UG10157

Android Quick Start Guide

Rev. android-17.0.0_1.0.0 — **User guide**

3 November 2026

Document information

Information	Content
Keywords	Android, i.MX, android-17.0.0_1.0.0
Abstract	This document guides you through the processes of downloading and running this release package. It only explains how to download and run the default release image with the default configuration.

1 Overview

This document guides you through the processes of downloading and running this release package. It only explains how to download and run the default release image with the default configuration. For details on using the release package, see the *Android User's Guide* (UG10156) included in this release package.

2 Hardware requirements

The hardware requirements for using this release package are as follows:

Supported system-on-chips (SoCs):

- i.MX 8M Mini
- i.MX 8M Nano
- i.MX 8M Plus
- i.MX 8M Quad
- i.MX 8ULP
- i.MX 8QuadMax (Silicon Revision B0)
- i.MX 8QuadXPlus (Silicon Revision B0 and Silicon Revision C0)
- i.MX 95
- i.MX 952
- i.MX 937

Supported boards:

- EVK board and Platform
- WEVK board and Platform
- MEK board and Platform
- FRDM board and Platform

3 Working with the i.MX 8M Mini EVK board

3.1 Board hardware

The figures below show the different components of the i.MX 8M Mini EVK LPDDR4 board.

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

© 2026 NXP B.V. All rights reserved.

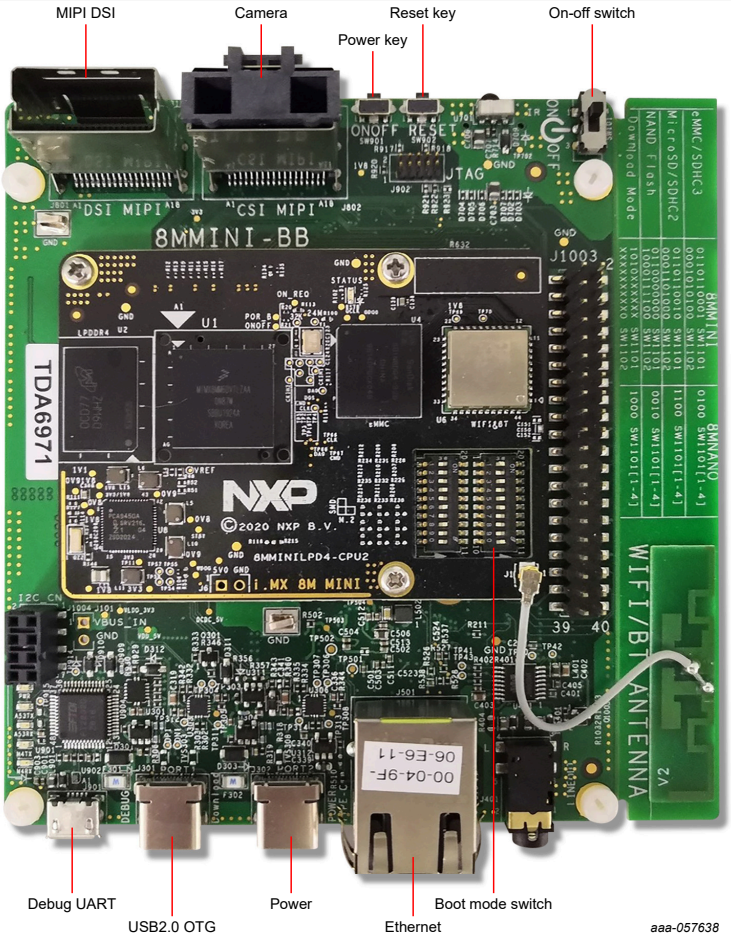


Figure 1. i.MX 8M Mini EVK LPDDR4 board

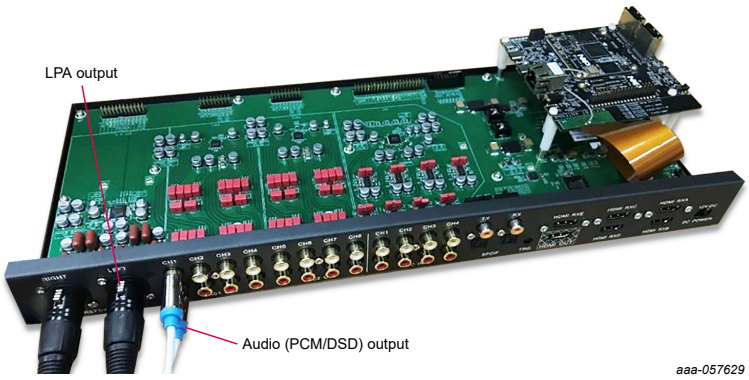


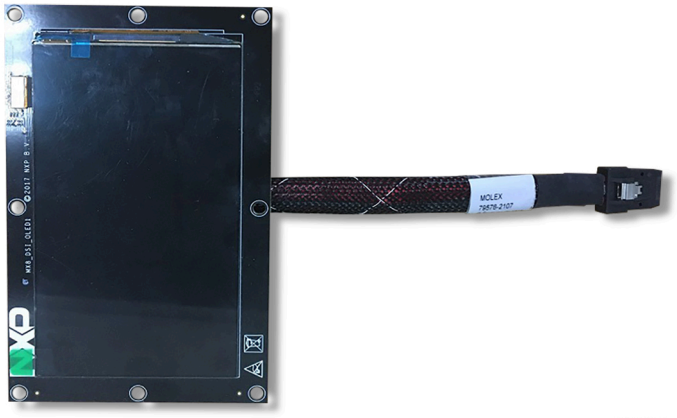
Figure 2. i.MX 8M Mini EVK with audio board



Figure 3. i.MX Mini SAS cable with DSI-to-HDMI adapter

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

© 2026 NXP B.V. All rights reserved.



aaa-057542

Figure 4. MX8-DSI-OLED1 MIPI panel



aaa-058037

Figure 5. OV5640 CSI MIPI camera



aaa-057512

Figure 6. i.MX 8MIC PDM Microphone board

Note:

- *i.MX 8M Mini EVK LPDDR4 Rev. C board and i.MX 8M Mini EVK DDR4 Rev. C board are supported in this release.*
- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "MIPI DSI" port.*
- *To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port.*
- *To test the camera, connect the OV5640 CSI MIPI camera to the "MIPI CSI" port.*
- *To test the i.MX 8MIC PDM microphone, connect the i.MX 8MIC PDM Microphone board to the J1003 connector.*

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

© 2026 NXP B.V. All rights reserved.

- *For i.MX 8M Mini EVK LPDDR4 board, Wi-Fi and Bluetooth functions are supported.*
- *For i.MX 8M Mini EVK DDR4 board, Wi-Fi and Bluetooth functions are not supported.*
- *"i.MX 8M Mini EVK REV C" indicates the revision of the base board.*

3.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8mmevk.tar.gz`.

Table 1. Board images

Image name	Download target	Description
`spl-imx8mm-dual.bin`	33 kB offset of MMC.	Secondary program loader image without Trusty related configurations for the i.MX 8M Mini EVK LPDDR4 board.
`spl-imx8mm-trusty-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty related configurations for the i.MX 8M Mini EVK LPDDR4 board.
`spl-imx8mm-trusty-rbidx-blob-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 8M Mini EVK LPDDR4 board.
`spl-imx8mm-trusty-secure-unlock-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8M Mini EVK LPDDR4 board.
`bootloader-imx8mm-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8M Mini EVK LPDDR4 board
`bootloader-imx8mm-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8M Mini EVK LPDDR4 board.
`bootloader-imx8mm-trusty-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8M Mini EVK LPDDR4 board.
`bootloader-imx8mm-trusty-secure-unlock-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of secure unlock mechanism for i.MX 8M Mini EVK LPDDR4 board.
`u-boot-imx8mm.imx`	33 kB offset of MMC.	An image containing U-Boot and ATF for the i.MX 8M Mini EVK LPDDR4 board.
`u-boot-imx8mm-evk-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8M Mini board with LPDDR4 on it. It is not flashed to MMC.
`u-boot-imx8mm-ddr4.imx`	33 kB offset of SD card.	An image containing U-Boot and ATF for the i.MX 8M Mini EVK DDR4 board.
`u-boot-imx8mm-ddr4-evk-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8M Mini board with DDR4 on it. it is not flashed to SD card.
`imx8mm_mcu_demo.img`	5120 kB offset of MMC.	The MCU FreeRTOS image for the i.MX 8M Mini EVK board.
`partition-table-13GB.img`	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
`partition-table-13GB-dual.img`	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.

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

© 2026 NXP B.V. All rights reserved.

Table 1. Board images*...continued*

Image name	Download target	Description
partition-table.img	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in VTS test with the GSI system image.
vbmeta-imx8mm.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output on the i.MX 8M Mini EVK LPDDR4 board.
vbmeta-imx8mm-m4.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output and audio playback based on Cortex-M4 Free RTOS on the i.MX 8M Mini EVK LPDDR4 board.
vbmeta-imx8mm-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67199 MIPI panel output on the i.MX 8M Mini EVK LPDDR4 board.
vbmeta-imx8mm-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67191 MIPI panel output on the i.MX 8M Mini EVK LPDDR4 board.
vbmeta-imx8mm-ddr4.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output on the i.MX 8M Mini EVK DDR4 board.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.

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

© 2026 NXP B.V. All rights reserved.

Table 1. Board images...*continued*

Image name	Download target	Description
`super.img`	Super partition.	Super image, which contains images for logical partitions.
`dtbo-imx8mm.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support MIPI-to-HDMI output on the i.MX 8M Mini EVK LPDDR4 board.
`dtbo-imx8mm-m4.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support MIPI-to-HDMI output and audio playback based on Cortex-M4 FreeRTOS on the i.MX 8M Mini EVK LPDDR4 board.
`dtbo-imx8mm-mipi-panel.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support RM67199 MIPI panel output on the i.MX 8M Mini EVK LPDDR4 board.
`dtbo-imx8mm-mipi-panel-rm67191.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support RM67191 MIPI panel output on the i.MX 8M Mini EVK LPDDR4 board.
`dtbo-imx8mm-ddr4.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support MIPI-to-HDMI output playback on the i.MX 8M Mini EVK DDR4 board.
`rpmb_key_test.bin`	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set RPMB key.
`testkey_public_rsa4096.bin`	N/A	Prebuilt AVB public key, which is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

3.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

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

© 2026 NXP B.V. All rights reserved.

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   For Rev. C boards, change the first four bits of the board's SW1101 to 1010 (from 1-4 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board USB 2.0 Type-C port to connect your PC with the board.
4. Decompress `release_package/android-17.0.0_1.0.0_image_8mmevk.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For i.MX 8M Mini board, related options are described as follows.

Table 2. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8M Mini EVK, it should be `imx8mm`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8M Mini EVK, it can be followed with "13". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If this option is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For i.MX 8M Mini EVK LPDDR4, it can be `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, or `trusty-secure-unlock-dual`. If this option is not used, the default `u-boot-imx8mm.imx` is flashed. For i.MX 8M Mini EVK DDR4, it should be `ddr4`.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8M Mini EVK LPDDR4, it can be `m4`, `mipi-panel`, or `mipi-panel-rm67191`. If this option is not used, default `dtbo-imx8mm.img` and `vbmeta-imx8mm.img` are flashed. For i.MX 8M Mini EVK DDR4, it should be `ddr4`.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8M Mini EVK, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

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

© 2026 NXP B.V. All rights reserved.

Obviously, `-m` and `-d m4` should be used together.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8mm -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8mm -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8mm-trusty-dual.bin` is flashed, and `bootloader-imx8mm-trusty-dual.img` with other default images are flashed into eMMC slot a for the i.MX 8M Mini EVK LPDDR4 board.

Note:

- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *For i.MX 8M Mini EVK LPDDR4 board:*
  - *To test the dual-bootloader, execute the tool with `-u dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
  - *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
  - *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
  - *To test MIPI-DSI to HDMI output, it does not need to use `-d` option. `dtbo-imx8mm.img` is flashed in this condition to support Wi-Fi expansion card and this image does not support to be booted from the SD card.*
  - *To test RM67199 MIPI panel output, execute the tool with `-d mipi-panel`.*
  - *To test RM67191 MIPI panel output, execute the tool with `-d mipi-panel-rm67191`.*
  - *To test support MIPI-DSI-to-HDMI output and audio playback based on Cortex-M4 FreeRTOS, execute the tool with `-m` and `-d m4`.*
- *For i.MX 8M Mini EVK DDR4 board: execute the command with `-u ddr4 -d ddr4 -t sd`.*
- *For i.MX 8M Mini EVK with audio board: To test low power audio, execute the tool with `-d m4 -m`. See the Android User's Guide (UG10156) for more steps to make the low-power audio work.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through samba, you need to map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is not any error, the command window displays information indicating that images are already flashed.

   Note:

   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.

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

© 2026 NXP B.V. All rights reserved.

7. Power off the board.
8. Change boot device to eMMC or SD card.
   For Rev. C boards:
   - To boot from the SD card, change SW1101 to 0110110010 (from 1-10 bit) and change SW1102 to 0001101000 (from 1-10 bit).
   - To boot from eMMC, change SW1101 to 0110110001 (from 1-10 bit) and change SW1102 to 0001010100 (from 1-10 bit).

3.4 Booting

After downloading the images, reboot the board using the power on/off switch.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

3.4.1 Booting with audio playback based on Cortex-M4 FreeRTOS

In the U-Boot prompt, set the U-Boot environment variables as follows:

- i.MX 8M Mini EVK LPDDR4 Board:

```
U-Boot > setenv bootcmd "bootmcu && boota"
U-Boot > saveenv
```

Note:

To use other dtbo images, do not add `bootmcu" to "bootcmd`. The following command can recover `bootcmd`:

```
U-Boot > setenv bootcmd "boota"
U-Boot > saveenv
```

3.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

4 Working with the i.MX 8M Nano EVK board

4.1 Board hardware

The figures below show the different components of the i.MX 8M Nano EVK board.

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

© 2026 NXP B.V. All rights reserved.

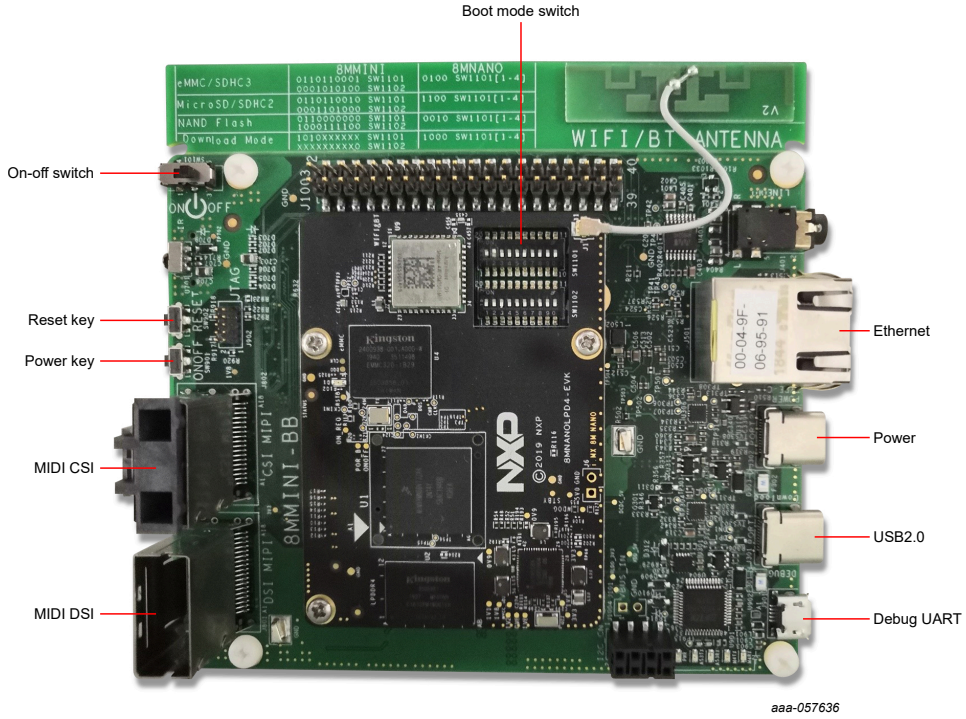


Figure 7. i.MX 8M Nano EVK board



Figure 8. i.MX mini SAS cable with DSI-to-HDMI adapter

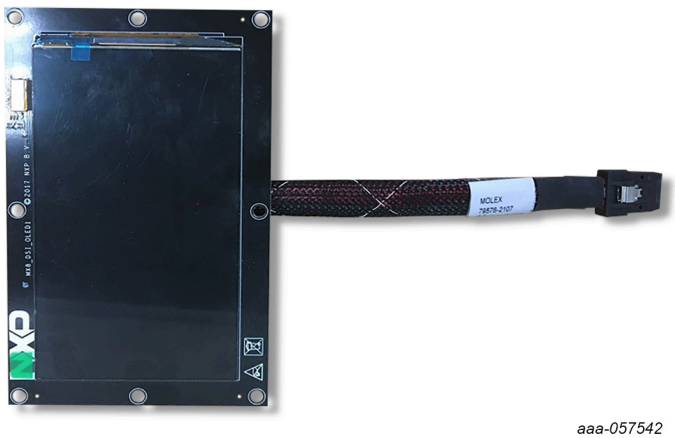


Figure 9. MX8-DSI-OLED1 MIPI panel



Figure 10. OV5640 CSI MIPI camera

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

© 2026 NXP B.V. All rights reserved.



Figure 11. i.MX 8MIC PDM Microphone board

Note:

- *i.MX 8M Nano EVK LPDDR4 board and i.MX 8M Nano EVK DDR4 board are supported in this release.*
- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "MIPI DSI" port.*
- *To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port.*
- *To test the camera, connect the OV5640 CSI MIPI camera to the "MIPI CSI" port.*
- *To test i.MX 8MIC PDM microphone, connect the i.MX 8MIC PDM Microphone board to the J1003 connector.*
- *For i.MX 8M Nano EVK LPDDR4 board, Bluetooth/Wi-Fi functions are supported.*
- *For i.MX 8M Nano EVK DDR4 board, Bluetooth/Wi-Fi functions are not maintained.*

4.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8mnevk.tar.gz`.

Table 3. Board images

Image name	Download target	Description
`spl-imx8mn-dual.bin`	32 kB offset of MMC.	Secondary program loader image without Trusty related configurations for the i.MX 8M Nano EVK LPDDR4 board.
`spl-imx8mn-trusty-dual.bin`	32 kB offset of MMC.	Secondary program loader image with Trusty related configurations for the i.MX 8M Nano EVK LPDDR4 board.
`spl-imx8mn-trusty-rbidx-blob-dual.bin`	32 kB offset of MMC.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 8M Nano EVK LPDDR4 board.
`spl-imx8mn-trusty-secure-unlock-dual.bin`	32 kB offset of MMC.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8M Nano EVK LPDDR4 board.
`bootloader-imx8mn-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8M Nano EVK LPDDR4 board.

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

© 2026 NXP B.V. All rights reserved.

Table 3. Board images...*continued*

Image name	Download target	Description
bootloader-imx8mn-trusty-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8M Nano EVK LPDDR4 board.
bootloader-imx8mn-trusty-rbidx-blob-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8M Nano EVK LPDDR4 board.
bootloader-imx8mn-trusty-secure-unlock-dual.img	bootloader_a and bootloader_b partition.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 8M Nano EVK LPDDR4 board.
u-boot-imx8mn.imx	32 kB offset of MMC.	An image containing U-Boot and ATF for the i.MX 8M Nano EVK LPDDR4 board.
u-boot-imx8mn-ddr4.imx	32 kB offset of MMC.	An image containing U-Boot and ATF for the i.MX 8M Nano EVK DDR4 board.
u-boot-imx8mn-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8M Nano EVK LPDDR4 board. It is not flashed to MMC.
u-boot-imx8mn-ddr4-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for i.MX 8M Nano EVK DDR4 board. It is not flashed to MMC.
imx8mn_mcu_demo.img	5120 kB offset of MMC.	The MCU demonstration image for the i.MX 8M Nano EVK board.
partition-table-13GB.img	0 kB offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-13GB-dual.img	0 kB offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
partition-table.img	0 kB offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 kB offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image, generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in the VTS test with GSI system image.
vbmeta-imx8mn.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output on the i.MX 8M Nano EVK LPDDR4 board.
vbmeta-imx8mn-rpmsg.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output and MCU image on the i.MX 8M Nano EVK LPDDR4 board.
vbmeta-imx8mn-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67199 MIPI panel output on the i.MX 8M Nano EVK LPDDR4 board.

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

© 2026 NXP B.V. All rights reserved.

Table 3. Board images...*continued*

Image name	Download target	Description
vbmeta-imx8mn-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67191 MIPI panel output on the i.MX 8M Nano EVK LPDDR4 board.
vbmeta-imx8mn-ddr4.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output on the i.MX 8M Nano EVK DDR4 board.
vbmeta-imx8mn-ddr4-rpmsg.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output and MCU image on the i.MX 8M Nano EVK DDR4 board.
vbmeta-imx8mn-ddr4-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67199 MIPI panel output on the i.MX 8M Nano EVK DDR4 board.
vbmeta-imx8mn-ddr4-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support RM67191 MIPI panel output on the i.MX 8M Nano EVK DDR4 board.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx8mn.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output on the i.MX 8M Nano EVK LPDDR4 board.
dtbo-imx8mn-rpmsg.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output and MCU image on the i.MX 8M Nano EVK LPDDR4 board.
dtbo-imx8mn-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support RM67199 MIPI panel output on the i.MX 8M Nano EVK LPDDR4 board.
dtbo-imx8mn-mipi-panel-rm67191.img	dtbo_a and dtbo_b partitions.	Device Tree image to support RM67191 MIPI panel output on the i.MX 8M Nano EVK LPDDR4 board.
dtbo-imx8mn-ddr4.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output on the i.MX 8M Nano EVK DDR4 board.

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

© 2026 NXP B.V. All rights reserved.

Table 3. Board images...*continued*

Image name	Download target	Description
`dtbo-imx8mn-ddr4-rpmsg.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support MIPI-to-HDMI output and MCU image on the i.MX 8M Nano EVK DDR4 board.
`dtbo-imx8mn-ddr4-mipi-panel.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support RM67199 MIPI panel output on the i.MX 8M Nano EVK DDR4 board.
`dtbo-imx8mn-ddr4-mipi-panel-rm67191.img`	`dtbo_a` and `dtbo_b` partitions.	Device Tree image to support RM67191 MIPI panel output on the i.MX 8M Nano EVK DDR4 board.
`rpmb_key_test.bin`	N/A	Prebuilt test RPMB key. It can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
`testkey_public_rsa4096.bin`	N/A	Prebuilt AVB public key. It is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

4.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   For Rev. C2 boards, change the first four bits of the board's SW1101 to 1000 (from 1-4 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board OTG port to connect your PC with the board.

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

© 2026 NXP B.V. All rights reserved.

4. Decompress `release_package/android-17.0.0_1.0.0_image_8mnevk.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For i.MX 8M Nano board, related options are described as follows.

Table 4. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8M Nano EVK, it should be `imx8mn`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8M Nano EVK, it can be followed with "13". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If this option is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For i.MX 8M Nano EVK LPDDR4 board, it can be `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, or `trusty-secure-unlock-dual`. For i.MX 8M Nano EVK DDR4, it should be `ddr4`. If this option is not used, the default `u-boot-imx8mn.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8M Nano EVK LPDDR4 board, it can be `rpmsg`, `mipi-panel`, `mipi-panel-rm67191`. For i.MX 8M Nano EVK DDR4 board, it can be `ddr4`, `ddr4-rpmsg`, `ddr4-mipi-panel`, or `ddr4-mipi-panel-rm67191`. If this option is not used, the default `dtbo-imx8mn.img` and `vbmeta-imx8mn.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8M Nano EVK, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

Obviously, `-m` should be used together with `-d rpmsg` or `-d ddr4-rpmsg`.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8mn -a -e -u trusty-dual
```


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

© 2026 NXP B.V. All rights reserved.

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8mn -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8mn-trusty-dual.bin` is flashed, and `bootloader-imx8mn-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 8M Nano EVK LPDDR4 board.

Note:

- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *For i.MX 8M Nano EVK LPDDR4 board:*
  - *To test dual bootloaders, execute the tool with `-u dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
  - *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
  - *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
  - *To test MIPI-DSI to HDMI output, it does not need to use the `-d` option.*
  - *To test MIPI-DSI to HDMI output and MCU image, execute the tool with `-d rpmsg`.*
  - *To test RM67199 MIPI panel output, execute the tool with `-d mipi-panel`.*
  - *To test RM67191 MIPI panel output, execute the tool with `-d mipi-panel-rm67191`.*
- *For i.MX 8M Nano EVK DDR4 board:*
  - *To test MIPI-DSI to HDMI output, execute the tool with `-u ddr4 -d ddr4`.*
  - *To test DSI to HDMI output and MCU image, execute the tool with `-u ddr4 -d ddr4-rpmsg`.*
  - *To test RM67199 MIPI panel output, execute the tool with `-u ddr4 -d ddr4-mipi-panel`.*
  - *To test RM67191 MIPI panel output, execute the tool with `-u ddr4 -d ddr4-mipi-panel-rm67191`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is no error, the command window displays the information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change boot device to eMMC or SD card.
   For Rev. C boards:
   - Change SW1101 first four bits ([1-4]) to 0100 to boot from eMMC.
   - Change SW1101 first four bits ([1-4]) to 1100 to boot from the SD card.

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

© 2026 NXP B.V. All rights reserved.

4.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

4.4.1 Booting with an MCU image

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv bootcmd "bootmcu && boota"
U-Boot > saveenv
```

Note:

To use other dtbo images, do not add `bootmcu` to `bootcmd`. The following command can recover bootcmd:

```
U-Boot > setenv bootcmd "boota"
U-Boot > saveenv
```

4.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

5 Working with the i.MX 8M Plus EVK board

5.1 Board hardware

The figure below shows the different components of the i.MX 8M Plus EVK board.

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

© 2026 NXP B.V. All rights reserved.

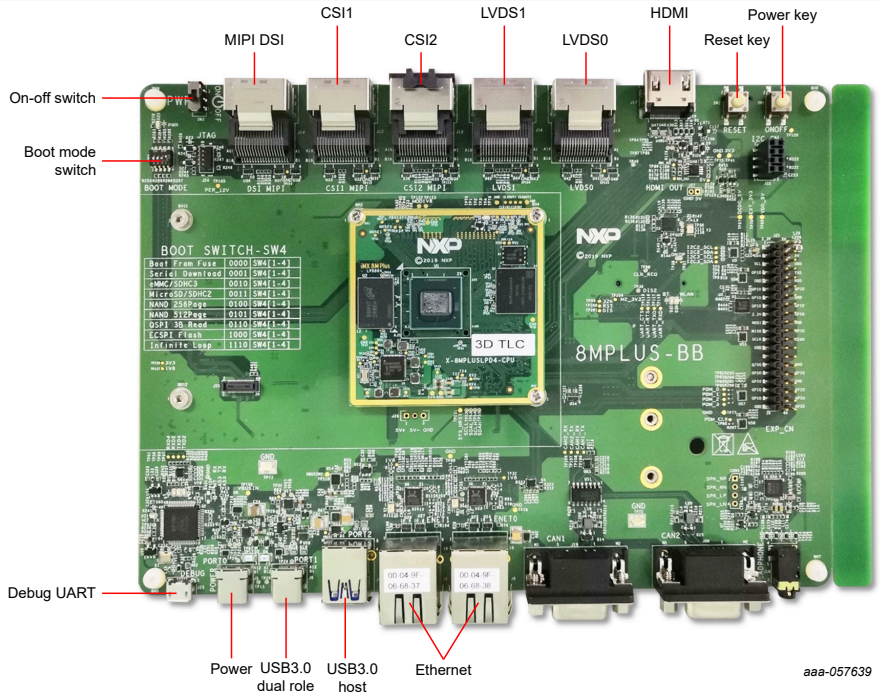


Figure 12. i.MX 8M Plus EVK board

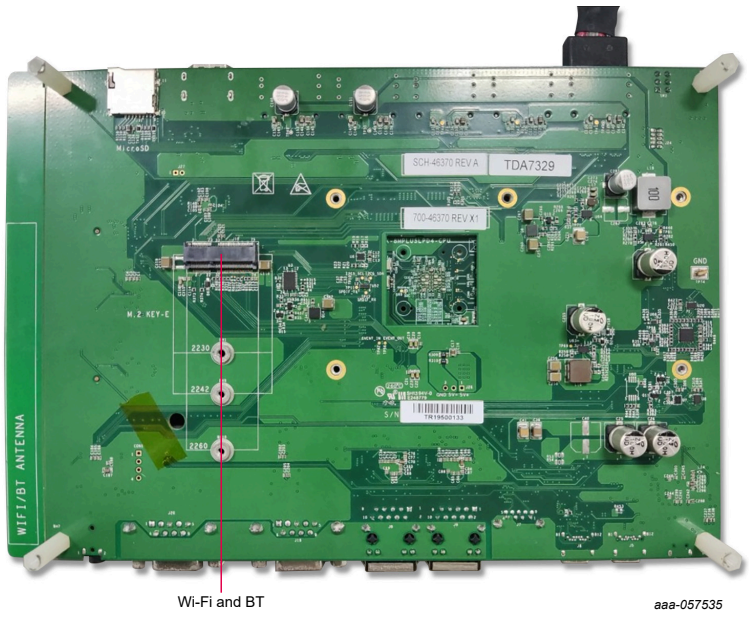


Figure 13. i.MX 8M Plus EVK board back view



Figure 14. i.MX mini SAS cable with DSI-to-HDMI adapter

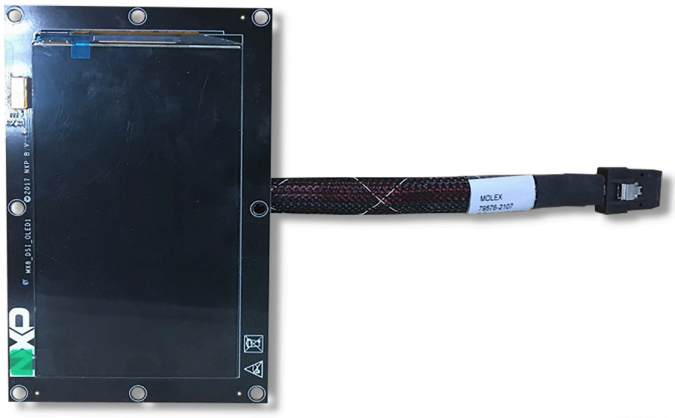
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



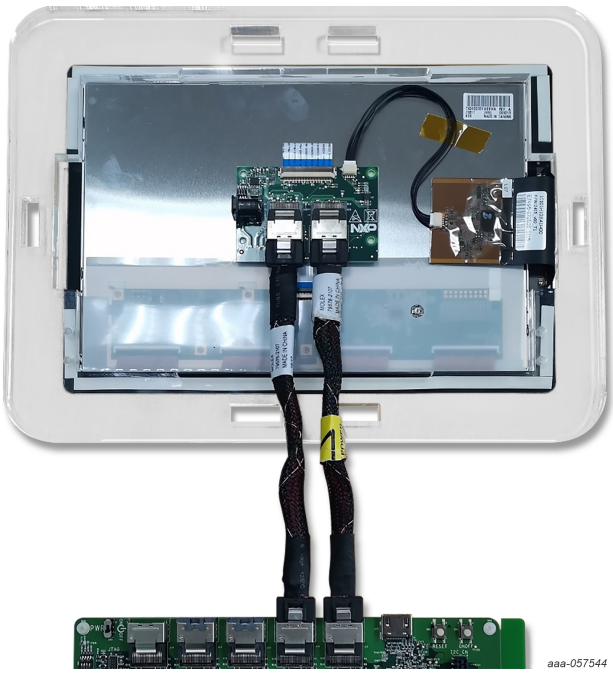
aaa-057541

Figure 15. i.MX mini SAS cable with LVDS-to-HDMI adapter



aaa-057542

Figure 16. MX8-DSI-OLED1 MIPI panel



aaa-057544

Figure 17. i.MX LVDS panel

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

© 2026 NXP B.V. All rights reserved.



aaa-057633

Figure 18. i.MX dual-channel LVDS-to-HDMI adapter



aaa-058037

Figure 19. OV5640 CSI MIPI camera



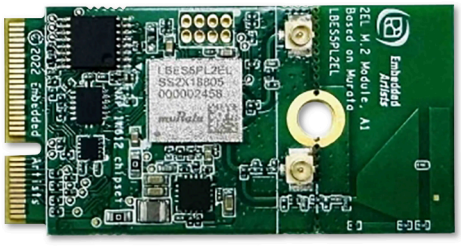
aaa-057630

Figure 20. Basler CSI MIPI camera



aaa-057540

Figure 21. OS08A20 CSI MIPI camera



aaaa-057479

Figure 22. SDIW612 (Murata LBES5PL2EL)

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

© 2026 NXP B.V. All rights reserved.

