

8. 부트로더

부팅 과정 소개
U-Boot 부트로더 실행
바이너리 이미지 실행

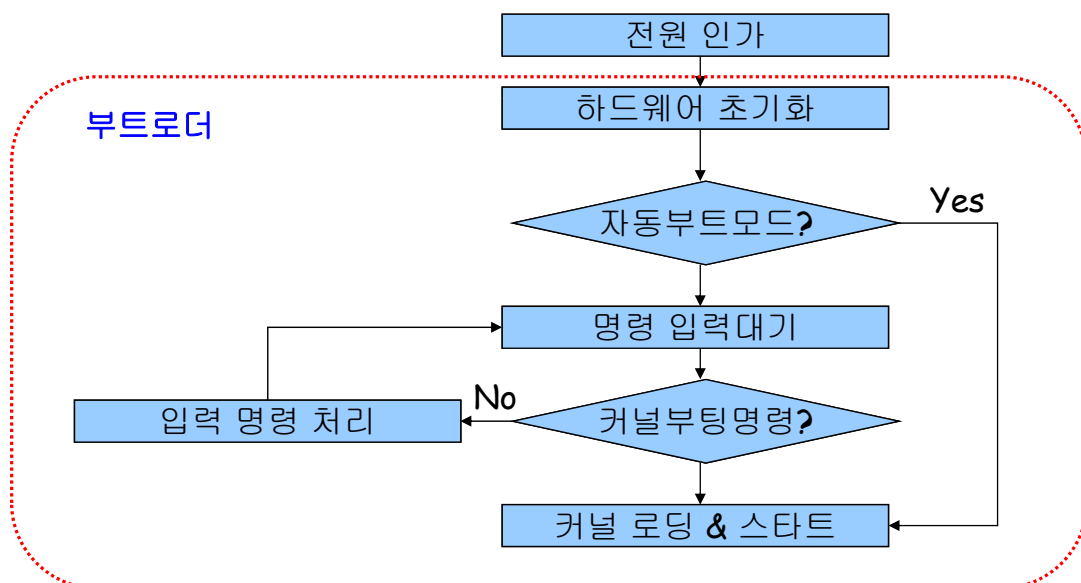
부팅 과정 소개

부트로더(Boot Loader)

- ❑ 시스템의 하드웨어를 초기화하고 운영체제의 커널을 메모리에 올려 실행시키는 시스템 프로그램
- ❑ 부트로더의 기능
 - 메모리 초기화
 - 하드웨어 초기화
 - 직렬포트의 초기화
 - 네트워크 초기화
 - 프로세서 속도, 인터럽트의 초기화
 - 커널과 루트 파일 시스템 적재
 - 커널과 루트 파일 시스템을 RAM에 적재하여 실행
- ❑ ODROID-X2
 - 부트로더로 U-Boot을 사용

부트로더의 동작 (1)

