

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 (Silicon Revision B0)
- i.MX 8QuadXPlus (Silicon Revision B0 and Silicon Revision C0)

Supported boards:

- i.MX 8QuadXPlus/8QuadMax MEK Board and Platform

3 Working with the i.MX 8QuadXPlus/8QuadMax MEK Board

3.1 Board hardware

The figures below show the different components of the i.MX 8QuadXPlus/8QuadMax MEK boards.

Contents

1	Overview.....	1
2	Hardware Requirements.....	1
3	Working with the i.MX 8QuadXPlus/ 8QuadMax MEK Board.....	1
3.1	Board hardware.....	1
3.2	Board images.....	4
3.3	Flashing board images.....	6
3.4	Booting with LVDS-to-HDMI display.....	8
4	Revision History.....	9



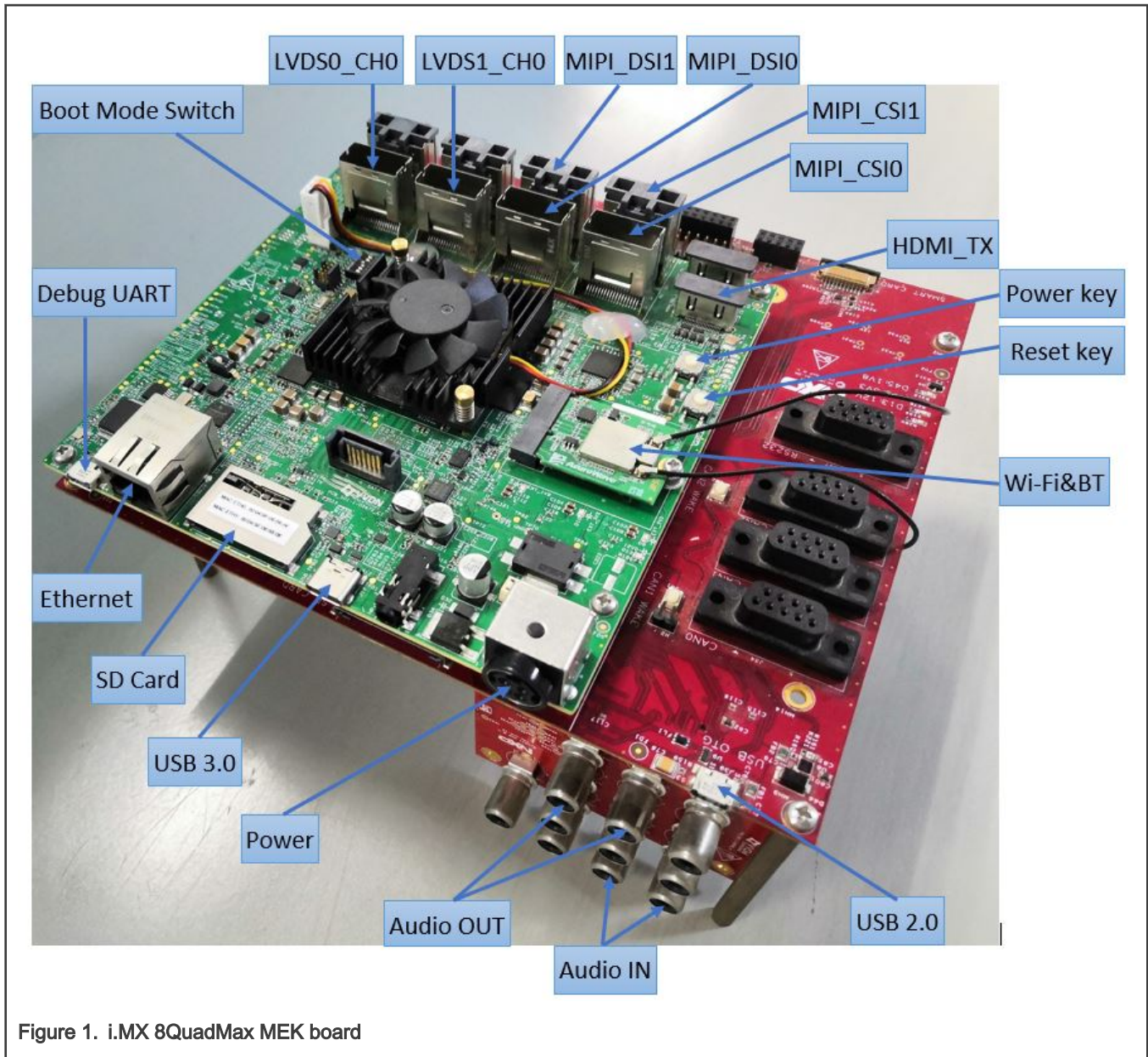


Figure 1. i.MX 8QuadMax MEK board

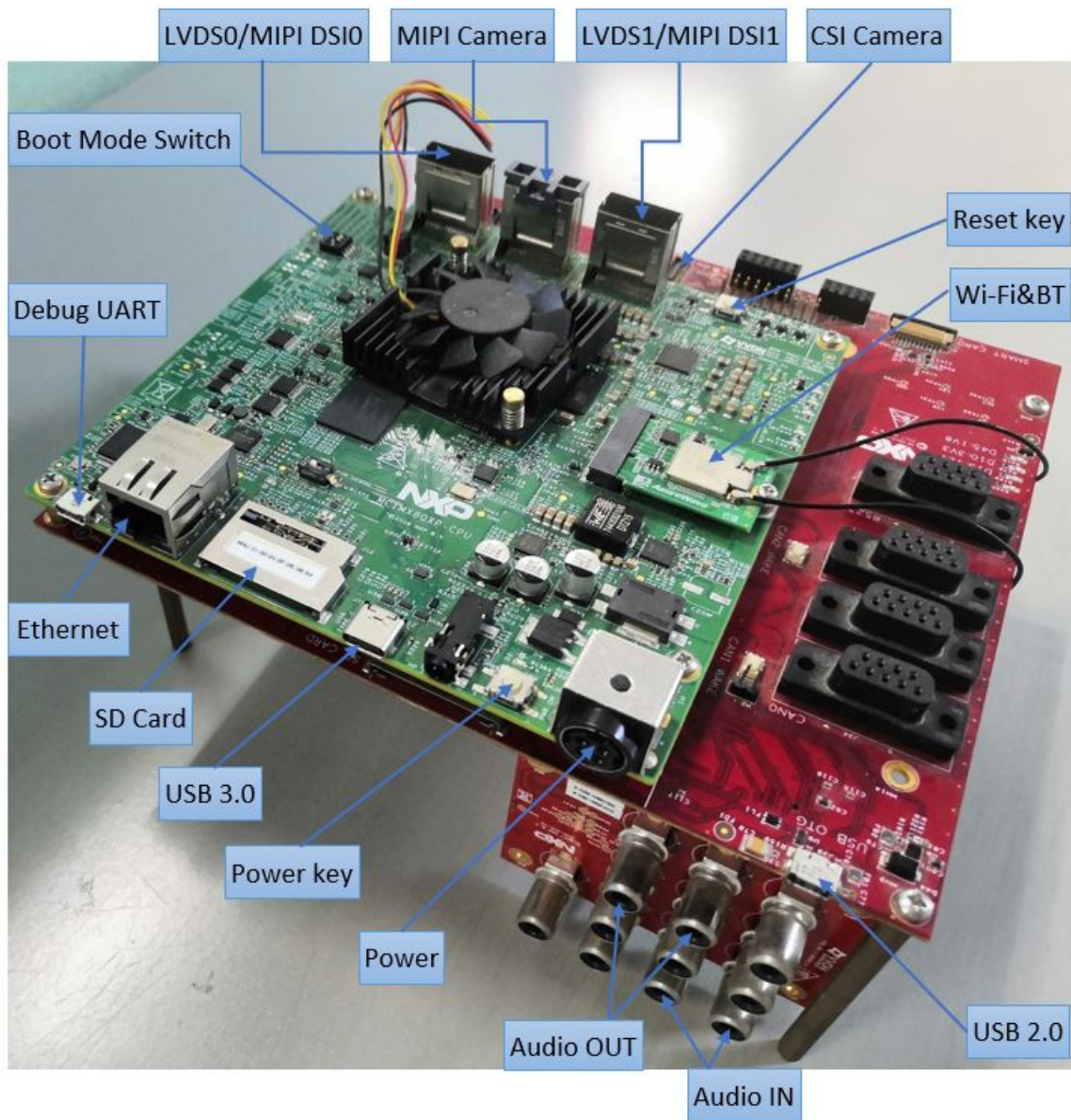


Figure 2. i.MX 8QuadXPlus MEK board



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

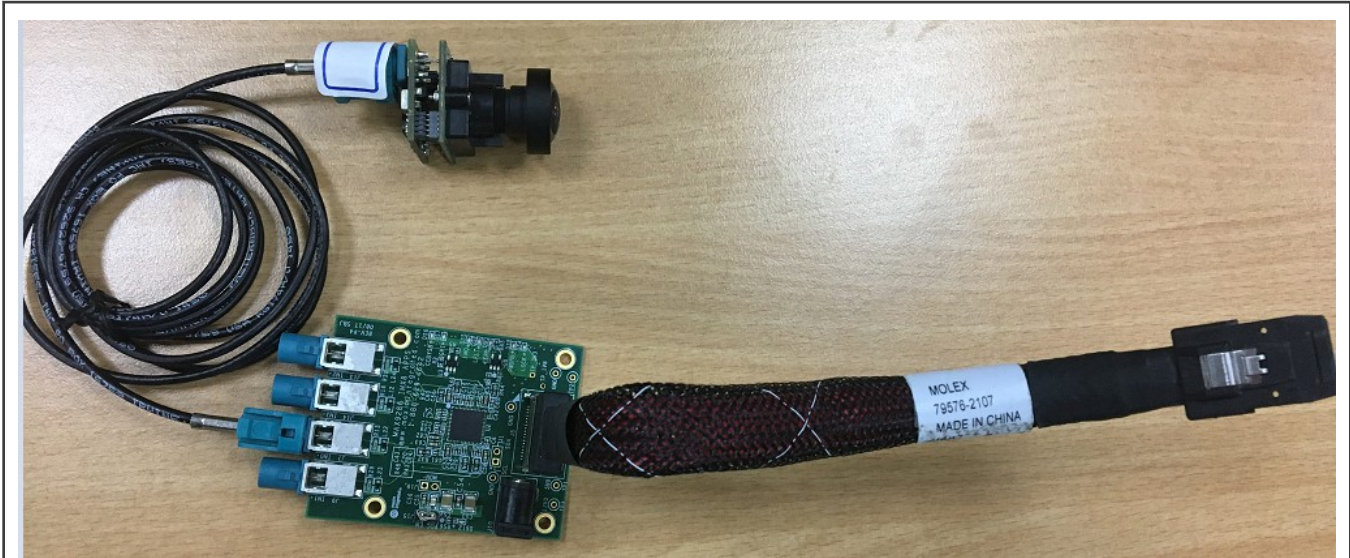


Figure 4. i.MX rearview camera (MAX9286)

NOTE

- To use i.MX rearview camera (MAX9286), connect the two pads of J15 on the camera converter board.
- i.MX 8QuadMax MEK
 - To test the display, connect the "LVDS1_CH0" port to the LVDS-to-HDMI adapter with the i.MX mini SAS cable.
 - To test the rearview camera, connect the "MIPI_CSI0" port with the i.MX MAX9286 MIPI camera.
- i.MX 8QuadXPlus MEK
