# Mango-IMX6Q Android 9.0 포팅가이드

http://www.mangoboard.com/ http://cafe.naver.com/embeddedcrazyboys Crazy Embedded Laboratory

# **Document History**

Revision	Date	Change note
Init	2020-09-03	전종인

1.	개발 환경	경	5
2.	문서 및	소스 다운로드	5
	2.1.	안드로이드 Pie 9.0, 커널,u-boot소스 다운로드 하기	6
	2.2.	U-boot, kernel, 안드로이드 컴파일 하기	7
	2.2.1.	Android 컴파일	7
	2.2.2.	U-boot 컴파일	7
	2.2.3.	커널 컴파일	8
3.	U-boot	디바이스 드라이버 포팅	8
	3.1.	U-boot 수정	8
	3.1.1.	RAM 살리기	8
	3.1.2.	PMIC 제거 에러 잡기	9
	3.1.3.	SD4 detect 하기	11
	3.1.4.	이더넷 포팅 하기	14
	3.1.5.	HDMI 확인 하기	15
	3.1.6.	Display 포팅하기	16
	3.1.7.	U-boot에서 로고 변경하기	20
4.	커널 디	바이스 드라이버 포팅	22
	4.1.	커널에서 불필요한 디바이스 삭제	22
	4.1.1.	Mango-IMX6Q에 없는 디바이스 config 없애기	22
	4.1.2.	PMIC 없으므로 LDO로 변경하기	25
	4.2.	Display , 터치 드라이브 포팅	25
	4.2.1.	LCD Power EN, Backlight GPIO 포팅	25
	4.2.2.	7인치 감압식 LCD 드라이버 포팅	
	4.2.3.	10.1인치 성선식 LCD 드라이버 포팅	
	4.2.4.	성선식 드라이버 포팅하기	
	4.2.5.	7인지 성선식 LCD 드라이버 포팅	
	4.2.6.	PMIC 없으므로 LDO로 변경하기	
	4.3.	USB Host 확인	
	4.4.	eMMC 모듈 데스트	
	4.5.	SDHC4 포팅	
	4.6.	Watchdog 포팅	
	4.7.	이미지 토 다이크드 쉐 머니	
	<b>4.ö.</b>	지미지 줄 나군도는 에 도자	
	4.8.1.	스크립느도 이용아어 이미지 Write아기	
	4.8.2.	꼬꼬새숙	49

# 1. 개발 환경

리눅스 PC 우분투 16.04 64bit 운영체제에서 개발을 한다. 빌드에 필요한 Package는 <u>https://source.android.com/setup/build/initializing</u> 참조해서 설치를 해야 한다.

Mango-IMX6Q 보드는 i.MX6 Quad를 사용하고 있다. 회로도, 하드웨어 매뉴얼이 필요하다. 회로도는 <u>help@crz-tech.com</u>으로 요청하면 된다. 하드웨어 매뉴얼은 <u>http://crztech.iptime.org:8080/Release/mango-imx6q/Doc/Hardware/</u> 에서 다운로드 받으면 된다. 다른 자료는 <u>http://crztech.iptime.org:8080/Release/mango-imx6q/</u> 에서 다운로드 받으면 된다.

# 2. 문서 및 소스 다운로드

https://www.nxp.com/design/software/embedded-software/i-mx-software/android-os-for-i-mxapplications-processors:IMXANDROID?&tab=Documentation\_Tab 링크에서 android\_p9.0.0\_2.2.0-ga\_docs.zip 파일을 다운로드 받는다. 다운로드 받으려면, nxp 홈페이지에 회원가입해야 한다.

압축을 풀면 Android\_Release\_Notes.pdf를 보면 i.MX6Quad Platform을 지원한다고 되어 있다. 관련 된 소스와 문서는 아래와 같이 정의가 되어 있다.

5

Android source code package	<ul> <li>imx-p9.0.0_2.2.0-ga.tar.gz: i.MX Android proprietary source code package to enable the Android platform on i.MX-based boards. For example, Hardware Abstraction Layer implementation and hardware codec acceleration.</li> </ul>
Documents	The following documents are included in android_p9.0.0_2.2.0-ga_docs.zip:
	<ul> <li>Android<sup>™</sup> Quick Start Guide (AQSUG): A document that explains how to run the Android platform on an i.MX board using prebuilt images.</li> <li>Android<sup>™</sup> User's Guide (AUG): A document describing procedures for configuring and building this release package.</li> <li>Android<sup>™</sup> Release Notes (ARN): A document that introduces key updates and known issues in this release.</li> <li><i>i.MX Android<sup>™</sup> Extended Codec Release Notes</i> (IMXACRN): A document that provides the extended codec information.</li> <li><i>i.MX Android<sup>™</sup> Camera Issues on the SDP Platform</i> (ACOI): A document that describes the camera issues on the SDP platform.</li> <li><i>i.MX Graphics User's Guide</i> (IMXGRAPHICUG): A document that describes GPU 2D API, Tools, Memory, and Application programming guidelines.</li> </ul>
Prebuilt images	You can test the Android platform with a prebuilt image on i.MX board before building any code:
	<ul> <li>android_p9.0.0_2.2.0-ga_image_6qsabresd.tar.gz:</li> </ul>
	Prebuilt images with NXP extended features for the SABRE-SD board.
	The extended features include more multimedia format support.
	<ul> <li>android_p9.0.0_2.2.0-ga_image_6qsabreauto.tar.gz:</li> </ul>
	Prebuilt images with NXP extended features for the SABRE-AI board.
	The extended features include more multimedia format support. • android_p9.0.0_2.2.0-ga_image_6sxsabresd.tar.gz:
	Prebuilt images with NXP extended features for the i.MX 6SoloX SABRE-SD board.
이제 포팅 할 준비가 되었디	, F

# 2.1. 안드로이드 Pie 9.0, 커널,u-boot소스 다운로드 하기

https://www.nxp.com/design/software/embedded-software/i-mx-software/android-os-for-i-mxapplications-processors:IMXANDROID?&tab=Design\_Tools\_Tab 링크에서 imx-p9.0.0\_2.2.0-ga.tar.gz를 다운로드 받는다. 해당 소스를 리눅스 PC 작업할 디렉토리에 다운로드 받는다. 압축을 푼다. 이제부터 android\_p9.0.0\_2.2.0-ga\_docs 디렉토리에 Android\_User's\_Guide.pdf 파일을 보면서 컴파일을 하면 된다. \$ tar xf imx-p9.0.0\_2.2.0-ga.tar.gz

\$ mkdir ~/bin

\$ curl <u>https://storage.googleapis.com/git-repo-downloads/repo</u> > ~/bin/repo

\$ chmod a+x ~/bin/repo

\$ export PATH=\${PATH}:~/bin						
\$ source ~/im	nx-p9.0.0_2.2.0-ga/imx_a	android_set	up.sh			
소스를 다운로	르드를 한다. 약 3시간 길	날린다.				
다운로드가 왼	<u></u> 토료가 되면, android_bu	ild 디렉토리	리가 생성(	이 되면서 다운희	르드 된 파일을 븕	볼 수 있다.
\$ cd android_	build/					
\$ ls						
Android.bp	SCR-p9.0.0_2.2.0-ga.txt	bootab	le	compatibil	ity developers	external
kernel	packages	prebuilts	test	vendor		
EULA.txt a	art	bootst	rap.bash	cts	development	frameworks
libcore	pdk	sdk	toolcha	in		
Makefile k	bionic	build		dalvik	device	hardware
libnativehelpe	r platform_testing sy	/stem 1	tools			

커널 소스는 vendor/nxp-opensource/kernel\_imx/ 디렉토리에 있다.

u-boot 소스는 vendor/nxp-opensource/uboot-imx/ 디렉토리에 있다.

# 2.2. U-boot, kernel, 안드로이드 컴파일 하기

안드로이드, 커널, u-boot 컴파일 해 보자.

# 2.2.1. Android 컴파일

Mango-IMX6Q 보드와 유사한 sabresd 보드 이름으로 컴파일 한다.

\$ cd android\_build

\$ source build/envsetup.sh

\$ lunch sabresd\_6dq-eng

\$ make -j 9 2>&1 | tee build-log.txt

"lunch sabreasd\_6dq-eng" 하면 보드는 sabreasd\_6dq이고, eng는 개발자 옵션을 넣어서 컴파일 하는 것이다. 좀 더 자세한 내용은 user guide를 살펴보기 바란다.

컴파일 시간이 약 1시간 이상 걸린다. 리눅스 PC 성능은 RAM 16GB 이상, 하드디스크는 300GB 이 상 있는 것이 좋다.

컴파일이 완료되면 아래와 같이 메시지가 나옵니다.

