DEMYSTIFYING DESIGN WITH I.MX PROCESSORS: AN OVERVIEW OF THE AVAILABLE DOCUMENTATION AND TOOLS TO ENABLE YOUR MPU DESIGN

LAT-DES-T2441

BRUNO CASTELUCCI

MPU/SECURITY FAE

NOVEMBER 2016

SECURE CONNECTIONS FOR A SMARTER WORLD

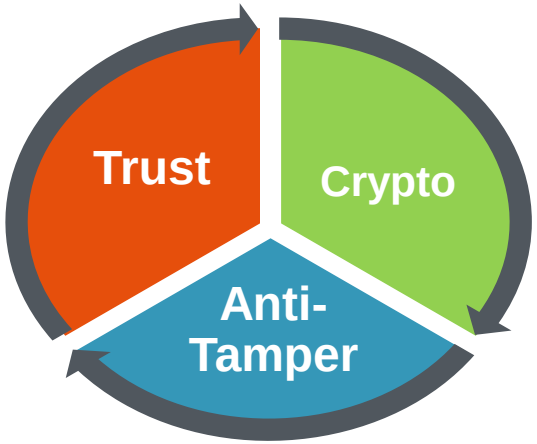
- **Introduction**
  - Evaluation
  - Boards
  - Binaries
  - Product
- **Hardware**
  - Reference Boards Files
  - Power Consumption
  - Data Sheet
  - IOMUX
  - Hardware Dev Guide
  - DDR Programming Aid
  - DDR Stress Test
  - MFGTools
  - PMICs
- **Secure boot**
  - Code Signing Tool
  - High Assurance Boot
  - Security RM
- **Software**
  - Reference BSP Source
  - BSP Document Bundle
  - Linux RM
  - BSP Release Notes
  - BSP User Guide
  - Yocto UG
  - Multimedia UG
  - Chip Errata
  - Device Reference Manual
  - Application Notes
  - Other Security SW
- **Support**
  - Community
  - Official Online Support
- **Partners**
  - SoM Partners
  - Software Partners
  - Training Partners

INTRODUCTION

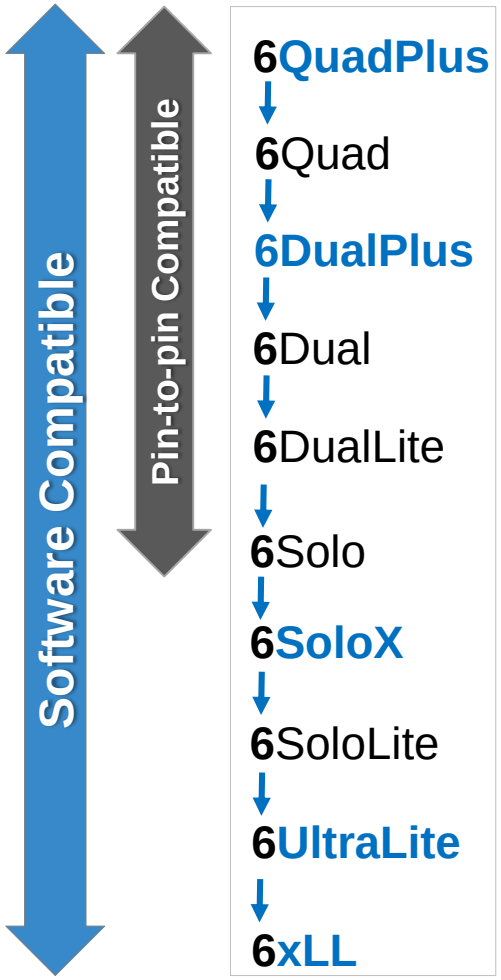
i.MX Applications Processor Values

Product Longevity

- **Scalability**
  - CPU (single/dual/quad, asymmetric), GPU, IO
  - Software: Linux, Android, Windows-embedded, RTOS
  - Pin compatibility and software portability
- **Trust**
  - Longevity: minimum of 10-15 years in all markets
  - Quality, robustness, zero-defect methodology
  - Security and Safety
  - Qualifications: AEC-Q100, JEDEC Industrial and Consumer
- **Ease of Use**
  - Industry-leading ecosystem, partnerships and support
  - Design collateral, distribution, communities
  - System solutions: SoC, sensors, memory, PMIC, connectivity



i.MX 8 Series: 3 families of parts with targeted features



i.MX 8 family

Advanced Graphics and Performance

ARM ® v8-A

(32-bit/ 64-bit)

i.MX 8M family

Advanced Audio and Video

ARM ® v8-A

(32-bit/ 64-bit)

i.MX 8X family

Safety Critical & Efficient Performance

ARM ® v8-A

(32-bit/ 64-bit)

i.MX 7

Power Efficiency & BOM Cost Optimizations

ARM® v7-A

(32-bit)

EVALUATION

Enablement through Reference Designs

SABRE Platform for Smart Devices

- Builds on SABRE Board design with additional features including 10.1” capacitive multi-touch display, 2x MIPI camera sensors, SPI Nor Flash, GPS, ambient light sensor and digital microphones

Supported	Description
i.MX 6Quad i.MX 6Dual (emulated)	Quad-core 1-1.2 GHz ARM Cortex-A9 Dual-core 1-1.2 GHz ARM Cortex-A9
i.MX 6DualLite i.MX 6Solo (emulated)	Dual-core 1GHz ARM Cortex-A9 Single-core 1GHz ARM Cortex-A9

SABRE for Auto Infotainment

- Support for terrestrial and satellite radio tuners, Wi-Fi, Bluetooth, GPS, cellular modem, iAP authentication modules, MOST vehicle networking, cameras and displays

Supported	Description
i.MX 6Quad i.MX 6Dual (emulated)	Quad-core 800MHz-1GHz ARM Cortex-A9 Dual-core 800MHz-1GHz ARM Cortex-A9
i.MX 6DualLite i.MX 6Solo (emulated)	Dual-core 800MHz ARM Cortex-A9 Single-core 800MHz ARM Cortex-A9
i.MX 6SoloX (Q3’15)	800MHz ARM Cortex-A9 200MHz ARM Cortex-M4

SABRE Board for Smart Devices

- Multiple connectivity options: Wi-Fi®, Bluetooth®, GPS, Ethernet, SD, parallel/serial interfaces, SATA (i.MX 6Quad only), and PCIe

Supported	Description
i.MX 6Quad i.MX 6Dual (emulated)	Quad-core 1-1.2 GHz ARM Cortex-A9 Dual-core 1-1.2 GHz ARM Cortex-A9
i.MX 6SoloX	1GHz ARM Cortex-A9 200MHz ARM Cortex-M4

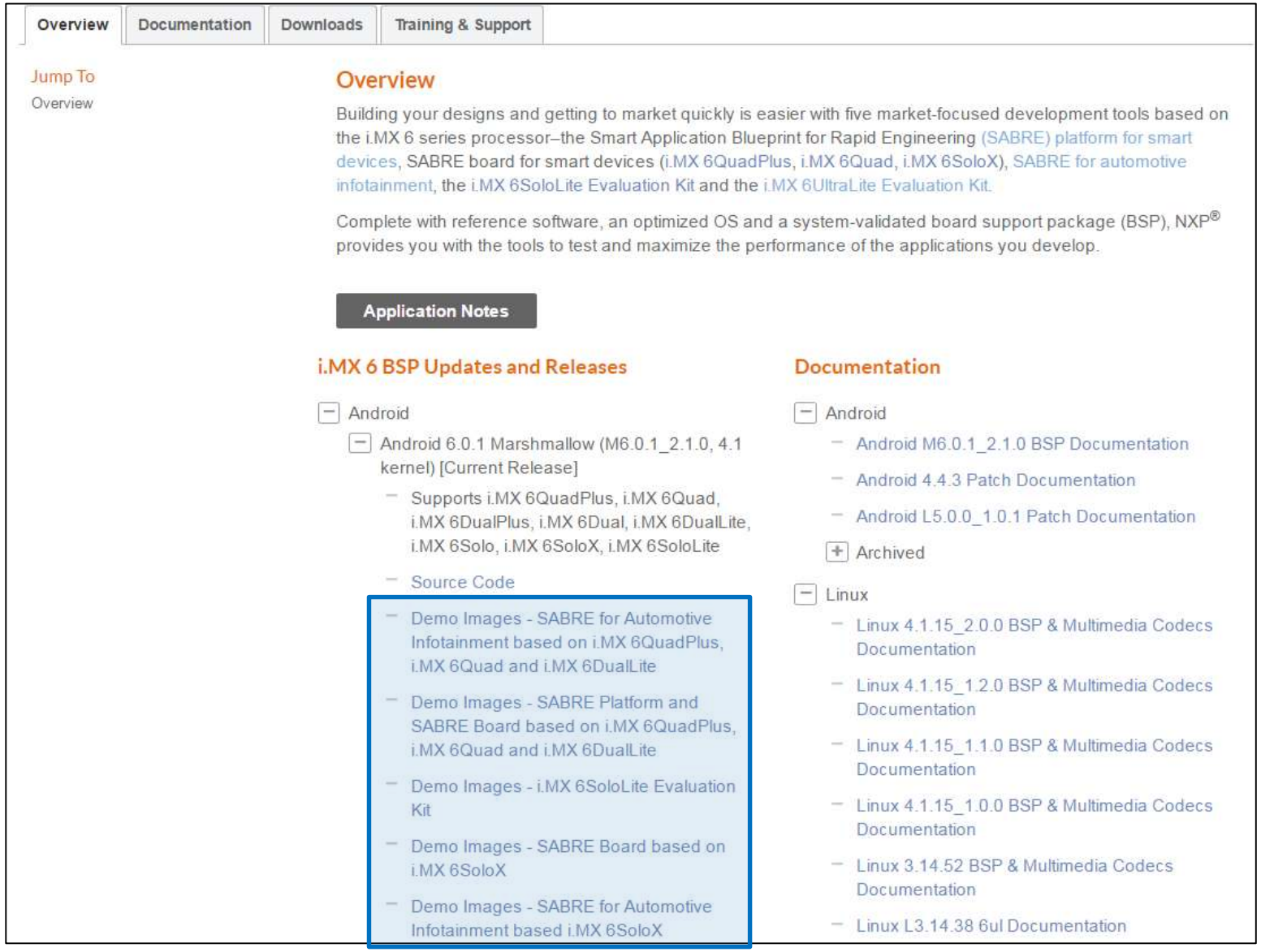
i.MX 6SoloLite Evaluation Kit

- Enables EPD and/or LCD or HDMI display, touch control and audio playback, and the ability to add WLAN, a 3G modem or Bluetooth technology
- E-Ink display available separately

Supported	Description
i.MX 6SoloLite	Single-core 1GHz ARM Cortex-A9

IMX6_SW: i.MX 6 Series Software and Development Tool Resources

Link



- Binaries
- Source
- Software Docs
- Tools
- Board page links
- Device page links

Board web page – link (iMX6ULEVK)

MCIMX6UL-EVK: i.MX6UltraLite Evaluation Kit ☆

Overview | Documentation | Downloads | Buy / Specifications | Training & Support

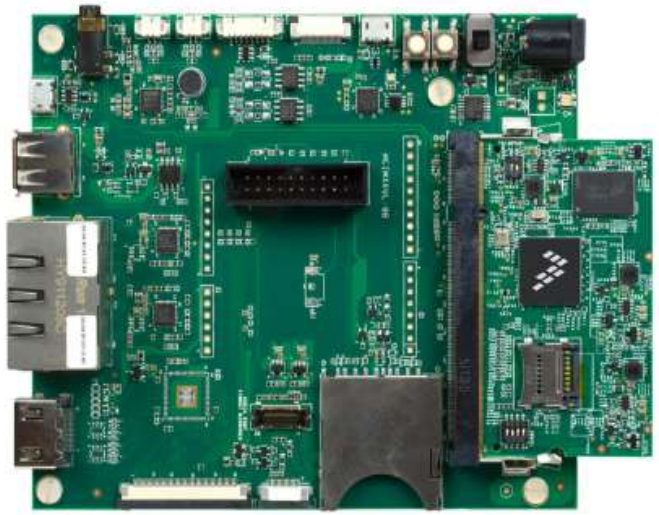
Jump To
- Overview
- Features
- Supported Devices
- Kit Contains
- Jump Start Your Design
- Related Products
- Community Discussions

Overview

NXP® delivers a highly flexible, market-focused development tools with an evaluation kit (EVK) based on the i.MX 6UltraLite applications processor. The i.MX 6UltraLite processor is the first device in the i.MX product line to have a single ARM® Cortex®-A7 core operating at speeds of up to 696 MHz. This EVK enables an LCD display and audio playback as well as many connectivity options. It is designed to showcase the most commonly used features of the processor in a small, low cost package and to facilitate software development with the ultimate goal of faster time to market through the support of the Linux® operating system.