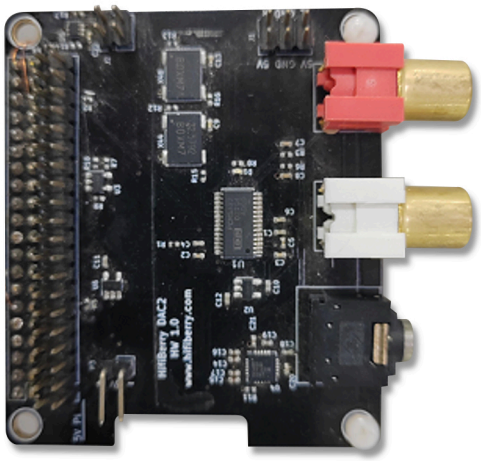


Figure 23. i.MX 8M Plus EVK PCM512 audio board (for powersave image)

Note:

- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "MIPI DSI" port.*
- *To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port.*
- *To test physical HDMI display, connect the HDMI cable to the "HDMI" port.*
- *To test the LVDS-to-HDMI display, use the i.MX mini SAS cable to connect the LVDS-to-HDMI adapter to the "LVDS0" port.*
- *To test the LVDS panel display, use two i.MX mini SAS cables to connect the LVDS panel to the "LVDS0" and "LVDS1" ports. Provide 5 V power through the DC jack on the LVDS panel or connect pin3 with pin4 of pin header j7 beside the DC jack.*
- *To test dual-channel LVDS-to-HDMI display, use two i.MX mini SAS cables to connect the dual-channel LVDS-to-HDMI adapter to the "LVDS0" and "LVDS1" ports.*
- *To test the camera, follow the rules below:*
  - *OS08A20 (CSI1) + OS08A20 (CSI2): Use* `dtbo-imx8mp.img` + `vbmeta-imx8mp.img`.
  - *Basler (CSI1) + OV5640 (CSI2): Use* `dtbo-imx8mp-basler-ov5640.img` + `vbmeta-imx8mp-basler-ov5640.img`.
  - *Only Basler (CSI1): Use* `dtbo-imx8mp-basler.img` + `vbmeta-imx8mp-basler.img`.
  - *Only OV5640 (CSI1): Use* `dtbo-imx8mp-ov5640.img` + `vbmeta-imx8mp-ov5640.img`.
  - *Basler (CSI1) + Basler (CSI2): Use* `dtbo-imx8mp-dual-basler.img` + `vbmeta-imx8mp-dual-basler.img`.
  - *OS08A20 (CSI1) + OV5640 (CSI2): Use* `dtbo-imx8mp-os08a20-ov5640.img` + `vbmeta-imx8mp-os08a20-ov5640.img`.
  - *Only OS08A20 (CSI1): Use* `dtbo-imx8mp-os08a20.img` + `vbmeta-imx8mp-os08a20.img`.
- *Connect the SDIW612 (Murata LBES5PL2EL) Wi-Fi&BT M.2 expansion card to the J10 connector to enable Wi-Fi and Bluetooth.*
- *The Basler sensor functionality is no longer maintained due to lack of continued support from Basler.*

5.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8mpevk.tar.gz`.

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

© 2026 NXP B.V. All rights reserved.

Table 5. Board images

Image name	Download target	Description
spl-imx8mp-dual.bin	0 kB offset of eMMC boot0 partition or 32 kB offset of SD card.	Secondary program loader image without Trusty related configurations for the i.MX 8M Plus EVK board.
spl-imx8mp-trusty-dual.bin	0 kB offset of eMMC boot0 partition.	Secondary program loader image with Trusty related configurations for the i.MX 8M Plus EVK board.
spl-imx8mp-trusty-rbidx-blob-dual.bin	0 kB offset of eMMC boot0 partition.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 8MPlus EVK board.
spl-imx8mp-trusty-secure-unlock-dual.bin	0 kB offset of eMMC boot0 partition.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8M Plus EVK LPDDR4 board.
bootloader-imx8mp-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8M Plus EVK board.
bootloader-imx8mp-trusty-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8M Plus EVK board.
bootloader-imx8mp-trusty-rbidx-blob-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8MPlus EVK board.
bootloader-imx8mp-trusty-secure-unlock-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 8M Plus EVK board.
u-boot-imx8mp.imx	0 kB offset of eMMC boot0 partition or 32 kB offset of SD card.	An image containing U-Boot and ATF for the i.MX 8M Plus EVK board.
u-boot-imx8mp-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8M Plus EVK board. It is not flashed to MMC.
imx8mp_mcu_demo.img	5120 kB offset of eMMC user partition or SD card.	The MCU image for the i.MX 8M Plus EVK board.
partition-table-13GB.img	0 kB offset of eMMC user partition or SD card.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-13GB-dual.img	0 kB offset of eMMC user partition or SD card.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
partition-table.img	0 kB offset of eMMC user partition or SD card.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 kB offset of eMMC user partition or SD card.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.

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

© 2026 NXP B.V. All rights reserved.

Table 5. Board images*...continued*

Image name	Download target	Description
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in VTS test with the GSI system image.
vbmeta-imx8mp.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support two OS08A20 cameras plug-in CSI-1 and CSI-2 ports.
vbmeta-imx8mp-basler-ov5640.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support Basler camera plug-in CSI-1 port and OV5640 camera plug-in CSI-2 port.
vbmeta-imx8mp-dual-basler.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support Basler + Basler camera.
vbmeta-imx8mp-os08a20-ov5640.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support OS08A20 (CSI-1) + OV5640 (CSI-2) camera.
vbmeta-imx8mp-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support only OS08A20 camera plug-in CSI-1 slot.
vbmeta-imx8mp-basler.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support only Basler camera plug-in CSI-1 slot.
vbmeta-imx8mp-ov5640.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support only OV5640 CSI MIPI camera plug-in CSI-1 slot.
vbmeta-imx8mp-lvds-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support LVDS panel output.
vbmeta-imx8mp-lvds.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support dual-channel LVDS-to-HDMI output.
vbmeta-imx8mp-mipi-panel.img	vbmeta_a and vbmeta_b partitions	Android Verify Boot metadata image to support RM67199 MIPI panel output.
vbmeta-imx8mp-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions	Android Verify Boot metadata image to support RM67191 MIPI panel output.
vbmeta-imx8mp-rpmsg.img	vbmeta_a and vbmeta_b partitions	Android Verify Boot metadata image to support MIPI-to-HDMI output and MCU image.
vbmeta-imx8mp-sof.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI output, and support the Sound Open Firmware audio output.
system.img	Logical partitions system_a and system_b in super partition	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.

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

© 2026 NXP B.V. All rights reserved.

Table 5. Board images*...continued*

Image name	Download target	Description
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx8mp.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and two OS08A20 cameras plug-in CSI-1 and CSI-2 ports.
dtbo-imx8mp-basler-ov5640.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support Basler camera plug-in CSI-1 port and OV5640 camera plug-in CSI-2 port.
dtbo-imx8mp-basler.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support only Basler camera plug-in CSI-1 slot.
dtbo-imx8mp-ov5640.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support only OV5640 CSI MIPI camera plug-in CSI-1 slot.
dtbo-imx8mp-dual-basler.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support Basler + Basler camera.
dtbo-imx8mp-os08a20-ov5640.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support OS08A20 (CSI-1) + OV5640 (CSI-2) camera.
dtbo-imx8mp-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support only OS08A20 camera plug-in CSI-1 slot.
dtbo-imx8mp-lvds-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support LVDS panel output.
dtbo-imx8mp-lvds.img	dtbo_a and dtbo_b partitions.	Device Tree image to support dual-channel LVDS to HDMI output.
dtbo-imx8mp-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support RM67199 MIPI panel output.
dtbo-imx8mp-mipi-panel-rm67191.img	dtbo_a and dtbo_b partitions.	Device Tree image to support RM67191 MIPI panel output.
dtbo-imx8mp-rpmsg.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output and MCU image.
dtbo-imx8mp-sof.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI output, and support the Sound Open Firmware audio output.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.

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

© 2026 NXP B.V. All rights reserved.

Table 5. Board images...*continued*

Image name	Download target	Description
`testkey_public_rsa4096.bin`	N/A	Prebuilt AVB public key, which is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

5.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   Change the board's SW4 (boot mode) to 0001 (from 1-4 bit) to enter serial download mode.
3. Power on the board.
   Use the USB cable to connect the USB 3.0 dual-role port (with silkprint "PORT1") on the board to your host PC.
4. Decompress `release_package/android-17.0.0_1.0.0_image_8mpevk.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For i.MX 8M Plus board, related options are described as follows.

Table 6. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays help information of this tool.

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

© 2026 NXP B.V. All rights reserved.

Table 6. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-f soc_name`	Specifies SoC information. For i.MX 8M Plus EVK, it should be `imx8mp`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8M Plus EVK, it can be followed with "13". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If this option is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&Bootloader images with `uboot_feature` in their names. For i.MX 8M Plus EVK board, it can be `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, or `trusty-secure-unlock-dual`. If this option is not used, the default `u-boot-imx8mp.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8M Plus EVK with Non-Rev. B4 BB, it can be `basler-ov5640`, `basler`, `ov5640`, `lvds-panel`, `lvds`, `mipi-panel`, `rpmsg`, `dual-basler`, `os08a20-ov5640`, `os08a20`, `mipi-panel-rm67191`, or `sof`. If this option is not used, the default `dtbo-imx8mp.img` and `vbmeta-imx8mp.img` are flashed.
`-e`	Erases the user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8M Plus EVK, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

Note: *`-m` should be used together with `-d rpmsg`.*

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8mp -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8mp -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8mp-trusty-dual.bin` is flashed, and `bootloader-imx8mp-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 8M Plus EVK.

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

© 2026 NXP B.V. All rights reserved.

Note:

- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test dual-bootloader, execute the tool with `-u dual`.*
- *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
- *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
- *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
- *To test features on the i.MX 8M Plus EVK board:*
  - *To test MIPI-to-HDMI display, it does not need to use `-d` option. Physical HDMI, LVDS-to-HDMI are also supported in this condition.*
  - *To test LVDS panel display, execute the tool with `-d lvds-panel`. MIPI-to-HDMI and physical HDMI are also supported in this condition.*
  - *To test dual-channel LVDS-to-HDMI display, execute the tool with `-d lvds`.*
  - *To test the RM67199 MIPI panel display, execute the tool with `-d mipi-panel`. Physical HDMI and LVDS-to-HDMI are also supported in this condition.*
  - *To test RM67191 MIPI panel display, execute the tool with `-d mipi-panel-rm67191`. Physical HDMI and LVDS-to-HDMI are also supported in this condition.*
  - *To test the MIPI-to-HDMI display and MCU image, execute the tool with `-d rpmsg`. Physical HDMI and LVDS-to-HDMI are also supported in this condition.*
  - *To test two OS08A20 cameras, it does not need to use `-d` option. The cameras can work by default.*
  - *To test the combination of one Basler camera and one OV5640 camera, execute the tool with `-d basler-ov5640`. See the Android User's Guide (UG10156) for more steps to make cameras work.*
  - *To test single Basler camera, execute the tool with `-d basler`. See the Android User's Guide (UG10156) for more steps to make the camera work.*
  - *To test single OV5640 camera, execute the tool with `-d ov5640`. See the Android User's Guide (UG10156) for more steps to make the camera work.*
  - *To test the combination of dual Basler cameras, execute the tool with `-d dual-basler`. Check Android User's Guide (UG10156) for more steps to make the camera work.*
  - *To test the combination of one OS08A20 camera and one OV5640 camera, execute the tool with `-d os08a20-ov5640`. See the Android User's Guide (UG10156) for more steps to make the camera work.*
  - *To test single OS08A20 camera, execute the tool with `-d os08a20`. See the Android User's Guide (UG10156) for more steps to make the camera work.*
  - *To test the Sound Open Firmware audio output, execute the tool with `-d sof`.*
  - *To test low-power audio, execute the tool with `-d rpmsg -m`. See the Android User's Guide (UG10156) for more steps to make the low-power audio work.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_image
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is no error, the command window displays the information indicating that images are already flashed.

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

© 2026 NXP B.V. All rights reserved.

7. Power off the board.
8. Change boot device to eMMC or SD card.
   - Change SW4 to switch the board back to 0011 (SD boot mode, from 1-4 bit).
   - Change SW4 to switch the board back to 0010 (eMMC boot mode, from 1-4 bit).

5.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

5.4.1 Booting with audio playback based on Cortex-M7 FreeRTOS

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv bootcmd "bootmcu && boota"
U-Boot > saveenv
```

Note:

To use other dtbo images, do not add `bootmcu` to `bootcmd`. The following command can recover bootcmd:

```
U-Boot > setenv bootcmd "boota"
U-Boot > saveenv
```

5.4.2 Booting with a single display: HDMI 4K display

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.displaymode=4kp30
 androidboot.lcd_density=480
U-Boot > saveenv
```

Note:

If other boot arguments need to be appended to the `bootargs` or `bootconfig`, set them together in one `setenv append_bootargs` or `setenv append_bootconfig` command.

5.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

6 Working with the i.MX 8M Quad WEVK board

6.1 Board hardware

The figures below show the different components of the i.MX 8M Quad WEVK board.

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

© 2026 NXP B.V. All rights reserved.

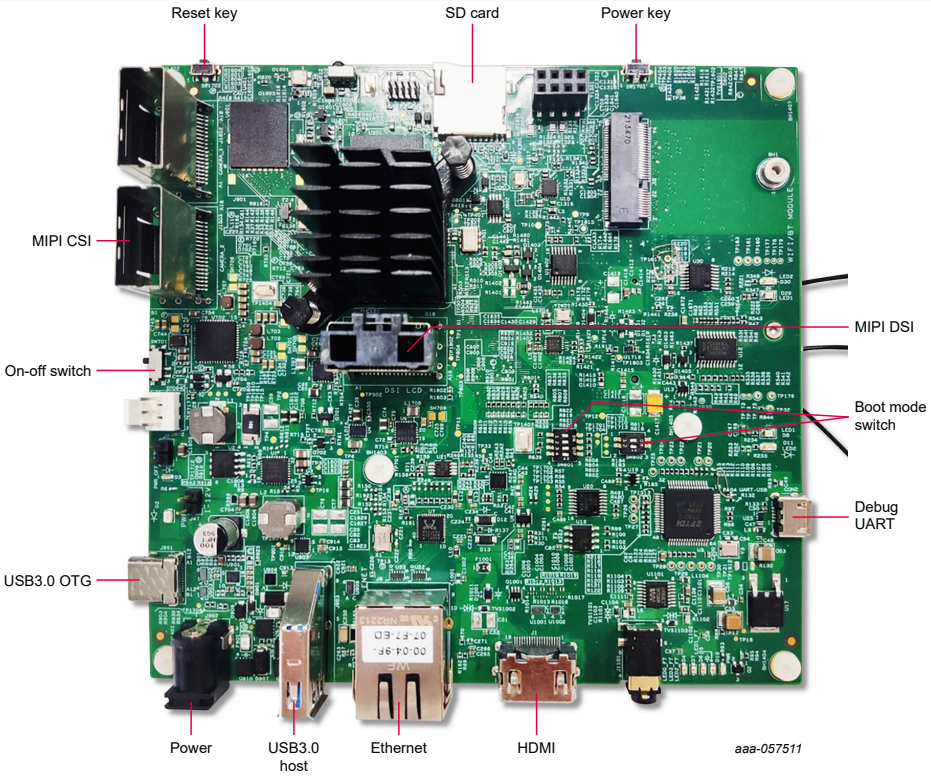


Figure 24. i.MX 8M Quad WEVK board

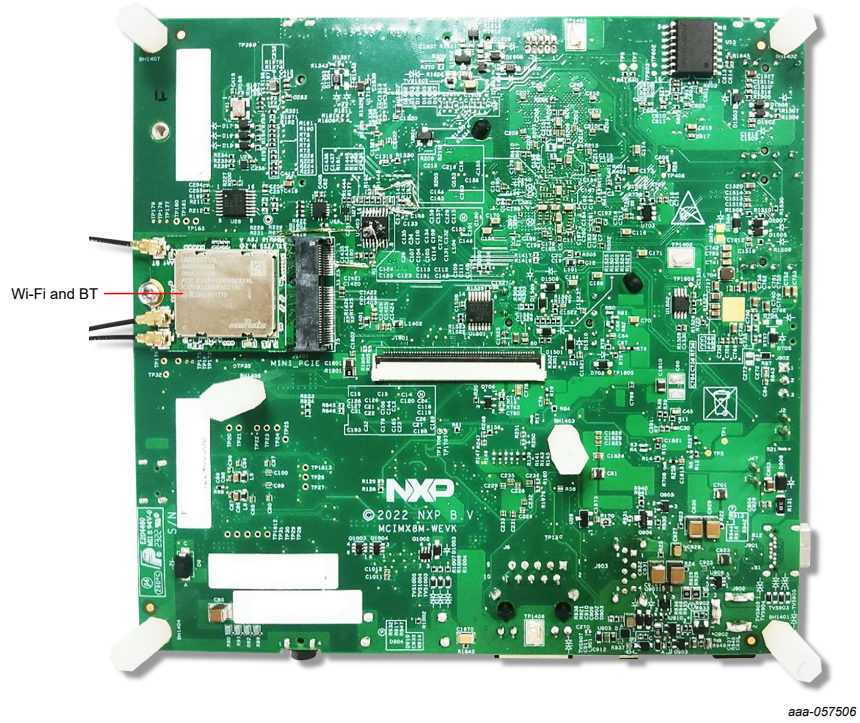


Figure 25. i.MX 8M Quad WEVK board back view

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

© 2026 NXP B.V. All rights reserved.

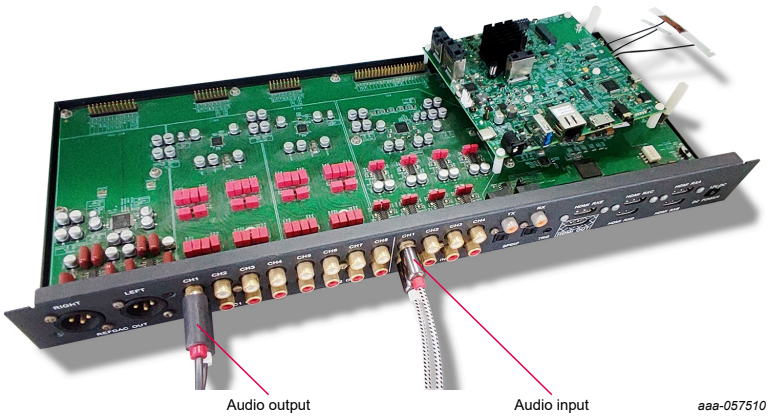


Figure 26. i.MX 8M Quad WEVK with audio board



Figure 27. i.MX mini SAS cable with DSI-to-HDMI adapter

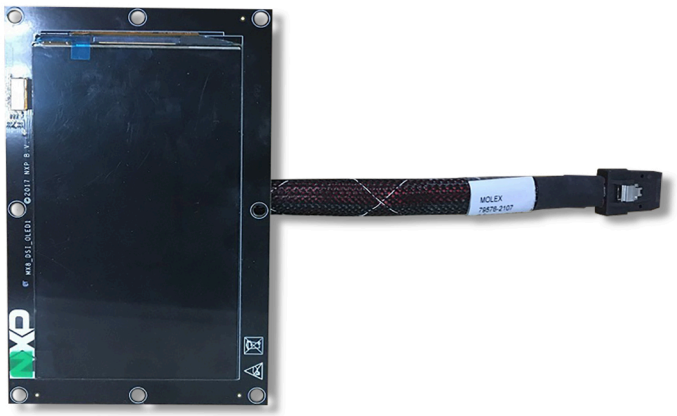


Figure 28. MX8-DSI-OLED1 MIPI panel



Figure 29. OV5640 CSI MIPI camera

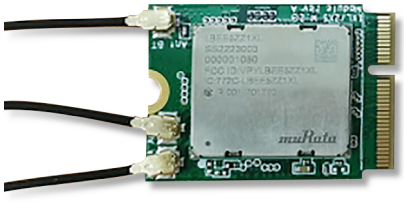


Figure 30. PCIE9098 (Murata LBEE5ZZ1XL)

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

© 2026 NXP B.V. All rights reserved.

Note:

- *i.MX 8M Quad WEVK board is supported in this release.*
- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "MIPI DSI" port.*
- *To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port.*
- *To test the camera, connect the OV5640 CSI MIPI camera to the "MIPI CSI" port.*
- *Connect the PCIE9098 (Murata LBEE5ZZ1XL) Wi-Fi and Bluetooth M.2 expansion card to the J1401 connector to enable Wi-Fi and Bluetooth.*

6.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8mqwevk.tar.gz`.

Table 7. Board images

Image name	Download target	Description
`spl-imx8mq-dual.bin`	33 kB offset of MMC.	Secondary program loader image without Trusty related configurations for the i.MX 8M Quad EVK board.
`spl-imx8mq-trusty-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty related configurations for the i.MX 8M Quad EVK board.
`spl-imx8mq-trusty-secure-unlock-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8M Quad EVK LPDDR4 board.
`spl-imx8mq-wevk-dual.bin`	33 kB offset of MMC.	Secondary program loader image without Trusty related configurations for the i.MX 8M Quad WEVK board.
`spl-imx8mq-trusty-wevk-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty related configurations for the i.MX 8M Quad WEVK board.
`spl-imx8mq-trusty-wevk-rbidx-blob-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for i.MX 8M Quad WEVK board.
`spl-imx8mq-trusty-secure-unlock-wevk-dual.bin`	33 kB offset of MMC.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8M Quad WEVK LPDDR4 board.
`bootloader-imx8mq-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8M Quad EVK board.
`bootloader-imx8mq-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8M Quad EVK board.
`bootloader-imx8mq-trusty-secure-unlock-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 8M Quad EVK LPDDR4 board.
`bootloader-imx8mq-wevk-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for i.MX 8M Quad WEVK board.
`bootloader-imx8mq-trusty-wevk-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8M Quad WEVK board.
`bootloader-imx8mq-trusty-wevk-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8MQuad WEVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 7. Board images*...continued*

Image name	Download target	Description
bootloader-imx8mq-trusty-secure-unlock-wevk-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 8M Quad WEVK LPDDR4 board.
u-boot-imx8mq.imx	33 kB offset of MMC.	An image containing U-Boot and ATF for the i.MX 8M Quad EVK board.
u-boot-imx8mq-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8M Quad EVK board. It is not flashed to MMC.
u-boot-imx8mq-wevk.imx	33 kB offset of MMC.	An image containing U-Boot and ATF for i.MX 8M Quad WEVK board.
u-boot-imx8mq-wevk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for i.MX 8M Quad WEVK board. It is not flashed to MMC.
partition-table.img	0 kB offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-dual.img	0 kB offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-28GB.img	0 kB offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-28GB-dual.img	0 kB offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in VTS test with the GSI system image.
vbmeta-imx8mq.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad EVK board HDMI output.
vbmeta-imx8mq-wevk.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad WEVK board HDMI output.
vbmeta-imx8mq-mipi.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad WEVK/EVK board MIPI-DSI-to-HDMI output.
vbmeta-imx8mq-dual.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad WEVK/EVK board HDMI and MIPI-DSI-to-HDMI dual output.
vbmeta-imx8mq-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad WEVK/EVK board RM67199 MIPI panel output.
vbmeta-imx8mq-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the i.MX 8M Quad WEVK/EVK board RM67191 MIPI panel output.
system.img	Logical partitions system_a and system_b in super partition.	System image.

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

© 2026 NXP B.V. All rights reserved.

Table 7. Board images...*continued*

Image name	Download target	Description
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partition product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains the images for logical partitions.
dtbo-imx8mq.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad EVK board HDMI output.
dtbo-imx8mq-wevk.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad WEVK board HDMI output.
dtbo-imx8mq-mipi.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad WEVK/EVK board MIPI-DSI-to-HDMI output.
dtbo-imx8mq-dual.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad WEVK/EVK board HDMI and MIPI-DSI-to-HDMI dual output.
dtbo-imx8mq-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad WEVK/EVK board RM67199 MIPI panel output.
dtbo-imx8mq-mipi-panel-rm67191.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8M Quad WEVK/EVK board RM67191 MIPI panel output.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key to fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key, which is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

With virtual A/B enabled, the `super.img` *file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like* `system_b` *is* ***0*** *in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

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

© 2026 NXP B.V. All rights reserved.

6.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   Change the board's SW802 (boot mode) to 01 (from 1-2 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board USB 3.0 port to connect your PC with the board.
   Note:
   - *There are three USB ports on the i.MX 8M Quad WEVK board: USB-to-UART, USB 3.0 host, and USB 3.0 OTG. The USB-to-UART can be referenced as the debug UART in the hardware image above. The debug UART can be used to watch the log of the hardware boot processing.*
   - *The SD card must be plugged in after the board is powered on.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_8mqwevk.tar.gz`, which contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For i.MX 8M Quad board, related options are described as follows.

Table 8. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8M Quad WEVK, it should be `imx8mq`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8M Quad WEVK, it can be followed with "28". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For i.MX 8M Quad WEVK, it can be `wevk`, `wevk-dual`, `trusty-wevk-dual`, `trusty-wevk-rbidx-blob-dual`, `trusty-secure-unlock-wevk-dual`. For i.MX 8M Quad EVK, it can be `dual`, `trusty-dual`, or `trusty-secure-unlock-dual`. If this option is not used, default `u-boot-imx8mq.imx` is flashed.

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

© 2026 NXP B.V. All rights reserved.