#### build completed successfully (01:37:58 (hh:mm:ss)) ####

# 2.2.2. U-boot 컴파일

\$ cd android\_build

\$ source build/envsetup.sh

\$ lunch sabresd\_6dq-eng

\$ make bootloader -j4

컴파일 결과 아래와 같이 메시지가 나옵니다.

#### build completed successfully (02:27 (mm:ss)) ####

# 2.2.3. 커널 컴파일

\$ export MY\_ANDROID=/home/icanjji/work/imx6q/android-work/android9.0-work/android\_build 경로는 각자 환경에 맞게 세팅한다. \$ cd \${MY\_ANDROID}/vendor/nxp-opensource/kernel\_imx \$ echo \$ARCH && echo \$CROSS\_COMPILE # Make sure you have those 2 environment variables set # If the two variables have not set, please set the as: \$ export ARCH=arm \$ export ARCH=arm \$ export CROSS\_COMPILE=\${MY\_ANDROID}/prebuilts/gcc/linux-x86/arm/arm-linux-androideabi-4.9/bin/arm-linux-androideabi-# Generate " config" according to default config file under arch/arm/config

# Generate ".config" according to default config file under arch/arm/configs.

# to build the kernel Image for i.MX 6Quad, 6QuadPlus, 6DualLite, 6Solo, 6SoloLite,

6SoloX ,7Dual and 7ULP

\$ make imx\_v7\_android\_defconfig

\$ make KCFLAGS=-mno-android

컴파일 하면 된다.

# 3. U-boot 디바이스 드라이버 포팅

# 3.1. U-boot 수정

u-boot를 Mango-IMX6Q에 맞게 수정해야 한다. 순서는 RAM-> Serial(Debug) -> Display(HDMI,10.1인치) -> 이더넷

#### 3.1.1. RAM 살리기

Mango-IMX6Q는 512MB\*4EA DDR3 장착되어 있다. RAM 설정 파일은 board/freescale/mx6sabresd/mx6q\_4x\_mt41j128.cfg에 정의가 되어 있다. Sabresd 보드는 1GB RAM이므로 설정 값을 변경한다. 같은 Sabreauto 보드에 설정값이 2GB RAM이다. 복사를 해 온다. 만약 같은 것이 없으면, 메모리 데 이터 시트와 i.MX6Q RAM 설정 부분을 보면서 모두 맞추어야 한다. board₩freescale₩mx6sabreauto₩imximage.cfg 파일에 RAM 설정 부분을 가지고 와서 똑같이 맞춘다. 컴파일 해서 RAM이 제대로 살아났는지 확인 해 보자.

테스트 결과 RAM은 2GB로 인식을 했다.

U-Boot 2018.03-dirty (Sep 04 2020 - 20:10:34 +0900)

CPU: Freescale i.MX6Q rev1.3 996 MHz (running at 792 MHz)

CPU: Extended Commercial temperature grade (-20C to 105C) at 36C

Reset cause: POR

Model: Freescale i.MX6 Quad SABRE Smart Device Board

Board: MX6-SabreSD

DRAM: 2 GiB

#### 3.1.2. PMIC 제거 에러 잡기

아래와 같이 에러가 발생을 한다.

U-Boot 2018.03-dirty (Sep 04 2020 - 20:10:34 +0900)

CPU: Freescale i.MX6Q rev1.3 996 MHz (running at 792 MHz)

CPU: Extended Commercial temperature grade (-20C to 105C) at 25C

Reset cause: POR

Model: Freescale i.MX6 Quad SABRE Smart Device Board

Board: MX6-SabreSD

DRAM: 2 GiB

read error from device: 89f18f10 register: 0x0!read error from device: 89f18f10 register: 0x3!PMIC: PFUZE100! DEV\_ID=0xffffffb REV\_ID=0xffffffb

read error from device: 89f18f10 register: 0x21!write error to device: 89f18f10 register: 0x21!read error from device: 89f18f10 register: 0x24!write error to device: 89f18f10 register: 0x24!read error from device: 89f18f10 register: 0x2f!write error to device: 89f18f10 register: 0x2f!read error from device: 89f18f10 register: 0x32!write error to device: 89f18f10 register: 0x32!write error from device: 89f18f10 register: 0x32!write error

initcall sequence 8ff9eb4c failed at call 17804cbc (err=-22)

### ERROR ### Please RESET the board ###

Mango-IMX6Q에서는 PFUZE100(PMIC) 사용하지 않는다.

board/freescale/mx6sabresd/mx6sabresd.c 파일을 수정

power\_init\_board함수를 호출 하는데

안드로이드 전체 이미지를 Write하기에는 시간이 많이 걸리므로, u-boot만 컴파일 후

Linux 이미지를 사용한다.

http://crztech.iptime.org:8080/Release/mango-imx6q/linux/kernel-4.1.15/20190321/mango-imx6q-kernel4.1.15-linux-10.1-image-20190321.tgz

다운로드 받아서, sd에 Write한다. U-boot.imx 파일을 복사한 후 아래 명령으로 Write하면 된다.

\$ sudo ./sdwriter-ubuntu16.04 sdb imx6q bin

u-boot 소스에 Device tree가 있다.

arch/arm/dts/imx6qdl-sabresd.dtsi 파일 수정해야 한다.

#if 0 //crazyboys 20200907

pmic: pfuze100@08

... #endif

PMIC 막으니까 아래와 같이 에러가 발생을 한다.

initcall sequence 8ff9eb4c failed at call 17804cbc (err=-19)

### ERROR ### Please RESET the board ###

찾아 보면 아래와 같다.

common/board\_f.c 파일에

void board\_init\_f(ulong boot\_flags)

=>

lib/initcall.c: printf("initcall sequence %p failed at call %p (err=%d)₩n",

common/board\_f.c 파일에

static const init\_fnc\_t init\_sequence\_f[] 에 순서가 정해져 있다. Pfuze 부분을 삭제 했으니 에러가 발 생한 것 같다.

찾았다. common/board\_r.c 파일에서

#### //crazyboys 20200907 power\_init\_board,

막으면 된다. 아래와 같이 console까지 간다.

U-Boot 2018.03-dirty (Sep 07 2020 - 14:16:16 +0900) CPU: Freescale i.MX6Q rev1.3 996 MHz (running at 792 MHz)

CPU: Extended Commercial temperature grade (-20C to 105C) at 33C

Reset cause: POR

Model: Freescale i.MX6 Quad SABRE Smart Device Board

Board: MX6-SabreSD

MANGO-IMX6Q-DRAM: 2 GiB

MMC: FSL\_SDHC: 0, FSL\_SDHC: 1, FSL\_SDHC: 2

Loading Environment from MMC... \*\*\* Warning - bad CRC, using default environment Failed (-5) No panel detected: default to Hannstar-XGA Display: Hannstar-XGA (1024x768) In: serial Out: serial Err: serial flash target is MMC:2 Net: No ethernet found.

Loading Environment from MMC... \*\*\* Warning - bad CRC, using default environment Failed (-5) 에러가 나오는 것은 SD가 제대로 detec가 되지 않아서 이다.

#### 3.1.3. SD4 detect 하기

CX-IMX6Q 회로도를 보면 SD4를 SD 부팅으로 사용한다.



board/freescale/mx6sabresd/mx6sabresd.c 소스에서 수정을 한다.

static iomux\_v3\_cfg\_t const usdhc4\_pads[] = {

IOMUX\_PADS(PAD\_SD4\_CLK\_SD4\_CLK| MUX\_PAD\_CTRL(USDHC\_PAD\_CTRL)),IOMUX\_PADS(PAD\_SD4\_CMD\_SD4\_CMD| MUX\_PAD\_CTRL(USDHC\_PAD\_CTRL)),IOMUX\_PADS(PAD\_SD4\_DAT0\_SD4\_DATA0| MUX\_PAD\_CTRL(USDHC\_PAD\_CTRL)),IOMUX\_PADS(PAD\_SD4\_DAT1\_SD4\_DATA1| MUX\_PAD\_CTRL(USDHC\_PAD\_CTRL)),

```
IOMUX_PADS(PAD_SD4_DAT2__SD4_DATA2 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
IOMUX_PADS(PAD_SD4_DAT3_SD4_DATA3 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
#if 0 //crazyboys 20200907 remove it
IOMUX_PADS(PAD_SD4_DAT4__SD4_DATA4 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
IOMUX_PADS(PAD_SD4_DAT5__SD4_DATA5 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
IOMUX_PADS(PAD_SD4_DAT6__SD4_DATA6 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
IOMUX_PADS(PAD_SD4_DAT7__SD4_DATA7 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
IOMUX_PADS(PAD_SD4_DAT7__SD4_DATA7 | MUX_PAD_CTRL(USDHC_PAD_CTRL)),
#endif
```

};

