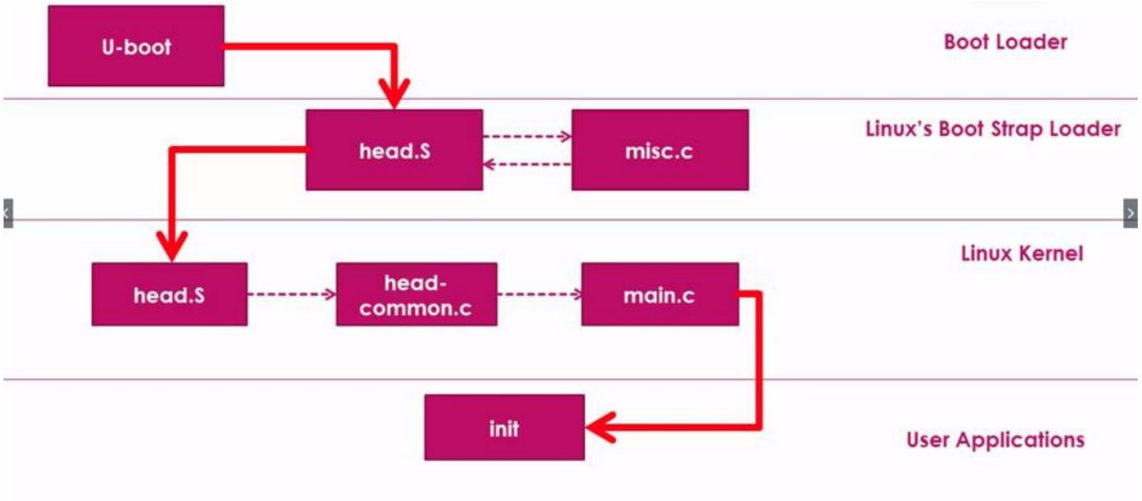
The Linux Boot Sequence



The above diagram shows the control flow of the boot process.

The steps involved are.

- CPU specific initializations
- Checks for valid processor architecture.
- Page table inits.
- Initialize and prepare MMU for the identified Processor Architecture.
- Enable MMU to support virtual memory.
- Calls "start_kernel" function of the main.c ("Arch" independent code).

To understand the handover from uboot to the boot strap loader we need to go through some files involved in the different stages of the boot process.

First Stage

- File Location: *u-boot-2017.05-rc2/arch/arm/lib*
- First open the file bootm.c.
- In this file we have a function "kernel_entry". This function holds 2 very important arguments i.e., r1 = machine id, r2 = DTB address.
- These values are passed on to Boot strap loader through the "kernel_entry" function.

Second Stage

- File Location: *Embedded_Linux/linux-4.14$ vi arch/arm/boot/compressed/*
- Coming to the Boot strap Loader we have two file – head.S and misc.c.
- The role of head.S file is to catch the r1 and r2 values and initiate the Linux decompression.

- For this there is a branch line named “decompress_kernel” which takes the flow to the misc.c file where the decompression of the kernel is to happen.
- The flow comes back to the head. S file where the function “enter_kernel” sends the control flow to the third stage.

Third Stage

- File Location: *Embedded_Linux/linux-4.14$ vi arch/arm/kernel/*
- In this stage three files are generated – head.S, head-common.S, and the main.c.
- The role of the head.S file is to disable the MMU, setup the initial page tables to the barest amount, which is required to get the kernel running, which generally means mapping in the kernel code and reinitialize the MMU.
- The flow then goes to the head-common.S file.
- The following fragment of code is executed with the MMU on in MMU mode and uses absolute addresses; this is not position independent.
- The absolute address can be obtained by looking up in the lookup_procssor_type.