Restoring and Updating the U-Boot Image in NAND Flash on the OMAP-L138 LCDK EVM

TEXAS INSTRUMENTS

Everything | Search | Login / Register

Products | Applications & designs | Tools & software | Support & training | Order Now | About TI | My History | Cart | English | myTI

TI Home > Semiconductors > Processors > OMAP-L138 Development Kit (LCDK)

Worldwide (in English)

OMAP-L138 Development Kit (LCDK)

(ACTIVE) TMDSLCDK138

Description & Features | Technical Documents | Support & Training | Order Now

Key Document

- OMAP-L138 C6000 DSP+ARM Processor datasheet (Rev. J) (PDF 2029 KB) 31 Jan 2017

» View All Technical Documents (3)

Description

The OMAP-L138 DSP+ARM9™ development kit will enable fast and easy Linux software and hardware development. This scalable platform will ease and accelerate software and hardware development of everyday applications that require real-time signal processing and control functional, including industrial control, medical diagnostics and communications. The low-cost kit, complete with freely downloadable and duplicable board schematics and design files, greatly reduces design work. A wide variety of standard interfaces for connectivity and storage allow developers to easily bring audio, video and other signals onto the board. Expansion headers such as LCD screen expansion headers and Leopard Imaging's camera sensor allow customers to extend the board's functionality.

The LCDK does not have an onboard emulator. An external emulator from TI (such as the XDS100, XDS200, XDS510, XDS560) or a third-party will be required to start development.

The TMDSLCDK138 replaces the TMDXLCDK138 with the same performance, price and features. It available on a limited quantity basis as inventory ramps.

Features

- Integrated floating-/fixed-point DSP with up to 456 MHz performance; and ARM9 with up to 456 MHz performance
- Software, expansion headers, schematics and application demos
- SDKs, DSP/BIOS RTOS, drivers, stacks and protocol, algorithm libraries, flash and boot utilities and StarterWare

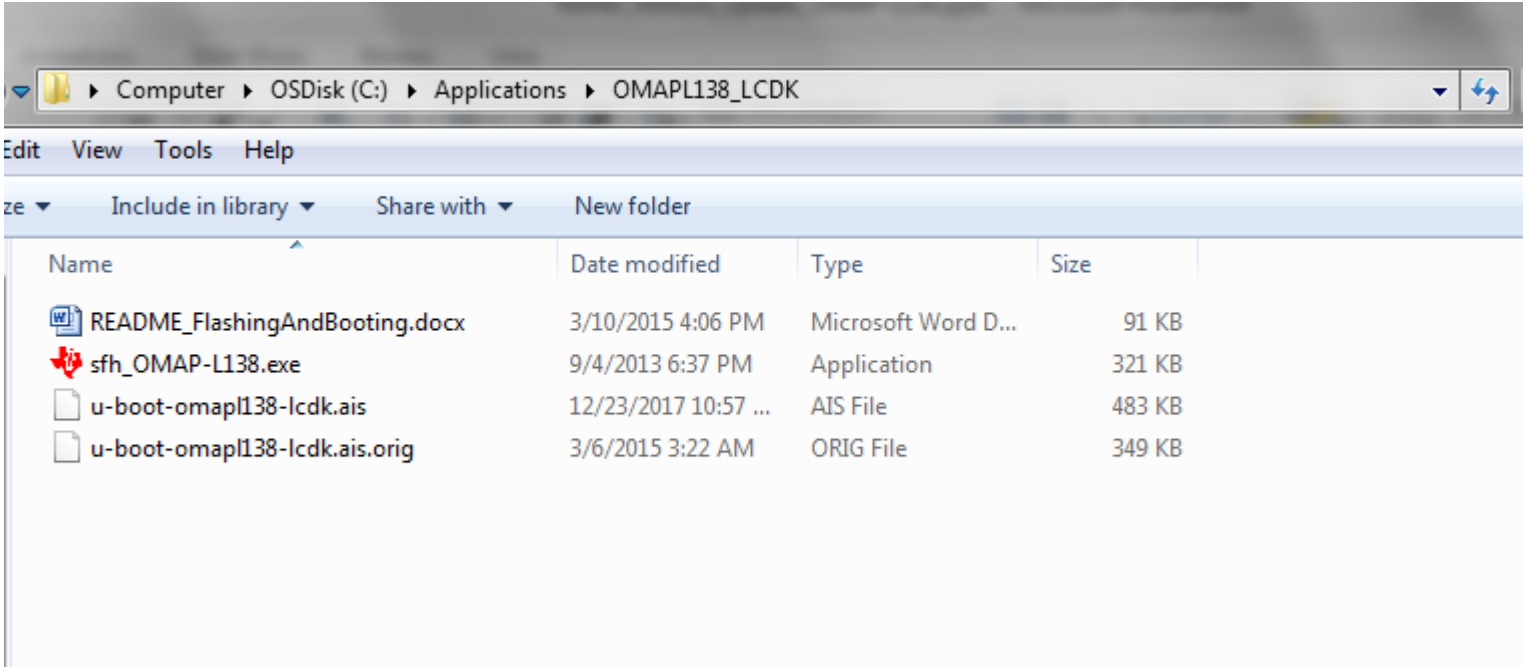
OMAP-L138 Development Kit (LCDK)

