Freescale Semiconductor EBxxx
Engineering Bulletin Preliminary Rev. 0, DraftC, 12/2015

External Bus Interface FAQs

related to MPC55xx and MPC56xx MCUs

by: David Tosenovjan

The document answers to frequently asked questions related to EBI module used with MPC55xx and MPC56xx MCUs, trying to fill gaps in existing manuals and application notes.

Table Of Contents

© Freescale Semiconductor, Inc., 2015. All rights reserved.

1 Non-CS access

1.1

Q: What is non-CS access and what’s its purpose? Please provide register configuration in case of non-CS access for whole EBI as per above configuration.

A: Non-CS access is an access with fixed “default” configuration (32-bit data bus width, burst inhibited, externally terminated) without possibilities to set it up according to some specific needs. It is selected when internal address fits into EBI address space (0x2000_0000 - 0x3FFF_FFFF) but does not fit into any configured CS address space (as defined by EBI_BRn and EBI_ORn registers). Also on certain devices (MPC55xx) it is the only option how to use external transfer acknowledge (TA), if needed (otherwise it can be set by EBI_BRx[SETA]). Using non-CS along with CS accesses do not interfere anyhow. It also saves one CS line. Non-CS will be often use in conjunction with FPGA and other memory-like devices where external TA is required and limitations given by fixed configuration are not causing any issues.

1.2

Q: How non-CS access is different from CS access in terms of timing and performance?

A: Non-CS does not use CS, OE and WE lines. But uses TA input (transfer acknowledge) so external device must drive this line (this signal substitutes configuration of wait states).

1.3

Q: Where to define address range for different peripheral devices in case of non-CS access?

A: Consider default MMU setup for EBI with physical address range defined as 0x2000_0000 - 0x3FFF_FFFF. From this whole EBI address space address ranges defined by EBI_BRn/EBI_ORn pairs are occupied by CS accesses, the rest are non-CS accesses.

The whole EBI address space may be reduced by MMU setup (thus non-CS address space as well), however it does not affect other crossbar master than the core (for instance eDMA).

1.4

Q: Whether ‘ORn’ and ‘BRn’ registers will be valid during non CS access? What should be their value for non CS accesses?

A: These registers are related to CS accesses only.

1.5

Q: Can we define different bus width for different devices (16 bit /32 bit) in non CS access? OR in non CS access, bus width is fixed for all devices, either 16/32 bit?

A: It can be set-up by EBI_MCR[DBM] what's however influence the whole bus width.

1.6

Q: In case of non CS access, how to use RD_WR#, WE#/BE#, OE#.

A: Non-CS does not use CS, OE and WE lines. RD_WR works same way as for CS accesses.

1.7

Q: How does EBI latch input data in case of non-chip select access (as OE line is not driven)?

A: EBI latches DATA every cycle during non-chip select accesses. The peripheral must assert TA line to specify when read data are valid and signal that the bus cycle is complete. If an MCU has no TA pin available, this restricts the MCU to CS accesses only.

1.8

Q: On EBI we have peripherals of 16-bit data bus as well as 32 bit-data bus. We will be setting EBI to 32-bit. In program how we are reading and writing data on EBI? Can we do 16-bit read and write even if we have set the EBI to 32-bit?

A: Yes you can. But as WE lines are not working in non-CS access, you will have to decode information about data position (I mean on which data lines the data are present) from address lines A[30:31] and TSIZ[0:1], if you need to know it.

2 EBI timing for lower frequencies than specified

2.1

Q: Data sheet figures for EBI timing data are categorized under device speed grade. For instance MPC5554 datasheet with the respective columns labeled "40MHz" "56MHz" and "66MHz." We need specification for 33MHz.

A: Characteristics for 33MHz can be estimated by linear extrapolation from 40, 56 and 66 MHz values specified for particular device variants in the datasheet. Output delay and setup time are frequency dependent, both hold times will stay the same.

The EBI bus speed is determined by CLKOUT settings so in your case you would look at column 33MHz (must be estimated as it is not specified).

Let me show example based on MPC5554 Microcontroller Data Sheet, Rev. 4, Table 22.

Extrapolated output delay is 12.1ns and setup time is 11.3ns (for ETBS=1) for external bus frequency 33MHz.