- ❑ 부트로더의 실행 순서



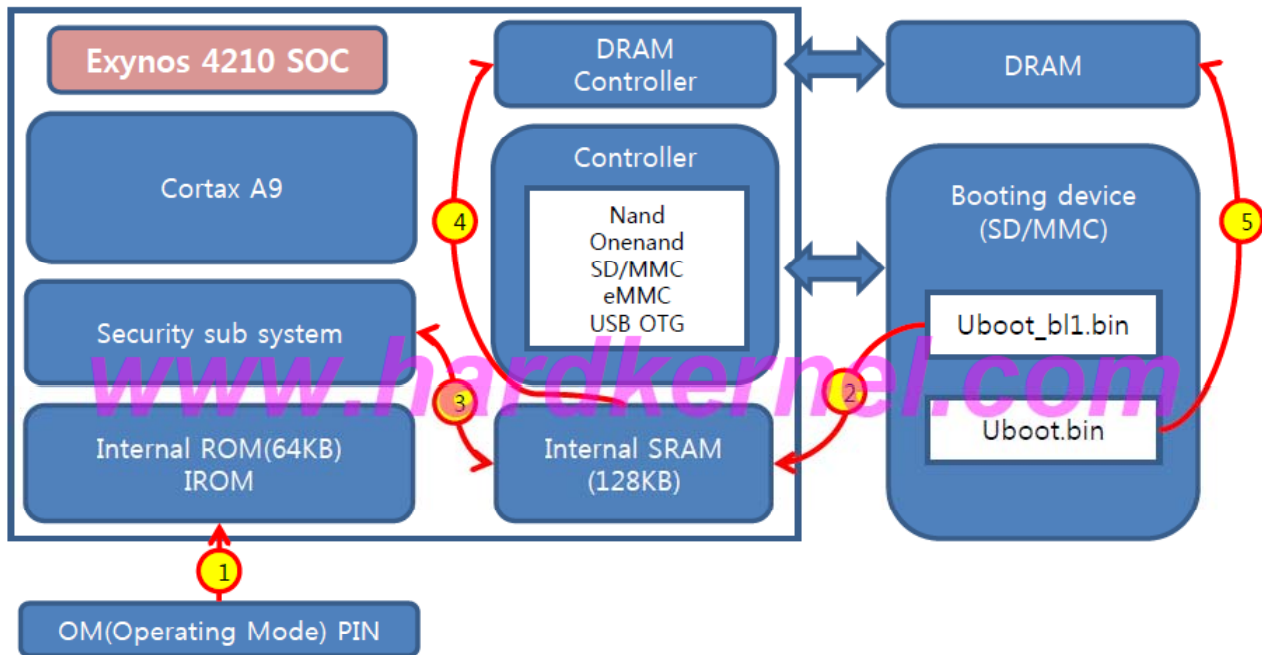
부트로더의 동작 (2)

- ❑ 하드웨어 초기화
 - 프로세서 및 메모리 초기화
 - 네트워크 및 시리얼 포트등의 디바이스 초기화
- ❑ 자동 부트
 - 커널 등의 프로그램 이미지를 로딩하고 실행 시킴
 - 자동 부트가 아닌 경우, 사용자 입력 대기 상태
- ❑ 입력 명령 처리
 - 간단한 수준의 디버깅
 - IP 주소 설정 및 자동 부팅설정 등의 보드 환경 설정
 - 각종 상태 체크
- ❑ 커널 로딩 및 스타트
 - 플래시에 저장되어 있는 커널 및 루트 파일 시스템을 타겟 메모리에 로딩
 - 커널 프로그램의 시작번지로 점프 하여 커널 부팅

ODROID 부팅 과정 - 부트로더 적재 (1)

1. 전원이 켜지면 OM(Operating Pin) 핀을 읽어 부트 디바이스 결정
2. 부트 디바이스로 부터 BL1 영역의 16KB를 내부 SRAM (iRAM)에 복사
 - BL1은 U-Boot의 첫번째 부트로더로 플랫폼 독립의 작은 코드로 두번째 부트로더(BL2)를 적재
3. 보안 서브 시스템이 16KB BL1의 체크섬(checksum) 검사
4. 적재된 BL1으로 부터 시스템 클럭 및 DRAM 컨트롤러 초기화
5. 부트로더(U-Boot, BL2)를 DRAM에 적재하고 실행