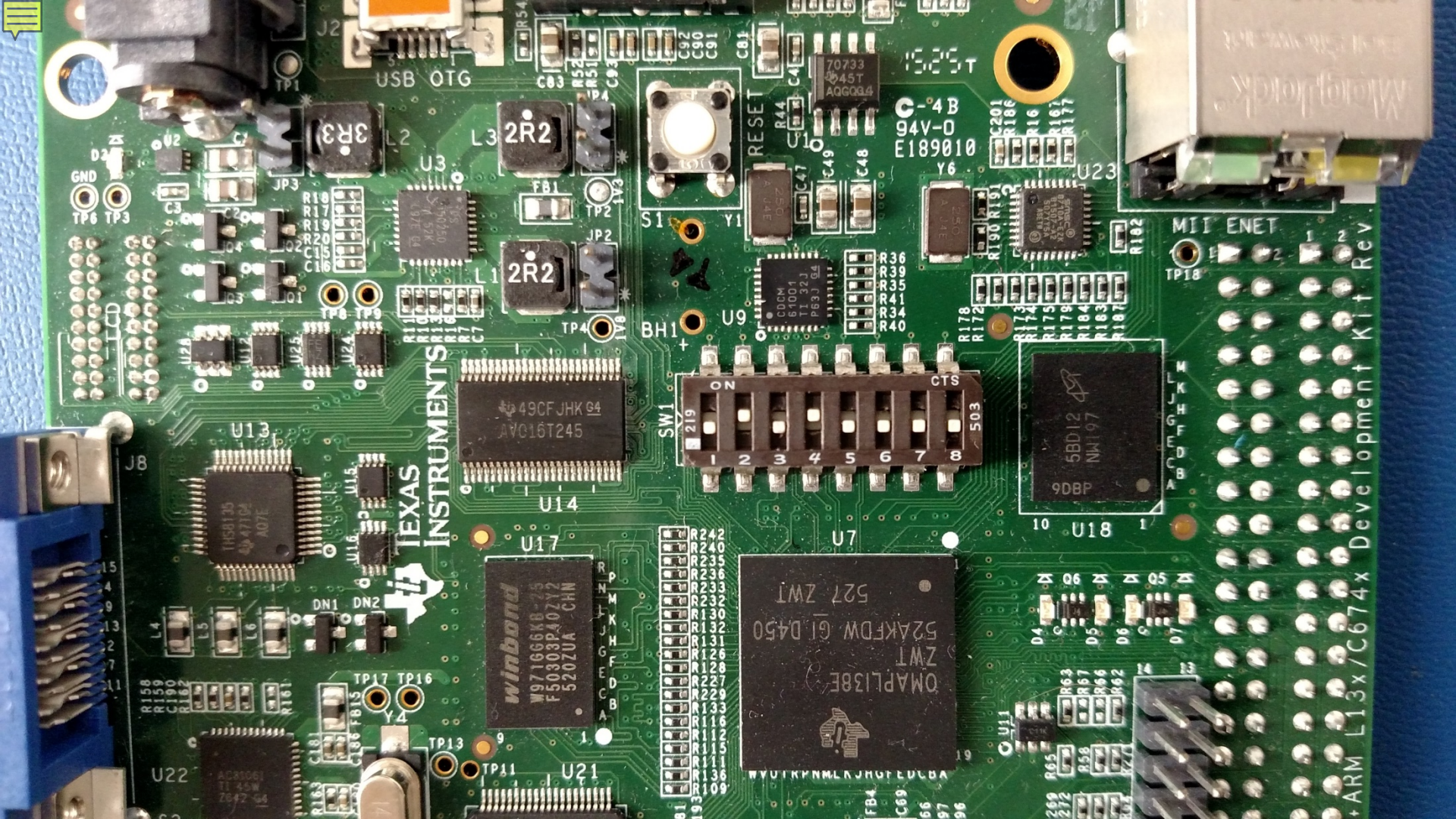
There are three methods for flashing NAND on the OMAP-L138 LCDK:

- Using UART2 mode through USB-UART port
- Using “nand write” command from u-boot prompt.
- From Linux kernel using the user space utility

Serial Boot and Flash-Loading Utility for OMAP-L138

http://processors.wiki.ti.com/index.php/File:OMAPL138_LCDK.zip




USB OTG
RESET
MII ENET
TEXAS INSTRUMENTS
ON
CTS
SW1
U14
U17
U7
U18
U21
U22
U23
U13
U3
U9
U1
S1
BH1
J2
J8
Y1
Y4
Y6
L1
L2
L3
2R2
3R3
GND
TP1
TP2
TP3
TP4
TP6
TP8
TP9
TP11
TP13
TP16
TP17
TP18
JP2
JP3
FB1
DN1
DN2
Q5
Q6
94V-0
E189010
ARM L13x/C674x Development Kit Rev

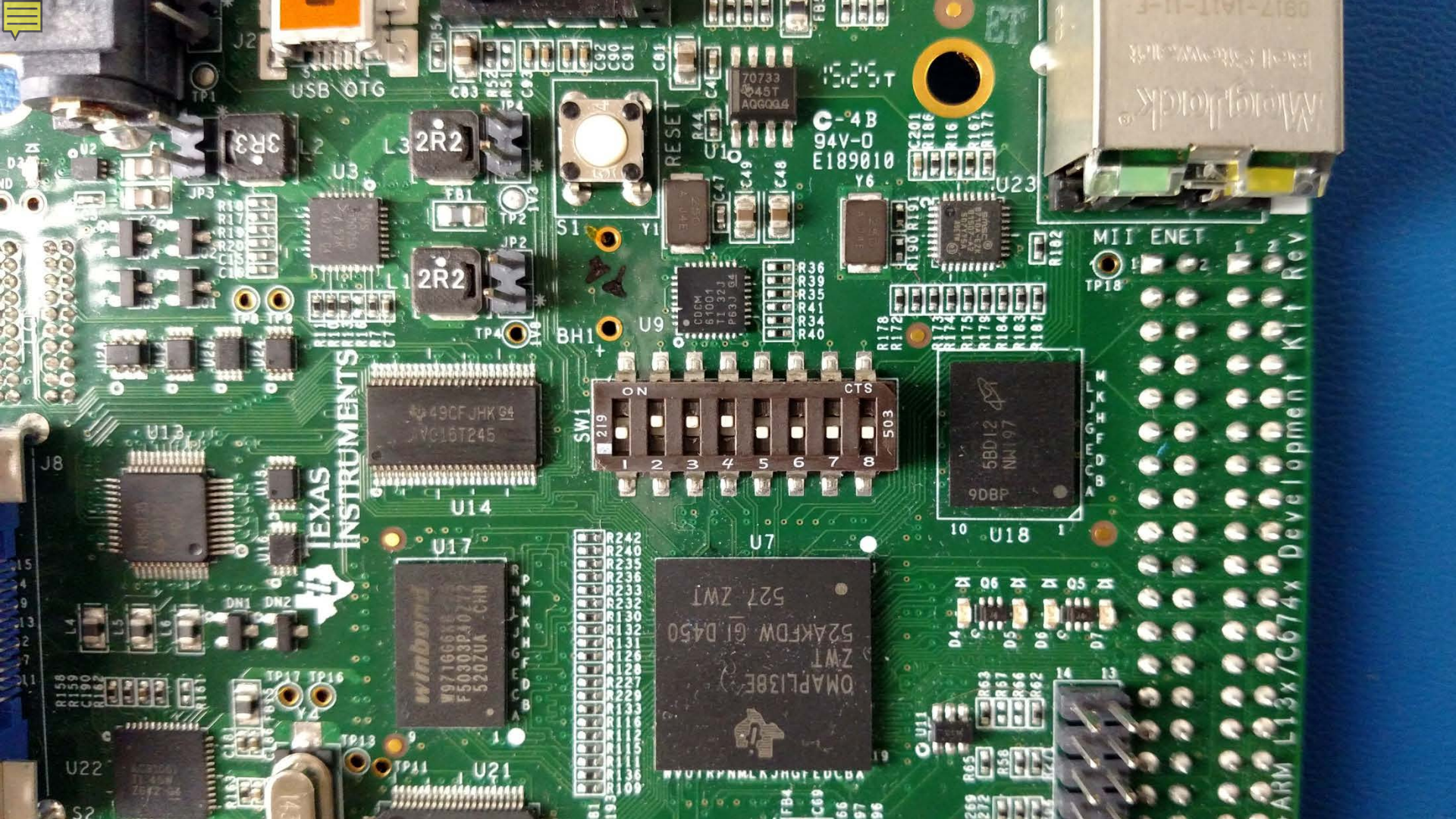
Restore NAND using UART2 mode through USB-UART port