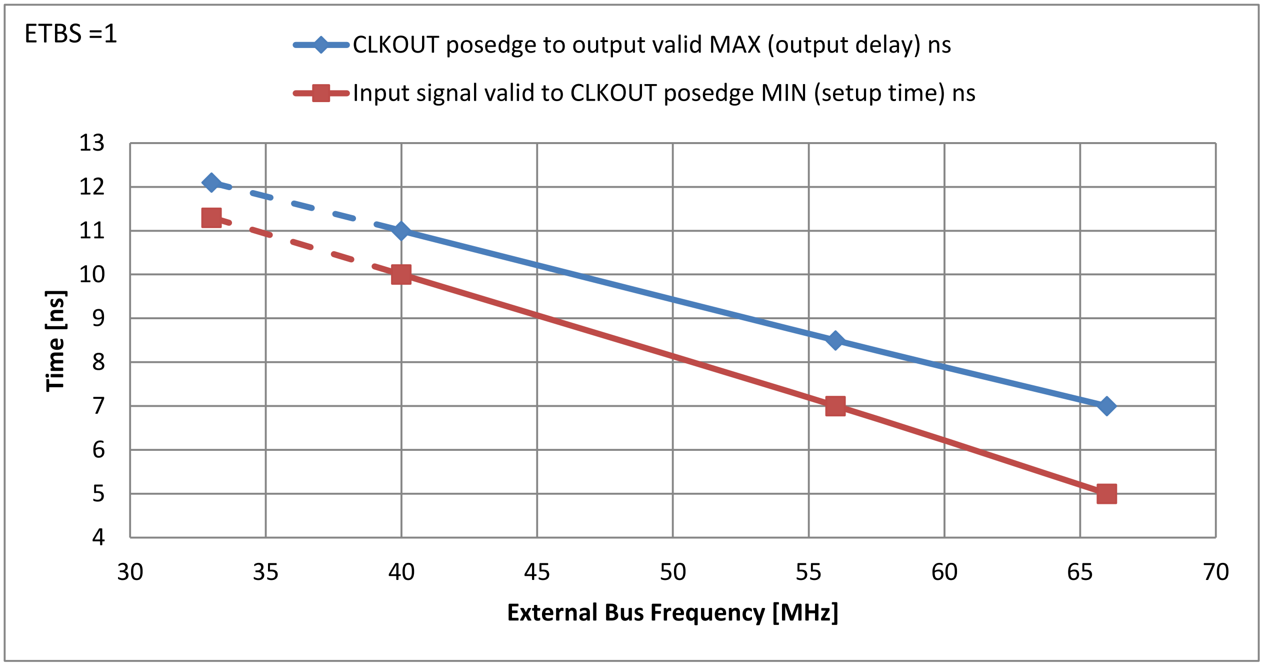


Figure 1. EBI timing

2.2

Q: We are driving CLKOUT by SYSCLK/4 instead of SYSCLK/2. We need EBI timing for CLKOUT = SYSCLK/4.

A: Technically the EBI does not use the CLKOUT signal directly as its clock. It uses the system clock, with a "sync" signal to tell it which posedge of system clock lines up to the external CLKOUT pin. Thus what was told before is truth in case system clock itself is changing down from maximum (in this case linear approximation will be used), but CLKOUT frequency divider does not affect timing values. Thus values for lower CLKOUT divider are valid.

In the case SIU_ECCR[EBDF] is set for 'Divide by 4' you will look at column 66MHz although 33MHz bus is used.

3 Memory size, address bus relations

Q: What is relationship between internal and external addresses? What is the size of SRAM memory I can connect to EBI module?

A: Internal address space is defined fixed within range 0x2000_0000-0x3FFF_FFFF. How it correlates with physical addresses on EBI address bus is defined by EBI_BRx and EBI_ORx that define base address and address mask for particular chip select.

Internal address lines are assigned to external address lines the same way and in the same order. Considering you have set CS for whole EBI space in binary

0b001x_xxxx_xxxx_xxxx_ xxxx_xxxx_xxxx_xxxx_xxxx_xxxx

'x' here are mentioned bits 3-31

For instance, considering the device offers 22 address lines and non-multiplexed mode has been choosen, bits 9-30 will be reflected on the external bus D_ADD [9:30] ('X'):

0b001x_xxxx_xXXX_XXXX_XXXX_XXXX_XXXX_XXXx

There is 22 address line thus maximum memory size is 8Mbytes (=2^23 because of 16-bit data bus and so D_ADD31 is not used) per chip select.