아래 cd 핀 수정, SD4는 cd pin을 사용하지 않음

```
int board mmc getcd(struct mmc *mmc)
{
        struct fsl_esdhc_cfg *cfg = (struct fsl_esdhc_cfg *)mmc->priv;
        int ret = 0;
        switch (cfg->esdhc_base) {
        case USDHC2 BASE ADDR:
                ret = !gpio_get_value(USDHC2_CD_GPIO);
                break:
        case USDHC3_BASE_ADDR:
                //ret = !gpio get value(USDHC3 CD GPIO);
                ret = 1; /* eMMC/uSDHC3 is always present */
                break;
        case USDHC4_BASE_ADDR:
                ret = 1; /* uSDHC4 is always present */
                break:
        }
        return ret:
```

arch/arm/dts/imx6qdl-sabresd.dtsi 수정

pinctrl_usdhc4: usdhc4grp {			
fsl,pins	= <		
	MX6QDL_PAD_SD4_CMDSD4_CMD	0x17059	
	MX6QDL_PAD_SD4_CLKSD4_CLK	0x10059	
	MX6QDL_PAD_SD4_DAT0_SD4_DATA0	0x17059	
	MX6QDL_PAD_SD4_DAT1SD4_DATA1	0x17059	

	MX6QDL_PAD_SD4_DAT2_SD4_DATA2 0x17059		
	MX6QDL_PAD_SD4_DAT3_SD4_DATA3 0x17059		
#if 0 //crazyboys 20200907			
	MX6QDL_PAD_SD4_DAT4_SD4_DATA4 0x17059		
	MX6QDL_PAD_SD4_DAT5SD4_DATA5 0x17059		
	MX6QDL_PAD_SD4_DAT6_SD4_DATA6 0x17059		
	MX6QDL_PAD_SD4_DAT7SD4_DATA7 0x17059		
#endif			
>;			
};			
&usdhc4 {			
pinctrl-names = "default"			
pinctrl-0 = <&pinctrl_usc	nc4>;		
bus-width = <4>;			
non-removable;			
no-1-8-v;			
keep-power-in-suspend;			
status = "okay";			
};			
include/configs/mx6sabresd.h 파을	수정		
#define CONFIG_SYS_MMC_ENV_E	EV 2 /* crazyboys 20200907 SDHC3-> SDHC4 */		
#define CONFIG_MMCROOT	"/dev/mmcblk3p2" /* crazyboys 20200907 SDHC3 -		
> SDHC4 */			
이미지를 Write 후 environment를 설정을 한다.			
setenv mmcdev 2			
전체 설정을 하려면 http://crztech	intime.org:8080/Release/mango-imx6g/linux/kernel-		

4.1.15/20190321/HowTo.txt 파일을 열어서 설정을 한다.

=> mmcinfo
Device: FSL_SDHC
Manufacturer ID: 3
OEM: 5344
Name: SC16G
Bus Speed: 5000000
Mode : SD High Speed (50MHz)
Rd Block Len: 512

SD version 3.0
High Capacity: Yes
Capacity: 14.8 GiB
Bus Width: 4-bit
Erase Group Size: 512 Bytes
확인 할 수 있다.

# 3.1.4. 이더넷 포팅 하기



Mango-IMX6Q는 PHY칩을 Athros8035를 사용한다. PHY Address는 0을 사용한다.

include/configs/mx6sabre\_common.h 파일에서 수정

#define IMX_FEC_BASE	ENET_BASE_ADDR
#define CONFIG_FEC_XCV_TYPE	RGMII
#ifdef CONFIG_DM_ETH	
#define CONFIG_ETHPRIME	"eth0"
#else	
#define CONFIG_ETHPRIME	"FEC"
#endif	
#define CONFIG_FEC_MXC_PHYADDR	0 /* crazyboys 20200907 1->0 */

테스트 방법 MAC address를 지정을 해야 한다.

=> setenv ethaddr 04:32:F4:FE:D5:1C

=> saveenv

=> dhcp fec

BOOTP broadcast 1 BOOTP broadcast 2 BOOTP broadcast 3 DHCP client bound to address 192.168.100.215 (1354 ms) \*\*\* Warning: no boot file name; using 'COA864D7.img' Using FEC device TFTP from server 0.0.0.0; our IP address is 192.168.100.215; sending through gateway 192.168.100.1 Filename 'COA864D7.img'. Load address: 0xfec Loading: \* TFTP error: 'File not found' (1) Not retrying... 이더넷이 ip를 할당 받으면 된다.

# 3.1.5. HDMI 확인 하기

=> setenv panel HDMI

=> saveenv

Saving Environment to MMC... Writing to MMC(2)... OK

=> reset

HDMI 포트에 LCD를 연결하면 아래와 같이 화면이 나온다.



#### 3.1.6. Display 포팅하기

회로도를 보면 LCD Power Enable, Backlight GPIO가 있다.

구분	Pad Name	GPIO
LCD_BL_CTRL	SD1_DAT2	GPIO1_19
DISP0_PWR_EN	ENET_TXD0	GPIO1_30

IMX6DQRM-Reference Manual.pdf 데이터시트를 보면 아래와 같이 정의가 되어 있다.

LCD\_BL\_CTRL과 DISP0\_PWR\_EN을 모두 High해야 LCD가 밝아져서 화면이 표시가 된다. 정확히 말하면 LCD\_BL\_CTRL 핀은 PWM 신호로 제어하는 것이 맞으나, u-boot에서 화면 밝기를 조절하지 않고, 커널에서 조절하는 것으로 하겠다. DISP0\_PWR\_EN핀은 LCD에 Power를 공급할 때 제어하는 핀이다.

board/freescale/mx6sabresd/mx6sabresd.c파일에서 아래와 같이 정의를 한다.

#define DISP0\_PWR\_EN IMX\_GPIO\_NR(1, 30)//crazyboys 20200907
#define DISP0\_BACKLIGHT\_EN IMX\_GPIO\_NR(1, 19) //crazyboys 20200907

static void enable\_backlight(void)
{
 SETUP\_IOMUX\_PADS(bl\_pads);
 gpio\_request(DISP0\_PWR\_EN, "Display Power Enable");
 gpio\_direction\_output(DISP0\_PWR\_EN, 1);
#if 1 //crazyboys 20200907
 gpio\_request(DISP0\_BACKLIGHT\_EN, "Display Backlight Enable");
 gpio\_direction\_output(DISP0\_BACKLIGHT\_EN, 1);
#endif

LCD에 따라서 타이밍 값을 아래와 같이 수정한다.

static struct display\_info\_t const displays[] 배열에 아래와 같이 추가한다.

```
.bus
           = 0,
    addr = 0
    .pixfmt = IPU_PIX_FMT_BGR24,
    .detect = NULL,
    .enable = enable rgb,
    .mode
            = {
        .name
                        = "MANGO-CAP7",//crazyboys 20151005
        .refresh
                    = 60,
        .xres
                     = 1024,//800,
        .yres
                      = 600, //480,
                    = 51200,//29850,
        .pixclock
        .left margin
                      = 60,
        .right_margin = 60,
        .upper_margin = 5,//23,
        .lower_margin = 5,//10,
        .hsync_len
                      = 200, 10,
        .vsync_len
                    = 25,//10,
                      = 0,
        .sync
        .vmode
                       = FB_VMODE_NONINTERLACED
} }, {
    .bus
           = 0,
    .addr
           = 0,
```

```
.pixfmt = IPU_PIX_FMT_BGR24,
    .detect = NULL
    .enable = enable rgb,
    .mode
            = {
        .name
                        = "MANGO-PRESS10.4",//crazyboys 20151005
        .refresh
                      = 60,
                       = 800,//800,
        .xres
        .yres
                       = 600, //480,
        .pixclock
                      = 20000,//51200,//29850,
        .left_margin
                      = 60,
        .right_margin = 60,
        .upper_margin = 5,//23,
        .lower_margin = 5_{1}/(10_{1})
        .hsync_len
                      = 200,10,
                      = 25,//10,
        .vsync_len
        .sync
                       = 0,
        .vmode
                       = FB_VMODE_NONINTERLACED
.bus
           = 0.
    .addr = 0,
    .pixfmt = IPU_PIX_FMT_BGR24,
    .detect = NULL,
    .enable = enable_rgb,
    .mode
            = {
        .name
                        = "MANGO-CAP10.1",//crazyboys 20151005
        .refresh
                      = 60,
                      = 1280, //800,
        .xres
                      = 800,//480,
        .yres
                      = 13468,//51200,//29850,
        .pixclock
        .left_margin
                      = 68,
        .right_margin = 60,
        .upper_margin = 8,//23,
        .lower_margin = 8_{i}/(10_{i})
        .hsync_len
                       = 2,
                       = 2_{1}/(10_{1})
        .vsync_len
                       = 0,
        .sync
                        = FB_VMODE_NONINTERLACED
        .vmode
```

} }, { .bus = 0, .addr = 0,.pixfmt = IPU\_PIX\_FMT\_BGR24, .detect = NULL.enable = enable\_rgb, .mode  $= \{$ = "MANGO-PRESS7",//crazyboys 20151005 .name .refresh = 60, = 800, .xres = 480, .yres .pixclock = 37879, .left\_margin = 46, .right\_margin = 16, .upper\_margin = 23, .lower\_margin =  $7_{1/10}$ , .hsync\_len = 1, .vsync\_len = 1,//10,.sync = 0,.vmode = FB\_VMODE\_NONINTERLACED u-boot에서 검증한다.