 - To test the display, connect the "LVDS0" port to the LVDS-to-HDMI adapter with the i.MX mini SAS cable.
 - To test the rearview camera, connect the "MIPI Camera" port with the i.MX MAX9286 MIPI camera.

3.2 Board images

There are two sets of images of automotive-11.0.0_1.1.0_image_8qmek.tar.gz and automotive-11.0.0_1.1.0_image_8qmek2.tar.gz. The following table lists their differences with "8qmek" and "8qmek2" representing these two packages.

Table 1. Image differences

	8qmek	8qmek2
EVS function enabled in Cortex-M core	Y	N
EVS function enabled in Cortex-A core	Y	Y
Support dynamic partition	N	Y
Support updatable apex	N	Y

The table below describes the location in the board partitions of the software images in automotive-11.0.0_1.1.0_image_8qmek.tar.gz.

Table 2. Board images

Image name	Download target
spl-imx8qm.bin	0 KB offset of eMMC boot0 partition for i.MX 8QuadMax MEK.
spl-imx8qm-secure-unlock.bin	0 KB offset of eMMC boot0 partition for i.MX 8QuadMax MEK.
spl-imx8qxp.bin	32 KB offset of eMMC boot0 partition for i.MX 8QuadXPlus MEK with silicon revision B0 chip.
spl-imx8qxp-secure-unlock.bin	32 KB offset of eMMC boot0 partition for i.MX 8QuadXPlus MEK with silicon revision B0 chip.
spl-imx8qxp-c0.bin	32 KB offset of eMMC boot0 partition for i.MX 8QuadXPlus MEK with silicon revision C0 chip.