You can use also Muxed 32-bit mode using external buffer and ALE signal and in this case the only limitation is 512Mbyte overall (because of used range assigned for external memories is 0x2000_0000–0x3FFF_FFFF) as such configuration sends addresses via data lines.

4 Connection, Endianness

Q: What is the coding of the External Bus Interface, Big-endian or Little-endian? Or in other words, is Bit31 = MSB or LSB? How to connect external SRAM device? Why bits of registers are marked in reverse?

A: The question is not targeted to endianness, but PowerArch bit marking/line ordering.

MPC 55xx / 56xx are natively Big-endian devices as well as most of Freescale devices. However Power Architecture uses numbering style that bits in registers and address lines on external bus are marked in reverse. It is probably due to historical reasons, but it’s just matter of habit.

Bit 0 is most significant bit, and Bit 31 is least significant bit.

Figure 2 shows mnemotechnical help that can be supportive when programming PowerArch registers.

Just for clarification - endianness is about byte ordering what requires byte swapping, but that’s not the case. Refer to Figure 3.

64-bit H

standard	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
PowerArch	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

64-bit L

standard	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PowerArch	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63

32-bit

standard	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PowerArch	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

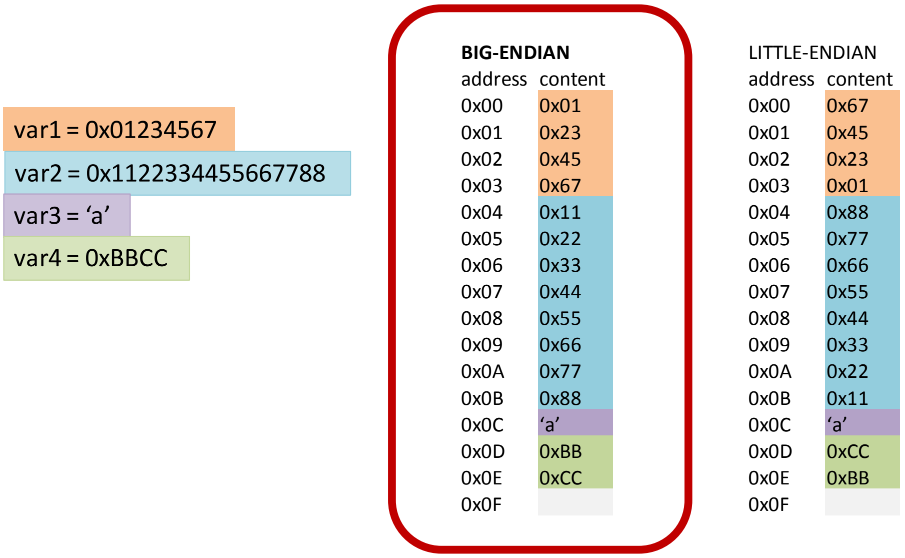
16-bit

standard	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PowerArch	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

8-bit

standard	7	6	5	4	3	2	1	0
PowerArch	0	1	2	3	4	5	6	7

Figure 2. Power architecture register organization



var1 = 0x01234567
var2 = 0x1122334455667788
var3 = ‘a’
var4 = 0xBBCC

BIG-ENDIAN

address	content
0x00	0x01
0x01	0x23
0x02	0x45
0x03	0x67
0x04	0x11
0x05	0x22
0x06	0x33
0x07	0x44
0x08	0x55
0x09	0x66
0x0A	0x77
0x0B	0x88
0x0C	‘a’
0x0D	0xBB
0x0E	0xCC
0x0F	

LITTLE-ENDIAN

address	content
0x00	0x67
0x01	0x45
0x02	0x23
0x03	0x01
0x04	0x88
0x05	0x77
0x06	0x66
0x07	0x55
0x08	0x44
0x09	0x33
0x0A	0x22
0x0B	0x11
0x0C	‘a’
0x0D	0xCC
0x0E	0xBB
0x0F	

Figure 3. Big/Little endian clarification (PowerArch higlighted in red)

The same goes for EBI connection.

