

3.7.3. Flash Linux to eMMC

This section describes how to install a full Linux system to eMMC so that Linux can boot completely from eMMC.

The procedures below are all to be performed under Linux prompt, assuming that Linux has already booted properly from another memory device such as SD card or from an interface such as UART. There are three steps to achieve the goal:

- Flash U-Boot to eMMC boot0 partition
- Flash Linux kernel, device tree and root filesystem to eMMC user partition
- Change boot mode and U-Boot Env so that Linux boots from eMMC

3.7.3.1. Flash U-Boot to eMMC

First check and verify that eMMC is available on the EVM. For AM64x, eMMC is `mmcblk0` and command `ls /dev/mmcblk0*` should return the following:

```
root@am64xx-evm:~# ls -l /dev/mmcblk0*
brw-rw---- 1 root disk 179, 32 Nov 16 01:52 /dev/mmcblk0
brw-rw---- 1 root disk 179, 64 Nov 16 02:01 /dev/mmcblk0boot0
brw-rw---- 1 root disk 179, 96 Nov 16 01:44 /dev/mmcblk0boot1
crw----- 1 root root 237,  0 Nov 16 01:44 /dev/mmcblk0rpm
```

Next write U-Boot firmware to eMMC boot0 partition. In order to do that, we need to enable write access to boot0 partition. Please refer to [Linux kernel documentation](#) regarding this. Use the command below to enable write access:

```
$ echo 0 > /sys/block/mmcblk0boot0/force_ro
```

Once write access is enabled, U-Boot firmware can be written to eMMC boot0 partition. First copy the U-Boot binaries from the host to the EVM:

- tboot3.bin
- tispl.bin
- u-boot.img

Then use command `dd` to write the binaries to eMMC boot0. Please refer to [eMMC layout](#) to calculate the offset numbers for the `dd` command. For example, to write tispl.bin to offset 0x400 (1024), we need to use `seek = 1024`. Write all U-Boot binaries to boot0 as below:

```
$ dd if=tboot3.bin of=/dev/mmcblk0boot0 seek=0
$ dd if=tispl.bin of=/dev/mmcblk0boot0 seek=1024
$ dd if=u-boot.img of=/dev/mmcblk0boot0 seek=5120
```

3.7.3.2. Flash Kernel, Device Tree and Root Filesystem to eMMC

Linux kernel, device tree and root filesystem need to be written to eMMC user partition. First, an ext4 partition needs to be created. This can be done by the `fdisk` command:

```
$ fdisk /dev/mmcblk0
```

Use the following options for this interactive command:

- Enter “p” to display current partitions
- Enter “n” to create a new partition (or enter “d” first to delete existing partitions)
- Use default for all the subsequent options
- Enter “w” to write table to disk and exit

After this is finished, `ls /dev/mmcblk0*` should show a `/dev/mmcblk0p1` partition. For example:

```
root@am64xx-evm:~# ls -l /dev/mmcblk0*
brw-rw---- 1 root disk 179, 32 Nov 16 02:02 /dev/mmcblk0
brw-rw---- 1 root disk 179, 64 Nov 16 02:01 /dev/mmcblk0boot0
brw-rw---- 1 root disk 179, 96 Nov 16 01:44 /dev/mmcblk0boot1
brw-rw---- 1 root disk 179, 33 Nov 16 02:02 /dev/mmcblk0p1
crw----- 1 root root 237,  0 Nov 16 01:44 /dev/mmcblk0rpb
```

Now that an ext partition is created, the Linux kernel, device tree and root filesystem can be written to this partition. First copy the Linux image from the host to the EVM, for example, the default image for AM64x:

- `tisdsk-default-image-am64xx-evm.tar.xz`

Please refer to [Root Filesystem](#) regarding Linux images from the Processor SDK.

Then untar this image to eMMC user partition, for example:

```
root@am64xx-evm:~# mkdir -p /mnt/temp
root@am64xx-evm:~# mount -t ext4 /dev/mmcblk0p1 /mnt/temp
root@am64xx-evm:~# cd /mnt/temp
root@am64xx-evm:/mnt/temp# tar xf ~/tisdsk-default-image-am64xx-evm.tar.xz
root@am64xx-evm:/mnt/temp# cd ..
root@am64xx-evm:/mnt# umount temp
```

3.7.3.3. Change Boot Mode and U-Boot Env

Now that the complete Linux system has been installed to eMMC, the final step is to set up the EVM boot mode and change the U-Boot environment so that Linux will boot from eMMC.

Please refer to [EVM Boot Mode](#) for how to set the boot mode to eMMC.

Reboot the EVM and stop the autoboot to go to U-Boot prompt. Issue the following command to change the environment variables:

```
=> setenv mmcdev 0
=> setenv bootpart 0
=> saveenv
=> boot
```

Now Linux should boot from the eMMC.