

망고 210 ICS nand booting 메뉴얼 작성 및 patch

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

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

Crazy Embedded Laboratory

Document History

Revision	Date	Change note

1. android nand build.....	4
2. kernel build.....	6
3. nand 부팅 방법	6
4. 소스 변경 사항.....	7
5. kernel config 파일 확인사항	22

1. android nand build

android/device/crazyboys/mango210\$ vi BoardConfig.mk

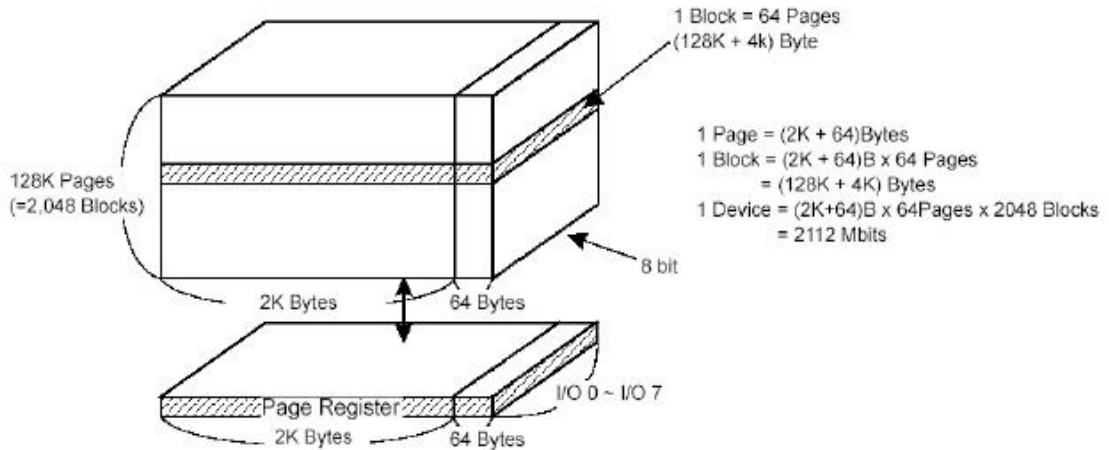
```
#false select nand, true select sd
#TARGET_USERIMAGES_USE_EXT4 := true
TARGET_USERIMAGES_USE_EXT4 := false

ifeq ($(TARGET_USERIMAGES_USE_EXT4),true)
TARGET_USERIMAGES_SPARSE_EXT_DISABLED := true
#TARGET_USERIMAGES_SPARSE_EXT_DISABLED := false
#BOARD_SYSTEMIMAGE_PARTITION_SIZE := 268435456
BOARD_SYSTEMIMAGE_PARTITION_SIZE := 241172480
#BOARD_USERDATAIMAGE_PARTITION_SIZE := 367001600
BOARD_USERDATAIMAGE_PARTITION_SIZE := 241172480
BOARD_FLASH_BLOCK_SIZE := 4096
endif

ifeq ($(TARGET_USERIMAGES_USE_EXT4),false)
#INTERNAL_USERIMAGES_USE_EXT := false
BOARD_NAND_PAGE_SIZE := 2048
BOARD_NAND_SPARE_SIZE := 64
endif
```

TARGET_USERIMAGES_USE_EXT4를 false로 해서 nand page와 space size를 설정합니다.

Figure 2-1. K9F2G08U0M Organization



삼성 낸드플래시 의 구조입니다.

yaffs 파일 시스템은 특별히 낸드 플래시를 위해 개발되었습니다.

파일 시스템을 yaffs2와 ext4 두가지 중 선택을 해야합니다.

android/device/crazyboys/mango210\$ vi init.rc

```
on fs
# mount mtd partitions
# Mount /system rw first to give the filesystem a chance to save a checkpoint
mount yaffs2 mtd@system /system
mount yaffs2 mtd@system /system rw remount
mount yaffs2 mtd@userdata /data nosuid nodev
mount yaffs2 mtd@cache /cache nosuid nodev
# mount ubifs ubi0:system /system
# mount ubifs ubi0:system /system rw remount
# mount ubifs ubi2:userdata /data nosuid nodev
# mount ubifs ubi1:cache /cache nosuid nodev
```

android/device/crazyboys/mango210\$ vi init.mango210.rc

```
#mount ext4 partitions
#mount ext4 /dev/block/mmcblk0p2 /system wait rw
#mount ext4 /dev/block/mmcblk0p3 /data wait rw noatime nosuid nodev
#mount ext4 /dev/block/mmcblk0p4 /cache wait rw noatime nosuid nodev
export EXTERNAL_STORAGE /mnt/sdcard
export SECONDARY_STORAGE /mnt/ext_sd:/mnt/usb
```

위와 같이 변경 후 ./android_build.sh를 실행합니다.