=> setenv panel MANGO-PRESS7

=> saveenv

Saving Environment to MMC...

Writing to MMC(2)... done

=> reset

resetting ...

리부팅하고 u-boot에서 프롬프트에서 멈추면 로고가 나온다.

<10.1" LCD>

setenv panel MANGO-CAP10.1

saveenv

reset

<7" 1024x600 LCD>

setenv panel MANGO-CAP7

saveenv

reset

U-Boot 2018.03-dirty (Sep 07 2020 - 17:41:30 +0900)

CPU Freescale i.MX6Q rev1.3 996 MHz (running at 792 MHz) CPU: Extended Commercial temperature grade (-20C to 105C) at 45C Reset cause: POR Model: Freescale i.MX6 Ouad SABRE Smart Device Board Board: MX6-SabreSD MANGO-IMX6Q-DRAM: 2 GiB fec\_phy\_reset MMC: FSL\_SDHC: 0, FSL\_SDHC: 1, FSL\_SDHC: 2 Loading Environment from MMC... OK Display: MANGO-CAP7 (1024x600) IPU DMFC NORMAL mode: 1(0~1), 5B(4,5), 5F(6,7) panel size =  $1024 \times 600$ pixel clk = 19531000Hz crz pixel fmt=861030210 IPU DMFC DP HIGH RES: 1(0,1), 5B(2~5), 5F(6,7)

아래와 같이 에러 발생 시 수정 방법은 아래와 같다.

Display: MANGO-PRESS10.4 (800x600)

mxcfb: Error initializing panel.

"IPU\_PIX\_FMT\_BGR24" 정의가 되어 있는데, 제대로 동작을 하지 않는다.

.pixfmt = IPU\_PIX\_FMT\_BGR24,

drivers/video/ipu\_disp.c 파일에서 "IPU\_PIX\_FMT\_BGR24" 추가한다.

static int ipu\_pixfmt\_to\_map(uint32\_t fmt)

switch (fmt) {
case IPU\_PIX\_FMT\_GENERIC:
case IPU\_PIX\_FMT\_RGB24:
case IPU\_PIX\_FMT\_BGR24: //crazyboys 20200907

#### 3.1.7. U-boot에서 로고 변경하기

https://cafe.naver.com/embeddedcrazyboys/40199

글을 참조해서 변경 한다.

tools/logos/

디렉토리에

{

```
freescale.bmp 파일을 변경하면 됩니다.
해당 파일은
tools/Makefile 에서 보면
아래와 같이 컴파일이 되어 진다.
# Use board logo and fallback to vendor
ifneq ($(wildcard $(srctree)/$(src)/logos/$(BOARD).bmp),)
LOGO_BMP= $(srctree)/$(src)/logos/$(BOARD).bmp
else
ifneq ($(wildcard $(srctree)/$(src)/logos/$(VENDOR).bmp),)
LOGO_BMP= $(srctree)/$(src)/logos/$(VENDOR).bmp
endif
endif
아래 명령이 실행되면서 만들어 진다.
              $(obj)/bmp_logo $(LOGO_BMP)
$(LOGO H):
      $(obj)/bmp_logo --gen-info $(LOGO_BMP) > $@
[출처] u-boot 로고 이미지 변경하기 (Embedded Crazyboys) | 작성자 머털도사
```

7인치 감압식 LCD를 사용하면 해상도가 800x480이므로 800x480에 8bit에 BMP 파일을 만든다.



# **CQZ**, TECHNOLOGY

# 4. 커널 디바이스 드라이버 포팅

# 4.1. 커널에서 불필요한 디바이스 삭제

커널에서 Mango-IMX6Q에서 사용하지 않는 Device부터 삭제하고, 디바이별로 포팅하는 것이 좋다.

# 4.1.1. Mango-IMX6Q에 없는 디바이스 config 없애기

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일을 수정하면 된다. Device tree는 공부를 해야 한다.

```
#if 0 //crazyboys 20200907
    hannstar_cabc {
        compatible = "hannstar,cabc";
        lvds0 {
            gpios = <&gpio6 15 GPIO_ACTIVE_HIGH>;
        };
        lvds1 {
            gpios = <&gpio6 16 GPIO_ACTIVE_HIGH>;
        lvds1 {
            gpios = <&gpio6 16 GPIO_ACTIVE_HIGH>;
        l;
        lvds1 {
            gpios =
```

```
reg = \langle 0x1a \rangle;
        clocks = <&clks IMX6QDL CLK CKO>;
        DCVDD-supply = <&reg_audio>;
        DBVDD-supply = <&reg_audio>;
        AVDD-supply = <&reg_audio>;
        CPVDD-supply = <&reg_audio>;
        MICVDD-supply = <&reg audio>;
        PLLVDD-supply = <&reg_audio>;
        SPKVDD1-supply = <&reg_audio>;
        SPKVDD2-supply = <&reg_audio>;
        qpio-cfq = <
                 0x0000 /* 0:Default */
                 0x0000 /* 1:Default */
                 0x0013 /* 2:FN_DMICCLK */
                 0x0000 /* 3:Default */
                 0x8014 /* 4:FN_DMICCDAT */
                 0x0000 /* 5:Default */
        >;
        amic-mono:
};
mma8451@1c {
        compatible = "fsl,mma8451";
        reg = \langle 0x1c \rangle;
        position = <0>;
        vdd-supply = <&reg_sensor>;
        vddio-supply = <&reg_sensor>;
        interrupt-parent = <&gpio1>;
        interrupts = <18 \ 8>;
        interrupt-route = <1>;
};
ov564x: ov564x@3c {
        compatible = "ovti,ov564x";
        reg = \langle 0x3c \rangle;
        pinctrl-names = "default";
        pinctrl-0 = <&pinctrl_ipu1_2>;
        clocks = <&clks IMX6QDL_CLK_CKO>;
```

```
clock-names = "csi_mclk";
                DOVDD-supply = <&vgen4_reg>; /* 1.8v */
                AVDD-supply = <&vgen3_reg>; /* 2.8v, on rev C board is VGEN3,
                                                  on rev B board is VGEN5 */
                DVDD-supply = <&vgen2_reg>; /* 1.5v*/
                pwn-gpios = <&gpio1 16 1>; /* active low: SD1_DAT0 */
                rst-gpios = <&gpio1 17 0>; /* active high: SD1 DAT1 */
                csi id = \langle 0 \rangle;
                mclk = <2400000>;
                mclk_source = <0>;
        };
#endif
#if 0 //crazyboys 0907
        egalax_ts@04 {
                compatible = "eeti,egalax_ts";
                reg = <0x04>;
                pinctrl-names = "default";
                pinctrl-0 = <&pinctrl_i2c2_egalax_int>;
                interrupt-parent = <&gpio6>;
                interrupts = \langle 8 \rangle;
                wakeup-gpios = <&gpio6 8 0>;
        };
        max11801@48 {
                compatible = "maxim,max11801";
                req = <0x48>;
                interrupt-parent = <&gpio3>;
                interrupts = <26 2>;
                work-mode = <1>;/*DCM mode*/
        };
        pmic: pfuze100@08 {
                compatible = "fsl,pfuze100";
                reg = <0x08>;
#endif
#if 0
        ov564x_mipi: ov564x_mipi@3c { /* i2c2 driver */
                compatible = "ovti,ov564x_mipi";
```

```
reg = <0x3c>;
clocks = <&clks 201>;
clock-names = "csi_mclk";
DOVDD-supply = <&vgen4_reg>; /* 1.8v */
AVDD-supply = <&vgen3_reg>; /* 2.8v, rev C board is VGEN3
rev B board is VGEN5 */
DVDD-supply = <&vgen2_reg>; /* 1.5v*/
pwn-gpios = <&gpio1 19 1>; /* active low: SD1_CLK */
rst-gpios = <&gpio1 20 0>; /* active high: SD1_DAT2 */
csi_id = <1>;
mclk = <24000000>;
mclk_source = <0>;
};
#endif
```

# 4.1.2. PMIC 없으므로 LDO로 변경하기

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일 수정

```
&reg_arm {
		/delete-property/ vin-supply;
};
&reg_pu {
		/delete-property/ vin-supply;
};
&reg_soc {
		/delete-property/ vin-supply;
};
```

# 4.2. Display , 터치 드라이브 포팅

#### 4.2.1. LCD Power EN, Backlight GPIO 포팅

LCD 화면을 디스플레이하기 위해서는 아래 회로도에 GPIO 핀을 컨트롤해야 한다.

