ReceiverProtocolError* response received from Rx Device 1xxxx = *ReceiverBreak* command received from Rx Device	r,w,u
WSTS	Write Status	- Software initializes to zero, hardware updates - Status states:* x0x00 = idle xx1xx = command protocol error detected 1xxxx = *AsyncBreak*/*ControlBreak* command received from Tx Device	r,w,u
Reserved	Reserved	Software writes a zero to all *Reserved* bits when the entry is initialized. The *Reserved* bits are *Read-only* after initialization.	r,w,u
* Only valid for DMA pointers associated with the MediaLB block (Not valid for HBI block related pointers)			

43.4.4 Memory Interface Block

The Memory Interface (MIF) block implements a bridge between the I/O and the CTB or DBB interfaces.

The MIF block diagram is shown in the figure below. The targeted RAM (CTR or DBR) is determined by the target location bit in the memory address register (MADR.TB).

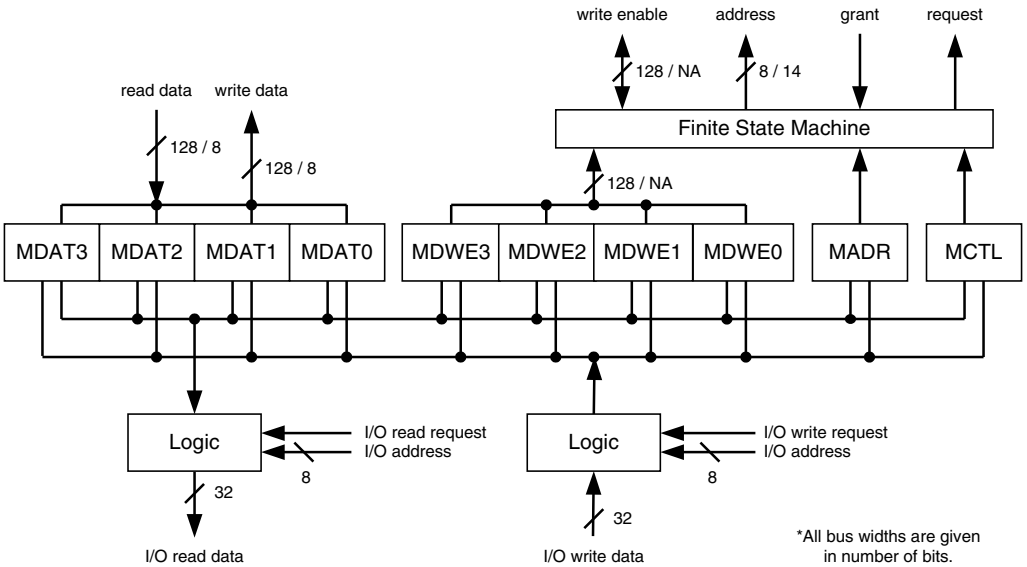


Figure 43-3. MIF Block Diagram

NOTE

The size of the read data, write data, write enable, and address buses are dependent on whether the CTB or DBB interfaces are used. In the figure above, the first number is for CTR accesses while the second is for DBR accesses (i.e. CTR bus width / DBR bus width).

43.4.4.1 CTR Access

The MIF block allows the HC to directly access the external Channel Table RAM (CTR) when MADR.TB is cleared. Any write to the MADR register triggers a single read or write cycle. Reading from the MADR register does not initiate read/write access.

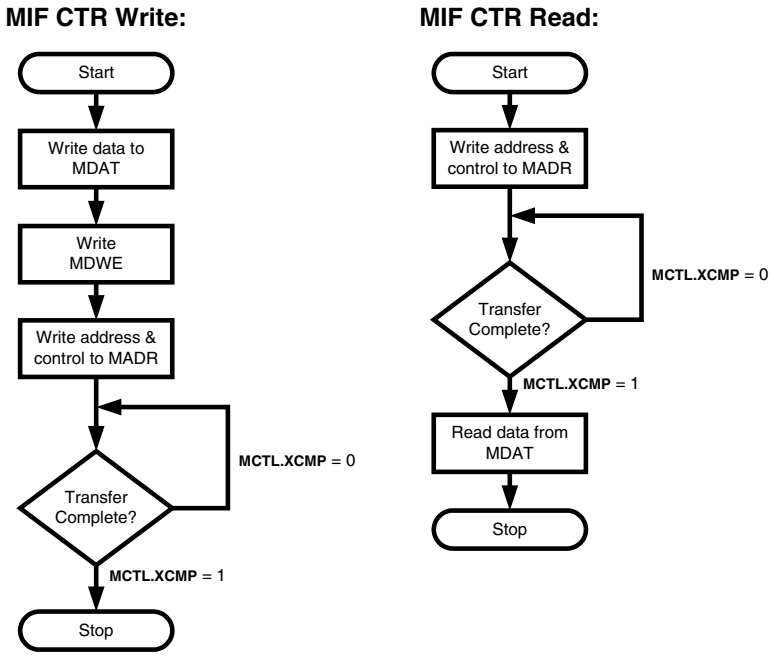


Figure 43-4. MIF CTR Read and Write Flow Diagrams

43.4.4.1.1 Direct CTR Writes

For a direct write of the CTR, the HC first loads the 128-bit data entry into the MDAT0-3 registers. Bitwise write enable control is available via the MDWE0-3 registers.

After the MDATn and MDWEn registers are set up, a write cycle is initiated by writing the address and control information to MADR as follows:

- MADR.WNR = 1
- MADR.TB = 0
- MADR.ADDR[7:0] = 8-bit Target Address

The MIF block sets MCTL.XCMP = 1 to inform the HC when the write is complete.

43.4.4.1.2 Direct CTR Reads

For a direct read of the CTR, the HC initiates a read cycle by writing the address and control information to MADR as follows:

- MADR.WNR = 0

- MADR.TB = 0
- MADR.ADDR[7:0] = 8-bit Target Address

The MIF block sets MCTL.XCMP = 1 to inform the HC when the read is complete. The HC can then read the 128-bit data entry from the MDAT0-3 registers.

43.4.4.1.3 CTR Addressing

The CTR is addressed as a 128-bit wide value. However, the MIF block can only access 32 bits of the addressed CTR data in a single access. Therefore, four 32-bit accesses through the MIF block are required to access a single 128-bit value (e.g. CDT entry).

To access a 16-bit CAT entry in the CTR, only a single access through the MIF is required. For example, to load a CAT61 entry for an isochronous Tx channel with mute and flow control enabled:

- Write MDAT2 = 7B070000h (assumes Connection Label = 7)
- MDWE2 = FFFF0000h (bitwise write enable for 16 msbs; assumes MDWE0/1/3 = 00000000h)
- MADR = 80000087h (write CTR address 87h)

43.4.4.2 DBR Access

The MIF block allows the HC to access the external Data Buffer RAM (DBR) directly when MADR.TB is set. Any write to the MADR register triggers a single read or write cycle. Reading from the MADR register does not initiate read/write access.

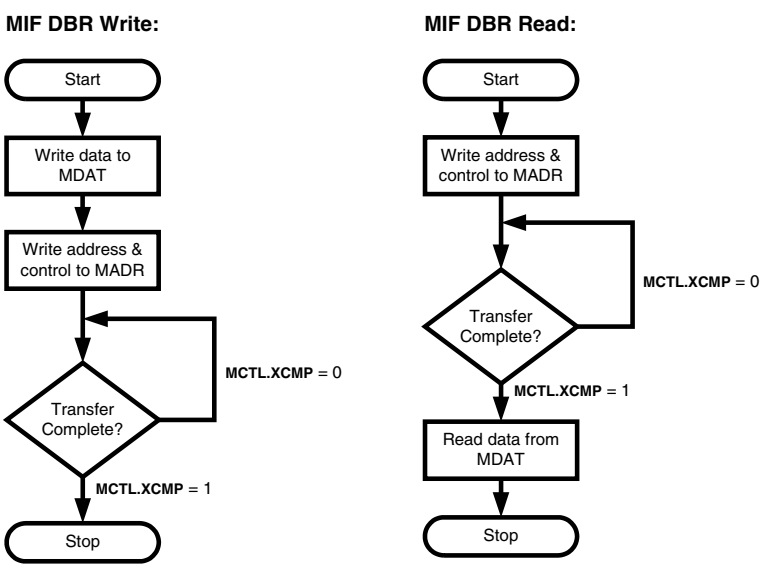


Figure 43-5. MIF DBR Read and Write Flow Diagrams

43.4.4.2.1 Direct DBR Writes

For a direct write of the DBR, the HC first loads the 8-bit data entry into the MDAT0 register at bits[7:0]. MDAT1-3 and MDWE0-3 are not used for DBR access.

After the MDAT0 register is set up, a write cycle is initiated by writing the address and control information to MADR as follows:

- MADR.WNR = 1
- MADR.TB = 1
- MADR.ADDR[13:0] = 14-bit Target Address

The MIF block sets MCTL.XCMP = 1 to inform the HC when the write is complete.

43.4.4.2.2 Direct DBR Reads

For a direct read of the DBR, the HC initiates a read cycle by writing the address and control information to MADR as follows:

- MADR.WNR = 0
- MADR.TB = 1
- MADR.ADDR[13:0] = 14-bit Target Address

The MIF block sets MCTL.XCMP = 1 to inform the HC when the read is complete. The HC can then read the 8-bit data entry from the MDAT0 register at bits[7:0].

43.4.5 Interrupt Interface Block

The Interrupt Interface (INTIF) block performs a low-priority polling algorithm of each of the HBI channel descriptors.

The INTIF alerts the HBI block when specific changes to HBI Channel Descriptors occur.

- For asynchronous and control read/write channels:
  - a packet is available to read in the channel buffer, or
  - sufficient empty space is available in the channel buffer to accept a requested packet write.
- For isochronous read/write channels:
  - the number of valid bytes in the channel buffer exceeds the block size, or
  - the number of empty bytes in the channel buffer exceeds the block size.

43.4.6 AMBA AHB Block

The AMBA AHB block manages data exchange between local channel data buffers within the MLB and the system memory buffer.

To support system memory buffering, a ping-pong memory structure is implemented on a per-channel basis using 128-bit descriptors for AHB Descriptor Table (ADT) entries.

NOTE

The 64 ADT entries are directly mapped to the 64 HBI physical channels.

Each logical channel is assigned a separate 128-bit descriptor, defining the data buffers in the system memory used by the DMA interface for that channel. The descriptors are stored at fixed addresses in the external CTR.

43.4.6.1 AHB Descriptor Table

The table below provides an overview of field definitions for ADT entries.