bootloader-imx8qm-secure-unlock.img	bootloader_a and bootloader_b partitions for i.MX 8QuadMax.
bootloader-imx8qxp.img	bootloader_a and bootloader_b partitions for i.MX 8QuadXPlus MEK with silicon revision B0 chip.
bootloader-imx8qxp-secure-unlock.img	bootloader_a and bootloader_b partitions for i.MX 8QuadXPlus MEK with silicon revision B0 chip.
bootloader-imx8qxp-c0.img	bootloader_a and bootloader_b partitions for i.MX 8QuadXPlus MEK with silicon revision C0 chip.
u-boot-imx8qm-mek-uuu.imx	Bootloader used by UUU for i.MX 8QuadMax MEK board. It is not flashed to MMC.
u-boot-imx8qxp-mek-uuu.imx	Bootloader used by UUU for i.MX 8QuadXPlus MEK board with silicon revision B0 chip. It is not flashed to MMC.
u-boot-imx8qxp-mek-c0-uuu.imx	Bootloader used by UUU for i.MX 8QuadXPlus MEK board with silicon revision C0 chip. It is not flashed to MMC.
boot.img	boot_a and boot_b partitions to support LVDS-to-HDMI display.
partition-table.img	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.
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-imx8qxp.img	vbmeta_a and vbmeta_b partitions for i.MX 8QuadXPlus to support LVDS-to-HDMI display.
system.img	system_a and system_b partitions.
system_ext.img	system_ext_a and system_ext_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.