- To verify the setting, the emulator console should display “BOOTME” after the RESET button is pressed.
- The console setting should be 115200 baudrate and 8N1.
- Once verified, console needs to be closed while flashing the NAND.
- On the PC, start the cmd.exe screen, and cd to the flashing utility folder.
- First erase the NAND with the erase command:

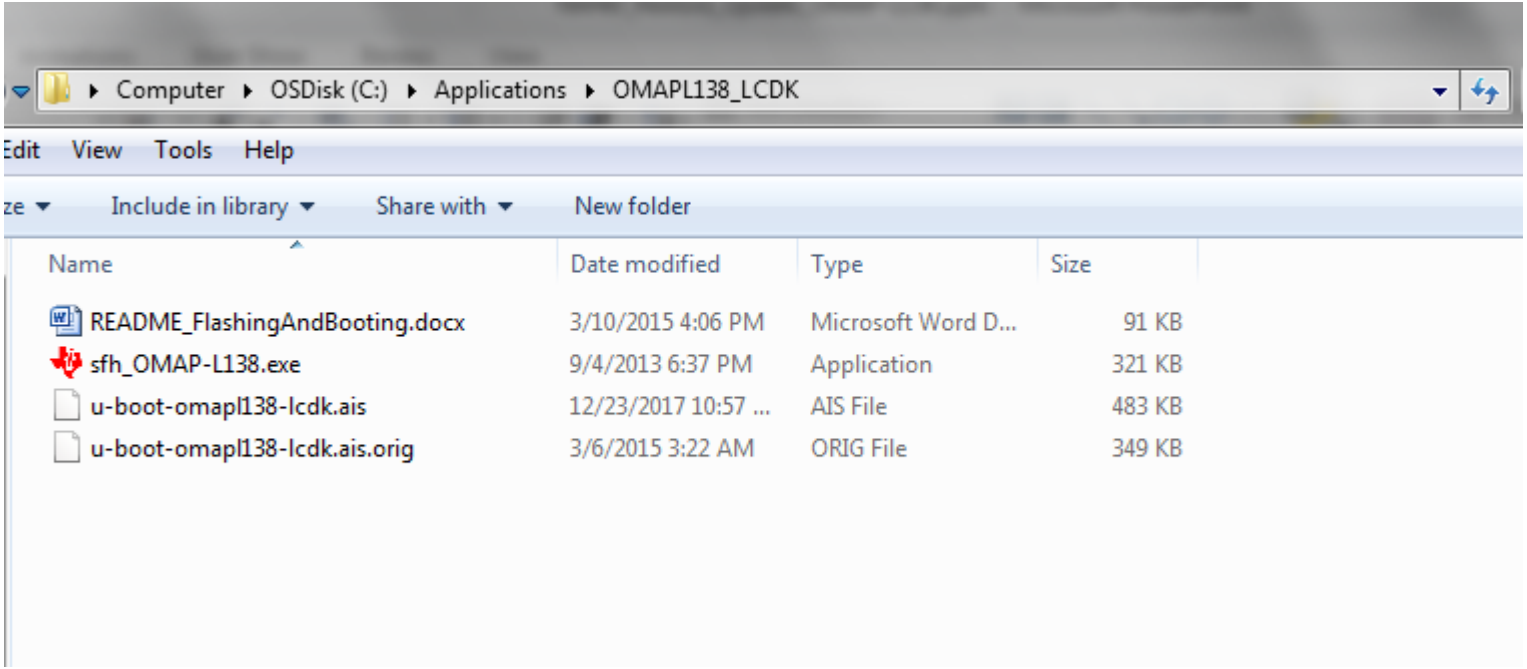
```
sfh_OMAP-L138.exe -flashType NAND -targetType OMAPL138_LCDK -erase -p COM3
```

- When it prompts for “Waiting for BOOTME,” press the RESET button on the OMAP-L138 LCDK.
- The PC screen will show the erase is completed.
- Then issue the flash command:

```
sfh_OMAP-L138.exe -flashType NAND -targetType OMAPL138_LCDK -flash_noubl
u-boot-omapl138-lcdk.ais -p COM3
```



USB OTG
RESET
C-4B
94V-0
E189010
MII ENET
TEXAS INSTRUMENTS
ON
CTS
SW1
U14
U17
U7
U18
U23
U21
U22
U13
J8
J2
S1
S2
BH1
U9
Y1
Y6
Y4
ARM L13x/C674x Development Kit Rev


