Bootloader

D. Balakrishna,
Research Associate, IIIT-H

Bootloaders are important when we are developing embedded systems, depending on the capabilities of the board and processor on which an embedded system is based. The first program that will run whenever the system is power on is the Bootloader. A bootloader is a small application that is executed when a system is powered on, loads an executable kernel image into memory and then begins its execution. The Bootloader is a small piece of machine dependent code responsible for hardware initialization, load kernel image into memory and start kernel execution, while loading the kernel image the boot loader might also need to load an initial root file system and make that available to the kernel. This is especially true when using Linux on an embedded system which does not have a storage area for the file system. The bootloaders are separate for Desktop machines and embedded devices.

Desktop/ x86 specific bootloaders

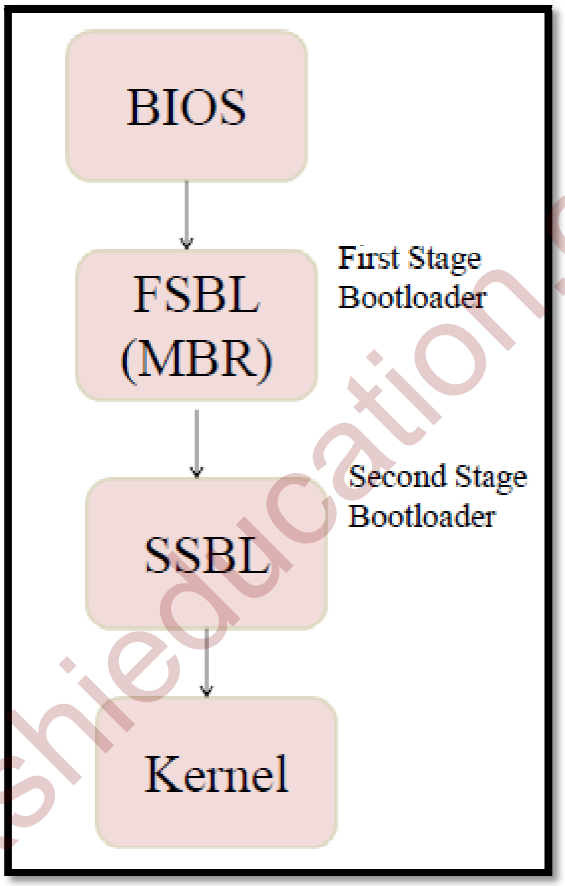
Whenever we power on our Desktop machine a piece of code called BIOS (Basic Input/output Software) is responsible for hardware initialization and loading executable image into memory. A single stage booltloaders initializes the hardware and loads the kernel image from flash/ROM or storage device and begins its execution. In a multi stage bootloader the BIOS loads the first 512 bytes from MBR (Master Boot Record) is the first stage bootloader. The MBR is the important component on the software side in the boot procedure on BIOS-based machines. The first stage bootloader (MBR) loads the second stage bootloader and so on eventually loading the executable kernel image.

Now days most of the Linux distributions for x86 architectures use two different types of bootloaders depending on the boot media and boot process that you want support.

1. **GRUB (Grand Unified Boot Loader):**

   The most common and powerful bootloader used on Linux desktop and Server systems is GRUB is a multi-stage, chain loading bootloader that is most

commonly used to facilitate interaction with the boot process. The GRUB can read many file system formats to load kernel image and the configurations, provide a powerful shell with various commands and can load kernel images over the network also.



2. Syslinux bootloder:

Syslinux bootloader is used to enable booting from network and removable media like USB, CD-ROM and DVD. It is light weight multi stage bootloader that supports booting from DOS/Windows or Linux file systems.

3. LILO (Linux loader):

LILO was the default bootloader for many Linux distributions before GRUB became popular. LILO can also boot Linux kernel images from floppy disks and hard drives. The advantage LILO has over many bootloaders is it is not file system specific.

www.sakshieducation.com

Boot loaders for Embedded Devices

There are several open source generic bootloaders are available for embedded CPUs like U-boot, Redboot, Yaboot, barebox U-boot, etc. In this article we will study U-Boot bootlaoder.