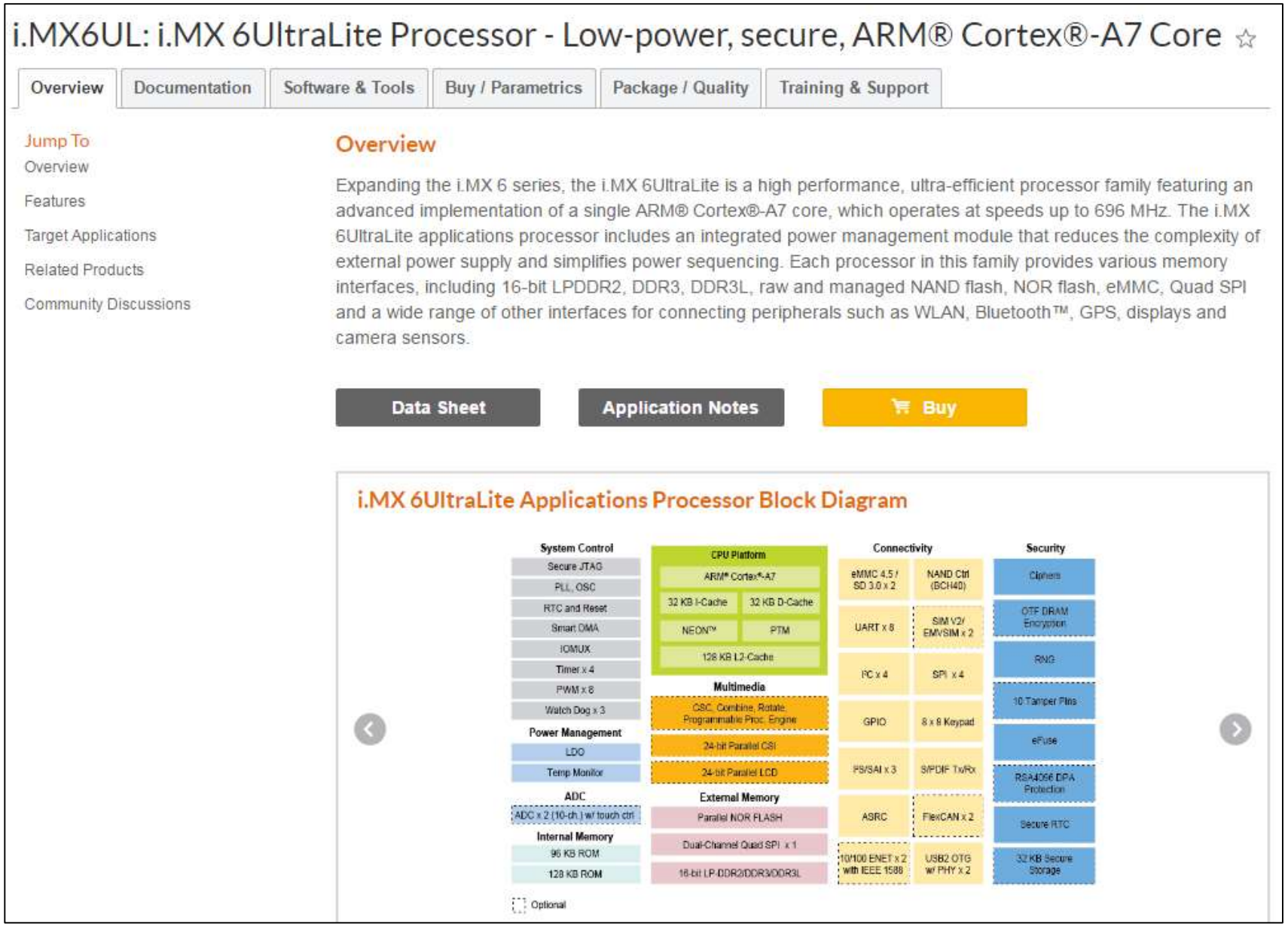
Buy

i.MX 6UltraLite Evaluation Kit



- Description
- User Guide
- Design Files
  - Schematics
  - Layout
  - BOM
- Buy Direct

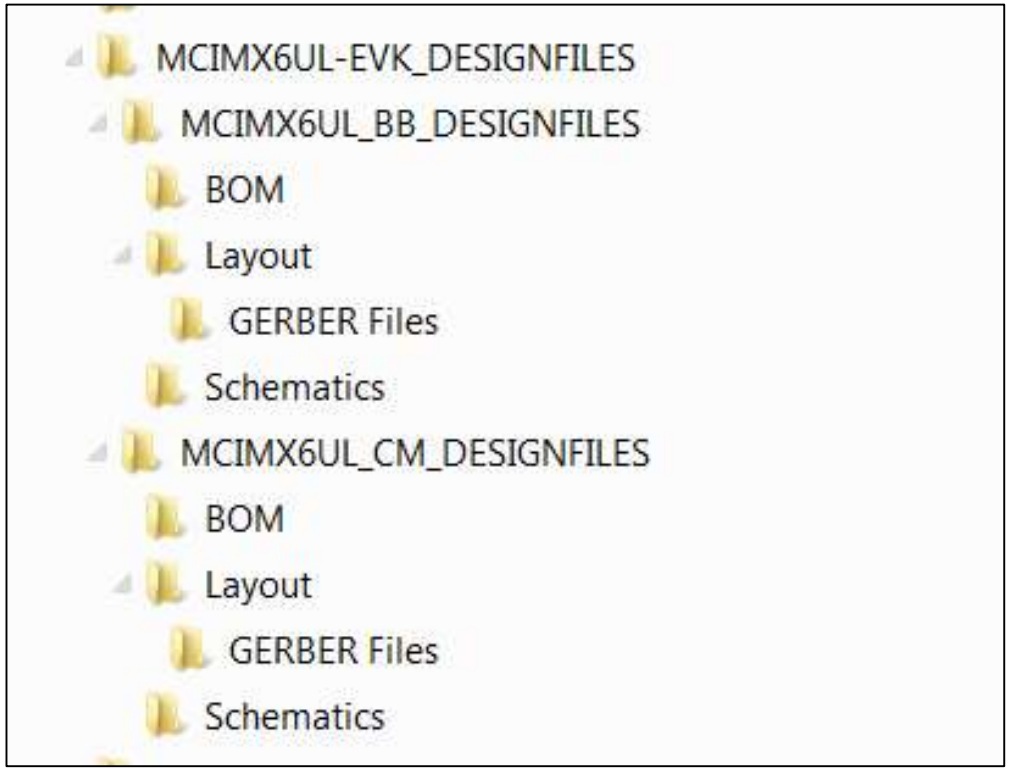
Device Page – link (iMX6UL)



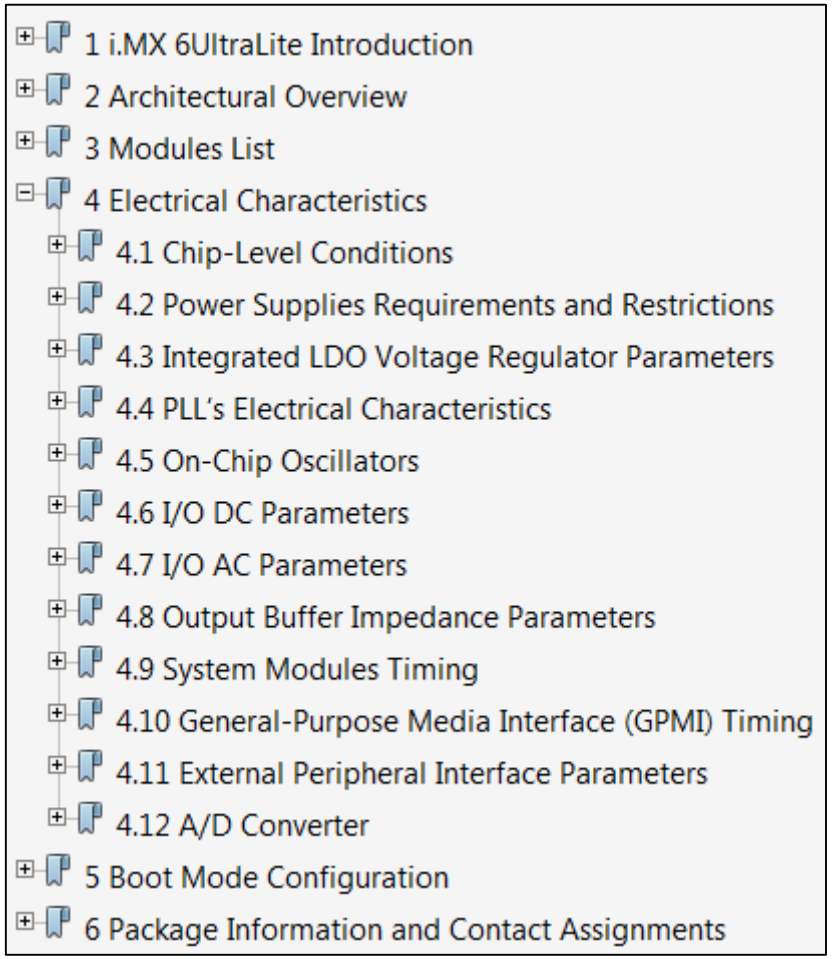
- Documents
  - DS
  - RM
  - SRM
  - Errata
- App Notes
- Samples
- Other

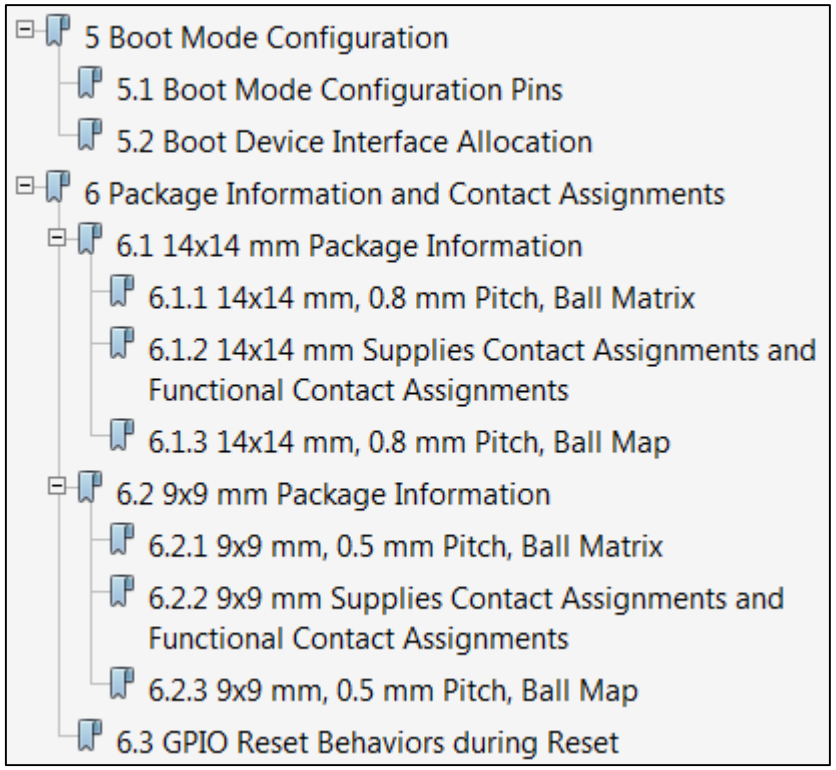
HARDWARE DESIGN

Board Reference Design Files - link (iMX6ULEVK)