The rule is following:

EBI line with highest number (dependent on data bus width – A31 for 8bit bus, A30 16bit bus and A29 for 32bit bus) is connected to memory LSB address line, then downwards according to size of connected memory.

So your SRAM connection will be EBI_ADR[30:x] -> SRAM_ADR[0:y]

DATA0 is most significant bit, and DATA31 is least significant bit.

ADDR0 is most significant bit, and ADDR31 is least significant bit.

There are no changes in byte ordering in comparison to common practice.

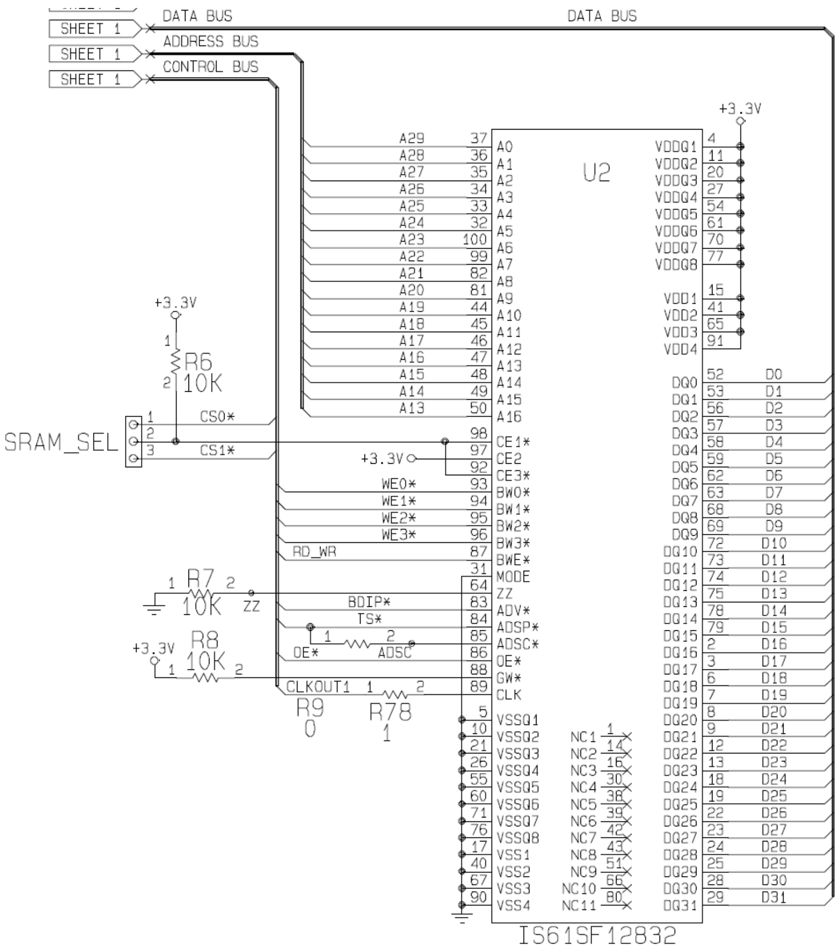


Figure 4. Example connection of external SRAM

5 Base address and address mask setting

Q: How to set base address and address mask?

A: EBI address range is 0x2000_0000–0x3FFF_FFFF. That’s fixed.

But for particular chip selects you can select ranges (that must be within EBI range above and not overlapped) using base address EBI_BRn[BA] and address mask EBI_ORn[AM]. Address mask actually specifies size of address space addressed by particular chip select.

Base address and mask are both in format: 0b111x_xxxx_xxxx_xxxx_x000_0000_0000_0000

Ones from left side are taken from EBI fixed address range and zeros from right side specifies address granularity that is minimum CS addressable area (2^15 = 32kB).

To Base Address - It must be from 0x2000_0000 - 0x3FFF_FFFF as it is address range for external memories and ranges for particular chip selects are not allowed to overlay.

To Address Mask - Number of zeros from right side (in binary) determines block size for particular chip select. For instance

17 zeros means (2^17)/1024=128kBytes. Then Address mask would be as follows 0xFFFE0000.

For example EBI_BR0[BA]=0x2000_0000 and EBI_OR0[AM]=0xFFFE0000 resulting in address range 0x2000_0000-0x2001_FFFF.

6 Line mixing

Q: Can data and address lines be connected in different order to simplify layout?

