

Bootable SD-card 생성, 생산용도 가능

(주의 : Disk 선택 시 ubuntu server가 설치된 저장공간을 선택하면 ubuntu server가 모두 지워 질 수 있으므로 sd-card Disk로 올바르게 택해야 합니다.)

- Ti sdk에서 제공하는 "create-sdcard.sh"를 사용하며, 그대로 사용하면 70MByte size partition이므로 RFS이미지가 들어갈 수 있는 크기로 조정하여 생성합니다.
- "/opt/ti-processor-sdk-linux-rt-am57xx-evm-05.03.00.07/bin"폴더로 이동하여 "create-sdcard.sh"파일에서 아래와 같이 563, 565 line의 빨강색 숫자로 변경합니다.

```
562: parted -s $DRIVE mklabel msdos
563: parted -s $DRIVE unit cyl mkpart primary fat32 -- 0 399
564: parted -s $DRIVE set 1 boot on
565: parted -s $DRIVE unit cyl mkpart primary ext2 -- 399 -2
```

수정된 파일: [create-sdcard.sh](#)

- sd-card를 ubuntu server에 연결하고 "sudo ./create-sdcard.sh" 로 실행하면 아래로그를 보이며 빨강색 글씨를 입력하면 됩니다.

```
sudo ./create-sdcard.sh
[sudo] password for hsoh: 본인 password
```

```
#####
```

This script will create a bootable SD card from custom or pre-built binaries.

The script must be run with root permissions and from the bin directory of the SDK

Example:

```
$ sudo ./create-sdcard.sh
```

Formatting can be skipped if the SD card is already formatted and partitioned properly.

```
#####
```

Available Drives to write images to:

```
# major minor size name
1: 8 0 1953514584 sda
2: 8 32 62586880 sdc
```

Enter Device Number or n to exit: 2

(주의: 1번은 ubuntu server가 설치된 저장공간이라 1번을 선택하면 ubuntu server가 지워므로 절대로 선택하면 안됩니다.)

2번이 보여지지 않으면 sd-card를 인식되지 않은 상태이므로 인식될 수 있도록 추가로 확인이 필요합니다.)
sdc was selected

```
#####
```

*****WARNING*****

Selected Device is greater then 16GB
Continuing past this point will erase data from device
Double check that this is the correct SD Card

```
#####
```

Would you like to continue [y/n] : y

/dev/sdc is an sdx device
Current size of sdc1 62584832 bytes
SD Card is not correctly partitioned

#####

Select 2 partitions if only need boot and rootfs (most users).
Select 3 partitions if need SDK & other content on SD card. This is
usually used by device manufacturers with access to partition tarballs.

****WARNING**** continuing will erase all data on sdc

#####

Number of partitions needed [2/3] : **2**

Now partitioning sdc with 2 partitions...

#####

Now making 2 partitions

#####

1024+0 records in
1024+0 records out
1048576 bytes (1.0 MB, 1.0 MiB) copied, 0.18748 s, 5.6 MB/s
DISK SIZE - 64088965120 bytes

#####

Partitioning Boot

#####

mkfs.fat 4.1 (2017-01-24)
mkfs.fat: warning - lowercase labels might not work properly with DOS or Windows

#####

Partitioning rootfs

#####

mke2fs 1.44.1 (24-Mar-2018)
/dev/sdc2 contains a ext3 file system labelled 'rootfs'
last mounted on Tue Jan 31 09:33:03 2023
Proceed anyway? (y,N) **y**
Creating filesystem with 15624448 4k blocks and 3907584 inodes
Filesystem UUID: 80cead9e-de5d-4410-9def-7f4d4e6ae924
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000, 7962624, 11239424

Allocating group tables: 0/477 done
Writing inode tables: 0/477 done
Creating journal (65536 blocks): done
Writing superblocks and filesystem accounting information: 0/477 done

#####

Partitioning is now done
Continue to install filesystem or select 'n' to safe exit

****Warning**** Continuing will erase files any files in the partitions

#####

Would you like to continue? [y/n] : **y**

Mount the partitions

Emptying partitions

Syncing....

