

emCON-MX8MM Developer Kit for Android 15.0.0

User Manual

© Copyright 2025 **emtrion GmbH**

All rights reserved. This documentation may not be photocopied or recorded on any electronic media without written approval. The information contained in this documentation is subject to change without prior notice. We assume no liability for erroneous information or its consequences. Trademarks used from other companies refer exclusively to the products of those companies.

Revision: Rev. 4 / 28.10.2025

Rev.	Date/Initial	Changes
1	04.11.2019/Ft	Initial Revision
2	29.06.2021/Rr	Changes for Android 11
3	08.07.2025/Rr	Changes for Android 13, Android Studio example removed
4	28.10.2025/Rr	Changes for Android 15

Contents

1	Definitions	4
2	Introduction	5
3	Bootloader	6
3.1	Communication settings	6
3.2	DIP switch setup for emCON-MX8MM	6
3.3	Bootloader prompt	7
3.3.1	Print/change environment variables	7
3.3.2	Network setup	7
3.4	Updating Android Images	8
3.4.1	Prerequisite	8
3.4.2	Updating	8
3.4.2.1	Script for UUU-Tool	9
4	Further information	12

1 Definitions

The table below lists some definitions of terms in this manual.

EMCON-MX8MM	Target platform
AOSP	Android Open Source Project
IDE	integrated development environment
OS	operating system
SoC	System on chip, here the i.MX8MM
UUU-Tool	universal update utility, Freescale/NXP i.MX chip image deploy tools

Table 1.1: *Definitions*

2 Introduction

emtrion has designed this Android developer kit to help you design your Android application quickly and efficiently, and evaluate the hardware.

The version of Android OS running on the developer kit is a custom Android version made by **emtrion**. It is compliant with Android 15.0.0 and bases on the version 15.0.0_2.0.0 from Google[®] with the extensions imx-android-15.0.0_2.0.0 coming from NXP[®]. Additionally some modifications have been added so that it is compliant with the emtrion module emcon-mx8mm.

The process of making or rebuilding the Android OS is substantial and requires a lot of knowledge regarding Linux and Android. If you have any requests or particular needs, **emtrion** can build a custom Android OS for you. Please contact sales@emtrion.de for any questions regarding this point.

3 Bootloader

This section gives a brief description of the bootloader used in this Developer Kit. When you are more interested in the function scope of the bootloader, please refer to the detailed description of the bootloader on the homepage of the U-Boot project: <http://www.denx.de/wiki/U-Boot/>

3.1 Communication settings

The bootloader communicates through the connector J26 on the Avari base board or through the connector J8 (UART A) on the Bvari base board. The communication settings are:

baud rate	115200 bps
data bits	8
stop bits	1
parity	none
handshake	none

Table 3.2: communication settings

3.2 DIP switch setup for emCON-MX8MM

The emCON-MX8MM module and the Avari base board contains DIP switches, which have to be setup as follows for a successful start up of the bootloader.

	1	2	3	4
Boot device eMMC	off	off	on	off

Table 3.4: DIP switches on emCON-MX8 module

	1	2	3	4	5	6	7	8
normal boot from on board device	on	off	on	don't care	don't care	don't care	don't care	don't care
force serial downloader mode	off	on	on	don't care	don't care	don't care	don't care	don't care

Table 3.6: DIP switches SW1 on Avari base board

3.3 Bootloader prompt

The bootloader prompt is reached if you press a key in the console window when the boot delay is counted down. The bootloader prompt allows you to change settings of the bootloader or to update the Android image in the eMMC flash.

We are using the U-Boot provided from NXP with some modifications and extensions related to our hardware. The U-Boot provided by NXP is a mainline U-Boot extended with code to support the i.MX8MM SoC. A detailed description of the bootloader can be found on the homepage of the U-Boot project: <http://www.denx.de/wiki/U-Boot/>.