구분	Pad Name	GPIO
LCD_BL_CTRL	SD1_DAT2	GPIO1_19

DISP0_PWR_EN	ENET_TXD0	GPIO1_30	
10 SD1 DAT2 CSI0_	RST_N -	EIM A20	찾기
12 SD1 DA13 SLCD_F	BL_CTRL -	EIM A21	
14	1_10_1 —	EIM_A22	DISPO_PWR_EN
16		EIM_A23	이저 다
		EIM_A24	
		EIM_A25	19
22 SD2_DAT0		<b>EN1 D</b> 10	21
		EIM_D16	23
26 SD2_DAT3		EIM_D17	25
28	_	EIM_D18	27
30 SD4 CL		EIM_D19	29
32 SD4 CM		EIM_D20	31
34 SD4 DA	T0		33
30 SD4 DA	T1 —		35
38 SD4 DA	T2 —	LIM_DZ3	37
40 SIM HALAA SD4 DA	T3	EIM D24	39
42 SD4_DATA4 SD4_CD	N —	EIM_024	41
44 SD4_DATA6 KLVDS0_I	0_0	EIM_D26	45
40 SD4_DATA7 (LVDS0_I	O_1 should have TS_INT_N	FIM D27	43
50 CLVDS1_I	O_0 internal IO KEY1 >>-	FIM D28	47
52	pull-ups KEY2 🌺	FIM D29	51
54 SATA_TXP	PCIE_PWR_EN 🛠—	FIM D30	53
56 SATA_TXM	—	EIM D31	55
58		EIM_LBA	57
60 SATA_RXM Check	if EB2/EB3 is interruptable PIN	EIMOE	59
62 K SATA_RXP	· ·	EIM_RW	61
64		EIM_CS0	63
66 // SD2_CMD		EIM_CS1	65
		EIM_EB2	67
		EIM_EB3	69
72 SD3_DAT1		FIM_BCLK	71
74 SD3_DAT2		EIM_WAIT	73
76 SD3 DAT3	_	EIM_EB0	75
78 SD3 DAT4		EIM_EB.I	11
80 SD3 DAT5	NVCC_SD3	ENET MDC	79
SD3 DAT6			81
SD3 DAT7	RGMII MDIO S		83
SD3 RST	Y RGMILRST N ≫		85
00	= =	ENEL RX FR	80
02 VDD 5v0 11	USB_OTG_ID >>-	ENEL IX EN	09 Q1
94 0 to 5V s	vlaan	ENET RXD0	93
96 (\		ENET RXD1	95
98 VISB_HOST_DP	RGMIL_INT >>	ENELIXUU	97
100 USB_HOST_DN	DISPO PWR EN <		

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일에서 아래와 같이 수정을 한다.

```
mxcfb1: fb@0 {
    compatible = "fsl,mxc_sdc_fb";
    disp_dev = "lcd";
    interface_pix_fmt = "BGR24";
    //interface_pix_fmt = "RGB24";
    mode_str ="MANGO-CAP7";
    default_bpp = <24>;
    int_clk = <0>;
    late_init = <0>;
    status = "disabled";
    };
    lcd@0 {
        compatible = "fsl,lcd";
    }
}
```

```
ipu id = \langle 0 \rangle;
                 disp id = \langle 0 \rangle;
                 default ifmt = "BGR24";
                 pinctrl-names = "default";
                 pinctrl-0 = <&pinctrl_ipu1>;
                 power_en_gpio = <&gpio1 30 0>; /* Power EN */
                 backlight ctl gpio = <&gpio1 19 0>; /* Backlight CTRL */
                status = "okay";
Default ifmt가 BGR24인 것은 LCD와 연결하는 핀맵이 B->G->R로 연결되어 있다.
"power en gpio"와 "backlight ctl gpio"는
drivers/video/fbdev/mxc/mxc_lcdif.c 파일에 추가를 한다.
//crazyboys 20200908
#include <linux/gpio.h>
#include <linux/of_gpio.h>
static int mxc lcdif probe(struct platform device *pdev)
{
        int ret;
        struct pinctrl *pinctrl;
        struct mxc lcdif data *lcdif;
        struct mxc_lcd_platform_data *plat_data;
//crazyboys 20200908
        struct device_node *np = pdev->dev.of_node;
        enum of gpio flags flags;
        unsigned int pwr_gpio,backlight_gpio;
        int status=-1;
//end
//crazyboys 20200908
#if 1
        pwr_gpio = of_get_named_gpio_flags(np, "power_en_gpio", 0, &flags);
        if(pwr_gpio == -EPROBE_DEFER){
                 dev_err(&pdev->dev, "LCD Power Enalbe Gpio fail₩n");
                 return pwr_gpio;
        }
        status = devm_gpio_request_one(&pdev->dev,pwr_gpio,GPIOF_OUT_INIT_HIGH,NULL);
```

```
if(status < 0)
        {
                 dev_err(&pdev->dev, "failed to request gpio %d: %d₩n",pwr_gpio,status);
                 return status;
        }
        backlight_gpio = of_get_named_gpio_flags(np, "backlight_ctl_gpio", 0, &flags);
        if(backlight_gpio == -EPROBE_DEFER){
                 dev_err(&pdev->dev, "Backlight Ctrl Gpio fail₩n");
                 return backlight_gpio;
        }
        status = devm_gpio_request_one(&pdev->dev,backlight_gpio,GPIOF_OUT_INIT_HIGH,NULL);
        if(status < 0)
        {
                 dev_err(&pdev->dev, "Backlight Ctrl GPIO Request Fail status %d\#n",status);
                 return status;
        }
#endif
```

probe하면서 GPIO를 High한다.

PWM으로 Backlight를 조정할 수 있다.

IMX6DQRM-Reference Manual.pdf 파일을 보면 PWM2\_OUT으로 사용 할 수 있다.

SD1_DAT2	ALT0	SD1_DATA2	HYS - ENABLED	SW_PAD_CTL_PAD_SD1_DATA2
	ALT1	ECSPI5_SS1	PUS - 100K_OHM_PU	
	ALT2	GPT_COMPARE2	PUE - PULL	
	ALT3	PWM2_OUT	PKE - ENABLED	
	ALT4	WDOG1_B		

Table continues on the next page ...

#### i.MX 6Dual/6Quad Applications Processor Reference Manual, Rev. 3, 07/2015

286

Freescale Semiconductor, Inc.

#### Chapter 4 External Signals and Pin Multiplexing

Table 4-1.	Pin Assignments	(continued)
------------	-----------------	-------------

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT5	GPIO1_IO19	ODE - DISABLED	
	ALT6	WDOG1_RESET_B_DEB	SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	

#### arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일 수정

backlight {	
	compatible = "pwm-backlight";
	//pwms = <&pwm2 0 50000>;
	pwms = <&pwm2 0 400000>;//2.5KHz
	//pwms = <&pwm2 0 40000>;//25KHz
	//pwms = <&pwm2 0 4000>;//250KHz
	brightness-levels = <
	0 1 2 3 4 5 6 7 8 9
	10 11 12 13 14 15 16 17 18 19
	20 21 22 23 24 25 26 27 28 29
	30 31 32 33 34 35 36 37 38 39
	40 41 42 43 44 45 46 47 48 49
	50 51 52 53 54 55 56 57 58 59
	60 61 62 63 64 65 66 67 68 69
	70 71 72 73 74 75 76 77 78 79
	80 81 82 83 84 85 86 87 88 89
	90 91 92 93 94 95 96 97 98 99
	100
	>;
	default-brightness-level = <100>;

```
status = "okay";
```

```
pinctrl_pwm2: pwm2grp {
```

};

테스트 방법

# echo 10 > /sys/class/backlight/backlight/brightness => 어둡게 # echo 100 > /sys/class/backlight/backlight/brightness => 밝게

# 4.2.2. 7인치 감압식 LCD 드라이버 포팅

드라이버 포팅을 위해서 7인치 감압식 LCD(AT070TN94) 데이터 시트, Mango-IMX6Q 회로도가 필요 하다. LCD 드라이버에 타이밍을 보기 위해서 7인치 감압식 LCD 데이터 시트가 필요하다. 데이터시트를 보면 타이밍이 아래와 같이 나와 있다.

lterre	Complex		Values	11	Dented	
nem	Symbol	Min.	Тур.	Max.	Unit	Remark
Horizontal Display Area	thd	-1	800	-	DCLK	
DCLK Frequency	fclk	26.4	33.3	46.8	MHz	
One Horizontal Line	th	862	1056	1200	DCLK	
HS pulse width	thpw	1	-	40	DCLK	
HS Blanking	thb	46	46	46	DCLK	
HS Front Porch	thfp	16	210	354	DCLK	

Itom	Symbol	Values			Unit	Bomark
Rem		Min.	Тур.	Max.	Unit	Remark
Vertical Display Area	tvd	0	480	1 -	тн	
VS period time	tv	510	525	650	тн	
VS pulse width	tvpw	1	-	20	тн	
VS Blanking	tvb	23	23	23	тн	
VS Front Porch	tvfp	7	22	147	тн	

타이밍 표를 보고, 타이밍을 맞추면 된다.

