

ARM926EJ-S AM1808

Mango1808 Linux Manual

<http://www.mangoboard.com/>

<http://cafe.naver.com/embeddedcrazyboys>

Crazy Embedded Laboratory

Document History

Revision	Date	Change note

1.	필요한 Tool.....	4
1.1.	Flash utility 준비하기	4
1.2.	환경설정 및 도구.....	4
1.3.	U-boot 컴파일 및 Write 하기	4
1.4.	커널 컴파일 및 Write 하기.....	11
1.4.1.	커널 컴파일 하기.....	12
1.4.2.	Kernel Write 하기	12
1.5.	Buildroot 컴파일 하기	12
1.6.	Rootfs 만들기.....	13
1.7.	UBI FS 만들기	15
1.8.	WiFi/BT Test.....	15
1.6.1.	WiFi 테스트	15
1.6.1.	BT(Bluetooth) 테스트	17
1.9.	이더넷 테스트.....	18
1.10.	SATA 테스트.....	18

1. 필요한 Tool

소스는

<http://crztech.iptime.org:8080/Release/mango1808/>

기초 매뉴얼은

https://docs.google.com/open?id=0Bxdf_3fzbUgSYB2T0pyT3JXTIE

참조 링크는 아래와 같습니다.

http://processors.wiki.ti.com/index.php/Download_CCS

위의 링크에서 TI CCS v5 버전을 다운로드 합니다.

<http://sourceforge.net/projects/dvflashutils/files/OMAP-L138/v2.40/>

다운로드 하면 됩니다.

1.1. Flash utility 준비하기

다운로드

http://crztech.iptime.org:8080/Release/mango1808/utility/OMAP-L138_FlashAndBootUtils_2_40-20120810.tar.bz2

리눅스 Host PC에서

```
$ tar xjf OMAP-L138_FlashAndBootUtils_2_40-20120810.tar.bz2
```

```
$ cd OMAP-L138_FlashAndBootUtils_2_40-20120810/
```

1.2. 환경설정 및 도구

PC : Ubuntu 12.04 64bit

Toolchain 및 설치 Package

<http://cafe.naver.com/embeddedcrazyboys/19626>

1.3. U-boot 컴파일 및 Write 하기

컴파일 툴 체인 설치

```
$ cd /opt/  
$ sudo wget http://crztech.iptime.org:8080/Release/Toolchain/arm-2009q1-203-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar
```

```
$ sudo tar xf arm-2009q1-203-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar  
$ls  
arm-2009q1
```

디렉토리 확인 합니다.

```
$ tar xf mango1808-uboot-20121128.tgz
```

U-boot 컴파일 하기

```
$ ./build_uboot  
$ cp u-boot.bin ~/work/Mango1808/mango1808-20121112/OMAP-L138_FlashAndBootUtils_2_40-20120810/OMAP-L138/GNU/
```

Linux PC에서

```
# minicom
```

다른 터미널 창을 띄우고,

전원, uart를 연결하면, Minicom 창에 "BOOTME" 라고 메시지가 나타납니다.

```
Welcome to minicom 2.5
```

```
OPTIONS: I18n
```

```
Compiled on May 2 2011, 10:05:24.
```

```
Port /dev/ttyUSB0
```

Press CTRL-A Z for help on special keys

```
◆BOOTME ◆◆◆◆◆BOOTME
```

"BOOTME" 글자가 나오면, 이상없이 동작하는 것입니다.

이제 minicom 창을 닫습니다.

```
# cd OMAP-L138_FlashAndBootUtils_2_40-20120810/OMAP-L138/GNU
```

```
# sudo apt-get install mono-runtime
```

```
# sudo apt-get install mono-devel
```

```
[root@HP-note-jji GNU]# sudo ./sfh_OMAP-L138.exe -p /dev/ttyUSB0 -erase -flashType NAND
```

The assembly mscorlib.dll was not found or could not be loaded.

It should have been installed in the `usr/lib/mono/2.0/mscorlib.dll' directory.