U-Boot boot loader

Das U-boot also known as the Universal Boot Loader or u-boot, is an open source boot loader that supports a wide range of different architectures such as ARM, PowerPC, XScale, MIPS , MicroBlaze and also used on x86 systems. U-boot Loaded by FSBL from DDR, and it is responsible to load another application through a serial, an Ethernet and Flash memories. U-boot flexible configuration makes it easy to customize for specific board and processor requirements while keeping the bootloader as small as possible. The u-boot bootloader has actually become the most widely used boot loader on ARM based systems. Besides supporting a wide range of architectures the u-boot also supports a wide range of booting options.

Below is a list of some of these booting options:

- From Flash memory (for example NOR or NAND)
- From a USB mass storage device
- From an MMC/SD memory card
- From a Hard Disk or CDROM
- Using Ethernet: TFTP, BOOTP, DHCP or NFS
- Using a serial connection

A booting option means a location from where the u-boot searches for the kernel image to load. If a MMC/SD card has been selected the u-boot will initialize the memory card controller and try to read the images from that device.

U-boot Features

- Auto boot will automatically boot the system on power on and reset the hardware board.
- It supports variety of commands to load kernel image.
- Files can be downloaded through serial communication.

www.sakshieducation.com

- It supports all network commands like ping, tftp, nfs and dhcp.
- We can set and save the environment variables and even print environmental variables with respective command.
- It support NAND/NOR flash, SD/MMC, USB flashes.

U-Boot Configuration

Steps for configuring U-boot: we are referring arndale 5250 board.

- Download the bootloader source code from the linaro website for arndale board.
- The include/configs/ directory contains one configuration file for each supported board.
    - It defines the CPU type, the peripherals and their configurations, memory mapping, etc.
    - It is a simple .h file consists of C pre-processor constants. We can also add or remove some features to this file. For any help use the README file available.
- U-boot must be configured before compilation.
    - Make “boardname_config” -- make arndale5250 – available in the board.cfg file.
- Compile the U-boot by specifying cross compiler path.
    - export CROSS_COMPILE=arm-linux-gnueabihf-
    - export ARCH=arm
    - make arndale5250
- It will create u-boot.bin file which is u-boot image. We need port this image into the board.
- U-boot must be installed in flash memory to be executed by hardware.
- Connect the target to host machine through serial console.
- Powers up the board and on serial console we will see the following script.

Welcome to minicom 2.5

OPTIONS: I18n

Compiled on May 2 2011, 00:39:27.

Port /dev/ttyUSB0

www.sakshieducation.com

Press CTRL-A Z for help on special keys

U-Boot 2013.01.-rc1 (Jul 10 2013 - 15:16:09) for ARNDALE5250

CPU: Exynos5250@1000MHz

Board: for ARNDALE5250
I2C: ready
DRAM: 2 GiB
WARNING: Caches not enabled

Checking Boot Mode ... SDMMC
MMC: EXYNOS DWMMC: 0, EXYNOS DWMMC: 1, EXYNOS DWMMC: 2
In: serial
Out: serial
Err: serial
Net: No ethernet found.
(Re)start USB...
USB0: USB EHCI 1.00
scanning bus 0 for devices... 4 USB Device(s) found
scanning usb for storage devices... 0 Storage Device(s) found
scanning usb for ethernet devices... 1 Ethernet Device(s) found
Hit any key to stop autoboot: 0 //

Here it will wait for 3 seconds and asking press any key to set boot arguments manually otherwise it will boot automatically with default boot arguments. The u-boot version we are using is U-Boot 2013.01.-rc1 for arndale board.

