The Boot Process

From Texas Instruments Wiki
Jump to: navigation, search

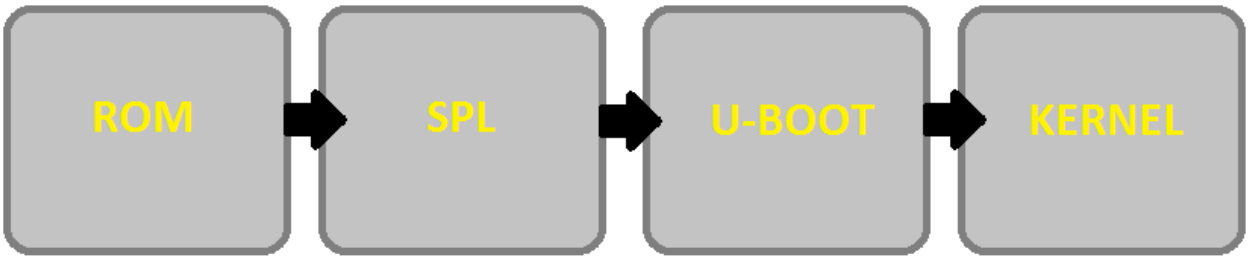
Contents

Overview - The 4 Bootloader Stages

Booting the Linux kernel on an embedded platform is not as simple as simply pointing a program counter to the kernel location and letting the processor run. This wiki will review the four bootloader software stages that must be run *before* the kernel can be booted and run on the device.

The AM335x is complex piece of hardware, but has limited internal RAM (128 kB). Because of this limited amount of RAM, multiple bootloader stages are needed. These bootloader stages systematically unlock the full functionality of the device so that all complexities of the device are available to the kernel.

The AM335x has four distinct bootloader stages:



The four bootloader stages are:

1) ROM
2) SPL (or Secondary Program Loader)
3) U-BOOT

4) Linux Kernel

1st Stage Bootloader: ROM Code

The first stage bootloader is housed in ROM on the device. The ROM code is the first block of code that is automatically run on device start-up or after power-on reset (POR). The ROM bootloader code is hardcoded into the device and cannot be changed by the user. Because of this, it is important to get an understanding of what exactly the ROM code is doing.

The ROM code has two main functions:

- Configuration of the device and initialization of primary peripherals

  - Stack setup
  - Configure Watchdog Timer 1 (set to three minutes)
  - PLL and System Clocks configuration

- Ready device for next bootloader

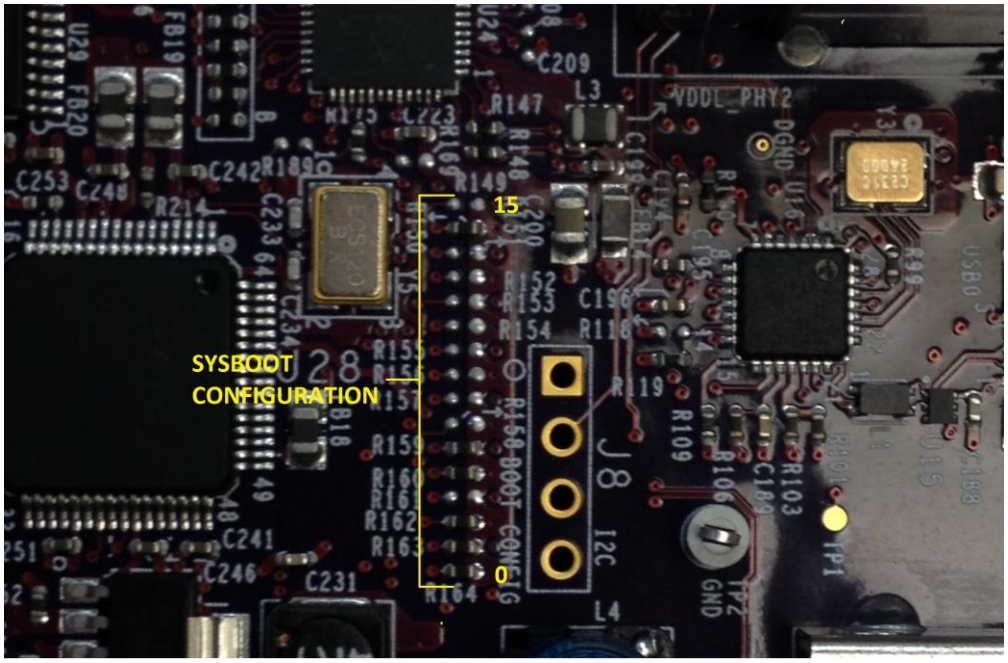
  - Check boot sources for next bootloader (SPL)
  - Moves next bootloader code into memory to be run