Table 8. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8M Quad WEVK, it can be `wevk`, `dual`, `mipi-panel`, `mipi-panel-rm67191`, or `mipi`. For i.MX 8M Quad EVK, it can be `dual`, `mipi-panel`, `mipi-panel-rm67191`, or `mipi`. If this option is not used, default `dtbo-imx8mq.img` and `vbmeta-imx8mq.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8M Quad WEVK, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no images are flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not executes UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8mq -a -e -d wevk -u trusty-wevk-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8mq -a -e -d wevk -u trusty-wevk-dual
```

When the command above is executed, `spl-imx8mq-trusty-wevk-dual.bin` is flashed, and `bootloader-imx8mq-trusty-wevk-dual.img` with other default images are flashed into eMMC slot a for i.MX 8M Quad WEVK.

Note:

- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash the SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 16 GB or the on-board eMMC is used as the boot device, it does not need to use `-c` option.*
- *If your SD card is 32 GB, execute the tool with `-c 28`.*
- *To test dual-bootloader, execute the tool with `-u dual`.*
- *To test Trusty OS and dual-bootloader both in enabled condition for i.MX 8M Quad EVK, execute the tool with `-u trusty-dual`.*
- *To test Trusty OS and dual-bootloader both in enabled condition for i.MX 8M Quad WEVK, execute the tool with `-u trusty-wevk-dual`.*

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

© 2026 NXP B.V. All rights reserved.

- *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition for i.MX 8MQuad WEVK, execute the tool with `-u trusty-wevk-rbidx-blob-dual`.*
- *To test the demonstration implementation of secure unlock i.MX 8M Quad EVK, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
- *To test the demonstration implementation of secure unlock i.MX 8M Quad WEVK, execute the tool with `-u trusty-secure-unlock-wevk-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
- *To test feature on i.MX 8M Quad WEVK board:*
  - *To test HDMI display, execute the tool with `-d wevk`.*
  - *To test MIPI-to-HDMI display, execute the tool with `-d mipi`.*
  - *To test RM67199 MIPI panel display, execute the tool with `-d mipi-panel`.*
  - *To test RM67191 MIPI panel display, execute the tool with `-d mipi-panel-rm67191`.*
  - *To test HDMI and MIPI-DSI-to-HDMI multiple displays, execute the tool with `-d dual`*
- *To test feature on i.MX 8M Quad EVK board:*
  - *For i.MX 8M Quad EVK board, only HDMI display supports Wi-Fi and Bluetooth function. When connecting to other displays, Wi-Fi and Bluetooth function is not supported.*
  - *To test HDMI output, it does not need to use `-d` option.*
  - *To test MIPI-DSI-to-HDMI output, execute the tool with `-d mipi`.*
  - *To test RM67199 MIPI panel output, execute the tool with `-d mipi-panel`.*
  - *To test RM67191 MIPI panel output, execute the tool with `-d mipi-panel-rm67191`.*
  - *To test HDMI and MIPI-DSI-to-HDMI multiple displays, execute the tool with `-d dual`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is no error, the command window displays the information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change boot device to eMMC or SD card. Change the board's SW802 (boot mode) to 10 (from 1-2 bit) to exit serial download mode.
   - Change SW801 to switch the board back to 1100 (SD boot mode, from 1-4 bit).
   - Change SW801 to switch the board back to 0010 (eMMC boot mode, from 1-4 bit).

6.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

It does not need to set U-Boot environment variables for the currently available boot scenarios.

6.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

7 Working with the i.MX 8ULP EVK board

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

© 2026 NXP B.V. All rights reserved.

7.1 Board hardware

The figure below shows the different components of the i.MX 8ULP EVK board.

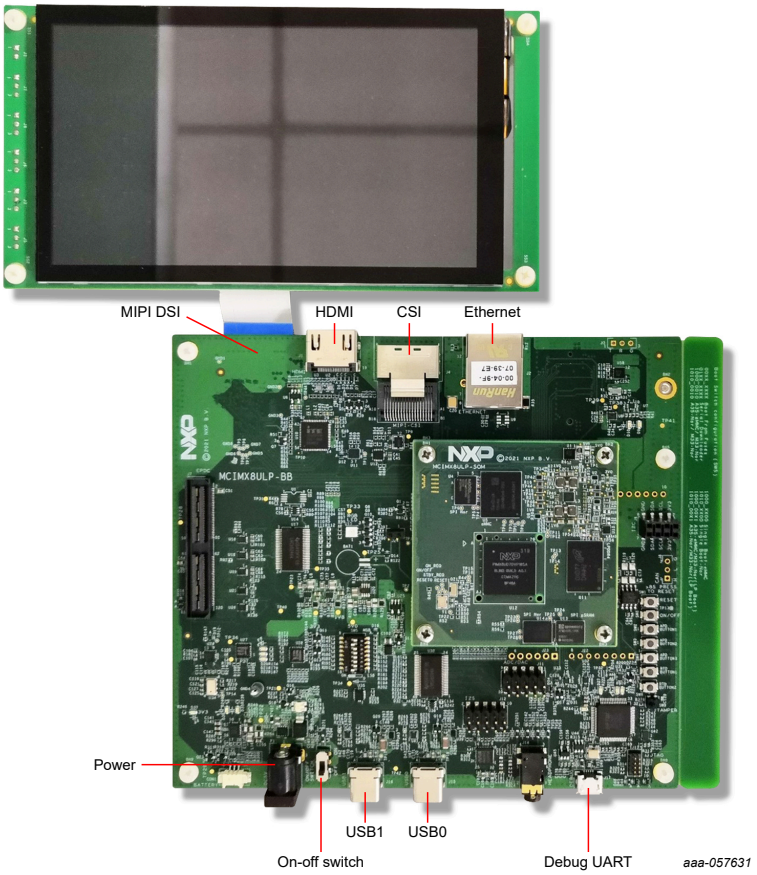


Figure 31. i.MX 8ULP EVK board with MIPI display

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

© 2026 NXP B.V. All rights reserved.

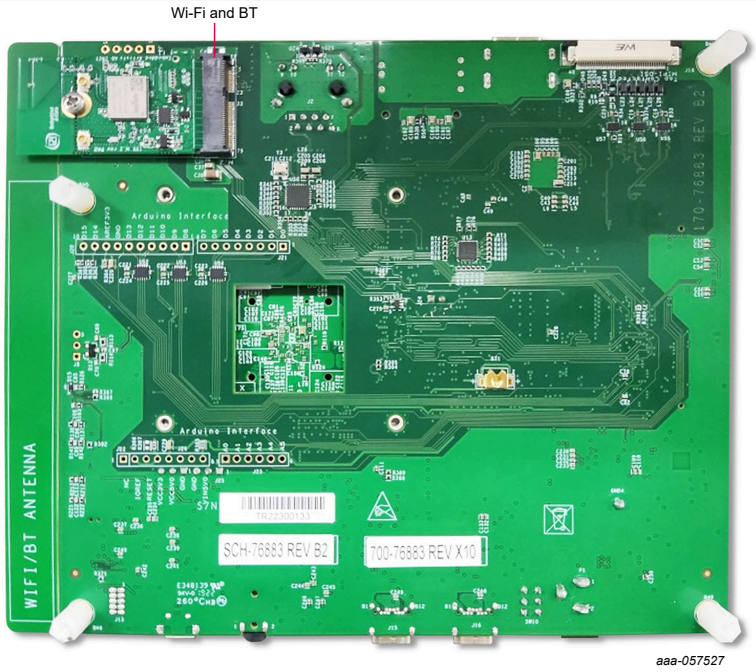


Figure 32. i.MX 8ULP EVK board back view

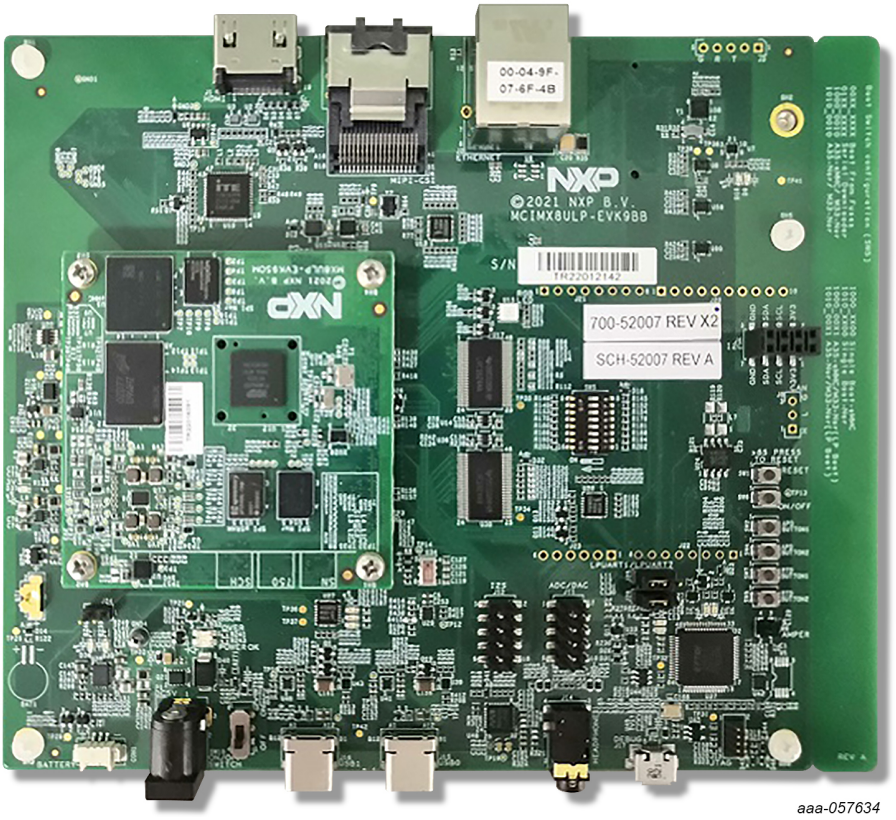


Figure 33. i.MX 8ULP EVK 9x9 board

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

© 2026 NXP B.V. All rights reserved.



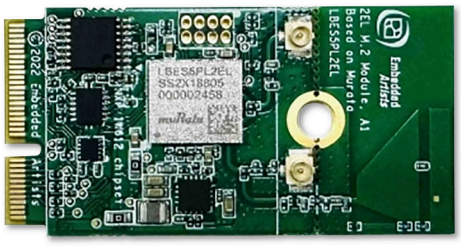
aaa-058037

Figure 34. i.MX CSI MIPI camera



aaa-057635

Figure 35. EPDC display



aaaa-057479

Figure 36. SDIW612 (Murata LBES5PL2EL)

Note:

- *To test the camera, connect the i.MX CSI MIPI camera to the "CSI" port.*
- *To test EPDC display, connect the EPDC display to the base board of i.MX 8ULP EVK, the connector on the base board can be found near the silkprint of "NXP".*
- *The EPDC display board has a power switch, beside which is a DC jack. This DC jack should be connected to the 5 V power to make the EPDC display board work.*
- *i.MX 8ULP EVK 9x9 board is named so because the i.MX 8ULP SoC used on the board is of size 9 mm x 9 mm. Also note that the base board does not have an EPDC interface.*
- *Connect the SDIW612 (Murata LBES5PL2EL) Wi-Fi&BT M.2 expansion card to the J3 connector to enable Wi-Fi and Bluetooth.*

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

© 2026 NXP B.V. All rights reserved.

7.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8ulp-evk.tar.gz`.

Table 9. Board images

Image name	Download target	Descriptions
`spl-imx8ulp-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image without Trusty related configurations for the i.MX 8ULP EVK board.
`spl-imx8ulp-trusty-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image with Trusty related configurations for the i.MX 8ULP EVK board.
`spl-imx8ulp-9x9-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image without Trusty related configurations for the i.MX 8ULP EVK 9x9 board.
`spl-imx8ulp-trusty-9x9-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image with Trusty related configurations for the i.MX 8ULP EVK 9x9 board.
`spl-imx8ulp-trusty-9x9-rbidx-blob-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for i.MX 8ULP EVK 9x9 board.
`spl-imx8ulp-trusty-lpa-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image with Trusty related configurations and Low Power Audio enabled for the i.MX 8ULP EVK board.
`spl-imx8ulp-trusty-dualboot-dual.bin`	0 offset of eMMC boot partition.	Secondary program loader image with Trusty related configurations and dual boot enabled (low-power display) for the i.MX 8ULP EVK board.
`bootloader-imx8ulp-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8ULP EVK board.
`bootloader-imx8ulp-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8ULP EVK board.
`bootloader-imx8ulp-9x9-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8ULP 9x9 EVK board.
`bootloader-imx8ulp-trusty-9x9-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8ULP 9x9 EVK board.
`bootloader-imx8ulp-trusty-9x9-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8ULP 9x9 EVK board.
`bootloader-imx8ulp-trusty-lpa-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Low Power Audio MCU firmware, and Trusty OS. It is for the i.MX 8ULP EVK board.
`bootloader-imx8ulp-trusty-dualboot-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8ULP EVK board.
`u-boot-imx8ulp-9x9.imx`	0 offset of eMMC boot partition.	An image containing U-Boot and ATF for the i.MX 8ULP EVK 9x9 board.
`u-boot-imx8ulp-9x9-evk-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8ULP EVK 9x9 board. It is not flashed to MMC.
`u-boot-imx8ulp.imx`	0 offset of eMMC boot partition.	An image containing U-Boot and ATF for the i.MX 8ULP EVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 9. Board images*...continued*

Image name	Download target	Descriptions
u-boot-imx8ulp-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8ULP EVK. It is not flashed to MMC.
imx8ulp_mcu_demo_sf.img	NOR flash	MCU image that contains a low-power display demo. It is not flashed to the serial NOR flash on the MCU side.
partition-table-13GB.img	0 offset of eMMC user data area.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-13GB-dual.img	0 offset of eMMC user data area.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
partition-table.img	0 offset of eMMC user data area.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 offset of eMMC user data area.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in the VTS test with the GSI system image.
vbmeta-imx8ulp-9x9-hdmi.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support HDMI output on the i.MX 8ULP EVK 9x9 board.
vbmeta-imx8ulp-9x9.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI panel output on the i.MX 8ULP EVK 9x9 board.
vbmeta-imx8ulp.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI panel output on the i.MX 8ULP EVK.
vbmeta-imx8ulp-hdmi.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support HDMI output on the i.MX 8ULP EVK.
vbmeta-imx8ulp-epdc.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support EPDC output on the i.MX 8ULP EVK.
vbmeta-imx8ulp-sof.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Sound Open Firmware audio output on the i.MX 8ULP EVK.
vbmeta-imx8ulp-lpa.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Low Power Audio feature on the i.MX 8ULP EVK.
vbmeta-imx8ulp-lpd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Low Power Display feature on the i.MX 8ULP EVK.

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

© 2026 NXP B.V. All rights reserved.

Table 9. Board images...*continued*

Image name	Download target	Descriptions
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx8ulp-9x9-hdmi.img	dtbo_a and dtbo_b partitions.	Device Tree image to support HDMI output on the i.MX 8ULP EVK 9x9 board.
dtbo-imx8ulp-9x9.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI panel output on the i.MX 8ULP EVK 9x9 board.
dtbo-imx8ulp.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI panel output on the i.MX 8ULP EVK.
dtbo-imx8ulp-hdmi.img	dtbo_a and dtbo_b partitions.	Device Tree image to support HDMI output on the i.MX 8ULP EVK.
dtbo-imx8ulp-epdc.img	dtbo_a and dtbo_b partitions.	Device Tree image to support EPDC output on the i.MX 8ULP EVK.
dtbo-imx8ulp-sof.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Sound Open Firmware audio output on the i.MX 8ULP EVK.
dtbo-imx8ulp-lpa.img	dtbo_a and dtbo_b partitions.	Device Tree image to support Low Power Audio on the i.MX 8ULP EVK.
dtbo-imx8ulp-lpd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Low Power Display on the i.MX 8ULP EVK.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key. It can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key. It is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

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

© 2026 NXP B.V. All rights reserved.

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

7.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   Change the boot switch SW5 to 00000010 (from 1-8 bit) to enter serial download mode.
3. Power on the board. Use the USB cable to connect the PC with the board through the USB0 port on the board.
4. Decompress `release_package/android-17.0.0_1.0.0_image_8ulpevk.tar.gz`, which contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For the i.MX 8ULP EVK board, related options are described as follows.

Table 10. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8ULP EVK, it should be `imx8ulp`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. i.MX 8ULP EVK, it can be followed with "13". If this option is not used, default `partition-table.img` is flashed.
`-m`	Flashes the MCU image. If this option is not used, the MCU image is not flashed.

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

© 2026 NXP B.V. All rights reserved.

Table 10. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For i.MX 8ULP EVK, it can be `9x9`, `dual`, `9x9-dual`, `trusty-dual`, `trusty-9x9-dual`, `trusty-9x9-rbidx-blob-dual`, `trusty-lpa-dual`, `trusty-dualboot-dual`, `trusty-dualboot-dual`. If this option is not used, the default `u-boot-imx8ulp.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8ULP EVK, it can be `9x9`, `9x9-hdmi`, `hdmi`, `epdc`, `sof`, `lpa` and `lpd`. If this option is not used, default `dtbo-imx8ulp.img` and `vbmeta-imx8ulp.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in current working directory are flashed.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not executes UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8ulp -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8ulp -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8ulp-trusty-dual.bin` is flashed, and `bootloader-imx8ulp-trusty-dual.img` with other default images are flashed into the eMMC slot for i.MX 8ULP EVK.

Note:

- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *For i.MX 8ULP EVK:*
  - *To test dual-bootloader, execute the tool with `-u dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
  - *To test the MIPI panel display, it does not need to use the `-d` option.*
  - *To test the HDMI display, execute the tool with `-d hdmi`.*
  - *To test the EPDC display, execute the tool with `-d epdc`.*
  - *To test low-power audio, execute the tool with `-d lpa -u trusty-lpa-dual -m`. See the Android User's Guide (UG10156) for more steps to make the low-power audio work.*
  - *To test low-power display, execute the tool with `-d lpd -u trusty-dualboot-dual -m`. See the Android User's Guide (UG10156) for more steps to make the low-power display work.*
- *For i.MX 8ULP EVK 9x9 board:*
  - *To boot up the i.MX 8ULP EVK 9x9 board, ensure that you execute the tool with `-u 9x9`.*

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

© 2026 NXP B.V. All rights reserved.

- *To test dual-bootloader, execute the tool with `-u 9x9-dual`.*
- *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-9X9-dual`.*
- *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-9x9-rbidx-blob-dual`.*
- *To test the MIPI panel display, execute the tool with `-d 9x9`.*
- *To test the HDMI display, execute the tool with `-d 9x9-hdmi`.*

- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is no error, the command window displays the information indicating that images are already flashed.
7. Power off the board.
8. Change the boot device to eMMC.
   - Change SW5 to 00000001 (from 1-8 bit) to boot from eMMC.
   - Change SW5 to 01000001 (dual boot mode, from 1-8 bit), The Cortex-A core is booted from eMMC, and the Cortex-M core is booted from serial flash.

7.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

7.4.1 Booting with an HDMI display

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.lcd_density=120
U-Boot > saveenv
```

Note:

If other boot arguments need to be appended to `bootargs` or `bootconfig`, set them together in one `setenv append_bootargs` or `setenv append_bootconfig` command.

As the camera orientation is set to **90** for the default image (MIPI Panel, 720x1280), for the HDMI display (720x480), the orientation needs to be set to **0**. To try the camera with the prebuilt image without modifying the code and rebuilding the images, follow the instructions below on the host after the system boots up:

```
adb reboot
adb remount
adb pull /vendor/etc/configs/camera_config_imx8ulp.json
# set "orientation" to 0 in the json file.
adb push camera_config_imx8ulp.json /vendor/etc/configs
adb reboot
```

7.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

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

© 2026 NXP B.V. All rights reserved.

8 Working with the i.MX 8QuadMax MEK board

8.1 Board hardware

The figures below show the different components of the i.MX 8QuadMax MEK board.

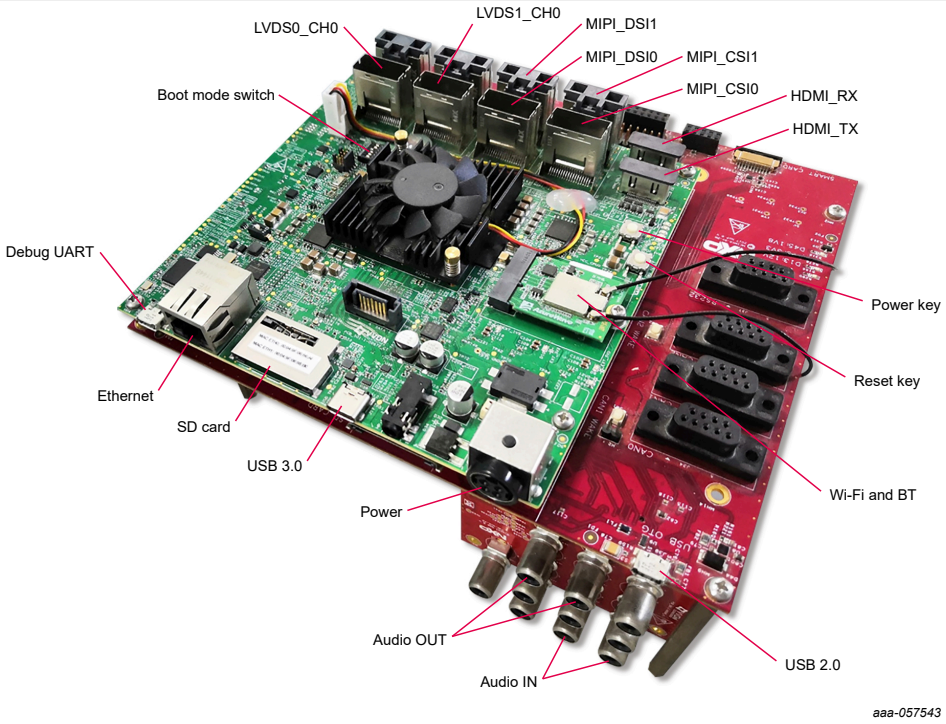


Figure 37. i.MX 8QuadMax MEK board



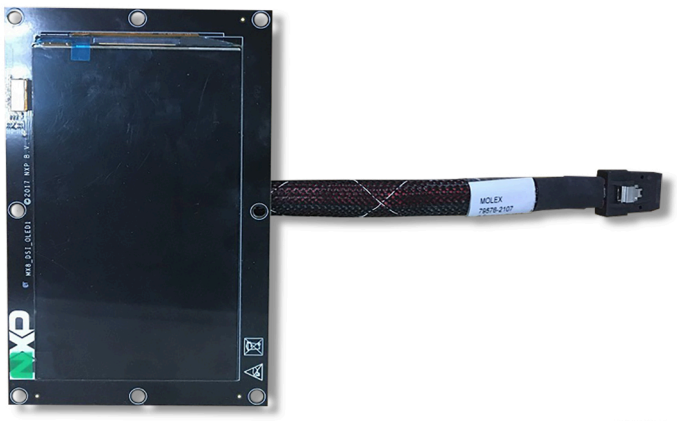
Figure 38. i.MX mini SAS cable with DSI-to-HDMI adapter



Figure 39. i.MX mini SAS cable with LVDS-to-HDMI adapter

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

© 2026 NXP B.V. All rights reserved.



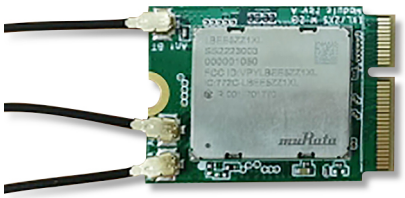
aaa-057542

Figure 40. MX8-DSI-OLED1 MIPI panel



aaa-058037

Figure 41. i.MX MIPI camera



aaa-057632

Figure 42. PCIE9098 (Murata LBEE5ZZ1XL)

Note:

- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "HDMI DSI" port.*
- *To test the LVDS-to-HDMI display, use the i.MX mini SAS cable to connect the LVDS-to-HDMI adapter to the "LVDS0/LVDS1" port.*
- *To test the camera, connect two OV5640 CSI MIPI cameras to the "MIPI Camera0" and "MIPI Camera1" ports.*
- *To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port. OLED1 (RM67191) MIPI panel is supported, while OLED1A (RM67199) is not supported.*
- *To test single camera, connect OV5640 CSI MIPI camera to the "MIPI Camera0" or "MIPI Camera1" port.*
- *To test dual displays, use the i.MX mini SAS cable to connect both "LVDS0" and "LVDS1" ports with the LVDS-to-HDMI adapter.*
- *Connect the PCIE9098 (Murata LBEE5ZZ1XL) Wi-Fi&BT M.2 expansion card to the J12 connector to enable Wi-Fi and Bluetooth.*

8.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8qmek.tar.gz`.

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

© 2026 NXP B.V. All rights reserved.

Table 11. Board images

Image name	Download target	Description
spl-imx8qm-dual.bin	0 kB offset of eMMC or 32 kB offset of SD card.	Secondary program loader image without Trusty related configurations for the i.MX 8QuadMax MEK board.
bootloader-imx8qm-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8QuadMax MEK board.
spl-imx8qm-trusty-dual.bin	0 kB offset of eMMC.	Secondary program loader image with Trusty related configurations for the i.MX 8QuadMax MEK board.
bootloader-imx8qm-trusty-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8QuadMax MEK board.
spl-imx8qm-trusty-rbidx-blob-dual.bin	0 kB offset of eMMC.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 8Quad Max MEK board.
bootloader-imx8qm-trusty-rbidx-blob-dual.img | bootloader_a and bootloader_b partitions.	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8Quad Max MEK board.
spl-imx8qm-trusty-secure-unlock-dual.bin	0 kB offset of eMMC.	Secondary program loader image with Trusty and secure unlock related configurations for i.MX 8QuadMax EVK board.
bootloader-imx8qm-trusty-secure-unlock-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 8QuadMax EVK board.
u-boot-imx8qm.imx	0 kB offset of eMMC and 32 kB offset of SD card.	An image containing U-Boot and ATF for the i.MX 8Quad Max MEK board.
u-boot-imx8qm-hdmi.imx	0 kB offset of eMMC and 32 kB offset of SD card.	An image containing U-Boot and ATF for the i.MX 8Quad Max MEK board to support the physical HDMI display.
u-boot-imx8qm-md.imx	0 kB offset of eMMC and 32 kB offset of SD card.	An image containing U-Boot and ATF for i.MX 8QuadMax MEK board to support multiple displays.
u-boot-imx8qm-mek-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8QuadMax MEK board. It is not flashed to MMC.
partition-table-13GB.img	Programs to the first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for single-bootloader condition and for boot storage larger than 13 GB.
partition-table-13GB-dual.img	Programs to the first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for dual-bootloader condition and for boot storage larger than 13 GB.
partition-table.img	Programs to the first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for single-bootloader condition and for boot storage larger than 28 GB.
partition-table-dual.img	Programs to the first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for dual-bootloader condition and for boot storage larger than 28 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.

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

© 2026 NXP B.V. All rights reserved.

Table 11. Board images...*continued*

Image name	Download target	Description
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in the VTS test with the GSI system image.
vbmeta-imx8qm.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display with dual camera.
vbmeta-imx8qm-ov5640-csi0.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display with CSI0 camera.
vbmeta-imx8qm-ov5640-csi1.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display with CSI1 camera.
vbmeta-imx8qm-md.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support multiple displays.
vbmeta-imx8qm-hdmi.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the physical HDMI display.
vbmeta-imx8qm-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the OLED1 (RM67199) MIPI panel display.
vbmeta-imx8qm-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the OLED1 (RM67191) MIPI panel display.
vbmeta-imx8qm-hdmi-rx.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI input.
vbmeta-imx8qm-lvds1-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel display.
vbmeta-imx8qm-sof.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Sound Open Firmware audio output.
vbmeta-imx8qm-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display on i.MX 8QuadMax MEK REVD board with dual camera.
vbmeta-imx8qm-ov5640-csi0-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display on i.MX 8QuadMax MEK REVD board with CSI0 camera.
vbmeta-imx8qm-ov5640-csi1-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI/MIPI-to-HDMI display on i.MX 8QuadMax MEK REVD board with CSI1 camera.
vbmeta-imx8qm-md-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support multiple displays on the i.MX 8QuadMax MEK REVD board.
vbmeta-imx8qm-hdmi-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the physical HDMI display on the i.MX 8QuadMax MEK REVD board.
vbmeta-imx8qm-mipi-panel-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI panel (RM67199) display on the i.MX 8QuadMax MEK REVD board.
vbmeta-imx8qm-mipi-panel-rm67191-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI panel (RM67191) display on the i.MX 8QuadMax MEK REVD board.

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

© 2026 NXP B.V. All rights reserved.

Table 11. Board images...*continued*

Image name	Download target	Description
vbmeta-imx8qm-hdmi-rx-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI input on the i.MX 8QuadMax MEK REVD board.
vbmeta-imx8qm-lvds1-panel-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel display on the i.MX 8QuadMax MEK REVD board.
vbmeta-imx8qm-sof-revd.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Sound Open Firmware audio output on the i.MX 8QuadMax MEK REVD board.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx8qm.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI/MIPI-to-HDMI displays with dual camera.
dtbo-imx8qm-ov5640-csi0.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI/MIPI-to-HDMI displays with CSI0 camera.
dtbo-imx8qm-ov5640-csi1.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI/MIPI-to-HDMI displays with CSI1 camera.
dtbo-imx8qm-md.img	dtbo_a and dtbo_b partitions.	Device Tree image to support multiple displays.
dtbo-imx8qm-hdmi.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the physical HDMI display.
dtbo-imx8qm-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the OLED1 (RM67199) MIPI panel display.
dtbo-imx8qm-mipi-panel-rm67191.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the OLED1 (RM67191) MIPI panel display.
dtbo-imx8qm-hdmi-rx.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI input.

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

© 2026 NXP B.V. All rights reserved.