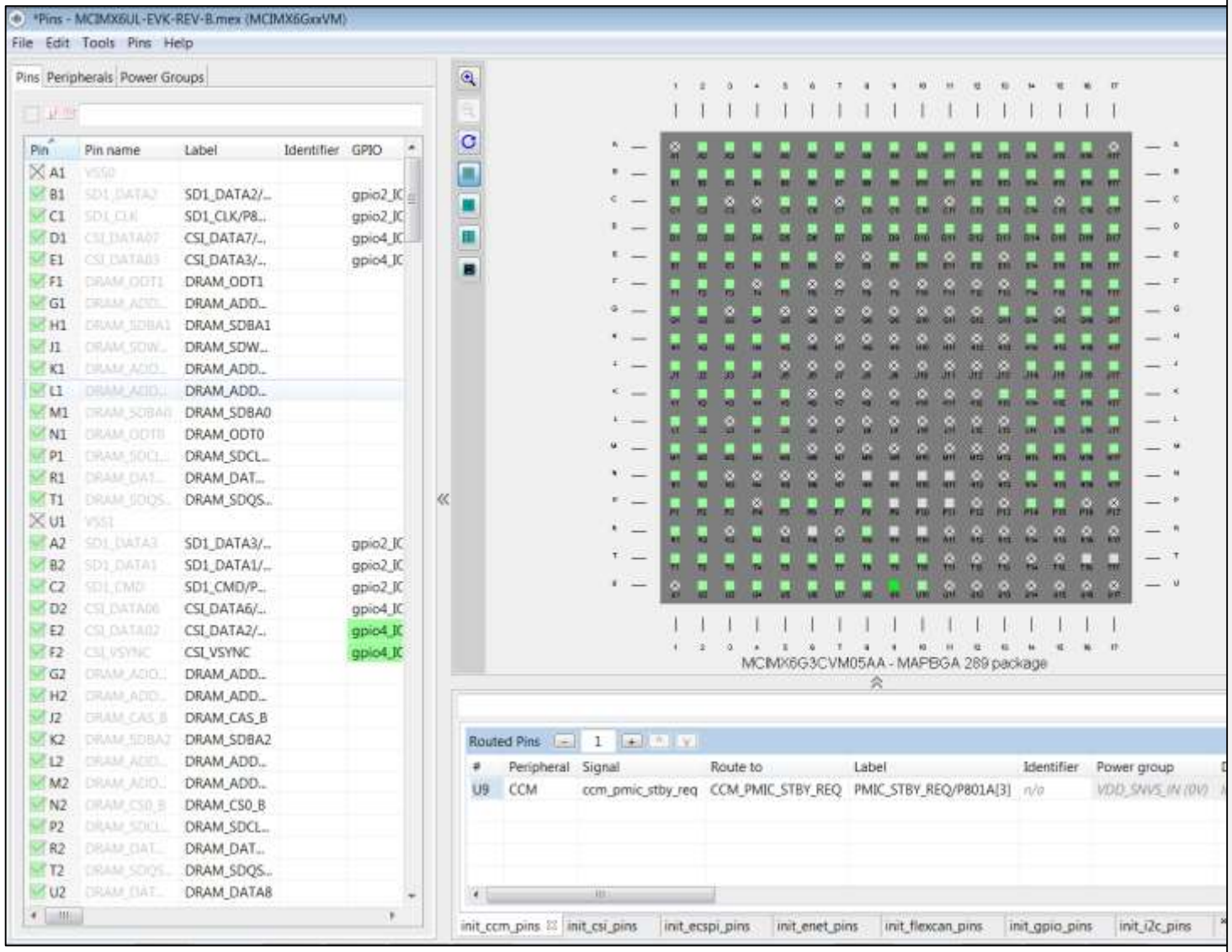


Datasheet – Device Page – link (iMX6UL)





IOMUX – Tools Page – link (iMX6)



imx6ul-board.dtsi | iomux_config.c | iomux_config.h

```
#include "skeleton.dtsi"
#include "imx6ul-pinfunc.h"

/ {
    model = "Freescale i.MX 6UltraLite User Board";
    compatible = "fsl,imx6ul-board", "fsl,imx6ul";

    soc {
        #address-cells = <1>;
        #size-cells = <1>;

        iomuxc: iomuxc@020e0000 {
            compatible = "fsl,imx6ul-iomuxc";
            reg = <0x020e0000 0x4000>;
        };
    };
};

&iomuxc {
    pinctrl-names = "default";
    pinctrl-0 = <&init_ccm_pins>;

    imx6ul-board {

        init_ccm_pins: init_ccm_pinsgrp {
/*
 * TEXT BELOW IS USED AS SETTING FOR THE PINS TOOL ***************************
init_ccm_pins:
- options: {coreID: singlecore}
- pin_list:
  - {pin_num: U9, peripheral: CCM, signal: ccm_pmic_stby_req, pin_signal: CCM_PM
 * BE CAREFUL MODIFYING THIS COMMENT - IT IS YAML SETTINGS FOR THE PINS TOOL ***
 */
            fsl,pins = <
            >;
        };


        init_csi_pins: init_csi_pinsgrp {
/*
 * TEXT BELOW IS USED AS SETTING FOR THE PINS TOOL ***************************
init_csi_pins:
- options: {coreID: singlecore}
```

Power Consumption – Device Page – link (iMX6UL)

i.MX 6UltraLite Power Consumption Measurement

1. Introduction

This application note helps to design power management systems. Through several use cases this application note illustrates the current drain measurements of the i.MX 6UltraLite Applications Processors taken on the MX6UL EVK Evaluate Kit Platform. You will be enabled to choose the appropriate power supply domains for the i.MX 6UltraLite chips and become familiar with the expected chip power in various scenarios.

Because the data presented in this application note is based on empirical measurements taken on a small sample size, the presented results are not guaranteed.

Contents

i.MX 6UltraLite Power Consumption Measurement **(REV 2)**

PDF (486.9 kB) AN5170 [English]

Slide 1 - Hardware Design Guide - Device Page – link (iMX6UL)

Hardware Development Guide for the i.MX 6UltraLite Applications Processor **(REV 1)**

PDF (1.2 MB) IMX6ULHDG [English]

Hardware Development Guide for the i.MX 6UltraLite Applications Processor

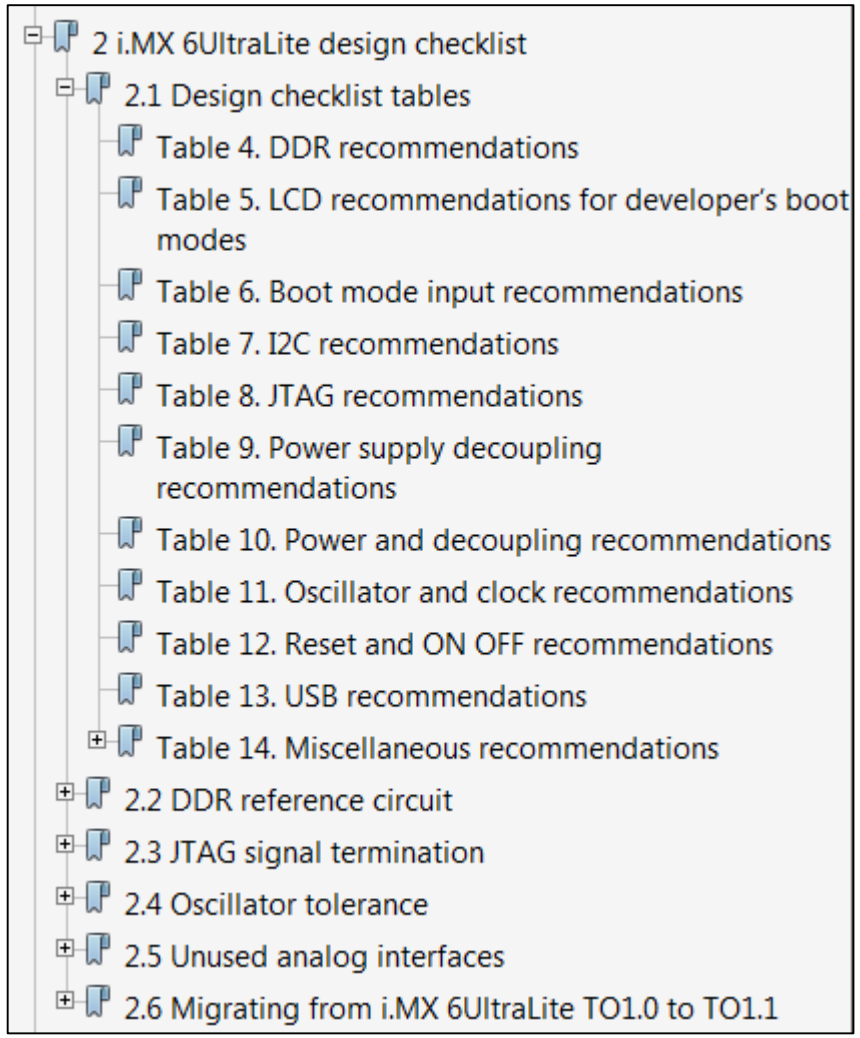
1 About This Book

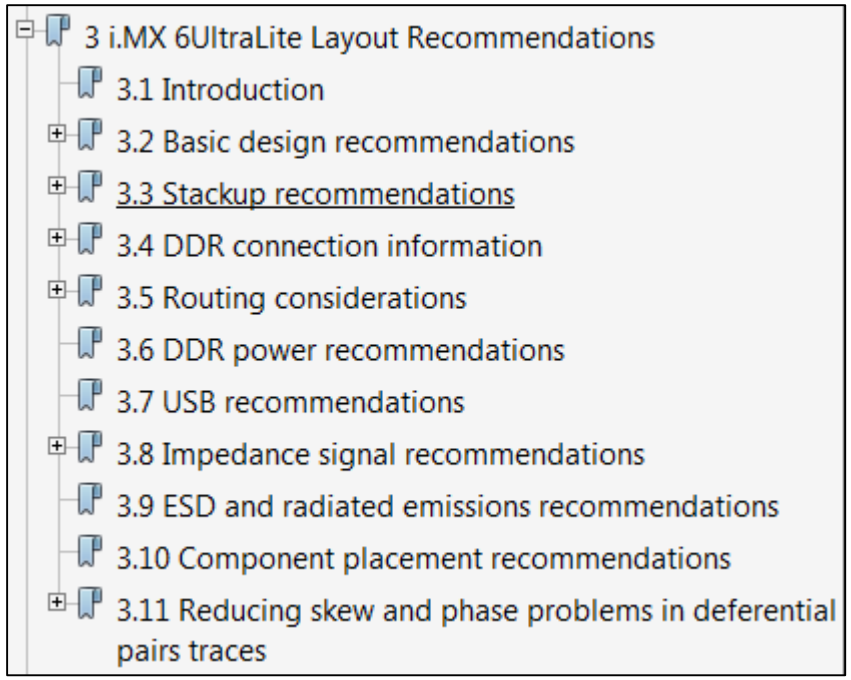
1.1 Overview

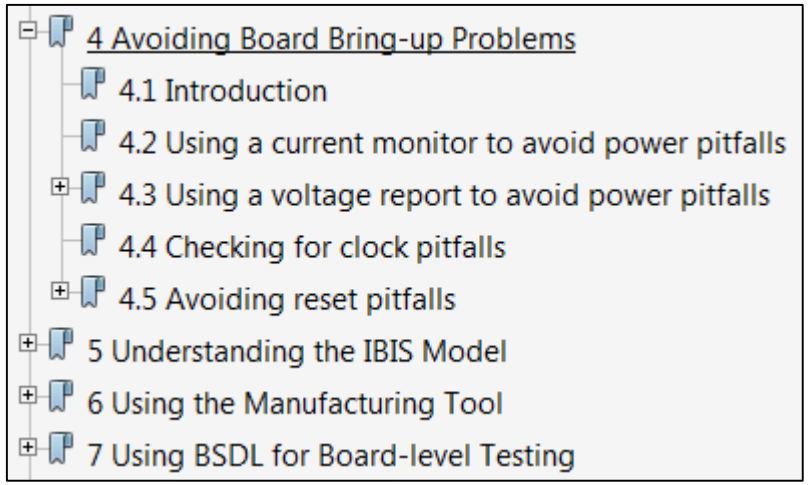
This document's purpose is to help hardware engineers design and test their i.MX 6UltraLite processor-based designs. It provides information on board layout recommendations, and design checklists to ensure first-pass success and avoidance of board bring-up problems. It also provides information on board-level testing and simulation such as using BSDL for board-level testing, using the IBIS model for electrical integrity simulation and more.