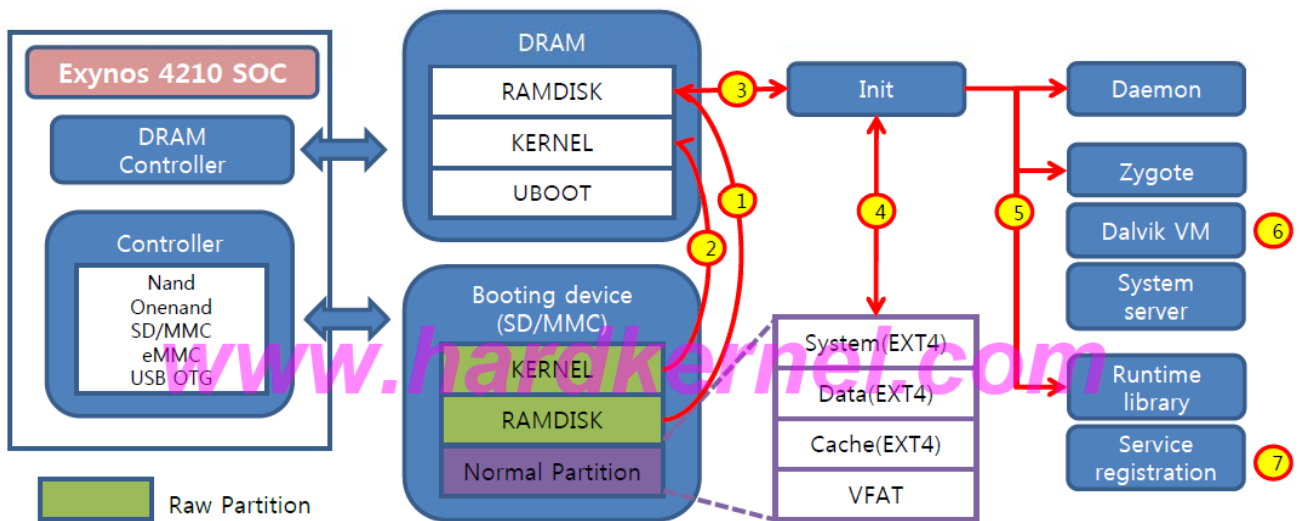
ODROID 부팅 과정 - 부트로더 적재 (2)



ODROID 부팅 과정 - 안드로이드 적재 예 (1)

1. U-Boot가 부트 디바이스의 램디스크 영역을 DRAM에 적재
2. U-Boot가 부트 디바이스의 커널 영역을 DRAM에 적재하고 실행
3. 커널은 램디스크 마운트하고 root 디렉토리의 init을 실행
4. Init은 init.rc을 해석하여 시스템 속성 등을 설정하고, 안드로이드에서 사용할 디스크 파티션을 마운트
5. Init.rc에 등록된 서비스를 실행
6. Zygote를 실행하여 Dalvik VM을 초기화하고 시스템 서버를 구동
7. 시스템 서버로부터 각 서비스를 등록

ODROID 부팅 과정 - 안드로이드 적재 예 (2)

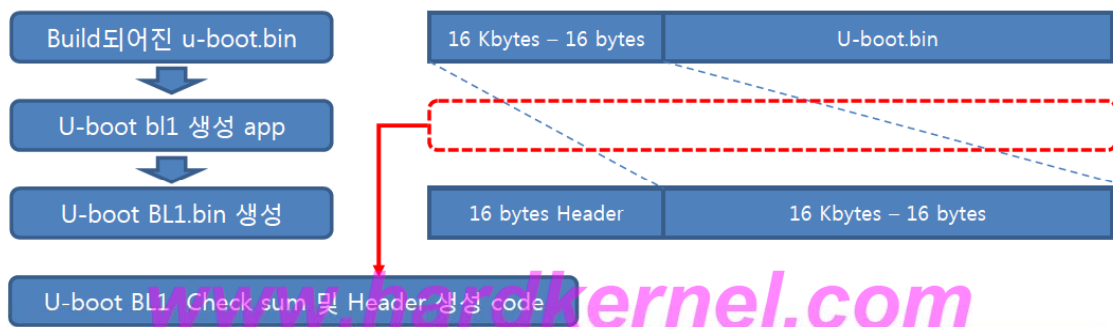


부트 디바이스 파티션 예 - ODROID A4

Partition Type	Start Sector	Partition Name	Partition Size
Normal Partition		Media (VFAT)	
		CACHE (EXT4)	128MB
		DATA (EXT4)	1GB
	131072	SYSTEM (EXT4)	512MB
Low Level Partition	50561	Reserved	
	17793	RAMDISK	16MB
	1409	KERNEL	8MB
	1507	U-BOOT Env	16KB
	33	U-BOOT	512KB
	1	U-BOOT BL1	16KB
	0	MBR	512B