Table 43-15. ADT Field Definitions

Field	No. of Bits	Description	Accessibility
CE	1	Channel enable: 0 = Disabled 1 = Enabled	r,w,u
LE	1	Endianess select: 0 = Big Endian 1 = Little Endian	r,w
PG	1	Page pointer. Software initializes to zero, hardware writes thereafter. 0 = Ping buffer 1 = Pong buffer	r,w,u
RDY1	1	Buffer ready bit for ping buffer page: 0 = Not ready 1 = Ready	r,w
RDY2	1	Buffer ready bit for pong buffer page: 0 = Not ready 1 = Ready	r,w
DNE1	1	Buffer done bit for ping buffer page: 0 = Not done 1 = Done	r,u,c0
DNE2	1	Buffer done bit for pong buffer page: 0 = Not done 1 = Done	r,u,c0
ERR1	1	AHB error response detected for ping buffer page: 0 = No error 1 = Rrror	r,u,c0
ERR2	1	AHB error response detected for pong buffer page: 0 = No error 1 = Error	r,u,c0
PS1	1	Packet start bit for ping buffer page: 0 = No packet start 1 = Packet start Reserved for synchronous and isochronous channels.	r,w,u (both Tx and Rx)
PS2	1	Packet start bit for pong buffer page: 0 = No packet start 1 = Packet start Reserved for synchronous and isochronous channels.	r,w,u (both Tx and Rx)
MEP1	1	Most Ethernet Packet (MEP) indicator for ping buffer page: 0 = Not MEP 1 = MEP MEP1 only valid for the first page of a segmented buffer. Reserved for control, synchronous and isochronous channels.	Rsvd for Tx r,u,c0 for Rx

Table continues on the next page...

Table 43-15. ADT Field Definitions (continued)

Field	No. of Bits	Description	Accessibility
MEP2	1	MEP packet indicator for pong buffer page: 0 = not MEP 1 = MEP MEP2 only valid for the first page of a segmented buffer. Reserved for control, synchronous and isochronous channels.	Rsvd for Tx r,u,c0 for Rx
BD1*	11 to 13	Buffer depth for ping buffer page: 11 or 12-bits for asynchronous and control channels. 13-bits for synchronous and isochronous channels.	r,w
BD2*	11 to 13	Buffer depth for pong buffer page: 11 or 12-bits for asynchronous and control channels. 13-bits for synchronous and isochronous channels.	r,w
BA1	32	Buffer base address for ping buffer page	r,w
BA2	32	Buffer base address for pong buffer page	r,w
Reserved	varies	Software writes a zero to all *Reserved* bits when the entry is initialized. The *Reserved* bits are *Read-only* after initialization.	r,w,u
* The buffer depth (BD1 and BD2) for synchronous channels must consider if *Multi-Frame per Sub-buffer* mode is enabled.			

Data exchange across the AHB interface can be configured as Little Endian (LE = 1) or Big Endian (LE = 0). The figure below provides an overview of the endian options, chosen by an ADT descriptor field.

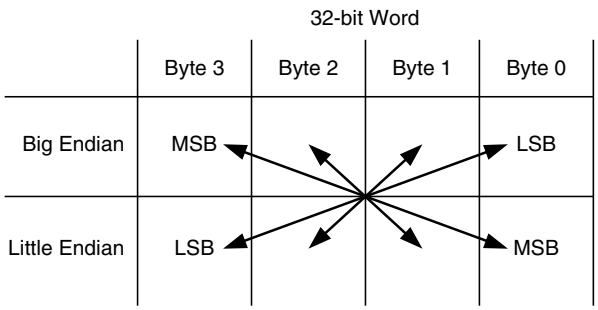


Figure 43-6. Endianness Overview

The figure below shows an example of the ping-pong system memory structure. This system memory structure is similar for all channel types and shows the relationship between the BAn, BDn, and PG descriptor fields.

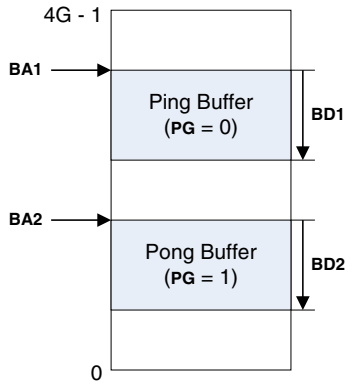


Figure 43-7. Ping-Pong System Memory Structure

Each ADT entry holds a 32-bit BAn field which defines the start of each ping or pong buffer within system memory. The BDn field is used to indicate the size for the respective ping or pong page. The maximum size is 2k-entries for asynchronous and control channels; 8k-entries for isochronous and synchronous channels.

43.4.6.2 AHB Synchronous Channel Descriptors

The table below shows the format for a synchronous ADT entry. The field definitions are defined in Table 43-15. Each synchronous channel buffer can be up to 8k-bytes deep.

Table 43-16. Synchronous ADT Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CE	LE	PG	Reserved												
16	Reserved															
32	RDY 1	DNE 1	ERR 1	BD1[12:0]												
48	RDY 2	DNE 2	ERR 2	BD2[12:0]												
64	BA1[15:0]															
80	BA1[31:16]															
96	BA2[15:0]															
112	BA2[31:16]															

43.4.6.3 AHB Isochronous Channel Descriptors

The isochronous buffering scheme allows each ping or pong buffer to contain a single block or a multiple number of blocks. For this reason, the isochronous buffer depth (BDn) must be defined in terms of an *integer number* (n) and *block size* (BS) (e.g. BDn = n x (BS + 1) - 1).

Table 43-17 shows the format for an isochronous ADT entry. The field definitions are defined in Table 43-15. Each isochronous channel buffer can be up to 8k-bytes deep.

Table 43-17. Isochronous ADT Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CE	LE	PG	Reserved												
16	Reserved															
32	RDY1	DNE1	ERR1	BD1[12:0]												
48	RDY2	DNE2	ERR2	BD2[12:0]												
64	BA1[15:0]															
80	BA1[31:16]															
96	BA2[15:0]															
112	BA2[31:16]															

43.4.6.4 AHB Asynchronous and Control Channel Descriptors

Every asynchronous and control packet adheres to the Port Message Protocol (PMP), which designates the first two bytes of each packet as the packet length (PML). Each packet must be no more than 2048 bytes.

Software must set the buffer ready bit (RDYn) for each buffer as it programs the DMA. As hardware processes each buffer, it sets the done bit (DNEn) and generates an interrupt to inform HC. When hardware finishes processing a buffer it can begin processing another buffer if RDYn is set. The application is responsible for setting up and configuring the channel buffer descriptor prior to every DMA access on the channel.

Two packet modes are supported by hardware for programming the DMA, single-packet mode and multiple-packet mode.

43.4.6.4.1 Single-packet Mode

The single-packet mode asynchronous and control buffering scheme supports a maximum of one packet per buffer (e.g. ping or pong). Both non-segmented and segmented data packets are allowed while using single-packet mode.

Non-segmented packets are exchanged when only one buffer (e.g. ping or pong) is needed for packet transfer. Segmented packets are exchanged when a single packet is too long for one buffer and the packet must span multiple buffers. The figure below shows the memory space usage for both non-segmented and segmented asynchronous or control packets along with the packet start bit (PSn). While using single-packet mode, buffer done (DNEn) is set in hardware when a packet is done or the buffer is full.

Table 43-18 shows the format for single-packet mode asynchronous and control ADT entries. The field definitions are defined in Table 43-15.

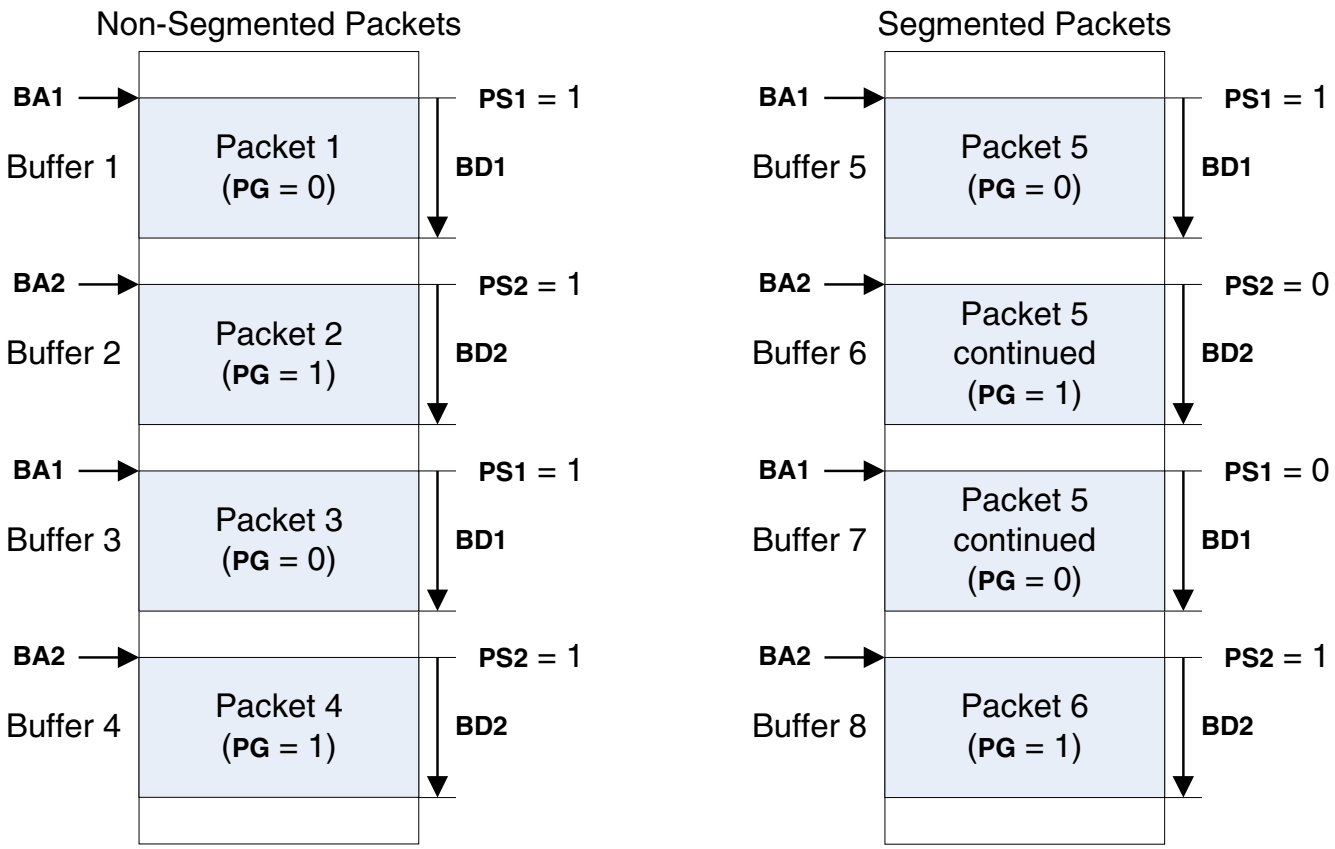


Figure 43-8. Single-packet Asynchronous or Control System Memory Structure

Table 43-18. Single-packet Asynchronous and Control Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CE	LE	PG	Reserved												
16	Reserved															
32	RDY1	DNE1	ERR1	PS1	MEP1	BD1[10:0]										
48	RDY2	DNE2	ERR2	PS2	MEP2	BD2[10:0]										
64	BA1[15:0]															
80	BA1[31:16]															

Table continues on the next page...