Under Linux

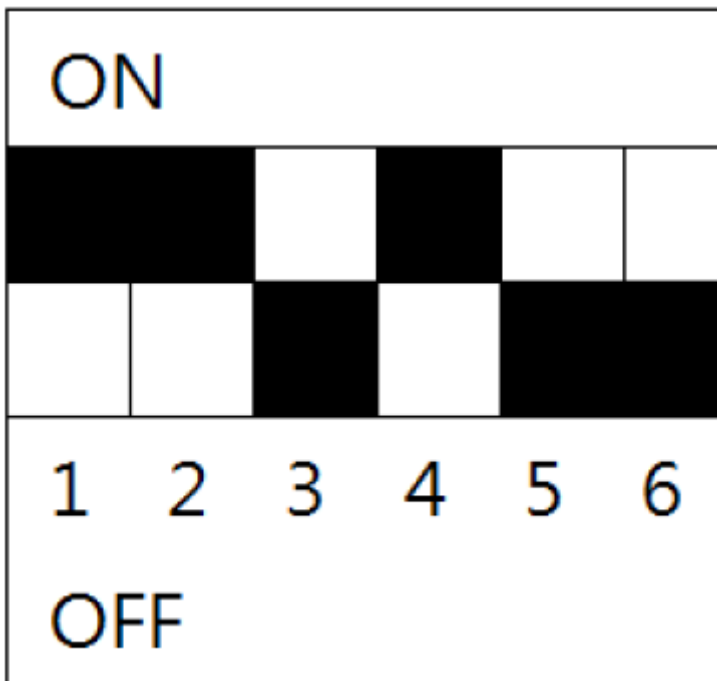
The Mono Framework must be installed and in the path. RPMs are available at <http://www.mono-project.com/Downloads>. The below instructions assume that the GNU ARM cross-compiler tools (arm-arago-linux-gnueabi-gcc, etc.) are in the current PATH.

The C6x Compiler Tools are also required to build some components. These are available free of charge at https://www.ti.com/downloads/sds_support/TICodeGenerationTools/download.htm. You will need to add the bin directory to your PATH and to set the environment variables as directed by the installer. Edit the build.mak file under Common directory and add the paths to ARM and C6x compilers.

Then go to the GNU directory of the package and run:

아래 명령을 실행하기전에 꼭 uart2에 linux host PC에 연결 후 minicom 또는 터미널 프로그램을 실행하면 안됩니다.

부팅 모드는 uart2 부팅 모드로 합니다.



```
[root@HP-note-jji GNU]# sudo ./sfh_OMAP-L138.exe -p /dev/ttyUSB0 -erase -flashType NAND
```

실행 후

(AIS Parse): Waiting for BOOTME... (power on or reset target now)

메시지가 나오면, 리셋 버튼을 누릅니다.

TI Serial Flasher Host Program for OMAP-L138
(C) 2012, Texas Instruments, Inc.
Ver. 1.67

[TYPE] Global erase
[TARGET] MANGO1808
[DEVICE] NAND
[NAND Block] 1

Attempting to connect to device /dev/ttyUSB0...
Press any key to end this program at any time.

(AIS Parse): Read magic word 0x41504954.
(AIS Parse): Waiting for BOOTME... (power on or reset target now)
(AIS Parse): BOOTME received!
(AIS Parse): Performing Start-Word Sync...
(AIS Parse): Performing Ping Opcode Sync...
(AIS Parse): Processing command 0: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 14376-Byte section to address 0x80000000.
(AIS Parse): Processing command 1: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 1320-Byte section to address 0x80003828.
(AIS Parse): Processing command 2: 0x58535906.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Performing jump and close...
(AIS Parse): AIS complete. Jump to address 0x80000000.
(AIS Parse): Waiting for DONE...
(AIS Parse): Boot completed successfully.

Waiting for SFT on the OMAP-L138...

Erasing flash

Operation completed successfully.

이제 NAND 모드로 부팅을 합니다.

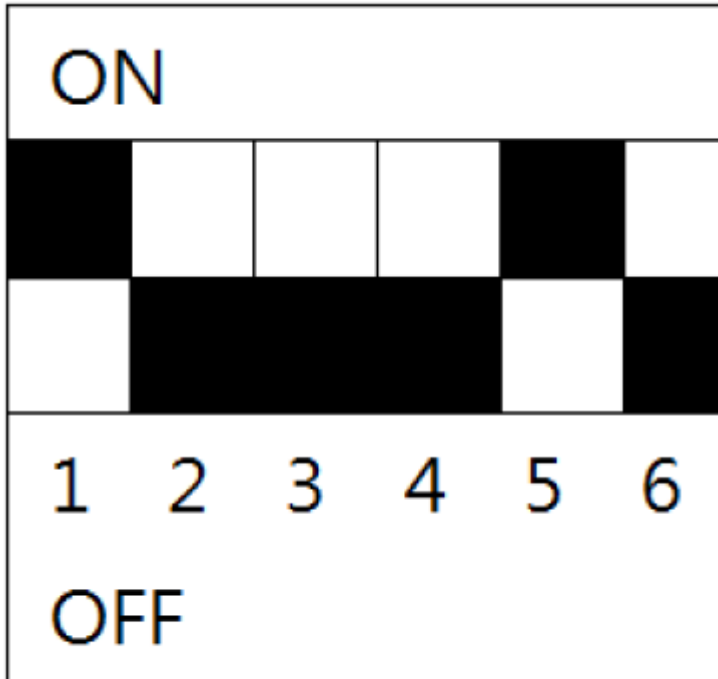


그림 1.2: NAND Flash boot mode.

