IMXDCHPE

i.MX Digital Cockpit Hardware Partitioning Enablement for i.MX 8QuadMax

Rev. LF6.6.3_1.0.0 — 29 March 2024   **User guide**

Document information

Information	Content
Keywords	i.MX, Linux, LF6.6.3_1.0.0
Abstract	This document describes the i.MX Digital Cockpit hardware partitioning enablement for i.MX 8QuadMax.

1   Introduction

The main feature of Cockpit project is the hardware isolation.

Each OS has its own hardware resources.

One partition is running on Arm Cortex-A53 cluster for Instruent Cluster, and one is running on Arm Cortex-A72 cluster for In-Vehicle Infotainment, while two others are running RTOS on Arm Cortex-M4s.

Instrument Cluster's High Level Operating System (HLOS) uses SD card in the CPU board slot as boot media. In-Vehicle Infotainment's HLOS uses eMMC as boot media. For this reason, two SD card images must be built:

- One with a standard Linux BSP used for flashing the other image.
- One used at runtime for booting the board. The image contains firmware, HLOS images for Cortex A-53, Cortex-A72, and optionally Cortex-M4s.

All BSP components are modified to allow running two HLOS and two RTOS in a secure way.

- SCU Firmware board file
- Arm Trusted Firmware
- U-Boot/SPL
- mkimage
- Linux Kernel
- TEE

The Cockpit architecture provides a way to implement two different ECU software systems in a single SoC ECU.

The ECU software systems enforce different safety levels, so it is important to isolate them from each other to avoid interference. The Digital Cockpit concept implements strong hardware isolation without the need for a Hypervisor.

Hardware isolation is under the control of the System Controller Unit (SCU) secured with Extended Resource Domain Controller (XRDC).

This isolation is implemented by:

- Defining hardware partitions, each with its own resources.
- Defining Memory regions.
- Assigning IO pads to hardware partitions.
- Routing IRQs to their respective subsystem.

The following figure shows how hardware resources are allocated between the two partitions in this release.
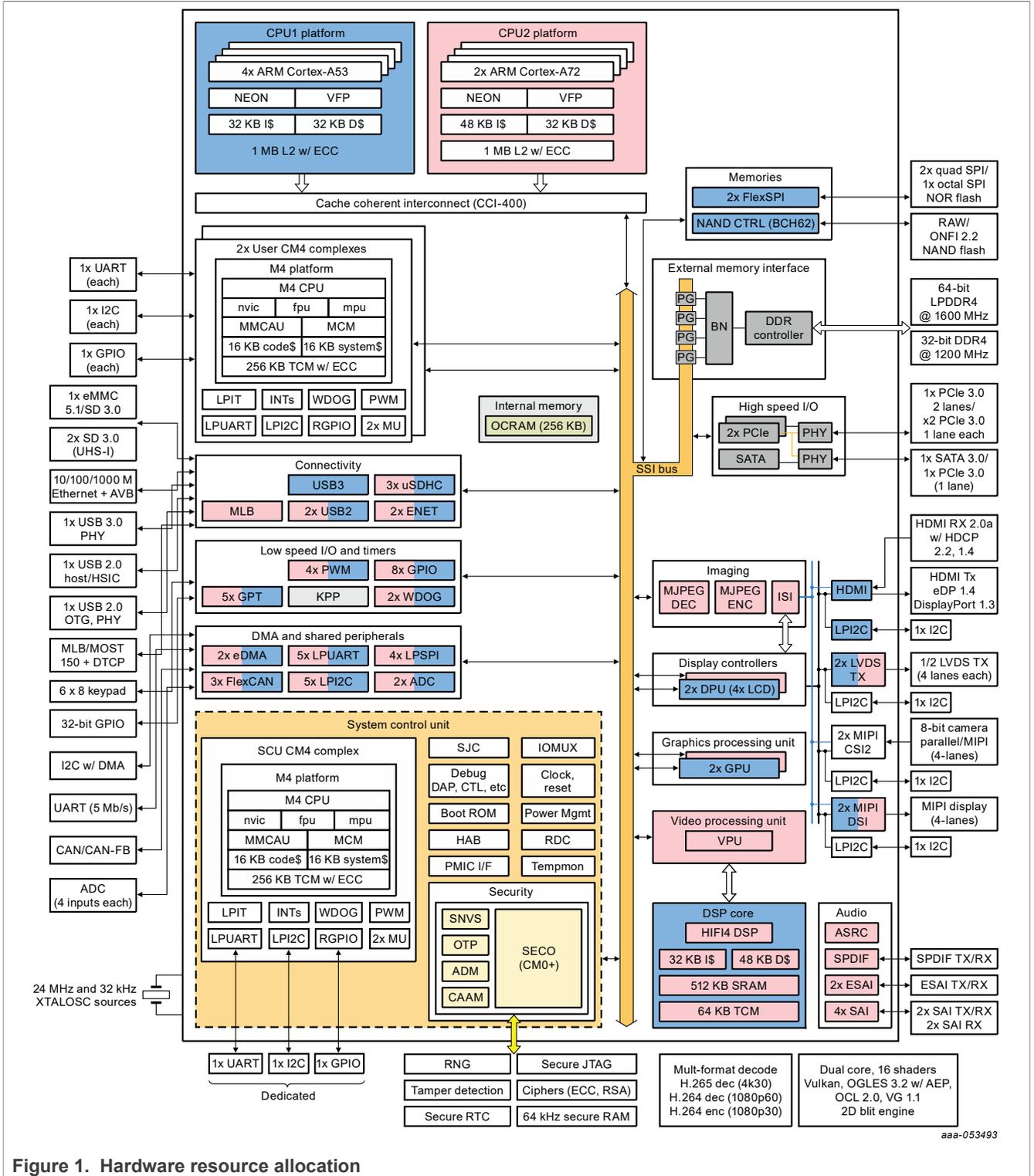
Resources allocated to Cortex-A53 cluster (IC) are colored in blue, those allocated to Cortex-A72 cluster (IVI) are in red, and the resources uncolored are assigned by default to Cortex-A53's partition.

