

Flashing Binaries to DRA7xx Factory Boards Using Device Firmware Upgrade (DFU)

RaviB

ABSTRACT

This application report provides detailed procedures for flashing the binary images to eMMC Flash memory using Device Firmware Upgrade (DFU). Generally, the MMC/SD boot mode can be used to boot the fresh production board/EVM. In case there is not an external MMC/SD card available as part of production EVM or final product, this application report will be useful to flash the images to the factory boards using DFU and USB peripheral boot mode of DRA7xx.

Contents

1	Introduction	1
2	Building First Stage Bootloader With DFU Support	2
3	ROM Downloading u-boot-spl.bin From Ubuntu Host	3
4	Setting Up EVM and Reset the Board	3
5	ROM Download the u-boot-spl.bin From Ubuntu Host	3
6	Flashing the Boot Image to eMMC Device	3
7	Executing From eMMC.....	4
8	References	4

List of Figures

1	ROM Downloads MLO From Ubuntu Host PC.....	2
2	MLO Downloads Binaries From Ubuntu Host PC	2
3	Setup of Flashing eMMC From ubuntu-host Using USB Interface	2

1 Introduction

The Device Firmware Upgrade feature is used to program the firmware to flash memory devices such as eMMC, MMC/SD, and so forth. The DFU feature is available as part of second stage boot loader (U-Boot) in mainline. The dfu-util tool (version 0.5) executes from the Ubuntu host machine used to program the target resident Flash memory devices, which communicate through USB interface with the target resident DFU software.

The first time assembled or manufactured, the factory boards/EVMs do not have any binaries loaded in the Flash memory devices. In order to boot the EVM, various boot modes are available as shown below:

- MMC/SD boot mode
- QSPI boot mode
- eMMC boot mode
- USB Peripheral boot mode

The MMC/SD boot mode can be used to boot the fresh production. In case there is not an external MMC/SD card available as part of the production EVM or final product, this application report will be useful to flash the images to the factory boards using USB Peripheral mode and DFU support in single stage bootloader or MLO.

Flashing binaries to eMMC via DFU is a two step procedure:

1. DRA7xx Boot ROM downloads single stage bootloader (MLO) via USB peripheral boot mode from Ubuntu host PC.

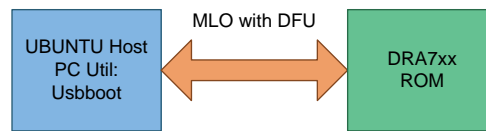


Figure 1. ROM Downloads MLO From Ubuntu Host PC

2. MLO on DRA7xx will download eMMC boot image via DFU from Ubuntu host.

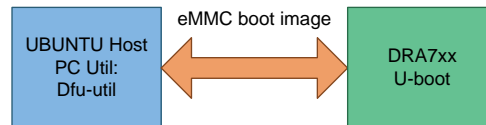


Figure 2. MLO Downloads Binaries From Ubuntu Host PC

Figure 3 shows the setup, where J6 EVM with the SW2 switch are set to the peripheral USB boot mode connected to the Ubuntu Host PC through a super speed (SS) USB device cable. From the Ubuntu host, the usbbboot is used to load the single stage bootloader (MLO+DFU) and dfu-util tool that are used to flash the boot images to the eMMC device.

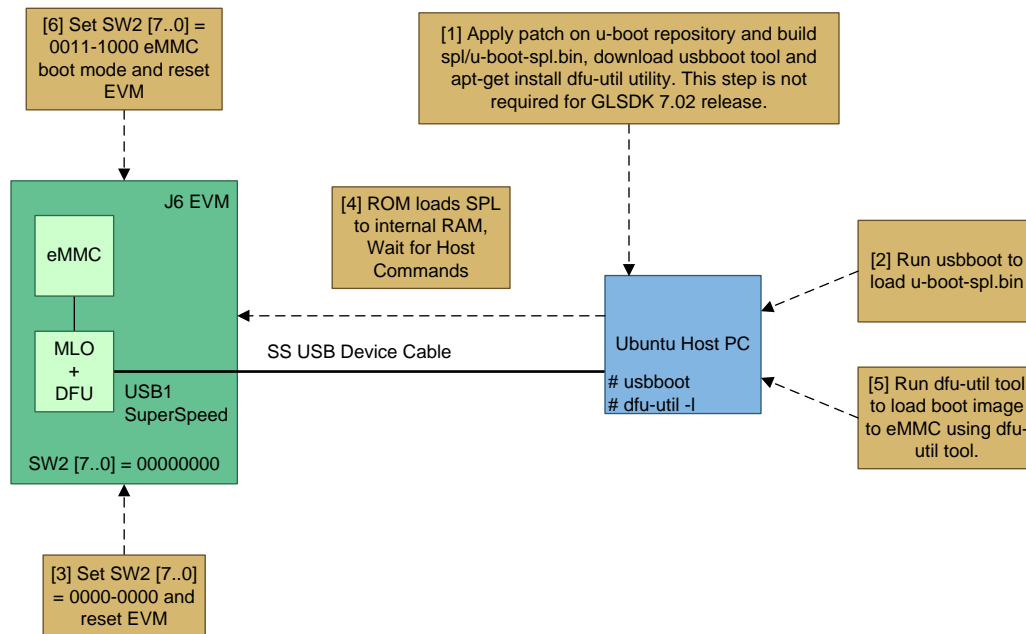


Figure 3. Setup of Flashing eMMC From ubuntu-host Using USB Interface

2 Building First Stage Bootloader With DFU Support

Git clone the U-boot repository as per the GLSDK Software Development User’s Guide . Apply the DFU patch from the [3] the link on top of GLSDK 7.01 release (3.14 kernel). No need to apply the patch for the GLSDK 7.02 release onwards. Run the command below:

```
# make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- distclean
# make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- dra7xx_evm_usbboot_config
# make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf-
```

The spl/u-boot-spl.bin will be created.