Table 43-18. Single-packet Asynchronous and Control Entry Format (continued)

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
96	BA2[15:0]															
112	BA2[31:16]															

43.4.6.4.2 Multiple-packet Mode

The multiple-packet mode asynchronous and control buffering scheme supports more than one packet per system memory buffer, as shown in the figure below. Multiple-packet mode reduces the interrupt rate for packet channels at the cost of increasing buffering and latency.

For Tx packet channels in multiple-packet mode, software sets the packet start bit (PSn) for every buffer. Setting PSn informs hardware that the first two bytes of the buffer contains the port message length (PML) of the first packet. After the first packet, hardware keeps track of where packets start and end within the current buffer. Software should not write to PSn while the buffer is active (RDYn = 1 and DNEn = 0). For Tx packet channels, the buffer is done (DNEn= 1) when the last byte of the last packet in the buffer is read from system memory. Software should set the buffer depth to contain the exact number of complete packets for that buffer. Segmented buffers are not supported for Tx packet channels in multiple-packet mode.

For Rx packet channels in multiple-packet mode, PSn has no meaning and should be ignored. Software is responsible for keeping track of where each packet starts and ends within the multiple-packet buffer via the packet PML. The buffer done bit (DNEn) is set in hardware for Rx channels when a buffer is full (see Buffer 1 in the figure below) or if a packet ends exactly 1-byte before the end of the buffer (see Buffer 2 in the figure below). Multiple-packet mode also supports segmented Rx packets spanning two or more buffers (see Buffers 3 - 6in the figure below).

Table 43-19 shows the format for multiple-packet mode asynchronous and control ADT entries. The field definitions are defined in Table 43-15.

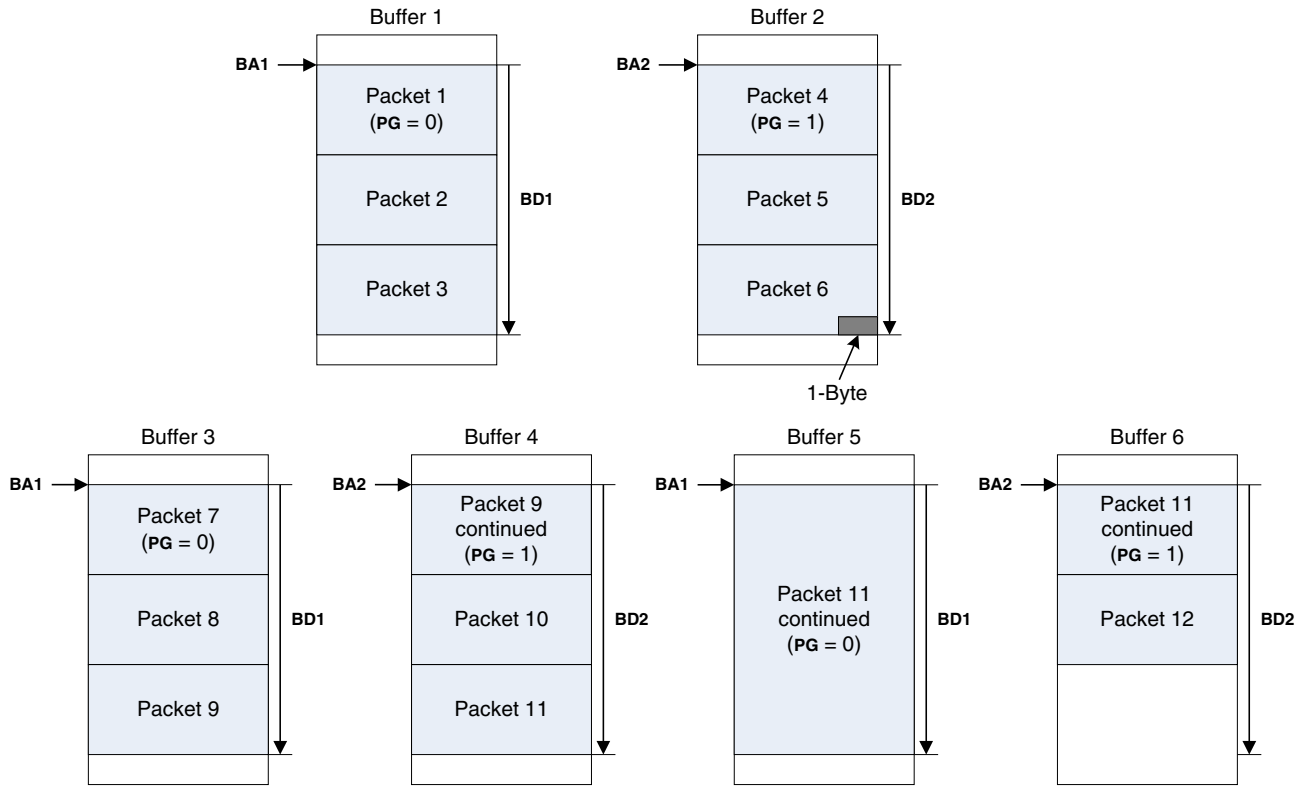


Figure 43-9. Multiple-packet Asynchronous or Control System Memory Structure

Table 43-19. Multiple-packet Asynchronous and Control Entry Format

Bit Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CE	LE	PG	Reserved												
16	Reserved															
32	RDY1	DNE1	ERR1	PS1*	BD1[11:0]											
48	RDY2	DNE2	ERR2	PS2*	BD2[11:0]											
64	BA1[15:0]															
80	BA1[31:16]															
96	BA2[15:0]															
112	BA2[31:16]															
* PSn is only valid for TX channels. Set PSn = 1 at the start of the buffer.																

43.5 Software Flow

The top-level software tasks the application must perform can be placed in two categories.

- Channel Initialization
- Channel Servicing

43.5.1 Channel Initialization

The software flow required to initialize a channel must be performed in order to ensure proper operation.

For clarity, the software flow is grouped as follows:

- Configure the Hardware
- Program the Routing Fabric Block
- Program the AMBA AHB Block DMAs
- Synchronize and Unmute Synchronous Channel

43.5.1.1 Configure the Hardware

The MLBC0, HMCR0, HMCR1 and HCTL registers are accessible directly via APB reads and writes.

1. Initialize CTR and registers
   a. Set all bit of the CTR (CAT, CDT, and ADT) to'0'.
   b. Set all bits of all registers to'0'.
2. Configure the MediaLB interface
   a. Select 3-pin MediaLB operation: MLBC0.MLBPEN = 0 (3-pin)
   b. Select MediaLB clock speed via MLBC0.MLBCLK
   c. Set MediaLB enable via MLBC0.MLBEN
3. Configure the HBI interface
   a. Set HMCR0 and HMCR1 = FFFFFFFFh to activate all channels
   b. Set the HBI enable bit: HCTL.EN = 1

43.5.1.2 Program the Routing Fabric Block

The CAT and CDT reside in the external CTR and are programmed indirectly via APB or I/O reads and writes to the MIF block.

1. Initialize all bits of the CAT to '0'
2. Select a logical channel: N = 0 - 63
3. Program the CDT for channel N
   a. Set the 14-bit base address (BA)

b. Set the 12-bit or 13-bit buffer depth (BD): BD = buffer depth in bytes -1
   - i. For synchronous channels: (BD + 1) = 4 x frames per sub-buffer (m) x bytes-per-frame (bpf)
   - ii. For isochronous channels: (BD + 1) mod (BS + 1) = 0
   - iii. For asynchronous channels: (BD + 1) >= max packet length (1024 for a MOST Data Packet (MDP); 1536 for a MOST Ethernet Packet (MEP))
   - iv. For control channels: (BD + 1) >= max packet length (64)

c. For isochronous channels, set the block size (BS): BS = block size in bytes - 1
d. Set all other bits of the CDT to '0'

4. Program the CAT for the inbound DMA
   a. For Tx channels (to MediaLB) HBI is the inbound DMA
   b. For Rx channels (from MediaLB) MediaLB is the inbound DMA
   c. Set the channel direction: RNW = 0
   d. Set the channel type: CT[2:0] = 010 (asynchronous), 001 (control), 011 (isochronous), or 000 (synchronous)
   e. Set the connection label: CL[5:0] = N
   f. If CT[2:0] = 000 (synchronous), set the mute bit (MT = 1).
   g. Set the channel enable: CE = 1
   h. Set all other bits of the CAT to '0'
5. Program the CAT for the outbound DMA
   a. For Tx channels (to MediaLB) MediaLB is the outbound DMA
   b. For Rx channels (from MediaLB) HBI is the outbound DMA
   c. Set the channel direction: RNW = 1
   d. Set the channel type: CT[2:0] = 010 (asynchronous), 001 (control), 011 (isochronous), or 000 (synchronous)
   e. Set the channel label: CL[5:0] = N
   f. If CT[2:0] = 000 (synchronous), set the mute bit (MT = 1)
   g. Set the channel enable: CE = 1
   h. Set all other bits of the CAT to '0'
6. Repeat steps 2-5 to initialize all logical channels

43.5.1.3 Program the AMBA AHB Block DMAs

The ADT resides in the external CTR and is programmed indirectly via APB reads and writes to the MIF.