www.sakshieducation.com

U-boot Commands

U-boot has a set of built in commands for booting the system, managing the memory and updating the firmware. For a complete list and description of command can be known by typing "help" or "?" at boot prompt.

```
ARNDALE5250 # help
help - print command description/usage

Usage:
help
      - print brief description of all commands
help command ...
            –  print detailed usage of 'command'
            –
Arndale #
Arndale # ?
?      - alias for 'help'

Arndale # help
?      - alias for 'help'
base   - print or set address offset
bdinfo  - print Board Info structure
boot   - boot default, i.e., run 'bootcmd'
bootd   - boot default, i.e., run 'bootcmd'
bootelf - Boot from an ELF image in memory
bootm   - boot application image from memory
bootvx  - Boot vxWorks from an ELF image
cmp    - memory compare
coninfo - print console devices and information
cp     - memory copy
```

www.sakshieducation.com

crc32 - checksum calculation
dcache - enable or disable data cache
dnw - dnw - initialize USB device and ready to receive for Windows serv)
echo - echo args to console
editenv - edit environment variable
emmc - Open/Close eMMC boot Partition
env - environment handling commands
exit - exit script
ext2format- ext2format - disk format by ext2
ext2load- load binary file from a Ext2 filesystem
ext2ls - list files in a directory (default /)
ext3format- ext3format - disk format by ext3
false - do nothing, unsuccessfully
fastboot- fastboot- use USB Fastboot protocol
fatformat- fatformat - disk format by FAT32
fatinfo - fatinfo - print information about filesystem
fatload - fatload - load binary file from a dos filesystem
fatls - list files in a directory (default /)
fdisk - fdisk - fdisk for sd/mmc.
go - start application at address 'addr'
help - print command description/usage
icache - enable or disable instruction cache
iminfo - print header information for application image
imxtract- extract a part of a multi-image
itest - return true/false on integer compare
loadb - load binary file over serial line (kermit mode)
loads - load S-Record file over serial line
loady - load binary file over serial line (ymodem mode)
loop - infinite loop on address range
md - memory display
md5sum - compute MD5 message digest

www.sakshieducation.com

mm - memory modify (auto-incrementing address)
mmc - MMC sub system
mmcinfo - mmcinfo <dev num>-- display MMC info
movi - movi - sd/mmc r/w sub system for SMDK board
mtest - simple RAM read/write test
mw - memory write (fill)
nm - memory modify (constant address)
printenv- print environment variables
reginfo - print register information
reset - Perform RESET of the CPU
run - run commands in an environment variable
saveenv - save environment variables to persistent storage
setenv - set environment variables
showvar - print local hushshell variables
sleep - delay execution for some time
source - run script from memory
test - minimal test like /bin/sh
true - do nothing, successfully
usb - USB sub-system
version - print monitor version

Understanding Environment variables

U-boot can be configured through environment variables, which affect the behavior of the different commands. Environment variables are loaded from flash to RAM at U-Boot boot up and can be modified and saved back to flash memory.

Commands to manipulate environment variables

Printenv: shows all the environment variables previously stored on the board. The following are the default environment variables stored in the board.

Printenv < variable name > -- shows the value of one variable.

www.sakshieducation.com

setenv < variable name > < variable value > -- changes the value of variable only in the RAM.
Saveenv – saves the current state of environment variable into flash.

Arndale # printenv
baudrate=115200
bootcmd=run bootcmd_normal;run bootcmd_extend
bootcmd_extend=mmc read 1 40008000 7800 8; source 40008000
bootcmd_normal=movi read kernel 0 40008000;movi read rootfs 0 41000000 100000;b0
bootdelay=1

Environment size: 246/16380 bytes

Scripts in environment variables

Environment variables can contain scripts which will used to execute several commands and test the results of the commands.

- Useful to automate booting process.
- Several commands can be chained using the operator ";"
- Scripts are executed using run <variable name>

Example: "setenv step1 'ext2load mmc 0:2 $kernel_addr_r $kernel_path; ext2load mmc 0:2 $xen_addr_r $xen_path; ext2load mmc 0:2 $dtb_addr_r $dtb_path'"

Transferring files to the Target board

U-Boot is used to load and boot the kernel image and also it allows changing kernel image and the root file system stored in flash. The images and files can be exchanged between the host system and target board, this is possible through the network if the target board has Ethernet connection. This method is the fastest and efficient solution.