LPUART0 is assigned to the Cortex-A53 cluster, whereas LPUART2 is assigned to the Cortex-A72 cluster.

Each cluster has one identical display controller and one identical GPU, GC7000.

Only one GIC is available and allocated to SCU but made accessible to both Clusters. Interrupts dedicated to a Cortex-A cluster are only routed to it.

IMXDCHPE

User guide

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

Rev. LF6.6.3_1.0.0 — 29 March 2024

2 / 15

CPU1 platform
- 4x ARM Cortex-A53
- NEON | VFP
- 32 KB I$ | 32 KB D$
- 1 MB L2 w/ ECC

CPU2 platform
- 2x ARM Cortex-A72
- NEON | VFP
- 48 KB I$ | 32 KB D$
- 1 MB L2 w/ ECC

Cache coherent interconnect (CCI-400)

2x User CM4 complexes
M4 platform
M4 CPU
nvic | fpu | mpu
MMCAU | MCM
16 KB code$ | 16 KB system$
256 KB TCM w/ ECC
LPIT | INTs | WDOG | PWM
LPUART | LPI2C | RGPIO | 2x MU

- 1x UART (each)
- 1x I2C (each)
- 1x GPIO (each)
- 1x eMMC 5.1/SD 3.0
- 2x SD 3.0 (UHS-I)
- 10/100/1000 M Ethernet + AVB
- 1x USB 3.0 PHY
- 1x USB 2.0 host/HSIC
- 1x USB 2.0 OTG, PHY
- MLB/MOST 150 + DTCP
- 6 x 8 keypad
- 32-bit GPIO
- I2C w/ DMA
- UART (5 Mb/s)
- CAN/CAN-FB
- ADC (4 inputs each)

Connectivity
- USB3
- 3x uSDHC
- MLB
- 2x USB2
- 2x ENET

Low speed I/O and timers
- 4x PWM
- 8x GPIO
- 5x GPT
- KPP
- 2x WDOG

DMA and shared peripherals
- 2x eDMA
- 5x LPUART
- 4x LPSPI
- 3x FlexCAN
- 5x LPI2C
- 2x ADC

Internal memory
- OCRAM (256 KB)

SSI bus

Memories
- 2x FlexSPI
- NAND CTRL (BCH62)

External memory interface
- PG / PG / PG / PG
- BN
- DDR controller

High speed I/O
- 2x PCIe | PHY
- SATA | PHY

System control unit
SCU CM4 complex
M4 platform
M4 CPU
nvic | fpu | mpu
MMCAU | MCM
16 KB code$ | 16 KB system$
256 KB TCM w/ ECC
LPIT | INTs | WDOG | PWM
LPUART | LPI2C | RGPIO | 2x MU

