How to program and access fuses on Embedded Artists i.MX based boards

Embedded Artists AB
Jörgen Ankersgatan 12
SE-211 45 Malmö
Sweden

http://www.EmbeddedArtists.com

Disclaimer

Embedded Artists AB makes no representation or warranties with respect to the contents hereof and specifically disclaim any implied warranties or merchantability or fitness for any particular purpose. Information in this publication is subject to change without notice and does not represent a commitment on the part of Embedded Artists AB.

Feedback

We appreciate any feedback you may have for improvements on this document. Send your comments by using the contact form: www.embeddedartists.com/contact.

Trademarks

All brand and product names mentioned herein are trademarks, services marks, registered trademarks, or registered service marks of their respective owners and should be treated as such.

Table of Contents

1 Document Revision History

Revision	*Date*	*Description*
A	2016-09-06	First release
B	2020-03-18	Added chapters 6 – *Boot from eMMC* and chapter 7 – *Program Fuses from UUU*.

2 Introduction

The i.MX processors have One Time Programmable (OTP) fuses which are used to store different kinds of permanent configuration settings. This could, for example, be boot configuration, MAC address, secure boot settings, and so on.

This document describes how to access and program these fuses from within the u-boot bootloader and Linux.

2.1 Conventions in the document

A number of conventions have been used throughout to help the reader better understand the content of the document.

`Constant width text` – is used for file system paths and command, utility and tool names.

```
$ This field illustrates user input in a terminal running on the
development workstation, i.e., on the workstation where you edit,
configure and build Linux
```

```
# This field illustrates user input on the target hardware, i.e.,
input given to the terminal attached to the COM Board
```

```
This field is used to illustrate example code or excerpt from a
document.
```

This field is used to highlight important information

3 Fuses

3.1 Organization

On the i.MX processor the fuses are normally organized in a number of banks where each bank consists of a number of words and each word is typically 32 bits. Each bit can be a one-time-programmable fuse.

As an example, the i.MX7 Dual processor has 16 pieces of 128-bit wide banks with four words in each bank.

NOTE: Always refer to the User's Manual for the processor you are using to get the exact description of the fuses on that specific processor.

3.2 How to get bank and word index

If you need to determine which bank and word index a specific fuse is located at the easiest is to look at the OCOTP memory map table in the User's Manual for the processor you are working with.

As an example, look at Figure 1 and see that OCOTP_LOCK is located at Bank0 Word0.

3035_0400	Value of OTP Bank0 Word0 (Lock controls) (OCOTP_LOCK)	32	R/W	0000_0000h	6.4.5.17/ 1131

Table continues on the next page...

i.MX 7Dual Applications Processor Reference Manual, Rev. 1, 01/2018

1116 NXP Semiconductors

Chapter 6 SNVS, Reset, Fuse and Boot

OCOTP memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
3035_0410	Value of OTP Bank0 Word1 (Tester Information) (OCOTP_TESTER0)	32	R/W	0000_0000h	6.4.5.18/ 1133
3035_0420	Value of OTP Bank0 Word2 (Tester Information) (OCOTP_TESTER1)	32	R/W	0000_0000h	6.4.5.19/ 1133
3035_0430	Value of OTP Bank0 Word3 (Tester Information) (OCOTP_TESTER2)	32	R/W	0000_0000h	6.4.5.20/ 1134
3035_0440	Value of OTP Bank1 Word0 (Tester Information) (OCOTP_TESTER3)	32	R/W	0000_0000h	6.4.5.21/ 1134
3035_0450	Value of OTP Bank1 Word1 (Tester Information) (OCOTP_TESTER4)	32	R/W	0000_0000h	6.4.5.22/ 1135
3035_0460	Value of OTP Bank1 Word2 (Tester Information) (OCOTP_TESTER5)	32	R/W	0000_0000h	6.4.5.23/ 1135
3035_0470	Value of OTP Bank1 Word3 (Boot Configuration Information) (OCOTP_BOOT_CFG0)	32	R/W	0000_0000h	6.4.5.24/ 1136
3035_0480	Value of OTP Bank2 Word0 (Boot Configuration Information) (OCOTP_BOOT_CFG1)	32	R/W	0000_0000h	6.4.5.25/ 1136
3035_0490	Value of OTP Bank2 Word1 (Boot Configuration Information) (OCOTP_BOOT_CFG2)	32	R/W	0000_0000h	6.4.5.26/ 1137

Figure 1 OCOTP memory map for iMX7 Dual

The table in Figure 1 also specifies the absolute address of the register which for OCOTP_LOCK is 0x30350400. The last part of this address (0x400) is the fuse address usually listed in the Fusemap table in the User's manual, see Figure 2.

Table 6-19. Fusemap Descriptions

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Setting	Used by
0x400[1:0]	TESTER_LOCK	2	Lock for tester related fuses at 0x400-0x460.	00 - Unlock 10 - OP 01 - WP 11 - OP + WP	Programmed to 11 by NXP during device manufacturing.
0x400[3:2]	BOOT_CFG_LOCK	2	Lock for BOOT related fuses at 0x470-4B0.	00 - Unlock 10 - OP 01 - WP 11 - OP + WP	OCOTP
0x400[9]	SRK_LOCK	1	Locking SRK_HASH[255:0]	0 - Unlock 1 - OP + WP	OCOTP
0x400[10]	SJC_RESP_LOCK	1	Lock for SJC_RESP[55:0] fuses.	0 - Unlock 1 - Lock	OCOTP
0x400[13:12]	USB_ID_LOCK	2	Lock for USB_PID and USB_VID fuses.	00 - Unlock 10 - OP 01 - WP 11 - OP + WP	OCOTP
0x400[15:14]	MAC_ADDR_LOCK	2	Lock MAC_ADDR fuses.	00 - Unlock 10 - OP 01 - WP 11 - OP + WP	OCOTP
0x400[21:20]	GP1_LOCK	2	Lock for GP1[63:0] fuses.	00 - Unlock 10 - OP	OCOTP

Table continues on the next page...