DCLK Frequency를 26.4MHz로 했을 때 타이밍 값이다.

drivers/video/fbdev/mxc/mxc\_lcdif.c 파일에 static struct fb\_videomode lcdif\_modedb[] 배열에 추가하 면 된다.

"MANGO-PRESS7",	//name
60,	//refresh
800,	//xres
480,	//yres
37879,	//pixclock(ns)
46,	//left_margin(HBP)
16,	//right_margin(HFP)
23,	//upper_margin(VBP)
7,	//lower_margin(VFP)
1,	//hsync_len

1,

//vsync\_len

FB\_SYNC\_CLK\_LAT\_FALL, //sync

FB\_VMODE\_NONINTERLACED | FB\_VMODE\_ODD\_FLD\_FIRST, //vmode

0,}, //flag

터치 드라이버 포팅 감압식 LCD이다.

중간에 tsc2007 칩이 있다.

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일에 추가

touchscreen: tsc2	2007@4a {
	compatible = "ti,tsc2007";
	reg = <0x4a>;
	pinctrl-names = "default";
	pinctrl-0 = <&pinctrl_mango_ts>;
	interrupt-parent = <&gpio3>;
	interrupts = <26 0>;
	gpios = <&gpio3 26 GPIO_ACTIVE_LOW>;
	ti,x-plate-ohms = <660>;
	linux,wakeup;
};	

커널 드라이버는 drivers/input/touchscreen/tsc2007\_core.c 사용하면 된다.

#### 테스트 방법은 아래와 같이 하면 된다. 리눅스 파일 시스템에서 테스트 했다.

```
# ts calibrate
xres = 800, yres = 480
Took 7 samples...
Top left : X = 337 Y = 534
Took 7 samples...
Top right : X = 3727 Y = 590
Took 8 samples...
Bot right : X = 3683 Y = 3496
Took 7 samples...
Bot left : X = 314 Y = 3519
Took 5 samples...
Center : X = 2045 Y = 2030
-23.375244 0.207105 0.002348
-21.038757 -0.000633 0.128980
Calibration constants: -1531920 13572 153 -1378796 -41 8452 65536
[CRZ] drivers/input/touchscreen/tsc2007_core.c (193) tsc2007_stop:
# ts_test
```

## 4.2.3. 10.1인치 정전식 LCD 드라이버 포팅

10.1인치 타이밍은 아래와 같이 맞춘다.

drivers/video/fbdev/mxc/mxc\_lcdif.c 파일 수정

"MANGO-CAP10.1",	//name
60,	//refresh
1280,	//xres
800,	//yres
13468,	//pixclock(ns)
68,	//left_margin(HBP)
60,	//right_margin(HFP)
8,	//upper_margin(VBP)
8,	//lower_margin(VFP)
2,	//hsync_len
2,	//vsync_len
0, //sync	
FB_VMODE_NONINTERLACED, //vi	mode
0,}, //flag	
테스트 결과	

# fbset		
mode "1280x800-64"		
# D: 74.250 MHz, H: 52.660 kHz, V: 64.376 Hz		
geometry 1280 800 1280 800 24		
timings 13468 68 60 8 8 2 2		
accel false		
rgba 8/16,8/8,8/0,0/0		
endmode		

# 4.2.4. 정전식 드라이버 포팅하기

7인치 정전식 LCD와 10.1인치 정전식 LCD에 사용하는 터치 드라이버이다. 정전식 터치를 사용한다. CX-IMX6Q 회로도를 보면, I2C 2번 bus를 사용한다. 인터럽터 핀과 터치 리셋 핀이 있다.

구분	핀 이름	GPIO	
CT_RST_N	SD1_CMD	GPIO1_IO18	
C_TS_INT_N	EIM_D26	GPIO3_IO26	

우선 핀을 device tree에서 정의를 한다.

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일 수정



U1 부품 6번 PIN이 LOW를 유지한다.





CR-LIF2-CT101R001 LCD보드에서 TSC2007 과 연결되는 C\_TOUCH\_XP, C\_TOUCH\_YP, C\_TOUCH\_XM, C\_TOUCH\_YM 핀이 Ground로 되어 있어, 터치가 눌린 것으로 인식한다.

정전식 Touch를 사용할 경우 TSC2007 칩에 IRQ Disable을 하여 사용하도록 수정

#### COMMAND BYTE

	Table 2. Command Byte Demittion (Excluding the Setup Command)			
BIT	NAME	DESCRIPTION		
D7-D4	C3-C0	All Converter Function Select bits as detailed in Table 3, except for the setup command ('1011').		
D3-D2	PD1-PD0	00: Power down botween cycles. <u>PENIRQ</u> enabled. 01: A/D converter on. <u>PENIRQ</u> disabled. 10: A/D converter on. <u>PENIRQ</u> disabled. 11: A/D converter on. <u>PENIRQ</u> disabled.		
D1	м	0: 12-bit (Lower speed referred to as the 2MHz clock). 1: 8-bit (Higher speed referred to as the 4MHz clock).		
D0	х	Don't care.		

Table 2 Command Bute Definition (Evoluting the Setue Command)<sup>(1)</sup>

커널 imx6qdl-sabresd.dtsi 파일 수정한다.

mango-ts@38 {
compatible = "mango,mango-ts";
reg = <0x38>;
pinctrl-names = "default";
pinctrl-0 = <&pinctrl_mango_ts>;
interrupt-parent = <&gpio3>;
interrupts = <26 0>;
//resets = <&mango_ts_reset>;
reset-gpios = <&gpio1 18 GPIO_ACTIVE_LOW>;
};
tsc2007@4a {
compatible = "ti,tsc2007";
$reg = \langle 0x4a \rangle;$
//pinctrl-names = "default";
//pinctrl-0 = <&pinctrl_mango_ts>;
//interrupt-parent = <&gpio3>;
//interrupts = <26 0>;
//gpios = <&gpio3 26 GPIO_ACTIVE_LOW>;
// ti,x-plate-ohms = <660>;
// linux,wakeup;
};

터치 드라이버는 http://crztech.iptime.org:8080/Release/mango-imx6q/Android6.0/20171204/mangoimx6qxf-android6.0-10.1inch-sd-wifi-5M-MIPI-src-20171204.tgz 에서 터치 드라이버를 추출한다.

mango\_ft5x06\_ts.c, mango\_ft5x06\_ts.h 파일이다.

컴파일 시 에러가 발생을 한다.

drivers/input/touchscreen/mango ft5x06 ts.c:1419:13: error: 'struct file' has no member named 'f dentry' ⇒ Include/linux/fs.h 파일에 정의가 되어 있지 않다. 아래와 같이 수정한다. struct file { union { struct llist node fu llist; struct rcu\_head fu\_rcuhead; } f\_u; struct path f path: #define f\_dentry f\_path.dentry //crazyboys 20200910 drivers/input/touchscreen/mango\_ft5x06\_ts.c 파일에서 include 파일 수정. 드라이버가 3.x 커널에서 사용하던 거라, 커널 4.14로 되면서 많이 변경이 되었네요. #include <linux/module.h> #include <linux/init.h> #include <linux/i2c.h> #include <linux/interrupt.h> #include <linux/input.h> #include <linux/irg.h> #include <linux/gpio.h> #include <linux/delay.h> #include <linux/slab.h> #include <linux/bitops.h>

#include <linux/input/mt.h>
#include <linux/of\_gpio.h>

#include <linux/reset.h>

#include <linux/fs.h>

drivers/input/touchscreen/Makefile 파일 추가

obj-\$(CONFIG\_MANGO\_TOUCH\_FT5x06\_101INCH) += mango\_ft5x06\_ts.o obj-\$(CONFIG\_MANGO\_TOUCH\_FT5x06\_7INCH) += mango\_ft5x06\_ts.o

drivers/input/touchscreen/Kconfig 파일에 아래 내용 추가

config MANGO\_TOUCH\_FT5x06\_7INCH

tristate "crztech mango 7 inch FT5x06 PCAP touch "

depends on INPUT\_TOUCHSCREEN

default n

help

Say Y here to enable the driver for the 10.1" FT5x06 PCAP touchscreen on the Mango bard.

board.