The main purpose of the ROM code is to set up the device for the second stage bootloader. The list of booting devices that the ROM code will search through for the second stage bootloader is configured by the voltage levels set on the AM335x SYSBOOT [15:0] pins on startup. These pins also set other boot parameters (i.e. expected crystal frequency, bus width of external memory). For more information on these other boot parameters, see Chapter 26 of the Technical Reference Manual Coloured text

On the Starter Kit EVM, there is no *easy* way to control the boot device list, as the SYSBOOT pins are at fixed voltages where:

```
SYSBOOT[15:0] = 0100000000000110111b
```

The SYSBOOT pins configure the boot device order (set by SYSBOOT[4:0]) for MMC,SPI0, UART0, USB0. For the curious, the SYSBOOT pin pullup/pulldown resistors can be found on the bottom of the Starter Kit EVM near the J8 I2C header. From the picture below, the unpopulated pads correlate to sysboot pins that are pulled down (10K Ohm), while the populated pads are pullups (100K Ohm).Note: The Starter Kit **DOES NOT** support boot from flash.



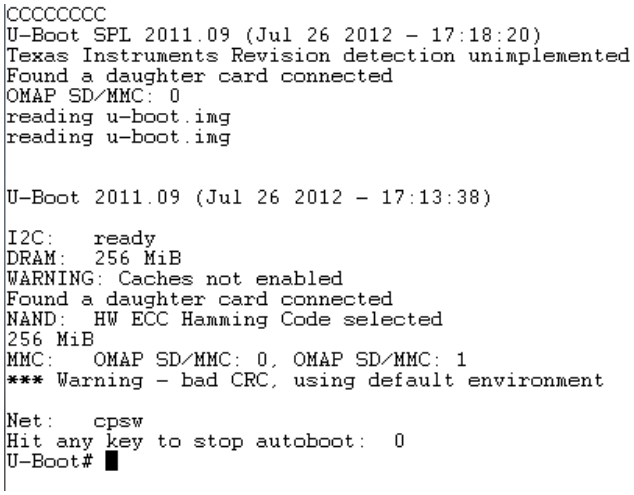
On the General purpose EVM, the SYSBOOT pins can be freely configured using the SYSBOOT Dip Switches (SW3+SW4). The Dip switches and their relation to the SYSBOOT pins can be seen below.

2nd Stage Bootloader: SPL

The second stage bootloader is known as the SPL, but is sometimes referred to as the MLO. The SPL is the first stage of U-boot, and must be loaded from one of the boot sources into internal RAM. The SPL has very limited configuration or user interaction, and mainly serves to set-up the boot process for the next bootloader stage: U-boot.

3rd Stage Bootloader: U-BOOT

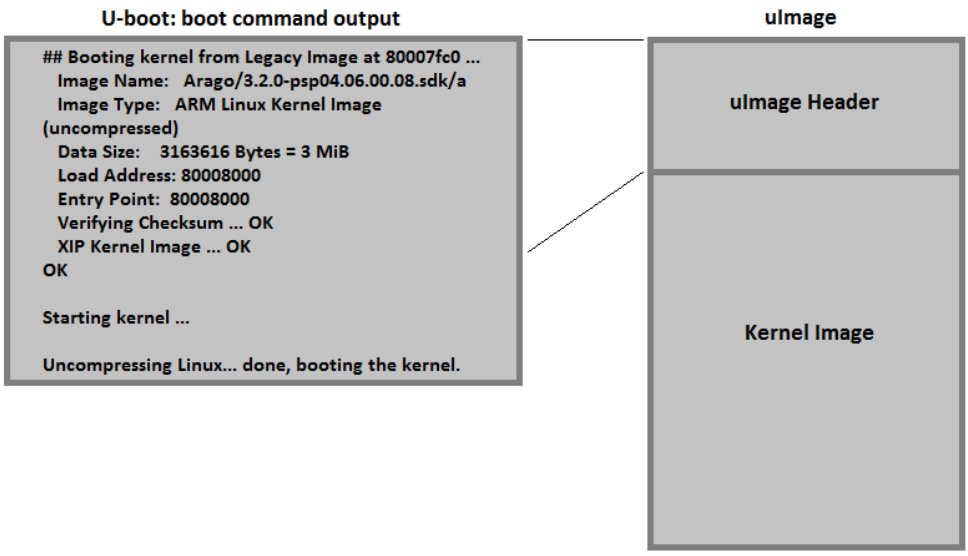
U-BOOT allows for powerful command-based control over the kernel boot environment via a serial terminal. The user has control over a number of parameters such as boot arguments and the kernel boot command. In addition, U-boot environment variables can be configured. These environment variables are stored in the uEnv.txt file on your storage medium. These environment variables can be viewed, modified, and saved using the printenv, setenv, and saveenv commands, respectively. Below is an example screenshot from a serial terminal first running U-BOOT.