Figure 2 - Fusemap table for iMX7 Dual

3.3 User's Manual reference: OCOTP memory map

The table below contains the section number in the User's Manual to the OCOTP memory map for a specific processor. The revision of the manual is given so if you are using a different revision the section number can also be different.

Processor	Manual revision	Section
i.MX6 Dual/Quad	Rev 5	46.5
i.MX6 SoloX	Rev 3	44.5
i.MX6 UltraLite	Rev 2	35.5
i.MX7 Dual	Rev 1	6.4.5
i.MX8 M	Rev 2	6.3.5
i.MX8 M Mini	Rev 2	6.3.4
i.MX8 M Nano	Rev 0	6.3.4
i.MX7ULP	Rev 0	34.7.1.1

3.4 User's Manual reference: Fusemap Descriptions table

The table below contains the section number in the User's Manual to the Fusemap Descriptions table for a specific processor. The revision of the manual is given so if you are using a different revision the section number can also be different.

Processor	Manual revision	Section
i.MX6 Dual/Quad	Rev 5	5.2
i.MX6 SoloX	Rev 3	5.3
i.MX6 UltraLite	Rev 2	5.3
i.MX7 Dual	Rev 1	6.3.3
i.MX8 M	Rev 2	6.2.3
i.MX8 M Mini	Rev 2	6.2.3
i.MX8 M Nano	Rev 0	6.2.3
i.MX7ULP	Rev 0	Not available in the User's Manual at the time of writing this document. Must be requested from NXP.

4 Examples in Linux

NOTE: Not all kernel versions support OCOTP access. The recommended way of accessing the OTP fuses are via u-boot as described in chapter 5

4.1 OCOTP register mapped to sysfs

In Linux it is quite easy to access the fuses since the OCOTP registers are mapped as files in sysfs, more specifically in the directory `/sys/fsl_otp`. All the files/registers have the prefix HW_OCOTP_ followed by the register name. As an example, the register OCOTP_LOCK described in section 3.2 has the name HW_OCOTP_LOCK in the file system.

Get value of a register:

```
# cat /sys/fsl_otp/HW_OCOTP_LOCK
0xa0030103
```

Write fuses in a register:

```
# echo 0x11223344 > HW_OCOTP_YOUR_REGISTER
```

NOTE: Double or even triple check the value written to a register because once written it cannot be undone.

4.2 Example: iMX7 Dual – Get silicon revision

In the Fusemap description table in the User's manual (section 6.3.3 in Rev 1 of the manual) it is specified that the silicon revision is available at address 0x440[3:0], that is, bits 0 to 3 at address 0x440. In the OCOTP memory map table (section 6.4.5 in rev 1) absolute address 0x30350**440** is called OCOTP_TESTER3 and located at bank 1 word 0.

```
# cat /sys/fsl_otp/HW_OCOTP_TESTER3
0x20000200
```

4.3 Example: iMX6 SoloX – Get speed grading

In the Fusemap description table in the User's manual (section 5.3 in Rev 3 of the manual) it is specified that speed grading is available at address 0x440[17:16], that is, bits 16 and 17 at address 0x440. In the OCOTP memory map table (section 44.5 in rev 3 manual) absolute address 0x021BC**440** is called OCOTP_CFG3 and located at bank 0 word 4.

```
# cat /sys/fsl_otp/HW_OCOTP_CFG3
0x420002
```

This means that the binary value of bits 16 and 17 is 10b. According to the manual this means that the speed grade is 1000 MHz

5 Examples in U-boot

5.1 Fuse command

In the u-boot the fuse command is available that let you read and write fuses. Bank and word index must be specified when accessing the fuses.

Read fuses:

```
=> fuse read 0 0
```

Write fuses (bank and word should be changed to the bank index and word index):

```
=> fuse prog bank word 0x11223344
```

NOTE: Double or even triple check the value written to a register because once written it cannot be undone.

5.2 Example: iMX7 Dual – Get silicon revision

In the Fusemap description table in the User's manual (section 6.3.3 in Rev 1 of the manual) it is specified that the silicon revision is available at address 0x440[3:0], that is, bits 0 to 3 at address 0x440. In the OCOTP memory map table (section 6.4.5 in rev 1) absolute address 0x30350**440** is called OCOTP_TESTER3 and located at bank 1 word 0.

```
=> fuse read 1 0
Word 0x00000000: 20000200
```

5.3 Example: iMX6 SoloX – Get speed grading

In the Fusemap description table in the User's manual (section 5.3 in Rev 3 of the manual) it is specified that the speed grading is available at address 0x440[17:16], that is, bits 16 and 17 at address 0x440. In the OCOTP memory map table (section 44.5 in rev 3 manual) absolute address 0x021BC**440** is called OCOTP_CFG3 and located at bank 0 word 4.

```
=> fuse read 0 4
Word 0x00000004: 00420002
```

This means that the binary value of bits 16 and 17 is 10b. According to the manual this means that the speed grade is 1000 MHz.