1. Uboot build하여 생성되는 u-boot.bin 파일이 기록되는 위치.
2. mkb12 app을 사용하여 생성되는 u-boot bl1.bin 파일이 기록되는 위치.

U-Boot BL1 구조



Uboot/sd_fuse/c210-mkbl1.c

```

...
for(i = 0, checksum = 0; i < 16368; i++) {
    Buf1[i+16] = Buf[i];    checksum += Buf1[i+16];
}
*(unsigned long *) (Buf1+0) = 0x1f;    *(unsigned long *) (Buf1+4) = checksum;
Buf1[ 0] ^= 0x53;    Buf1[ 1] ^= 0x35;    Buf1[ 2] ^= 0x50;    Buf1[ 3] ^= 0x43;
Buf1[ 4] ^= 0x32;    Buf1[ 5] ^= 0x31;    Buf1[ 6] ^= 0x30;    Buf1[ 7] ^= 0x20;
Buf1[ 8] ^= 0x48;    Buf1[ 9] ^= 0x45;    Buf1[10] ^= 0x41;    Buf1[11] ^= 0x44;
Buf1[12] ^= 0x45;    Buf1[13] ^= 0x52;    Buf1[14] ^= 0x20;    Buf1[15] ^= 0x20;

for(i = 1; i < 16; i++)    Buf1[i] ^= Buf1[i-1];
...

```

U-Boot 부트로더 실행

U-Boot 소개

□ U-Boot (Universal Boot Loader)

- 오픈 소스로서 아래 사이트에서 소스코드를 다운로드
<http://www.denx.de/wiki/U-Boot/WebHome>
- 리눅스를 탑재하는 임베디드 시스템에서 가장 많이 사용되는 부트로더
- PPCBOOT와 ARMBOOT를 통합하여 개발
- 다양한 CPU 지원 (PPC, ARM, MIPS, SH, x86...)
- 다양한 시스템에서 사용 가능하고, 직렬포트, 네트워크, USB, 플래시를 지원하는 등의 많은 기능

U-Boot 실행

1. PuTTY 실행하여 터미널 연결 대기
2. 타겟 시스템 전원을 연결 후 임의의 키를 누름
 - 실패 시 RESET 버튼 누르고 임의의 키를 누름

Exynos4412 #

```

COM4 - PuTTY
U-Boot 2010.12-svn (Dec 21 2012 - 19:44:04) for Exynox4412

CPU: S5PC220 [Samsung SOC on SMP Platform Base on ARM CortexA9]
APLL = 1000MHz, MPLL = 880MHz
DRAM: 2047 MiB

PMIC VERSION : 0x00, CHIP REV : 2
TrustZone Enabled BSP
BL1 version: 20121128

Checking Boot Mode ... SDMMC
MMC Device 0: 7517 MB
MMC Device 1 not found
ModeKey Check... run normal_boot
Net: No ethernet found.
Hit any key to stop autoboot: 0
Exynos4412 #
Exynos4412 #
  
```

U-Boot 명령어

```

Exynos4412 # 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
bootp - boot image via network using BOOTP/TFTP protocol
bootvx - Boot vxWorks from an ELF image
chpart - change active partition
cmp - memory compare
coninfo - print console devices and information
cp - memory copy
crc32 - checksum calculation
dcache - enable or disable data cache
dnw - dnw - initialize USB device and ready to
for Windows server (specific)

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
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
mtdparts - define flash/nand partitions
mttest - simple RAM read/write test
mw - memory write (fill)
nfs - boot image via network using NFS protocol
nm - memory modify (constant address)
ping - send ICMP ECHO_REQUEST to network host
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
tftpboot - boot image via network using TFTP protocol
true - do nothing, successfully
usb - USB sub-system
version - print monitor version
wipeenv - Wipe uboot env.