If unsure, say N. To compile this driver as a module, choose M here: the module will be called  $s3c_ts$ .

config MANGO\_TOUCH\_FT5x06\_7INCH\_90\_ROT

tristate "90 Rotation mango 7 inch FT5x06 PCAP touch "

depends on INPUT\_TOUCHSCREEN

default n

help

Say Y here to enable the driver for the 7" FT5x06 PCAP touchscreen on the Mango board.

If unsure, say N. To compile this driver as a module, choose M here: the module will be

called s3c\_ts.

config MANGO\_TOUCH\_FT5x06\_7INCH\_270\_ROT

tristate "270 Rotation mango 7 inch FT5x06 PCAP touch "

depends on INPUT\_TOUCHSCREEN

default n

help

Say Y here to enable the driver for the 7" FT5x06 PCAP touchscreen on the Mango board.

If unsure, say N. To compile this driver as a module, choose M here: the module will be called s3c ts.

config MANGO\_TOUCH\_FT5x06\_101INCH

tristate "crztech mango 10.1 inch FT5x06 PCAP touch "

depends on INPUT\_TOUCHSCREEN

default n
help
Say Y here to enable the driver for the 10.1" FT5x06 PCAP touchscreen on the Mango
board.
If unsure, say N. To compile this driver as a module, choose M here: the module will be
called s3c_ts.
config MANGO_TOUCH_FT5x06_10INCH_90_ROT
tristate "90 Rotation mango 10.1 inch FT5x06 PCAP touch "
depends on INPUT_TOUCHSCREEN
default n
help
Say Y here to enable the driver for the 10.1" FT5x06 PCAP touchscreen on the Mango
board.
If unsure, say N. To compile this driver as a module, choose M here: the module will be
called s3c_ts.
config MANGO_TOUCH_FT5x06_10INCH_270_ROT
tristate "270 Rotation mango 10.1 inch FT5x06 PCAP touch "
depends on INPUT_TOUCHSCREEN
default n
help
Say Y here to enable the driver for the 10.1" FT5x06 PCAP touchscreen on the Mango
board.
If unsure, say N. To compile this driver as a module, choose M here: the module will be
called s3c_ts.

커널 Configuration에서 CONFIG\_MANGO\_TOUCH\_FT5x06\_101INCH=y 추가한다.

tsc2007.c 파일에서 수정 한다. drivers/input/touchscreen/Kconfig 파일에 아래 내용 추가

config MANGO\_TSC2007\_IRQ\_DISABLE

tristate "TSC2007 based touchscreens irq disable" depends on I2C help Say Y here if you have a TSC2007 based touchscreen irq disable.

If unsure, say N.

To compile this driver as a module, choose M here: the module will be called tsc2007.

```
"tsc2007_core.c" 파일 수정
```

```
static void tsc2007_stop(struct tsc2007 *ts)
{
        int err;//crazyboys 20160517
        //MANGO_DBG("₩n");
        ts->stopped = true;
        mb();
        wake_up(&ts->wait);
        disable_irq(ts->irq);
#ifdef CONFIG_MANGO_TSC2007_IRQ_DISABLE//crazyboys 20160517
        err = tsc2007_xfer(ts, ADC_ON_12BIT);
#endif
static int tsc2007_open(struct input_dev *input_dev)
{
        struct tsc2007 *ts = input_get_drvdata(input_dev);
        int err;
        ts->stopped = false;
        mb();
        enable_irq(ts->irq);
#ifndef CONFIG_MANGO_TSC2007_IRQ_DISABLE //crazyboys 20200910
        /* Prepare for touch readings - power down ADC and enable PENIRQ */
        err = tsc2007_xfer(ts, PWRDOWN);
        if (err < 0) {
                 tsc2007_stop(ts);
                 return err;
        }
#endif
        return 0;
}
static int tsc2007_probe(struct i2c_client *client,
```

```
const struct i2c_device_id *id)
{
        const struct tsc2007_platform_data *pdata =
                 dev_get_platdata(&client->dev);
        struct tsc2007 *ts;
        struct input_dev *input_dev;
        int err;
        if (!i2c_check_functionality(client->adapter,
                                       I2C_FUNC_SMBUS_READ_WORD_DATA))
                 return -EIO;
        ts = devm_kzalloc(&client->dev, sizeof(struct tsc2007), GFP_KERNEL);
        if (!ts)
                 return -ENOMEM;
#ifndef CONFIG_MANGO_TSC2007_IRQ_DISABLE //crazyboys 20200910
        if (pdata)
                 err = tsc2007_probe_pdev(client, ts, pdata, id);
        else
                 err = tsc2007_probe_dt(client, ts);
        if (err)
                 return err;
#endif
ifndef CONFIG_MANGO_TSC2007_IRQ_DISABLE
        err = devm_request_threaded_irq(&client->dev, ts->irq,
                                           tsc2007_hard_irq, tsc2007_soft_irq,
                                           IRQF_ONESHOT,
                                           client->dev.driver->name, ts);
        if (err) {
                 dev_err(&client->dev, "Failed to request irq %d: %d₩n",
                         ts->irq, err);
                 return err;
        }
#endif
#ifndef CONFIG_MANGO_TSC2007_IRQ_DISABLE
```

```
/* power down the chip (TSC2007 SETUP does not ACK on I2C) */
        err = tsc2007 xfer(ts, PWRDOWN);
        if (err < 0) {
                 dev err(&client->dev,
                          "Failed to setup chip: %d₩n", err);
                                 /* chip does not respond */
                 return err:
        }
        err = input_register_device(input_dev);
        if (err) {
                 dev_err(&client->dev,
                          "Failed to register input device: %d₩n", err);
                 return err:
        }
        err = tsc2007_iio_configure(ts);
        if (err) {
                 dev err(&client->dev,
                          "Failed to register with IIO: %d₩n", err);
                 return err;
        }
#endif
```

#### 에러 발생

[CRZ] drivers/input/touchscreen/tsc2007\_core.c (342) tsc2007\_probe:imx6q-pinctrl 20e0000.iomuxc: pin MX6Q\_PAD\_SD1\_CMD already requested by 20e0000.iomuxc;cannot claim for 1-0038imx6q-pinctrl 20e0000.iomuxc: pin-210 (1-0038) status -22imx6q-pinctrl 20e0000.iomuxc: could not request pin 210 (MX6Q\_PAD\_SD1\_CMD) from groupmango\_ts on device 20e0000.iomuxcmango-ts 1-0038: Error applying setting, reverse things backmango-ts: probe of 1-0038 failed with error -22snvs\_rtc 20cc000.snvs:snvs-rtc-lp: rtc core: registered 20cc000.snvs:snvs-rtc-lp as rtc0MX6Q\_PAD\_SD1\_CMD 이중으로 define되어있다는 것이다.중복으로 선언 된 부분을 삭제하면 된다.커널 메시지에서 아래와 같이 나오면, 인식이 된 것이다.

[CRZ] drivers/input/touchscreen/mango\_ft5x06\_ts.c (1792) ft5x06\_i2c\_ts\_probe\_dt:

input: mango-ts as /devices/virtual/input/input1

CRZ ft5x0x\_ts\_probe reset timer start

-----[FTS] Firmware version = 0x61

-----[FTS] report rate is 0Hz.

-----[FTS] touch threshold is 500.

테스트 방법과 확인 방법은 아래와 같다.

# cat /proc/bus/input/devices
I: Bus=0018 Vendor=0000 Product=0000 Version=0000
N: Name="mango-ts"
P: Phys=
S: Sysfs=/devices/virtual/input/input1
U: Uniq=
H: Handlers=kbd event0
B: PROP=0
B: EV=b
B: KEY=400 0 4 0 0 0 0 c0000 0 0 0
B: ABS=2650000 1000000

# 4.2.5. 7인치 정전식 LCD 드라이버 포팅

커널 Configuration 한다.

CONFIG\_MANGO\_TOUCH\_FT5x06\_7INCH=y CONFIG\_MANGO\_TSC2007\_IRQ\_DISABLE=y

타이밍은 아래와 같이 세팅을 한다.