Contents

Slide 2 - Hardware Design Guide - Device Page – link (iMX6UL)







Board Bring-Up and Debug Checklist

Checklist Item
The following items need to be completed serially
Visual Inspection
Verify all SoC voltage rails
Verify power up sequence
Measure/probe input clocks (32kHz, 24MHz, others)
JTAG connectivity (DS-5, RV-ICE, Lauterbach, Macraigor, etc)
Access internal RAM
Run basic DDR initialization and test memory
Run DDR Stress test
The following items may be worked on in parallel with other bring up tasks
Verify CLKO outputs (measure and verify default clock frequencies for desired clock output options); this assumes that the board design supports probing of the CLKO pin.
Measure boot mode frequencies (set the boot mode switch for each boot mode and measure the following, depending on what is available in the system): - NAND (probe CE to verify boot, measure RE frequency) - SPI-NOR (probe slave select and measure clock freq) - MMC/SD (measure clock freq)
Run other unit tests

Recommended Flow and Tools for Optimal DDR Initialization

Create initial DDR initialization script

- Use the **DDR Register Programming Aid**
- Input values from Memory vendor datasheet

Perform calibration to optimize script

- Use the **DDR Stress Test** : Calibration Routine
- Generates optimal calibration values

Validate DDR interface with optimized script

- Use **DDR Stress Test** : Memory Sub routines
- Using updated calibration values

More details? Look for **FTF-DES-N1936** FTF Americas Presentation on the community.

DDR Programming Aid – community – link (iMX6)

i.MX6UL

- i.MX6UL DDR3 DRAM Register Programming Aid
- i.MX6UL LPDDR2 DRAM Register Programming Aid

How to use the DRAM register programming aid outline

Step 1. Obtain the desired DRAM data sheet from the DRAM vendor

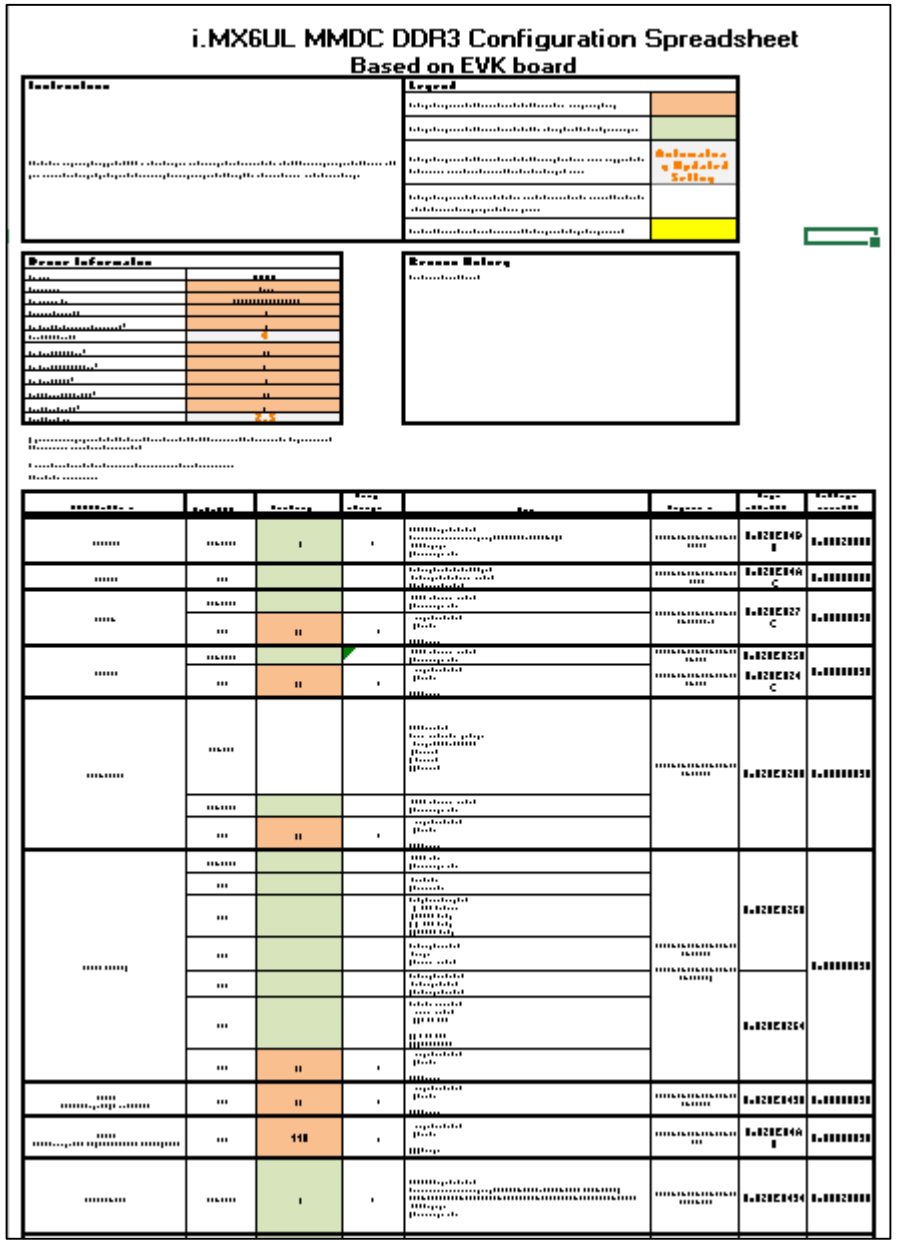
The following are to be completed in the Register Configuration Worksheet tab.

How To Use | **Register Configuration** | DStream .ds file

Step 2. Update the Device Information table to include the DRAM information and system usage

Device Information	
Memory type:	DDR3
Manufacturer:	Micron
Memory part number:	MT41J128M16HA-15E
Density of each DDR3 device (Gb):	2
Number of DRAM devices per chip select	4
Density per chip select (Gb)[1]:	8
Number of Chip Selects used[2]	2
Number of ROW Addresses[2]	14
Number of COLUMN Addresses[2]	10
Number of BANKS[2]	8
Bus Width (input 16, 32, or 64 bits)[2]	64
Clock Cycle Freq (MHz)[3]	533
Clock Cycle Time (ns)	1.876

Step 3. Go through the various shaded cells in the spread sheet to update with data from the DRAM sheet (t



DDR Stress Test – community – link - (iMX6)

Option 1 GUI based:

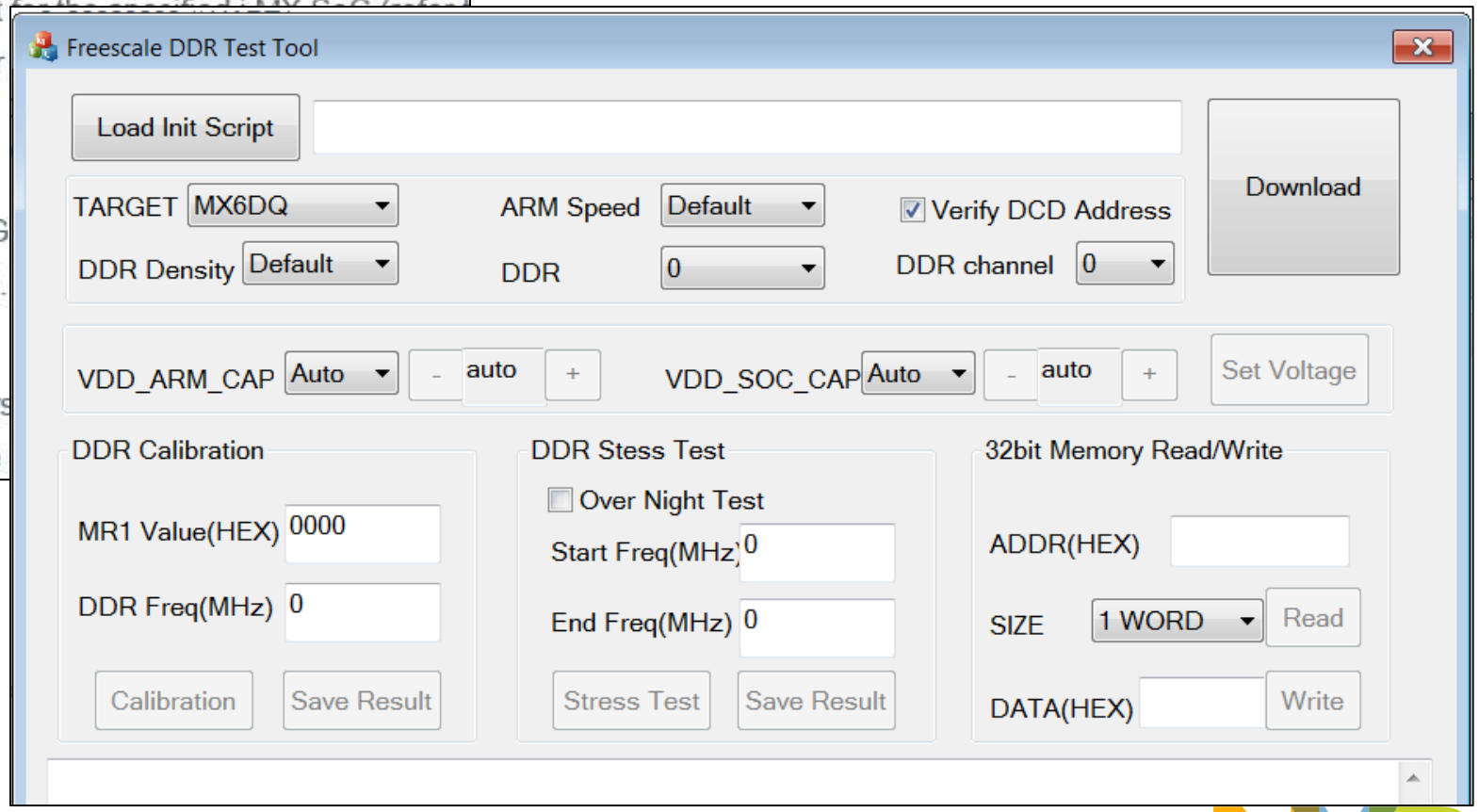
Run the GUI executable and connect your board to the host PC via USB

- Archive file: *ddr_stress_tester_vX.xx.zip*
- The tool will first need to run a DDR initialization script
  development boards can be found in this zip file under

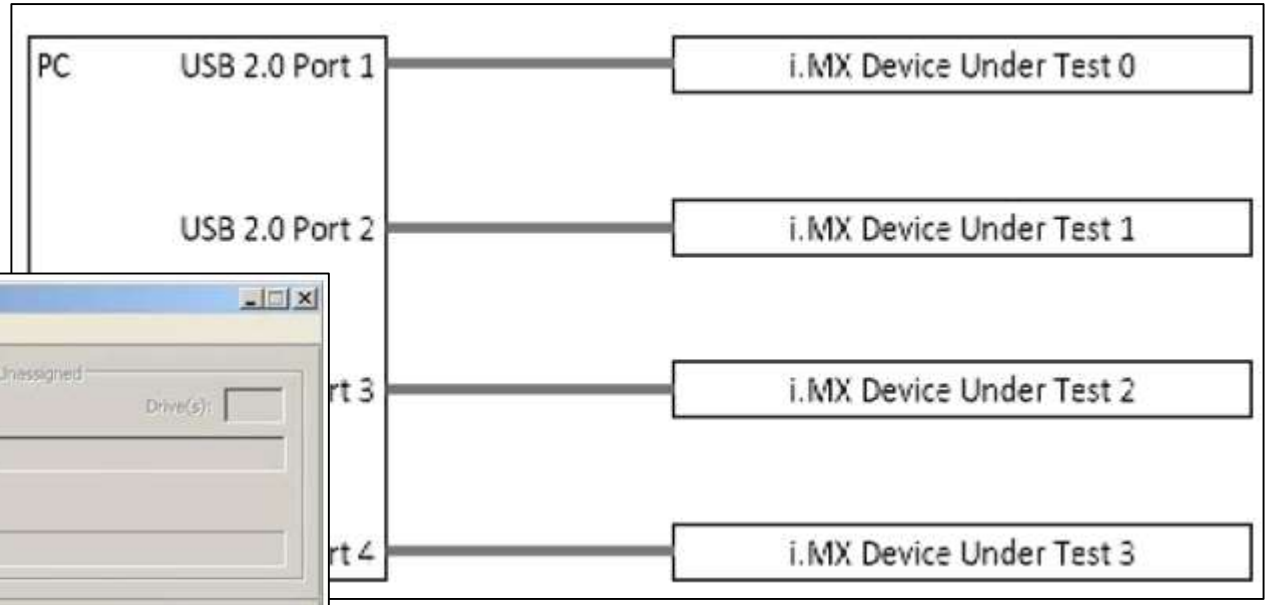
Option 2 DDR Stress Tester: JTAG Interface