```

명령어 - printenv

□ printenv 명령

- 설정된 환경변수를 표시

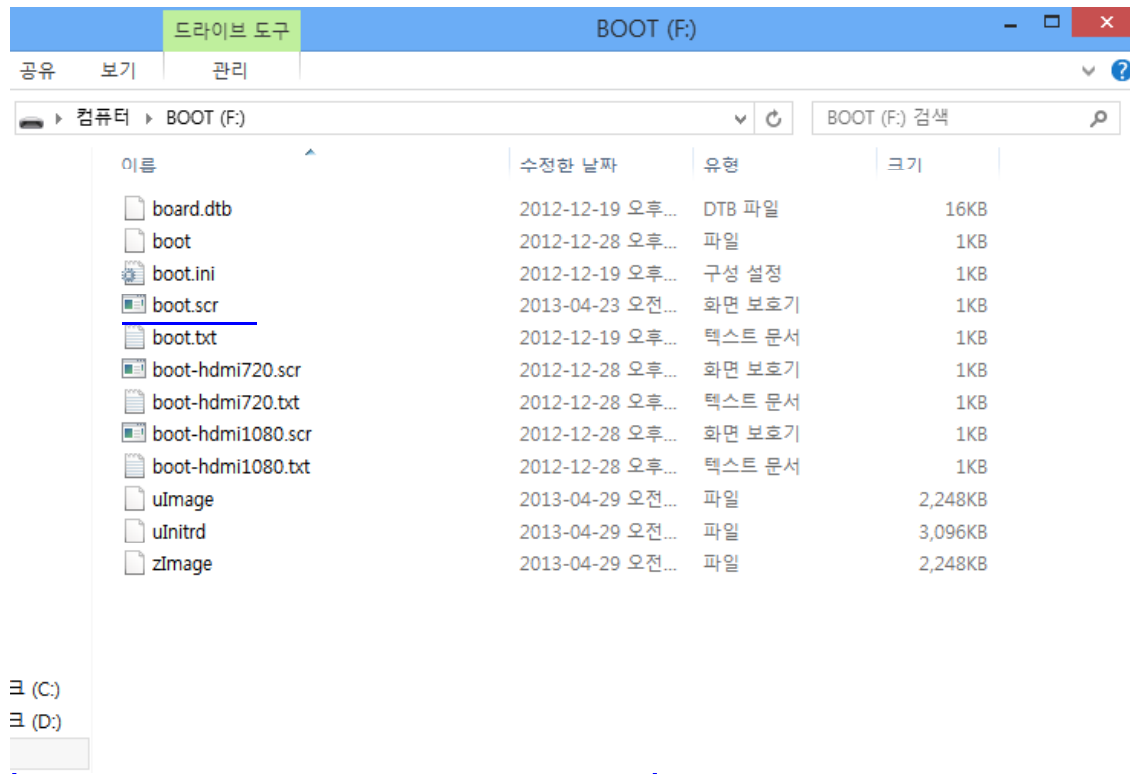
```

Exynos4412 # version
U-Boot 2010.12-svn (Dec 21 2012 - 19:44:04) for Exynox4412
Exynos4412 # printenv
baudrate=115200
bootargs=fb_x_res=1280 fb_y_res=720 hdmi_phy_res=720
bootcmd=mmc rescan 0:1; mmc rescan 0:2; if run loadbootscript_1; then run bootscrip; else
    if run loadbootscript_2; then run bootscrip; else run default_bootcmd;
fi; fi;
bootdelay=1
bootscrip=source 40008000
default_bootcmd=echo >>> Run Default Bootcmd <<<;movi read kernel 0 40008000;movi read rootfs 0 4
1000000 1000000;bootm 40008000 41000000
ethaddr=00:40:5c:26:0a:5b
gatewayip=192.168.0.1
ipaddr=192.168.0.20
loadbootscript_1=echo >>> Load Boot Script from mmc 0:1 <<<;fatload mmc 0:1 40008000 boot.scr
loadbootscript_2=echo >>> Load Boot Script from mmc 0:2 <<<;fatload mmc 0:2 40008000 boot.scr
netmask=255.255.255.0
serverip=192.168.0.10

Environment size: 781/16380 bytes
Exynos4412 #

```


SD 카드 파일 확인 - 윈도우 호스트



boot.scr 스크립트 파일 내용

```

F:\boot.scr - Notepad++
파일(F) 편집(E) 찾기(S) 보기(V) 인코딩(N) 언어(L) 설정(T) 매크로 실행 플러그인 창 ?
change.log prices.html boot.txt boot.scr
1 'ENOEMV??NULNULSOHNULNULNULNULNULNULNULNULNUL*0f?ENOSTXACKNULBoot
script for
odroid-u2NULNULNULNULNULNULNULNULNULSOHNULNULNULNULsetenv
initrd_high "0xffffffff"
2 setenv fdt_high "0xffffffff"
3 setenv fb_x_res "1920"
4 setenv fb_y_res "1080"
5 setenv hdmi_phy_res "1080"
6 setenv bootcmd "fatload mmc 0:1 0x40008000 zImage; fatload mmc 0:1
0x42000000 uInitrd; bootm 0x40008000 0x42000000"
7 setenv bootargs "console=tty1 console=ttySAC1,115200n8
fb_x_res=${fb_x_res} fb_y_res=${fb_y_res} hdmi_phy_res=${hdmi_phy_res}
root=UUID=e139ce78-9841-40fe-8823-96a304a09859 rootwait ro mem=2047M"
8 boot
9
length : 523 lines : 9 Ln : 9 Col : 1 Sel : 0 | 0 Dos#Windows ANSI INS

