Document Number: AQSUG
Rev. O8.1.0_2.0.1-ga, 06/2019

# Android™ Quick Start Guide

Contents

1 Overview....1
2 Hardware Requirements....1
3 Working with the i.MX 8QuadMax MEK Board....1
4 Revision History....8

## 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 default configuration. For details on using the release package, see the *Android™ User's Guide* (AUG) 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 8QuadMax

Supported boards:

- MEK board and Platform

## 3 Working with the i.MX 8QuadMax MEK Board

## 3.1 Board hardware

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

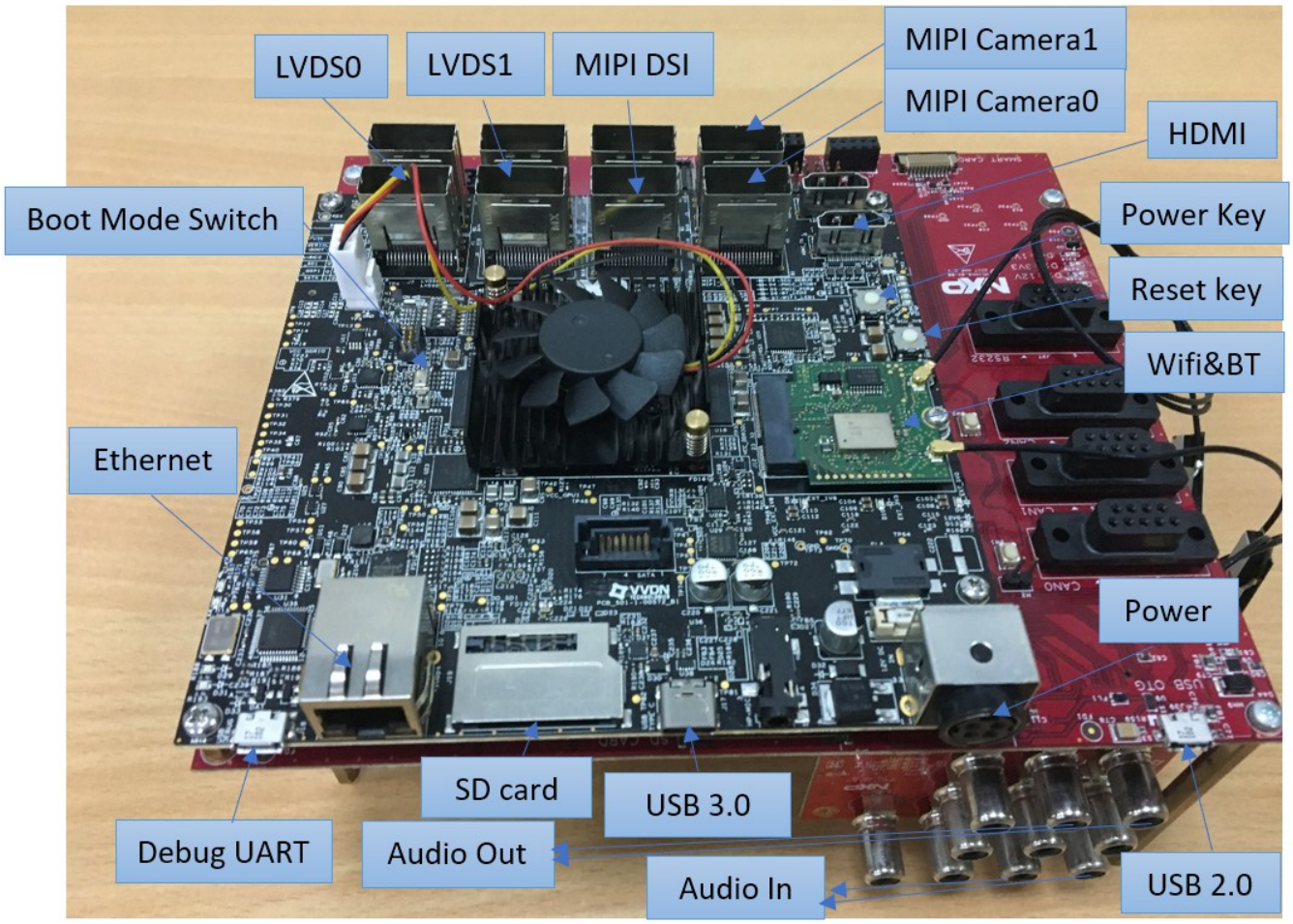


**Figure 1. i.MX 8QuadMax MEK board**

**Figure 2. i.MX mini SAS cable with DSI-to-HDMI adapter**

**Figure 3. i.MX mini SAS cable with LVDS-to-HDMI adapter**

**Figure 4. i.MX MIPI panel**

**Figure 5. i.MX MIPI camera**

**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 the i.MX MIPI Camera to the "MIPI Camera" port and connect the OV5640 Camera to "CSI Camera" port at the same time.
    - i.MX 8QuadMax MEK
        - To test a single camera, connect the i.MX MIPI camera to the "MIPI Camera0" or "MIPI Camera1" port.
        - To test dual cameras, connect both.
        - To test the MIPI panel display, connect the i.MX MIPI panel to the "MIPI DSI" port.