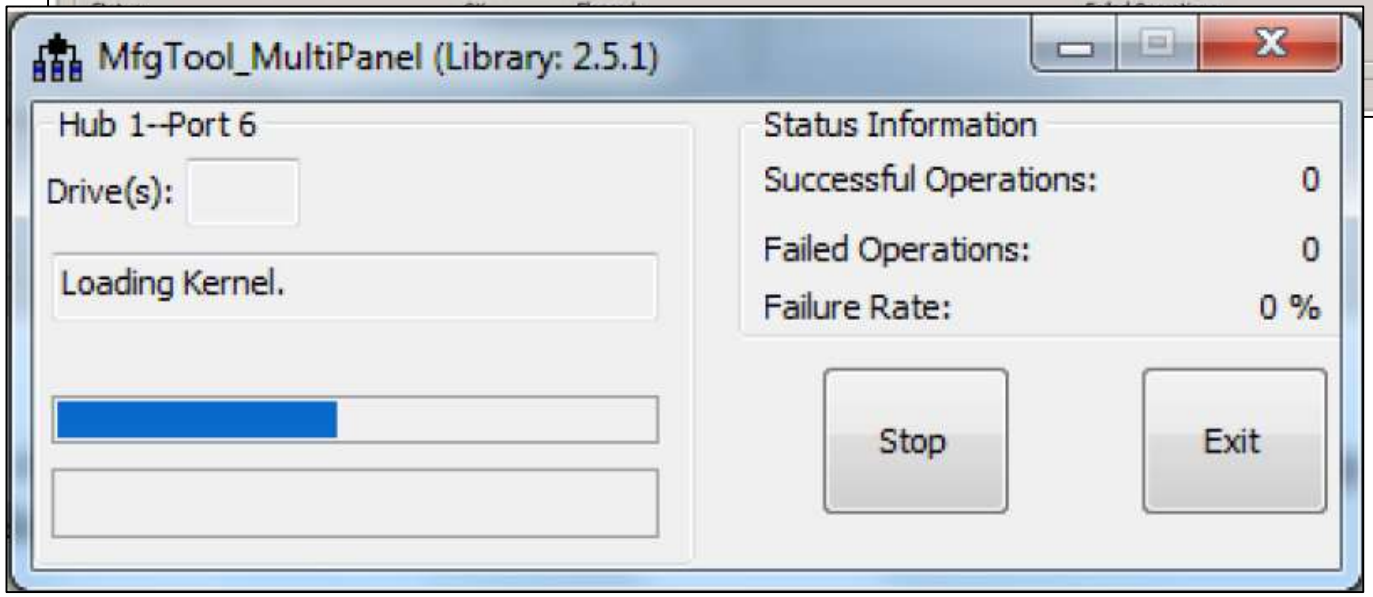
A hardware debugger connected to the board via the JTAG
Results are shown on the UART serial port (115200-8-n-1).

- Archive file: *ddr_stress_tester_jtag_vX.xx.zip*
- As with the GUI tool, the JTAG/debugger option will firs
  location of the example scripts (which are found in the



MFGTools - Tools Page – link (i.MX6)





- Configurable through xml input files.
- Flexible to program NAND, eMMC, QSPI, SDCard, etc.
- Can burn fuses and other internal registers.
- Can work with multiple boards at the same time.

Power Management – Device Page – link – (PF3000)

NXP › Power Management › PMICs › PMICs for i.MX Processors

PF3000: 12-Channel Configurable PMIC UPDATED ☆

Overview | Documentation | Software & Tools | Buy / Parametrics | Package / Quality | Training & Support

Jump To
- Overview
- Features
- Related Products

Overview

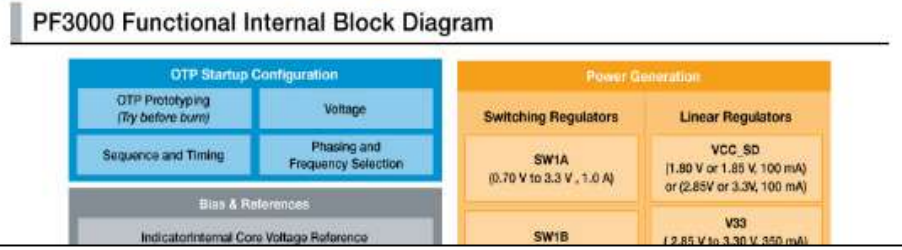
The PF3000 power management integrated circuit (PMIC) features a configurable architecture that supports numerous outputs with various current ratings as well as programmable voltage and sequencing. This enables the PF3000 to power the core processor, external memory and peripherals to provide a single-chip system power solution.

The PF3000 is ideally suited to Cortex™ A7 based i.MX 7Solo and i.MX 7Dual application processors to meet low power application requirements. The PF3000 is a perfect companion for the very low power Cortex™ A9 core product family including the i.MX 6DuaLite and all single-core Cortex A9 i.MX processors.

Compatibility with i.MX applications processors is shown in multiple reference designs and facilitates software controlled, dynamic voltage scaling.

Data Sheet | Application Notes | Buy | Sample

PF3000 Functional Internal Block Diagram



- Documentation
  - App Notes
  - Data Sheets
- Dev Boards

Other Tips

1. Copy the DDR, eMMC and PMIC Design.
2. NAND, eMMC or QSPI Flash?
3. Leave the boot selection for prototype.
4. Leave the Ethernet for development.
5. No production rootfs on SDCard.

   But leave an SDCard for bring-up.
6. Leave the OTG USB connector for factory programming.
7. Follow the design check list.
8. Submit your schematics for review.
9. Thermal on higher performance devices.

SECURE BOOT

High Assurance Boot – AN4581 – link – (HABv4)

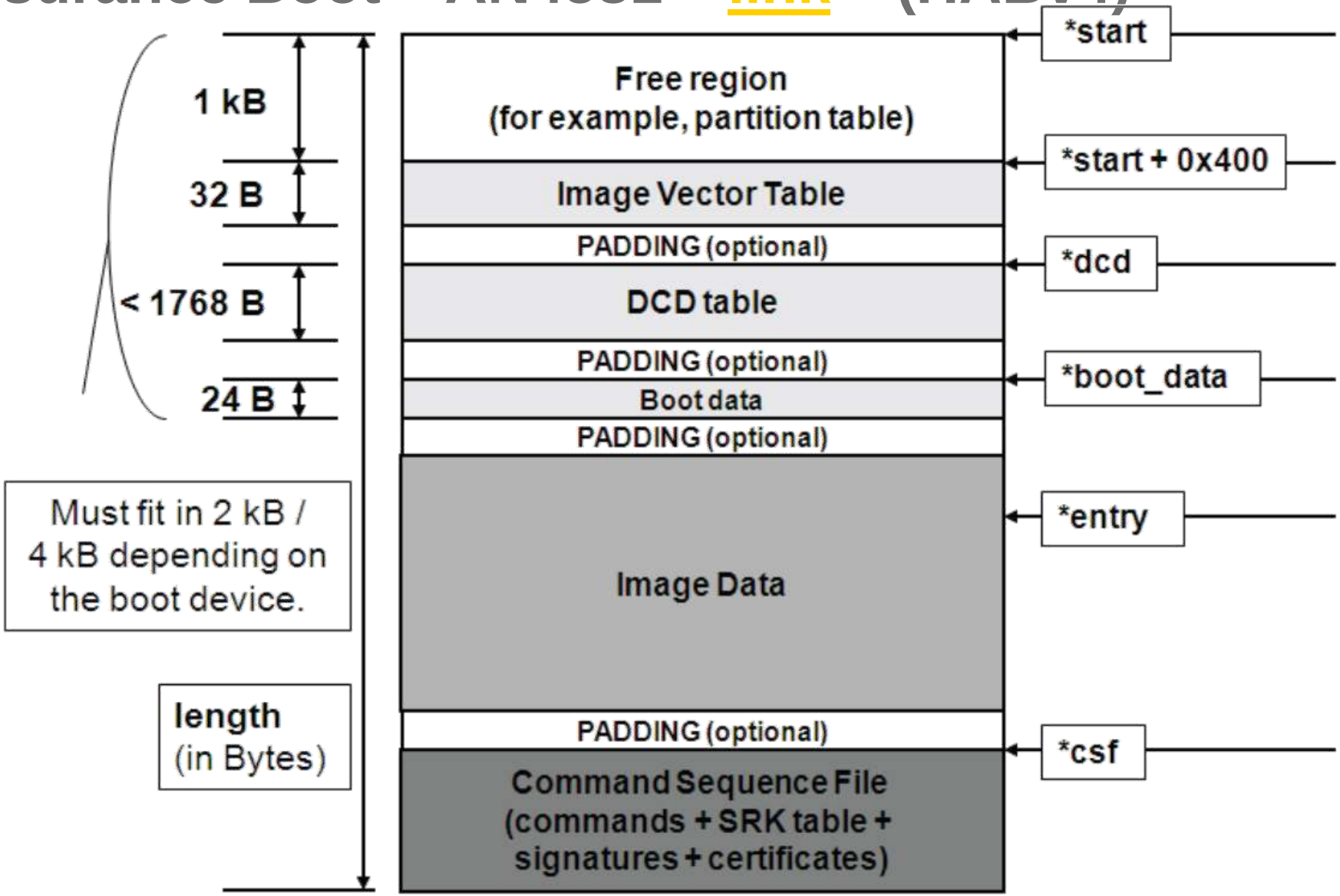


Figure 1. Typical memory layout of a signed image

Code Signing Tool - Tools Page – link (iMX6)

```
## Gerenating Keys
bruno@bruno:~/cst-2.3.2/keys$ ./hab4_pki_tree.sh

## Generating Key Table
bruno@bruno:~/cst-2.3.2/keys$ ../linux64/srktool –help

## Extracting fuse values
bruno@bruno:~/cst-2.3.2/keys$ hexdump -e '/4 "0x"' -e '/4 "%X""\n"' SRK_1_2_3_4_fuse.bin

## Programming fuses
Uboot: fuse prog <bank> <word> <value>

## Create a boot command file on xml-like format
bruno@bruno:~/cst-2.3.2/keys$ gedit csf-uboot

## Generate the signature binaty from the cs
bruno@bruno:~/cst-2.3.2/keys$ ./cst --o csf-uboot.bin < csf-uboot

## Concatenate the signature to u-boot
bruno@bruno:~/cst-2.3.2/keys$ cat u-boot.imx csf-uboot.bin > u-boot-signed.imx

## On u-boot, check boot status
Uboot: hab_status

## Burn last fuses
```