부팅시 메시지가 아래와 같이 나옵니다.

```
MANGO1808 initialization passed!  
Booting TI User Boot Loader  
  UBL Version: 1.65  
  UBL Flashtype: NAND  
Starting NAND Copy..  
Valid magicnum, 0x55424CBB, found in block 0x00000006.  
  DONE  
?mping to entry point at 0xC1080000.  
  
U-Boot 2010.12, MANGO1808, r1_00, 2012-08-07  
NAND: 256 MiB  
MMC: davinci: 0
```

Bad block table found at page 131008, version 0x01
Bad block table found at page 130944, version 0x01
nand_read_bbt: Bad block at 0x0000015a0000
nand_read_bbt: Bad block at 0x000005860000
nand_read_bbt: Bad block at 0x0000059c0000
nand_read_bbt: Bad block at 0x000005a80000
nand_read_bbt: Bad block at 0x0000062e0000
nand_read_bbt: Bad block at 0x000006500000
nand_read_bbt: Bad block at 0x000006740000
nand_read_bbt: Bad block at 0x000007420000
nand_read_bbt: Bad block at 0x0000075e0000
nand_read_bbt: Bad block at 0x000007b00000
nand_read_bbt: Bad block at 0x000007b20000
nand_read_bbt: Bad block at 0x000007cc0000
nand_read_bbt: Bad block at 0x000008040000
nand_read_bbt: Bad block at 0x0000086a0000
nand_read_bbt: Bad block at 0x000008880000
nand_read_bbt: Bad block at 0x00000c220000
nand_read_bbt: Bad block at 0x00000d5e0000
nand_read_bbt: Bad block at 0x00000e460000
In: serial
Out: serial
Err: serial
ARM Clock : 456000000 Hz
DDR Clock : 150000000 Hz
EMIFA Clock : 24000000 Hz
DSP Clock : 456000000 Hz
ASYNC3 Clock : 228000000 Hz
MMCSDB Clock : 228000000 Hz
Net: Ethernet PHY: RTL8201F @ 0x00
DaVinci-EMAC
Hit any key to stop autoboot: 0

1.4. 커널 컴파일 및 Write 하기

1.4.1. 커널 컴파일 하기

```
$ cd kernel/
```

```
$ ./build_kernel clean
```

```
$ ./build_kernel defconfig
```

```
$ ./build_kernel
```

하면 됩니다.

1.4.2. Kernel Write 하기

u-boot 실행 후

ipaddress 설정

```
MANGO1808 > setenv ipaddr 192.168.3.20; setenv gatewayip 192.168.3.1; setenv serverip  
192.168.3.9
```

```
MANGO1808 > saveenv
```

```
Saving Environment to NAND...
```

```
Erasing Nand...
```

```
Erasing at 0x0 -- 100% complete.
```

```
Writing to Nand... done
```

```
MANGO1808 > tftp c0700000 uImage
```

받은 커널 이미지 사이즈를 보고 Write 사이즈를 결정합니다.

```
MANGO1808 > nand erase 0x200000 0x400000;nand write 0xc0700000 0x200000 0x400000
```

부팅 후

```
MANGO1808 > setenv bootcmd "nand read 0xc0700000 0x200000 0x400000;bootm 0xc0700000"
```

```
MANGO1808 > saveenv
```

1.5. Buildroot 컴파일 하기

```
$ cd buildroot-2012.05-mango1808/
```

```
$ ./build_rootFS_clean.sh
```

```
$ ./build_rootFS.sh
```

```
$ cd buildroot-2012.05-mango1808/output/images
```

```
$ cp rootfs.tar ../../image/
```

1.6. Rootfs 만들기

"mmc 8GB " Host PC에 삽입한다.

```
#sudo dmesg | tail
[10993.575042] sd 7:0:0:0: [sdb] Assuming drive cache: write through
[10993.576432] sdb: sdb1 sdb2 sdb3 sdb4
```

```
[root@HP-note-jji image]# sudo fdisk /dev/sdb
```