- SJC
- IOMUX
- Debug DAP, CTL, etc
- Clock, reset
- Boot ROM
- Power Mgmt
- HAB
- RDC
- PMIC I/F
- Tempmon

Security
- SNVS
- OTP
- ADM
- CAAM
- SECO (CM0+)

24 MHz and 32 kHz XTALOSC sources

- 1x UART | 1x I2C | 1x GPIO
- Dedicated

- RNG
- Secure JTAG
- Tamper detection
- Ciphers (ECC, RSA)
- Secure RTC
- 64 kHz secure RAM

Imaging
- MJPEG DEC
- MJPEG ENC
- ISI

Display controllers
- 2x DPU (4x LCD)

Graphics processing unit
- 2x GPU

Video processing unit
- VPU

- HDMI
- LPI2C
- 2x LVDS TX
- LPI2C
- 2x MIPI CSI2
- LPI2C
- 2x MIPI DSI
- LPI2C

DSP core
- HIFI4 DSP
- 32 KB I$ | 48 KB D$
- 512 KB SRAM
- 64 KB TCM

Audio
- ASRC
- SPDIF
- 2x ESAI
- 4x SAI

- 2x quad SPI/ 1x octal SPI NOR flash
- RAW/ ONFI 2.2 NAND flash
- 64-bit LPDDR4 @ 1600 MHz
- 32-bit DDR4 @ 1200 MHz
- 1x PCIe 3.0 2 lanes/ x2 PCIe 3.0 1 lane each
- 1x SATA 3.0/ 1x PCIe 3.0 (1 lane)
- HDMI RX 2.0a w/ HDCP 2.2, 1.4
- HDMI Tx eDP 1.4 DisplayPort 1.3
- 1x I2C
- 1/2 LVDS TX (4 lanes each)
- 1x I2C
- 8-bit camera parallel/MIPI (4-lanes)
- 1x I2C
- MIPI display (4-lanes)
- 1x I2C
- SPDIF TX/RX
- ESAI TX/RX
- 2x SAI TX/RX 2x SAI RX

Mult-format decode
H.265 dec (4k30)
H.264 dec (1080p60)
H.264 enc (1080p30)

Dual core, 16 shaders
Vulkan, OGLES 3.2 w/ AEP,
OCL 2.0, VG 1.1
2D blit engine

aaa-053493

Figure 1. Hardware resource allocation

IMXDCHPE

User guide

All information provided in this document is subject to legal disclaimers.

Rev. LF6.6.3_1.0.0 — 29 March 2024

© 2024 NXP B.V. All rights reserved.

3 / 15

2 System Boot Sequence

2.1 System boot sequence

The partitioning for the subsystems is defined at boot time and is immutable after the system is booted.