1. Initialize all bits of the ADT to '0'
2. Select a logical channel: N = 0 - 63
3. Program the AMBA AHB block ping page for channel N
   a. Set the 32-bit base address (BA1)
   b. Set the 11-bit buffer depth (BD1): BD1 = buffer depth in bytes - 1

- i. For synchronous channels: (BD1 + 1) = n x frames per sub-buffer (m) x bytes-per-frame (bpf)
- ii. For isochronous channels: (BD1 + 1) mod (BS + 1) = 0
- iii. For asynchronous channels: 5 <= (BD1 + 1) <= 4096 (max packet length)
- iv. For control channels: 5 <= (BD1 + 1) <= 4096 (max packet length)

c. For asynchronous and control Tx channels set the packet start bit (PS1) iff the page contains the start of the packet
d. Clear the page done bit (DNE1)
e. Clear the error bit (ERR1)
f. Set the page ready bit (RDY1)

4. Program the AMBA AHB block pong page for channel N
   a. Set the 32-bit base address (BA2)
   b. Set the 11-bit buffer depth (BD2): BD2 = buffer depth in bytes - 1
      - i. For synchronous channels: (BD2 +1) = n x frames per sub-buffer (m) x bytes-per-frame (bpf)
      - ii. For isochronous channels: (BD2 + 1) mod (BS + 1) = 0
      - iii. For asynchronous channels: 5 <= (BD2 + 1) <= 4096 (max packet length)
      - iv. For control channels: 5 <= (BD2 + 1) <= 4096 (max packet length)

   c. For asynchronous and control Tx channels set the packet start bit (PS2) if the page contains the start of the packet
   d. Clear the page done bit (DNE2)
   e. Clear the error bit (ERR2)
   f. Set the page ready bit (RDY2)
5. Select Big Endian (LE = 0) or Little Endian (LE = 1)
6. Select the active page: PG = 0 (ping), PG = 1 (pong)
7. Set the channel enable (CE) bit for all active logical channels
8. Repeat steps 2-7 for all active logical channels

NOTE

All asynchronous and control packets must start with a PMP header. The first two bytes of the PMP header contains the Port Message Length (PML), which defines the length of the message that follows in bytes (not including PML itself). Hardware uses the PML to determine when a packet is complete. Asynchronous and control packets can also be segmented into two or more pages as well as contain multiple packets per page within system memory.

43.5.1.4 Synchronize and Unmute Synchronous Channel

The MLBC0 and MLBC1 registers are accessible directly via APB reads and writes.

1. Check that MediaLB clock is running (MLBC1.CLKM = 0)
2. If MLBC1.CLKM = 1, clear the register bit, wait one APB or I/O clock cycle and repeat step 1.
3. Poll for MediaLB lock (MLBC0.MLBLK = 1)
4. Wait four frames
5. Unmute synchronous channel(s)

43.5.2 Channel Servicing

After initialization, each channel will require periodic servicing.

The following software flows can be performed concurrently and in any order:

- Servicing the AMBA AHB Block (DMA) Interrupts
- Servicing the MediaLB Interrupts
- Polling for MediaLB System Commands

43.5.2.1 Servicing the AMBA AHB Block (DMA) Interrupts

The ACMR0, ACMR1, ACTL, ACSR0, and ACSR1 registers are accessible directly via APB reads and writes.

1. Program the ACMRn registers to enable interrupts from all active DMA channels
2. Select the status clear method: SCTL.SCE = 0 (hardware clears on read), SCTL.SCE = 1 (software writes a '1' to clear)
3. Select 1 or 2 interrupt signals: SCTL.SMX = 0 (one interrupt for channels 0-31 on *ahb_int[0]* and another interrupt for channels 32-63 on *ahb_int[1]*), SCTL.SMX = 1 (single interrupt for all channels on *ahb_int[0]*)
4. Wait for an interrupt from *ahb_int[1:0]*
5. Read the ACSRn registers to determine which channel or channels are causing the interrupt
6. If SCTL.SCE = 1, write the results of step 5 back to ACSR0 and ACSR1 to clear the interrupt
7. Select a logical channel (N = 0-63) with an interrupt to service
8. Read the ADT entry for channel N
    a. Determine the active page (ping or pong) via the PG bit

b. Determine which page(s) are done via the DNEn bits
c. Determine which channels encountered an AHB error via the ERRn bit
d. Determine which asynchronous and control Rx channel pages contain a packet start via the PSn bit (extract the PML)

9. Reprogram the expired or broken AHB page(s) via steps 3 and 4 in Program the AMBA AHB Block DMAs,
10. Repeat steps 6-9 for all channels with pending interrupts
11. Repeat steps 4-10 while there are active channels

NOTE

Channels that receive an AHB error response are disabled (CE = 0) by hardware.

43.5.2.2 Servicing the MediaLB Interrupts

1. Select the MediaLB Channel Status Register (MSn) to be cleared by software, writing a '0' to the appropriate bits
2. Program MIEN to enable protocol error interrupts for all active MediaLB channels (MIEN.CTX_PE = 1, MIEN.CRX_PE = 1, MIEN.ATX_PE = 1, MIEN.ARX_PE = 1, MIEN.SYNC_PE = 1, and MIEN.ISOC_PE = 1)
3. Wait for an interrupt on the *mlb_int* signal.
4. Read the MSn registers to determine which channel(s) are causing the interrupt
5. Read RSTS/WSTS of the appropriate CDT(s) to determine the interrupt type
6. Clear RSTS/WSTS errors to resume channel operation
   a. For synchronous channels: WSTS[3] = 0
   b. For isochronous channels: WSTS[2:1] = 00
   c. For asynchronous and control channels: RSTS[4]/WSTS[4] = 0 and RSTS[2]/WSTS[2] = 0

43.5.2.3 Polling for MediaLB System Commands

The MLB supports the MediaLB System Commands (e.g. MlbScan, MlbReset, MOST_Unlock). The MediaLB System Status (MSS) Register is used to detect a System Command received from the MediaLB Controller. The MLB automatically sends the appropriate system response to the MediaLB Controller.

The procedure for the application is:

1. The application periodically polls the MSS register.
2. Clear by writing a '0' to the appropriate bit in MSS register after the application finishes the service.

3. If MSS.SWSYSCMD = 1, read the MSD register to receive the system data sent from MediaLB Controller.

43.5.3 Low Power Mode

MLB doesn't provide dedicated low power mode features.

In case the clocks of digital IP need to shut down to save power, the following operations are recommended before entering low power mode:

- Finish any active MLB transfer
- Disable MLB (clear the MLBEN and MLBPEN bits in MLBC0)
- Disable HBI (clear all bits in HCMR0 and HCMR1, clear EN bit in HCTL)
- Mask AHB interrupts (clear all bits in ACMR0 and ACMR1)

For information on configuring the MLB IP if the clocks are re-enabled, see Configure the Hardware.

43.6 MLB Memory Map/Register Definition

The MLB registers are divided into 4 sections.

The first section begins at offset 0h00 and describes the MediaLB block registers.

The second section begins at offset 0h80 and it shows the address mapping of the Internal HBI Registers.

The third set begins at offset 0hC0 and implements ten 32-bit I/O registers for CTR transfers, including: data registers (MDATn), write enable registers (MDWEn), a control register (MCTL), and an address register (MADR).

The fourth set, the AMBA AHB registers, begins at offset 0h3C0. They consist of:

- one 32-bit register for control
- two 32-bit registers for interrupt status
- two 32-bit registers for channel interrupt masks

MLB memory map

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_C000	MediaLB Control 0 Register (MLB_MLBC0)	32	R/W	0000_0000h	43.6.1/3808
218_C00C	MediaLB Channel Status 0 Register (MLB_MS0)	32	R	0000_0000h	43.6.2/3810
218_C00D	MediaLB 6-pin Control 2 Register (MLB_MLBPC2)	32	R	0000_0000h	43.6.3/3811
218_C014	MediaLB Channel Status1 Register (MLB_MS1)	32	R	0000_0000h	43.6.4/3811
218_C020	MediaLB System Status Register (MLB_MSS)	32	R	0000_0000h	43.6.5/3812
218_C024	MediaLB System Data Register (MLB_MSD)	32	R	0000_0000h	43.6.6/3813
218_C02C	MediaLB Interrupt Enable Register (MLB_MIEN)	32	R/W	0000_0000h	43.6.7/3814
218_C03C	MediaLB Control 1 Register (MLB_MLBC1)	32	R	0000_0000h	43.6.8/3815
218_C080	HBI Control Register (MLB_HCTL)	32	R/W	0000_0000h	43.6.9/3816
218_C088	HBI Channel Mask 0 Register (MLB_HCMR0)	32	R/W	0000_0000h	43.6.10/ 3817
218_C08C	HBI Channel Mask 1 Register (MLB_HCMR1)	32	R/W	0000_0000h	43.6.11/ 3817
218_C090	HBI Channel Error 0 Register (MLB_HCER0)	32	R	0000_0000h	43.6.12/ 3818
218_C094	HBI Channel Error 1 Register (MLB_HCER1)	32	R	0000_0000h	43.6.13/ 3818
218_C098	HBI Channel Busy 0 Register (MLB_HCBR0)	32	R	0000_0000h	43.6.14/ 3819
218_C09C	HBI Channel Busy 1 Register (MLB_HCBR1)	32	R	0000_0000h	43.6.15/ 3819
218_C0C0	MIF Data 0 Register (MLB_MDAT0)	32	R/W	0000_0000h	43.6.16/ 3820
218_C0C4	MIF Data 1 Register (MLB_MDAT1)	32	R/W	0000_0000h	43.6.17/ 3820
218_C0C8	MIF Data 2 Register (MLB_MDAT2)	32	R/W	0000_0000h	43.6.18/ 3820
218_C0CC	MIF Data 3 Register (MLB_MDAT3)	32	R/W	0000_0000h	43.6.19/ 3821
218_C0D0	MIF Data Write Enable 0 Register (MLB_MDWE0)	32	R/W	0000_0000h	43.6.20/ 3821
218_C0D4	MIF Data Write Enable 1 Register (MLB_MDWE1)	32	R/W	0000_0000h	43.6.21/ 3821
218_C0D8	MIF Data Write Enable 2 Register (MLB_MDWE2)	32	R/W	0000_0000h	43.6.22/ 3822
218_C0DC	MIF Data Write Enable 3 Register (MLB_MDWE3)	32	R/W	0000_0000h	43.6.23/ 3822
218_C0E0	MIF Control Register (MLB_MCTL)	32	R	0000_0000h	43.6.24/ 3823
218_C0E4	MIF Address Register (MLB_MADR)	32	R/W	0000_0000h	43.6.25/ 3823
218_C3C0	AHB Control Register (MLB_ACTL)	32	R/W	0000_0000h	43.6.26/ 3824