Table continues on the next page...

Table 2. Board images (continued)

dtbo-imx8qxp.img	dtbo_a and dtbo_b partitions for i.MX 8QuadXPlus.
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.

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.
 - Change the board's SW2 (boot mode) to 1000 (1-4 bit) to enter serial download mode for i.MX 8QuadXPlus.
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/8QuadXPlus 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.
- USB 2.0 is USB Host and USB 3.0 is USB OTG.

4. Decompress release_package/automotive-11.0.0_1.1.0_image_8qmek.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/8QuadXPlus MEK board, related options are described as follows

Table 3. Options for uuu_imx_android_flash tool

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

Table continues on the next page...

Table 3. Options for uuu_imx_android_flash tool (continued)

Option	Description
-f soc_name	Specifies the SoC information. For i.MX 8QuadMax, it should be "imx8qm". For i.MX 8QuadXPlus, it should be "imx8qxp". 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. For i.MX 8QuadMax MEK, it can be "secure-unlock". For i.MX 8QuadXPlus MEK, it can be "c0" or "secure-unlock". If this option is not used, the default "spl-imx8qm.bin" and "bootloader-imx8qm.img" are flashed for i.MX 8QuadMax MEK, and the default "spl-imx8qxp.bin" and "bootloader-imx8qxp.img" are flashed for i.MX 8QuadXPlus MEK.
-d dtb_feature	Flashes dtbo and vbmeta images with "dtb_feature" in their names. For i.MX 8QuadMax MEK, it can be "md". For i.MX 8QuadXPlus MEK, do not use this option. If this option is not used, default "dtbo-imx8qm.img" and "vbmeta-imx8qm.img" are flashed for i.MX 8QuadMax MEK.
-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, it must be followed with an absolute path. 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 -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 -e
```

When the command above is executed, the default images will be flashed into eMMC both slot a and slot b for i.MX 8QuadMax and all user data will be erased.

NOTE

- 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.

- Do not flash Automotive images with "-t sd", because Automotive images can only boot from eMMC.
- To test the demonstration implementation of secure unlock, execute the tool with "-u secure-unlock". For secure unlock details, see the *i.MX Android Security User's Guide (ASUG)*.
- To test multiple displays on i.MX 8QuadMax MEK with images in `automotive-11.0.0_1.1.0_image_8qmek.tar.gz`, execute the tool with "-d md".
- To test multiple displays on i.MX 8QuadMax MEK with images in `automotive-11.0.0_1.1.0_image_8qmek2.tar.gz`, execute the tool with "-u md -d md".

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.
 - Change SW2 to switch the board back to 000100 (1-6 bit) to enter eMMC boot mode for i.MX 8QuadMax.
 - Change SW2 to switch the board back to 0100 (1-4 bit) to enter eMMC boot mode for i.MX 8QuadXPlus.

3.4 Booting with LVDS-to-HDMI display

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

```
U-Boot > setenv bootargs console=ttyLP0,115200 earlycon androidboot.console=ttyLP0
androidboot.xen_boot=default init=/init androidboot.hardware=nxp androidboot.fbTileSupport=enable
cma=800M@0x960M-0xe00M galcore.contiguousSize=33554432 androidboot.primary_display=imx-drm
firmware_class.path=/vendor/firmware loop.max_part=7 androidboot.vendor.sysrq=1
transparent_hugepage=never video=HDMI-A-2:d androidboot.wificountrycode=CN androidboot.lcd_density=240
```

To disable selinux for userdebug/eng build mode images, "androidboot.selinux=permissive" needs to be appended to the U-Boot's bootargs. Boot environment variables are as follows:

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

NOTE

i.MX 8QuadXPlus/8QuadMax MEK supports LVDS-to-HDMI display. They share the same bootargs.

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:

```
U-Boot > boota
```

Then wait for the Android platform to boot up.

4 Revision History

Table 4. 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
android-11.0.0_1.1.0-AUTO	01/2021	i.MX 8QuadXPlus/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.

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.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. 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, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, 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, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, 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, μ Vision, 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. 2018-2021.

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: 25 January 2021

Document Identifier: AQSG