6 Boot from eMMC

An i.MX application processor can boot its software from different boot devices such as eMMC, SD-card, NAND-flash, and so on. Which device to boot from can be selected by for example boot mode pins or by programming fuses. On Embedded Artists COM boards boot mode pins are used by default and these have been setup to use eMMC as boot device.

There can be situations where using boot mode pins can cause problems, for example, if you need to externally drive these pins during reset or if you need to use these pins for other purposes. In this case it is better to program the on-chip OTP fuses for eMMC boot. Please note that you can program the fuses to boot from a different device as well, but the instructions in this chapter only describe booting from eMMC.

6.1 Supported COM boards

Booting from eMMC via fuses is not supported by all COM boards or all revisions of a COM board. The table below lists the COM boards (and from which revision) this feature is supported.

COM Board	Board revision
iMX6 SoloX COM	Rev B2
iMX6 UltraLite COM	Rev B
iMX7 Dual COM	Rev B
iMX8MQ COM	All revisions
iMX8M Mini uCOM	All revisions
iMX8M Nano uCOM	All revisions
iMX7ULP uCOM	All revisions

6.2 COM Carrier Board v2

You can test eMMC booting via fuses using a COM Carrier Board v2, rev E or later. Open (remove) jumper J27, highlighted in Figure 3, and you will boot from fuses. Please note that if fuses haven't been programmed the board will default to USB OTG download (serial download) if J27 is removed.

If you have an older revision of the carrier board you won't be able to easily test booting via fuses. The reason is that on the older revisions the BOOT_CTRL pin is always grounded.

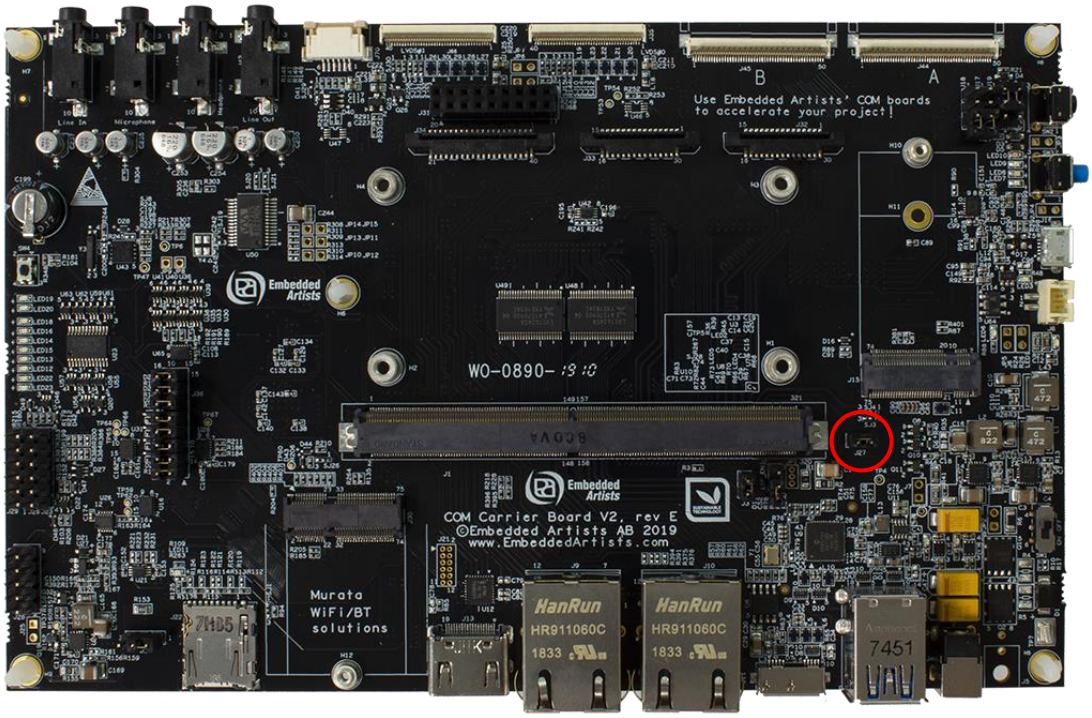

Figure 3 - COM Carrier board v2

6.3 iMX6 SoloX COM

Section 5.1 (Boot Fusemap) in the i.MX6 SoloX User's Manual contains the table shown in Figure 4 below. In this table you can see that for MMC/eMMC boot BOOT_CFG[6] and BOOT_CFG[5] must be set to 1 while BOOT_CFG[7] must be 0.

Table 5-1. Boot Device Select

Boot Device	BOOT_CFG1[7]	BOOT_CFG1[6]	BOOT_CFG1[5]	BOOT_CFG1[4]	BOOT_CFG1[3]
QSPI	0	0	0	1	QXIP Instance 0: QuadSPI0 1: QuadSPI1
EIM	0	0	0	0	Memory Type: 0 - NOR Flash 1 - OneNAND
Serial-ROM	0	0	1	1	x
SD/eSD	0	1	0	x	x
MMC/eMMC	0	1	1	x	x
NAND	1	x	x	x	x

Figure 4 - Boot Device Select for i.MX6 SoloX (Table 5-1)

There is a more detailed fusemap table specifically for MMC/eMMC booting, see Figure 5. For the iMX6 SoloX COM board the eMMC is on the eSDHC3 interface. This means that Port Select must be set to 10, that is, bit 12 must be set to 1.