2.1 Creating Boot Image to Flash to eMMC From Ubuntu Host

Create bootable disk image from the MMC/SD card:

1. Install the GLSDK package and set the GLSDK environment variable to point to the GLSDK installation path.

```
# chmod +x ti-glsdk_dra7xx-evm_7_00_00_04_linux-installer.bin
# ./ti-glsdk_dra7xx-evm_7_00_00_04_linux-installer.bin
# export GLSDK="${HOME}/ti-glsdk_dra7xx-evm_7_01_00_03"
```

2. Insert the MMC/SD card to the Ubuntu machine and create the MMC/SD bootable card.

```
# sudo ${GLSDK}/bin/mksdboot.sh --device /dev/sdY --sdk ${GLSDK}
```

NOTE: This creates two partitions in the mmc/sd card: boot(vfat) and rootfs(ext4). Overwrite uenv.txt using uenv-emmc.txt in the boot partition before creating boot.img.

3. Create the SD card image and program it to eMMC.

```
# dd if=/dev/sdY of=boot.img bs=1M count=<size, 2048 for 2GB card>
```

3 ROM Downloading u-boot-spl.bin From Ubuntu Host

Download the usbboot host tool from the link to load the first-stage bootloader to target EVM [4], then run the command below from the Ubuntu host PC.

- # chmod +x usbboot
- #sudo ./usbboot -S u-boot-spl.bin

4 Setting Up EVM and Reset the Board

- Connect the EVM to the Ubuntu host machine using the USB superspeed device cable. Connect the USB device cable to the USB1 port (superspeed) of the EVM.
- Set the SYSBOOT setting to the peripheral USB boot mode.

```
SW2[7..0] = OFF OFF OFF OFF - OFF OFF OFF OFF
```

5 ROM Download the u-boot-spl.bin From Ubuntu Host

After executing step 4, issue reset to the board; the ROM loads u-boot-spl.bin from the host into internal RAM and executes. You can observe the message below on the host machine.

```
reading ASIC ID
CHIP: 5641
rom minor version: 02
IDEN: 0000000000000000000000000000000000000000000000000000000000000000
MPKH: 0000000000000000000000000000000000000000000000000000000000000000
CRC0: 51d2f9a7
CRC1: 00000000
device is GP
sending 2ndstage to target...
```

On Target console you will see console output similar to mentioned below:

```
U-Boot SPL 2014.07-01768-gbb8c815 (Apr 21 2015 - 12:27:16)
DRA722 ES1.0
set_config: high speed config #1: usb_download
```

6 Flashing the Boot Image to eMMC Device

Programming the Sdcard image (boot.img) to eMMC device.

- From Ubuntu PC
Install dfu-util if not installed on Ubuntu machine.

```
#sudo apt-get install dfu-util
# sudo dfu-util -l
```

The following output will be appear:

```
dfu-util 0.5
(C) 2005-2008 by Weston Schmidt, Harald Welte and OpenMoko Inc.
(C) 2010-2011 Tormod Volden (DfuSe support)
This program is Free Software and has ABSOLUTELY NO WARRANTY
dfu-util does currently only support DFU version 1.0
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=0, name="rawemmc"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=1, name="MLO.raw"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=2, name="u-boot.img.raw"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=3, name="spl-os-args.raw"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=4, name="spl-os-image.raw"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=5, name="spl-os-args"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=6, name="spl-os-image"
Application Note
Flashing binaries to factory boards using DFU
6
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=7, name="u-boot.img"
Found DFU: [0403:bd00] devnum=0, cfg=1, intf=0, alt=8, name="uEnv.txt"
```

The dfu-util command list the available number of interfaces with alternate interface for configuration #1. "cfg" stands for DFU configuration, "intf" stands for interface number, "alt" stands for alternate interface with specific name for each alternate interface.

- Load the boot.img created from [Section 4](#)

```
# sudo dfu-util -D boot.img -c 1 -i 0 -a 0
```

7 Executing From eMMC

1. Set the SYSBOOT setting to Automotive eMMC boot mode.

```
SW2[7..0] = OFF OFF ON ON - ON OFF OFF OFF
```

2. Reset the board, the board will boot from eMMC.

8 References

1. [DRA7xx GLSDK Software Developers Guide](#)
2. [GLSDK DFU Use's Guide](#)
3. [DFU Support in U-Boot SPL](#)
4. [usbboot Host Utility](#)

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Applications Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community

e2e.ti.com