## 3.2 Board images

The table below describes the location in the board partitions of the software images in android_o8.1.0_2.0.1-ga_image_8qmek.tar.gz.

**Table 1. Board images**

| Image name | Download target |
|---|---|
| /u-boot-imx8qm.imx | 0 KB offset of eMMC and 32 KB offset of SD card for i.MX 8QuadMax. |
| u-boot-imx8qm-mek-uuu.imx | Bootloader used by UUU for i.MX 8QuadMax MEK board. It is not flashed to MMC. |
| /dtbo-imx8qm.img | dtbo_a and dtbo_b partitions for i.MX 8QuadMax to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display. |
| /dtbo-imx8qm-hdmi.img | dtbo_a and dtbo_b partitions for i.MX 8QuadMax to support physical HDMI display. |
| dtbo-imx8qm-mipi-panel.img | dtbo_a and dtbo_b partitions for i.MX 8QuadMax to support MIPI panel display. |
| /vbmeta-imx8qm.img | vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax to support single LVDS-to-HDMI/MIPI-to-HDMI or dual LVDS-to-HDMI display. |
| /vbmeta-imx8qm-hdmi.img | vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax to support physical HDMI display. |
| /vbmeta-imx8qm-mipi-panel.img | vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax to support MIPI panel display. |
| /boot.img | boot_a and boot_b paritions |
| /partition-table.img | Program to the first 17 KB, and then back up to the last 17 KB of the boot storage. GPT table image for 16 GB boot storage. |
| /partition-table-7GB.img | Program to the first 17 KB, and then back up to last 17 KB of the boot storage. GPT table image for 8 GB boot storage. |
| /partition-table-28GB.img | Program to first 17 KB, and then back up to last 17 KB of the boot storage. GPT table image for 32 GB boot storage. |
| /system.img | system_a and system_b partitions. |
| /vendor.img | vendor_a and vendor_b partitions. |

## 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 1.2.91 version. Download 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 direclty 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 for i.MX 8QuadMax.
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 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_8.1.0_2.0.1-ga_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 Android images on the i.MX 8QuadMax MEK 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<br>soc_name | Specifies the SoC information. For 8QuadMax, it should be "imx8qm". 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. |
| -d<br>dev | Specifies some images with "dev" in its name. For 8QuadMax, it can be "hdmi", "mipi-panel". If this option is not used, default dtbo and vbmeta images are flashed. |
| -e | Erases user data after images are flashed. |
| -D<br>directory | Specifies the directory in which there are the images to be flashed. If this option is not used, images in the current working directory are flashed. |
| -daemon | Run 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 will be flashed. The script loads U-Boot to RAM and executes to fastboot mode. this option is used for development. |

- On 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
```

- On 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
```

When the command above is executed, the default images will be flashed into eMMC slot a for 8QuadMax.

**NOTE**

- 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".
- If your SD card is 8 GB, execute the tool with "-c 7".
- i.MX 8QuadMax MEK:
  - To test LVDS-to-HDMI/MIPI-to-HDMI display, it does not need to use -d option.
  - To test MIPI panel display, execute the tool with "-d mipi-panel".
  - To test physical HDMI display, execute the tool with "-d hdmi".
- uuu_imx_android_flash.bat generates a temporary file under the current working directory. Make sure you have Write permission under the current working directory.
- 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: \\10.193.108.179\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, you will get information on the command window indicating that images are already flashed.
7. Power off the board.
8. Change boot device as eMMC or SD card.
   - Change SW2 to switch the board back to 000100 (from 1-6 bit) to enther eMMC boot mode for i.MX 8QuadMax.
   - Change SW2 to sithch the board back to 001100 (from 1-6 bit) to enther SD boot mode for i.MX 8QuadMax.

## 3.4 Booting

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

### 3.4.1 Booting with single LVDS-to-HDMI/MIPI-to-HDMI/MIPI panel or dual LVDS-to-HDMI display

The i.MX 8QuadMax MEK boards support single LVDS-to-HDMI, single MIPI-to-HDMI and dual LVDS-to-HDMI Display, and MIPI panel display. They share the same bootargs.