Computer ▸ OSDisk (C:) ▸ Applications ▸ OMAPL138_LCDK
Edit View Tools Help
Include in library
Share with
New folder
Name
Date modified
Type
Size
README_FlashingAndBooting.docx 3/10/2015 4:06 PM Microsoft Word D... 91 KB
sfh_OMAP-L138.exe 9/4/2013 6:37 PM Application 321 KB
u-boot-omapl138-lcdk.ais 12/23/2017 10:57 ... AIS File 483 KB
u-boot-omapl138-lcdk.ais.orig 3/6/2015 3:22 AM ORIG File 349 KB

	01	02	03	04	05	06	07	08
01								
02								
03								
04								
05								
06								
07								
08								

ECC Layout - RBL

	01	02	03	04	05	06	07	08
01								
02								
03								
04								
05								
06								
07								
08								

ECC Layout - Linux

u-boot/drivers/mtd/nand/davinci_nand.c

```
+       .eccpos = {6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
+                 22, 23, 24, 25, 26, 27, 28, 29, 30, 31,
+                 38, 39, 40, 41, 42, 43, 44, 45, 46, 47,
+                 54, 55, 56, 57, 58, 59, 60, 61, 62, 63,
+                 },
+       .oobfree = {{2, 4}, {16, 6}, {32, 6}, {48, 6}},
```

Restore NAND using nand write command

- Once you have confirmed that the u-boot ECC layout aligns with the RBL, then the u-boot can be updated using the “nand” command from u-boot. You will erase the existing image and replace it by writing the new image.
- First, use TFTP to load the new u-boot binary .ais file to the OMAP-L138 LCDK

```
setenv serverip [tftp server IP]
dhcp u-boot.ais
```

- The downloaded image file is 0x78a38 bytes
- The command “nand info” shows that the NAND page size is 2048 bytes.

```
> nand info
Device 0: nand0, sector size 128 KiB
  Page size       2048 b
  OOB size          64 b
  Erase size    131072 b
  subpagesize       512 b
  options     0x40000002
  bbt options 0x   28000
```

Restore NAND using nand write command

- For the nand erase and nand write commands, use the file size of 0x79000, which is aligned to the next 2048 byte page. The commands are:
  - Erase: => **`nand erase 0x20000 0x79000`**
  - Write: => **`nand write 0xc0700000 0x20000 0x79000`**
- After nand write finishes, the update is complete.

flashtool.c change for Kernel version > 3.2

```
include "genecc.h"

+#define MTD_MODE_RAW MTD_FILE_MODE_RAW
+
 enum exit_codes {
        EXIT_OK                 = 0,
        EXIT_FAIL               = 1,    // general
fatal error
```

Restore NAND from Linux kernel using user space utility

- From the EVM Kernel console, the mtd sysfs shows mtd1 is for u-boot:

```
root@omapl138-lcdk:~# cat /proc/mtd
dev:    size   erasesize  name
mtd0: 00020000 00020000 "u-boot env"
mtd1: 00080000 00020000 "u-boot"
mtd2: 1ff60000 00020000 "free space"
root@omapl138-lcdk:~#
```

- u-boot.ais can be flashed onto the OMAP-L138 using the following command:

```
root@omapl138-lcdk:~# ./flashtool -w -s 0 --dm365-rbl /dev/mtd1 u-boot.ais
Write block at 0x0
Write block at 0x20000
Write block at 0x40000
Write block at 0x60000
```

- The updated u-boot will take effect after the next power cycle or pushing the RESET button.

For More Information

- OMAP-L138/C6748 Development Kit (LCDK) Wiki: http://processors.wiki.ti.com/index.php/L138/C6748_Development_Kit_(LCDK)
- Processor SDK for OMAP-L138:

  http://www.ti.com/tool/PROCESSOR-SDK-OMAPL138
- OMAP-L138 Processor Technical Reference Manual (TRM):

  http://www.ti.com/lit/spruh77
- Processor SDK U-Boot User's Guide: http://software-dl.ti.com/processor-sdk-linux/esd/docs/latest/linux/Foundational_Components.html#u-boot
- For questions about this training, refer to the E2E Community Forums for Sitara Processors at http://e2e.ti.com/support/arm/sitara_arm/

©Copyright 2017 Texas Instruments Incorporated. All rights reserved.

This material is provided strictly “as-is,” for informational purposes only, and without any warranty.
Use of this material is subject to TI’s **Terms of Use**, viewable at TI.com