

Serial Boot and Flash Loading Utility for OMAP-L138

Contents

TI Flash and Boot Utilities

Obtaining the software

Compiling

Running

Serial Flasher Options

Restoring the OMAP-L138 EVM SPI Flash

Modifications for Custom Boards

Known Issues

License

TI Flash and Boot Utilities

The following software supports the OMAP-L138, C6748, and AM1808 families of devices.

This package contains a set of utilities running from the command-line on Windows for flashing the NAND, NOR, and SPI Flash of the OMAP-L138 EVM via the serial port.

The Serial Flashing Host Utility (which offers the same functionality as a previous program called DVFlasher) executable is called sfh_OMAP-L138.exe.

These programs each encapsulate a distinct binary UBL which is transferred via the UART. This implies that the chip must be operating in the UART boot mode, showing the BOOTME prompt.

NOTE: The assumption is made that the UART of the device operates at 115200, 8N1. If the oscillator used with the device does not match the one used on the EVM (e.g. on a custom platform), the baud rate assumption may be incorrect.

Obtaining the software

The latest source code and binaries can be download from [here \(https://sourceforge.net/projects/dvflashutils/files/OMAP-L138/v2.40/\)](https://sourceforge.net/projects/dvflashutils/files/OMAP-L138/v2.40/).

Note that for flashing UBLs from PSP releases 03.20.00.13 and earlier, v2.29 of the serial flasher should be used.

For flashing UBLs from PSP releases 3.20.00.14 and later, the latest version of the serial flasher should be used.

Compiling

A makefile is included for compiling the host and target parts of each utility. The target portions are built first since they are embedded into the the host executables. These target portions are loaded to the OMAP-L138 device via the UART boot mode.

Note that the pre-built executables have been tested to work on the EVMs. It is not necessary to rebuild them unless changes need to be made for a custom board.

Under Windows

Please see the following page for detailed instructions for rebuilding the utilities in included in this package:

http://processors.wiki.ti.com/index.php/Rebuilding_the_Flash_and_Boot_Utils_Package (http://processors.wiki.ti.com/index.php/Rebuilding_the_Flash_and_Boot_Utils_Package)

Under Linux

The Mono Framework must be installed and in the path. RPMs are available at '<http://www.mono-project.com/Downloads>'. The below instructions assume the that GNU ARM cross-compiler tools (arm-arago-linux-gnueabi-gcc, etc.) are in the current PATH.

The C6x Compiler Tools are also required to build some components. These are available free of charge at https://www-a.ti.com/downloads/sds_support/TICodegenerationTools/download.htm. You will need to add the bin directory to your PATH and to set the environment variables as directed by the installer. Edit the build.mak file under Common directory and add the paths to ARM and C6x compilers.

Then go to the GNU directory of the package and run:

```
make
```

Running

Under Windows

This utility can be run from the command line under Windows with the .Net Framework 2.0 or later installed through the following steps:

1. Set the boot pins to UART2 boot mode. This is done by setting switch S7 on the LogicPD OMAP-L138 EVM according to the following table:

Pin#	1	2	3	4	5	6	7	8

Position	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON
----------	-----	-----	-----	-----	-----	-----	----	----

For Spectrum Digital EVM:

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON

1. Connect a serial cable from the host computer to the OMAP-L138 EVM
2. Open a command prompt on the host computer and change directory to where the SFH executable is located.
3. Run the flashing utility with this command:

- `..\OMAP-L138\gnu\sfh_OMAP-L138.exe [options]`

4. Turn on the board

Under Linux

The serial flasher can be run on a Linux machine with the latest open-source Mono Framework installed. The steps are identical to the Windows environment except the command:

- `mono ./sfh_OMAP-L138.exe [options]`

Serial Flasher Options

There are three modes for using the serial flasher:

1. Erase the target flash type - This will erase the entire contents of the flash.

- `..\sfh_OMAP-L138.exe -erase`

2. Flash the memory with a single application image - This will place an application image at address 0x0 of the flash. This must be an AIS format binary, which can be created with the AISgen utilities bundled with the Bootloader User Guide and associated files.

- `..\sfh_OMAP-L138.exe -flash_noubl <binary application file>`

3. Flash the memory with a UBL and application image - This will place a UBL at address 0x0 of the flash and an application image, such as u-boot, at address 0x10000. In general, a UBL is not required, and this mode is only used when restoring the default flash contents, as explained in the following section.

- `..\sfh_OMAP-L138.exe -flash <UBL binary file> <binary application file>`

For C6748 devices, use the following option: "-targetType C6748"

For AM1808 devices, use the following option: "-targetType AM1808"

Note: Only the UBL found in the OMAP-L138/GNU/UBL directory can be used with the serial flasher. Other UBLs are not compatible with the serial flasher format.

Note: The 'flash_noubl' and 'flash' options will automatically erase the necessary amount of memory in order to fit the UBL and/or application image.

Currently, the only supported flash types are NAND, NOR, and SPI.

Additional options are shown below:

```

-targetType      : Specifies exact target type within OMAP-L138 family (default OMAP-L138)
-flashType       : Specifies exact flash type (default SPI_MEM)
-p <COM PORT NAME> : Allows specifying com port other than default 'COM1' or '/dev/ttyS0'.
-h               : Show help text.
-v               : See verbose output from target device
-baud <BAUD RATE> : Allows specifying baud rate other than default (115200)
-APPStartAddr    : Changes entry point of application (default 0xC1080000)
-APPLoadAddr     : Changes load address of application (default 0xC1080000)
-APPFlashBlock   : Changes the block to flash the image into (only for no_ubl mode)

```

Once any command is run, the "Waiting for BOOTME..." prompt shows. Power cycle the board or press the reset button to continue.