The following sequence diagram shows the **Secure Boot** flow across different CPUs.
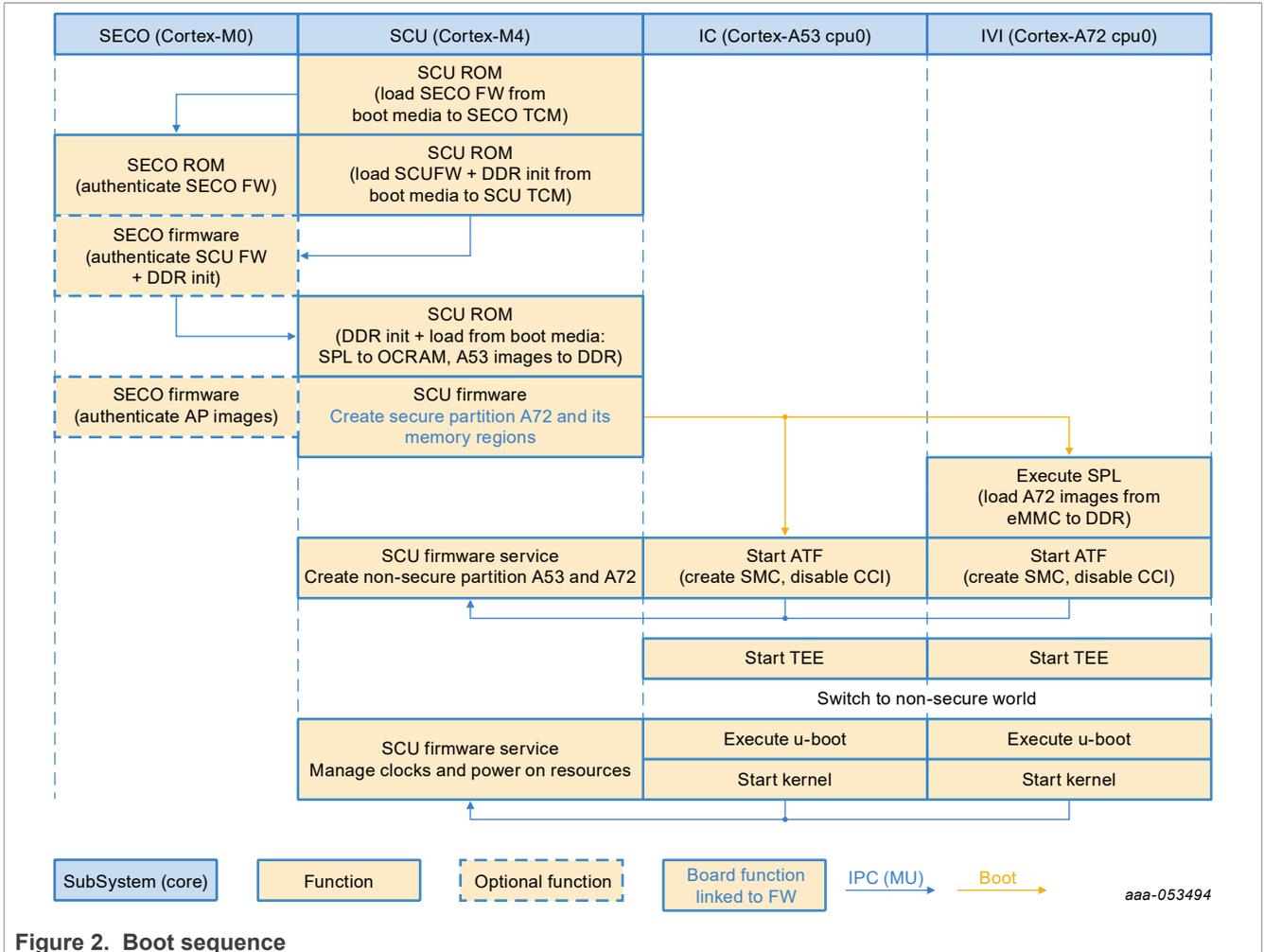

Figure 2. Boot sequence

2.2 Boot details

2.2.1 SECO boot

The images that will be loaded as part of the boot process are stored in containers. A container contains multiple images, as blocks of data specifying the load address. The containers can be authenticated, see Section Signing boot images in Digital Cockpit (e.g., in certain lifecycle states, the SECO FW needs to be authenticated, but the SCU FW can be loaded even it is not signed).

At least three containers are needed for the boot process:

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

4 / 15

- The first container with the SECO FW image only. This container is NXP signed, and NXP provided.
- The second container (can be signed by customer using OEM SRKs) that typically contains:
  - SCU FW
  - DDR initialization image (optional)
  - Images for the user-controlled Cortex-M4s (optional)
  - The TEE image for Cortex-A53 (optional)
  - The U-Boot-SPL for Cortex-A72 cluster.
  - Boot image for each Cortex-A53 cluster, containing:
    - Arm Trusted Firmware (BL31)
    - U-boot
- The third container (can also be signed by customer using OEM SRKs) that contains:
  - TEE image for Cortex-A72 (optional)
  - Boot image for Cortex-A72 cluster, containing:
    - Arm Trusted Firmware (BL31)
    - U-Boot

The second and third containers containing Cortex-A boot images can be encrypted, but in this release, they are not encrypted by default.

2.2.2 SCU boot

SCU ROM supports the following boot devices:

- FlexSPI NOR Flash (on FlexSPI0)
- NAND Flash
- SD/MMC
- USB

The current release boots from the SD card slot and eMMC located on the CPU board.

In the i.MX 8QuadMax processor B0 revision, the SCU ROM loads the following from the SD card:

1. Cortex-A53's boot image at 0x8000_0000 in DDR (ATF, TEE, U-Boot)
2. Cortex-A72 SPL at address 0x0 in OCRAM
3. Cortex-M4s images in their respective TCM or into DDR RAM.