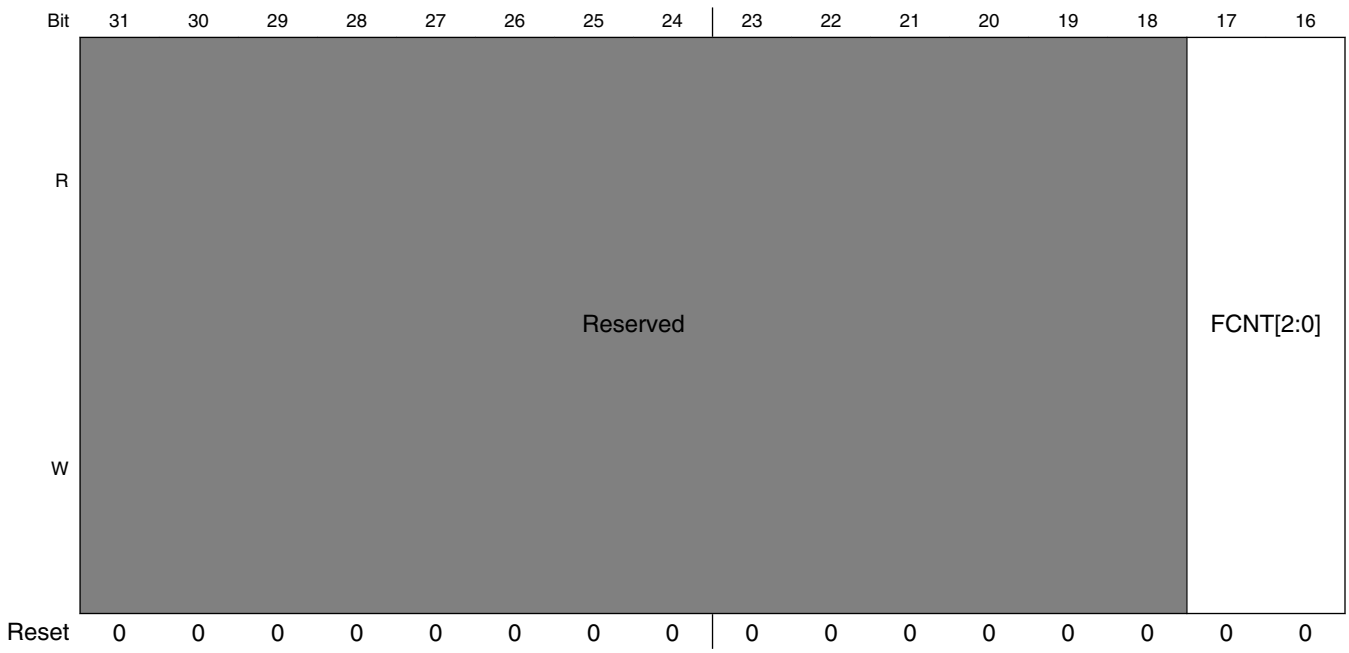
Table continues on the next page...

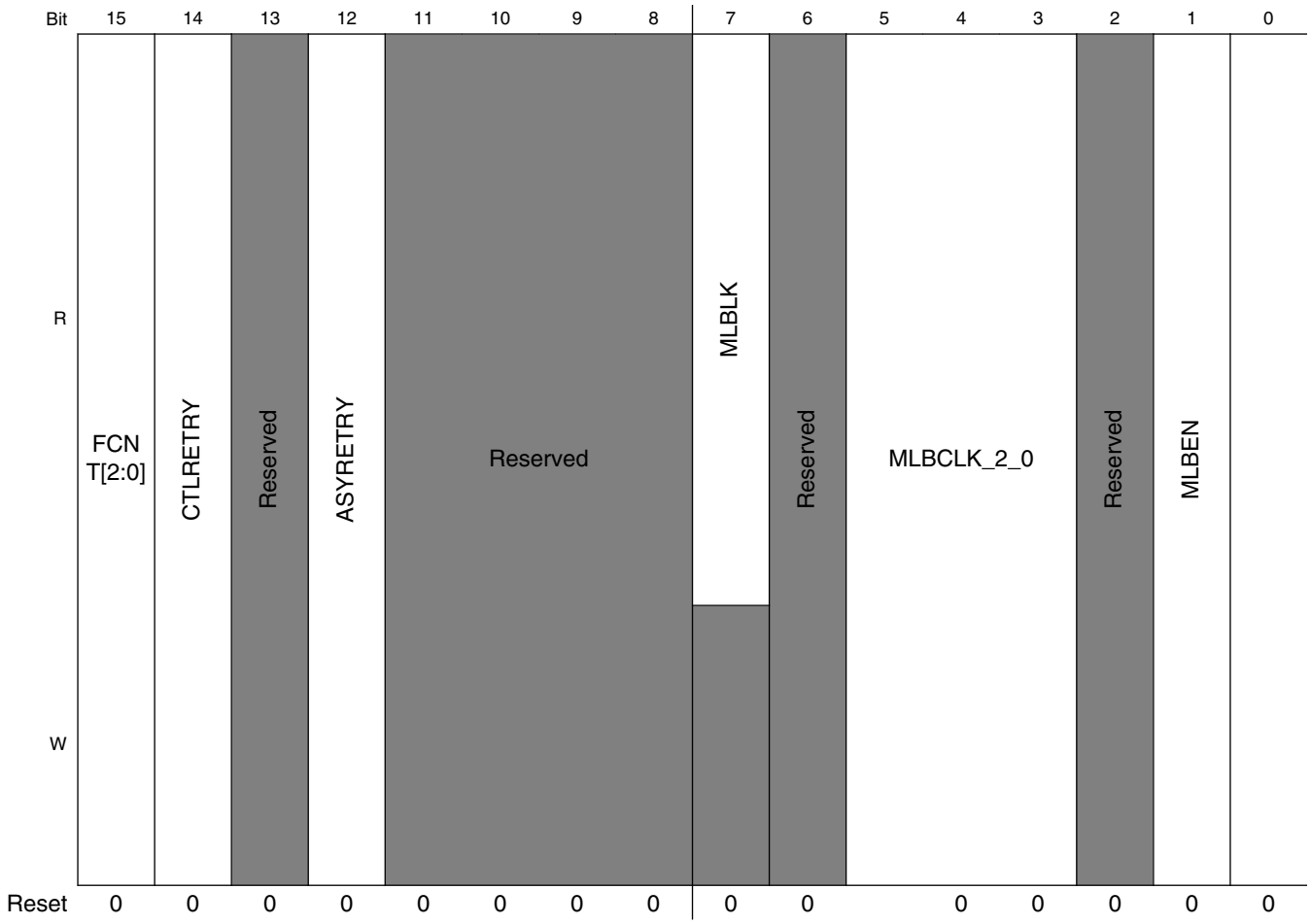
MLB memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
218_C3D0	AHB Channel Status 0 Register (MLB_ACSR0)	32	R	0000_0000h	43.6.27/ 3825
218_C3D4	AHB Channel Status 1 Register (MLB_ACSR1)	32	R	0000_0000h	43.6.28/ 3826
218_C3D8	AHB Channel Mask 0 Register (MLB_ACMR0)	32	R/W	0000_0000h	43.6.29/ 3826
218_C3DC	AHB Channel Mask 1 Register (MLB_ACMR1)	32	R/W	0000_0000h	43.6.30/ 3827

43.6.1 MediaLB Control 0 Register (MLB_MLBC0)

Address: 218_C000h base + 0h offset = 218_C000h





MLB_MLBC0 field descriptions

Field	Description
31–18 -	This field is reserved. Reserved
17–15 FCNT[2:0]	The number of frames per sub-buffer for synchronous channels. 000 1 frame per sub-buffer (Operation is the same as Standard mode.) 001 2 frames per sub-buffer 010 4 frames per sub-buffer 011 8 frames per sub-buffer 100 16 frames per sub-buffer 101 32 frames per sub-buffer 110 64 frames per sub-buffer 111 Reserved
14 CTLRETRY	Control Tx packet retry. When set, a control packet that is flagged with a Break or ProtocolError by the receiver is retransmitted. When cleared, a control packet that is flagged with a Break or ProtocolError by the receiver is skipped.
13 -	This field is reserved. Reserved

Table continues on the next page...

MLB_MLBC0 field descriptions (continued)

Field	Description
12 ASYRETRY	Asynchronous Tx packet retry. When set, an asynchronous packet that is flagged with a Break or ProtocolError by the receiver is retransmitted. When cleared, an asynchronous packet that is flagged with a Break or ProtocolError by the receiver is skipped.
11–8 -	This field is reserved. Reserved
7 MLBLK	MediaLB lock status. When set, indicates that the MediaLB block is synchronized to the incoming MediaLB frame. If **MLBLK** is clear (unlocked), **MLBLK** is set after FRAMESYNC is detected at the same position for three consecutive frames. If **MLBLK** is set (locked), **MLBLK** is cleared after not receiving FRAMESYNC at the expected time for two consecutive frames. While **MLBLK** is set, FRAMESYNC patterns occurring at locations other than the expected one are ignored. (read-only)
6 -	This field is reserved. Reserved
4–2 MLBCLK_2_0	MLB_CLK (MediaLB clock) speed select. 000 256xFs (for **MLBPEN** = 0) 001 512xFs (for **MLBPEN** = 0) 010 1024xFs (for **MLBPEN** = 0)
1 -	This field is reserved. Reserved
0 MLBEN	MediaLB enable. When set, MLB_CLK (MediaLB clock), MLB_SIG (signal), and MLB_DATA (data) are received and transmitted on the appropriate MediaLB pins.

43.6.2 MediaLB Channel Status 0 Register (MLB_MS0)

Address: 218_C000h base + Ch offset = 218_C00Ch

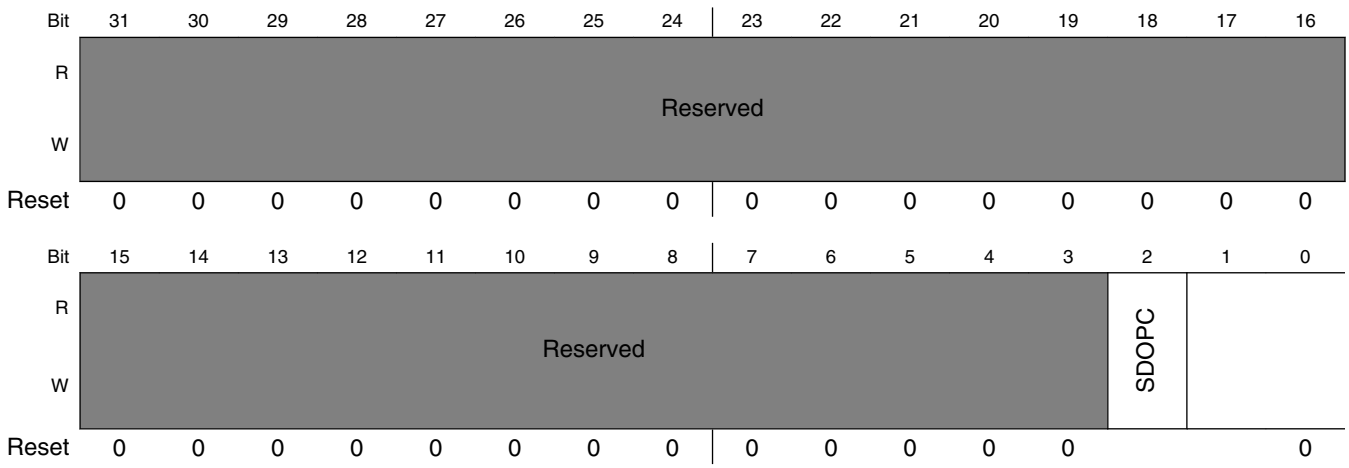
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	MCS_31_0																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MS0 field descriptions

Field	Description
MCS_31_0	MediaLB channel status. Indicates the channel status for MediaLB channels 31 to 0. Channel status bits are set by hardware and cleared by software. Status is only set if the appropriate bits in the MIEN register are set.