Table 5-6. MMC/eMMC Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	1	1	Fast Boot: 0 - Regular 1 - Fast Boot	SD/MMC Speed 0 - High 1- Normal	Fast Boot Acknowledge Disable: 0 - Boot Ack Enabled 1 - Boot Ack Disabled	SD Power Cycle Enable 0 - No power cycle 1 - Enabled via USDHC_RST pad (uSDHC3 & 4 only)	SD Loopback Clock Source Sel (for SDR50 and SDR104 only) 0 - through SD pad 1 - direct
0x450[15:8] (BOOT_CFG2)	Bus Width: 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - Reserved			Port Select: 00 - eSDHC1 01 - eSDHC2 10 - eSDHC3 (eMMC4.4) 11 - eSDHC4		Boot Frequencies (ARM/DDR) 0 - 792 / 400 MHz 1 - 528 / 307 MHz	SD3 VOLTAGE SELECTION 0 - 3.3V 1 - 1.8V	Reserved

Figure 5 - MMC/eMMC Boot Fusemap for i.MX6 SoloX (Table 5-6)

The *Fuse Description Table* from section 5.3 in the i.MX6 SoloX User's Manual gives that BOOT_CFG is at fuse address **0x450**, see Figure 6. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses. This fuse is available at address **0x460**.

0x440[31:18]	Reserved	14	-	-	-
0x450[7:0]	BOOT_CFG1	8	BOOT configuration register #1, Usage varies, depending on selected boot device.	0x0000XXXX - WEIM (NOR/ OneNAND) boot. 0x0011XXXX - Serial ROM (I2C/SPI) boot. 0x1XXXXXXX - NAND FLASH boot. 0x010XXXXX -	SRC
0x460[4]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development) 0=Boot mode configuration is taken from GPIOs. 1=Boot mode configuration is taken from fuses. If boot_mode="10" (Production). 0 - Boot using Serila Loader (USB). 1- Boot mode configuration is taken from fuses.	SRC SW(ROM)

Figure 6 - Fusemap Description Table for i.MX6 SoloX (Table 5-9)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x450 and 0x460 is mapped to. As shown in Figure 7 below this is **Bank0 Word5** and **Bank0 Word6**.

OCOTP memory map (continued)

Absolute address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
21B_C450	Value of OTP Bank0 Word5 (Configuration and Manufacturing Info.) (OCOTP_CFG4)	32	R/W	0000_0000h	44.5.14/ 2814
21B_C460	Value of OTP Bank0 Word6 (Configuration and Manufacturing Info.) (OCOTP_CFG5)	32	R/W	0000_0000h	44.5.15/ 2814
21B_C470	Value of OTP Bank0 Word7 (Configuration and Manufacturing Info.) (OCOTP_CFG6)	32	R/W	0000_0000h	44.5.16/ 2815
	Value of OTP Bank1 Word0 (Memory Related Info.)				44.5.17/

Figure 7 - OCOTP memory map for i.MX6 SoloX

With this information we can now determine that bank0 and word5 should be programmed with a value where bit 5, bit 6 and bit 12 (BOOT_CFG) are set to 1. The hexadecimal value that corresponds to these bits being set to 1 is 0x00001060.

For bank0 word6, bit 4 (BT_FUSE_SEL) should be set to 1. The hexadecimal value is 0x00000010.

6.3.1 Instructions

From the **u-boot console** run the following commands to program the fuses:

NOTE: These commands cannot be undone.

```
=> fuse prog 0 5 00001060
=> fuse prog 0 6 00000010
```

6.4 iMX6 UltraLite COM

Section 5.1 (Boot Fusemap) in the i.MX6 UltraLite User's Manual contains the table shown in Figure 8 below. In this table you can see that for MMC/eMMC boot, BOOT_CFG[6] and BOOT_CFG[5] must be set to 1 while BOOT_CFG[7] must be 0.

Table 5-1. Boot Device Select (continued)

Boot Device	BOOT_CFG1[7]	BOOT_CFG1[6]	BOOT_CFG1[5]	BOOT_CFG1[4]	BOOT_CFG1[3]	BOOT_CFG1[2]	BOOT_CFG1[1]	BOOT_CFG1[0]
MMC/eMMC	0	1	1	Fast boot: 0 - Regular 1 - Fast boot	SD/MMC speed: 0 - Normal 1 - High	Fast boot acknowledge disable: 0 - Boot ack enabled 1 - Boot ack disabled	SD power cycle enable: 0 - No power cycle 1 - Enabled via USDHC_RST pad (uSDHC3 and 4 only)	SD loopback clock source sel (for SDR50 and SDR104 only): 0 - Through SD pad 1 - Direct
NAND	1	BT_TOGGLEMODE	Pages in block: 00 - 128 01 - 64 10 - 32 11 - 256	Pages in block: 00 - 128 01 - 64 10 - 32 11 - 256	Nand number of devices: 00 - 1 01 - 2 10 - 4 11 - Reserved	Nand number of devices: 00 - 1 01 - 2 10 - 4 11 - Reserved	Nand_Row_address_bytes: 00 - 3 01 - 2 10 - 4 11 - 5	Nand_Row_address_bytes: 00 - 3 01 - 2 10 - 4 11 - 5

Figure 8 - Boot Device Select for i.MX6 UltraLite (Table 5-1)

There is a more detailed fusemap table specifically for MMC/eMMC booting, see Figure 9. For the iMX6 UltraLite COM board the eMMC is on the eSDHC2 interface. This means that Port Select must be set to 01, that is, bit 11 must be set to 1.