The Cortex-A72 boots at address 0x0 and executes SPL, which then loads the Cortex-A72's images (ATF, TEE, U-Boot) at 0xC000_0000 from eMMC.

Alternatively, it is possible to revert to a legacy boot mode (without SPL and TEE) where SCU ROM loads the **two** Cortex-A boot images (ATF and U-Boot) from the SD card, respectively Cortex-A53's image at 0x8000_0000, and Cortex-A72's image at 0xC000_0000 in DDR RAM. In this legacy mode, the TEE for Cortex-A72 cannot be loaded, as its address (0xFE00_0000) is not reachable by SCU ROM.

SCU enforces the hardware partitioning by setting the Hardware Resources, Memory Regions and IO pads.

The partitioning is done in the *Board System Config* phase of the SCU Firmware initialization, which is implemented in the board specific customizable source file. At this step, a second secure partition is created for the Cortex-A72 cluster.

Note:

Each ATF instance will later create a non-secure partition and assign most resources (hardware peripheral and hardware accelerators) to it.

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

5 / 15

SCU Firmware releases Cortex-A53 from reset at address 0x8000 0000, and executes *Board Init* phase, which is also implemented in the board specific customizable source file. Similarly, Cortex-A72 is released from reset at 0x0 (OCRAM SPL) or 0x80000000 depending on compile option of SCFW (NOSPL=1).

2.2.3 Hardware partitions

Table 1. Hardware partitions

Partition number	Secure/non-secure	Comments
0	N/A	Contains all hardware resources at SCU boot
1	Secure	All hardware resources are given to this partition used by Cortex-A53 cluster
2	Secure	Contains Crypto, CAAM resources, used by SECO
3	Secure	Contains all the hardware resources that are removed afterwards from Cortex-A53 partition 1, for Cortex-A72 cluster
4	Non-secure	Contains only a shared memory region for Cortex-A53 to Cortex-A72 communication
5	Non-secure	Contains all the hardware resources required by Cortex-M4 core 0
6	Non-secure	Contains all the hardware resources required by Cortex-M4 core 1
7	Non-secure	Contains only a shared memory region for Cortex-A53 to Cortex-M4 communication
8	Non-secure	Depending on boot order, contains all hardware resources for Cortex-A53 or Cortex-A72 cluster
9	Non-secure	Depending on boot order, contains all hardware resources for Cortex-A53 or Cortex-A72 cluster

2.2.4 Cortex-A53 (IC) boot details

SCU ROM loads ATF and U-Boot for Cortex-A53 cluster from boot media into RAM.

Cortex-A53 is booting first as it runs the Instrument Cluster, which has a faster boot time requirement than Infotainment.

Cortex-A53 core#0 is booting at address 0x8000 0000 where ATF and U-Boot resides.

ATF is a secure partition and it creates a non-secure partition moving most of the hardware resources to this partition through the SCUFW service.

U-Boot loads the Linux kernel from the SD card, and then executes it.

2.2.5 Cortex-A72 (IVI) boot details

SCU ROM loads SPL for the Cortex-A72 cluster from the boot media into OCRAM.

Cortex-A72 core#0 is booting at address 0x0, an alias for OCRAM address where SPL resides. Cortex-A72 executes SPL, which loads TEE at 0xFE00_0000, loads ATF and U-boot at address 0xC000_0000, and then Cortex-A72 jumps to ATF. Similar to Cortex-A53, ATF creates a non-secure partition and moves most of the hardware resources to this partition through SCUFW service.

2.2.6 Cortex-M4s (RPMSG demo) boot details

SCU ROM loads a FreeRTOS image on both M4_0 and M4_1 from boot media into their respective TCM.

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

6 / 15

The Cortex-M4 images contains a demonstration of a simple "ping-pong" communication with Cortex-A53 using an RPMSG channel. Those Cortex-M4 images are pre-built images downloaded by the Yocto recipe.

3 Linux BSP Integration

Follow the instructions on the meta-cockpit README in meta-imx layer for integrating cockpit into a Linux BSP build.