#####

Choose file path to install from

- 1) Install pre-built images from SDK
- 2) Enter in custom boot and rootfs file paths

#####

Choose now [1/2] : **1**

Will now install from SDK pre-built images
now installing: ti-processor-sdk-linux-rt-am57xx-evm-05.03.00.07

#####

Multiple rootfs Tarballs found

#####

- 1:tisdk-docker-rootfs-image-am57xx-evm.tar.xz
- 2:tisdk-rootfs-image-am57xx-evm.tar.xz

Enter Number of rootfs Tarball: **1**

#####

Copying files now... will take minutes

#####

Copying boot partition

MLO copied

u-boot.img copied

uEnv.txt copied

Copying rootfs System partition

Written 1% Written 2% Written 3% Written 4% Written 5% Written 6% Written 7% Written 8%
Written 9% Written 10% Written 11% Written 12% Written 13% Written 14% Written 15% Written
16% Written 17% Written 18% Written 19% Written 20% Written 21% Written 22% Written 23%
Written 24% Written 25% Written 26% Written 27% Written 28% Written 29% Written 30%
Written 31% Written 32% Written 33% Written 34% Written 35% Written 36% Written 37%
Written 38% Written 39% Written 40% Written 41% Written 42% Written 43% Written 44%
Written 45% Written 46% Written 47% Written 48% Written 49% Written 50% Written 51%
Written 52% Written 53% Written 54% Written 55% Written 56% Written 57% Written 58%
Written 59% Written 60% Written 61% Written 62% Written 63% Written 64% Written 65%
Written 66% Written 67% Written 68% Written 69% Written 70% Written 71% Written 72%

Written 73% Written 74% Written 75% Written 76% Written 77% Written 78% Written 79%
Written 80% Written 81% Written 82% Written 83% Written 84% Written 85% Written 86%
Written 87% Written 88% Written 89% Written 90% Written 91% Written 92% Written 93%
Written 94% Written 95% Written 96% Written 97% Written 98% Written 99% Written 100%

Syncing...

Un-mount the partitions

Remove created temp directories

Operation Finished

- 생성이 완료되면 SD-card의 "boot" 파티션에 MLO, README, u-boot.img, uEnv.txt파일 생성되며 모두 삭제하고 아래 파일들을 모두 SD-card로 복사 합니다.

- //bsp/target-maker/output_img폴더의 모든 파일들
- //bsp/target-maker/boot/MLO 파일
- //bsp/target-maker/boot/u-boot.img 파일

- "UpdateFlag"파일로 원하는 update type으로 변경합니다.

- SD-card를 세트에 연결 하고 power on하면 아래와 같이 boot되면서 uboot 모드로 진입하여 image write가 가능합니다. HW booting 순서 setting은 1번 sd-card, 2번 QSPI flash입니다.

U-Boot SPL 2018.01-g5161b46a (Jan 30 2023 - 10:15:03)

DRA722-GP ES2.0

omap_revision = 0x7220200

sys_clk = 20000 KHz

m = 333, n+1 = 5

Synthesis ddr_clk = 2664000 KHz

(2 * core_dpll_params->m2) = 4

output ddr_clk = 666000 KHz

DDR clock = 666000000 Hz

Trying to boot from MMC1

no pinctrl state for default mode

no pinctrl state for default mode

*** Warning - bad CRC, using default environment

reading u-boot.img

reading u-boot.img

reading u-boot.img

reading u-boot.img

(위 log는 SD-card로 부팅되었을 때만 출력되는 log입니다.)

U-Boot 2018.01-g5161b46a (Jan 30 2023 - 10:15:03 +0900)

CPU : DRA722-GP ES2.0

omap_revision = 0x7220200

DDR clock = 666000000 Hz

Model: TI AM5718 IDK, SUR5000-version: 0.18

Board: AM571x IDK REV

DRAM: omap_hw_init_context()= 2

EMIF_SDRAM_CONFIG1 = 0x61863332, EMIF_SDRAM_CONFIG2 = 0x0

1 GiB

MMC: OMAP SD/MMC: 0, OMAP SD/MMC: 1
*** Warning - bad CRC, using default environment

Net:

Warning: ethernet@48484000 using MAC address from ROM
eth0: ethernet@48484000

** File not found UpdateFlag **

[fw_update] Flag not found! skip..

gpio: pin 167 (gpio 167) value is 1

gpio: pin 179 (gpio 179) value is 0

gpio: pin 179 (gpio 179) value is 1

Hit any key to stop autoboot: 0

HUR-BOOT>