U-Boot

- U-boot is an opensource universal bootloader for embedded [know as Das U-Boot].
- Loaded by FSBL from DDR, and it is responsible to load an another application through a serial, an ethernet, and Flash memories.
- Performed board, CPU and particular flash specific initialization & configurations.
- Also parses different types of filesystems on many types of storage devices.
- Responsible to pass some setoff parameters called boot parameters to kernel.

U-boot Features :

Source tree

Well structured tree as like Linus' tree, called ad Denx tree

Autoboot

Will automatically boot the system on power up or reset of the board

Environment variables

Env. variables can set, save and even print with respective commands.

Networking

Supports all possible n/w commands like ping, dhcp, tftp and nfs

O/S loading

Supports variety of commands to load an O/S

Flash support

Can read parallel NOR, NAND, SD/MMC, serial NOR, USB flashes

Serial download

Files can be loaded through serial via loady, loadb

U-Boot commands :

- U-Boot has a set of built-in commands for booting the system, managing memory, and updating an embedded system's firmware.
- For a complete list and brief descriptions of the built-in commands, he U-Boot prompt, enter either of these commands
  MINI2440 # help (or) ?

Important commands:

- Flash information (NOR and SPI flash)
  flinfo
- NAND flash information
  nand info
- Board info structure.
  bdinfo
- Shows NAND bad blocks
  nand bad

boot - runs the default boot command, stored in bootcmd
bootm <address> - starts a kernel image loaded at the given address in RAM
fatload - loads a file from a FAT filesystem to RAM
tftp - loads a file from the network to RAM
ping - to test the network

cp[.b, .w, .l]	source target count	- Copies memory contents from address source to target for as many count bytes, words, or long words.

nand[erase, read, write] - nane read/write commands

Environment variables :

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

Commands to manipulate environment variables:

printenv - shows all variables
printenv <variable-name> - shows the value of one variable
setenv <variable-name> <variable-value> - changes the value of a variable, only in RAM.
saveenv - saves the current state of the environment to flash

Important U-Boot env variables :

bootcmd - Defines a command string that is automatically executed when the initial countdown is not interrupted.Executed only when the bootdelay variable is also defined.
bootdelay - Seconds to wait before running the automatic boot process in bootcmd.
bootargs - contains the arguments passed to the Linux kernel
serverip - the IP address of the server that U-Boot will contact for network related commands .
ipaddr - the IP address of target on which u-boot running
ethaddr - the MAC address, can only be set once
autostart - if yes, U-Boot starts automatically an image that has been loaded into memory.
baudrate - a decimal number that selects the console baudrate (in bps).
mtdparts - This variable allows to share a common MTD partition scheme between U-Boot and the Linux kernel.
hostname - Target hostname

Building u-boot for mini2440 board t:

```
MINI2440 # printenv
bootargs=root=/dev/mtdblock3 rootfstype=jffs2 console=ttySAC0,115200
bootcmd=
bootdelay=3
baudrate=115200
ethaddr=08:08:11:18:12:27
ipaddr=10.0.0.111
serverip=10.0.0.4
netmask=255.255.255.0
usbtty=cdc_acm
mtdparts=mtdparts=mini2440-nand:256k@0(u-boot),128k(env),5m(kernel),-(root)
mini2440=mini2440=0tb
bootargs_base=console=ttySAC0,115200 noinitrd
bootargs_init=init=/sbin/init
root_nand=root=/dev/mtdblock3 rootfstype=jffs2
root_mmc=root=/dev/mmcblk0p2 rootdelay=2
root_nfs=/mnt/nfs
set_root_nfs=setenv root_nfs root=/dev/nfs rw nfsroot=${serverip}:${root_nfs}
ifconfig_static=run setenv ifconfig ip=${ipaddr}:${serverip}::${netmask}:mini24400
ifconfig_dhcp=run setenv ifconfig ip=dhcp
ifconfig=ip=dhcp
set_bootargs_mmc=setenv bootargs ${bootargs_base} ${bootargs_init} ${mini2440} ${}
set_bootargs_nand=setenv bootargs ${bootargs_base} ${bootargs_init} ${mini2440} $}
set_bootargs_nfs=run set_root_nfs; setenv bootargs ${bootargs_base} ${bootargs_in}
mtdids=nand0=mini2440-nand
partition=nand0,0
mtddevnum=0
mtddevname=u-boot

Environment size: 1089/131068 bytes
MINI2440 #
```

- **Download u-boot source**

git clone git://repo.or.cz/u-boot-openmoko/mini2440.git

- Configure u-boot for target board

veda@linux # cd mini2440

veda@linux # make mini2440_config

- Building u-boot for target board

veda@linux # make CROSS_COMPILE=arm-linux-

NOTE: export cross compilation toolchain path **PATH=$PATH:$(PATH_OF_CROSSTOOL)**

```
root@linux:~/elinux/workspace# ls
buildroot-2013.08.1  linux-3.9  rootfs.img  yaffs2
busybox-1.21.1       rootfs     rootfs.ubi
root@linux:~/elinux/workspace# tar -xf ../elinuxsrc/u-boot-mini2440.tar.xz
root@linux:~/elinux/workspace# ls
buildroot-2013.08.1  linux-3.9  rootfs.img  u-boot-mini2440
busybox-1.21.1       rootfs     rootfs.ubi  yaffs2
root@linux:~/elinux/workspace# cd u-boot-mini2440/
root@linux:~/elinux/workspace/u-boot-mini2440# ls
api          config.mk  drivers    include      mkconfig  rules.mk
arch         COPYING    dts        lib          nand_spl  snapshot.commit
board        CREDITS    examples   MAINTAINERS  net       spl
boards.cfg   disk       fs         MAKEALL      post      test
common       doc        helper.mk  Makefile     README    tools
root@linux:~/elinux/workspace/u-boot-mini2440# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local
/games:/root/elinux/workspace/buildroot-2013.08.1/output/host/usr/bin/
root@linux:~/elinux/workspace/u-boot-mini2440# make mini2440_config
Configuring for mini2440 board...
root@linux:~/elinux/workspace/u-boot-mini2440# make CROSS_COPILE=arm-linux-
```

- After a successful compilation, you should get the following U-Boot images.

1. **u-boot.bin** - is a raw binary image
2. **u-boot** - is an image in ELF binary format
3. **u-boot.srec** - is in Motorola S-Record format

```
root@linux:~/elinux/workspace/u-boot-mini2440# ls
api          COPYING  examples     MAKEALL   README          tools
arch         CREDITS  fs           Makefile  rules.mk        u-boot
board        disk     helper.mk    mkconfig  snapshot.commit u-boot.bin
boards.cfg   doc      include      nand_spl  spl             u-boot.lds
common       drivers  lib          net       System.map      u-boot.map
config.mk    dts      MAINTAINERS  post      test            u-boot.srec
root@linux:~/elinux/workspace/u-boot-mini2440#
```