3.1 Digital Cockpit hardware requirements

- 1 x Linux Host Machine with a minimum 120 GB HDD space available and Internet connection
- 1 x [MCIMX8QM-CPU](#)
- 1 x [MCIMX8-8X-BB](#)
- 1 x SD card (tested on 16 GB SD card)
- 2 x HDMI displays
- 2 x [IMX-LVDS-HDMI](#)
- 1 x IMX-MIPI-HDMI (optional)

4 Signing boot images in Digital Cockpit

See the [Secure Boot on i.MX 8 and i.MX 8X Families using AHAB](#) for a detailed explanation of the signing procedure.

The [CST tool version 3.3.0](#) is required for this release. Download and install the CST tool before proceeding. The following instructions assume that the environment variable $CST contains the CST tool location (ex: `export CST=/home/user/CST/release/linux64/bin/cst`).

The image signing step must be performed **after** the generation of the boot image by the tool 'mkimage'.

4.1 Signing SCFW and Cortex-A53 boot image

Parse the log of the building step of boot image (located at `yocto/build/tmp/work/imx8qmmekcockpit-poky-linux/imx-boot/1.0-r0/temp/log.do_compile`), and search for offset values on the same line as '*CST: CONTAINER 0*' after the key word Output: flash.bin.

As recommended in Secure Boot document, prepare a CST configuration file named `csf_boot_image.txt`.

Store the previously found offset values in this file.

The following is an example of resulting configuration file:

```
[Header]
Target = AHAB
Version = 1.0
[Install SRK]
# SRK table generated by srktool
File = "/home/b47544/cockpit/official/CST/release/crts/SRK_1_2_3_4_table.bin"
# Public key certificate in PEM format
Source = "/home/b47544/cockpit/official/CST/release/crts/
SRK1_sha384_secp384r1_v3_usr_crt.pem"
# Index of the public key certificate within the SRK table (0 .. 3)
Source index = 0
# Type of SRK set (NXP or OEM)
Source set = OEM
# bitmask of the revoked SRKs
```

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

7 / 15

```
Revocations = 0x0
[Authenticate Data]
# Binary to be signed generated by mkimage
File = "iMX8QM/flash.bin"
# Offsets = Container header  Signature block (printed out by mkimage)
Offsets   = 0x400             0x710
```

Put your file in the mkimage folder for the i.MX 8QuadMax, and run the following command:

```
$CST -i iMX8QM/csf_boot_image.txt -o iMX8QM/signed-boot-spl-container-a53.img
```

Then replace the old boot image with the signed one:

```
mv iMX8QM/signed-boot-spl-container-a53.img iMX8QM/flash.bin
```

4.2 Signing the Cortex-A72 boot image

By default, the Cortex-A72 boot image container (containing ATF, U-Boot, and optionally OP-TEE) is loaded and authenticated by U-Boot-SPL executed by Cortex-A72.

CAUTION:

A limitation in the AHAB firmware version 2.6.1 and older versions does not allow authenticating the boot image by SPL-A72 (as Cortex-A72 does not belong to domain ID 0 or 1). For those AHAB versions, if authentication is needed, it is recommended to switch to a legacy boot mode where Cortex-A72 and Cortex-A53 images are both loaded and authenticated by SCU (domain ID 0).

To enable the authentication by U-Boot-SPL-A72, modify the configuration file `<yocto>/build/ tmp/work/imx8qmmekcockpit-poky-linux/u-boot-imx-a72/2021.04-r0/git/configs/ imx8qm_mek_cockpit_a72_defconfig` and add the line CONFIG_AHAB_BOOT=y.

Parse the log of the building step of boot image (located at `yocto/build/tmp/work/imx8qmmekcockpit- poky-linux/imx-boot/1.0-r0/temp/log.do_compile`), and search for offset values on the same line as '*CST: CONTAINER 0*' after the key word `Output: u-boot-atf-container-a72.img`.

As recommended in Secure Boot document, prepare a CST configuration file named `cst_uboot_atf_a72.txt`.

Store the previously found offset values in this file.

