

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:

- i.MX 8QuadMax 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 boards.

Contents

1 Overview.....	1
2 Hardware Requirements.....	1
3 Working with the i.MX 8QuadMax MEK Board.....	1
3.1 Board hardware.....	1
3.2 Board images.....	3
3.3 Flashing board images.....	4
3.4 Booting.....	6
4 Revision History.....	7



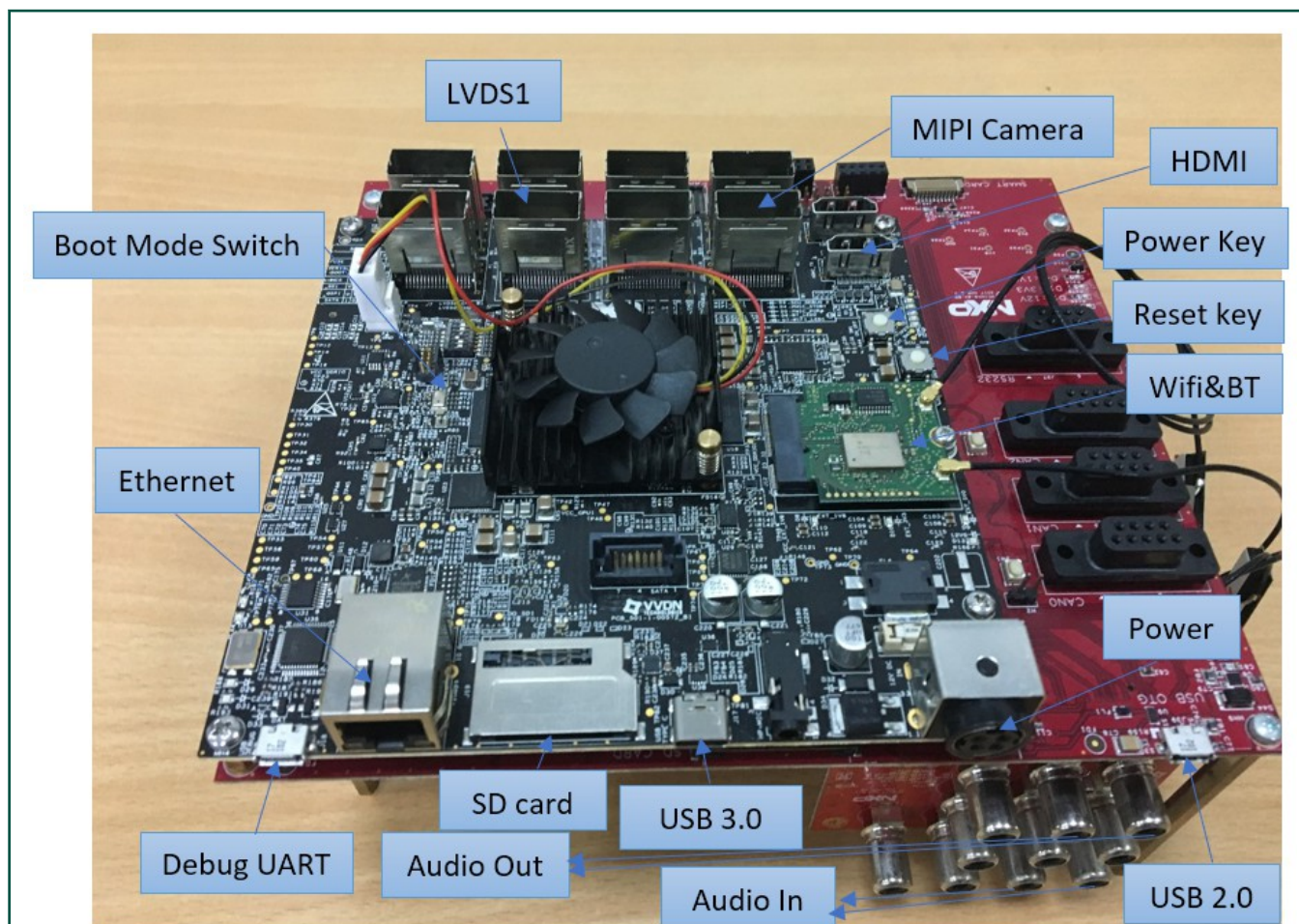


Figure 1. i.MX 8QuadMax MEK board



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

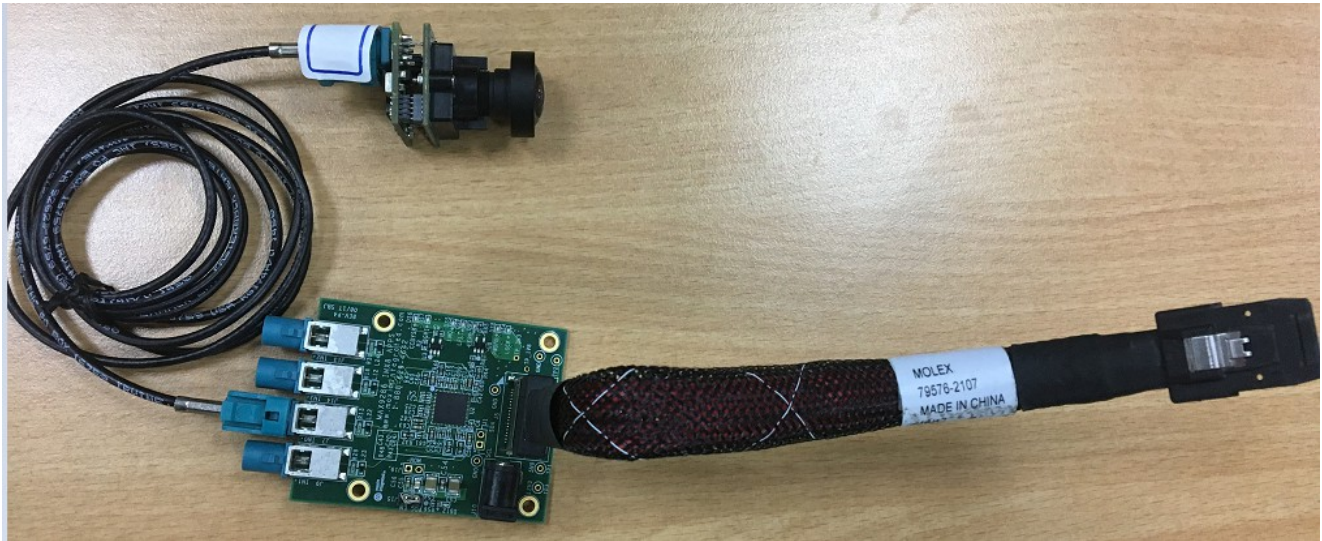


Figure 3. i.MX rearview camera (MAX9286)

NOTE

- To use i.MX rearview camera (MAX9286), connect the two pads of J15.
- i.MX 8QuadMax MEK
 - "LVDS0" is used for Android Automotive running on domain U as display by default.
 - "LVDS1" is used for Linux running on domain 0 as display.
 - To test the rearview camera, connect the "MIPI Camera" port with the i.MX MAX9286 MIPI camera.

3.2 Board images

The table below describes the location in the board partitions of the software images in `automotive-10.0.0_2.4.0_image_8qmek2.tar.gz`.

Table 1. Board images

Image name	Download target
<code>spl-imx8qm.bin</code>	0 KB offset of eMMC boot0 partition for i.MX 8QuadMax MEK.
<code>spl-imx8qm-md.bin</code>	0 KB offset of eMMC boot0 partition for i.MX 8QuadMax MEK.
<code>spl-imx8qm-xen.bin</code>	FAT filesystem on SD card.
<code>bootloader-imx8qm.img</code>	bootloader_a and bootloader_b partitions for i.MX 8QuadMax MEK.
<code>bootloader-imx8qm-md.bin</code>	bootloader_a and bootloader_b partitions for i.MX 8QuadMax MEK.
<code>bootloader-imx8qm-xen.bin</code>	bootloader_a and bootloader_b partitions for i.MX 8QuadMax MEK.
<code>u-boot-imx8qm-xen-dom0.imx</code>	32 KB offset of SD card for i.MX 8QuadMax MEK.