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

```
U-Boot > setenv bootargs console=ttyLP0,115200 earlycon=lpuart32,0x5a060000,115200 init=/
init androidboot.console=ttyLP0 consoleblank=0 androidboot.hardware=freescale
androidboot.fbTileSupport=enable cma=800M@0x960M-0xe00M androidboot.primary_display=imx-drm
firmware_class.path=/vendor/firmware transparent_hugepage=never
U-Boot > saveenv
```

With above settings, the Android platform does not start the shell console. To disable selinux, append "androidboot.selinux=permissive" to the U-Boot's bootargs. Boot environment variables are as follows:

```
U-Boot > setenv append_bootargs androidboot.selinux=permissive
U-Boot > saveenv
```

### 3.4.2 Booting with physical HDMI display

The i.MX 8QuadMax MEK board supports Physical HDMI display.

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

```
U-Boot > setenv bootargs console=ttyLP0,115200 earlycon=lpuart32,0x5a060000,115200 init=/
init androidboot.console=ttyLP0 consoleblank=0 androidboot.hardware=freescale
androidboot.fbTileSupport=enable cma=1184M@0x960M-0xe00M androidboot.primary_display=imx-drm
firmware_class.path=/vendor/firmware transparent_hugepage=never
U-Boot > saveenv
```

With the settings above, the Android platform does not start the shell console. To disable selinux, append "androidboot.selinux=permissive" to the U-Boot's bootargs. Boot environment variables are as follows:

```
U-Boot > setenv append_bootargs androidboot.selinux=permissive
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.

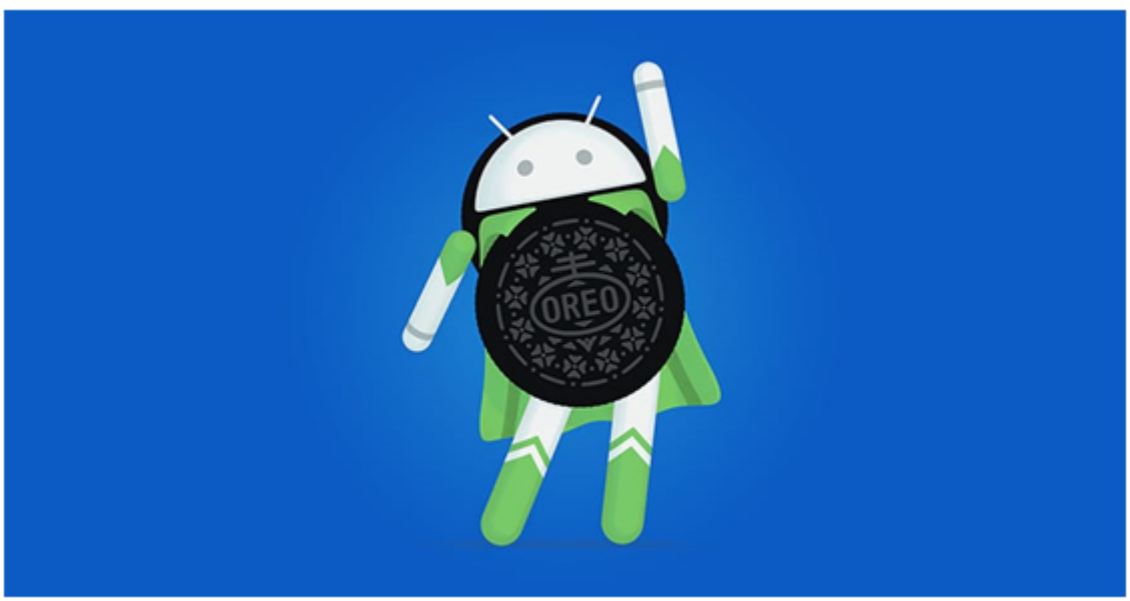


**Figure 6. Android Oreo image**

# 4 Revision History

**Table 3. Revision history**

| Revision number | Date | Substantive changes |
|---|---|---|
| O8.0.0_1.2.0_8QXP-EAR | 12/2017 | Initial release |
| O8.1.0_1.2.0_8QXP-PRC | 03/2018 | i.MX 8QuadXPlus PRC/Beta release |
| O8.1.0_1.2.0_8QXP-beta2 | 08/2018 | i.MX 8QuadXPlus Beta2 release |
| O8.1.0_2.0.0-beta | 01/2019 | i.MX 8QuadXPlus/8QuadMax Beta release |
| O8.1.0_2.0.0-ga | 04/2019 | i.MX 8QuadXPlus/8QuadMax GA release |
| O8.1.0_2.0.1ga | 06/2019 | i.MX 8QuadMax GA release |

***How to Reach Us:***

**Home Page:**
nxp.com

**Web Support:**
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, and UMEMS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2019 NXP B.V.

Document Number AQSUG
Revision O8.1.0_2.0.1-ga, 06/2019