3.3.1 Print/change environment variables

The environment variables are handled by using three commands: *printenv*, *setenv*, *saveenv*. *printenv* shows you the current setting of all environment variables. *printenv <variable>* shows only the current value of an environment variable. *setenv <variable> <value>* changes the value of an environment variable. This change is only in RAM and will be lost after reset. The changes can be made permanent by using *saveenv*. The following example shows how the boot command is set up.

```
1 emCON-MX8MM U-Boot > setenv bootcmd 'run bootcmd_android'
2 emCON-MX8MM U-Boot > saveenv
```

Listing 3.1:

3.3.2 Network setup

The network setup of the bootloader is also handled by environment variables:

variable	description
autoload	Set this to "no". This prevents that the use of the "dhcp" command automatically starts a tftp download
ipaddr	IP address of the device. Only effective if dhcp is disabled
serverip	IP address of the host PC which acts as TFTP server
netmask	Subnet mask of the device
ip-method	Set this to "static" or "dhcp" according to your setup. This is used by the update_uboot script

Table 3.8: environment variables related to network setup

If you have a DHCP server in your network and want to configure the device via dhcp simple use the command "dhcp":

```
1 emCON-MX8MM U-Boot > dhcp
2 BOOTP broadcast 1
3 DHCP client bound to address 172.26.1.10 (6ms)
```

Listing 3.2:

If there is no DHCP server you have to set the variables "*ipaddr*" and "*netmask*" by hand.

To test your network settings you can ping the host PC from the device running the bootloader. To do so use the command *ping <ip address>*. **Please note, that the device running the bootloader cannot be pinged.**

3.4 Updating Android Images

3.4.1 Prerequisite

To update the Android on the target system you need the Universal Update Utility (UUU-Tool) on your host PC. This utility is the Freescale/NXP i.MX chip image deploy tool. You can download this utility from this URL <https://github.com/NXPmicro/mfgtools/releases> (click on the revision number to see the download links).

3.4.2 Updating

For the update process you need to connect the emCON-MX8MM device as a USB device to your host PC. You can use the connector J6 on the Avari baseboard. You also need the serial communication between bootloader and host PC, so you also need to connect the serial cable (see chapter 3.1 for settings.)

On the host PC start the UUU-Tool using

```
1 .\uuu.exe <script file without path>
```

Listing 3.3:

Before powering up the device, set the DIP switches SW1 on the Avari base board to the state *force serial downloader mode* as shown below.

	1	2	3	4	5	6	7	8
force serial downloader mode	off	on	on	don't care	don't care	don't care	don't care	don't care

Table 3.10: DIP switches SW1 on Avari base board

After powering up the device, the fastboot mechanism which connects to the UUU-Tool on the host PC through the USB device connection is started automatically. You see the process in the UUU-Tool output and in the output of the bootloader on the serial line. When you see *Done* in the UUU-Tool or the message *Update finished successfully* when you have used the script listed in section 3.4.2.1 below.

Power off the device and move back the DIP switches SW1 to *normal boot from on board device* as shown below.

	1	2	3	4	5	6	7	8
normal boot from on board device	on	off	on	don't care	don't care	don't care	don't care	don't care

Table 3.12: DIP switches SW1 on Avari base board

After powering on the device, the boot sequence with the updated bootloader and the updated Android BSP is visible. The first boot usually takes a longer time because several partitions are formatted and the user partition is filled initially.

3.4.2.1 Script for UUU-Tool

The following script can be used to update the Android stored on the emCON-MX8MM device. You have to replace the part *<path to image directory>* with the path to the directory where you have stored the new image files on your host PC.

```

1  uuu_version 1.5.201
2
3  # This command will be run when i.MX6/7 i.MX8MM, i.MX8MQ
4  SDP: boot -f < path to image directory >/flash.bin
5
6
7  # This command will be run when ROM support stream mode
8  # i.MX8QXP, i.MX8QM
9  SDPS: boot -f < path to image directory >/flash.bin
10
11 # These commands will be run when use SPL and will be skipped if no spl
12 # SDPU will be deprecated. please use SDPV instead of SDPU
13 # {
14     SDPU: delay 1000
15     SDPU: write -f < path to image directory >/flash.bin -offset 0x57c00
16     SDPU: jump
17     # }
18
19 # These commands will be run when use SPL and will be skipped if no spl
20 # if (SPL support SDPV)
21 # {
22     SDPV: delay 1000
23     SDPV: write -f < path to image directory >/flash.bin -skipspl
24     SDPV: jump
25     # }
26
27 FB: delay 1000
28 FB: ucmd echo
29     "*****"
30 FB: ucmd echo "* Preparing environment for updating stored U-Boot"
31 FB: ucmd printenv emmc_dev
32 FB: ucmd setenv fastboot_dev mmc
33 FB: ucmd setenv emmc_dev 0
34 FB: ucmd setenv fastboot_buffer ${loadaddr}
35 #FB: ucmd setenv mmcdev ${emmc_dev}
36 FB: ucmd mmc dev
37 FB: ucmd mmc info
38 FB: ucmd echo
39     "*****"
40 FB: ucmd echo "* Store U-Boot in Flash"
41 FB: ucmd mmc dev ${emmc_dev} 1
42 FB: ucmd mmc dev
43 FB: ucmd mmc info
44 FB: ucmd mmc bootpart-resize 0:0 16 16
45 FB: flash bootloader0 < path to image directory >/flash.bin