Addr	7	6	5	4	3	2	1	0
0x450[7:0] (BOOT_CFG1)	0	1	1	Fast Boot: 0 - Regular 1 - Fast Boot	SD/MMC Speed 0 - High 1- Normal	Fast Boot Acknowledge Disable: 0 - Boot Ack Enabled 1 - Boot Ack Disabled	SD Power Cycle Enable 0 - No power cycle 1 - Enabled via USDHC_RST pad	SD Loopback Clock Source Sel (for SDR50 and SDR104 only) 0 - through SD pad 1 - direct
0x450[15:8] (BOOT_CFG2)	Bus Width: 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - Reserved			Port Select: 00 - eSDHC1 01 - eSDHC2 10 - Reserved 11 - Reserved		Boot Frequencies (ARM/DDR) 0 - 500 / 400 MHz 1 - 250 / 200 MHz	SD VOLTAGE SELECTION 0 - 3.3V 1 - 1.8V	Reserved

Figure 9 - MMC/eMMC Boot Fusemap for i.MX6 UltraLite (Table 5-6)

The *Fuse Description Table* from section 5.3 in the i.MX6 UltraLite User's Manual gives that BOOT_CFG is at fuse address **0x450**, see Figure 10. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses. This fuse is available at address **0x460**.

0x450[7:0]	BOOT_CFG1	8	BOOT configuration register #1, Usage varies, depending on selected boot device.	0x0000XXXX - WEIM (NOR/ OneNAND) boot 0x0011XXXX - Serial ROM (I2C/SPI) boot 0x1XXXXXXX - NAND FLASH boot 0x010XXXXX - SD/eSD 0x011XXXXX - MMC/eMMC boot Others - Reserved Refer to Fuse Map for details.	SRC
0x450[15:8]	BOOT_CFG2	8	BOOT configuration register #2, Usage varies, depending on selected boot device.	See fuse-map tab for details.	SRC
				is not allowed	
0x460[4]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development) 0=Boot mode configuration is taken from GPIOs. 1=Boot mode configuration is taken from	SRC SW(ROM)

Figure 10 - Fusemap Description Table for i.MX6 UltraLite (Table 5-9)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x450 and 0x460 is mapped to. As shown in Figure 11 below this is **Bank0 Word5** and **Bank0 Word6**.

21B_C450	Value of OTP Bank0 Word5 (Configuration and Manufacturing Info.) (OCOTP_CFG4)	32	R/W	0000_0000h	35.5.16/ 2190
21B_C460	Value of OTP Bank0 Word6 (Configuration and Manufacturing Info.) (OCOTP_CFG5)	32	R/W	0000_0000h	35.5.17/ 2190

Figure 11 - OCOTP memory map for i.MX6 UltraLite

With this information we can now determine that bank0 and word5 should be programmed with a value where bit 5, bit 6 and bit 11 (BOOT_CFG) are set to 1. The hexadecimal value that corresponds to these bits being set to 1 is 0x00000860.

For bank0 word6, bit 4 (BT_FUSE_SEL) should be set to 1. The hexadecimal value is 0x00000010.

6.4.1 Instructions

From the **u-boot console** run the following commands to program the fuses:

NOTE: These commands cannot be undone.

```
=> fuse prog 0 5 00000860
=> fuse prog 0 6 00000010
```

6.5 iMX7 Dual COM

Section 6.3.1 (Boot Fusemap) in the i.MX7 Dual User's Manual contains the table shown in Figure 12 below. In this table you can see that for MMC/eMMC boot, BOOT_CFG[13] must be set to 1 while BOOT_CFG[12], BOOT_CFG[14], and BOOT_CFG[15] must be 0.

Table 6-10. Boot Device Select

Boot Device	BOOT_CFG[15]	BOOT_CFG[14]	BOOT_CFG[13]	BOOT_CFG[12]	BOOT_CFG[11]
SD/eSD	0	0	0	1	x
MMC/eMMC	0	0	1	0	x
NAND	0	0	1	1	x
QSPI	0	1	0	0	x
EIM	0	1	0	1	Memory Type: 0 - NOR Flash 1 - OneNAND
SPI/NOR	0	1	1	0	x

Figure 12 - Boot Device Select for i.MX7 Dual (Table 6-10)

For iMX7 Dual COM board, eMMC is attached to the USDHC3 interface. Fuses must be programmed to select the correct USDHC port. This is described in section 6.6.5.3.1 – Expansion device eFUSE configuration – in the User's Manual for the i.MX7 Dual processor, see Figure 13. BOOT_CFG[11:10] must be set to 10 to choose USDHC-3.

BOOT_CFG[15:12]	OEM	Boot device selection	Yes	0000	0001 - Boot from SD/eSD 0010 - Boot from MMC/eMMC
BOOT_CFG[11:10]	OEM	USDHC port selection	Yes	00	00 - USDHC-1 01 - USDHC-2 10 - USDHC-3 else - reserved

Figure 13 - USDHC boot eFUSE descriptions for i.MX7 Dual (Table 6-42)

The *Fuse Description Table* from section 6.3.3 in the i.MX7 Dual User's Manual gives that BOOT_CFG is at fuse address **0x470**, see Figure 14. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses. This fuse is also available at address **0x470**.