The following is an example of resulting configuration file:

```
[Header]
Target = AHAB
Version = 1.0
[Install SRK]
# SRK table generated by srktool
File = "/home/b47544/cockpit/official/CST/release/crts/SRK_1_2_3_4_table.bin"
# Public key certificate in PEM format on this example only using SRK key
Source = "/home/b47544/cockpit/official/CST/release/crts/
SRK1_sha384_secp384r1_v3_usr_crt.pem"
# Index of the public key certificate within the SRK table (0 .. 3)
Source index = 0
# Type of SRK set (NXP or OEM)
Source set = OEM
# bitmask of the revoked SRKs
Revocations = 0x0
[Authenticate Data]
# Binary to be signed generated by mkimage
```

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

8 / 15

```
File = "iMX8QM/u-boot-atf-container-a72.img"
# Offsets = Container header  Signature block (printed out by mkimage)
Offsets  = 0x0              0x110
```

Put your file in the mkimage folder for the i.MX 8QuadMax, and run the following command:

```
$CST -i iMX8QM/cst_uboot_atf_a72.txt -o iMX8QM/signed-u-boot-atf-container-
a72.img
```

Then append the signed Cortex-A72 container after the Cortex-A53 image container using the following script:

```
flashbinsize=$(wc -c iMX8QM/signed-boot-spl-container-a53.img | awk '{print
 $1}')
padcnt=$((($flashbinsize + 0x400 - 1) / 0x400)) dd if=iMX8QM/signed-u-boot-atf-
container-a72.img
of=iMX8QM/flash.bin bs=1K seek=$padcnt
```

5   SD/eMMC Image Flashing

This chapter explains how to flash the two OSes on the SD card and eMMC.

5.1 Flashing dual-Linux OS images

To simplify the build and flashing process, the same generated image can be used for the two OSes: this is possible because Cortex-A53 only loads the first and second containers of the boot image, while Cortex-A72 only loads the third container of the boot image.

The result of the build process is a compressed image which can be found in the following location:

```
tmp/deploy/images/imx8qmmekcockpit/imx-image-<type>-imx8qmmekcockpit-
<timestamp>.wic.zst
```

where <type> is one of *multimedia* or *full*, depending on whether you've built an image with or without OpenCV + QT5 + Machine Learning packages, and <timestamp> is the image timestamp (i.e., 20200509080732). Depending on the configuration chosen for IVI cluster, one additional image is available to facilitate flashing the eMMC used by the Cortex-A72 cluster.

1. First step is to use a standard Yocto build to flash the eMMC for Cortex-A72 partition. Download a Yocto image that supports the i.MX 8QuadMax chip, e.g., https://www.nxp.com/webapp/sps/download/license.jsp?colCode=L5.10.9_1.0.0_MX8QM&app Type=file1&location=null&DOWNLOAD_ID=null

2. Unzip the downloaded file `LF5.10.72_2.2.0_images_MX8QMMEK.zip` inside the created directory, decompress and flash `fsl-image-validation-imx-imx8qmmek.sdcard` to the SD card:

```
$: sudo dd if=fsl-image-validation-imx-imx8qmmek.wic of=/dev/path/to/sdcard/
device bs=1M
```

3. Enlarge the second partition to at least 4 GB using a tool like GParted. It is necessary to copy flashing files. Copy the image generated from the Yocto build for Cockpit solution to the root filesystem partition from the SD Card.

```
$: sudo cp imx-image-<type>- imx8qmmekcockpit-<timestamp>.wic.zst /path/to/
SDcard/2nd_partition/home/root/
```

4. Boot the SD Card and flash the eMMC:

```
#: sudo dd if=imx-image-<type>-imx8qmmekcockpit-<timestamp>.wic of=/
dev/mmcblk0 bs=1M && sync
```

5. Re-flash the SD Card with the image used for the Cortex-A53 cluster:

```
$: zstdcat  imx-image-<type>-imx8qmmekcockpit-<timestamp>.wic.zst | sudo dd
 of=/dev/path/to/sdcard/device bs=1M && sync
```

5.2 Flashing the bootloader using UUU

Put the board in serial boot by switching the boot switch. Then, power up the board.

Flash the rootfs image to eMMC using a normal (non-Cockpit) bootloader:

```
$: sudo uuu -v -b emmc_all imx-boot-imx8qmmek-sd.bin-flash imx-image-<type>-
imx8qmmekcockpit-<timestamp>.wic.zst
```

Flash the rootfs image to SD using a normal (non-Cockpit) bootloader:

```
$: sudo uuu -v -b sd_all imx-boot-imx8qmmek-sd.bin-flash imx-image-<type>-
imx8qmmekcockpit-<timestamp>.wic.zst
```

5.3 Flashing the non-OP-TEE bootloader using UUU

Put the board in serial boot by switching the boot switch. Then, power up the board. Write Cockpit bootloader on SD using a normal (non-Cockpit) bootloader:

```
$: sudo uuu -v -b sd imx-boot-imx8qmmek-sd.bin-flash imx-boot-imx8qmmek-sd.bin-
flash_cockpit
$: sudo uuu -v -b emmc imx-boot-imx8qmmek-sd.bin-flash imx-boot-imx8qmmek-
sd.bin-flash_cockpit_spl
```

Once the flashing of the images above is complete, switch back the boot switch to the SD1 configuration. When the board is powered up again, it will boot the cockpit configuration from SD and eMMC.