2. kernel build

```
## 7inch 800x480 LCD(감압식)
$ ./build_kernel defconfig mango210_7inch_nand256MB_wifi8787_defconfig
$ ./build_kernel
```

3. nand 부팅 방법

부트 이미지를 가진 sd카드를 SDIO 0/BOOT에 삽입합니다.
sd 부팅을 합니다.

```
mango210 보드 1, 3, 4 ON
CM-V210 보드 2, 3, 6 ON
```

usb otg, 3PIN 연결 후 전원을 인가합니다.

u-boot 실행 후
아무키나 누르고

```
MANGO210 # fdisk -c 0; fatformat mmc 0:1;fastboot
MANGO210 # nand scrub
MANGO210 # fastboot
```

[Linux PC]

```
$ cd image
$ sudo ./fastboot flash bootloader u-boot.bin
```

부팅 스위치를 NAND로 변경합니다.

```
Mango210 : 1,2 ON
CM-V210 : 1, 6 ON
```

전원인가하면 u-boot 실행 후 아무키나 누릅니다.

```
MANGO210 # fastboot
Fastboot: employ default partition information
[Partition table on NAND]
```

```
ptn 0 name='bootloader' start=0x0 len=0x100000(~1024KB)
ptn 1 name='kernel' start=0x100000 len=0x500000(~5120KB)
ptn 2 name='ramdisk' start=0x600000 len=0x300000(~3072KB)
ptn 3 name='system' start=0x900000 len=0xA00000(~163840KB) (Yaffs)
ptn 4 name='cache' start=0xA900000 len=0xA00000(~10240KB) (Yaffs)
ptn 5 name='userdata' start=0xB300000 len=N/A (Yaffs)
```

[Linux PC]

```
sudo ./fastboot -w flash kernel zImage
sudo ./fastboot flash ramdisk ramdisk-u-boot.img
sudo ./fastboot flash system system.img
sudo ./fastboot flash userdata userdata.img
```

재 부팅 후

```
sudo ./fastboot flash kernel zImage;
sudo ./fastboot flash system system.img;
sudo ./fastboot flash ramdisk ramdisk-uboot.img ;
sudo ./fastboot -w
```

이 후 리셋 버튼을 누르면 됩니다.

4. 소스 변경 사항

커널 변경 부분

dev-nand.c 를 추가합니다.

/kernel/arch/arm/mach-s5pv210\$ vi dev-nand.c

```
/*
 * linux/arch/arm/mach-s5pv210/dev-onenand.c
 *
 * Copyright (c) 2008-2010 Samsung Electronics
 * Kyungmin Park <kyungmin.park@samsung.com>
 *
 * S5PC110 series device definition for OneNAND devices
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License version 2 as
 * published by the Free Software Foundation.
 */
```

```

#include <linux/gfp.h>
#include <linux/kernel.h>
#include <linux/platform_device.h>

#include <linux/mtd/mtd.h>
#include <linux/mtd/partitions.h>

#include <mach/map.h>
#include <plat/devs.h>
#include <plat/nand.h>

static struct resource s3c_nand_resource[] = {
    [0] = {
        .start = S5P_PA_NAND,
        .end   = S5P_PA_NAND + SZ_1M,
        .flags = IORESOURCE_MEM,
    }
};

struct platform_device s3c_device_nand = {
    .name      = "s3c2440-nand",
    .id       = -1,
    .num_resources = ARRAY_SIZE(s3c_nand_resource),
    .resource  = s3c_nand_resource,
};

EXPORT_SYMBOL(s3c_device_nand);

static int __init s3c_nand_copy_set(struct s3c2410_nand_set *set)
{
    void *ptr;
    int size;

    size = sizeof(struct mtd_partition) * set->nr_partitions;
    if (size) {
        ptr = kmemdup(set->partitions, size, GFP_KERNEL);
        set->partitions = ptr;
    }
}