0x470[19:0]	BOOT_CFG	20	BOOT configuration register, Usage varies, depending on selected boot device.	See "Boot" sheet	SRC SW(ROM)
0x470[28]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development): • 0=Boot mode configuration is taken from GPIOs. • 1=Boot mode configuration is taken from fuses. If boot_mode="10" (Production): • 0 - Boot using Serila Loader (USB) • 1- Boot mode configuration is taken from fuses.	SRC SW(ROM)

Figure 14 - Fusemap Description Table for i.MX7 Dual (Table 6-19)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x470 is mapped to. As shown in Figure 15 below this is **Bank1 Word3**.

3035_0460	Value of OTP Bank1 Word2 (Tester Information) (OCOTP_TESTER5)	32	R/W	0000_0000h	6.4.5.23/ 1135
3035_0470	Value of OTP Bank1 Word3 (Boot Configuration Information) (OCOTP_BOOT_CFG0)	32	R/W	0000_0000h	6.4.5.24/ 1136
3035_0480	Value of OTP Bank2 Word0 (Boot Configuration Information) (OCOTP_BOOT_CFG1)	32	R/W	0000_0000h	6.4.5.25/ 1136

Figure 15 - OCOTP memory map for i.MX7 Dual

With this information we can now determine that bank1 and word3 should be programmed with a value where bit 11 (port selection), bit 13 (device selection) and bit 28 (BT_FUSE_SEL) are set to 1. The hexadecimal value that corresponds to these bits being set to 1 is 0x10002800.

6.5.1 Instructions

From the **u-boot console** run the following command to program the fuses:

NOTE: This command cannot be undone.

```
=> fuse prog 1 3 10002800
```

6.6 iMX8M Quad COM

Section 6.2.1 (Boot Fusemap) in the i.MX8 M User's Manual contains the table shown in Figure 16 below. In this table you can see that for MMC/eMMC boot, BOOT_CFG[13] must be set to 1 while both BOOT_CFG[12] and BOOT_CFG[14] must be 0.

Table 6-37. Boot Device Select

Boot Device	BOOT_CFG[14]	BOOT_CFG[13]	BOOT_CFG[12]
SD/eSD	0	0	1
MMC/eMMC	0	1	0
NAND	0	1	1

Figure 16 - Boot Device Select for i.MX8 M (Table 6-37)

There is a more detailed fusemap table specifically for MMC/eMMC booting, see Figure 17. For the iMX8M Quad COM board the eMMC is on the SD1 interface. This means that Port Select must be set to 00 which is the default value so there is no need to change these fuse bits.

Table 6-39. MMC/eMMC Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x470[15:8]	Reserved	Boot Device Select			Port Select: 00 - SD1 01 - SD2		Power Cycle Enable 0 - Disable 1 - Enable	SD Loopback Clock Source SEL (SDR50 and SDR104 Only) 0 - Disable 1 - Enable
0x470[7:0]	Fast Boot: 0 - Regular 1 - Fast Boot	Bus Width: 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - reserved			Speed 00 - Normal 01 - High 10 - Reserved for HS200 11 - Reserved		USDHC1 IO VOLTAGE SELECTION 0 - 3.3V 1 - 1.8V	USDHC2 IO VOLTAGE SELECTION 0 - 3.3V 1 - 1.8V

Figure 17 - MMC/eMMC Boot Fusemap for i.MX8 M (Table 6-39)

The *Fuse Description Table* from section 6.2.3 in the i.MX8 M User's Manual gives that BOOT_CFG is at fuse address **0x470**, see Figure 18. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses.

0x470[15:0]	BOOT_CFG	16	BOOT configuration register, Usage varies, depending on selected boot device.	See boot fusemap for details.	SRC SW(ROM)
0x470[24:16]	Reserved	9	Reserved	Reserved	Reserved
0x470[25]	SEC_CONFIG[1]	1	Security Configuration (with SEC_CONFIG[0])	00 - FAB (Open) 01 - Open - allows any code to be flashed and executed, even if it has no valid signature. 1x - Closed (Security On)	SW (ROM), SRC, SNVS, TPSMP
0x470[26]	Reserved	1	Reserved	Reserved	Reserved
0x470[28]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development) 0=Boot mode configuration is taken from GPIOs. 1=Boot mode configuration is taken from fuses. If boot_mode="10" (Production) 0 - Boot using Serial Loader (USB) 1- Boot mode configuration is taken from fuses.	SRC SW(ROM)

Figure 18 - Fusemap Description Table for i.MX8 M (Table 6-43)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x470 is mapped to. As shown in Figure 19 below this is **Bank1 and Word3**.

3035_0460	Value of OTP Bank1 Word2 (Tester Info.) (OCOTP_HW_OCOTP_TESTER5)	32	R/W	0000_0000h	6.3.5.15/901
3035_0470	Value of OTP Bank1 Word3 (Boot Configuration Info.) (OCOTP_HW_OCOTP_BOOT_CFG0)	32	R/W	0000_0000h	6.3.5.16/902
3035_0480	Value of OTP Bank2 Word0 (Boot Configuration Info.) (OCOTP_HW_OCOTP_BOOT_CFG1)	32	R/W	0000_0000h	6.3.5.17/902

Figure 19 - OCOTP memory map for i.MX 8M

With this information we can now determine that bank1 and word3 should be programmed with a value where bit 13 (BOOT_CFG[13]) and bit 28 (BT_FUSE_SEL) are set to 1. The hexadecimal value that corresponds to these two bits being set to 1 is 0x10002000.