Command (m for help): p

```
Disk /dev/sdb: 8010 MB, 8010072064 bytes
214 heads, 8 sectors/track, 9138 cylinders, total 15644672 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		8192	15644671	7818240	b	W95 FAT32

Command (m for help): d

Selected partition 1

Command (m for help): n

Partition type:

- p primary (0 primary, 0 extended, 4 free)
- e extended

Select (default p): p

Partition number (1-4, default 1):

Using default value 1

First sector (2048-15644671, default 2048):

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-15644671, default 15644671):

Using default value 15644671

Command (m for help): p

Disk /dev/sdb: 8010 MB, 8010072064 bytes
214 heads, 8 sectors/track, 9138 cylinders, total 15644672 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		2048	15644671	7821312	83	Linux

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

```
[root@HP-note-jji image]# mkfs
```

```
mkfs      mkfs.ext2    mkfs.ext4dev  mkfs.ntfs
mkfs.bfs  mkfs.ext3    mkfs.minix    mkfs.vfat
mkfs.cramfs  mkfs.ext4    mkfs.msdos
```

```
[root@HP-note-jji image]# mkfs.ext4 -help
```

```
mkfs.ext4: invalid option -- 'h'
```

```
Usage: mkfs.ext4 [-c|-l filename] [-b block-size] [-C cluster-size]
        [-i bytes-per-inode] [-I inode-size] [-J journal-options]
        [-G meta group size] [-N number-of-inodes]
        [-m reserved-blocks-percentage] [-o creator-os]
        [-g blocks-per-group] [-L volume-label] [-M last-mounted-directory]
        [-O feature[,...]] [-r fs-revision] [-E extended-option[,...]]
        [-T fs-type] [-U UUID] [-jnqvFKSV] device [blocks-count]
```

```
# mkfs.ext4 -L mango1808 /dev/sdb1
```

```
# sudo mkdir fs
```

```
# sudo mount /dev/sdb1 fs
```

```
# cd fs/
```

```
# sudo tar xf ../rootfs.tar
```

```
# cd ..
```

```
# sudo sync
```

```
# sudo umount /dev/sdb1
```

Mango1808 Board에 MMC 카드를 삽입 후 부팅을 합니다.

```
setenv bootargs "root=/dev/mmcblk0p1 rootfstype=ext4 console=ttyS2,115200 rootwait"
```

1.7. UBI FS 만들기

부팅 후

```
# ubiformat /dev/mtd4 -s 512 -O 2048
```

```
# ubiattach /dev/ubi_ctrl -m 4 -O 2048
```

```
# ubimkvol /dev/ubi0 -N rootfs -m
```

```
# mkdir /mnt/nand
```

```
# mount -t ubifs ubi0:rootfs /mnt/nand
```

```
# tar xvf rootfs.tar -C /mnt/nand
```

```
## Nand Flash UBIFS 부팅 u-boot 설정
```

```
# setenv bootargs "noinitrd console=ttyS2,115200 rw ubi.mtd=4,2048 root=ubi0:rootfs  
rootfstype=ubifs rootwait"
```

```
# saveenv
```

```
# boot
```

1.8. WiFi/BT Test

1.6.1. WiFi 테스트

WiFi 모듈을 삽입 후 커널 로그에서

```
mmc1: new SDIO card at address 0001
```

위의 메시지가 나오는 것을 확인 합니다.

```
# insmod mlan.ko
```

```
mlan: module license 'Marvell Proprietary' taints kernel.
```

```
Disabling lock debugging due to kernel taint
```

```
# insmod sd8787.ko
```

WLAN FW is active

```
# iwconfig
```

```
lo          no wirnet eth0: DaVinci EMAC: ioctl not supported
           eless extensions.
```

```
eth0       no wireless extensions.
```

```
m lan0     IEEE 802.11-DS  ESSID:""  Nickname:""
           Mode:Managed  Access Point: Not-Associated  Bit Rate:1 Mb/s
           Tx-Power=8 dBm
           Retry limit:9  RTS thr=2347 B  Fragment thr=2346 B
           Encryption key:off
           Power Management:on
           Link Quality=0/5  Signal level=0 dBm  Noise level=0 dBm
           Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:2
           Tx excessive retries:0  Invalid misc:0  Missed beacon:0
```

```
uap0      IEEE 802.11-DS  ESSID:""
           Mode:Master  Frequency:2.437 GHz  Access Point: Not-Associated
           Encryption key:off
           Link Quality:0  Signal level:0  Noise level:0
           Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
           Tx excessive retries:0  Invalid misc:0  Missed beacon:0
```

```
# ifconfig mlan0 up
```

```
ADDRCONF(NETDEV_UP): mlan0: link is not ready
```

```
# iwlist mlan0 scanning
```

스캐닝 후 AP "iwconfig" 명령으로 ap 이름을 넣으면 됩니다.