Table 11. Board images...*continued*

Image name	Download target	Description
dtbo-imx8qm-lvds1-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel display.
dtbo-imx8qm-sof.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Sound Open Firmware audio output.
dtbo-imx8qm-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8QuadMax MEK REVD board with dual camera.
dtbo-imx8qm-ov5640-csi0-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8QuadMax MEK REVD board with CSI0 camera.
dtbo-imx8qm-ov5640-csi1-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the i.MX 8QuadMax MEK REVD board with CSI1 camera.
dtbo-imx8qm-md-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support multiple displays on the i.MX 8QuadMax MEK REVD board.
dtbo-imx8qm-hdmi-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the physical HDMI display on the i.MX 8QuadMax MEK REVD board.
dtbo-imx8qm-mipi-panel-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI panel (RM67199) display on the i.MX 8QuadMax MEK REVD board.
dtbo-imx8qm-mipi-panel-rm67191-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI panel (RM67191) display on the i.MX 8QuadMax MEK REVD board.
dtbo-imx8qm-hdmi-rx-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI input on the i.MX 8 QuadMax MEK REVD board.
dtbo-imx8qm-lvds1-panel-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel display on the i.MX 8QuadMax MEK REVD board.
dtbo-imx8qm-sof-revd.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Sound Open Firmware audio output on the i.MX 8QuadMax MEK REVD board.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key. It can be used to set the RPMB key to fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key, which is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

8.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

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

© 2026 NXP B.V. All rights reserved.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   Change the board's SW2 (boot mode) to 001000 (from 1-6 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board USB 3.0 port to connect your PC with the board.
   Note:
   - *There are three USB ports on the i.MX 8QuadMax MEK board: USB-to-UART, USB 2.0, and USB 3.0.*
   - *The USB-to-UART port can be referenced as debug UART, which can be used to watch the log of the hardware boot processing.*
   - *USB 2.0 is USB Host and USB 3.0 is USB OTG.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_8qmek.tar.gz`, which contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For i.MX 8QuadMax board, related options are described as follows.

Table 12. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8QuadMax MEK, it should be `imx8qm`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8QuadMax, it can be followed with "13". If this option is not used, default `partition-table.img` is flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&Bootloader images with `uboot_feature` in their names. For i.MX 8QuadMax MEK, it can be `hdmi`, `md`, `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, or `trusty-secure-unlock-dual`. If this option is not used, the default `u-boot-imx8qm.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8Quad Max MEK, it can be `ov5640-csi0`, `ov5640-csi1`, `hdmi-rx`, `hdmi`, `md`, `mipi-panel`, `mipi-panel-rm67191`, `lvds1-panel`, or `sof`. For i.MX 8QuadMax MEK REVD, it can be `revd`, `ov5640-csi0-revd`, `ov5640-csi1-revd`, `hdmi-rx-revd`, `hdmi-revd`, `md-revd`, `mipi-panel-revd`, `mipi-panel-rm67191-revd`, `lvds1-panel-`

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

© 2026 NXP B.V. All rights reserved.

Table 12. Options for uuu_imx_android_flash tool*...continued*

Option	Description
	`revd`, or `sof-revd`. If this option is not used, the default `dtbo-imx8qm.img` and `vbmeta-imx8qm.img` are flashed.
-e	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8QuadMax, it can be `emmc` or `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in Daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not executes UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8qm -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8qm -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8qm-trusty-dual.bin` is flashed, and `bootloader-imx8qm-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 8QuadMax MEK.

Note:

- *If `-d` the option is not used or `-d revd` is used, two OV5640 must be connected to the board, or neither OV5640 can work.*
- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash the SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or the on-board eMMC is used as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test dual-bootloader, execute the tool with `-u dual`.*
- *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
- *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
- *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*

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

© 2026 NXP B.V. All rights reserved.

- *To test the LVDS-to-HDMI/MIPI-DSI-to-HDMI display, it does not need to use the `-d` option. Four display ports of LVDS0_CH0, LVDS1_CH0, MIPI_DSI0, and MIPI_DSI1 can be used for display individually or together.*
- *To test the MIPI panel (RM67199) display, execute the tool with `-d mipi-panel`. The MIPI_DSI0 port is used for display.*
- *To test the MIPI panel (RM67191) display, execute the tool with `-d mipi-panel-rm67191`. The MIPI_DSI0 port is used for display.*
- *To test the LVDS panel display, execute the tool with `-d lvds1-panel`. The LVDS1 port is used for display. The connection is channel 0 (J8) -> LCD J9, channel 1 (J3) -> LCD J8.*
- *To test the physical HDMI display, execute the tool with `-u hdmi -d hdmi`. The HDMI_TX port is used for display.*
- *To test the physical HDMI RX as the external camera input device, execute the tool with `-u hdmi -d hdmi-rx`. The HDMI_RX port is used for the HDMI RX.*
- *To plug single-OV5640 on the CSI0 port or CSI1 port, flash the image with `-d ov5640-csi0` or `-d ov5640-csi1`.*
- *To test the Sound Open Firmware audio output, execute the tool with `-d sof`.*
- *To test multiple displays, execute the tool with `-u md -d md`. Four display ports can be used for display:*
  – If HDMI_TX is used, the other three ports are LVDS0_CH0, LVDS1_CH0, and MIPI_DSI1.
  – If HDMI_TX is not used, the four ports are LVDS0_CH0, LVDS1_CH0, MIPI_DSI0, and MIPI_DSI1.
- *To test the i.MX 8QuadMax MEK REVD board (the board with tags "700-29420 REV D" and "SCH-29420 REV E" on the back of the board), the `-d` option with the parameter containing `revd` needs to be used, the corresponding `-d` option parameters are `revd`, `ov5640-csi0-revd`, `ov5640-csi1-revd`, `hdmi-rx-revd`, `hdmi-revd`, `md-revd`, `mipi-panel-revd`, `mipi-panel-rm67191-revd`, `lvds1-panel-revd`, and `sof-revd`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash execution` to complete. If there is no error, the command window displays the information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change boot device to eMMC or SD card.
   - Change SW2 to switch the board back to 000100 (from 1-6 bit) to enter eMMC boot mode.
   - Change SW2 to switch the board back to 001100 (from 1-6 bit) to enter SD boot mode.

8.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios.

8.4.1 Booting with multi-channel audio support

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.audio.multichannel=1
```


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

© 2026 NXP B.V. All rights reserved.

```
U-Boot > saveenv
```

8.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

9 Working with the i.MX 8QuadXPlus MEK board

9.1 Board hardware

The figures below show the different components of the i.MX 8QuadXPlus MEK board.

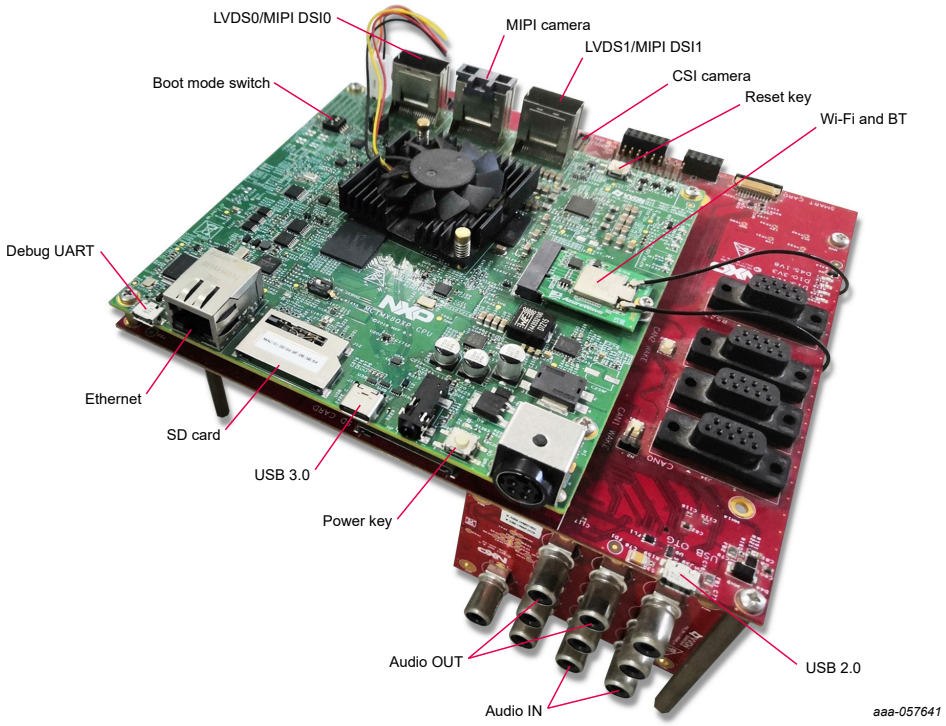


Figure 43. i.MX 8QuadXPlus MEK board



Figure 44. i.MX mini SAS cable with DSI-to-HDMI adapter



Figure 45. i.MX mini SAS cable with LVDS-to-HDMI adapter

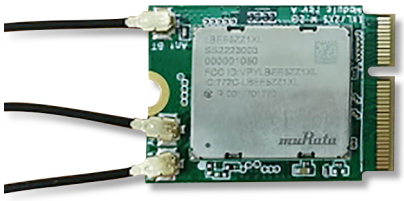
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-058037

Figure 46. OV5640 CSI MIPI camera



aaa-057632

Figure 47. PCIE9098 (Murata LBEE5ZZ1XL)

Note:

- *To test the MIPI-DSI to HDMI display, use the i.MX mini SAS cable to connect the DSI to HDMI adapter to the "MIPI DSI" port.*
- *To test the LVDS-to-HDMI display, use the i.MX mini SAS cable to connect the LVDS-to-HDMI adapter to the "LVDS0/LVDS1" port.*
- *To test a single camera, connect the OV5640 CSI MIPI camera to the "MIPI Camera" port or connect OV5640 Camera to the "CSI Camera" port.*
- *Connect the PCIE9098 (Murata LBEE5ZZ1XL) Wi-Fi&BT M.2 expansion card to the J5 connector to enable Wi-Fi and Bluetooth.*

9.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_8qmek.tar.gz`.

Table 13. Board images

Image name	Download target	Description
`spl-imx8qxp-dual.bin`	32 kB offset of MMC.	Secondary program loader image without Trusty related configurations for the i.MX 8QuadXPlus MEK board with the Silicon revision B0 chip.
`bootloader-imx8qxp-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8 QuadXPlus MEK board with the Silicon revision B0 chip.
`spl-imx8qxp-c0-dual.bin`	0 kB offset of eMMC or 32 kB offset of SD card.	Secondary program loader image without Trusty related configurations for the i.MX 8QuadXPlus MEK board with the Silicon revision C0 chip.
`bootloader-imx8qxp-c0-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 8 QuadXPlus MEK board with the Silicon revision C0 chip.
`spl-imx8qxp-trusty-dual.bin`	32 kB offset of MMC.	Secondary program loader image with Trusty related configurations for the i.MX 8QuadXPlus MEK board with the Silicon revision B0 chip.
`bootloader-imx8qxp-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8QuadXPlus MEK board with the Silicon revision B0 chip.

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

© 2026 NXP B.V. All rights reserved.

Table 13. Board images*...continued*

Image name	Download target	Description
`spl-imx8qxp-trusty-secure-unlock-dual.bin`	32 kB offset of MMC.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 8QuadXPlus EVK board with the Silicon revision B0 chip.
`bootloader-imx8qxp-trusty-secure-unlock-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of secure unlock mechanism for the i.MX 8Quad XPlus EVK board with the Silicon revision B0 chip.
`spl-imx8qxp-trusty-c0-dual.bin`	0 kB offset of eMMC.	Secondary program loader image with Trusty related configurations for the i.MX 8QuadXPlus MEK board with Silicon revision C0 chip.
`bootloader-imx8qxp-trusty-c0-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 8QuadXPlus MEK board with Silicon revision C0 chip.
`spl-imx8qxp-trusty-rbidx-blob-dual.bin`	32 kB offset of eMMC.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 8Quad XPlus MEK board with Silicon revision B0 chip.
`bootloader-imx8qxp-trusty-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 8QuadXPlus MEK board with Silicon revision B0 chip.
`u-boot-imx8qxp.imx`	32 kB offset of MMC.	An image containing U-Boot and ATF for the i.MX 8QuadXPlus MEK board with the Silicon revision B0 chip.
`u-boot-imx8qxp-c0.imx`	0 kB offset of MMC or 32 kB offset of SD card.	An image containing U-Boot and ATF for the i.MX 8QuadXPlus MEK board with Silicon revision C0 chip.
`u-boot-imx8qxp-mek-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8 QuadXPlus MEK board with the Silicon revision B0 chip. It is not flashed to MMC.
`u-boot-imx8qxp-mek-c0-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 8QuadXPlus MEK board with Silicon revision C0 chip. It is not flashed to MMC.
`partition-table-13GB.img`	Programmed to first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for single-bootloader condition and for boot storage larger than 13 GB.
`partition-table-13GB-dual.img`	Programmed to first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for dual-bootloader condition and for boot storage larger than 13 GB.
`partition-table.img`	Programmed to first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for single-bootloader condition and for boot storage larger than 28 GB.
`partition-table-dual.img`	Programmed to first 17 kB, and then back up to last 17 kB of the boot storage.	GPT partition table image used for dual-bootloader condition and for boot storage larger than 28 GB.
`boot.img`	`boot_a` and `boot_b` partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
`boot-imx.img`	`boot_a` and `boot_b` partitions.	Boot image built with i.MX kernel tree for debugging.

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

© 2026 NXP B.V. All rights reserved.

Table 13. Board images...*continued*

Image name	Download target	Description
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in the VTS test with GSI system image.
vbmeta-imx8qxp.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support single LVDS-to-HDMI/MIPI-DSI-to-HDMI or dual LVDS-to-HDMI displays with dual-cameras support.
vbmeta-imx8qxp-ov5640-csi.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display with CSI camera support.
vbmeta-imx8qxp-ov5640-parallel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display with parallel camera support.
vbmeta-imx8qxp-lvds0-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel display.
vbmeta-imx8qxp-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI panel (RM67199) display.
vbmeta-imx8qxp-mipi-panel-rm67191.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI panel (RM67191) display.
vbmeta-imx8qxp-sof.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Sound Open Firmware audio output.
system.img	Logical partitions system_a and logical partition system_b in super partition.	System image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image which contains images for logical partitions.

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

© 2026 NXP B.V. All rights reserved.

Table 13. Board images...*continued*

Image name	Download target	Description
dtbo-imx8qxp.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the single LVDS-to-HDMI/MIPI-DSI-to-HDMI or dual LVDS-to-HDMI displays with dual-camera support.
dtbo-imx8qxp-ov5640-csi.img	dtbo_a and dtbo_b partitions.	Device Tree image to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display with CSI camera support.
dtbo-imx8qxp-ov5640-parallel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display with parallel camera support.
dtbo-imx8qxp-lvds0-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel display.
dtbo-imx8qxp-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI panel (RM67199) display.
dtbo-imx8qxp-mipi-panel-rm67191.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI panel (RM67191) display.
dtbo-imx8qxp-sof.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Sound Open Firmware audio output.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key. It can be used to set the RPMB key to fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key, which is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note: *In this document, logical partition is also called dynamic partition. For more Android dynamic partitions, see https://source.android.com/docs/core/ota/dynamic_partitions.*

*With virtual A/B enabled, the `super.img` file generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images. To boot from slot b, take either of the following actions:*

- *Apply an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

9.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to automatically flash all Android images.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

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

© 2026 NXP B.V. All rights reserved.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   Change the board's SW2 (boot mode) to 1000 (from 1-4 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board USB 3.0 Type-C port to connect your PC with the board.
   Note:
   - *There are three USB ports on the 8QuadXPlus MEK board: USB-to-UART, USB 2.0, and USB 3.0.*
   - *The USB-to-UART port can be referenced as debug UART, which can be used to watch the log of the hardware boot processing.*
   - *USB 2.0 is USB Host and USB 3.0 is USB OTG.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_8qmek.tar.gz`, which contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For 8QuadXPlus board, related options are described as follows.

Table 14. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies SoC information. For i.MX 8QuadXPlus MEK, it should be `imx8qxp`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and the `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and the `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 8QuadXPlus MEK, it can be followed with "13". If this option is not used, default `partition-table.img` is flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For i.MX 8QuadXPlus MEK, it can be `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, `trusty-secure-unlock-dual`, `c0`, `c0-dual` or `trusty-c0-dual`. If this option is not used, the default `u-boot-imx8qxp.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. For i.MX 8 QuadXPlus MEK, it can be `ov5640-csi`, `ov5640-parallel`, `sof`, `mipi-panel`, `mipi-panel-rm67191`, or `lvds0-panel`. If this option is not used, the default `dtbo-imx8qxp.img` and `vbmeta-imx8qxp.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 8QuadXPlus MEK, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.

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

© 2026 NXP B.V. All rights reserved.

Table 14. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not executes UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx8qxp -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> uuu_imx_android_flash.bat -f imx8qxp -a -e -u trusty-dual
```

When the command above is executed, `spl-imx8qxp-trusty-dual.bin` is flashed, and `bootloader-imx8qxp-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 8QuadXPlus MEK board with silicon revision B0 chip.

Note:

- *if the `-d` option is not used, dual-OV5640 must be plugged, or neither OV5640 can work.*
- *If single OV5640 is plugged to CSI0 port or PARALLEL port, flash image with para `-d ov5640-csi` or `-d ov5640-parallel`.*
- *`-b` is not suggested to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash the SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or the on-board eMMC is used as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test dual-bootloader, execute the tool with `-u dual`.*
- *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
- *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
- *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
- *To test the Sound Open Firmware audio output, execute the tool with `-d sof`.*
- *To test the MIPI panel (RM67199) display, execute the tool with `-d mipi-panel`. The MIPI_DSI0 port is used for display.*
- *To test the MIPI panel (RM67191) display, execute the tool with `-d mipi-panel-rm67191`. The MIPI_DSI0 port is used for display.*
- *To test the LVDS panel display, execute the tool with `-d lvds0-panel`. The LVDS0/1 port is used for display. The connection is LVDS 0 (J3) -> LCD J9, LVDS 1 (J1) -> LCD J8.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```


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

© 2026 NXP B.V. All rights reserved.

"z" in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is no error, the command window displays the information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change boot device to eMMC or SD card.
   - Change SW2 to switch the board back to 0100 (from 1-4 bit) to enter eMMC boot mode.
   - Change SW2 to switch the board back to 1100 (from 1-4 bit) to enter SD boot mode.

9.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios.