<code>u-boot-imx8qm-mek-uuu.imx</code>	Bootloader used by UUU for i.MX 8QuadMax MEK board. It is not flashed to MMC.
<code>boot.img</code>	boot_a and boot_b partitions to support LVDS-to-HDMI display.
<code>partition-table.img</code>	Program to first 17 KB, and then back up to last 17 KB of the boot storage. GPT table image for 16 GB boot storage.

Table continues on the next page...

Table 1. Board images (continued)

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.
vbmeta-imx8qm-md.img	vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax to support multiple displays.
vbmeta-imx8qm.img	vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax MEK to support LVDS-to-HDMI display.
vbmeta-imx8qm-xen.img	vbmeta_a and vbmeta_b partitions for i.MX 8QuadMax to support XEN.
system.img	system_a and system_b partitions.
vendor.img	vendor_a and vendor_b partitions.
product.img	product_a and product_b partitions.
dtbo-imx8qm.img	dtbo_a and dtbo_b partitions for i.MX 8QuadMax.
dtbo-imx8qm-md.img	dtbo_a and dtbo_b partitions for i.MX 8QuadMax to support multiple displays.
dtbo-imx8qm-xen.img	dtbo_a and dtbo_b partitions for i.MX 8QuadMax to support XEN.
rpmb_key_test.bin	Prebuilt test RPMB key, which can be used to set the RPMB key as fixed 32 bytes 0x00.
testkey_public_rsa4096.bin	Prebuilt AVB public key. It is extracted from the default AVB private key.
xen	XEN firmware to support Trusty OS.

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 1.3.124 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 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 (1-6 bit) to enter serial download mode for i.MX 8QuadMax.
 - To flash images with Xen support, plug an SD card to the SD card slot.
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 is known as debug UART, which can be used to watch the log of hardware boot processing.