SOFTWARE DESIGN

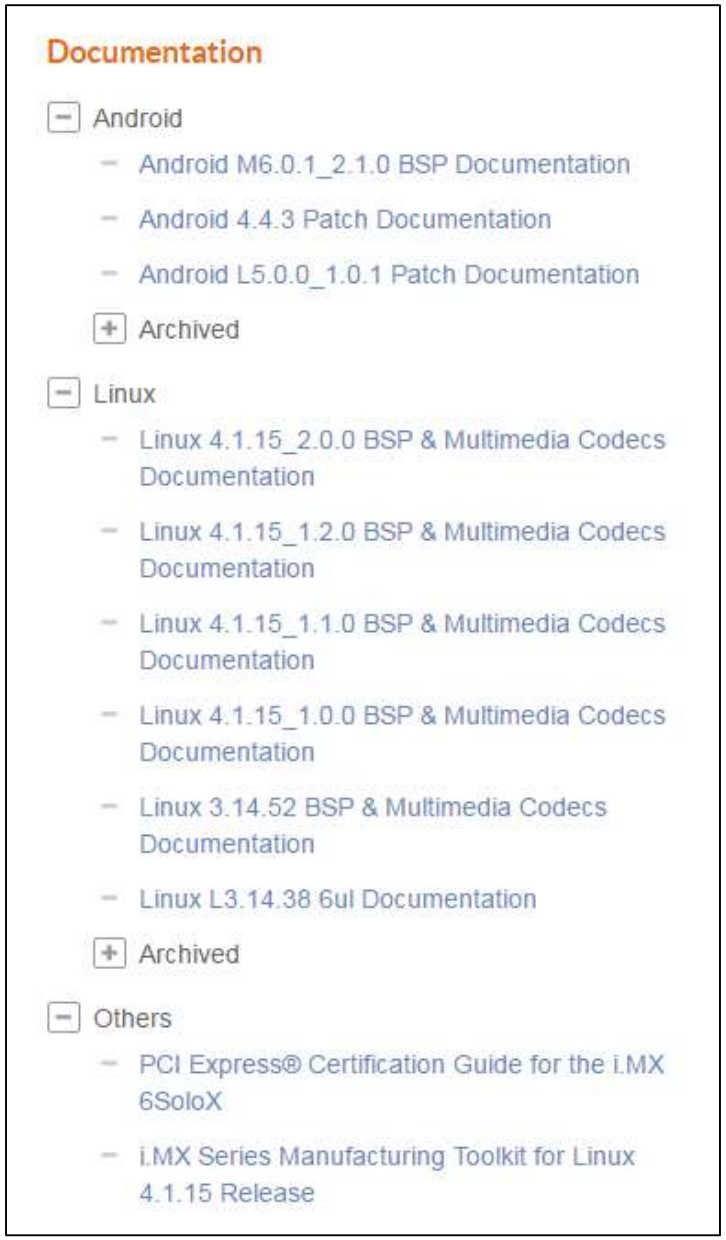
NXP Public GIT – git repository – link (all NXP products)

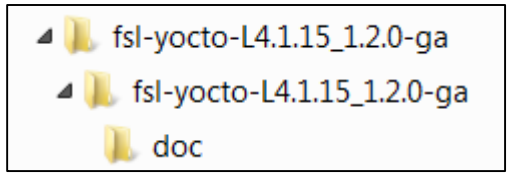
index about

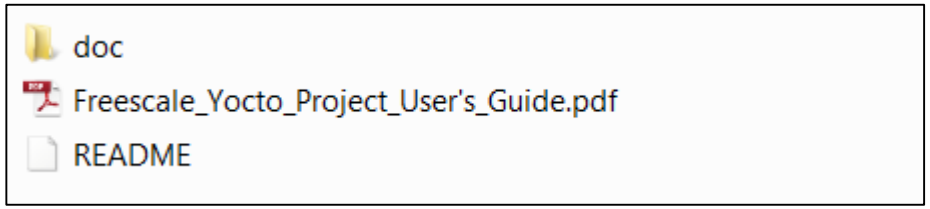
Name	Description
auto	
buildroot.git	Automotive Buildroot Tree
freertos.git	Automotive FreeRTOS Tree
linux.git	Automotive Linux Tree
u-boot.git	Automotive U-Boot Tree
crypto	
cryptodev.git	Freescale SEC upstream Linux driver tre
embedded-software	
usbdm-eclipse.git	Unnamed repository; edit this file 'descript
graphics	
svg_tool.git	Tool for converting an SVG image file to a
imx	
fsl-arm-yocto-bsp.git	i.MX Linux BSP Release Yocto Project m
imx-firmware.git	i.MX Firmware
linux-2.6-imx.git	Freescale i.MX Linux Tree
linux-imx.git	i.MX Linux Kernel
meta-fsl-bsp-release.git	i.MX BSP Yocto Project Release Layer
meta-nxp-agl.git	NXP AGL Yocto Project Demo Layer
meta-nxp-genivi.git	NXP Genivi Yocto Project Demo Layer
meta-nxp-imx-scm.git	NXP SCM i.MX Yocto Project Layer
meta-nxp-iotg.git	NXP IOTG Yocto Project Demo Layer
meta-nxp-xbmc.git	NXP XBMC Yocto Project Demo Layer
uboot-imx.git	Freescale i.MX u-boot Tree

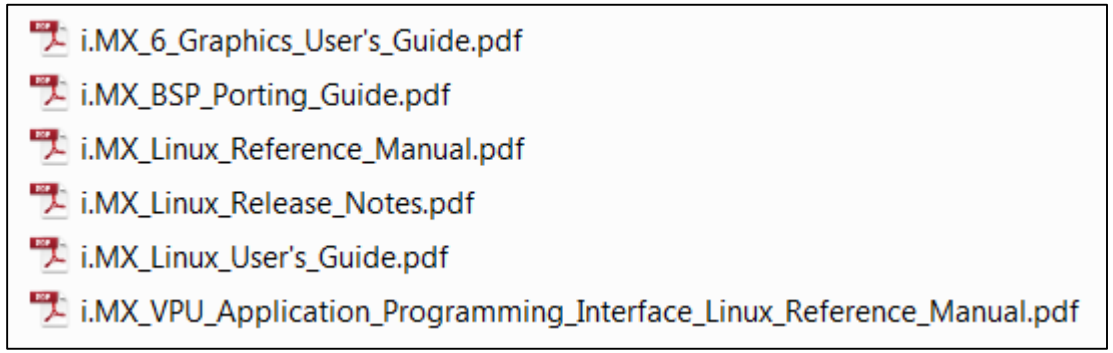
README	18593
REPORTING-BUGS	7485
arch	1033
block	1795
crypto	3861
drivers	3937
firmware	1734
fs	4968
include	873
init	508
ipc	506
kernel	4071
lib	7656
mm	3748
net	2033
samples	509
scripts	3974

Document Bundle - Tools Page – link (iMX6)









BSP User Guide – Docs Bundle – link (iMX6)

- Introduction
- Basic Terminal Setup
- Booting Linux OS
- Enabling Solo Emulation
- Power Management
- Multimedia
- Graphics
- Security
- Connectivity

- Booting Linux OS
  - Software overview
  - Manufacturing Tool
  - Preparing an SD/MMC card to boot
  - Downloading images
  - How to boot the i.MX boards
  - Flash memory maps
  - Running Linux OS on the target
- Enabling Solo Emulation

- Power Management
  - Suspend and resume
  - CPU frequency scaling
  - Bus frequency scaling

- Multimedia
  - Freescale multimedia packages
  - Building limited access packages
  - Multimedia use cases
    - Playback use cases
    - Audio encoding
    - Video encoding
    - Transcoding
    - Audio recording
    - Video recording
    - Audio/Video recording
    - Camera preview
    - Recording the TV-in source
    - Web camera
    - HTTP streaming
    - Real Time Streaming Protocol (RTSP) playback
    - RTP/UDP MPEGTS streaming
    - RTSP streaming server
    - Video conversion
    - Video composition
  - Pulseaudio input/output settings
  - Overlaysink usage
  - Installing gstreamer1.0-libav into rootfs

- Graphics
  - fsl-gpu-sdk
  - G2D-fsl-samples
  - viv_samples
  - Qt 5

BSP Release Notes - Tools Page – link (iMX6)

1 Release Contents

This document contains important information about the package contents, supported features, known issues and limitations in this release.

This release contains the latest updates for the last GA release. For more information on changes, see the manifest Readme at index: fsl-arm-yocto-bsp.git and the Change Log at index: fsl-arm-yocto-bsp.git.

Supported hardware SoC/board

- i.MX 6QuadPlus SABRE-SD Board and Platform
- i.MX 6QuadPlus SABRE-AI Board
- i.MX 6Quad SABRE-SD Board and Platform
- i.MX 6DualLite SABRE-SD Board
- i.MX 6Quad SABRE-AI Board
- i.MX 6DualLite SABRE-AI Board
- i.MX 6SoloLite EVK Board
- i.MX 6SoloX SABRE-SD Board
- i.MX 6SoloX SABRE-AI Board
- i.MX 7Dual SABRE-SD Board
- i.MX 6UltraLite EVK Board

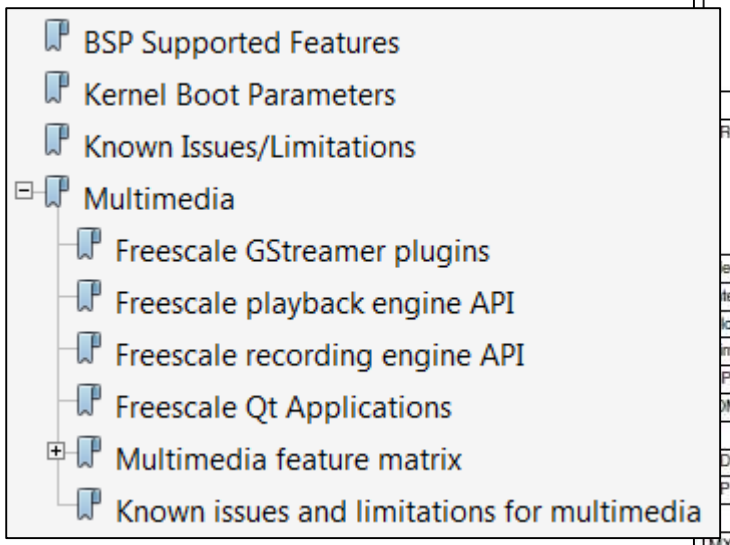


Table 10. Supported features (continued)

Feature	Supported board	Comment
		i.MX 6SoloLite EVK supports LPDDR2 400 MHz @ 32 bit and boot using L2Cache as OCRAM i.MX 7Dual SABRE-SD supports DDR3 533 MHz @ 32 bit and boot using L2Cache as OCRAM i.MX 6UltraLite EVK supports DDR3 400 Mhz @ 16 bit
Machine-specific layer		
RM® Core	All i.MX	6SABRE-SD, 6SABRE-AI, 6SoloLite, 6SoloX-SD, and 6SoloX-AI support the ARM Cortex-A9 processor. 7D-SABRE-SD supports the ARM Cortex-A7 processor. 6UltraLite EVK supports the ARM Cortex-A7 processor. Supports reboot and power-off.
emory	All i.MX	The user/kernel space is split 2G/2G.
terrupt	All i.MX	GIC.
lock	All i.MX	Controls the system frequency and clock tree distribution.
mer (GPT)	All i.MX	System timer tick support.
PIO/EDIO	All i.MX	GPIO is initialized in earlier phase according to hardware design.
OMUX	All i.MX	Provides the interfaces for I/O configuration. IOMUX-V3 version is used.
DMA engine		
DMA	All i.MX	Conforms to the DMA engine framework.
PBH-Bridge-DMA	6SABRE-AI	Conforms to the DMA engine framework. This feature requires a NAND U-Boot.
Character device drivers		
MXC UART	All i.MX	i.MX 6 SABRE-SD, and SoloLite EVK support console through internal Debug UART1. i.MX 6SoloX SABRE-SD and SABRE-AI support Cortex-A9 processor through UART1 and Cortex-M4 processor through UART2. i.MX 7Dual SABRE-SD Cortex-A7 processor through UART1 and Cortex-M4 processor through UART2. i.MX 6UltraLite EVK Cortex-A7 processor through UART1. i.MX 6 SABRE-AI supports console through internal Debug UART 4.
Power Management Drivers		
	All i.MX	Supports Anatop regulator management.
	All i.MX	Supports standby mode and dormant (mem) mode.
	All i.MX	2 levels CPUIdle supported: purely WFI and WFI with wait mode enabled.
	All i.MX	CPUFreq can be used for CPU frequency adjustment. The Interactive governor is added and enabled by default.
	All i.MX	Supports the system bus clock frequency scaling.
	All i.MX	-
Networking drivers		
	All i.MX	i.MX 6Quad/SoloX board supports AR8031 PHY, i.MX 6UltraLite EVK board supports KSZ8081 PHY, and i.MX 7Dual SABRE-SD board supports BCM54220 PHY. i.MX 6SoloX SABRE-SD, SABRE-AI, and i.MX 7Dual SABRE-SD support AVB Features.