9.4.1 Booting with multi-channel audio support

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.audio.multichannel=1
U-Boot > saveenv
```

9.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

10 Working with the i.MX 95 EVK and FRDM-IMX95 boards

10.1 Board hardware

The figures below show the different components of the i.MX 95 EVK board.

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

© 2026 NXP B.V. All rights reserved.

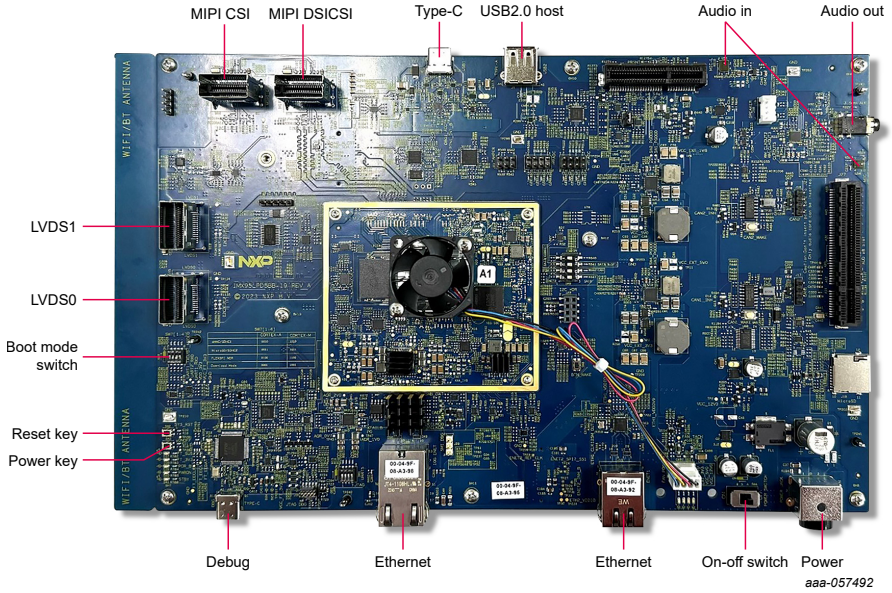


Figure 48. i.MX 95 EVK

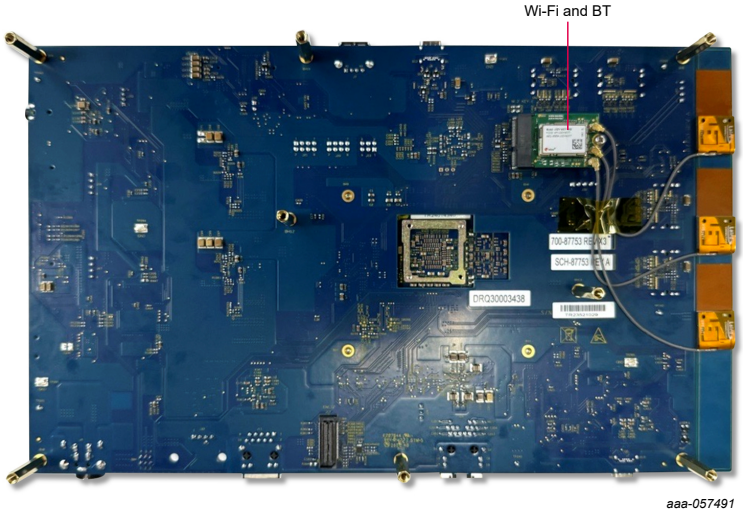


Figure 49. i.MX 95 EVK board back view

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

© 2026 NXP B.V. All rights reserved.

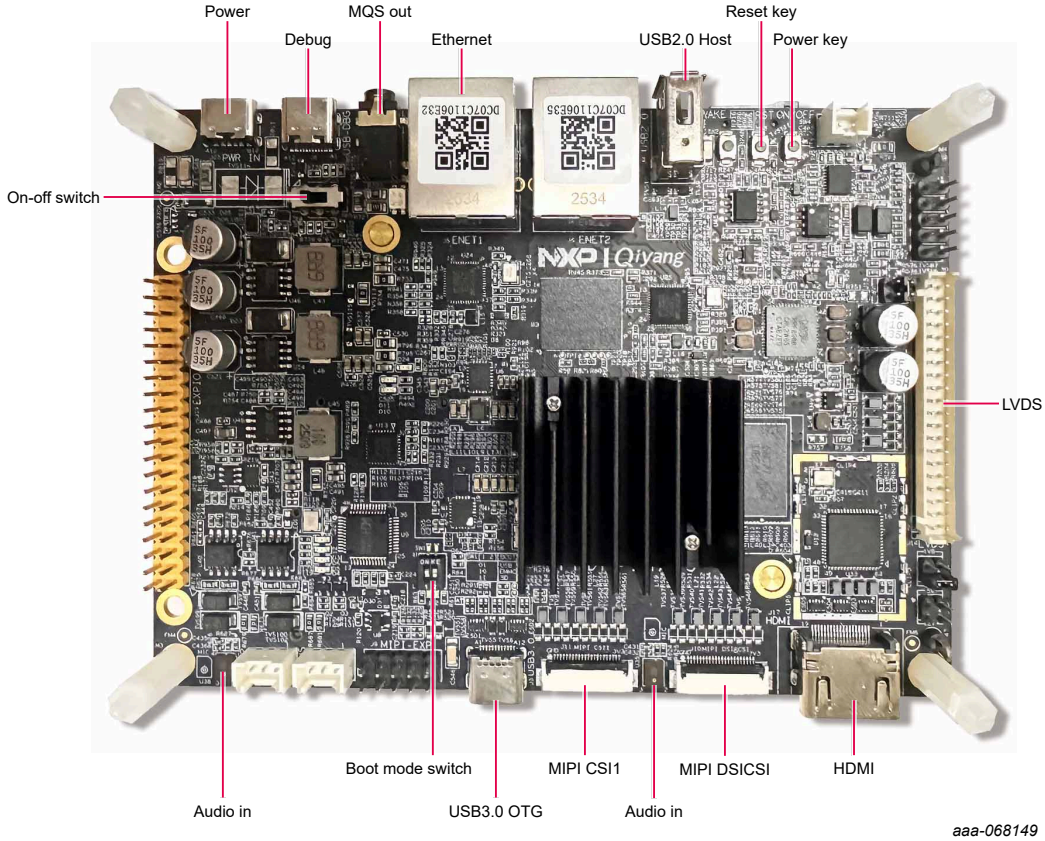


Figure 50. FRDM-IMX95

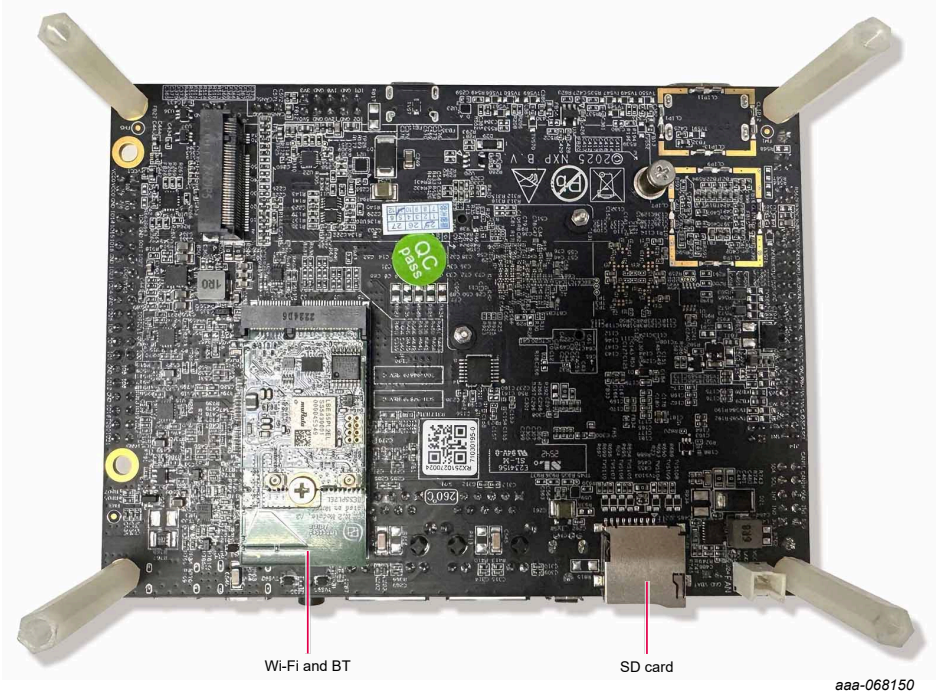


Figure 51. FRDM-IMX95 back view

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

© 2026 NXP B.V. All rights reserved.



aaa-057489

Figure 52. i.MX mini SAS cable with DSI-to-HDMI adapter



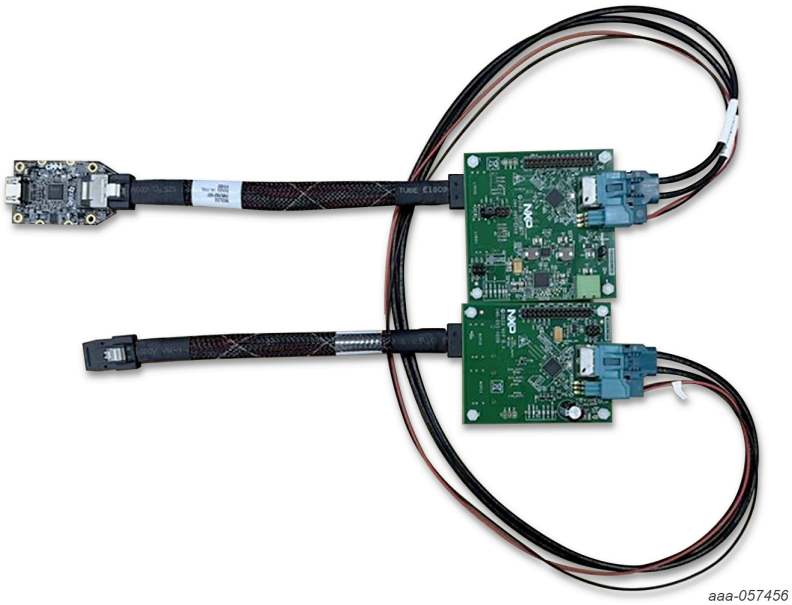
aaa-057488

Figure 53. i.MX mini SAS cable with LVDS-to-HDMI adapter



aaa-057457

Figure 54. i.MX mini SAS cable with DSI-to-HDMI4K adapter



aaa-057456

Figure 55. i.MX mini SAS cable with DSI-to-SDSB (Serialiser and Deserialiser) adapter



aaa-057483

Figure 56. i.MX MIPI panel RM692C9

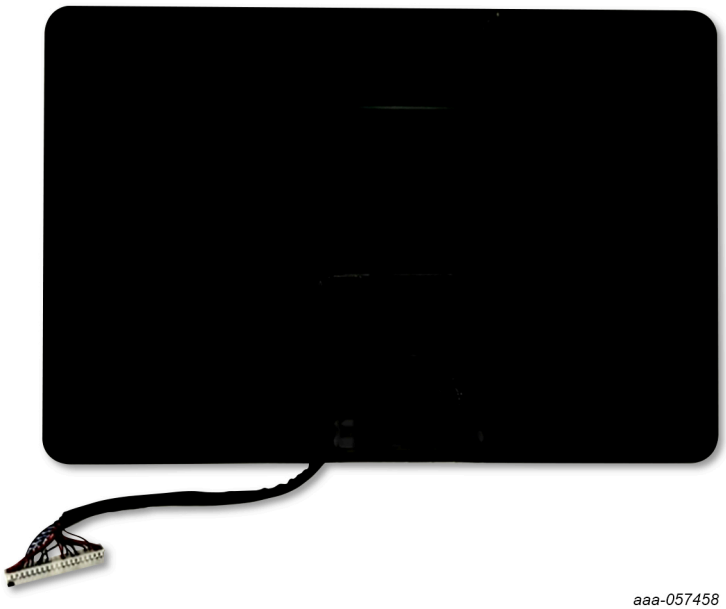
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-057482

Figure 57. i.MX LVDS panel



aaa-057458

Figure 58. i.MX BOE panel

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

© 2026 NXP B.V. All rights reserved.

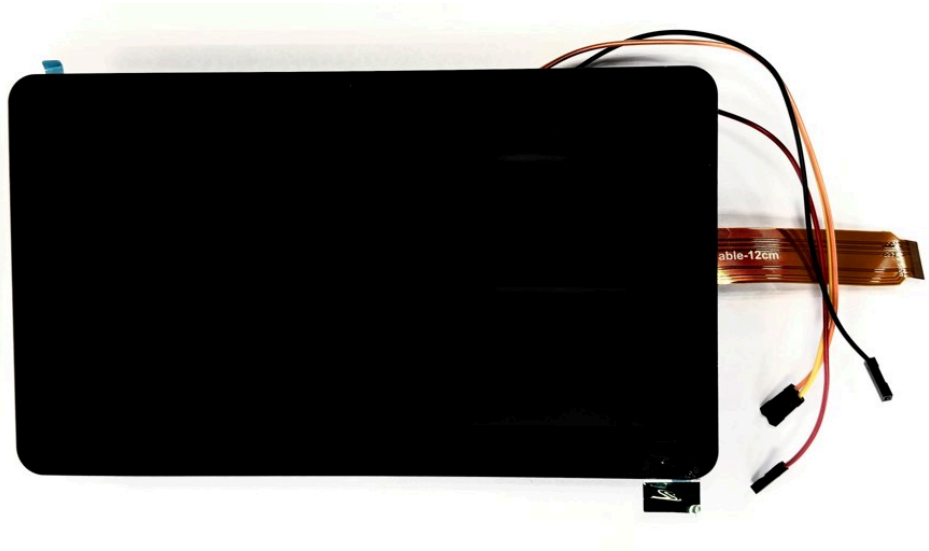


Figure 59. i.MX Waveshare panel

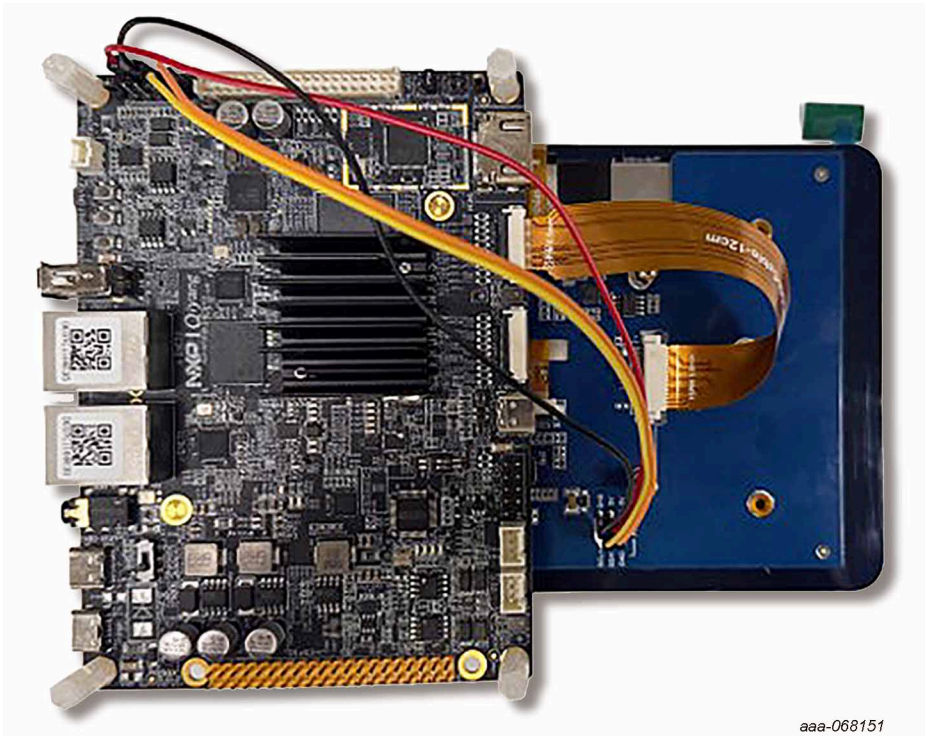
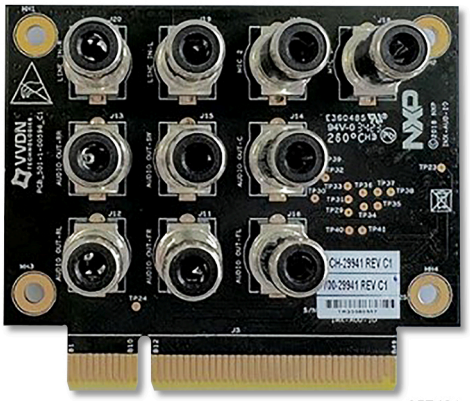


Figure 60. Hardware connection between FRDM-IMX95 and Waveshare panel

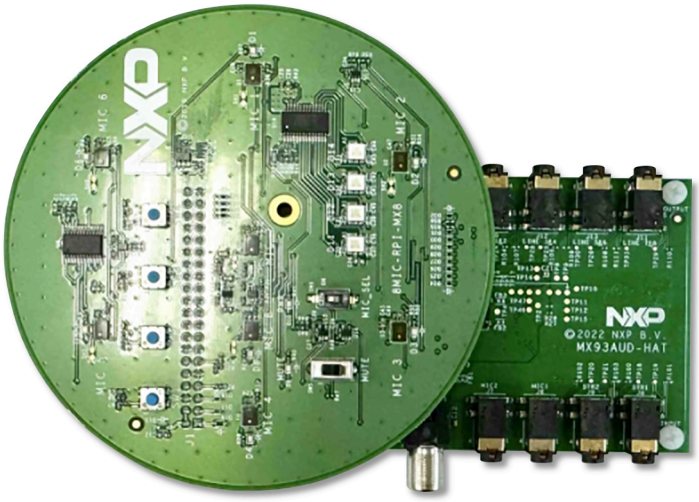
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-057481

Figure 61. AUDIO-IO board



aaa-057459

Figure 62. Audio HAT board + i.MX 8MIC PDM Microphone board



aaa-057540

Figure 63. i.MX CSI MIPI Camera OS08A20

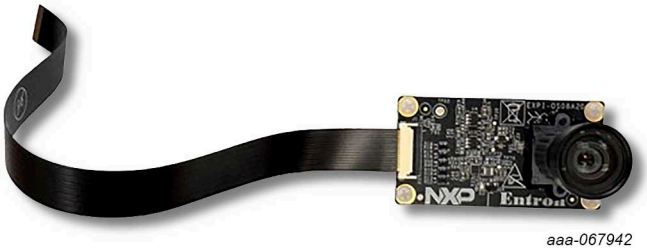


aaa-057480

Figure 64. i.MX CSI MIPI Camera AP1302 with adapter MiniSAS (EVK) -> RPi (CAM)

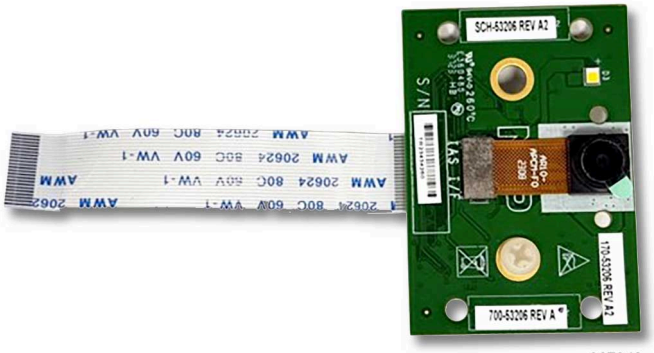
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



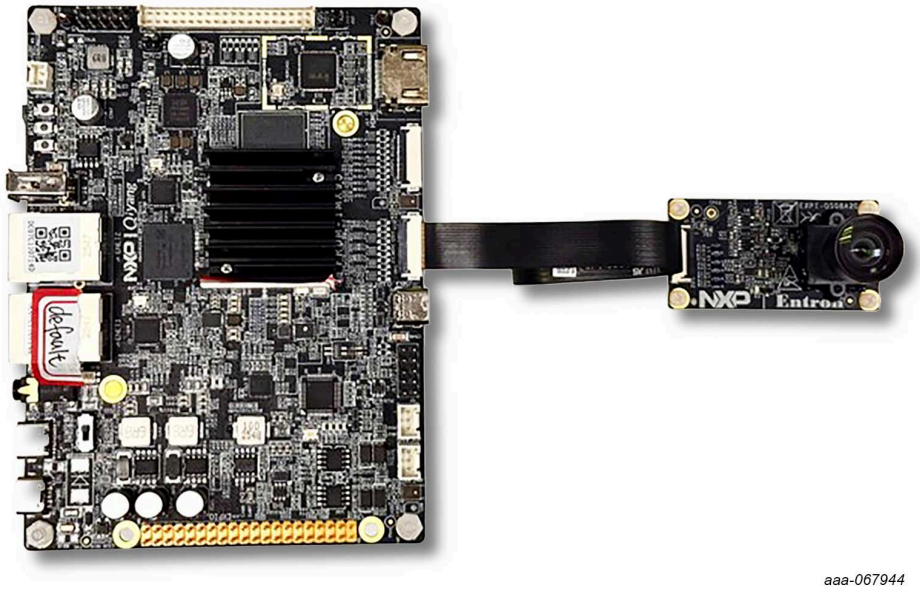
aaa-067942

Figure 65. i.MX Camera OS08A20 with 22-pin flat flex cable



aaa-067943

Figure 66. i.MX Camera AP1302 with 22-pin flat flex cable

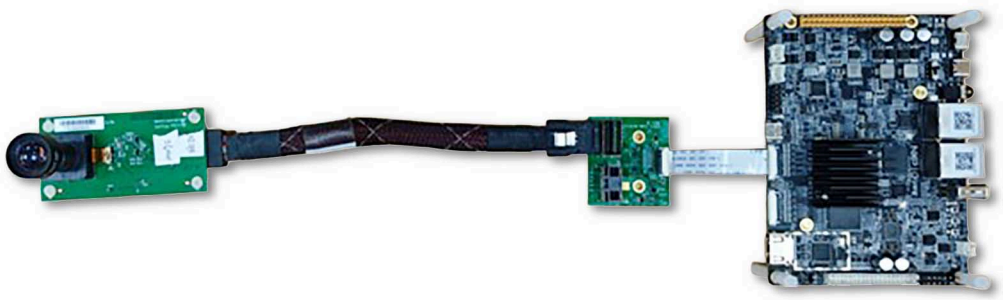


aaa-067944

Figure 67. Hardware connection between FRDM-IMX95 and OS08A20

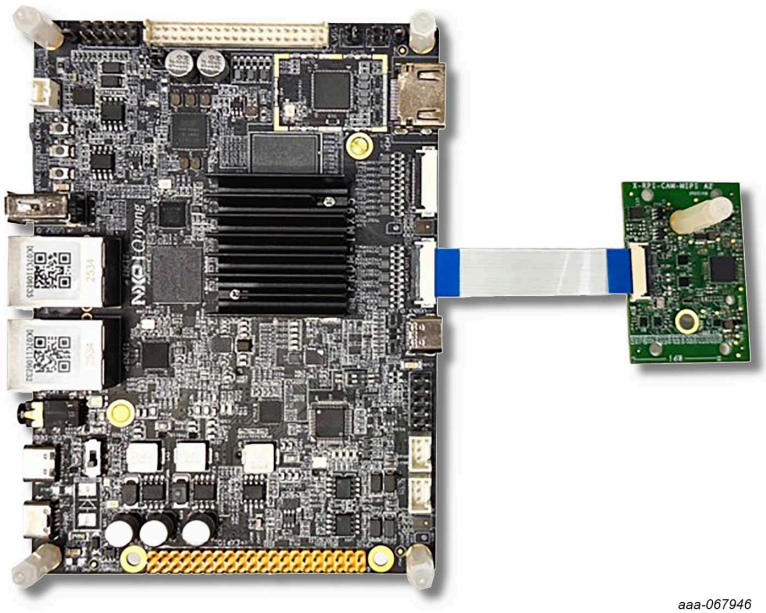
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-067945

Figure 68. Hardware connection between FRDM-IMX95 and OS08A20 with adapter RPi (EVK) -> MiniSAS (CAM)



aaa-067946

Figure 69. Hardware connection between FRDM-IMX95 and AP1302

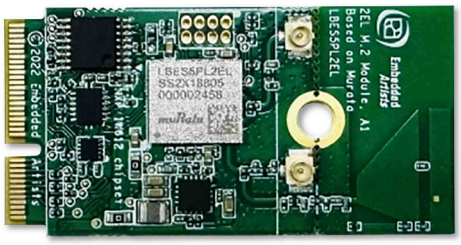


aaa-057501

Figure 70. PCIE9098 (U-Blox JODY-W377)

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

© 2026 NXP B.V. All rights reserved.



aaaa-057479

Figure 71. SDIW612 (Murata LBES5PL2EL)

Note:

- *i.MX 95 EVK, FRDM-IMX95 boards are supported in this release.*
- *To test MIPI-DSI to HDMI display on i.MX 95 EVK, connect i.MX mini SAS cable with DSI to HDMI adapter to "MIPI DSICSI" port.*
- *To test MIPI panel display on i.MX 95 EVK, connect i.MX MIPI panel RM692C9 to "MIPI DSICSI" port.*
- *To test DSI to HDMI4K display on i.MX 95 EVK, connect i.MX mini SAS cable with DSI to HDMI4K adapter to "MIPI DSICSI" port on boards.*
- *To test the LVDS-to-HDMI display on i.MX 95 EVK, use the i.MX mini SAS cable to connect the LVDS-to-HDMI adapter to the "LVDS0" port.*
- *To test the LVDS panel display on i.MX 95 EVK, use two i.MX mini SAS cables to connect the LVDS panel to the "LVDS0" and "LVDS1" ports, connect "LVDS0" to J9 of the LVDS panel, and connect "LVDS1" to J8 of the LVDS panel.*
- *To test the dual LVDS-to-HDMI displays on i.MX 95 EVK, use two i.MX mini SAS cables to connect two LVDS-to-HDMI adapters to the "LVDS0" and "LVDS1" ports.*
- *To test the MIPI-DSI-to-HDMI and LVDS-to-HDMI displays on i.MX 95 EVK, use the i.MX mini SAS cable to connect the DSI-to-HDMI adapter to the "MIPI DSI" port, and use the i.MX mini SAS cable to connect the LVDS-to-HDMI adapter to the "LVDS1" port.*
- *To test the DSI-to-SDSB display on i.MX 95 EVK, connect 1-3, 2-4 of the J16 jumper on Deserialiser, use the i.MX mini SAS cable to connect the DSI-to-SDSB (Serialiser and Deserialiser) adapter to the "MIPI DSICSI" port on boards.*
- *To test the HDMI display on FRDM-IMX95, connect the HDMI display to the "HDMI" port on FRDM-IMX95.*
- *To test the BOE panel display on FRDM-IMX95, connect the i.MX BOE panel to the "LVDS" port on FRDM-IMX95.*
- *To test the Waveshare panel display on FRDM-IMX95, connect the i.MX Waveshare panel to the "LVDS" port and J2 connector on FRDM-IMX95.*
  Note: *See Figure 60. Ensure that the pin 1 on the Waveshare connects to the 3V3 pin of the "MIPI DSICSI" port on FRDM-IMX95 through the 15-pin to 22-pin flat flex cable. Connect the Waveshare panel to the FRDM-IMX95 using the jumper wires: 5 V pin on the Waveshare to pin 2 of connector J2 on the FRDM-IMX95, connect the GND pin on the Waveshare to pin 6 of J2, connect the SCL pin to pin 7 of J2, and connect the SDA pin to pin 9 of J2.*
- *To test the camera on i.MX 95 EVK, connect the i.MX CSI MIPI camera OS08A20 or AP1302 to the "MIPI CSI" port.*
- *To test the camera on FRDM-IMX95, connect the i.MX camera OS08A20 or AP1302 with the 22-pin flat flex cable to the "MIPI CSI1" port.*
  Note:
  - *See Figure 67. Ensure that both the FRDM-IMX95 and the OS08A20 are facing upward, and connect them using a 22-pin flat flex cable.*
  - *See Figure 68. Ensure that the FRDM-IMX95, the OS08A20, and the adapter are facing upward, and use the port RPi (EVK) -> MiniSAS (CAM) on the adapter.*

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

© 2026 NXP B.V. All rights reserved.

  - *See Figure 69. Ensure that the FRDM-IMX95 is facing upward, the AP1302 is facing downward, and connect them using a 22-pin flat flex cable.*
- *To test the dual OS08A20 cameras on FRDM-IMX95, connect the i.MX camera OS08A20 with a 22-pin flat flex cable to the "MIPI CSI1" and "MIPI DSICSI" ports.*
- *To test the AUDIO-IO board on the i.MX 95 EVK board, connect the Audio HAT board and i.MX 8MIC PDM Microphone board to the J27 connector.*
- *To test the Audio HAT board and i.MX 8MIC PDM Microphone board on the FRDM-IMX95 board, connect the AUDIO-IO board to the J19 connector.*
- *Connect the PCIE9098 (U-Blox JODY-W377) Wi-Fi&BT M.2 expansion card to the J24 connector, and set the SW1 Bluetooth switch to 1111 (from 1-4 bit) to enable the Wi-Fi and Bluetooth functionality on the i.MX 95 EVK board.*
- *Connect the SDIW612 (Murata LBES5PL2EL) Wi-Fi&BT M.2 expansion card to the J7 connector to enable the Wi-Fi and Bluetooth functionality on the FRDM-IMX95 board.*

10.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_95evk.tar.gz`.

Table 15. Board images

Image name	Download target	Description
`spl-imx95-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the i.MX 95 EVK board.
`spl-imx95-trusty-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty related configurations for the i.MX 95 EVK board.
`spl-imx95-trusty-rbidx-blob-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 95 EVK board.
`spl-imx95-trusty-secure-unlock-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 95 EVK board.
`spl-imx95-b1-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the i.MX 95 EVK board with the B2.1 chip.
`spl-imx95-trusty-b1-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty related configurations for the i.MX 95 EVK board with the B2.1 chip.
`spl-imx95-trusty-secure-unlock-b1-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 95 EVK board with the B2.1 chip.
`spl-imx95-15x15-frdm-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the FRDM-IMX95 board.
`spl-imx95-trusty-15x15-frdm-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the FRDM-IMX95 board.
`spl-imx95-15x15-frdm-b1-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the FRDM-IMX95 board with the B2.1 chip.
`spl-imx95-trusty-15x15-frdm-b1-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the FRDM-IMX95 board with the B2.1 chip.
`spl-imx95-15x15-frdm-lp5-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the FRDM-IMX95 board with LPDDR5 DRAM.

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

© 2026 NXP B.V. All rights reserved.

Table 15. Board images...*continued*

Image name	Download target	Description
spl-imx95-trusty-15x15-frdm-lp5-dual.bin	bootloader0 partition.	Secondary program loader image with Trusty and secure unlock related configurations for the FRDM-IMX95 board with LPDDR5 DRAM.
bootloader-imx95-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 95 EVK board.
bootloader-imx95-trusty-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 95 EVK board.
bootloader-imx95-trusty-rbidx-blob-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 95 EVK board.
bootloader-imx95-trusty-secure-unlock-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 95 EVK board.
bootloader-imx95-b1-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 95 EVK board with the B2.1 chip.
bootloader-imx95-trusty-b1-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 95 EVK board with the B2.1 chip.
bootloader-imx95-trusty-secure-unlock-b1-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is a demonstration of the secure unlock mechanism for the i.MX 95 EVK board with the B2.1 chip.
bootloader-imx95-15x15-frdm-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the FRDM-IMX95 board.
bootloader-imx95-trusty-15x15-frdm-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the FRDM-IMX95 board.
bootloader-imx95-15x15-frdm-b1-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the FRDM-IMX95 board with the B2.1 chip.
bootloader-imx95-trusty-15x15-frdm-b1-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the FRDM-IMX95 board with the B2.1 chip.
bootloader-imx95-15x15-frdm-lp5-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper and ATF. It is for the FRDM-IMX95 board with LPDDR5 DRAM.
bootloader-imx95-trusty-15x15-frdm-lp5-dual.img	bootloader_a and bootloader_b partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the FRDM-IMX95 board with LPDDR5 DRAM.
u-boot-imx95.imx	bootloader0 partition.	An image containing U-Boot and ATF for the i.MX 95 EVK board.
u-boot-imx95-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 95 EVK board. It is not flashed to MMC.
u-boot-imx95-rpmsg.imx	bootloader0 partition.	An image containing U-Boot and ATF to support RPMSG on the i.MX 95 EVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 15. Board images...*continued*

Image name	Download target	Description
u-boot-imx95-sof.imx	bootloader0 partition.	An image containing U-Boot and ATF to support Sound Open Firmware (SOF) on the i.MX 95 EVK board.
u-boot-imx95-b1.imx	bootloader0 partition.	An image containing U-Boot and ATF for the i.MX 95 EVK board with the B2.1 chip.
u-boot-imx95-evk-b1-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 95 EVK board with the B2.1 chip. It is not flashed to MMC.
u-boot-imx95-rpmsg-b1.imx	bootloader0 partition.	An image containing U-Boot and ATF to support rpmsg on the i.MX 95 EVK board with B2.1 chip.
u-boot-imx95-sof-b1.imx	bootloader0 partition.	An image containing U-Boot and ATF to support SOF on the i.MX 95 EVK board with B2.1 chip.
u-boot-imx95-15x15-frdm.imx	bootloader0 partition.	An image containing U-Boot and ATF for the FRDM-IMX95 board.
u-boot-imx95-15x15-frdm-uuu.imx	bootloader0 partition.	An image containing U-Boot and ATF, used by UUU for the FRDM-IMX95 board. It is not flashed to MMC.
u-boot-imx95-15x15-frdm-b1.imx	bootloader0 partition.	An image containing U-Boot and ATF for the FRDM-IMX95 board with the B2.1 chip.
u-boot-imx95-15x15-frdm-b1-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the FRDM-IMX95 board with the B2.1 chip. It is not flashed to MMC.
u-boot-imx95-15x15-frdm-lp5.imx	bootloader0 partition.	An image containing U-Boot and ATF for the FRDM-IMX95 board with LPDDR5 DRAM.
u-boot-imx95-15x15-frdm-lp5-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the FRDM-IMX95 board with LPDDR5 DRAM. It is not flashed to MMC.
partition-table.img	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-13GB.img	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-13GB-dual.img	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
boot.img	boot_a and boot_b partitions.	AOSP GKI boot image. It contains the AOSP generic kernel image and generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image built with the i.MX kernel tree for debugging.
init_boot.img	init_boot_a and init_boot_b partitions.	Contains generic ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor RamDisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in the VTS test with the GSI system image.
vbmeta-imx95.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and the OS08A20 camera on the i.MX 95 EVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 15. Board images...*continued*

Image name	Download target	Description
vbmeta-imx95-ap1302.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and the AP1302 camera on the i.MX 95 EVK board.
vbmeta-imx95-mipi-lvds1.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI and LVDS-to-HDMI multiple displays, and the AP1302 camera on the i.MX 95 EVK board.
vbmeta-imx95-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the RM692C9 MIPI panel display on the i.MX 95 EVK board.
vbmeta-imx95-lvds0.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI display and the OS08A20 camera on the i.MX 95 EVK board.
vbmeta-imx95-lvds0-dual-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI display and dual OS08A20 cameras on the i.MX 95 EVK board.
vbmeta-imx95-lvds-dualdisp.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the dual LVDS-to-HDMI displays and dual OS08A20 cameras on the i.MX 95 EVK board.
vbmeta-imx95-lvds-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel display on the i.MX 95 EVK board.
vbmeta-imx95-cs42888.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display, AUDIO-IO board, and AP1302 camera on the i.MX 95 EVK board.
vbmeta-imx95-rpmsg.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and MCU image on the i.MX 95 EVK board.
vbmeta-imx95-sof.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support MIPI-to-HDMI display and SOF on the i.MX 95 EVK board.
vbmeta-imx95-mipi4k.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI4 K display and the AP1302 camera on the i.MX 95 EVK board.
vbmeta-imx95-dsi-serdes.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the DSI-to-SDSB display on the i.MX 95 EVK board.
vbmeta-imx95-15x15-frdm.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and the OS08A20 camera on the FRDM-IMX95 board.
vbmeta-imx95-15x15-frdm-ap1302.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and the AP1302 camera on the FRDM-IMX95 board.
vbmeta-imx95-15x15-frdm-aud-hat.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and audio HAT on the FRDM-IMX95 board.
vbmeta-imx95-15x15-frdm-boe-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the BOE panel display on the FRDM-IMX95 board.
vbmeta-imx95-15x15-frdm-dual-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and dual OS08A20 cameras on the FRDM-IMX95 board.
vbmeta-imx95-15x15-frdm-waveshare-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Waveshare panel display on the FRDM-IMX95 board.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_ext.img	Logical partitions system_ext_a and	System extension image.

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

© 2026 NXP B.V. All rights reserved.

Table 15. Board images...*continued*

Image name	Download target	Description
	system_ext_b in super partition.	
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx95.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and the OS08A20 camera on the i.MX 95 EVK board.
dtbo-imx95-ap1302.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and the AP1302 camera on the i.MX 95 EVK board.
dtbo-imx95-mipi-lvds1.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI and LVDS-to-HDMI multiple displays, and the AP1302 camera on the i.MX 95 EVK board.
dtbo-imx95-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the RM692C9 MIPI panel display on the i.MX 95 EVK board.
dtbo-imx95-lvds0.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI display and the OS08A20 camera on the i.MX 95 EVK board.
dtbo-imx95-lvds0-dual-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support LVDS-to-HDMI display and dual OS08A20 cameras on the i.MX 95 EVK board.
dtbo-imx95-lvds-dualdisp.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the dual-channel LVDS-to-HDMI display and dual OS08A20 cameras on the i.MX 95 EVK board.
dtbo-imx95-lvds-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel display on the i.MX 95 EVK board.
dtbo-imx95-cs42888.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display, AUDIO-IO board, and AP1302 camera on the i.MX 95 EVK board.
dtbo-imx95-rpmsg.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and MCU image on the i.MX 95 EVK board.
dtbo-imx95-sof.img	dtbo_a and dtbo_b partitions.	Device Tree image to support MIPI-to-HDMI display and SOF on the i.MX 95 EVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 15. Board images...*continued*

Image name	Download target	Description
dtbo-imx95-mipi4k.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI4K display and the AP1302 camera on the i.MX 95 EVK board.
dtbo-imx95-dsi-serdes.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the DSI-to-SDSB display on the i.MX 95 EVK board.
dtbo-imx95-15x15-frdm.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and the OS08A20 camera on the FRDM-IMX95 board.
dtbo-imx95-15x15-frdm-ap1302.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and the AP1302 camera on the FRDM-IMX95 board.
dtbo-imx95-15x15-frdm-aud-hat.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and audio HAT on the FRDM-IMX95 board.
dtbo-imx95-15x15-frdm-boe-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the BOE panel display on the FRDM-IMX95 board.
dtbo-imx95-15x15-frdm-dual-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and dual OS08A20 cameras on the FRDM-IMX95 board.
dtbo-imx95-15x15-frdm-waveshare-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Waveshare panel display on the FRDM-IMX95 board.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key. It is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.
android-17.0.0_1.0.0_image_trout.tar.gz	N/A	Trout Android image, which runs as an i.MX 95 Xen Guest. For more details, see Section "Building images" and Section "Installing the Trout image" in the *Android User's Guide* (UG10156).

10.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to flash all Android images automatically.

- uuu_imx_android_flash.sh for Linux OS
- uuu_imx_android_flash.bat for Windows OS

For this release, these two scripts are validated on UUU 1.5.201 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named uuu.
- For Windows OS, download the file named uuu.exe.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

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

© 2026 NXP B.V. All rights reserved.

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   a. For i.MX 95 EVK:
      Change the board's SW7 to 1001 (from 1-4 bit) to enter serial download mode.
   b. For FRDM-IMX95:
      Change the board's SW1 to 01 (from 1-2 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board Type-C port to connect your PC with the board.
   Note: *The debug PORT can be used to watch the logs of the hardware boot processing.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_95evk.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For the i.MX 95 EVK board, related options are described as follows.

Table 16. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies the SoC information. For i.MX 95 EVK, it should be `imx95`. This option is **mandatory**.
`-a`	Only flashes slot a. If this option and `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b. If this option and `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 95 EVK and FRDM-IMX95, it can be followed by "13". If this option is not used, the default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If it is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. • For i.MX 95 EVK, it can be `dual`, `trusty-dual`, `trusty-rbidx-blob-dual`, `trusty-secure-unlock-dual`, or `rpmsg`. • For i.MX 95 EVK with the B2.1 chip, it can be `b1`, `b1-dual`, `trusty-b1-dual`, `trusty-secure-unlock-b1-dual`, `rpmsg-b1`, or `sof-b1`. • For FRDM-IMX95, it can be `15x15-frdm`, `15x15-frdm-dual`, or `trusty-15x15-frdm-dual`. • For FRDM-IMX95 board with the B2.1 chip, it can be `15x15-frdm-b1`, `15x15-frdm-b1-dual`, or `trusty-15x15-frdm-b1-dual`. • For FRDM-IMX95 board with LPDDR5 DRAM, it can be `15x15-frdm-lp5`, `15x15-frdm-lp5-dual`, or `trusty-15x15-frdm-lp5-dual`.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. • For i.MX 95 EVK, it can be `ap1302`, `mipi-lvds1`, `mipi-panel`, `lvds0`, `lvds-dualdisp`, `lvds-panel`, `cs42888`, `rpmsg`, `mipi4k`, or `dsi-serdes`. • For FRDM-IMX95, it can be `15x15-frdm`, `15x15-frdm-ap1302`, `15x15-frdm-aud-hat`, `15x15-frdm-boe-panel`, `15x15-frdm-dual-os08a20`, or `15x15-frdm-waveshare-panel`. If it is not used, the default `dtbo-imx95.img` and `vbmeta-imx95.img` are flashed.
`-e`	Erases user data after images are flashed.

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

© 2026 NXP B.V. All rights reserved.

Table 16. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 95 EVK, FRDM-IMX95, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

Note: *`-m` should be used together with `-d rpmsg` and `-u rpmsg` for i.MX 95 EVK, or `-m` should be used together with `-d rpmsg` and `-u rpmsg-b1` for i.MX 95 EVK with the B2.1 chip.*

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx95 -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> .\uuu_imx_android_flash.bat -f imx95 -a -e -u trusty-dual
```

When the command above is executed, `spl-imx95-trusty-dual.bin` is flashed, and `bootloader-imx95-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 95 EVK.