```

부트 파티션(Boot Partition) – FAT 명령(1)

□ ODROID의 부트 디바이스의 부트 파티션

- DOS 파일 시스템(FAT) 사용

□ U-Boot FAT 관련 명령

- `fatinfo <interface> <dev[:part]>`
파일 시스템 관련 정보 표시
- `fatls <interface> <dev[:part]>`
파일들의 리스트 표시
- `fatload <interface> <dev[:part]> <addr> <filename> [bytes]`
지정된 파일을 지정된 메모리에 적재

부트 파티션(Boot Partition) – FAT 명령(2)

```
Exynos4412 #
Exynos4412 # fatinfo mmc 0:1
-----Partition 1-----
Partition1: Start Address(0x1000), Size(0x40000)
-----
-----Partition 2-----
Partition1: Start Address(0x1000), Size(0x40000)
-----
-----Partition 3-----
Partition1: Start Address(0x1000), Size(0x40000)
-----
-----Partition 4-----
Partition1: Start Address(0x1000), Size(0x40000)
-----
Interface: SD/MMC
Device 0: Vendor: Man 744a45 Snr 03a04d27 Rev: 0.2 Prod: SDC
Type: Removable Hard Disk
Capacity: 14.6 MB = 0.0 GB (30068 x 512)
Partition 1: Filesystem: FAT32 "BOOT"
Exynos4412 #
```

부트 파티션(Boot Partition) – FAT 명령(3)

```
Exynos4412 #
Exynos4412 # fatls mmc 0:1
Partition1: Start Address(0x1000), Size(0x40000)
    306 boot.txt
    306 boot.txt
    523 boot-hdmi1080.scr
    523 boot
    2301272 zimage
    3169476 uinitrd
    418 boot.ini
    451 boot-hdmi1080.txt
    521 boot-hdmi720.scr
    16123 board.dtb
    523 boot.scr
    449 boot-hdmi720.txt
    2301336 uimage
13 file(s), 0 dir(s)
Exynos4412 #
```

명령어 – bootm

□ bootm 명령

- 지정된 메모리 주소의 이미지를 사용하여 부팅
- 리눅스 커널인 경우 인수 전달

```
Exynos4412 # help bootm
bootm - boot application image from memory

Usage:
bootm [addr [arg ...]]
    - boot application image stored in memory
      passing arguments 'arg ...'; when booting a Linux kernel,
      'arg' can be the address of an initrd image

Sub-commands to do part of the bootm sequence. The sub-commands must be
issued in the order below (it's ok to not issue all sub-commands):
    start [addr [arg ...]]
    loados - load OS image
    cmdline - OS specific command line processing/setup
    bdt - OS specific bd_t processing
    prep - OS specific prep before relocation or go
    go - start OS
Exynos4412 #
```

ODROID 부팅 설정

□ 부팅 설정

```
setenv bootcmd "fatload mmc 0:1 0x40008000 zImage;
fatload mmc 0:1 0x42000000 uinitrd; bootm 0x40008000
0x42000000"
```

- 리눅스 압축커널 이미지인 zImage를 메모리 0x40008000 번지에 적재
- 초기 램디스크 uinitrd를 메모리 0x42000000에 적재
- 0x40008000 번지 커널을 실행하여 부팅하고 인수로 램디스크 적재 주소 전달