```
# iwconfig mlan0 essid CRZ_icanjji
```

```
# iwconfig
```

```
lo          no wirnet eth0: DaVinci EMAC: ioctl not supported
           eless extensions.
```

```
eth0       no wireless extensions.
```

```
m lan0     IEEE 802.11-DS  ESSID:"CRZ_icanjji" [14]  Nickname:""
```


Mode:Managed Frequency:2.437 GHz Access Point: 00:26:5A:23:0D:E8
Bit Rate:39 Mb/s Tx-Power=8 dBm
Retry limit:9 RTS thr=2347 B Fragment thr=2346 B
Encryption key:off
Power Management:on
Link Quality=0/5 Signal level=-93 dBm Noise level=-101 dBm
Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:43
Tx excessive retries:11 Invalid misc:50 Missed beacon:0

```
# udhcpc -i mlan0
udhcpc (v1.20.1) started
Sending discover...
Sending select for 192.168.3.12...
Lease of 192.168.3.12 obtained, lease time 86400
deleting routers
route: SIOCDELRT: No such process
adding dns 192.168.3.1
```

ip를 할당 받아 옵니다.

Iperf 테스트

<http://cafe.naver.com/embeddedcrazyboys/20119>

1.6.1. BT(Bluetooth) 테스트

```
# insmod mbtchar.ko
# insmod bt8787.ko
# hciconfig
hci0: Type: BR/EDR Bus: SDIO
      BD Address: 00:00:00:00:00:00 ACL MTU: 0:0 SCO MTU: 0:0
      DOWN
      RX bytes:47 acl:0 sco:0 events:0 errors:0
      TX bytes:11 acl:0 sco:0 commands:0 errors:0
```

```
# hciconfig hci0 up
# hcitool -i hci0 scan
```

디버깅

```
# hciconfig hci0 up
Can't init device hci0: Operation not possible due to RF-kill (132)
위와 같이 에러가 발생 합니다.
커널에서 아래 configuration을 하지 않도록 수정
< > RF switch subsystem support --->
```

1.9. 이더넷 테스트

```
# ifconfig eth0 up
# udhcpc -i eth0
```

Iperf 테스트

<http://cafe.naver.com/embeddedcrazyboys/20119>

1.10. SATA 테스트

SATA 테스트

커널 로그에서 아래와 같이 나오면 됩니다.

```
ata1: SATA link up 3.0 Gbps (SStatus 123 SControl 300)
ata1.00: ATA-8: INDILINX BAREFOOT-SATA, 2146, max UDMA/133
ata1.00: 62533296 sectors, multi 1: LBA48 NCQ (depth 31/32)
ata1.00: configured for UDMA/133
scsi 0:0:0:0: Direct-Access    ATA        INDILINX BAREFOO 2146 PQ: 0 ANSI: 5
sd 0:0:0:0: [sda] 62533296 512-byte logical blocks: (32.0 GB/29.8 GiB)
sd 0:0:0:0: Attached scsi generic sg0 type 0
sd 0:0:0:0: [sda] Write Protect is off
sd 0:0:0:0: [sda] Write cache: enabled, read cache: enabled, doesn't support DPA
sda: sda1
sd 0:0:0:0: [sda] Attached SCSI disk
```

```
# mkdir /mnt/sata
# mount /dev/sda1 /mnt/sata/
EXT2-fs (sda1): error: couldn't mount because of unsupported optional features )
EXT4-fs (sda1): recovery complete
EXT4-fs (sda1): mounted filesystem with ordered data mode. Opts: (null)
# df
Filesystem          Size      Used Available Use% Mounted on
/dev/root           7.3G    215.4M      6.8G   3% /
devtmpfs            60.5M         0    60.5M   0% /dev
tmpfs               60.6M         0    60.6M   0% /dev/shm
tmpfs               60.6M    52.0K    60.6M   0% /tmp
/dev/sda1          29.3G    202.3M    27.7G   1% /mnt/sata
```