```



```

        if (!ptr)
            return -ENOMEM;
    }

    if (set->nr_map && set->nr_chips) {
        size = sizeof(int) * set->nr_chips;
        ptr = kmemdup(set->nr_map, size, GFP_KERNEL);
        set->nr_map = ptr;

        if (!ptr)
            return -ENOMEM;
    }

    if (set->ecc_layout) {
        ptr = kmemdup(set->ecc_layout,
                      sizeof(struct nand_ecclayout), GFP_KERNEL);
        set->ecc_layout = ptr;

        if (!ptr)
            return -ENOMEM;
    }

    return 0;
}

void __init s3c_nand_set_platdata(struct s3c2410_platform_nand *nand)
{
    struct s3c2410_platform_nand *npd;
    int size;
    int ret;

    /* note, if we get a failure in allocation, we simply drop out of the
     * function. If there is so little memory available at initialisation
     * time then there is little chance the system is going to run.
     */

    npd = kmemdup(nand, sizeof(struct s3c2410_platform_nand), GFP_KERNEL);

```

```

if (!npd) {
    printk(KERN_ERR "%s: failed copying platform data\n", __func__);
    return;
}

/* now see if we need to copy any of the nand set data */

size = sizeof(struct s3c2410_nand_set) * npd->nr_sets;
if (size) {
    struct s3c2410_nand_set *from = npd->sets;
    struct s3c2410_nand_set *to;
    int i;

    to = kmemdup(from, size, GFP_KERNEL);
    npd->sets = to; /* set, even if we failed */

    if (!to) {
        printk(KERN_ERR "%s: no memory for sets\n", __func__);
        return;
    }

    for (i = 0; i < npd->nr_sets; i++) {
        ret = s3c_nand_copy_set(to);
        if (ret) {
            printk(KERN_ERR "%s: failed to copy set %d\n",
                __func__, i);
            return;
        }
        to++;
    }
}

s3c_device_nand.dev.platform_data = npd;
}

```

.config파일은 Kconfig파일을 참고해서 만들어집니다.

Kconfig에서 y로 할지 n으로 할지 명시하면 컴파일 할 때 자동으로 취합해서 하나의 파일인 .config 파일이 됩니다.

kernel\arch\arm\mach-s5pv210\Kconfig

```
# arch/arm/mach-s5pv210/Kconfig
#
# Copyright (c) 2010 Samsung Electronics Co., Ltd.
#      http://www.samsung.com/
#
# Licensed under GPLv2

# Configuration options for the S5PV210/S5PC110

if ARCH_S5PV210

config CPU_S5PV210
    bool
    select S3C_PL330_DMA
    select S5P_EXT_INT
    select S5P_HRT
    select S5PV210_PM if PM
    help
        Enable S5PV210 CPU support

config S5PV210_SETUP_I2C1
    bool
    help
        Common setup code for i2c bus 1.

config S5PV210_SETUP_I2C2
    bool
    help
        Common setup code for i2c bus 2.

config S5PV210_SETUP_IDE
    bool
    help
```

Common setup code for S5PV210 IDE GPIO configurations

config S5PV210_SETUP_FB_24BPP

bool

help

Common setup code for S5PV210 with an 24bpp RGB display helper.

config S5PV210_SETUP_KEYPAD

bool

help

Common setup code for keypad.

config S5PV210_SETUP_SDHCI

bool

select S5PV210_SETUP_SDHCI_GPIO

help

Internal helper functions for S5PV210 based SDHCI systems

config S5PV210_SETUP_SDHCI_GPIO

bool

help

Common setup code for SDHCI gpio.

config S5PV210_SETUP_FIMC

bool

depends on VIDEO_FIMC

default y

help

Common setup code for the camera interfaces.