5.4 Flashing the OP-TEE bootloader using UUU

Programming OP-TEE enabled bootloader using UUU is not supported. You can flash the entire SD/eMMC using UUU instead.

5.5 Flashing entire SD/eMMC using UUU

There is also an option of flashing the entire eMMC or SD at once using UUU. For this, you need the `uuu-imx8qm-cockpit-scripts.zip` file, which contains a script for eMMC and one for SD flashing. These scripts need a normal (non-Cockpit) bootloader and rootfs image (usually a `.wic` file). Their name needs to be `_flash.bin` for bootloader and `_rootfs.wic` for the rootfs image, so rename them accordingly before moving on. They also need to be placed in the current working directory, along with the zip file. You do not need to decompress the zip file.

Then, you need to tell UUU what script to use from the zip file. For eMMC, run this one:

```
$: sudo uuu uuu-imx8qm-cockpit-scripts/uuu.cockpit.emmc
```

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

10 / 15

For SD, run this one:

```
$: sudo uuu uuu-imx8qm-cockpit-scripts/uuu.cockpit.sd
```

This step takes a while to finish, due to the size of the rootfs image.

6   References

- [i.MX 8 Family of Applications Processors](#)
- [i.MX Software and Development Tools](#)
- [Secure Boot on i.MX 8 and i.MX 8X Families using AHAB](#)

7   Known Issues/Limitations

None.

8   Note About the Source Code in the Document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2024 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

9   Revision History

Table 2. Revision history

Document ID	Release date	Description
IMXDCHPE v.LF6.6.3_1.0.0	29 March 2024	Upgraded to the 6.6.3 kernel.
IMXDCHPE v.LF6.1.55_2.2.0	12/2023	Upgraded to the 6.1.55 kernel.
IMXDCHPE v.LF6.1.36_2.1.0	09/2023	Upgraded to the 6.1.36 kernel.
IMXDCHPE v.LF6.1.22_2.0.0	06/2023	Upgraded to the 6.1.22 kernel.
IMXDCHPE v.LF6.1.1_1.0.0	03/2023	Upgraded to the 6.1.1 kernel.

IMXDCHPE

User guide

All information provided in this document is subject to legal disclaimers.

Rev. LF6.6.3_1.0.0 — 29 March 2024

© 2024 NXP B.V. All rights reserved.

11 / 15

Table 2. Revision history...*continued*

Document ID	Release date	Description
IMXDCHPE v.LF5.15.71_2.2.0	12/2022	Upgraded to the 5.15.71 kernel.
IMXDCHPE v.LF5.15.52_2.1.0	09/2022	Upgraded to the 5.15.52 kernel, and added the i.MX 93.
IMXDCHPE v.LF5.15.32_2.0.0	06/2022	Published for the new release, no content update.
IMXDCHPE v.LF5.15.5_1.0.0	03/2022	Back page updated, no content update.
IMXDCHPE v.LF5.10.72_2.2.0	12/2021	Minor updates for the Linux LF5.10.72_2.2.0 release.
IMXDCHPE v.LF5.10.52_2.1.0	09/2021	Minor updates for the Linux LF5.10.52_2.1.0 release.
IMXDCHPE v.LF5.10.35_2.0.0	06/2021	Initial release of Linux Cockpit.

IMXDCHPE

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

12 / 15

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Amazon Web Services, AWS, the Powered by AWS logo, and FreeRTOS — are trademarks of Amazon.com, Inc. or its affiliates.

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

13 / 15

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

i.MX — is a trademark of NXP B.V.

IMXDCHPE

All information provided in this document is subject to legal disclaimers.

© 2024 NXP B.V. All rights reserved.

User guide

Rev. LF6.6.3_1.0.0 — 29 March 2024

14 / 15

Contents