43.6.3 MediaLB 6-pin Control 2 Register (MLB_MLBPC2)

Address: 218_C000h base + Dh offset = 218_C00Dh

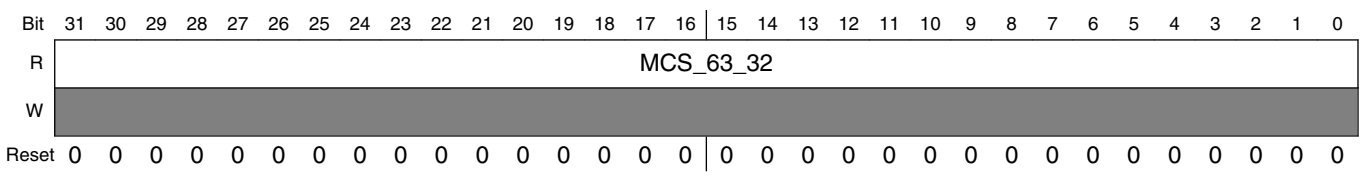


MLB_MLBPC2 field descriptions

Field	Description
31–3 -	This field is reserved. Reserved.
0 SDOPC	MLB 3-pin interface: Signal/Data output phase control. 0 MLB_SIG / MLB_DATA launch at rising edge of MLB_CLK(default) 1 MLB_SIG / MLB_DATA launch at falling edge of MLB_CLK

43.6.4 MediaLB Channel Status1 Register (MLB_MS1)

Address: 218_C000h base + 14h offset = 218_C014h

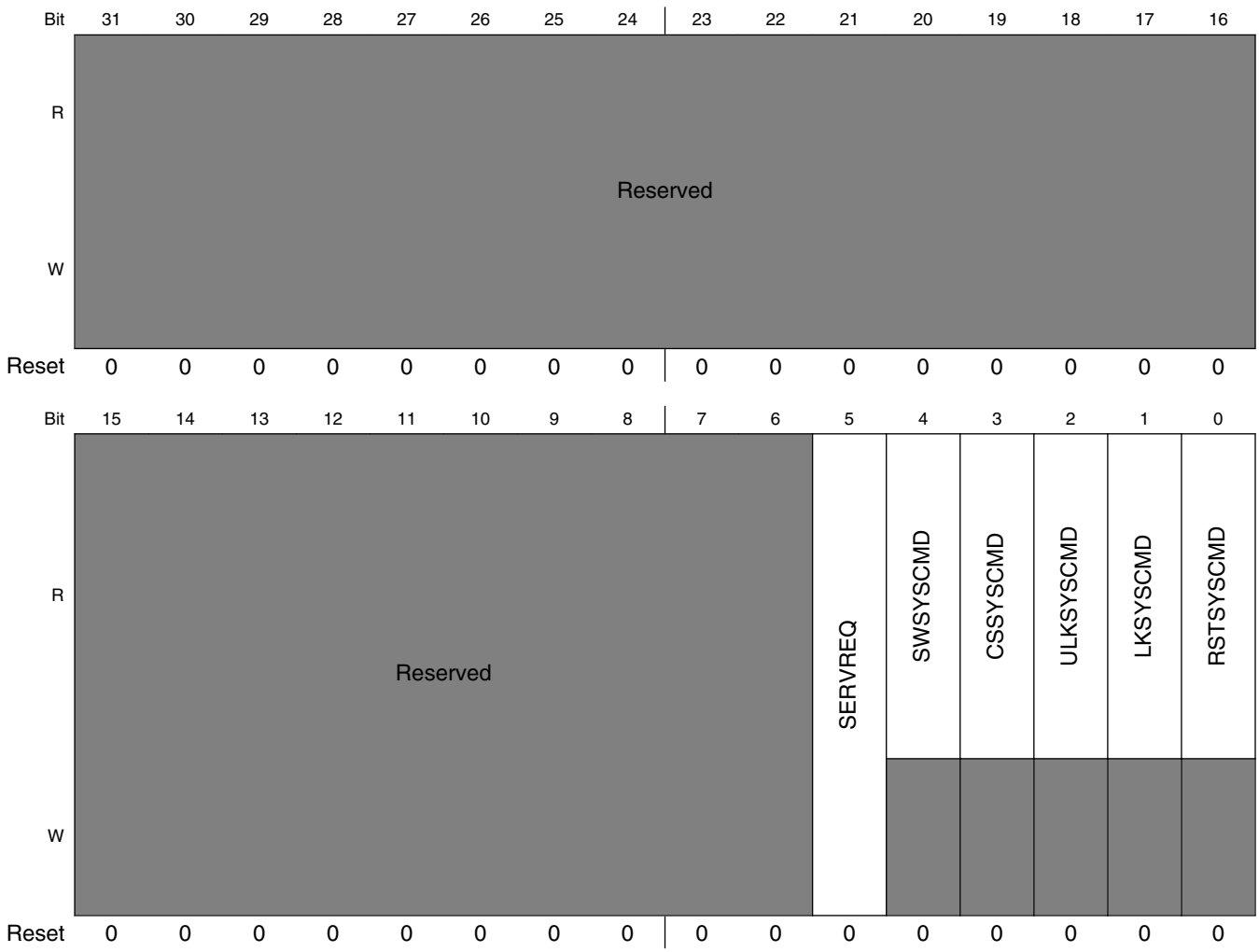


MLB_MS1 field descriptions

Field	Description
MCS_63_32	MediaLB channel status. Indicates the channel status for MediaLB channels 63 to 32. Channel status bits are set by hardware and cleared by software. Status is only set if the appropriate bits in the MIEN register are set.

43.6.5 MediaLB System Status Register (MLB_MSS)

Address: 218_C000h base + 20h offset = 218_C020h



MLB_MSS field descriptions

Field	Description
31–6 -	This field is reserved. Reserved
5 SERVREQ	Service request enabled. When set, the MediaLB block responds with a "device present, request service" system response if a matching channel scan system command is detected. When cleared, the MediaLB block responds with a "device present" system response.
4 SWSYSCMD	Software system command detected (in the system quadlet). Set by hardware, cleared by software. Data is stored in the MSD register for this command.
3 CSSYSCMD	Channel scan system command detected (in the system quadlet). Set by hardware, cleared by software. If the node address specified in *Data* quadlet matches the value in **MLBC1.NDA**, the device responds either "device present" or "device present, request service" system response in the next system quadlet.

Table continues on the next page...

MLB_MSS field descriptions (continued)

Field	Description
2 ULKSYSCMD	Network unlock system command detected (in the system quadlet). Set by hardware, cleared by software.
1 LKSYSCMD	Network lock system command detected (in the system quadlet). Set by hardware, cleared by software.
0 RSTSYSCMD	Reset system command detected (in the system quadlet). Set by hardware, cleared by software.

43.6.6 MediaLB System Data Register (MLB_MSD)

Address: 218_C000h base + 24h offset = 218_C024h

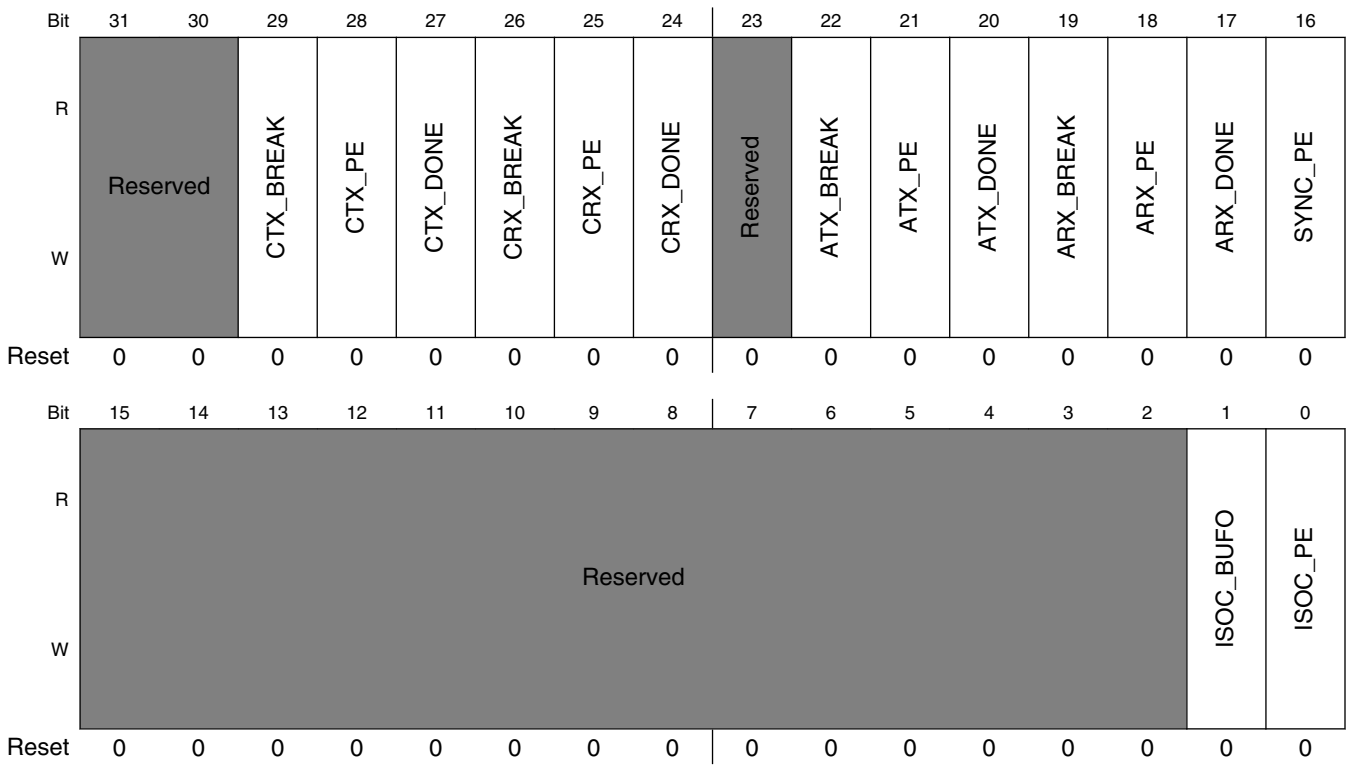
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	SD3_7_0								SD2_7_0								SD1_7_0								SD0_7_0							
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MSD field descriptions