config S5PV210_SETUP_USB_PHY

bool

help

Common setup code for USB PHY controller

config S5PV210_POWER_DOMAIN

bool

depends on REGULATOR

```
default y
help
    Enable S5PV210 power domain support.
```

```
config S5PV210_DEV_NAND
```

```
bool
default y if MACH_MANGO210
help
    Compile in platform device definition for NAND controller
```

```
menu "S5PC110 Machines"
```

```
config MACH_AQUILA
```

```
bool "Aquila"
select CPU_S5PV210
select S3C_DEV_FB
select S5P_DEV_FIMC0
select S5P_DEV_FIMC1
select S5P_DEV_FIMC2
select S3C_DEV_HSMMMC
select S3C_DEV_HSMMMC1
select S3C_DEV_HSMMMC2
select S5P_DEV_ONENAND
select S5PV210_SETUP_FB_24BPP
select S5PV210_SETUP_SDHCI
help
    Machine support for the Samsung Aquila target based on S5PC110 SoC
```

```
config MACH_GONI
```

```
bool "GONI"
select CPU_S5PV210
select S5P_GPIO_INT
select S3C_DEV_FB
select S5P_DEV_FIMC0
select S5P_DEV_FIMC1
select S5P_DEV_FIMC2
select S3C_DEV_HSMMMC
select S3C_DEV_HSMMMC1
```

```
select S3C_DEV_HSMCMC2
select S3C_DEV_I2C1
select S3C_DEV_I2C2
select S3C_DEV_USB_HSOTG
select S5P_DEV_ONENAND
select SAMSUNG_DEV_KEYPAD
select S5PV210_SETUP_FB_24BPP
select S5PV210_SETUP_I2C1
select S5PV210_SETUP_I2C2
select S5PV210_SETUP_KEYPAD
select S5PV210_SETUP_SDHCI
help
    Machine support for Samsung GONI board
    S5PC110(MCP) is one of package option of S5PV210
```

```
config MACH_SMDKC110
    bool "SMDKC110"
    select CPU_S5PV210
    select S3C_DEV_I2C1
    select S3C_DEV_I2C2
    select S3C_DEV_RTC
    select S3C_DEV_WDT
    select S5P_DEV_ONENAND
    select SAMSUNG_DEV_IDE
    select S5PV210_SETUP_I2C1
    select S5PV210_SETUP_I2C2
    select S5PV210_SETUP_IDE
    help
        Machine support for Samsung SMDKC110
        S5PC110(MCP) is one of package option of S5PV210
```

endmenu

menu "NAND Select"

config NAND_512MB

```
    bool "NAND_512MB"
```

```
    help
```

```
        nand 512MB
```

```

config NAND_256MB
    bool "NAND_256MB"
    help
        nand 256MB
endmenu
menu "S5PV210 Machines"

config MACH_SMDKV210
    bool "SMDKV210"
    select CPU_S5PV210
    select S3C_DEV_FB
    select S3C_DEV_HSMMC
    select S3C_DEV_HSMMC1
    select S3C_DEV_HSMMC2
    select S3C_DEV_HSMMC3
    select S3C_DEV_I2C1
    select S3C_DEV_I2C2
    select S3C_DEV_RTC
    select S3C_DEV_WDT
    select S3C64XX_DEV_SPI
    select SAMSUNG_DEV_ADC
    select S5P_DEV_USB_HSDEVICE
    select S5P_DEV_USB_OHCI
    select S5P_DEV_USB_EHCI
    select HAVE_PWM
    select SAMSUNG_DEV_PWM
    select SAMSUNG_DEV_TS
    select S5PV210_SETUP_FB_24BPP
    select S5PV210_SETUP_I2C1
    select S5PV210_SETUP_I2C2
    select S5PV210_SETUP_IDE
    select S5PV210_SETUP_SDHCI
    select S5PV210_SETUP_USB_PHY
    help
        Machine support for SMDKV210
config MACH_MANGO210
    bool "MANGO210"
    select CPU_S5PV210