커널 적재 부팅 예

□ U-Boot에서 커널 및 램디스크를 적재하여 오드로이드 부팅

- 부팅 시 전달할 값인 **bootargs** 설정
- 커널은 메모리 0x50001000 번지에 적재
- 램 디스크는 메모리 0x53000000 번지에 적재
 - md(memory display)명령 사용하여 메모리 확인

```
Exynos4412 # setenv bootargs "console=tty1
console=ttySAC1,115200n8 root=UUID=e139ce78-9841-40fe-
8823-96a304a09859 rootwait ro mem=2047M"
```

```
Exynos4412 # fatload mmc 0:1 0x50001000 zImage
```

```
Exynos4412 # fatload mmc 0:1 0x53000000 uinitrd
```

```
Exynos4412 # md 0x53000000
```

```
Exynos4412 # bootm 0x50001000 0x53000000
```

```
COM4 - PuTTY
Exynos4412 # setenv bootargs "console=tty1 console=ttySAC1,115200n8 root=UUID=e139ce78-9841-40fe-8823-96a304a09859 rootwait ro mem=2047M"
Exynos4412 #
Exynos4412 # fatload mmc 0:1 0x50001000 zImage
Partition1: Start Address(0x1000), Size(0x40000)
reading zImage

2301272 bytes read
Exynos4412 #
Exynos4412 # fatload mmc 0:1 0x53000000 uinitrd
Partition1: Start Address(0x1000), Size(0x40000)
reading uinitrd

3169476 bytes read
Exynos4412 #
Exynos4412 # md 0x53000000
53000000: 56190527 f297b73a d8da7d51 845c3000   '..V:...Q}...0#.
53000010: 00000000 00000000 9023381d 01030205   .....8#.....
53000020: 6e756255 72207574 69646d61 33206b73   Ubuntu ramdisk 3
53000030: 372e302e 646f2d35 64696f72 00003278   .0.75-odroidx2..
53000040: 00088b1f 517dd8a3 5bb40300 b8db5bed   .....Q...[...
53000050: fc57efb2 bb26a857 84271740 36965bb6   ..W.W.&.@.'...[.6
53000060: 2cf5e5dd 070b6dcf 3cf7bbda 638f27a5   ....m.....<..c
53000070: 1clf892b 900bf5db efdff9a5 1cb248bc   +.....H..
53000080: 3ddca12 258a1fb7 d1a3468d 919bf468   ...=...%.F..h...
53000090: eb6af5b2 0b7756d5 0fb775fe 2fbfc67e   ..j..Vw..u..~/
530000a0: 5b8747ba f2fdff8b f0eafbb2 496f6770   .G.[.....pgol
530000b0: fd497773 68feb6d6 c7aad93f e0eee83b   swl....h?...;...
530000c0: e3e59ddf 5f765f75 cf07bb74 c5fe98f7   ....u_v_t.....
530000d0: 678e47cb 2a24a2e8 7d593c9f 494419da   .G.g..$*.<Y}..DI
530000e0: ce3b18a7 a285e167 f16bca13 26fb7c45   ...;g.....k.El.&
530000f0: 94615726 b6226f0b 497cbd5e 4f228af5   &Wa..o".^.|l.."0
Exynos4412 #
```

임베디드 소프트웨어

```
Exynos4412 #
Exynos4412 # bootm 0x50001000 0x53000000
Boot with zImage
## Loading init Ramdisk from Legacy Image at 53000000 ...
   Image Name:   Ubuntu ramdisk 3.0.75-odroidx2
   Image Type:   ARM Linux RAMDisk Image (gzip compressed)
   Data Size:    3169412 Bytes = 3095 KiB
   Load Address: 00000000
   Entry Point:  00000000

Starting kernel ...

Uncompressing Linux...

Begin: Running /scripts/init-bottom ... done.
[ 4.660579] init: ureadahead main process (855) terminated with status 5
[ 5.047995] init: mounted-proc main process (864) terminated with status 1

speech-dispatcher disabled; edit /etc/default/speech-dispatcher
saned disabled; edit /etc/default/saned
Last login: Sat Jan 1 07:40:11 UTC 2000 on ttySAC1
Welcome to Ubuntu 13.04 (GNU/Linux 3.0.75-odroidx2 armv7l)

 * Documentation:  https://help.ubuntu.com/

root@odroid:~#
```

- 앞에서 소개되지 않은 u-boot 명령어 3개의 사용 예를 보여라