Field	Description
31–24 SD3_7_0	System data (byte 3). Updated with MediaLB Data[31:24] when a MediaLB software system command is received in the system quadlet. If MSS.SWSYSCMD is already set, then SD3 is not updated. (read-only)
23–16 SD2_7_0	System data (byte 2). Updated with MediaLB Data[23:16] when a MediaLB software system command is received in the system quadlet. If MSS.SWSYSCMD is already set, then SD2 is not updated. (read-only)
15–8 SD1_7_0	System data (byte 1). Updated with MediaLB Data[15:8] when a MediaLB software system command is received in the system quadlet. If MSS.SWSYSCMD is already set, then SD1 is not updated. (read-only)
SD0_7_0	System data (byte 0). Updated with MediaLB Data[7:0] when a MediaLB software system command is received in the system quadlet. If MSS.SWSYSCMD is already set, then SD0 is not updated. (read-only)

43.6.7 MediaLB Interrupt Enable Register (MLB_MIEN)

Address: 218_C000h base + 2Ch offset = 218_C02Ch



MLB_MIEN field descriptions

Field	Description
31–30 -	This field is reserved. Reserved
29 CTX_BREAK	Control Tx break enable. When set, a *ReceiverBreak* response received from the receiver on a control Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
28 CTX_PE	Control Tx protocol error enable. When set, a *ProtocolError* generated by the receiver on a control Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
27 CTX_DONE	Control Tx packet done enable. When set, a packet transmitted with no errors on a control Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
26 CRX_BREAK	Control Rx break enable. When set, a *ControlBreak* command received from the transmitter on a control Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
25 CRX_PE	Control Rx protocol error enable. When set, a *ProtocolError* detected on a control Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
24 CRX_DONE	Control Rx packet done enable. When set, a packet received with no errors on a control Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
23 -	This field is reserved. Reserved
22 ATX_BREAK	Asynchronous Tx break enable. When set, a *ReceiverBreak* response received from the receiver on an asynchronous Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.

Table continues on the next page...

MLB_MIEN field descriptions (continued)

Field	Description
21 ATX_PE	Asynchronous Tx protocol error enable. When set, a *ProtocolError* generated by the receiver on an asynchronous Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
20 ATX_DONE	Asynchronous Tx packet done enable. When set, a packet transmitted with no errors on an asynchronous Tx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
19 ARX_BREAK	Asynchronous Rx break enable. When set, a *AsyncBreak* command received from the transmitter on an asynchronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
18 ARX_PE	Asynchronous Rx protocol error enable. When set, a *ProtocolError* detected on an asynchronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
17 ARX_DONE	Asynchronous Rx done enable. When set, a packet received with no errors on an asynchronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
16 SYNC_PE	Synchronous protocol error enable. When set, a *ProtocolError* detected on a synchronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.
15–2 -	This field is reserved. Reserved
1 ISOC_BUFO	Isochronous Rx buffer overflow enable. When set, a buffer overflow on an isochronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set. This occurs only when isochronous flow control is disabled.
0 ISOC_PE	Isochronous Rx protocol error enable. When set, a *ProtocolError* detected on an isochronous Rx channel causes the appropriate channel bit in the MS0 or MS1 registers to be set.

43.6.8 MediaLB Control 1 Register (MLB_MLBC1)

Address: 218_C000h base + 3Ch offset = 218_C03Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	Reserved															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	NDA_7_0								CLKM	LOCK	Reserved					
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MLBC1 field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
15–8 NDA_7_0	Node device address. Used for system commands directed to individual MediaLB nodes.
7 CLKM	MediaLB clock missing status. Set when MLB_CLK (MediaLB clock) is not toggling at the pin; cleared by software.
6 LOCK	MediaLB lock error status. Set when MediaLB is unlocked; cleared by software.
-	This field is reserved. Reserved

43.6.9 HBI Control Register (MLB_HCTL)

The HC can control and monitor general operation of the HBI block by reading and writing the HBI Control Register (HCTL) through the I/O interface. Each bit of HCTL is read/write.

Address: 218_C000h base + 80h offset = 218_C080h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	EN	Reserved													RST1	RST0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_HCTL field descriptions

Field	Description
31–16 -	This field is reserved. Reserved
15 EN	HBI enable 1 enabled 0 disabled
14–2 -	This field is reserved. Reserved
1 RST1	AGU1 software reset 1 reset 0 active

Table continues on the next page...

MLB_HCTL field descriptions (continued)

Field	Description
0 RST0	AGU0 software reset 1 reset 0 active

43.6.10 HBI Channel Mask 0 Register (MLB_HCMR0)

The HC can control which channel(s) are able to generate an HBI interrupt by writing the HBI Channel Mask Registers (HCMRn). The HCMRn registers mask the channel interrupt on the *hbi_hintb* signal (i.e. *hbi_hintb* will not become active for any masked channel). Each bit of HCMRn is read/write.

Address: 218_C000h base + 88h offset = 218_C088h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CHM_31_0_P																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_HCMR0 field descriptions

Field	Description
CHM_31_0_P	Bitwise channel mask bit 0 masked 1 unmasked

43.6.11 HBI Channel Mask 1 Register (MLB_HCMR1)

Address: 218_C000h base + 8Ch offset = 218_C08Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CHM_63_32																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

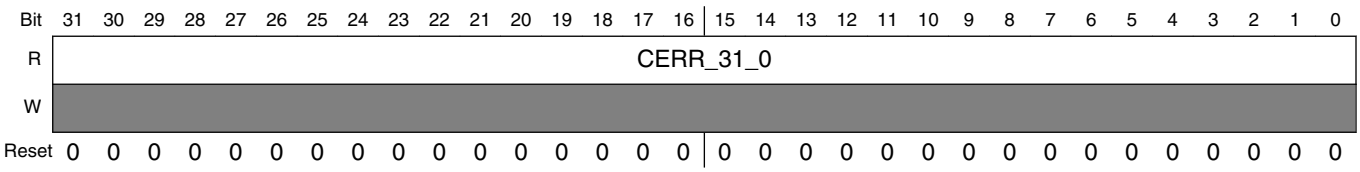
MLB_HCMR1 field descriptions

Field	Description
CHM_63_32	Bitwise channel mask bit 0 masked 1 unmasked

43.6.12 HBI Channel Error 0 Register (MLB_HCER0)

The HBI Channel Error Registers (HCERn) indicate which channel(s) have encountered fatal errors.

Address: 218_C000h base + 90h offset = 218_C090h



MLB_HCER0 field descriptions

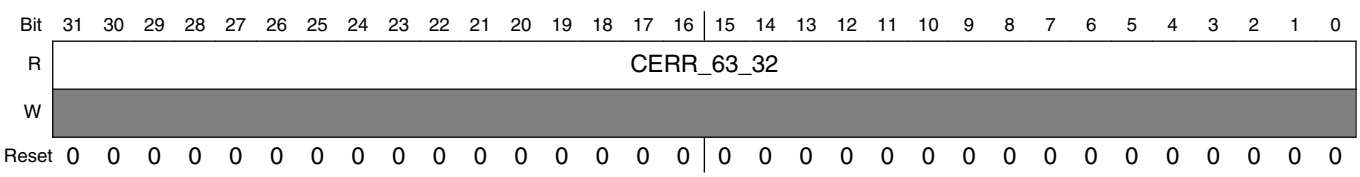
Field	Description
CERR_31_0	Bitwise channel error bit

43.6.13 HBI Channel Error 1 Register (MLB_HCER1)

HCERn status bits are set when hardware detects hardware errors on the given logical channel, including:

- Channel opened, but not enabled,
- Channel programmed with invalid channel type, or
- Out-of-range PML for asynchronous or control Tx channels

Address: 218_C000h base + 94h offset = 218_C094h



MLB_HCER1 field descriptions

Field	Description
CERR_63_32	Bitwise channel error bit

43.6.14 HBI Channel Busy 0 Register (MLB_HCBR0)

The HC can determine which channel(s) are busy by reading the HBI Channel Busy Registers (HCBRn). An HBI channel is busy if:

- it is currently loaded into one of the two AGUs
- the channel is enabled, CE = 1 from the Channel Allocation Table (Table 43-5), and
- the DMA is active

When an HBI channel is busy, hardware may write back its local copy of the channel descriptor at any time. System software should not write a CDT descriptor for a channel that is busy. Only two HBI channels can be busy at any given time. Each bit of HCBRn is read-only.