```

```
select S3C_DEV_FB
select S3C_DEV_HSMMC
select S3C_DEV_HSMMC1
select S3C_DEV_HSMMC2
select S3C_DEV_HSMMC3
select S3C_DEV_I2C1
select S3C_DEV_I2C2
select S3C_DEV_RTC
select S3C_DEV_WDT
#select S3C64XX_DEV_SPI
select SAMSUNG_DEV_ADC
select S5P_DEV_USB_HSDEVICE
select S5P_DEV_USB_OHCI
select S5P_DEV_USB_EHCI
select HAVE_PWM
select SAMSUNG_DEV_PWM
select SAMSUNG_DEV_TS
select S5PV210_SETUP_FB_24BPP
select S5PV210_SETUP_I2C1
select S5PV210_SETUP_I2C2
select S5PV210_SETUP_IDE
select S5PV210_SETUP_SDHCI
select S5PV210_SETUP_USB_PHY
help
    Machine support for MANGO210
```

```
config MACH_SMDKC110
    bool "SMDKC110"
    select CPU_SMDKC110
#    select ARCH_SPARSEMEM_ENABLE
    select S3C_DEV_FB
    select S3C_DEV_HSMMC
    select S3C_DEV_HSMMC1
    select S3C_DEV_HSMMC2
    select S3C_DEV_HSMMC3
    select S3C_DEV_I2C1
    select S3C_DEV_I2C2
```



```
select S3C_DEV_RTC
select S3C_DEV_WDT
select S3C64XX_DEV_SPI
select SAMSUNG_DEV_ADC
select SAMSUNG_DEV_IDE
select SAMSUNG_DEV_KEYPAD
select S5P_DEV_USB_HSDEVICE
select S5P_DEV_USB_OHCI
select S5P_DEV_USB_EHCI
select HAVE_PWM
select SAMSUNG_DEV_PWM
select SAMSUNG_DEV_TS
select S5PV210_SETUP_FB_24BPP
select S5PV210_SETUP_I2C1
select S5PV210_SETUP_I2C2
select S5PV210_SETUP_IDE
select S5PV210_SETUP_KEYPAD
select S5PV210_SETUP_SDHCI
select S5PV210_SETUP_USB_PHY
help
```

Machine support for Samsung SMDKC110

config MACH_TORBRECK

```
bool "Torbreck"
select CPU_S5PV210
select ARCH_SPARSEMEM_ENABLE
select S3C_DEV_HSMMC
select S3C_DEV_HSMMC1
select S3C_DEV_HSMMC2
select S3C_DEV_HSMMC3
select S3C_DEV_I2C1
select S3C_DEV_I2C2
select S3C_DEV_RTC
select S3C_DEV_WDT
select S5PV210_SETUP_I2C1
select S5PV210_SETUP_I2C2
select S5PV210_SETUP_SDHCI
help
```

Machine support for aESOP Torbreck

endmenu

config S5PV210_PM

bool

help

Power Management code common to S5PV210

config S5PV210_SETUP_FIMC0

bool

depends on VIDEO_FIMC || CPU_FREQ

default y

help

Common setup code for FIMC controller 0.

config S5PV210_SETUP_FIMC1

bool

depends on VIDEO_FIMC || CPU_FREQ

default y

help

Common setup code for FIMC controller 1.

config S5PV210_SETUP_FIMC2

bool

depends on VIDEO_FIMC || CPU_FREQ

default y

help

Common setup code for FIMC controller 2.

config S5PV210_SETUP_CSIS

bool

depends on VIDEO_FIMC

default y

help

Common setup code for MIPI-CSIS

endif

현재 nand 256MB와 512MB 선택할 수 있게 추가되었습니다.

₩kernel₩arch₩arm₩mach-s5pv210₩mach-mango210.c

아래는 변경된 점입니다.

```
#include <linux/gpio_keys.h>
#include <linux/leds.h>
    //by crazyboy 20130819
#include <linux/mtd/mtd.h>
#include <linux/mtd/nand.h>
#include <linux/mtd/nand_ecc.h>
#include <linux/mtd/partitions.h>

#include <plat/nand.h>

#ifdef CONFIG_SENSORS_L3G4200D
```

kernelWarchWarmWmach-s5pv210WMakefile

```
# machine support

obj-$(CONFIG_MACH_AQUILA)      += mach-aquila.o
obj-$(CONFIG_MACH_SMDKV210)   += mach-smdkv210.o
obj-$(CONFIG_MACH_MANGO210)  += mach-mango210.o
obj-$(CONFIG_MACH_SMDKC110)  += mach-smdkc110.o
obj-$(CONFIG_MACH_GONI)      += mach-goni.o
obj-$(CONFIG_MACH_TORBRECK)  += mach-torbreck.o

# device support

obj-y                          += dev-audio.o
obj-$(CONFIG_S5PV210_DEV_NAND) += dev-nand.o
obj-$(CONFIG_S3C64XX_DEV_SPI) += dev-spi.o
```

```
obj-$(CONFIG_S5PV210_DEV_NAND) += dev-nand.o
```

커널 configuration 파일에서 CONFIG_S5PV210_DEV_NAND 이 y로 정의가 되어 있습니다.