A: You may have data lines mixed on the memory side, but particular WEx line must be assigned to right data port.

WE3 -> data 31 to 24

WE2 -> data 23 to 17

WE1 -> data 15 to 8

WE0 -> data 7 to 0

Note that right WEx/BEx signal routing is necessary due to 16-bit, 8-bit or misaligned access if supported (considering 32-bit data port). This is related to volatile memories as FLASH memories mostly do not use WEx/BEx lines.

Address lines may be mixed as well in case of SRAM memory. In case of FLASH memory it becomes problematic because external flash programming algorithm would have to count with that and Block Erase operation could not be used, so it is not recommended practice in this case.

7 Multiplexed/Non-multiplexed mode

Q: I am using MCR5674F/MPC5676R device. May I know whether we can use EBI multiplexed and non-multiplexed mode in with same configuration of the corresponding PCR registers. For example: one chip is connected to D_ADD (i.e. non-muxed mode) and another chip is connected to D_ADD_DAT (i.e. muxed mode) and also keep the PCR seting same (i.e. enable D_ADD_DAT). Can Control via AD_MUX via BR register?

A: No, it cannot be used this way just because muxed addr/data lines and non-muxed address lines are routed to the same pins and their functionality are being selected by particular PCR register setting (primary or alternative function).

If your design is already connected incorrectly (thus some memory devices uses address lines before address latch a some after address latch) the only meaningful solution is to change layout and use address lines after latch device for address lines of all connected memory devices.

Software workaround would be to reconfigure PCR regs before every access to non-muxed memory and after access change PCR regs back i.e. 16 PCR writes before access and 16 PCR writes after access.

8 ALE rising edge time

Q: ALE Signal Timing in datasheet refers to two timing parameters. That is the minimum ALE pulse width and ALE negated to address invalid. I cannot find any timing relationship between the address becoming valid to the rising edge of ALE. We want to use a latch that will be clocked by the rising edge of ALE.

A: You can use EBI AC Timing specification tCOH along with the tAPW and tAAI to determine CLKOUT to ALE timing.

Start with the Address Invalid time. tCOH specifies the Address Invalid time with respect to CLKOUT. tAAI specifies the Address Invalid time with respect to ALE. So putting these two together, you will get CLKOUT to ALE.

The address phase in each case is maintained at one cycle as well as TS signal. So you have all info to determine ALE rising edge time (tC+tCOH-tAAI-tAPW). The result will be maximum.

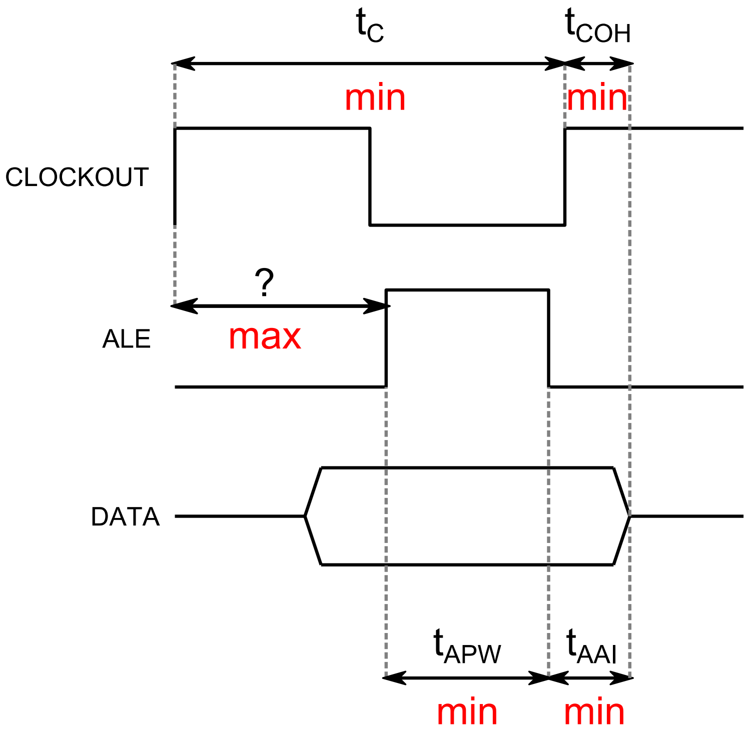


Figure 5. CLKOUT rising edge to ALE rising edge

9 External flash type

Q: Does EBI module support NAND flashes?