Note:

- *`-u` followed with a parameter and containing `trusty` cannot be used together with `-t sd`, because Trusty OS cannot boot from SD card.*
- *To flash the SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test the features on the i.MX 95 EVK board:*
  - *To test dual-bootloader, execute the tool with `-u dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition for i.MX 95 EVK, execute the tool with `-u trusty-dual`.*
  - *To test Trusty OS, dual-bootloader, and encrypted rollback index enabled condition on i.MX 95 EVK, execute the tool with `-u trusty-rbidx-blob-dual`.*
  - *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User Guide (UG10158).*
  - *To test dual-bootloader for i.MX 95 EVK with the B2.1 chip, execute the tool with `-u b1-dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition for i.MX 95 EVK with the B2.1 chip, execute the tool with `-u trusty-b1-dual`.*
  - *To test the MIPI-to-HDMI display, it does not need to use the `-d` option.*
  - *To test the MIPI-to-HDMI and LVDS-to-HDMI dual displays, execute the tool with `-d mipi-lvds1`.*
  - *To test the MIPI panel (RM692C9) display, execute the tool with `-d mipi-panel`.*
  - *To test the LVDS-to-HDMI display, execute the tool with `-d lvds0`.*

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

© 2026 NXP B.V. All rights reserved.

  - *To test the dual-channel LVDS-to-HDMI display, execute the tool with `-d lvds-dualdisp`.*
  - *To test the LVDS panel display, execute the tool with `-d lvds-panel`.*
  - *To test the MIPI-to-HDMI4K display, execute the tool with `-d mipi4k`.*
  - *To test the DSI-to-SDSB display, execute the tool with `-d dsi-serdes`.*
  - *To test the OS08A20 camera, it does not need to use the `-d` option, or execute the tool with `-d lvds0`.*
  - *To test the AP1302 camera, execute the tool with `-d ap1302`, `-d mipi-lvds1`, `-d cs42888`, or `-d mipi4k`.*
  - *To test dual OS08A20 cameras, execute the tool with `-d lvds0-dual-os08a20` or `-d lvds-dualdisp`.*
  - *To test the AUDIO-IO board, execute the tool with `-d cs42888` and use "MIPI DSI" for the MIPI-to-HDMI display.*
  - *To test the MIPI-to-HDMI display and MCU image, execute the tool with `-u rpmsg -d rpmsg -m`.*
  - *To test the MIPI-to-HDMI display and Sound Open Firmware (SOF) for i.MX 95 EVK, execute the tool with `-u sof -d sof`.*
  - *To test the MIPI-to-HDMI display and SOF for i.MX 95 EVK with the B2.1 chip, execute the tool with `-u sof-b1 -d sof`.*
- *To test the features on the FRDM-IMX95:*
  - *To test dual-bootloader for FRDM-IMX95, execute the tool with `-u 15x15-frdm-dual`.*
  - *To test the Trusty OS and dual-bootloader both enabled condition for FRDM-IMX95, execute the tool with `-u trusty-15x15-frdm-dual`.*
  - *To test dual-bootloader for FRDM-IMX95 with LPDDR5 DRAM, execute the tool with `-u 15x15-frdm-lp5-dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition for FRDM-IMX95 with LPDDR5 DRAM, execute the tool with `-u trusty-15x15-frdm-lp5-dual`.*
  - *To test the HDMI display, execute the tool with `-d 15x15-frdm`, `-d 15x15-frdm-ap1302`, `-d 15x15-frdm-aud-hat`, or `-d 15x15-frdm-dual-os08a20`.*
  - *To test the BOE panel display, execute the tool with `-d 15x15-frdm-boe-panel`.*
  - *To test the Waveshare panel display, execute the tool with `-d 15x15-frdm-waveshare-panel`.*
  - *To test the OS08A20 camera, execute the tool with `-d 15x15-frdm`.*
  - *To test the dual OS08A20 cameras, execute the tool with `-d 15x15-frdm-dual-os08a20`.*
  - *To test the AP1302 camera, execute the tool with `-d 15x15-frdm-ap1302`.*
  - *To test the Audio HAT with 8MIC PDM Microphone, execute the tool with `-d 15x15-frdm-aud-hat`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

`z` in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is not any error, the command window displays information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change boot device to eMMC or SD card.
   - For i.MX 95 EVK:
     - Change SW7 to switch the board back to 1010 (form 1-4 bit) to enter eMMC boot mode.
     - Change SW7 to switch the board back to 1011 (form 1-4 bit) to enter SD boot mode.
   - For FRDM-IMX95:
     - Change SW1 to switch the board back to 10 (form 1-2 bit) to enter eMMC boot mode.

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

© 2026 NXP B.V. All rights reserved.

– Change SW1 to switch the board back to 11 (form 1-2 bit) to enter SD boot mode.

10.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

10.4.1 Booting with an MCU image

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.lpa.enable=1
U-Boot > setenv append_bootargs snd_pcm.max_alloc_per_card=134217728
 pd_ignore_unused cma=600M
U-Boot > setenv bootcmd "bootmcu && boota"
U-Boot > saveenv
```

Note:

To use other dtbo images, do not add `bootmcu" to "bootcmd`. The following command can recover `bootcmd`:

```
U-Boot > setenv bootcmd "boota"
U-Boot > saveenv
```

10.4.2 Booting with a display: HDMI 4K display

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.displaymode=4kp30
 androidboot.lcd_density=480 androidboot.dpu_composition=0
U-Boot > saveenv
```

Note:

If other boot arguments need to be appended to the U-Boot environment variable `bootargs`, set them together in one `setenv append_bootargs` command.

10.4.3 Booting with multi-channel audio support

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.audio.multichannel=1
U-Boot > saveenv
```

10.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

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

© 2026 NXP B.V. All rights reserved.

11 Working with the i.MX 952 EVK and FRDM-IMX952 boards

11.1 Board hardware

The figures below show the different components of the i.MX 952 19x19 EVK and FRDM-IMX952 boards.

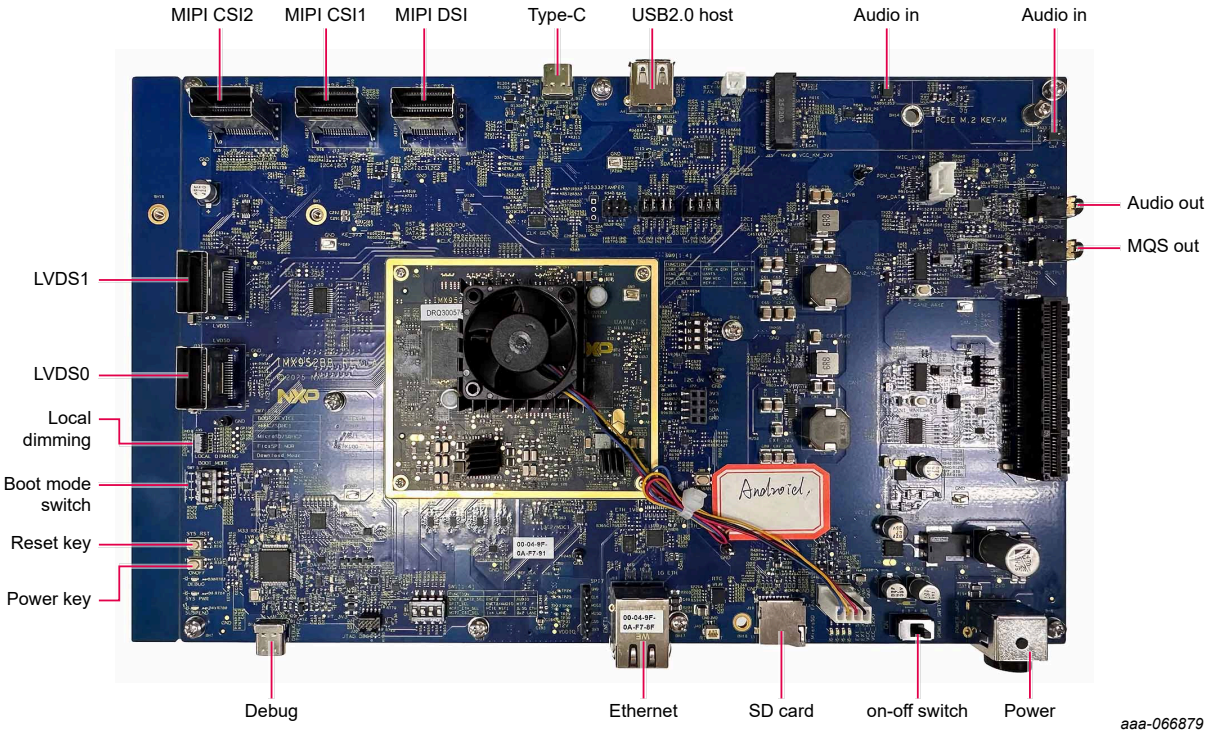


Figure 72. i.MX 952 19x19 EVK



Figure 73. i.MX 952 19x19 EVK back view

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

© 2026 NXP B.V. All rights reserved.

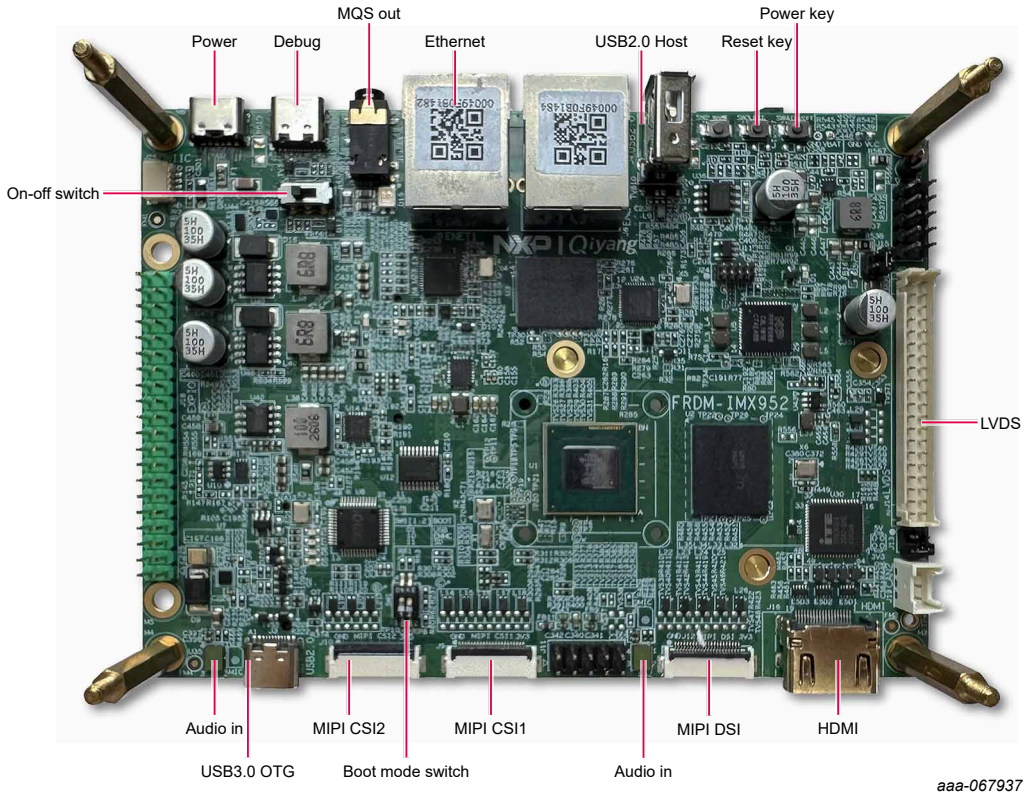


Figure 74. FRDM-IMX952

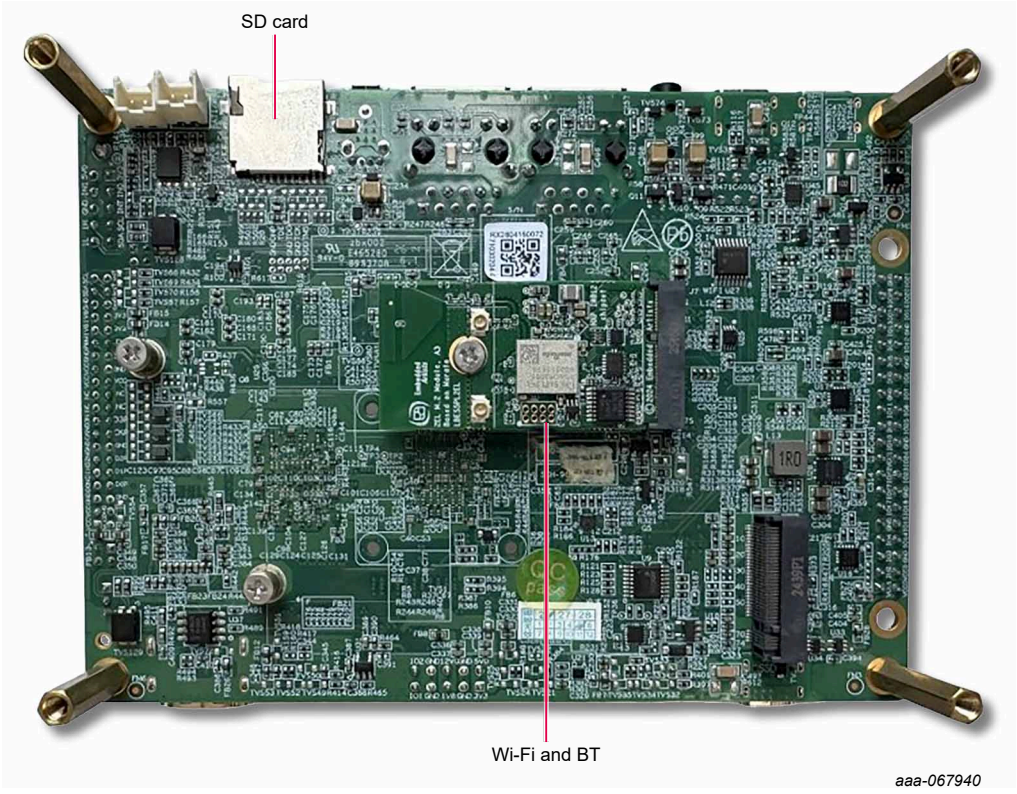


Figure 75. FRDM-IMX952 back view

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

© 2026 NXP B.V. All rights reserved.



aaa-057489

Figure 76. i.MX mini SAS cable with DSI-to-HDMI adapter



aaa-057541

Figure 77. i.MX mini SAS cable with LVDS-to-HDMI adapter



aaa-057457

Figure 78. i.MX mini SAS cable with DSI-to-HDMI4K adapter



aaa-057483

Figure 79. i.MX MIPI panel RM692C9



aaa-057482

Figure 80. i.MX LVDS panel

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

© 2026 NXP B.V. All rights reserved.

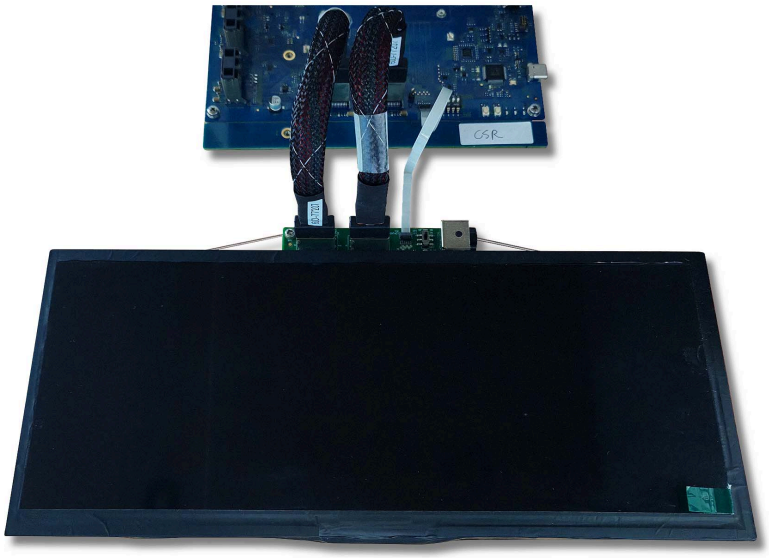


Figure 81. i.MX LVDS panel with local dimming

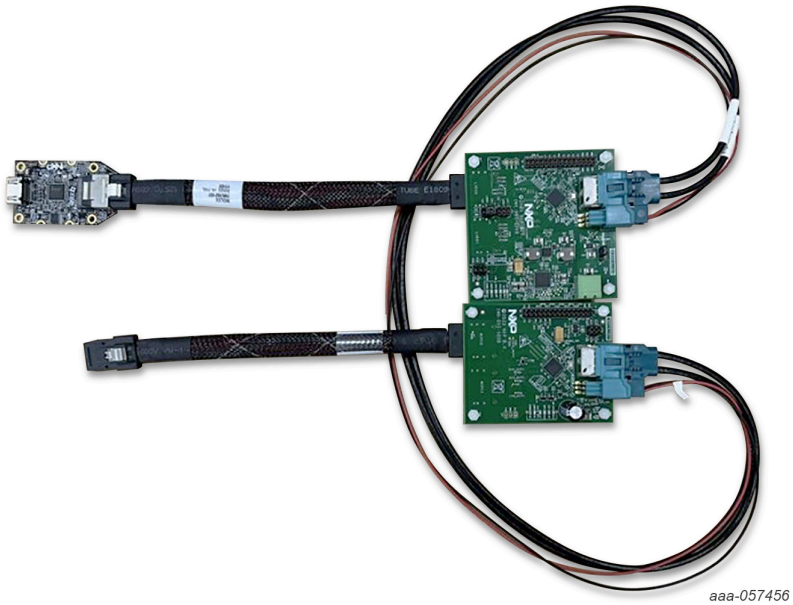


Figure 82. i.MX mini SAS cable with DSI-to-SDSB (Serialiser and Deserialiser) adapter

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

© 2026 NXP B.V. All rights reserved.

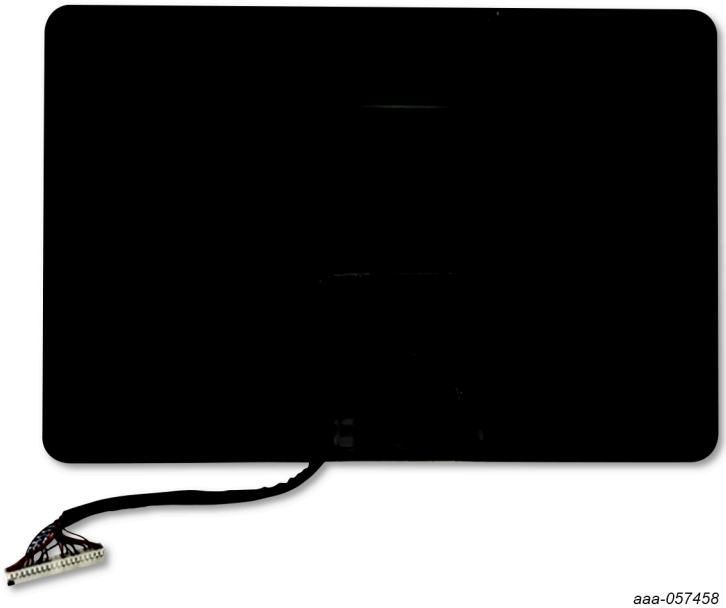


Figure 83. i.MX BOE panel

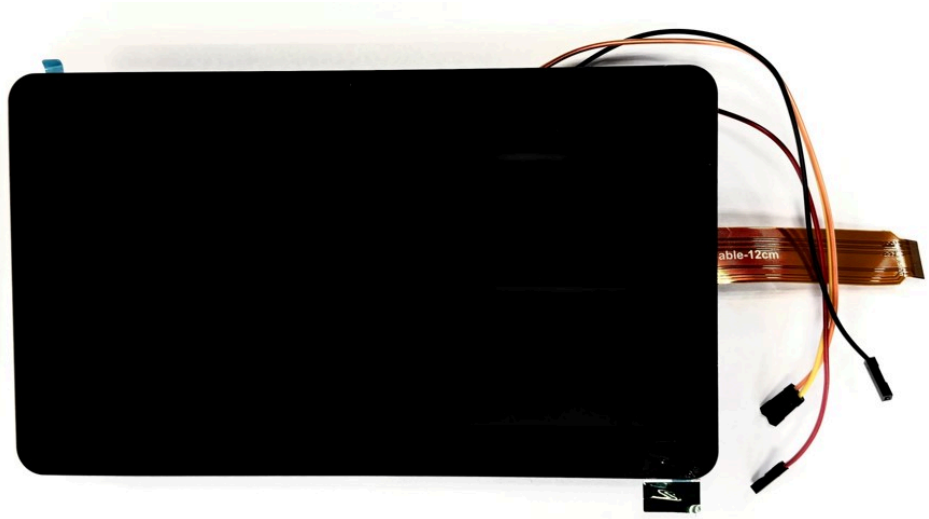


Figure 84. i.MX Waveshare panel

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

© 2026 NXP B.V. All rights reserved.

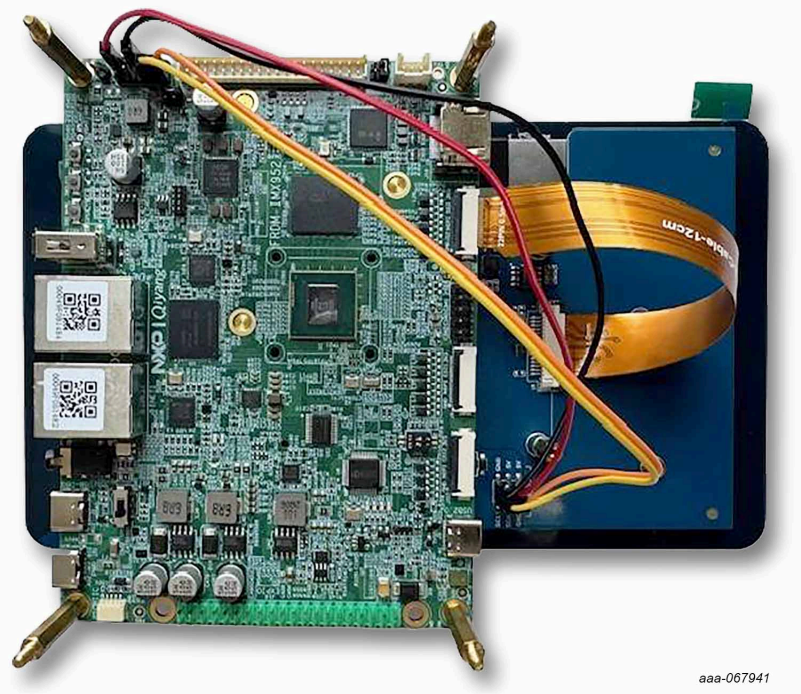


Figure 85. Hardware connection between FRDM-IMX952 and Waveshare panel



Figure 86. i.MX CSI MIPI Camera AP1302



Figure 87. i.MX CSI MIPI Camera OS08A20

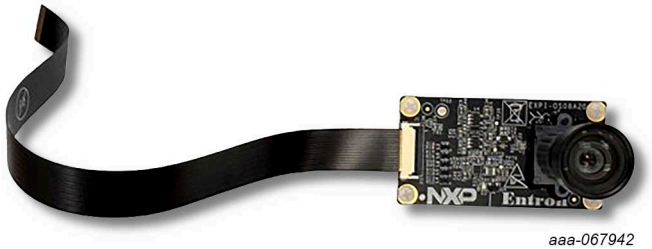


Figure 88. i.MX Camera OS08A20 with 22-pin flat flex cable

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

© 2026 NXP B.V. All rights reserved.

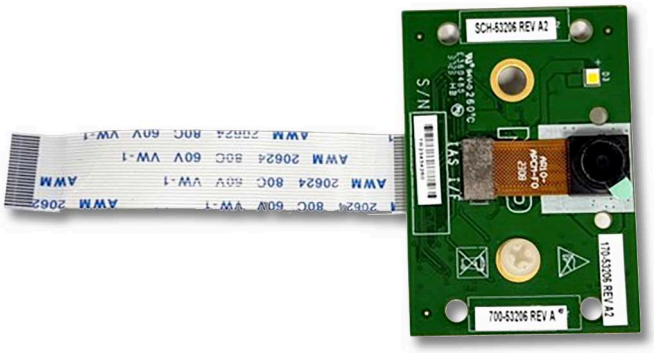


Figure 89. i.MX Camera AP1302 with 22-pin flat flex cable

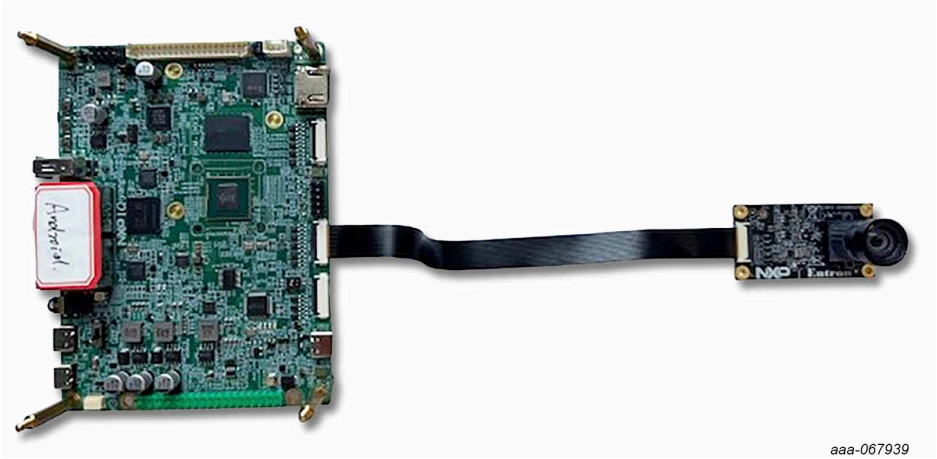


Figure 90. Hardware connection between FRDM-IMX952 and OS08A20

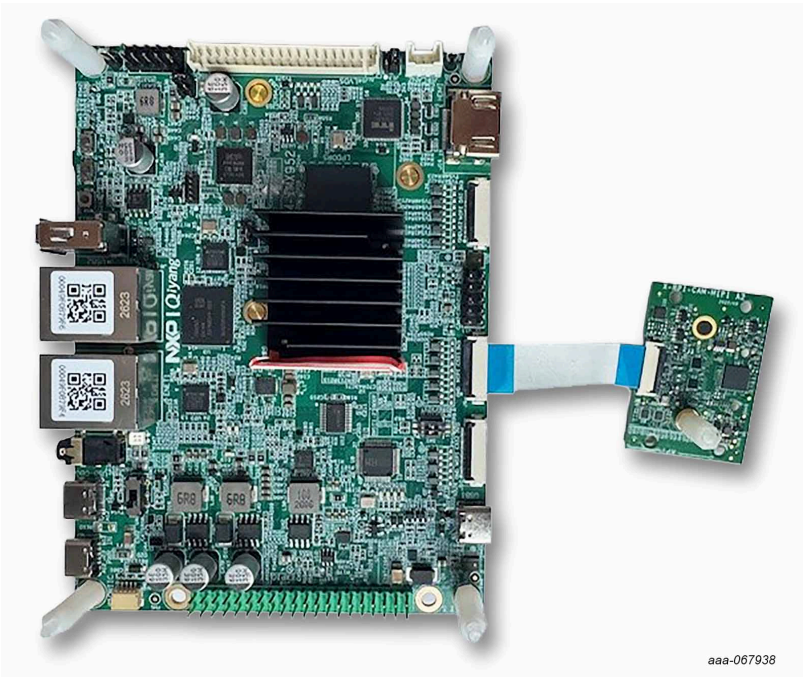
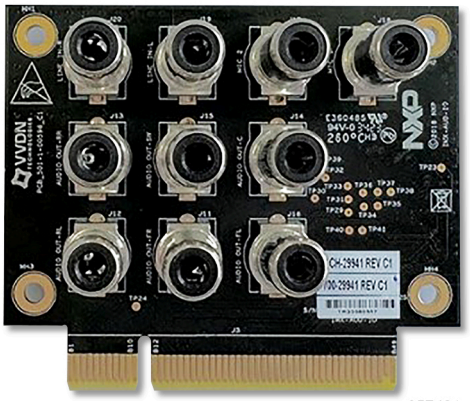


Figure 91. Hardware connection between FRDM-IMX952 and AP1302

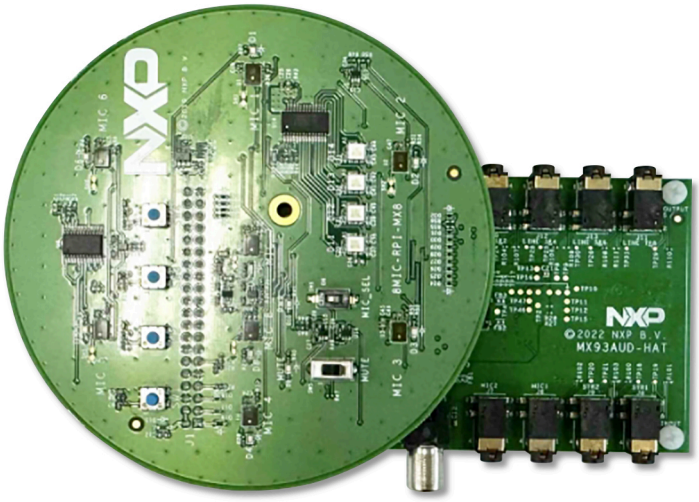
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-057481

Figure 92. AUDIO-IO board



aaa-057459

Figure 93. Audio HAT board + i.MX 8MIC PDM Microphone board

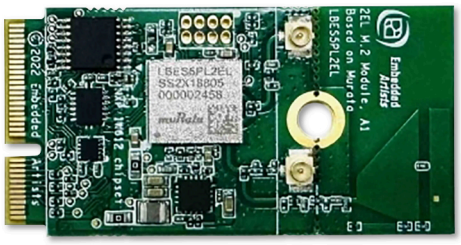


aaa-066882

Figure 94. AW693 (U-Blox JODY-W683-21A)

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

© 2026 NXP B.V. All rights reserved.



aaaa-057479

Figure 95. SDIW612 (Murata LBES5PL2EL)

Note:

- *The i.MX 952 19x19 EVK and FRDM-IMX952 boards are supported in this release.*
  - *The i.MX 952 19x19 EVK is labeled IMX952LPD5CPU-19.*
- *To test on i.MX 952 19x19 EVK:*
  - *To test the MIPI-DSI to HDMI display, connect the i.MX mini SAS cable with the DSI-to-HDMI adapter to the "MIPI DSI" port.*
  - *To test the MIPI panel display, connect the i.MX MIPI panel RM692C9 to the "MIPI DSI" port.*
  - *To test the DSI to the HDMI4K display, connect the i.MX mini SAS cable with the DSI-to-HDMI4K adapter to the "MIPI DSI" port on boards.*
  - *To test the LVDS-to-HDMI display, connect the i.MX mini SAS cable with the LVDS-to-HDMI adapter to the "LVDS0" port.*
  - *To test the dual LVDS-to-HDMI displays, use two i.MX mini SAS cables to connect two LVDS-to-HDMI adapters to the "LVDS0" and "LVDS1" ports.*
  - *To test the LVDS panel display, use two i.MX mini SAS cables to connect the LVDS panel to the "LVDS0" and "LVDS1" ports, connect "LVDS0" to J9 of the LVDS panel, and connect "LVDS1" to J8 of the LVDS panel.*
  - *To test the LVDS panel with a local dimming display, use two i.MX mini SAS cables to connect the LVDS panel to the "LVDS0" and "LVDS1" ports, connect "LVDS0" to "LVDS0 (J5)" on the LVDS panel with local dimming, connect "LVDS1" to "LVDS1 (J8)" on the LVDS panel with local dimming, and connect the "local dimming" port to "TOP CONTACT (J11)" on the LVDS panel with local dimming.*
  - *To test the DSI-to-SDSB display, connect 1-3 and 2-4 of the J16 jumper on Deserialiser, connect the i.MX mini SAS cable with the DSI-to-SDSB (Serialiser and Deserialiser) adapter to the "MIPI DSI" port on boards.*
  - *To test the single OS08A20 camera, connect the i.MX CSI MIPI Camera OS08A20 to the "MIPI CSI1" port.*
  - *To test the dual OS08A20 cameras, connect the i.MX CSI MIPI Camera OS08A20 to the "MIPI CSI1" port and "MIPI CSI2" port.*
  - *To test the AP1302 camera, connect the i.MX CSI MIPI Camera AP1302 to the "MIPI CSI1" port.*
  - *To test the audio playback on i.MX 952 EVK, connect the AUDIO-IO board to Audio IO SLOT (J27) on the baseboard.*
  - *To test the Wi-Fi/Bluetooth AW693 on i.MX 952 EVK, insert the Wi-Fi/Bluetooth module AW693 (JODY-W683-21A) into the J24 M.2 slot.*
- *To test on FRDM-IMX952:*
  - *To test the BOE panel display on FRDM-IMX952, connect the i.MX BOE panel to the "LVDS" port on FRDM-IMX952.*
  - *To test the Waveshare panel display on FRDM-IMX952, connect the i.MX Waveshare panel to the "LVDS" port and J2 connector on FRDM-IMX952.*
    Note: *See Figure 85. Ensure that Pin 1 on the Waveshare connects to the 3V3 pin of the "MIPI DSI" port on FRDM-IMX952 through the 15-pin to 22-pin flat flex cable. Connect the Waveshare panel to FRDM-IMX952 using the jumper wires: 5 V pin on the Waveshare to Pin 2 of connector J2 on FRDM-IMX952,*

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

© 2026 NXP B.V. All rights reserved.

connect the GND pin on the Waveshare to Pin 6 of J2, connect the SCL pin to Pin 7 of J2, and connect the SDA pin to Pin 9 of J2.

- *To test the camera on FRDM-IMX952, connect i.MX camera OS08A20 or AP1302 with the 22-pin flat flex cable to the "MIPI CSI1" port.*
  Note:
  - *See Figure 90. Ensure that both FRDM-IMX952 and OS08A20 are facing upward, and connect them using a 22-pin flat flex cable.*
  - *See Figure 91. Ensure that FRDM-IMX952 is facing upward and AP1302 is facing downward. Connect them using a 22-pin flat flex cable.*
- *To test the dual OS08A20 cameras on FRDM-IMX952, connect i.MX camera OS08A20 with the 22-pin flat flex cable to the "MIPI CSI1" and "MIPI CSI2" ports.*
- *To test the Audio HAT board + i.MX 8MIC PDM Microphone board on the FRDM-IMX952 board, connect the Audio HAT board to the J18 connector.*
- *To test the Wi-Fi/Bluetooth SDIW612 on FRDM-IMX952, insert the Wi-Fi/Bluetooth module SDIW612 (Murata LBES5PL2EL) into the J7 M.2 slot.*

11.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_952evk.tar.gz`.

Table 17. Board images

Image name	Download target	Description
`spl-imx952-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the i.MX 952 19x19 EVK board.
`spl-imx952-trusty-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty related configurations for the i.MX 952 19x19 EVK board.
`spl-imx952-trusty-rbidx-blob-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the i.MX 952 19x19 EVK board.
`spl-imx952-trusty-secure-unlock-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the i.MX 952 19x19 EVK board.
`spl-imx952-trusty-frdm-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty related configurations for the FRDM-IMX952 board.
`bootloader-imx952-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the i.MX 952 19x19 EVK board.
`bootloader-imx952-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the i.MX 952 19x19 EVK board.
`bootloader-imx952-trusty-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the i.MX 952 19x19 EVK board.
`bootloader-imx952-trusty-secure-unlock-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and secure unlock enabled. It is for the i.MX 952 19x19 EVK board.
`bootloader-imx952-trusty-frdm-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the FRDM-IMX952 board.

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

© 2026 NXP B.V. All rights reserved.

Table 17. Board images...*continued*

Image name	Download target	Description
u-boot-imx952.imx	bootloader0 partition.	An image containing U-Boot and ATF for the i.MX 952 19x19 EVK board.
u-boot-imx952-evk-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the i.MX 952 19x19 EVK board. It is not flashed to MMC.
u-boot-imx952-frdm.imx	bootloader0 partition.	An image containing U-Boot and ATF for the FRDM-IMX952 board.
u-boot-imx952-frdm-uuu.imx	N/A	An image containing U-Boot and ATF, used by UUU for the FRDM-IMX952 board. It is not flashed to MMC.
u-boot-imx952-rpmsg.imx	bootloader0 partition.	An image containing U-Boot and ATF to support RPMsg on the i.MX 952 19x19 EVK board.
imx952_mcu_demo.img	5120 kB offset of eMMC user partition or SD card.	The MCU image for the i.MX 952 19x19 EVK board.
partition-table.img	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-dual.img	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
partition-table-13GB.img	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
partition-table-13GB-dual.img	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.
boot.img	boot_a and boot_b partitions.	Boot image, which includes generic and vendor kernel images, generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image, which includes NXP specific kernel images, generic ramdisk.
init_boot.img	init_boot_a and init_boot_b partitions.	Initialization boot image, which includes vendor ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in VTS test with GSI system image.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and	System dlkm image built with the i.MX kernel tree for debugging.

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

© 2026 NXP B.V. All rights reserved.

Table 17. Board images*...continued*

Image name	Download target	Description
	system_dlkm_b in super partition.	
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx952.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and the OS08A20 camera on the i.MX 952 19x19 EVK board.
dtbo-imx952-ap1302.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and the AP1302 camera on the i.MX 952 19x19 EVK board.
dtbo-imx952-cs42888.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display, AUDIO-IO board, and OS08A20 camera on the i.MX 952 19x19 EVK board.
dtbo-imx952-dsi-serdes.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the DSI-to-SDSB display on the i.MX 952 19x19 EVK.
dtbo-imx952-dual-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
dtbo-imx952-lvds-dualdisp.img	dtbo_a and dtbo_b partitions.	Device Tree image to support dual LVDS-to-HDMI displays and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
dtbo-imx952-lvds-panel-ld.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel with a local dimming display on the i.MX 952 19x19 EVK board.
dtbo-imx952-lvds-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS panel display on the i.MX 952 19x19 EVK board.
dtbo-imx952-lvds0-dual-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI display and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
dtbo-imx952-lvds0.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the LVDS-to-HDMI display and the OS08A20 camera on the i.MX 952 19x19 EVK board.
dtbo-imx952-mipi-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the RM692C9 MIPI panel display on the i.MX 952 19x19 EVK board.
dtbo-imx952-mipi4k.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI4K display on the i.MX 952 19x19 EVK board.
dtbo-imx952-mqs.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display, audio MQS output, and OS08A20 camera on the i.MX 952 19x19 EVK board.
dtbo-imx952-rpmsg.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and MCU image on the i.MX 952 19x19 EVK board.
dtbo-imx952-frdm.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and the OS08A20 camera on the FRDM-IMX952 board.

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

© 2026 NXP B.V. All rights reserved.

Table 17. Board images...*continued*

Image name	Download target	Description
dtbo-imx952-frdm-ap1302.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and the AP1302 camera on the FRDM-IMX952 board.
dtbo-imx952-frdm-aud-hat.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and audio HAT on the FRDM-IMX952 board.
dtbo-imx952-frdm-boe-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the BOE panel display on the FRDM-IMX952 board.
dtbo-imx952-frdm-dual-os08a20.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and dual OS08A20 cameras on the FRDM-IMX952 board.
dtbo-imx952-frdm-waveshare-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Waveshare panel display on the FRDM-IMX952 board.
vbmeta-imx952.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and the OS08A20 camera on the i.MX 952 19x19 EVK board.
vbmeta-imx952-ap1302.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and the AP1302 camera on the i.MX 952 19x19 EVK board.
vbmeta-imx952-cs42888.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display, AUDIO-IO board, and OS08A20 camera on the i.MX 952 19x19 EVK board.
vbmeta-imx952-dsi-serdes.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the DSI-to-SDSB display on the i.MX 952 19x19 EVK board.
vbmeta-imx952-dual-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
vbmeta-imx952-lvds-dualdisp.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support dual LVDS-to-HDMI displays and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
vbmeta-imx952-lvds-panel-ld.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel with a local dimming display on the i.MX 952 19x19 EVK board.
vbmeta-imx952-lvds-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS panel display on the i.MX 952 19x19 EVK board.
vbmeta-imx952-lvds0-dual-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI display and dual OS08A20 cameras on the i.MX 952 19x19 EVK board.
vbmeta-imx952-lvds0.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the LVDS-to-HDMI display and the OS08A20 camera on the i.MX 952 19x19 EVK board.
vbmeta-imx952-mipi-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the RM692C9 MIPI panel display on the i.MX 952 19x19 EVK board.
vbmeta-imx952-mipi4k.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI4 K display on the i.MX 952 19x19 EVK board.
vbmeta-imx952-mqs.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display, audio MQS output, and OS08A20 camera on the i.MX 952 19x19 EVK board.
vbmeta-imx952-rpmsg.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the MIPI-to-HDMI display and MCU image on the i.MX 952 19x19 EVK board.

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

© 2026 NXP B.V. All rights reserved.

Table 17. Board images...*continued*

Image name	Download target	Description
vbmeta-imx952-frdm.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and the OS08A20 camera on the FRDM-IMX952 board.
vbmeta-imx952-frdm-ap1302.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and the AP1302 camera on the FRDM-IMX952 board.
vbmeta-imx952-frdm-aud-hat.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and audio HAT on the FRDM-IMX952 board.
vbmeta-imx952-frdm-boe-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the BOE panel display on the FRDM-IMX952 board.
vbmeta-imx952-frdm-dual-os08a20.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and dual OS08A20 cameras on the FRDM-IMX952 board.
vbmeta-imx952-frdm-waveshare-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Waveshare panel display on the FRDM-IMX952 board.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key. It is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note:

In this document, the logical partition is also called dynamic partition. For more Android dynamic partitions, see Dynamic partitions overview.

*With virtual A/B enabled, the `super.img` generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images.*

To boot from slot b, take either of the following actions:

- *Apply for an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

11.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to flash all Android images automatically.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.243 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

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

© 2026 NXP B.V. All rights reserved.

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.
   a. For i.MX 952 19x19 EVK:
      Change the board's SW7 to x001 (from 1-4 bit) to enter serial download mode.
   b. For FRDM-IMX952:
      Change the board's SW1 to 01 (from 1-2 bit) to enter serial download mode.
3. Power on the board. Use the USB cable on the board Type-C port to connect your PC with the board.
   Note: *The debug PORT can be used to watch the logs of the hardware boot processing.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_952evk.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For the i.MX 952 EVK board, related options are described as follows.

Table 18. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies the SoC information. For i.MX 952 19x19 EVK and FRDM-IMX952, it should be `imx952`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For i.MX 952 19x19 EVK and FRDM-IMX952, it can be followed by "13". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If it is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. • For i.MX 952 19x19 EVK, it can be `dual`, `trusty-dual`, `trusty-secure-unlock-dual`, `trusty-rbidx-blob-dual`, or `rpmsg`. • For FRDM-IMX952, it can be `frdm`, or `trusty-frdm-dual`. If it is not used, the default `u-boot-imx952.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. • For i.MX 952 19x19 EVK, It can be `ap1302`, `cs42888`, `dsi-serdes`, `dual-os08a20`, `lvds-dualdisp`, `lvds-panel-ld`, `lvds-panel`, `lvds0-dual-os08a20`, `lvds0`, `mipi-panel`, `mipi4k`, `mqs`, or `rpmsg`. • For FRDM-IMX952, it can be `frdm`, `frdm-ap1302`, `frdm-dual-os08a20`, `frdm-aud-hat`, `frdm-boe-panel`, or `frdm-waveshare-panel`. If this option is not used, the default `dtbo-imx952.img` and `vbmeta-imx952.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For i.MX 952 19x19 EVK and FRDM-IMX952, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.

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

© 2026 NXP B.V. All rights reserved.

Table 18. Options for uuu_imx_android_flash tool*...continued*

Option	Description
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

Note: *`-m` should be used together with `-d rpmsg` and `-u rpmsg`.*

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx952 -a -e -u trusty-dual
```