TFTP (Trivial File Transfer Protocol)

A tftp server s used to transfer files from workstation to target board and vice versa. A tftp server is installed on the host machine.

www.sakshieducation.com

- Sudo apt-get install tftpd-hpa
- All files are visible in /var/lib/tftpboot
- A tftp client is available in the tftp-hpa package for testing.
- A tftp client is integrated into u-boot.
    - Configure the ipaddr and serverip environment variables.
    - Use tftp <address> <filename> to load a file.

The kernel images that u-boot loads and boots must be prepared, so that u-boot specific header is added in front of the image. This header gives the details such as image size, load address and the type compression. This is done by tool in u-boot called mkimage. Using mkimage we will generate target specific file uImage of the kernel suitable for u-boot.

Examples of u-boot commands

Arndale #
ARNDALE5250 #
Arndale # version
U-Boot 2010.12 (Jun 25 2014 - 14:02:11) for Insignal Arndale

Arndale # env
env - environment handling commands
Usage:
env default -f - reset default environment
env edit name - edit environment variable
env export [-t | -b | -c] addr [size] - export environmnt
env import [-d] [-t | -b | -c] addr [size] - import environmnt
env print [name ...] - print environment
env run var [...] - run commands in an environment variable
env save - save environment
env set [-f] name [arg ...]

www.sakshieducation.com

Arndale # help boot

boot - boot default, i.e., run 'bootcmd'

Usage:

boot

ARNDALE5250 # bdinfo

arch_number = 0x00000EBE

boot_params = 0x40000100

DRAM bank = 0x00000000

-> start = 0x40000000

-> size = 0x10000000

DRAM bank = 0x00000001

-> start = 0x50000000

-> size = 0x10000000

DRAM bank = 0x00000002

-> start = 0x60000000

-> size = 0x10000000

DRAM bank = 0x00000003

-> start = 0x70000000

-> size = 0x10000000

DRAM bank = 0x00000004

-> start = 0x80000000

-> size = 0x10000000

DRAM bank = 0x00000005

-> start = 0x90000000

-> size = 0x10000000

DRAM bank = 0x00000006

-> start = 0xA0000000

-> size = 0x10000000

DRAM bank = 0x00000007

-> start = 0xB0000000

www.sakshieducation.com

-> size = 0x10000000
ethaddr = 02:75:2e:19:a9:04
ip_addr = <NULL>
baudrate = 115200 bps
TLB addr = 0xBFFF0000
relocaddr = 0xBFF77000
reloc off = 0x7C177000
irq_sp = 0xBFE6EF30
sp start = 0xBFE6EF20
FB base = 0x00000000
Arndale # mmc
Usage:
MMC sub system
Arndale # mmcinfo
Device: S5P_MSHC2
Manufacturer ID: 82
OEM: 4a54
Name: SD16G
Tran Speed: 0
Rd Block Len: 512
SD version 2.0
High Capacity: Yes
Size: 14911MB (block: 30537728)
Bus Width: 4-bit
Boot Partition Size: 0 KB
Arndale #
Arndale # sleep
sleep - delay execution for some time

Usage:
sleep N

www.sakshieducation.com

- delay execution for N seconds (N is _decimal_ !!!)

Arndale # iminfo

Checking Image at c3e00000 ...

Unknown image format!

Arndale # usb

usb - USB sub-system

Usage:

usb reset - reset (rescan) USB controller

usb tree - show USB device tree

usb info [dev] - show available USB devices

Arndale #

Arndale # iminfo

Checking Image at c3e00000 ...

Unknown image format!

Arndale # showvar

HUSH_VERSION=0.01

Arndale #

www.sakshieducation.com