6.6.1 Instructions

From the **u-boot console** run the following command to program the fuses:

NOTE: This command cannot be undone.

```
=> fuse prog 1 3 10002000
```

6.7 iMX8M Mini uCOM

Section 6.2.1 (Boot Fusemap) in the i.MX8 M Mini User's Manual contains the table shown in Figure 20 below. In this table you can see that for MMC/eMMC boot, BOOT_CFG[13] must be set to 1 while both BOOT_CFG[12] and BOOT_CFG[14] should be 0.

Table 6-41. Boot Device Select

Boot Device	BOOT_CFG[14]	BOOT_CFG[13]	BOOT_CFG[12]
SD/eSD	0	0	1
MMC/eMMC	0	1	0
NAND	0	1	1
FlexSPI	1	0	0
SPI/NOR	1	1	0

Figure 20 - Boot Device Select for i.MX8 M Mini (Table 6-41)

There is a more detailed fusemap table specifically for MMC/eMMC booting, see Figure 21. For the iMX8M Mini uCOM board the eMMC is on the SD3 (usdhc3) interface. This means that Port Select must be set to 10, that is, bit 11 must be set to 1.

Table 6-43. MMC/eMMC Boot Fusemap

Addr	7	6	5	4	3	2	1	0
0x470[15:8]	Reserved	Boot Device Select			Port Select: 00 - SD1 01 - SD2 10 - SD3		Power Cycle Enable 0 - Disable 1 - Enable	SD Loopback Clock Source SEL (SDR50 and SDR104 Only) 0 - Through SD pad 1 - Direct
0x470[7:0]	Fast Boot: 0 - Regular 1 - Fast Boot	Bus Width: 000 - 1-bit 001 - 4-bit 010 - 8-bit 101 - 4-bit DDR (MMC 4.4) 110 - 8-bit DDR (MMC 4.4) Else - Reserved			Speed 00 - Normal 01 - High 10 - Reserved for HS200 11 - Reserved		USDHC IO VOLTAGE SELECTION For Normal Boot Mode 0 - 3.3V 1 - 1.8V	USDHC IO VOLTAGE SELECTION For Manufacture Mode 0 - 3.3V 1 - 1.8V

Figure 21 - MMC/eMMC Boot Fusemap for i.MX8 M Mini (Table 6-43)

The *Fuse Description Table* from section 6.2.3 in the i.MX8 M Mini User's Manual gives that BOOT_CFG is at fuse address **0x470**, see Figure 22. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses.

Fuse Address	Fuses Name	Number of Fuses	Fuses Function	Setting	Used by
0x470[15:0]	BOOT_CFG	16	BOOT configuration register, Usage varies, depending on selected boot device.	See boot fusemap for details.	SRC SW(ROM)
0x470[24:16]	Reserved	9	Reserved	Reserved	Reserved
0x470[25]	SEC_CONFIG[1]	1	Security Configuration (with SEC_CONFIG[0])	00 - FAB (Open) 01 - Open - allows any code to be flashed and executed, even if it has no valid signature. 1x - Closed (Security On)	SW (ROM), SRC, SNVS, TPSMP
0x470[26]	Reserved	1	Reserved	Reserved	Reserved
0x470[27]	Reserved	1	Reserved	Reserved	Reserved
0x470[28]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development) 0=Boot mode configuration is taken from GPIOs. 1=Boot mode configuration is taken from fuses. If boot_mode="10" (Production) 0 - Boot using Serial Loader (USB) 1- Boot mode configuration is taken from fuses.	SRC SW(ROM)

Figure 22 - Fusemap Description Table for i.MX8 M Mini (Table 6-49)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x470 is mapped to. As shown in Figure 23 below this is **Bank1 Word3**.

3035_0460	Value of OTP Bank1 Word2 (Tester Info.) (OCOTP_HW_OCOTP_TESTER5)	32	R/W	0000_0000h	6.3.4.15/866
3035_0470	Value of OTP Bank1 Word3 (Boot Configuration Info.) (OCOTP_HW_OCOTP_BOOT_CFG0)	32	R/W	0000_0000h	6.3.4.16/867
3035_0480	Value of OTP Bank2 Word0 (Boot Configuration Info.) (OCOTP_HW_OCOTP_BOOT_CFG1)	32	R/W	0000_0000h	6.3.4.17/867

Figure 23 - OCOTP memory map for i.MX 8M Mini

With this information we can now determine that bank1 and word3 should be programmed with a value where bit 11 (BOOT_CFG[11] - Port Select), bit 13 (BOOT_CFG[13]) and bit 28 (BT_FUSE_SEL) are set to 1. The hexadecimal value that corresponds to these two bits being set to 1 is 0x10002800.

6.7.1 Instructions

From the **u-boot console** run the following command to program the fuses:

NOTE: This command cannot be undone.

```
=> fuse prog 1 3 10002800
```

6.8 iMX8M Nano uCOM

Section 6.2.1 (Boot Fusemap) of the i.MX8 M Nano User's Manual contains the table shown in Figure 24 below. In this table you can see that for USDHC3 (eMMC) boot code should be set to 0x02, that is, BOOT_MODE[1] should be 1.