Table 11. Common kernel boot parameters (continued)

Kernel parameter	Description	Typical value	Used when
video on 6SoloLite	Tells the ELCDIF FB driver which LCD panel is in use and which bpp should be used for the Frame Buffer.	video=mxc_elcdif_fb:SEIKO-WVGA,bpp=16	Used when enabling LCDIF to select the correct panel parameters to use. bpp=16 selects RGB565 FB pix format Note: if only use EPDC FB, then turn off ELCDIF FB by "video=mxc_elcdif_fb:off"
video on 7D SABRE-SD	Tells the ELCDIF FB driver which LCD panel is in use and which bpp should be used for the Frame Buffer.	video=mxcfb0:dev=mipi_dsi,TRULY-WVGA,if=RGB24	Used when primary displaying on the TRULY-WVGA MIPI DSI LCD panel.
dmfc	Tells the kernel/driver how to set the IPU DMFC segment size.	None Or dmfc=3	"dmfc=1" means DMFC_HIGH_RESOLUTION_DC. "dmfc=2" means DMFC_HIGH_RESOLUTION_DP. "dmfc=3" means DMFC_HIGH_RESOLUTION_ONLY_DP. DMFC_HIGH_RESOLUTION_ONLY_DP can only be set by the command line. It is recommended to set this when no IPU connects the two panels. When it is set, each IPU can only connect one panel.
mtdparts on 6SABRE-AI	Tells the kernel mtd partition information.	mtdparts=gpmi-nand:16m(boot),16m(kernel),1024m(rootfs),-(user)	When to enable NAND. The partition: 16m (boot),16m (kernel),1024m (rootfs) is an example, you can change it according to your needs.

Yocto User Guide - Tools Page – link (iMX6)

1 Overview

This document describes how to build an image for an i.MX Freescale board by using a Yocto Project build environment. It describes the Freescale release layer and Freescale-specific usage.

- imx6qpsabreauto
- imx6qpsabresd
- imx6ulevk
- imx6dlsabreauto
- imx6dlsabresd
- imx6qsabreauto
- imx6qsabresd
- imx6slevk
- imx6solosabreauto
- imx6solosabresd
- imx6sxsabresd
- imx6sxsabreauto
- imx7dsabresd

Table 1. Freescale Yocto project images (continued)

Image name	Target	Provided by layer
fsl-image-machine-test	An FSL Community i.MX core image with console environment - no GUI interface	meta-fsl-demos
fsl-image-gui	Builds a Freescale image with a GUI without any Qt content.	meta-fsl-bsp-release/imx/meta-sdk
fsl-image-qt5	Builds an opensource Qt 5 image. These images are only supported for i.MX SoC with hardware graphics. They are not supported on the i.MX 6UltraLite and i.MX 7Dual.	meta-fsl-bsp-release/imx/meta-sdk

Linux Reference Manual - Tools Page – link (iMX6)

- Chapter 3: Machine-Specific Layer (MSL)
- Chapter 4: Smart Direct Memory Access (SDMA) API
- Chapter 5: AHB-to-APBH Bridge with DMA (APBH-Bridge-DMA)
- Chapter 6: Image Processing Unit (IPU) Drivers
- Chapter 7: MIPI DSI Driver
- Chapter 8: LVDS Display Bridge(LDB) Driver
- Chapter 9: Video for Linux Two (V4L2) Driver
- Chapter 10: Electrophoretic Display Controller (EPDC) Frame Buffer Driver
- Chapter 11: Pixel Pipeline (PxP) DMA-ENGINE Driver
- Chapter 12: ELCDIF Frame Buffer Driver
- Chapter 13: Graphics Processing Unit (GPU)
- Chapter 14: Wayland
- Chapter 15: On-Chip High Definition Multimedia Interface (HDMI) Driver
- Chapter 16: External High-Definition Multimedia Interface (HDMI) for i.MX 6SoloLite
- Chapter 17: X Windows Acceleration
- Chapter 18: Video Processing Unit (VPU) Driver
- Chapter 19: OmniVision Camera Driver
- Chapter 20: MIPI CSI2 Driver
- Chapter 21: Low-level Power Management (PM) Driver
- Chapter 22: PF100 Regulator Driver
- Chapter 23: CPU Frequency Scaling (CPUFREQ) Driver
- Chapter 24: Dynamic Bus Frequency Driver
- Chapter 25: Thermal Driver
- Chapter 26: Anatop Regulator Driver
- Chapter 27: SNVS Real Time Clock (SRTC) Driver
- Chapter 28: Advanced Linux Sound Architecture (ALSA) System on a Chip (ASoC) Sound Driver
- Chapter 29: Asynchronous Sample Rate Converter (ASRC) Driver
- Chapter 30: The Sony/Philips Digital Interface (S/PDIF) Driver
- Chapter 31: SPI NOR Flash Memory Technology Device (MTD) Driver
- Chapter 32: MMC/SD/SDIO Host Driver
- Chapter 33: NAND GPMI Flash Driver
- Chapter 34: SATA Driver
- Chapter 35: Inter-IC (I2C) Driver
- Chapter 36: Enhanced Configurable Serial Peripheral Interface (ECSPI) Driver
- Chapter 37: FlexCAN
- Chapter 38: Media Lo
- Chapter 39: CHIPIDEA USB Driver
- Chapter 40: PCI Express Root Complex Driver
- Chapter 41: EIM NOR Driver
- Chapter 42: Quad Serial Peripheral Interface (QuadSPI) Driver
- Chapter 43: Fast Ethernet Controller (FEC) Driver
- Chapter 44: ENET IEEE-1588 Driver
- Chapter 45: Universal Asynchronous Receiver/Transmitter (UART) Driver
- Chapter 46: Wi-Fi BCM4339 Driver
- Chapter 47: Pulse-Width Modulator (PWM) Driver
- Chapter 48: Watchdog (WDOG) Driver
- Chapter 49: OProfile
- Chapter 50: CAAM (Cryptographic Acceleration and Assurance Module)
- Chapter 51: Remote Processor Messaging (RPMsg)
- Chapter 52: Display Content Integrity Checker (DCIC)
- Chapter 53: ADC Driver
- Chapter 54: Video Analog-to-Digital Converter (VADC)
- Chapter 55: Bluetooth® BCM4339 Driver
- Samsung MIPI DSI Driver
- Subscriber Identification Module (SIM) Driver

- Chapter 42: Quad Serial Peripheral Interface (QuadSPI) Driver
  - Introduction
  - Hardware Operation
  - Software Operation
  - Driver Features
  - Source Code Structure
  - Menu Configuration Options

Graphics User Guide - Tools Page – link (iMX6)

- Chapter 2 i.MX 6 G2D API
- Chapter 3 i.MX 6 EGL and OGL Extension Support
- Chapter 4 i.MX 6 Framebuffer API
- Chapter 5 OpenCL
- Chapter 6 XServer Video Driver
- Chapter 7 Vivante Software Tool Kit
- Chapter 8 GPU Tools
- Chapter 9 GPU Memory Introduction
- Chapter 10 Application Programming Recommendations
- Chapter 11 Demo Framework
- Chapter 12 Environment Variables Summary

- Chapter 10 Application Programming Recommendations
  - 10.1 Understand the system configuration and target application
  - 10.2 Optimize off chip data transfer such as accessing off-chip DDR memory/mobile DDR memory
  - 10.3 Avoid random cache or memory accesses
  - 10.4 Optimize your use of system memory
  - 10.5 Target a fixed frame rate that is visibly smooth
  - 10.6 Minimize GL state changes
  - 10.7 Batch primitives to minimize the number of draw calls
  - 10.8 Perform calculations per vertex instead of per fragment/pixel
  - 10.9 Enable early-Z, hierarchical-Z and back face culling
  - 10.10 Use branching carefully
  - 10.11 Do not use static or stack data as vertex data - use VBOs instead
  - 10.12 Use dynamic VBO if data is changing frame by frame
  - 10.13 Tessellate your data so that Hierarchical Z (HZ) can do its job

- Chapter 4 i.MX 6 Framebuffer API
  - 4.1 Overview
  - 4.2 API data types and environment variables
  - 4.3 API description and syntax
- Chapter 5 OpenCL
  - 5.1 Overview
  - 5.2 Vivante OpenCL Implementation
  - 5.3 Optimization for OpenCL Embedded Profile
  - 5.4 OpenCL Debug Messages
- Chapter 6 XServer Video Driver
  - 6.1 EXA driver
  - 6.2 XRandR
- Chapter 7 Vivante Software Tool Kit
- Chapter 8 GPU Tools
  - 8.1 gpuinfo tool
  - 8.2 gmem_info tool
  - 8.3 Apitrace user guide
- Chapter 9 GPU Memory Introduction
  - 9.1 GPU memory overview
  - 9.2 GPU memory pools
  - 9.3 GPU memory allocators
  - 9.4 GPU reserved memory
  - 9.5 GPU memory base address

Device Reference Manual - Device Page – link (iMX6UL) - about 4000 pages

Chip Errata - Device Page – link (iMX6UL)

Table 2. Summary of Silicon Errata

Errata	Name	Solution	Page
	ARM® – Cortex A7		
ERR008958	ARM/MP: 814220—B-Cache maintenance by set/way operations can execute out of order	No fix scheduled	5
ERR008959	ARM/MP: 809719—C PMU events 0x07, 0x0C, and 0x0E do not increment correctly	No fix scheduled	7
ERR008960	ARM/MP: 805420—C PMU event counter 0x14 does not increment correctly	No fix scheduled	9
ERR008961	ARM/MP: 804069—C Exception mask bits are cleared when an exception is taken in Hyp Mode	No fix scheduled	10
	CCM		
ERR007265	CCM: When improper low-power sequence is used, the SoC enters low power mode before the ARM core executes WFI	No fix scheduled	11
	Clock		
ERR009455	Clock: 24 MHz Oscillator does not start up	Fixed in silicon revision 1.1	12
	eCSPI		
ERR009606	eCSPI: In master mode, burst lengths of 32n + 1 will transmit incorrect data	No fix scheduled	13
ERR009535	eCSPI: Burst completion by SS signal in slave mode is not functional	No fix scheduled	14
	EIM		
ERR004446	EIM: AUS mode is nonfunctional for devices larger than 32 MB	No fix scheduled	15
	FlexCAN		
ERR005829	FlexCAN: FlexCAN does not transmit a message that is enabled to be transmitted in a specific moment during the arbitration process	No fix scheduled	16
	I2C		
ERR007805	I2C: When the I2C clock speed is configured for 400 kHz, the SCL low period violates the I2C specification	No fix scheduled	18
	MMDC		
ERR005778	MMDC: DDR Controller's measure unit may return an incorrect value when operating below 100 MHz	No fix scheduled	19
ERR009596	MMDC: ARCR_GUARD bits of MMDC Core AXI Re-ordering Control register (MMDC_MAARCR) does not behave as expected	No fix scheduled	20