A: No, only NOR flashes can be connected via EBI.

10 SRAM speed grade

Q: Does EBI module support DDR SRAM?

A: No, only SDR memories are compatible with MPC55xx/56xx external bus.

11 Burst option

11.1

Q: Can we use SSRAM with Interleaved Bursts?

A: No, only Linear Burst is supported.

11.2

Q: SSRAM Flow-through or Pipelined?

A: Our reference designs always contain Flow-through memories. However MPC55xx/56xx is capable of interfacing with SRAMs that operate in either Flow Through or Pipeline mode, which is selectable by the addition of wait states in the MCU’s read timing.

12 Dual-controller mode

12.1

Q: Why MPC5553/4 reference manual mentions Dual-Controller Mode as one of the possible FMPLL configuration whereas Erratum 3111 says “do not use dual controller mode“. Please clarify the relationship between „Dual-Controller“ and „Multi-Master“ modes.

A: Erratum 3111 says following:

„In dual controller mode, the specification for the phase relationship between EXTAL and CLKOUT is +/- 1 ns, however this does not allow adequate set up and hold times to guarantee successful operation of the external bus to a second MCU.“

Actually it means only that Multi-Master mode configuration will not work properly and this configuration is not supposed to be used. If mentioned phase shift between clock pads does not cause any issue, dual-controller configuration can be used.

Some customers use this option for clocking of another device (for instance FPGA) by MCU’s CLKOUT output. It brings some benefits to them:

1) time to lock the PLL is faster than in any other mode

2) CLKOUT is driven even if the MCU is under reset (and another device’s PLL can begin to lock).

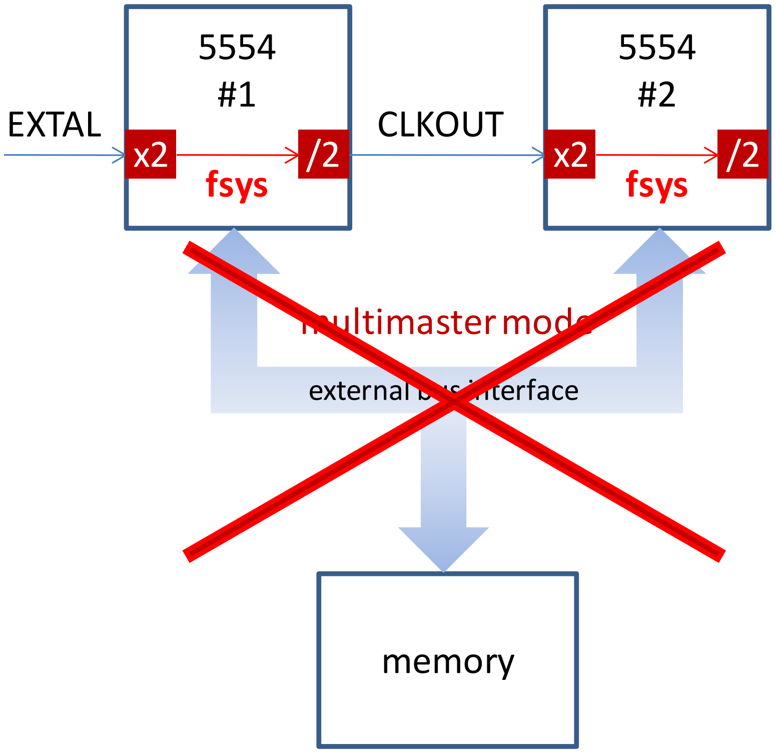


Figure 6. Dual-controller mode

12.2

Q: Is there a MPC5xxx device offering External Master Interface option?

A: Multi-master mode is not working on MPC5554 due to erratum 3111. On the later device as MPC5674F, multi-master mode is not implemented.

Single Master Mode is the only mode, it is actually normal mode. No MPC55xx/56xx MCU can work as a slave.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSeminconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

freescale™
semiconductor

Freescale™ and the Freescale logo are trademarks of
Freescale Semiconductor, Inc. All other product or service names
are the property of their respective owners.

© Freescale Semiconductor, Inc. 2015. All rights reserved.

EBxxx
Rev. Draft C
12/2015