Table 6-29. Boot Device Select

Boot Device Select	BOOT CODE	BOOT_MODE[5] (SAI3_TXD)	BOOT_MODE[4] (SAI2_TXD0)	BOOT_MODE[3]	BOOT_MODE[2]	BOOT_MODE[1]	BOOT_MODE[0]
Boot from internal fuses	0x00	0	0	0	0	0	0
USB Serial Download	0x01	0	0	0	0	0	1
USDHC3 (eMMC boot only, SD3 8 bit)	0x02	0	0	0	0	1	0
USDHC2 (SD boot only, SD2)	0x03	0	0	0	0	1	1
NAND 8-bit single device, 256 pages	0x04	0	0	0	1	0	0
NAND 8-bit single device, 512 pages	0x05	0	0	0	1	0	1
FlexSPI 3B Read	0x06	0	0	0	1	1	0
FlexSPI Hyperflash 3.3V	0x07	0	0	0	1	1	1
eCSPI Boot	0x08	0	0	1	0	0	0

Figure 24 - Boot Device Select for i.MX8 M Nano (Table 6-29)

The Boot fusemap shows that the boot mode fuses are located at bits 12-15 (BOOT_MODE[0] is on bit 12, BOOT_MODE[1] on bit 13, and so on), see Figure 25.

			(Mxic)		
0x470[15:8]	BOOT_MODE_FUSES Boot Rom will retrieve boot mode from these fuses instead of BOOT_MODE pins if: * BOOT_MODE_PINS=0x0 or * BT_FUSE_SEL blown		OVERRIDE_USDHC_BT_SEL 0 - Do not override 1 - Override	OVERRIDE_USDHC_BT_SEL_VAL 00 - uSDHC1 SD 01 - uSDHC1 eMMC 10 - uSDHC2 eMMC 11 - uSDHC3 SD	OVERRIDE_NAND_PG_PER_BLK 0 - Do not override 1 - Override

Figure 25 - Boot Fusemap for i.MX8 M Nano (Table 6-30)

The *Fuse Description Table* from section 6.2.3 in the i.MX8 M Nano User's Manual gives that BOOT_CFG is at fuse address **0x470**, see Figure 26. We also need to know the address of the BT_FUSE_SEL fuse since this must be set to boot from fuses.

0x460[31:0]	Reserved	32	Reserved	Reserved	Reserved
0x470[15:0]	BOOT_CFG	16	BOOT configuration register, Usage varies, depending on selected boot device.	See boot fusemap for details.	SRC SW(ROM)
0x470[24:16]	Reserved	9	Reserved	Reserved	Reserved
0x470[26:16]	Reserved	10	Reserved	Reserved	Reserved
0x470[27]	Reserved	1	Reserved	Reserved	Reserved
0x470[28]	BT_FUSE_SEL	1	Determines, whether using fuses for boot configuration, or GPIO /Serial loader.	If boot_mode="00" (Development) 0=Boot mode configuration is taken from GPIOs. 1=Boot mode configuration is taken from fuses. If boot_mode="10" (Production) 0 - Boot using Serial Loader (USB) 1- Boot mode configuration is taken from fuses.	SRC SW(ROM)

Figure 26 - Fuse Description table for i.MX8M Nano (Table 6-32)

As described in section 3.2 we can now use the OCOTP Memory Map to find which OTP bank and word the fuse address 0x470 is mapped to. As shown in Figure 23 below this is **Bank1 Word3**

3035_0460	Value of OTP Bank1 Word2 (Tester Info.) (OCOTP_HW_OCOTP_TESTER5)	32	R/W	0000_0000h	6.3.4.15/ 801
3035_0470	Value of OTP Bank1 Word3 (Boot Configuration Info.) (OCOTP_HW_OCOTP_BOOT_CFG0)	32	R/W	0000_0000h	6.3.4.16/ 802

Figure 27 - OCOTP memory map for i.MX 8M Nano

With this information we can now determine that bank1 and word3 should be programmed with a value where bit 13 (BOOT_MODE[1]), and bit 28 (BT_FUSE_SEL) are set to 1. The hexadecimal value that corresponds to these two bits being set to 1 is 0x10002000.

6.8.1 Instructions

From the **u-boot console** run the following command to program the fuses:

NOTE: This command cannot be undone.

```
=> fuse prog 1 3 10002000
```

6.9 iMX7ULP uCOM

For the iMX7ULP the Fusemap is not published in the User's Manual. Instead you need to request this from NXP. Because of this, there are no figures/screenshots included in this section for the fusemap. It is only described which OTP fuses you need to program.

The boot device selection is controlled via BT0_CFG1 which is located at Bank2 Word 4.

The BT_FUSE_SEL must also be set and this fuse is located at Bank2 Word7.

6.9.1 Instructions

From the **u-boot console** run the following command to program the fuses:

NOTE: This command cannot be undone.

```
=> fuse prog 2 4 00400002
=> fuse prog 2 7 00008000
```

7 Program Fuses from UUU

UUU (Universal Update Utility) is used to program the boards with new images / software. You can read more about UUU in the document *Working with Yocto to Build Linux* available on Embedded Artists website. You can download zip files containing the UUU utility and configuration files from http://imx.embeddedartists.com.

In short, UUU is a utility program used in combination with one or more configuration files. The configuration files contain the instructions to run when updating the board. If needed, you can program fuses from such a configuration file.

Below is an example from the configuration file `bootloader.uuu` belonging to the iMX8M Mini uCOM board. Here we have added the highlighted command (it is not normally included) below which will enable eMMC boot on the iMX8M Mini uCOM board. Please note the "-y" that is added so you don't need to confirm the command.

NOTE: Do not just copy the line below to your configuration file. Double-check the fuses you want to program before adding such a command. Once fuses have been programmed it cannot be undone.

```
...
FB: ucmd mmc dev ${emmc_dev}
FB: ucmd mmc info
FB: ucmd fuse prog -y 1 3 10002800
FB: flash bootloader files/imx-boot-imx8mmea-ucom-sd.bin
FB: done
```