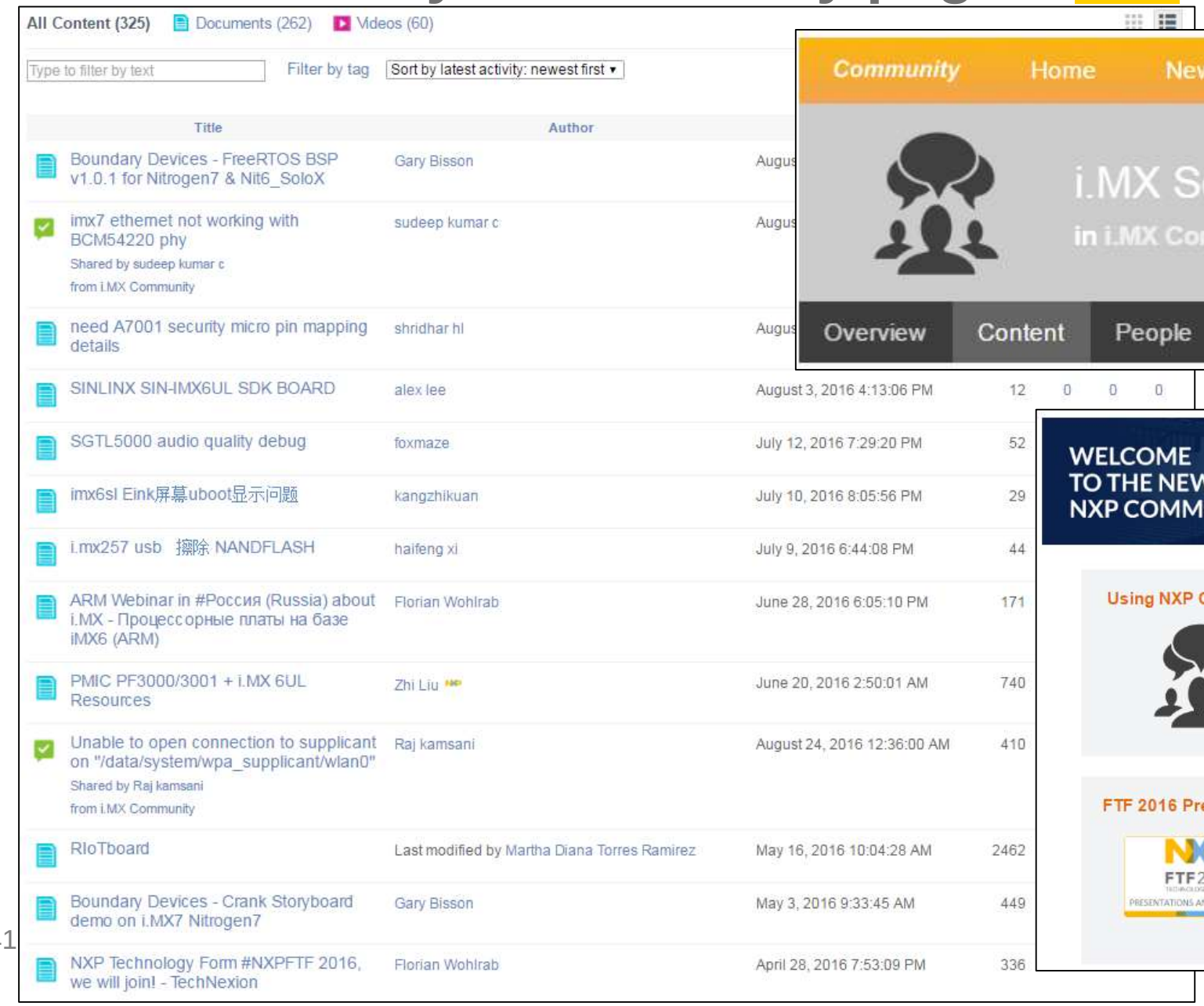
Application Notes - Device Page – link (iMX6Q)

Application Notes (17)

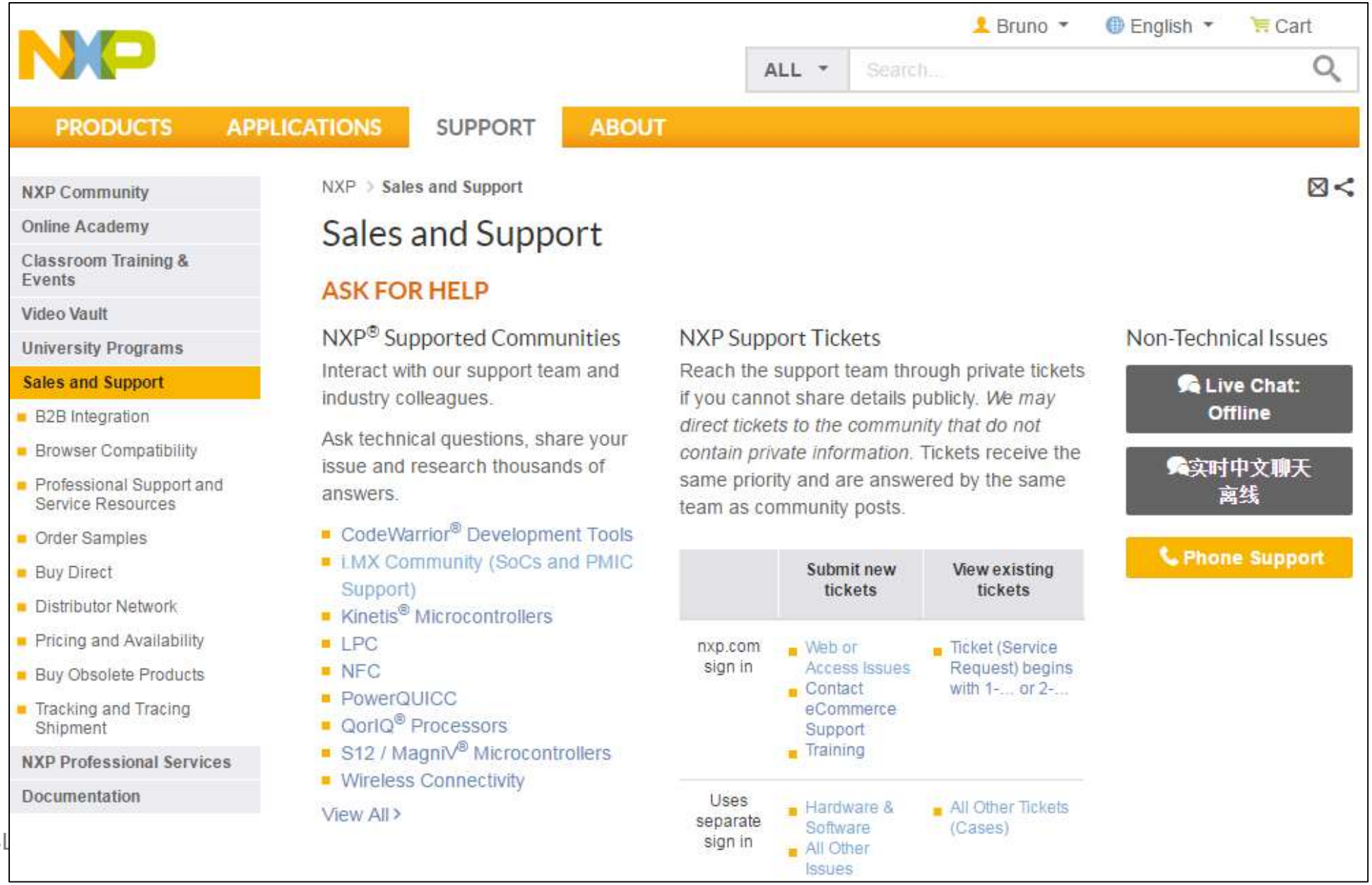
Name/Description	
i.MX 6Quad/6Dual Personality Fuses **(REV 0)** PDF (53.6 kB) AN5323 [English]	
MIPI–CSI2 Peripheral on i.MX6 MPUs **(REV 0)** PDF (862.5 kB) AN5305 [English]	
i.MX 6 Temperature Sensor Module - Application Note **(REV 0)** PDF (507.2 kB) AN5215 [English]	
Secure Boot on i.MX50, i.MX53, and i.MX 6 Series using HABv4 **(REV 1)** PDF (867.8 kB) AN4581 [English]	
Common Hardware Design for i.MX 6Dual/6Quad and i.MX 6Solo/6DualLite **(REV 2)** PDF (322.8 kB) AN4397 [English]	
Configuring Secure JTAG for the i.MX 6 Series Family of Applications Processors **(REV 1)** PDF (560.1 kB) AN4686 [English]	
Fast Image Processing with i.MX 6 Series **(REV 1)** PDF (1.2 MB) AN4629 [English]	
i.MX 6 Series DDR Calibration **(REV 2)** PDF (426.3 kB) AN4467 [English]	
Configuring USB on i.MX 6 Series Processors **(REV 1)** PDF (175.2 kB) AN4589 [English, 中文]	
Influence of Pin Setting on System Function and Performance **(REV 0)** PDF (687.7 kB) AN5078 [English]	09 Feb 2015
AN4724, i.MX 6Dual/6Quad/6DualPlus/6QuadPlus Product Usage Lifetime Estimates - Application Note **(REV 2)** PDF (264.1 kB) AN4724 [English]	06 Aug 2014
i.MX 6 Audio Clock Configuration Options **(REV 0)** PDF (1.3 MB) AN4952 [English]	26 Jun 2014
AN4784: PCIe Certification Guide for i.MX 6Dual/6Quad and i.MX 6Solo/6DualLite - Application Note **(REV 0)** PDF (271.7 kB) AN4784 [English]	14 Oct 2013
AN4671, i.MX 6 Series HDMI Test Method for Eye Pattern and Electrical Characteristics - Application Notes **(REV 0)** PDF (432.4 kB) AN4671 [English]	22 Apr 2013
i.MX 6 Series Thermal Management Guidelines **(REV 0)** PDF (3.7 MB) AN4579 [English]	10 Dec 2012
i.MX 6Dual/6Quad Power Consumption Measurement **(REV 0)** PDF (476.0 kB) AN4509 [English]	24 Oct 2012
Using Open Source Debugging Tools for Linux on i.MX Processors **(REV 0)** PDF (251.5 kB) AN4553 [English]	19 Jul 2012

SUPPORT RESOURCES

i.MX Community – community page – link – (all i.MX)

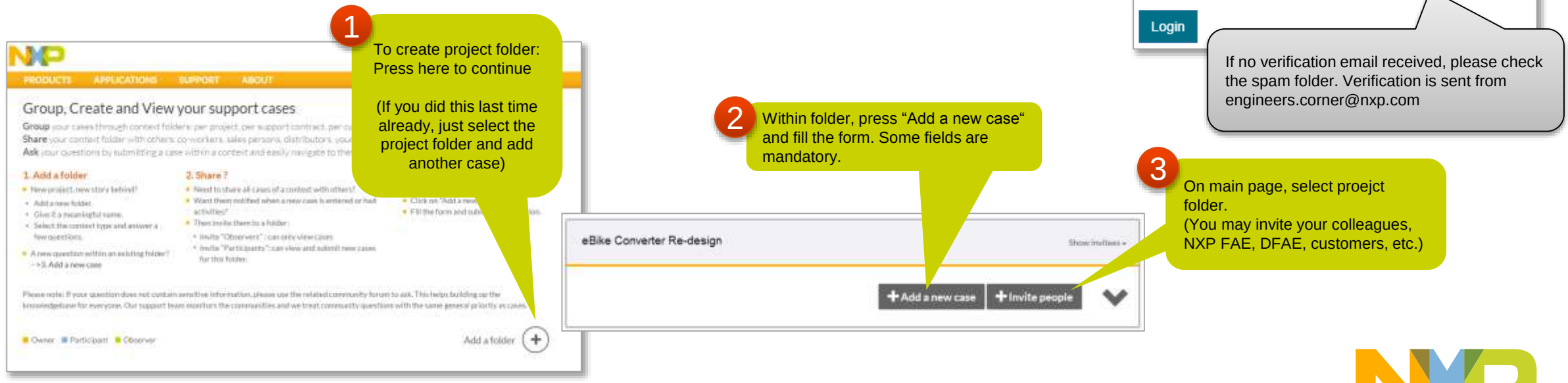


Support Center – Cases- support page – link (all NXP products)



Enter a CASE to get Online Technical Support from NXP

- Use this link: http://nxpcommunity.force.com/community/CommunityContactSupport
- Use your NXP TechSupport credentials (NOT the same like the www.nxp.com login !)
  or Request Password if you forget yours
  or Register as new user
- Make sure you have and select a support “context folder” for this project. If not, create it first.
- Enter your support CASE within this folder



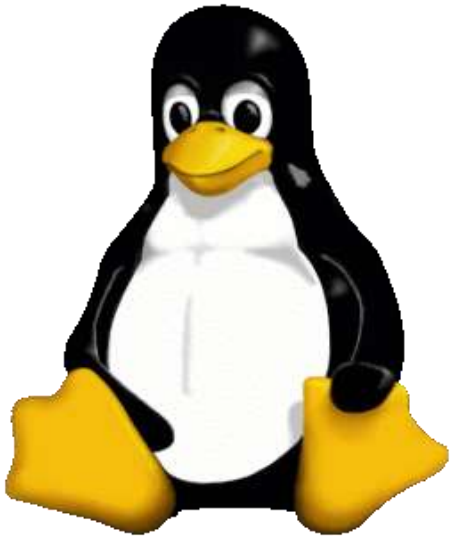
PARTNERS

Hardware SoM Partners

- Hardware & Software complexity.
- Time to market
- Support

Software Partners

- Time to market
- Yocto BSP
- Hardware + Software



yocto PROJECT

Training Partners

- Linux
  - Embedded Linux
  - Device Drivers
  - Programação
  - Debugging
- Yocto
- Android
- FreeRTOS

SECURE CONNECTIONS
FOR A SMARTER WORLD