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> .\uuu_imx_android_flash.bat -f imx952 -a -e -u trusty-dual
```

When the command above is executed, `spl-imx952-trusty-dual.bin` is flashed, and `bootloader-imx952-trusty-dual.img` with other default images are flashed into eMMC slot a for i.MX 952 EVK.

Note:

- *`-b` is not recommended to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter contains `trusty` cannot be used together with `-t sd`, because the Trusty OS cannot boot from the SD card.*
- *To flash SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use the `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use the `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test feature on i.MX 952 19x19 EVK:*
  - *To test dual-bootloader, execute the tool with `-u dual`.*
  - *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-dual`.*
  - *To test Trusty OS, dual-bootloader and encrypted rollback index enabled condition, execute the tool with `-u trusty-rbidx-blob-dual`.*
  - *To test the demonstration implementation of secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
  - *To test the MIPI-to-HDMI display, it does not need to use the `-d` option.*
  - *To test the MIPI panel (RM692C9) display, execute the tool with `-d mipi-panel`.*
  - *To test the MIPI-to-HDMI4K display, execute the tool with `-d mipi4k`.*
  - *To test the LVDS-to-HDMI display, execute the tool with `-d lvds0`.*
  - *To test the dual LVDS-to-HDMI displays, execute the tool with `-d lvds-dualdisp`.*
  - *To test the LVDS panel display, execute the tool with `-d lvds-panel`.*
  - *To test the LVDS panel with a local dimming display, execute the tool with `-d lvds-panel-ld`.*
  - *To test the DSI-to-SDSB display, execute the tool with `-d dsi-serdes`.*
  - *To test the OS08A20 camera, it does not need to use the `-d` option, or execute the tool with `-d lvds0`, `-d cs42888`, or `-d mqs`.*
  - *To test the AP1302 camera, execute the tool with `-d ap1302`.*

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

© 2026 NXP B.V. All rights reserved.

  - *To test the dual OS08A20 cameras, execute the tool with `-d dual-os08a20`, `-d lvds0-dual-os08a20`, or `-d lvds-dualdisp`.*
  - *To test the AUDIO-IO board, execute the tool with `-d cs42888` and use the "MIPI DSI" for the MIPI-to-HDMI display.*
  - *To test the MIPI-to-HDMI display and MCU image, execute the tool with `-u rpmsg -d rpmsg -m`.*
  - *To test MQS out, execute the tool with `-d mqs`.*
- *To test the features on the FRDM-IMX952 board:*
  - *To test Trusty OS and dual-bootloader both enabled condition, execute the tool with `-u trusty-frdm-dual`.*
  - *To test the HDMI display, execute the tool with `-d frdm`, `-d frdm-dual-os08a20`, `-d frdm-ap1302`, and `-d frdm-aud-hat`.*
  - *To test the BOE panel display, execute the tool with `-d frdm-boe-panel`.*
  - *To test the Waveshare panel display, execute the tool with `-d frdm-waveshare-panel`.*
  - *To test the OS08A20 camera, execute the tool with `-d frdm`.*
  - *To test the dual OS08A20 cameras, execute the tool with `-d frdm-dual-os08a20`.*
  - *To test the AP1302 camera, execute the tool with `-d frdm-ap1302`.*
  - *To test the Audio HAT with 8MIC PDM Microphone. execute the tool with `-d frdm-aud-hat`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

`z` in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is not any error, the command window displays information indicating that images are already flashed.
   Note:
   If the target device has a DOS partition table on it, the flash process fails for the first time. Push the reset key on the board and execute the flash script again.
7. Power off the board.
8. Change the boot device to eMMC or SD card.
   - For i.MX 952 19x19 EVK:
     - Change SW7 to switch the board back to x010 (from 1-4 bit) to enter eMMC boot mode.
     - Change SW7 to switch the board back to x011 (from 1-4 bit) to enter SD boot mode.
   - For FRDM-IMX952:
     - Change SW1 to switch the board back to 10 (from 1-2 bit) to enter eMMC boot mode.
     - Change SW1 to switch the board back to 11 (from 1-2 bit) to enter SD boot mode.

11.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

11.4.1 Booting with an MCU image

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.lpa.enable=1
U-Boot > setenv append_bootargs snd_pcm.max_alloc_per_card=134217728
 pd_ignore_unused cma=600M
```


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

© 2026 NXP B.V. All rights reserved.

```
U-Boot > setenv bootcmd "bootmcu && boota"
U-Boot > saveenv
```

Note:

To use other dtbo images, do not add `bootmcu" to "bootcmd`. The following command can recover `bootcmd`:

```
U-Boot > setenv bootcmd "boota"
U-Boot > saveenv
```

11.4.2 Booting with display: HDMI 4K display

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.displaymode=4kp30
 androidboot.lcd_density=480 androidboot.dpu_composition=0
U-Boot > saveenv
```

11.4.3 Booting with multi-channel audio support

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.audio.multichannel=1
U-Boot > saveenv
```

11.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

12 Working with the FRDM-IMX937 board

12.1 Board hardware

The figures below show the different components of the FRDM-IMX937 board.

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

© 2026 NXP B.V. All rights reserved.

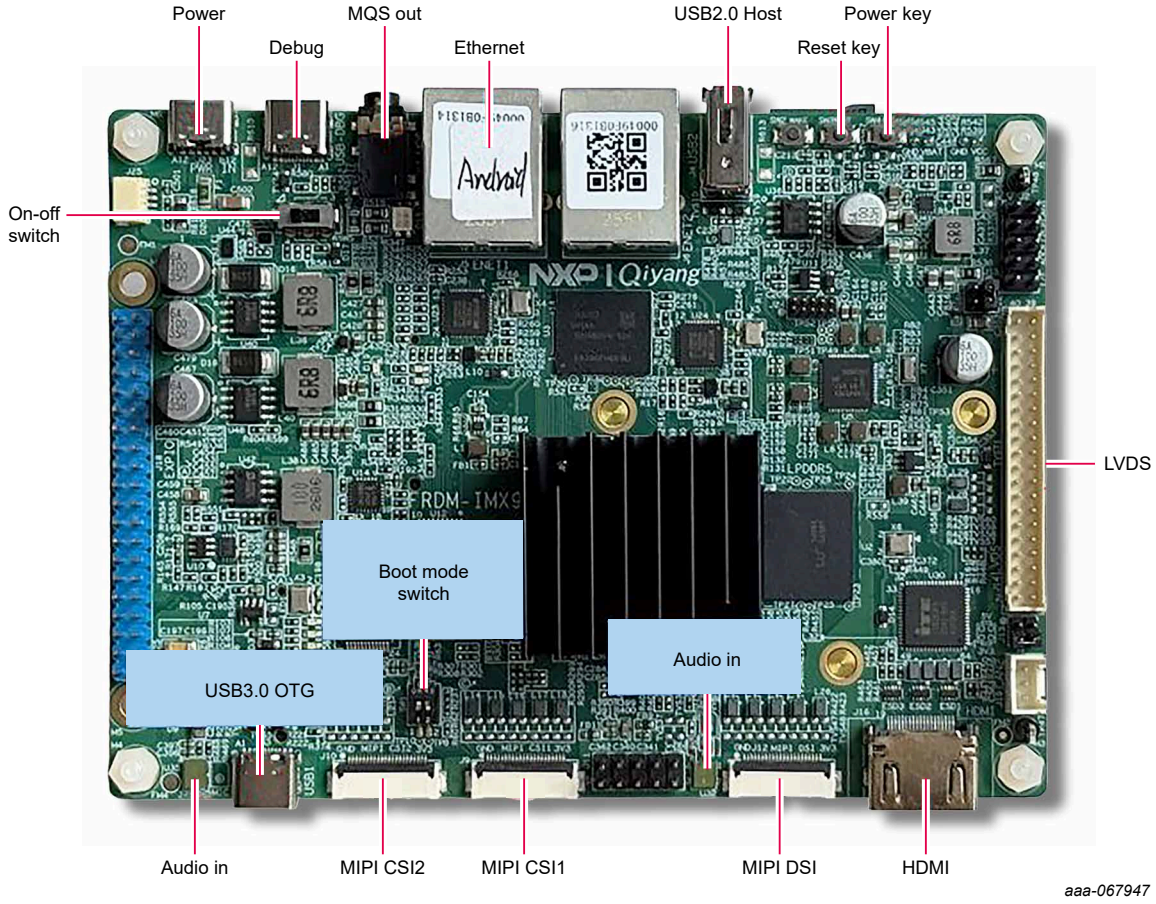


Figure 96. FRDM-IMX937

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

© 2026 NXP B.V. All rights reserved.

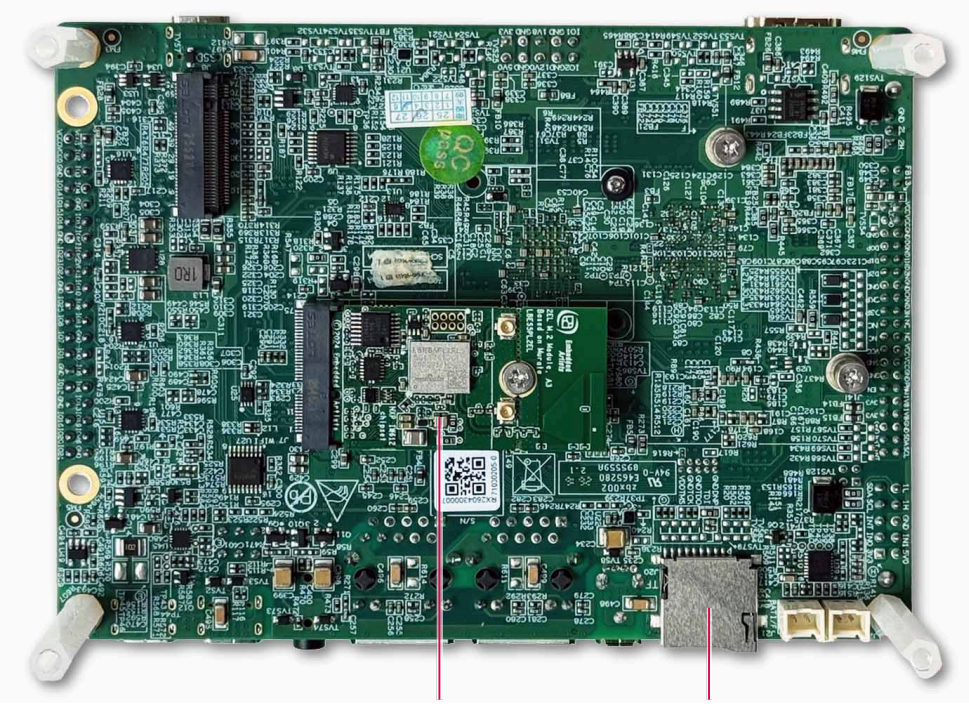


Figure 97. FRDM-IMX937 back view

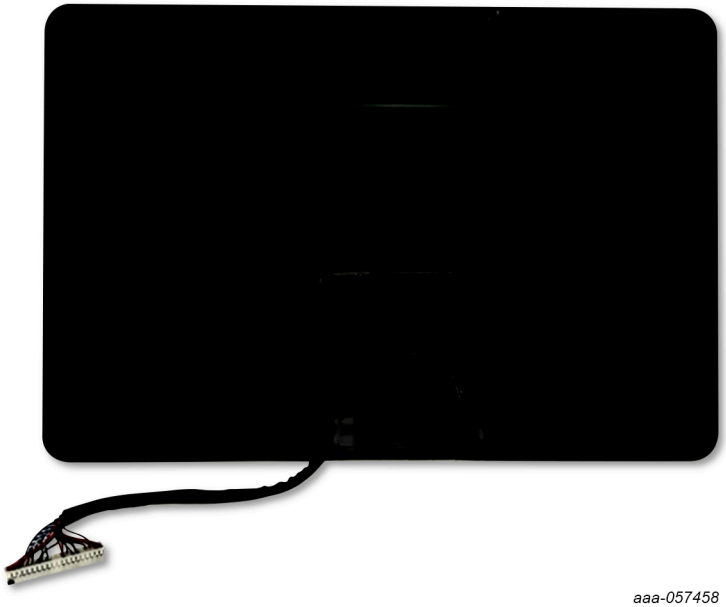


Figure 98. i.MX BOE panel

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

© 2026 NXP B.V. All rights reserved.