```

```

44 FB: ucmd mmc dev
45 FB: ucmd mmc dev ${emmc_dev} 0
46 FB: ucmd echo
    "*****"
47 FB: ucmd echo "* Setup eMMC boot partition for booting"
48 FB: ucmd mmc partconf ${emmc_dev} 0 1 0
49 FB: ucmd mmc partconf ${emmc_dev}
50 FB: ucmd echo
    "*****"
51 FB: ucmd echo "* Destroy stored U-Boot environment"
52 FB: ucmd mw.b ${fastboot_buffer} 0 0x20000
53 FB: ucmd md.b ${fastboot_buffer} 0x100
54 FB: ucmd mmc dev ${emmc_dev} 2
55 FB: ucmd printenv fastboot_buffer
56 FB: ucmd mmc write ${fastboot_buffer} 0 0x100
57 FB: ucmd echo
    "*****"
58 FB: ucmd echo "* Cleanup eMMC user partition"
59 FB: ucmd mmc erase 0 0x800
60 FB: ucmd echo
    "*****"
61 FB: ucmd echo "* Writing Android"
62 FB: ucmd setenv fastboot_dev mmc
63 FB: ucmd setenv mmcdev 0
64 FB: ucmd mmc dev ${mmcdev}
65 FB: ucmd mmc dev ${mmcdev} 0
66 FB: ucmd mmc info
67
68 FB[-t 600000]: flash gpt < path to image directory >/partition-table.img
69 FB: ucmd mmc part
70
71 FB[-t 600000]: ucmd setenv fastboot_dev mmc
72
73 FB[-t 600000]: flash init_boot_a < path to image directory >/init_boot.img
74 FB[-t 600000]: flash init_boot_b < path to image directory >/init_boot.img
75
76 FB[-t 600000]: flash dtbo_a < path to image directory >/dtbo-imx8mm.img
77 FB[-t 600000]: flash dtbo_b < path to image directory >/dtbo-imx8mm.img
78
79 FB[-t 600000]: flash boot_a < path to image directory >/boot-imx.img
80 FB[-t 600000]: flash boot_b < path to image directory >/boot-imx.img
81
82 FB[-t 600000]: flash vendor_boot_a < path to image directory >/vendor_boot.img
83 FB[-t 600000]: flash vendor_boot_b < path to image directory >/vendor_boot.img
84
85 FB[-t 600000]: flash super < path to image directory >/super.img
86 #
87 FB[-t 600000]: erase misc
88 FB[-t 600000]: erase persistdata
89 FB[-t 600000]: erase fbmisc
90 FB[-t 600000]: erase metadata

```

```
91 FB[-t 2400000]: erase userdata
92 #
93 FB[-t 600000]: flash vbmeta_a < path to image directory >/vbmeta.img --disable-
    verify --disable-verification
94 FB[-t 600000]: flash vbmeta_b < path to image directory >/vbmeta.img --disable-
    verify --disable-verification
95 #
96 FB: ucmd echo
    "*****"
97 FB: ucmd echo "* Production finished successfully"
98 FB: Done
```

Listing 3.4: UUU script for Android update

4 Further information

Online resources

Further information can be found on the Emtrion support pages.

<https://support.emtrion.de>

We support you

Emtrion offers different kinds of services, among them Support, Training and Engineering. Contact us at sales@emtrion.de if you need information or technical support.