Address: 218_C000h base + 98h offset = 218_C098h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CHB_31_0																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_HCBR0 field descriptions

Field	Description
CHB_31_0	Bitwise channel busy bit 0 idle 1 busy

43.6.15 HBI Channel Busy 1 Register (MLB_HCBR1)

Address: 218_C000h base + 9Ch offset = 218_C09Ch

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CHB_63_32																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_HCBR1 field descriptions

Field	Description
CHB_63_32	Bitwise channel busy bit 0 idle 1 busy

43.6.16 MIF Data 0 Register (MLB_MDAT0)

Address: 218_C000h base + C0h offset = 218_C0C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA_31_0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDAT0 field descriptions

Field	Description
DATA_31_0	CTR data - bits[31:0] of 128-bit entry or DBR data - bits[7:0] of 8-bit entry

43.6.17 MIF Data 1 Register (MLB_MDAT1)

Address: 218_C000h base + C4h offset = 218_C0C4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA_63_32																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDAT1 field descriptions

Field	Description
DATA_63_32	CTR data - bits[63:32] of 128-bit entry

43.6.18 MIF Data 2 Register (MLB_MDAT2)

Address: 218_C000h base + C8h offset = 218_C0C8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA_95_64																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDAT2 field descriptions

Field	Description
DATA_95_64	CTR data - bits[95:64] of 128-bit entry

43.6.19 MIF Data 3 Register (MLB_MDAT3)

Address: 218_C000h base + CCh offset = 218_C0CCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	DATA_127_96																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDAT3 field descriptions

Field	Description
DATA_127_96	CTR data - bits[127:96] of 128-bit entry

43.6.20 MIF Data Write Enable 0 Register (MLB_MDWE0)

Address: 218_C000h base + D0h offset = 218_C0D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MASK_31_0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDWE0 field descriptions

Field	Description
MASK_31_0	Bitwise write enable for CTR data - bits[31:0] 0 disabled 1 enabled

43.6.21 MIF Data Write Enable 1 Register (MLB_MDWE1)

Address: 218_C000h base + D4h offset = 218_C0D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MASK_63_32																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDWE1 field descriptions

Field	Description
MASK_63_32	Bitwise write enable for CTR data - bits[63:32] 0 disabled 1 enabled

43.6.22 MIF Data Write Enable 2 Register (MLB_MDWE2)

Address: 218_C000h base + D8h offset = 218_C0D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MASK_95_64																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDWE2 field descriptions

Field	Description
MASK_95_64	Bitwise write enable for CTR data - bits[95:64] 0 disabled 1 enabled

43.6.23 MIF Data Write Enable 3 Register (MLB_MDWE3)

Address: 218_C000h base + DCh offset = 218_C0DCh

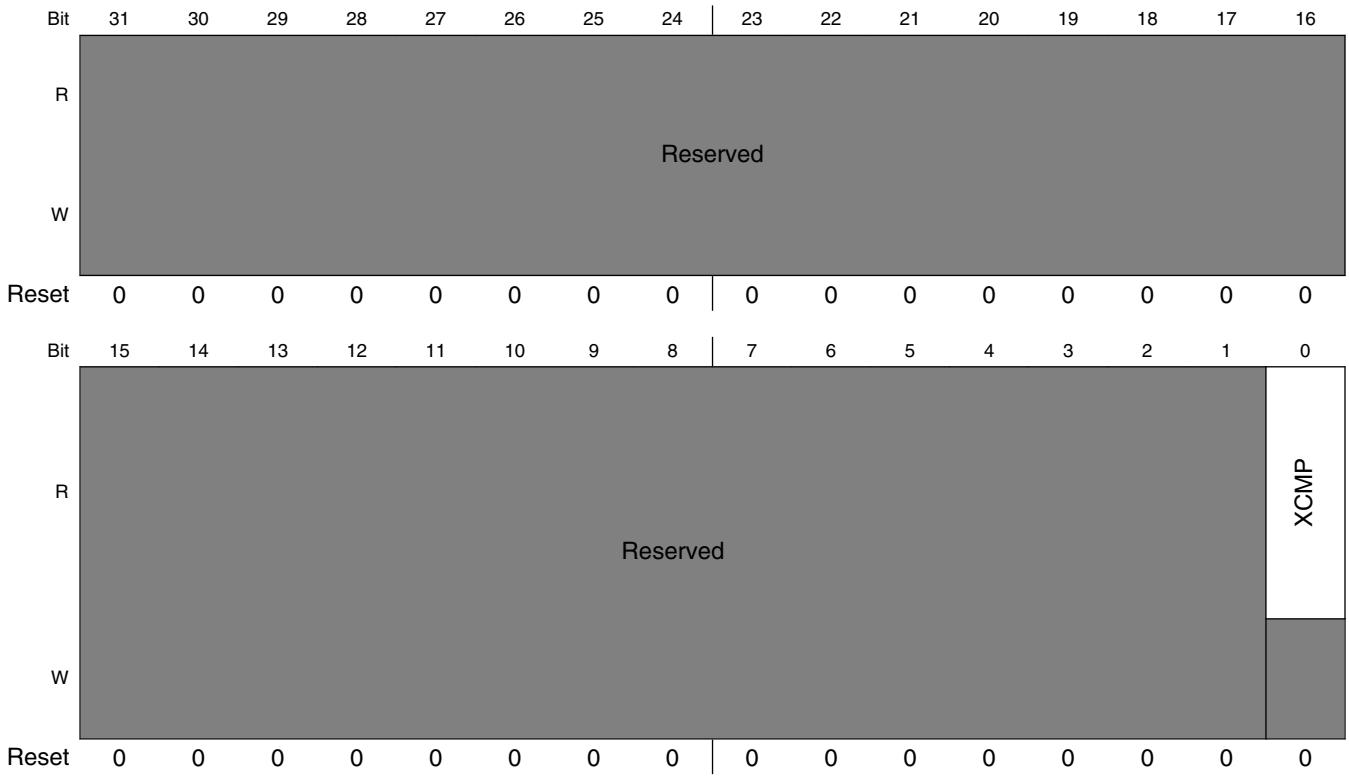
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	MASK_127_96																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_MDWE3 field descriptions

Field	Description
MASK_127_96	Bitwise write enable for CTR data - bits[127:96] 0 disabled 1 enabled

43.6.24 MIF Control Register (MLB_MCTL)

Address: 218_C000h base + E0h offset = 218_C0E0h

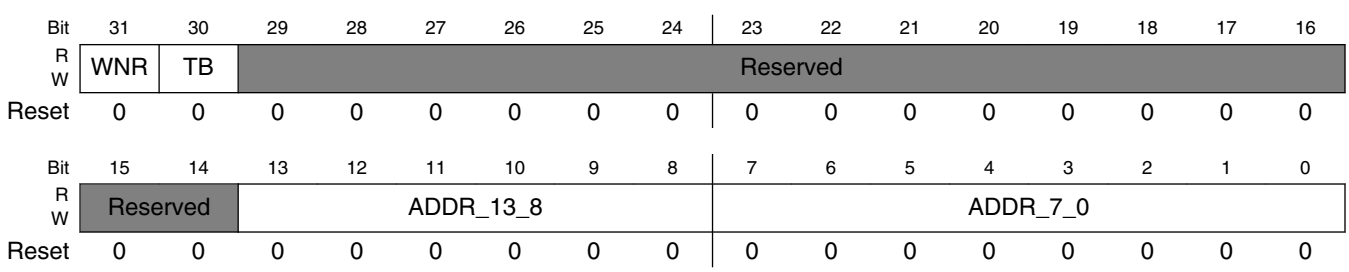


MLB_MCTL field descriptions

Field	Description
31–1 -	This field is reserved. Reserved
0 XCMP	Transfer complete (write 0 to clear)

43.6.25 MIF Address Register (MLB_MADR)

Address: 218_C000h base + E4h offset = 218_C0E4h



MLB_MADR field descriptions

Field	Description
31 WNR	Write-Not-Read selection 0 read 1 write
30 TB	Target location bit 0 selects CTR 1 selects DBR
29–14 -	This field is reserved. Reserved
13–8 ADDR_13_8	DBR address of 8-bit entry - bits[13:8]
ADDR_7_0	CTR address of 128-bit entry or DBR address of 8-bit entry - bits[7:0]

43.6.26 AHB Control Register (MLB_ACTL)

The AHB Control (ACTL) register is written by the HC to configure the AMBA AHB block for channel interrupts. ACTL contains three configuration fields, one is used to select the DMA mode, one is used to mux channel interrupts onto a single interrupt signal, and the last selects the method of clearing channel interrupts (either software or hardware).

Address: 218_C000h base + 3C0h offset = 218_C3C0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R W	Reserved															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	Reserved											MPB	-	DMA_MODE	SMX	SCE
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_ACTL field descriptions

Field	Description
31–5 -	This field is reserved. Reserved
4 MPB	DMA Packet buffering mode. 0 Single-packet mode 1 Multiple-packet mode
3 -	Reserved.
2 DMA_MODE	DMA Mode: 0 DMA Mode 0 1 DMA Mode 1
1 SMX	AHB interrupt mux enable: 0 ACSR0 generates an interrupt on *ahb_int[0]*; ACSR1 generates an interrupt on *ahb_int[1]* 1 ACSR0 and ACSR1 generate an interrupts on *ahb_int[0]* only
0 SCE	Software clear enable: 0 Hardware clears interrupt after a ACSRn register read 1 Software clears interrupt

43.6.27 AHB Channel Status 0 Register (MLB_ACSR0)

The AHB Channel Status (ACSRn) registers contain interrupt bits for each of the 64 physical channels. When an ACSRn register bit is set, it indicates that the corresponding physical channel has an interrupt pending.

An AHB interrupt is triggered when either DNEn or ERRn is set within the AHB Channel Descriptor. The HC is notified of the channel interrupt via *ahb_int[1:0]*. When an interrupt occurs in ACSR0 (for channels 31 to 0) *ahb_int[0]* is set. When an interrupt occurs in ACSR1 (for channels 63 to 32) *ahb_int[1]* is set.

Interrupts in ACSR0 and ACSR1 can be optionally muxed onto a single interrupt signal, *ahb_int[0]*, if ACTL.SMX = 1. If ACTL.SCE = 0, hardware automatically clears the interrupt bit(s) after the HC reads the ACSRn register. Alternatively, if ACTL.SCE = 1, software must write a 1 to the appropriate bit(s) of ACSRn to clear the interrupt(s).

Address: 218_C000h base + 3D0h offset = 218_C3D0h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CHS																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_ACSR0 field descriptions

Field	Description
CHS	Interrupt status for logical channels 31 to 0: 0 None 1 Interrupt

43.6.28 AHB Channel Status 1 Register (MLB_ACSR1)

Address: 218_C000h base + 3D4h offset = 218_C3D4h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	CHS[63:32]																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_ACSR1 field descriptions

Field	Description
CHS[63:32]	Interrupt status for logical channels 63 to 32: 0 None 1 Interrupt

43.6.29 AHB Channel Mask 0 Register (MLB_ACMR0)

Using the AHB Channel Mask (ACMRn) register, the HC can control which channel(s) generate interrupts on *ahb_int[1:0]*. All ACMRn register bits default as '0' ("masked"); therefore, the HC must initially write ACMRn to enable interrupts. Each bit of ACMRn is read/write accessible.

Address: 218_C000h base + 3D8h offset = 218_C3D8h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CHM_31_0																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_ACMR0 field descriptions

Field	Description
CHM_31_0	Bitwise channel mask bit:

MLB_ACMR0 field descriptions (continued)

Field	Description
	0 Masked 1 Unmasked

43.6.30 AHB Channel Mask 1 Register (MLB_ACMR1)

Address: 218_C000h base + 3DCh offset = 218_C3DCh

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R W	CHM[63:32]																															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MLB_ACMR1 field descriptions

Field	Description
CHM[63:32]	Bitwise channel mask bit: 0 Masked 1 Unmasked