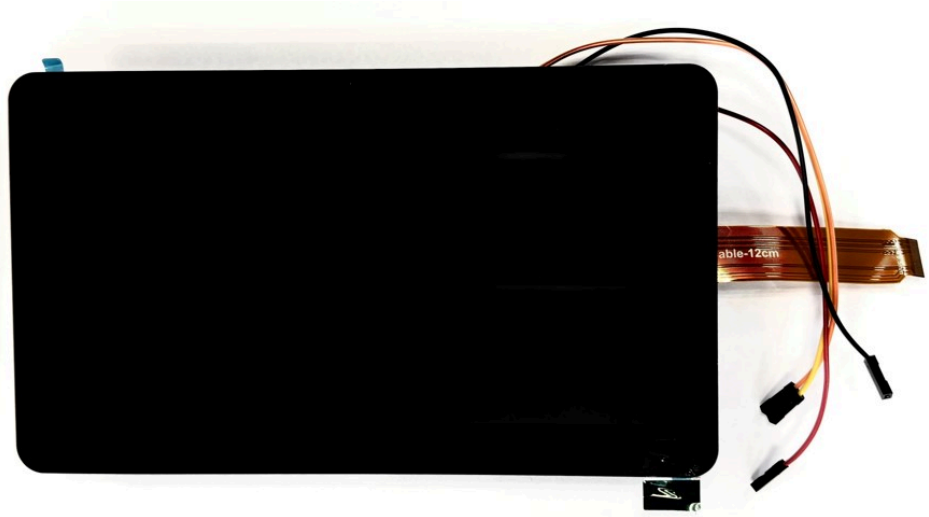


Figure 99. i.MX Waveshare panel

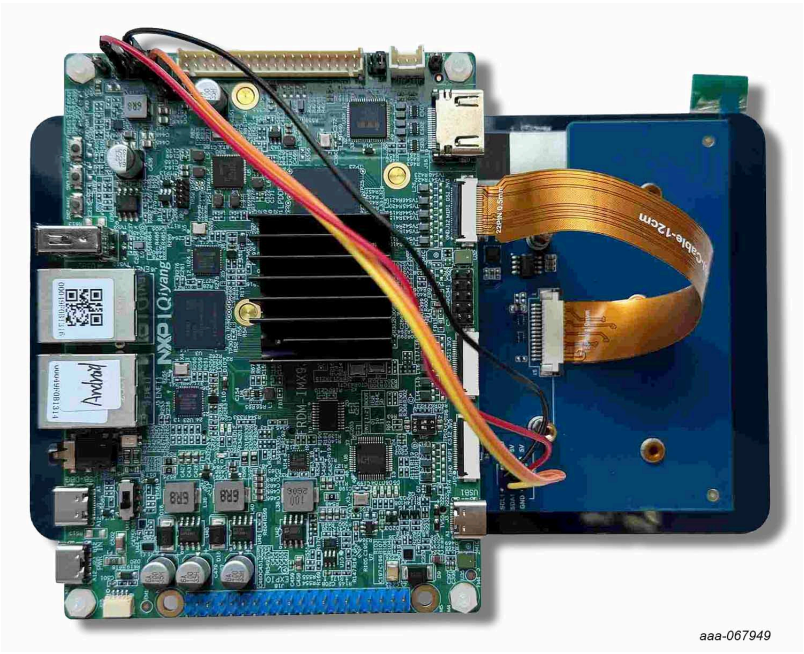
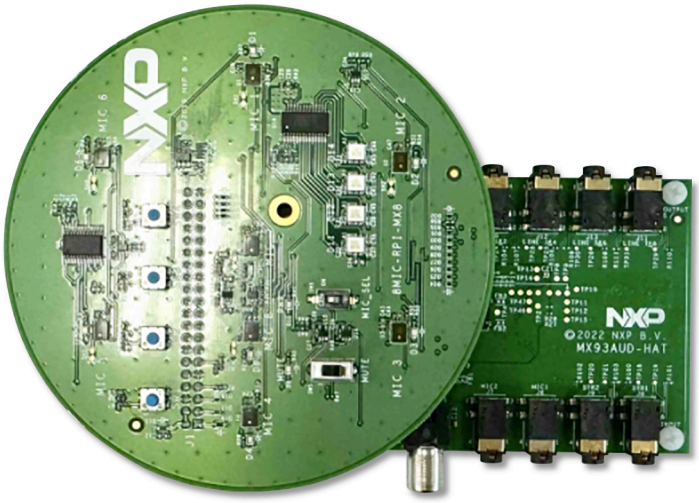


Figure 100. Hardware connection between FRDM-IMX937 and Waveshare panel

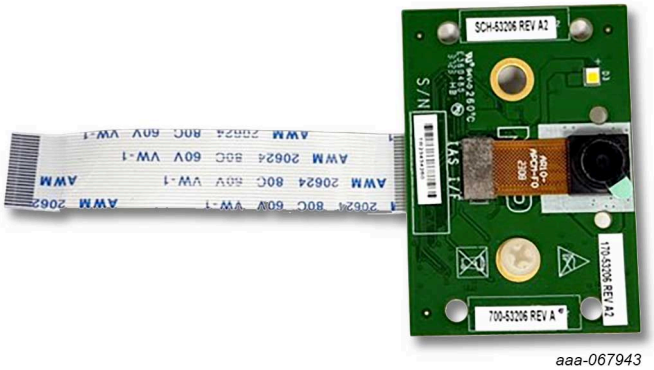
All information provided in this document is subject to legal disclaimers.

© 2026 NXP B.V. All rights reserved.



aaa-057459

Figure 101. Audio HAT board + i.MX 8MIC PDM Microphone board



aaa-067943

Figure 102. i.MX Camera AP1302 with 22-pin flat flex cable

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

© 2026 NXP B.V. All rights reserved.

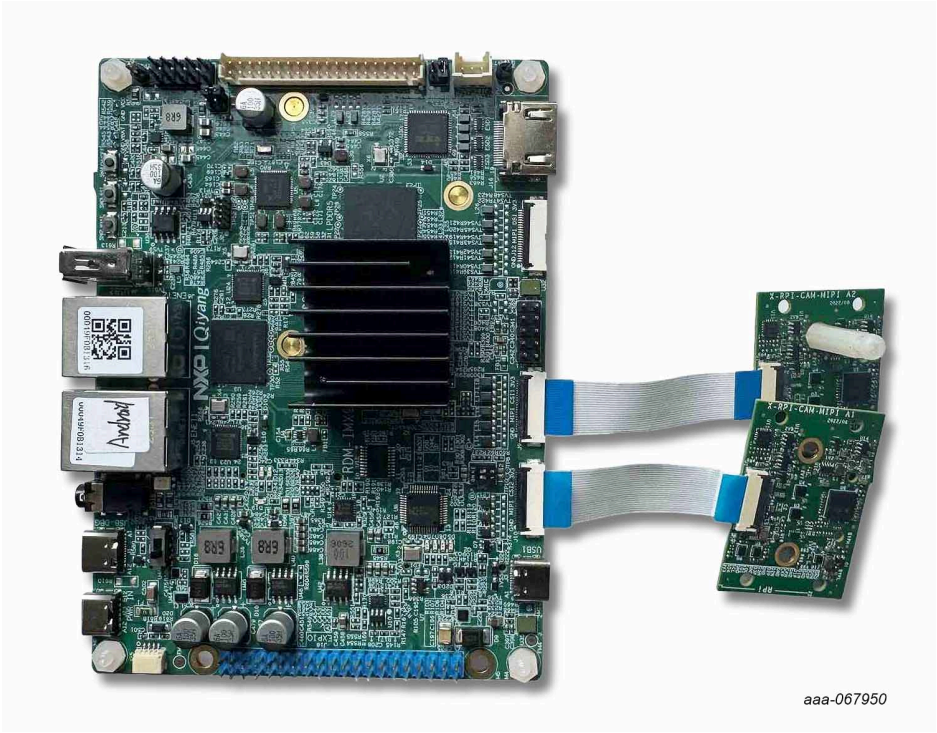


Figure 103. Hardware connection between FRDM-IMX937 and dual AP1302 cameras

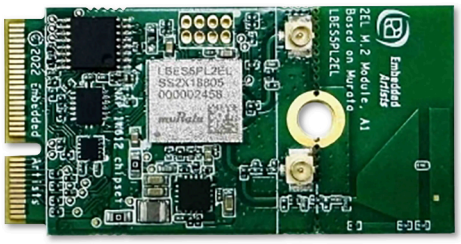


Figure 104. SDIW612 (Murata LBES5PL2EL)

Note:

- *The FRDM-IMX937 board is supported in this release.*
- *To test the HDMI display on FRDM-IMX937, connect the HDMI display to the "HDMI" port on FRDM-IMX937.*
- *To test the BOE panel display on FRDM-IMX937, connect i.MX BOE panel to the "LVDS" port on FRDM-IMX937.*
- *To test the Waveshare panel display on FRDM-IMX937, connect the i.MX Waveshare panel to the "MIPI DSI" port and the J2 connector on FRDM-IMX937.*
  See Figure 100. Ensure that Pin 1 on the Waveshare connects to the 3V3 pin of the "MIPI DSI" port on FRDM-IMX937 through the 15-pin to 22-pin flat flex cable.
  - *Connect the Waveshare panel to the FRDM-IMX937 using jumper wires: 5 V pin on the Waveshare to Pin 2 of connector J2 on the FRDM-IMX937.*
  - *Connect the GND pin on the Waveshare to Pin 6 of J2C.*
  - *Connect the SCL pin to Pin 7 of J2.*
  - *Connect the SDA pin to pin 9 of J2.*

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

© 2026 NXP B.V. All rights reserved.

- *To test dual AP1302 cameras on FRDM-IMX937, connect the i.MX camera AP1302 with the 22-pin flat flex cable to the "MIPI CSI1" and "MIPI CSI2" ports.*
  See Figure 103. Ensure that the FRDM-IMX937 is facing upward and the AP1302 is facing downward. Connect them using a 22-pin flat flex cable.
- *To test the Audio HAT board + i.MX 8MIC PDM Microphone board on the FRDM-IMX937 board, connect the AUDIO-IO board to the J18 connector.*
- *Connect the SDIW612 (Murata LBES5PL2EL) Wi-Fi&BT M.2 expansion card to the J7 connector to enable the Wi-Fi and Bluetooth functionality on the FRDM-IMX937 board.*

12.2 Board images

The table below describes the location in the board partitions of the software images in `android-17.0.0_1.0.0_image_937frdm.tar.gz`.

Table 19. Board images

Image name	Download target	Description
`spl-imx937-dual.bin`	`bootloader0` partition.	Secondary program loader image without Trusty related configurations for the FRDM-IMX937 board.
`spl-imx937-trusty-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty related configurations for the FRDM-IMX937 board.
`spl-imx937-trusty-rbidx-blob-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and encrypted rollback index blob related configurations for the FRDM-IMX937 board.
`spl-imx937-trusty-secure-unlock-dual.bin`	`bootloader0` partition.	Secondary program loader image with Trusty and secure unlock related configurations for the FRDM-IMX937 board.
`bootloader-imx937-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper and ATF. It is for the FRDM-IMX937 board.
`bootloader-imx937-trusty-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, and Trusty OS. It is for the FRDM-IMX937 board.
`bootloader-imx937-trusty-rbidx-blob-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and encrypted rollback index enabled. It is for the FRDM-IMX937 board.
`bootloader-imx937-trusty-secure-unlock-dual.img`	`bootloader_a` and `bootloader_b` partitions.	An image containing U-Boot proper, ATF, Trusty OS, and secure unlock enabled. It is for the FRDM-IMX937 board.
`u-boot-imx937.imx`	`bootloader0` partition.	An image containing U-Boot and ATF for the FRDM-IMX937 board.
`u-boot-imx937-frdm-uuu.imx`	N/A	An image containing U-Boot and ATF, used by UUU for the FRDM-IMX937 board. It is not flashed to MMC.
`partition-table.img`	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 28 GB.
`partition-table-dual.img`	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 28 GB.
`partition-table-13GB.img`	0 offset of MMC.	GPT partition table image for single-bootloader condition and the target storage device should be larger than 13 GB.
`partition-table-13GB-dual.img`	0 offset of MMC.	GPT partition table image for dual-bootloader condition and the target storage device should be larger than 13 GB.

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

© 2026 NXP B.V. All rights reserved.

Table 19. Board images...*continued*

Image name	Download target	Description
boot.img	boot_a and boot_b partitions.	Boot image, which includes generic and vendor kernel images, generic ramdisk.
boot-imx.img	boot_a and boot_b partitions.	Boot image, which includes NXP specific kernel images, generic ramdisk.
init_boot.img	init_boot_a and init_boot_b partitions.	Initialization boot image, which includes vendor ramdisk.
vendor_boot.img	vendor_boot_a and vendor_boot_b partitions.	Vendor boot image. It contains vendor ramdisk, kernel commandline, and bootconfig.
vendor_boot-debug.img	vendor_boot_a and vendor_boot_b partitions.	Used in VTS test with GSI system image.
system.img	Logical partitions system_a and system_b in super partition.	System image.
system_ext.img	Logical partitions system_ext_a and system_ext_b in super partition.	System extension image.
system_dlkm.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dynamically loadable kernel module image.
system_dlkm-imx.img	Logical partitions system_dlkm_a and system_dlkm_b in super partition.	System dlkm image built with the i.MX kernel tree for debugging.
vendor.img	Logical partitions vendor_a and vendor_b in super partition.	Vendor image.
vendor_dlkm.img	Logical partitions vendor_dlkm_a and vendor_dlkm_b in super partition.	Vendor dynamically loadable kernel module image.
product.img	Logical partitions product_a and product_b in super partition.	Product image.
super.img	Super partition.	Super image, which contains images for logical partitions.
dtbo-imx937.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the MIPI-to-HDMI display and the OS08A20 camera on the FRDM-IMX937 board.
dtbo-imx937-boe-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the BOE panel display on the FRDM-IMX937 board.
dtbo-imx937-waveshare-panel.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the Waveshare panel display on the FRDM-IMX937 board.
dtbo-imx937-aud-hat.img	dtbo_a and dtbo_b partitions.	Device Tree image to support the HDMI display and audio HAT on the FRDM-IMX937 board.

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

© 2026 NXP B.V. All rights reserved.

Table 19. Board images...*continued*

Image name	Download target	Description
vbmeta-imx937.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and dual AP1302 cameras on the FRDM-IMX937 board.
vbmeta-imx937-boe-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the BOE panel display on the FRDM-IMX937 board.
vbmeta-imx937-waveshare-panel.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the Waveshare panel display on the FRDM-IMX937 board.
vbmeta-imx937-aud-hat.img	vbmeta_a and vbmeta_b partitions.	Android Verify Boot metadata image to support the HDMI display and audio HAT on the FRDM-IMX937 board.
rpmb_key_test.bin	N/A	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00. See the *Android User's Guide* (UG10156) for how to set the RPMB key.
testkey_public_rsa4096.bin	N/A	Prebuilt AVB public key. It is extracted from the default AVB private key. See the *Android User's Guide* (UG10156) for how to provision the public key.

Note:

In this document, the logical partition is also called dynamic partition. For more Android dynamic partitions, see Dynamic partitions overview.

*With virtual A/B enabled, the `super.img` generated by the Android build system only has slot a dynamic partitions filled with images. Slot b dynamic partitions like `system_b` is **0** in size. Therefore, the system cannot boot up from slot b with prebuilt images.*

To boot from slot b, take either of the following actions:

- *Apply for an OTA update.*
- *Boot the board into fastbootd mode, and flash the images to slot b logical partitions. Set slot b as the active slot.*

12.3 Flashing board images

The board image files can be flashed into the target board using Universal Update Utility (UUU).

For the UUU binary file, download it from GitHub: uuu release page on GitHub.

To achieve more flexibility, two script files are provided to invoke UUU to flash all Android images automatically.

- `uuu_imx_android_flash.sh` for Linux OS
- `uuu_imx_android_flash.bat` for Windows OS

For this release, these two scripts are validated on UUU 1.5.243 version. Download the corresponding version from GitHub:

- For Linux OS, download the file named `uuu`.
- For Windows OS, download the file named `uuu.exe`.

Because the two script files will directly invoke UUU, make sure that UUU is in a path contained by the system environment variable of "PATH".

Perform the following steps to download the board images:

1. Download the UUU binary file from GitHub as described before. Install UUU into a directory contained by the system environment variable of "PATH".
2. Make the board enter serial download mode.

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

© 2026 NXP B.V. All rights reserved.

Change the board's SW1 to 01 (from 1-2 bit) to enter serial download mode.

3. Power on the board. Use the USB cable on the board Type-C port to connect your PC with the board.
   Note: *The debug PORT can be used to watch the logs of the hardware boot processing.*
4. Decompress `release_package/android-17.0.0_1.0.0_image_937frdm.tar.gz`. The package contains the image files and `uuu_imx_android_flash` tool.
5. Execute the `uuu_imx_android_flash` tool to flash images.
   The `uuu_imx_android_flash` tool can be executed with options to get help information and specify the images to be flashed. For the FRDM-IMX937 board, related options are described as follows.

Table 20. Options for uuu_imx_android_flash tool

Option	Description
`-h`	Displays the help information of this tool.
`-f soc_name`	Specifies the SoC information. For FRDM-IMX937, it should be `imx937`. This option is **mandatory**.
`-a`	Only flashes slot a physical partitions and other partitions do not have slot a/b. If this option and `-b` option are not used, slots a and b are both flashed.
`-b`	Only flashes slot b physical partitions and other partitions do not have slot a/b. If this option and `-a` option are not used, slots a and b are both flashed.
`-c card_size`	Specifies which partition table image file to flash. For FRDM-IMX937, it can be followed by "13". If this option is not used, default `partition-table.img` or `partition-table-dual.img` is flashed.
`-m`	Flashes the MCU image. If it is not used, the MCU image is not flashed.
`-u uboot_feature`	Flashes U-Boot or SPL&bootloader images with `uboot_feature` in their names. For FRDM-IMX937, it can be `dual`, `trusty-dual`, `trusty-secure-unlock-dual`, or `trusty-rbidx-blob-dual`. If it is not used, the default `u-boot-imx937.imx` is flashed.
`-d dtb_feature`	Flashes DTBO and vbmeta images with `dtb_feature` in their names. It can be `aud-hat`, `boe-panel`, or `waveshare-panel`. If this option is not used, the default `dtbo-imx937.img` and `vbmeta-imx937.img` are flashed.
`-e`	Erases user data after images are flashed.
`-D directory`	Specifies the directory in which there are the images to be flashed. For `uuu_imx_android_flash.bat`, it must be followed with an absolute path. If this option is not used, images in the current working directory are flashed.
`-t target_dev`	Specifies the target device. For FRDM-IMX937, it can be `emmc` and `sd`. If this option is not used, images are flashed to eMMC.
`-daemon`	Runs UUU in daemon mode. This option is used to flash multiple boards of the same type.
`-i`	If the script is executed with this option, no image is flashed. The script only loads U-Boot into RAM and then executes it in fastboot mode. This option is used for development.
`-dryrun`	Only generates a UUU script but not execute UUU with this script.
`-usb usb_path`	Specifies a USB path like 1:1 to monitor. It can be used multiple times to specify more than one path.

- On the Linux system, open the shell terminal. For example, you can execute a command as follows:

```
> sudo ./uuu_imx_android_flash.sh -f imx937 -a -e -u trusty-dual
```


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

© 2026 NXP B.V. All rights reserved.

- On the Windows system, open the command-line interface in administrator mode. The corresponding command is as follows:

```
> .\uuu_imx_android_flash.bat -f imx937 -a -e -u trusty-dual
```

When the command above is executed, `spl-imx937-trusty-dual.bin` is flashed, and `bootloader-imx937-trusty-dual.img` with other default images are flashed into eMMC slot a for FRDM-IMX937.

Note:

- *`-b` is not recommended to be used, because virtual A/B is enabled and the system cannot boot up from slot b with the prebuilt images.*
- *`-u` followed with a parameter contains `trusty` cannot be used together with `-t sd`, because the Trusty OS cannot boot from the SD card.*
- *To flash SD card, execute the tool with `-t sd`. To flash eMMC, it does not need to use the `-t` option.*
- *If your SD card is 32 GB or uses onboard eMMC as the boot device, it does not need to use the `-c` option.*
- *If your SD card is 16 GB, execute the tool with `-c 13`.*
- *To test dual-bootloader, execute the tool with `-u dual` for FRDM-IMX937.*
- *To test the Trusty OS and dual-bootloader both enabled condition for FRDM-IMX937, execute the tool with `-u trusty-dual`.*
- *To test the Trusty OS, dual-bootloader, and encrypted rollback index enabled condition for FRDM-IMX937, execute the tool with `-u trusty-rbidx-blob-dual`.*
- *To test the demonstration implementation of the secure unlock, execute the tool with `-u trusty-secure-unlock-dual`. For secure unlock details, see the i.MX Android Security User's Guide (UG10158).*
- *To test the features on FRDM-IMX937:*
  - *To test the HDMI display, it does not need to use the `-d` option.*
  - *To test the BOE panel display, execute the tool with `-d boe-panel`.*
  - *To test dual AP1302 cameras, it does not need to use the `-d` option.*
  - *To test Audio HAT with 8MIC PDM Microphone, execute the tool with `-d aud-hat`.*
- *If `uuu_imx_android_flash.bat` is used to flash images on a remote server through Samba, map the remote resource to the local environment first. Take the following command as an example:*

```
> net use z: \\192.168.1.1\daily_images
```

`z` in the command represents an available drive letter. It can be other available drive letter.

6. Wait for the `uuu_imx_android_flash` execution to complete. If there is not any error, the command window displays information indicating that images are already flashed.
7. Power off the board.
8. Change the boot device to eMMC or SD card.
   - Change SW1 to switch the board back to 10 (from 1-2 bit) to enter eMMC boot mode.
   - Change SW1 to switch the board back to 11 (from 1-2 bit) to enter SD boot mode.

12.4 Booting

After downloading the images, boot the board by connecting it to the power supply.

Some specific U-Boot environment variables may need to be set in some boot scenarios to boot the MCU image or to get better display effect.

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

© 2026 NXP B.V. All rights reserved.

12.4.1 Booting with multi-channel audio support

In the U-Boot prompt, set the U-Boot environment variables as follows:

```
U-Boot > setenv append_bootconfig androidboot.audio.multichannel=1
U-Boot > saveenv
```

12.5 Board reboot

After you have completed download and setup, reboot the board and wait for the Android platform to boot up.

13 Acronyms

Table 21. Acronyms

Acronym	Description
AOSP	Android Open Source Project
AVB	Android Verified Boot
BT	Bluetooth
CSI	Camera Serial Interface
CST	(NXP) Code Signing Tool
DDR4	Double Data Rate 4
DSI	Display Serial Interface
eMMC	Embedded Multi-Media Card
EVK	Evaluation Kit
GPT	GUID partition table
HDMI	High-Definition Multimedia Interface
LPDDR4	Lower Power Double Data Rate 4
LVDS	Low-Voltage Differential Signaling
MCU	Microcontroller Unit
MEK	Multisensory Enablement Kit
MIPI	Mobile Industry Processor Interface
OS	Operating system
OTA	Over-The-Air programming
OTG	On-The-Go
PC	Personal (host) computer
PCIe (or PCIE)	Peripheral Component Interconnect Express
PDM	Pulse Density Modulation
RAM	Random Access Memory
RPMB	Replay Protected Memory Block
SAS	Serial Attached SCSI (Serial Attached Small Computer System Interface)
SoC	System on Chip

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

© 2026 NXP B.V. All rights reserved.

Table 21. Acronyms...*continued*

Acronym	Description
SPL	U-Boot Secondary Program Loader
SOF	Sound Open Firmware
UART	Universal Asynchronous Receiver/Transmitter
U-Boot	Universal Boot Loader
UUU	Universal Update Utility

14 Note about the source code in the document

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

Copyright 2018-2026 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.

15 Revision history

Revision history

Document ID	Release date	Description
UG10157 v.android-17.0.0_1.0.0	3 November 2026	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus, i.MX 95, i.MX 952 GA release, and FRDM-IMX937 Beta release.
UG10157 v.android-16.0.0_2.0.0	24 July 2026	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus, i.MX 95 GA release, and i.MX 952 Beta release.
UG10157 v.android-16.0.0_1.4.0	1 April 2026	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus, and i.MX 95 GA release.
UG10157 v.android-16.0.0_1.2.0	6 January 2026	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus, and i.MX 95 GA release.

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

© 2026 NXP B.V. All rights reserved.

Revision history...*continued*

Document ID	Release date	Description
UG10157 v.android-16.0.0_1.0.0	28 October 2025	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, and i.MX 95 Beta release.
UG10157 v.android-15.0.0_2.0.0	24 July 2025	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, i.MX 95 19x19 EVK Beta release, and i.MX 95 15x15 EVK Alpha release.
UG10157 v.android-15.0.0_1.2.0	11 April 2025	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, and i.MX 95 Beta release.
UG10157 v.android-15.0.0_1.0.0	24 January 2025	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, and i.MX 95 Beta release.
UG10157 v.android-14.0.0_2.2.0	18 October 2024	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, i.MX 95 (A1 15x15) Alpha release, and i.MX 95 (A1 19x19) Beta release.
UG10157 v.android-14.0.0_2.0.0	9 August 2024	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8ULP, i.MX 8QuadMax, i.MX 8QuadXPlus GA release, and i.MX 95 Alpha release. Updated the document ID.
AQSUG v.android-14.0.0_1.2.0	19 April 2024	i.MX 8ULP EVK, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8Quad XPlus GA release.
AQSUG v.android-14.0.0_1.0.0	6 February 2024	i.MX 8ULP EVK, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8Quad XPlus GA release.
AQSUG v.android-13.0.0_2.2.0	24 October 2023	i.MX 8ULP EVK, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8Quad XPlus GA release.
AQSUG v.android-13.0.0_2.0.0	07/2023	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8 QuadXPlus GA release.
AQSUG v.android-13.0.0_1.2.0	03/2023	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8 QuadXPlus GA release.
AQSUG v.android-13.0.0_1.0.0	01/2023	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8 QuadXPlus GA release.
AQSUG v.android-12.1.0_1.0.0	10/2022	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8 QuadXPlus GA release.
AQSUG v.android-12.0.0_2.0.0	07/2022	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.
AQSUG v.android-12.0.0_1.0.0	03/2022	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.

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

© 2026 NXP B.V. All rights reserved.

Revision history...*continued*

Document ID	Release date	Description
AQSUG v.android-11.0.0_2.6.0	01/2022	i.MX 8ULP EVK Beta release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.
AQSUG v.android-11.0.0_2.4.0	10/2021	i.MX 8ULP EVK Alpha release, i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.
AQSUG v.android-11.0.0_2.2.0	07/2021	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.
AQSUG v.android-11.0.0_2.0.0	04/2021	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Plus, and i.MX 8M Quad GA release.
AQSUG v.android-11.0.0_1.0.0	12/2020	i.MX 8M Plus EVK Beta release, and all the other i.MX 8 GA release.
AQSUG v.android-10.0.0_2.3.0	07/2020	i.MX 8M Plus EVK Beta1 release, and all the other i.MX 8 GA release.
AQSUG v.android-10.0.0_2.0.0	05/2020	i.MX 8M Mini, i.MX 8M Nano, i.MX 8M Quad, i.MX 8Quad Max, and i.MX 8QuadXPlus GA release.
AQSUG v.android-10.0.0_2.1.0	04/2020	i.MX 8M Plus Alpha and i.MX 8QuadXPlus Beta release.
AQSUG v.android-10.0.0_1.0.0	03/2020	Deleted the Android 10 image.
AQSUG v.android-10.0.0_1.0.0	02/2020	i.MX 8M Mini, i.MX 8M Quad, i.MX 8QuadMax, and i.MX 8 QuadXPlus GA release.
AQSUG v.P9.0.0_2.0.0-ga	08/2019	Updated the location of the SCFW porting kit.
AQSUG v.P9.0.0_2.0.0-ga	04/2019	i.MX 8M, i.MX 8QuadMax, i.MX 8QuadXPlus GA release.
AQSUG v.P9.0.0_1.0.0-ga	01/2019	i.MX 8M, i.MX 8QuadMax, i.MX 8QuadXPlus GA release.
AQSUG v.P9.0.0_1.0.0-beta	11/2018	Initial release

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

© 2026 NXP B.V. All rights reserved.

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.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

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.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

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.

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

© 2026 NXP B.V. All rights reserved.

Contents

1 Overview 2
2 Hardware requirements 2
3 Working with the i.MX 8M Mini EVK board 2
3.1 Board hardware 2
3.2 Board images 5
3.3 Flashing board images 7
3.4 Booting 10
3.4.1 Booting with audio playback based on Cortex-M4 FreeRTOS 10
3.5 Board reboot 10
4 Working with the i.MX 8M Nano EVK board 10
4.1 Board hardware 10
4.2 Board images 12
4.3 Flashing board images 15
4.4 Booting 18
4.4.1 Booting with an MCU image 18
4.5 Board reboot 18
5 Working with the i.MX 8M Plus EVK board 18
5.1 Board hardware 18
5.2 Board images 22
5.3 Flashing board images 26
5.4 Booting 29
5.4.1 Booting with audio playback based on Cortex-M7 FreeRTOS 29
5.4.2 Booting with a single display: HDMI 4K display 29
5.5 Board reboot 29
6 Working with the i.MX 8M Quad WEVK board 29
6.1 Board hardware 29
6.2 Board images 32
6.3 Flashing board images 35
6.4 Booting 37
6.5 Board reboot 37
7 Working with the i.MX 8ULP EVK board 37
7.1 Board hardware 38
7.2 Board images 41
7.3 Flashing board images 44
7.4 Booting 46
7.4.1 Booting with an HDMI display 46
7.5 Board reboot 46
8 Working with the i.MX 8QuadMax MEK board 47
8.1 Board hardware 47
8.2 Board images 48
8.3 Flashing board images 52
8.4 Booting 55
8.4.1 Booting with multi-channel audio support 55
8.5 Board reboot 56
9 Working with the i.MX 8QuadXPlus MEK board 56
9.1 Board hardware 56
9.2 Board images 57
9.3 Flashing board images 60
9.4 Booting 63
9.4.1 Booting with multi-channel audio support 63
9.5 Board reboot 63
10 Working with the i.MX 95 EVK and FRDM-IMX95 boards 63
10.1 Board hardware 63
10.2 Board images 73
10.3 Flashing board images 78
10.4 Booting 82
10.4.1 Booting with an MCU image 82
10.4.2 Booting with a display: HDMI 4K display 82
10.4.3 Booting with multi-channel audio support 82
10.5 Board reboot 82
11 Working with the i.MX 952 EVK and FRDM-IMX952 boards 83
11.1 Board hardware 83
11.2 Board images 92
11.3 Flashing board images 96
11.4 Booting 99
11.4.1 Booting with an MCU image 99
11.4.2 Booting with display: HDMI 4K display 100
11.4.3 Booting with multi-channel audio support 100
11.5 Board reboot 100
12 Working with the FRDM-IMX937 board 100
12.1 Board hardware 100
12.2 Board images 106
12.3 Flashing board images 108
12.4 Booting 110
12.4.1 Booting with multi-channel audio support 111
12.5 Board reboot 111
13 Acronyms 111
14 Note about the source code in the document 112
15 Revision history 112
Legal information 115

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© 2026 NXP B.V. **All rights reserved.**