For U-boot command examples, please see the AM335x U-Boot User's Guide (link).

4th Stage Bootloader: Linux Kernel

uImage is the kernel image wrapped with header info that describes the kernel. This header is a 64kB block of information that includes the target architecture, the operating system, kernel size, entry points, etc. When booting uImage via U-boot, the header information is displayed on the command output.



After reading and outputting the header file, u-boot will begin to boot the linux kernel. If you encounter an issue in this stage of the boot process, please consult "Kernel - Common Problems Booting Linux" (link)

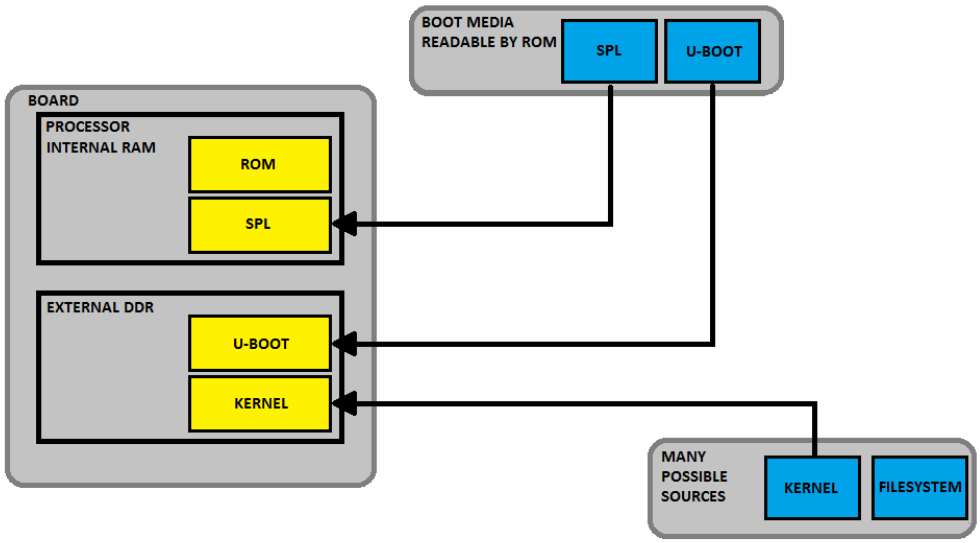
More Bootloader Information

Sourcing Bootloaders: Boot Devices

The AM335x can source bootloaders from a variety of sources/media including: MMC/SD, NAND, NOR, UART, ETHERNET, USB, SPI, and I2C. Which bootloader sources are available depends on the bootloader stage. As an example, SPL/U-BOOT can only be booted from boot media that the ROM code can recognize (both SPl and U-BOOT need to be flashed on the same media). The kernel and filesystem can be sourced from a number of boot locations.

Once a bootloader is sourced, it is important to understand where the bootloader is loaded into memory. Processors like the AM335x define a memory map that states where both internal memory and DDR are mapped to an address set. The processor expects the external bootloaders (SPL,U-BOOT, and the Linux kernel) to be loaded into specific memory locations. However, once started, these bootloaders may relocate themeselves in memory. More info on memory map can be found in Chapter 2 of the TRM (link to TRM)

ROM and the SPL bootloaders make use of the internal RAM to run, while U-boot and the kernel are expected in DDR (addresses 0x80000000 and 0x82000000, respectively). This is illustrated in the image below:



Navigation menu

Personal tools

- Log in
- Request account

Namespaces

- Page
- Discussion

Variants

Views

- Read
- View source
- View history

More

Search

Search Go

Navigation

- Main Page
- All pages
- All categories
- Recent changes
- Random page
- Help

Toolbox

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