Restoring the OMAP-L138 EVM SPI Flash

If the contents of the OMAP-L138 SPI flash become corrupted, they can be restored through the following steps:

1. Set the boot pins to UART2 boot mode. This is done by setting switch S7 on the OMAP-L138 EVM according to the following table:

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON

For Spectrum Digital EVM:

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON

1. Obtain the UBL and u-boot binary files and place them in the same directory as the SPI flashing utility.

- A pre-built UBL bin file is located in the `OMAP-L138/GNU/ubl/` directory of this serial flashing project. The UBL bin file from the PSP installation will also work, located at `images/u-boot/`.
- A pre-built u-boot image is located in the `images/u-boot/` directory of the PSP installation.

2. Run the following command to flash the UBL and u-boot binary files:

- `..\sfh_OMAP-L138.exe -flash <UBL binary file> <u-boot binary file>`

- Note that if the serial port is not on COM1, the correct port must be specified using the "-p" option. To determine which COM port to use, open HyperTerminal, connect to one of the ports, and make sure the word "BOOTME" appears when the EVM is reset.

- Note: If you are using a device other than OMAP-L138, such as C6748, you must use the "-flashType" option and specify the device you are using.
3. Turn on the board
 4. After the UBL and u-boot files have been flashed, set the boot pins to SPI boot mode and reboot the board. This is done by setting switch S7 on the OMAP-L138 EVM according to the following table:

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

For Spectrum Digital EVM:

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	ON	ON	OFF

1. Once u-boot has loaded, follow the instructions in the Getting Started Guide to copy the Linux kernel and filesystem to the SPI flash.

- **Booting from SPI Flash**
- **Note: This step requires that the setup procedures in the Getting Started guide have been followed:**
 - [Install SDK Software](#)
 - [Setup a TFTP Server](#)

Modifications for Custom Boards

The default settings used by the tools apply only to the EVMs. For custom boards, changes will most likely be required, and the tools must be rebuilt. The custom changes should be made to the files in the OMAP-L138/Common/src and OMAP-L138/Common/include directories. Common changes include:

- **DDR Configuration**
 - OMAP-L138/Common/src/device.c: Modify the parameters passed to the function DEVICE_ExternalMemInit to match the DDR timing requirements for the custom board
- **UART Settings**
 - OMAP-L138/Common/include/device_uart.h: Modify the #define DEVICE_UART_PERIPHNUM to set which UART is connected to the host PC.
 - Note that flow control is not used, so only the RX and TX lines need to be connected for boot and flashing purposes.
- **SPI Settings**
 - OMAP-L138/Common/include/device_spi.h: Modify the global macros to select the appropriate peripheral and chip select numbers.
 - OMAP-L138/Common/src/device_spi.c: Modify the flash organization in DEVICE_SPI_MEM_params.
 - The SPI flash on the EVM does not need to be unlocked in order to erase or write. Some SPI flashes may need to set the BL bits before writing. Check the device datasheet to see if this step needs to be added to the initialization.
- **NAND Settings**
 - OMAP-L138/Common/include/device_async_mem.h: Modify the #define DEVICE_ASYNC_MEM_NANDBOOT_BUSWIDTH to match the bus width of your NAND
- **PLL Settings (for non 24 MHz input clocks)**
 - OMAP-L138/Common/src/device.c: Modify the parameters passed into the function DEVICE_PLL0Init and DEVICE_PLL1Init to set the proper PLL output frequencies
 - OMAP-L138/Common/src/device_uart.c: Modify the 'divider' field in DEVICE_UART_config to achieve 115200 baud rate

Known Issues

The software provided has no known issues on TI supported evaluation platforms. The software may need modifications in case of custom hardware.

License

These tools are provided as freeware, with no guarantee.

<p>1. switchcategory:MultiCore=</p> <ul style="list-style-type: none"> ▪ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum ▪ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum <p>Please post only comments related to the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>Keystone=</p> <ul style="list-style-type: none"> ▪ For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum ▪ For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum <p>Please post only comments related to the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>C2000=For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>DaVinci=For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>MSP430=For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>OMAP35x=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>OMAPL1=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>	<p>MAVRK=For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article Serial Boot and Flash Loading Utility for OMAP-L138 here.</p>
---	--	--	---	---	--	---	--



Links

[Amplifiers & Linear](#)

[Audio](#)

[Broadband RF/IF & Digital Radio](#)

[Clocks & Timers](#)

[Data Converters](#)

[DLP & MEMS](#)

[High-Reliability](#)

[Interface](#)

[Logic](#)

[Power Management](#)

[Processors](#)

- [ARM Processors](#)
- [Digital Signal Processors \(DSP\)](#)
- [Microcontrollers \(MCU\)](#)
- [OMAP Applications Processors](#)

[Switches & Multiplexers](#)

[Temperature Sensors & Control ICs](#)

[Wireless Connectivity](#)

Retrieved from "https://processors.wiki.ti.com/index.php?title=Serial_Boot_and_Flash_Loading_UTILITY_for_OMAP-L138&oldid=221934"

This page was last edited on 13 October 2016, at 06:09.

Content is available under [Creative Commons Attribution-ShareAlike](#) unless otherwise noted.