Sourcing LED Lighting?
You Need The Leading Global B2B Platform
Alibaba.com
Global trade starts here.
Start Now ▸

Navigation

- Main page
- Recent changes
- Random page

Search

Search Go Search

Tools

- What links here
- Related changes
- Special pages
- Printable version
- Permanent link
- Page information

Personal tools

- Log in

By default, the ROM in the Sitara AM3359 will boot from the MMC1 interface first (the onboard eMMC), followed by MMC0 (MicroSD), UART0 and USB0.

If the boot switch (S2) is held down during power-up, the ROM will boot from the SPI0 Interface first, followed by MMC0, USB0 and UART0. This allows the BeagleBone Black to bypass the onboard eMMC and boot from the removable uSD (provided no valid boot device is found on SPI0.) This can be used to recover from a corrupted onboard eMMC.

The Sitara AM3359 will try to load and execute the first stage bootloader called "MLO" from a Fat 12/16 or 32 bit MBR based filesystem. If using eMMC, this file is loaded using RAW mode. This means the ROM looks for a TOC at four specific offsets.

MLO Booting (uBoot SPL Second Program Loader):

```
U-Boot SPL 2013.04-dirty (Jun 19 2013 - 09:57:14)
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
OMAP SD/MMC: 0
```

```
mmc_send_cmd : timeout: No status update
reading u-boot.img
reading u-boot.img
```

MLO is a first stage uBoot Bootloader designed to load a second stage uBoot bootloader with enhanced features. This second stage bootloader is also found on the FAT partition with the filename of "u-boot.img"

```

U-Boot 2013.04-dirty (Jun 19 2013 - 09:57:14)

I2C:   ready
DRAM:  512 MiB
WARNING: Caches not enabled
NAND:  No NAND device found!!!
0 MiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - readenv() failed, using default environment

musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Peripheral mode controller at 47401000 using PIO, IRQ 0
musb-hdrc: ConfigData=0xde (UTMI-8, dyn FIFOs, HB-ISO Rx, HB-ISO Tx, SoftConn)
musb-hdrc: MHDRC RTL version 2.0
musb-hdrc: setup fifo_mode 4
musb-hdrc: 28/31 max ep, 16384/16384 memory
USB Host mode controller at 47401800 using PIO, IRQ 0
Net:   <ethaddr> not set. Validating first E-fuse MAC
cpsw, usb_ether
Hit any key to stop autoboot:  1  0
gpio: pin 53 (gpio 53) value is 1
Card did not respond to voltage select!
mmc0(part 0) is current device
mmc_send_cmd : timeout: No status update
Card did not respond to voltage select!
No micro SD card found, setting mmcdev to 1
mmc1(part 0) is current device
gpio: pin 54 (gpio 54) value is 1
SD/MMC found on device 1
```

uBoot will load using a default environment space. This default space includes a variable bootenv=uEnv.txt and associated script that allows additional variables to be added or overwritten by adding them to an uEnv.txt file placed on the FAT partition. uBoot will attempt to load this file and append the extra variables:

```

reading uEnv.txt
26 bytes read in 2 ms (12.7 KiB/s)
Loaded environment from uEnv.txt
Importing environment from mmc ...
```

uBoot will then load the Linux Kernel and compiled Device Tree Binary blob from eMMC:

```

load mmc 1:2 0x80F80000 /boot/am335x-boneblack.dtb
load mmc 1:2 0x80007fc0 /boot/uImage
```

```

gpio: pin 55 (gpio 55) value is 1
4270840 bytes read in 747 ms (5.5 MiB/s)
gpio: pin 56 (gpio 56) value is 1
24129 bytes read in 53 ms (444.3 KiB/s)
Booting from mmc ...
```

And boot with the ext4 root filesystem being loaded from /dev/mmcblk0p2

```
setenv bootargs console=ttyO0,115200n8 ${optargs} root=/dev/mmcblk0p2 ro rootfstype=ext4 rootwait
bootm 0x80007fc0 - 0x80F80000
```

```
Booting from mmc ...
## Booting kernel from Legacy Image at 80007fc0 ...
   Image Name:   Angstrom/3.8.13/beaglebone
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    4270776 Bytes = 4.1 MiB
   Load Address: 80008000
   Entry Point:  80008000
   Verifying Checksum ... OK
## Flattened Device Tree blob at 80f80000
   Booting using the fdt blob at 0x80f80000
   XIP Kernel Image ... OK
OK
   Using Device Tree in place at 80f80000, end 80f88e40

Starting kernel ...

Uncompressing Linux... done, booting the kernel.
```

Retrieved from "http://wiki.beyondlogic.org/index.php?title=BeagleBoneBlack_Boot_Process&oldid=246"
| This page was last modified on 11 August 2013, at 08:25. | This page has been accessed 21,258 times. | About BeyondLogic