즉 ,

```
obj-y +=dev-nand.o
```

가 되는 것입니다.

커널 빌드 시 커널이미지에 built-in 하라는 의미입니다.

자세한 내용은

<http://lwn.net/Articles/21835/> 참조하시면 됩니다.

mtd는 memory technology device로 flash memory를 디스크처럼 사용하기 위한 기술입니다.,

kernel#wdrivers#wmtd# 아래에 실제적인 nand flash가 동작할 수 있는 드라이버가 있습니다.

kernel#wdrivers#wmtd#wnand#wkconfig

```
config MTD_NAND_S3C
```

```
    tristate "NAND Flash support for S3C SoC"
```

```
        depends on MTD_NAND && (ARCH_S5PC1XX || ARCH_S5PC11X || ARCH_S5PV2XX ||  
ARCH_S5PV210)
```

```
    help
```

```
        This enables the NAND flash controller on the S3C.
```

```
        No board specific support is done by this driver, each board  
        must advertise a platform_device for the driver to attach.
```

```
config MTD_NAND_S3C_DEBUG
```

```
    bool "S3C NAND driver debug"
```

```
    depends on MTD_NAND_S3C
```

```
    help
```

```
        Enable debugging of the S3C NAND driver
```

```
config MTD_NAND_S3C_HWECC
```

```
    bool "S3C NAND Hardware ECC"
```

```
    depends on MTD_NAND_S3C
```

```
    help
```

```
        Enable the use of the S3C's internal ECC generator when  
        using NAND. Early versions of the chip have had problems with  
        incorrect ECC generation, and if using these, the default of  
        software ECC is preferable.
```

```
        If you lay down a device with the hardware ECC, then you will  
        currently not be able to switch to software, as there is no  
        implementation for ECC method used by the S3C
```

config MTD_NAND_S5P

tristate "NAND Flash support for S5P SoC"

depends on MTD_NAND && (ARCH_S5PC1XX || ARCH_S5PC11X || ARCH_S5PV2XX || ARCH_S5PV210)

help

This enables the NAND flash controller on the S3C.

No board specific support is done by this driver, each board must advertise a platform_device for the driver to attach.

config MTD_NAND_S5P_DEBUG

bool "S5P NAND driver debug"

depends on MTD_NAND_S5P

help

Enable debugging of the S3C NAND driver

config MTD_NAND_S5P_HW ECC

bool "S5P NAND Hardware ECC"

depends on MTD_NAND_S5P

help

Enable the use of the S3C's internal ECC generator when using NAND. Early versions of the chip have had problems with incorrect ECC generation, and if using these, the default of software ECC is preferable.

If you lay down a device with the hardware ECC, then you will currently not be able to switch to software, as there is no implementation for ECC method used by the S3C

Makefile에 추가 합니다.

₩kernel₩drivers₩mtd₩nand₩Makefile

```
obj-$(CONFIG_MTD_NAND_S5P) += s5p_nand.o
```

```
obj-$(CONFIG_MTD_NAND_S3C) += s3c_nand.o
```

₩kernel₩drivers₩mtd₩nand₩ 밑에

s5c_nand.c 가 추가됩니다.

5. kernel config 파일 확인사항

위와 같이 커널을 변경하고 컴파일 후 .config파일에서 아래와 같이 수정 되었는지 확인합니다.

```
CONFIG_S5PV210_DEV_NAND=y

# NAND Select
#
# CONFIG_NAND_512MB is not set
CONFIG_NAND_256MB=y

CONFIG_MTD_NAND_S5P=y
# CONFIG_MTD_NAND_S5P_DEBUG is not set
CONFIG_MTD_NAND_S5P_HWECC=y
```