4. Decompress release_package/automotive-10.0.0_2.4.0_image_8qmek2.tar.gz, which contains the image files and UUU scripts.
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 Auto images on 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 soc_name	Specifies the SoC information. For i.MX 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.
-u uboot_feature	Flashes U-Boot or SPL&bootloader image with "uboot_feature" in their names.
-d dtb_feature	Flashes dtbo and vbmeta images with "dtb_feature" in their names. For i.MX 8QuadMax MEK in this release, it should be "xen".
-e	Erases user data after images are flashed.
-D 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.
-dryrun	Only generates a UUU script but not executes UUU with this script.

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

```
> sudo ./uuu_imx_android_flash.sh -f imx8qm -u xen -d xen -e -y imx-image-full-  
imx8qmmek.rootfs.sdcard
```

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

```
> .\uuu_imx_android_flash.bat -f imx8qm -u xen -d xen -e -y imx-image-full-  
imx8qmmek.rootfs.sdcard
```

When the command above is executed:

- Android images with Xen support are flashed into both eMMC slot a and slot b for i.MX 8QuadMax MEK.
- All user data are erased.
- imx-image-full-imx8qmmek.rootfs.sdcard is flashed to SD card.

- `u-boot-imx8qm-xen-dom0.imx` is flashed to SD card from 32 KB offset to replace the bootloader in `imx-image-full-imx8qmmek.rootfs.sdcard`.
- `spl-imx8qm-xen.bin` is written to the FAT file system on SD card.
- `xen` is written to the FAT file system on SD card to replace the one in `imx-image-full-imx8qmmek.rootfs.sdcard`.

NOTE

- `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: \\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, you will get information on the command window 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 as SD.

Change SW2 to switch the board back to 001100 (from 1-6 bit) to enter SD boot mode for i.MX 8QuadMax MEK.

3.4 Booting

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

In the U-Boot prompt from the SD card, boot up Dom0 with the following command:

```
=> run xenmmcboot
```

After Dom0 boots up, log in with user ID `root`.

To support Trusty OS and Android Automotive automatical reboot, for the first time to boot to this stage, execute the following commands to modify the configuration files:

```
root@imx8qmmek:~# sed -i '/on_reboot="soft-reset"/d' /etc/xen/domu-imx8qm-mek-androidauto.cfg
root@imx8qmmek:~# sed -i 's/memory = "3584"/memory = "3328"/g' /etc/xen/domu-imx8qm-mek-androidauto.cfg
root@imx8qmmek:~# sed -i -e '/#rtc pl031/,+1d' /etc/xen/domu-imx8qm-mek-androidauto.cfg
root@imx8qmmek:~# echo "tee=\"optee\"" >> /etc/xen/domu-imx8qm-mek-androidauto.cfg
```

Then execute the following command to start domainU and connect to its console.

```
root@imx8qmmek:~# xl create /etc/xen/domu-imx8qm-mek-androidauto.cfg; xl console DomU
```

To disable selinux for userdebug/eng build mode images, usually for debug purposes, stop at the U-Boot prompt from eMMC and append "`androidboot.selinux=permissive`" to the U-Boot's bootargs with the following command:

```
=> setenv append_bootargs androidboot.selinux=permissive
```

To shorten the boot time of Android Auto, some U-Boot commands including "saveenv" are removed. After you set the preceding U-Boot environment variables, execute the following command to boot the system:

```
=> boota
```

Then wait for the Android platform to boot up.

When Android Automotive reboots, its console is not automatically reconnected. Execute the following commands to connect it:

```
root@imx8qmmek:~# xl console DomU
```

4 Revision History

Table 3. Revision history

Revision number	Date	Substantive changes
O8.1.0_1.1.0_AUTO-EAR	02/2018	Initial release
O8.1.0_1.1.0_AUTO-beta	05/2018	i.MX 8QuadXPlus/8QuadMax Beta release
P9.0.0_1.0.2-AUTO-alpha	11/2018	i.MX 8QuadXPlus/8QuadMax Automotive Alpha release
P9.0.0_1.0.2-AUTO-beta	01/2019	i.MX 8QuadXPlus/8QuadMax Automotive Beta release
P9.0.0_2.1.0-AUTO-ga	04/2019	i.MX 8QuadXPlus/8QuadMax Automotive GA release
P9.0.0_2.1.0-AUTO-ga	08/2019	Updated the location of the SCFW porting kit
automotive-10.0.0_1.1.0	03/2020	i.MX 8QuadXPlus/8QuadMax MEK (Silicon Revision B0) GA release
automotive-10.0.0_1.1.0	03/2020	Deleted the Android 10 image
android-10.0.0_2.2.0-AUTO	06/2020	i.MX 8QuadXPlus/8QuadMax MEK GA release
android-10.0.0_2.4.0	07/2020	i.MX 8QuadMax MEK 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, UMEMS, EdgeScale, EdgeLock, eIQ, and Immersive3D 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.

© NXP B.V. 2018-2020.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 15 July 2020

Document identifier: AQSUG