"MANGO-CAP7",	//name
60,	//refresh
1024,	//xres
600,	//yres
19531,	//pixclock(ns)
60,	//left_margin(HBP)
60,	//right_margin(BFP)
5,	//upper_margin(VBP)
5,	//lower_margin(VFP)
200,	//hsync_len
25,	//vsync_len
0, //sync	
FB_VMODE_NONINTERLA	CED, //vmode
0,}, //flag	

#### 커널 로그에서 터치가 인식이 되면 아래와 같이 나오면 정상이다.

[CRZ] drivers/input/touchscreen/tsc2007\_core.c (342) tsc2007\_probe: [CRZ] drivers/input/touchscreen/tsc2007 core.c (193) tsc2007 stop: [CRZ] drivers/input/touchscreen/mango ft5x06 ts.c (1828) ft5x0x ts probe: [FTS] ft5x0x ts probe, driver version is 3.0. [CRZ] drivers/input/touchscreen/mango\_ft5x06\_ts.c (1792) ft5x06\_i2c\_ts\_probe\_dt: [CRZ] drivers/input/touchscreen/mango ft5x06 ts.c (1865) ft5x0x ts probe: tsdata->reset pin=18 usb 1-1: new high-speed USB device number 2 using ci hdrc input: mango-ts as /devices/virtual/input/input1 CRZ ft5x0x ts probe reset timer start usb 1-1: New USB device found, idVendor=05e3, idProduct=0608 usb 1-1: New USB device strings: Mfr=0, Product=1, SerialNumber=0 usb 1-1: Product: USB2.0 Hub hub 1-1:1.0: USB hub found hub 1-1:1.0: 4 ports detected -----[FTS] Firmware version = 0x13 -----[FTS] report rate is 100Hz. -----[FTS] touch threshold is 200.

"fbset" 명령으로 확인 시 아래와 같다.

mode "1024x600-60" # D: 51.201 MHz, H: 38.096 kHz, V: 59.993 Hz geometry 1024 600 1024 600 32 timings 19531 60 60 5 5 200 25 accel false rgba 8/16,8/8,8/0,8/24 endmode

# cat /sys/class/graphics/fb0/mode
U:1024x600p-60

# 4.2.6. PMIC 없으므로 LDO로 변경하기

# 4.3. USB Host 확인

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 아래와 같이 정의가 되어 있으면 된다.

&usbh1 {

```
vbus-supply = <&reg_usb_h1_vbus>;
status = "okay";
```

};

확인 방법

USB Storage를 꽂아서 확인하면 된다.

usb 1-1.2: new high-speed USB device number 3 using ci_hdrc		
usb 1-1.2: New USB device found, idVendor=05e3, idProduct=0723		
usb 1-1.2: New USB device strings: Mfr=3, Product=4, SerialNumber=2		
usb 1-1.2: Product: USB Storage		
usb 1-1.2: Manufacturer: Generic		
usb 1-1.2: SerialNumber: 00000009451		
usb-storage 1-1.2:1.0: USB Mass Storage device detected		
usb-storage 1-1.2:1.0: Quirks match for vid 05e3 pid 0723: 8000		
scsi host0: usb-storage 1-1.2:1.0		
scsi 0:0:0:0: Direct-Access Generic STORAGE DEVICE 9451 PQ: 0 ANSI: 0		
sd 0:0:0:0: [sda] 15628288 512-byte logical blocks: (8.00 GB/7.45 GiB)		
sd 0:0:0:0: [sda] Write Protect is off		
sd 0:0:0:0: [sda] No Caching mode page found		
sd 0:0:0:0: [sda] Assuming drive cache: write through		
sda: sda1		

# 4.4. eMMC 모듈 테스트

Mango-IMX6Q에서는 USDHC3이 eMMC로 사용한다.



arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일에서 cd-gpios, wp-gpios 핀을 사용하지 않으므로 아래와 같이 수정한다.

&usdhc3 {
 pinctrl-names = "default";
 pinctrl-0 = <&pinctrl\_usdhc3>;
 bus-width = <8>;
 //cd-gpios = <&gpio2 0 GPIO\_ACTIVE\_LOW>;
 //wp-gpios = <&gpio2 1 GPIO\_ACTIVE\_HIGH>;
 keep-power-in-suspend;
 enable-sdio-wakeup;
 status = "okay";
};

eMM\_RESET\_N 핀은 SD3\_RST 핀에 연결되어 있다.

pinctrl_usdhc3: usdhc3grp {		
fsl,pins	5 = <	
	MX6QDL_PAD_SD3_CMDSD3_CMD	0x17059
	MX6QDL_PAD_SD3_CLKSD3_CLK	0x10059
	MX6QDL_PAD_SD3_DAT0_SD3_DATA0	0x17059
	MX6QDL_PAD_SD3_DAT1_SD3_DATA1	0x17059
	MX6QDL_PAD_SD3_DAT2_SD3_DATA2	0x17059
	MX6QDL_PAD_SD3_DAT3_SD3_DATA3	0x17059
	MX6QDL_PAD_SD3_DAT4_SD3_DATA4	0x17059
	MX6QDL_PAD_SD3_DAT5SD3_DATA5	0x17059
	MX6QDL_PAD_SD3_DAT6_SD3_DATA6	0x17059

www.mangoboard.com cafe.naver.com/embeddedcrazyboys CRZ Technology 45

MX6QDL_PAD_SD3_DAT7SD3_DA	TA7 0x17059
MX6QDL_PAD_SD3_RSTSD3_RES	T 0x17059
//crazyboys 20200910	
>;	
};	
인식이 되는지 확인 한다.	
eMMC 모듈 장착부분은 하드웨어 매뉴얼을 참조한다.	
mmc1: SDHCI controller on 2194000.usdhc [2194000.usdhc] using AD	MA
mmc2: SDHCI controller on 2198000.usdhc [2198000.usdhc] using	ADMA
sdhci-esdhc-imx 219c000.usdhc: Got CD GPIO	
mmc3: SDHCI controller on 219c000.usdhc [219c000.usdhc] using AD	MA
mmc2: new DDR MMC card at address 0001	
mxc_vpu 2040000.vpu_fsl: VPU initialized	
mxc_hdmi_cec soc:hdmi_cec@00120000: HDMI CEC initialized	
mmcblk2: mmc2:0001 8GND3R 7.28 GiB	
Galcore version 6.2.4.190076	
mmcblk2boot0: mmc2:0001 8GND3R partition 1 4.00 MiB	
mmcblk2boot1: mmc2:0001 8GND3R partition 2 4.00 MiB	
mmcblk2rpmb: mmc2:0001 8GND3R partition 3 512 KiB	
mmcblk2: p1 p2 p3 < p5 p6 p7 p8 p9 > p4	

# 4.5. SDHC4 포팅



SD4\_CDN 핀이 있다. GPIO2\_IO12 사용한다.

SD4_DAT4	ALT1	SD4_DATA4
	ALT2	UART2_RX_DATA
	ALT5	GPIO2_IO12

Micro SD 카드 Slot에 Micro SD 카드를 삽입하여 테스트 한다.

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일 수정

pinctrl\_usdhc4: usdhc4grp {

tsl,pins = <		
	MX6QDL_PAD_SD4_CMDSD4_CMD	0x17059
	MX6QDL_PAD_SD4_CLKSD4_CLK	0x10059
	MX6QDL_PAD_SD4_DAT0_SD4_DATA0	0x17059
	MX6QDL_PAD_SD4_DAT1SD4_DATA1	0x17059
	MX6QDL_PAD_SD4_DAT2SD4_DATA2	0x17059
	MX6QDL_PAD_SD4_DAT3SD4_DATA3	0x17059
	MX6QDL_PAD_SD4_DAT4GPIO2_IO12	0x13069 /*
crazyboys 20200910 cd pin */		
#if 0 //crazyboys 20200910		
	MX6QDL_PAD_SD4_DAT4SD4_DATA4	0x17059
	MX6QDL_PAD_SD4_DAT5SD4_DATA5	0x17059
	MX6QDL_PAD_SD4_DAT6SD4_DATA6	0x17059
	MX6QDL_PAD_SD4_DAT7SD4_DATA7	0x17059
#endif		

&usdhc4 {
 pinctrl-names = "default";
 pinctrl-0 = <&pinctrl\_usdhc4>;
 bus-width = <4>;
 cd-gpios = <&gpio2 12 GPIO\_ACTIVE\_LOW>;
 no-1-8-v;

```
keep-power-in-suspend;
status = "okay";
```

};

# 4.6. Watchdog 포팅

arch/arm/boot/dts/imx6qdl-sabresd.dtsi 파일 수정

```
&gpc {
    fsl,ldo-bypass = <0>;/* CRZ_icanjji crazyboys 20161201 */
    fsl,wdog-reset = <1>; /* crazyboys 20160225 watchdog select of reset source */
    pu-supply = <&reg_pu>; /* ldo-bypass:use pu_dummy if VDDSOC share with VDDPU */
};

&wdog1 {
    status = "okay";
};

&wdog2 {
    pinctrl-names = "default";
    pinctrl-0 = <&pinctrl_wdog>;
    fsl,ext-reset-output;
    status = "disabled";
};
```

Watchdog 테스트 방법

cat > /dev/watchdog	
Hit return once	
Wait 60 seconds	
Wachdog reset.	

# 4.7. USB Device 테스트

# 4.8. 이미지 툴 다운로드 해 보자.

우선 안드로이드 툴이 필요하다.

https://www.nxp.com/webapp/sps/download/license.jsp?colCode=P9.0.0\_1.0.0\_GA\_TOOL&appType=file 1&DOWNLOAD\_ID=null

android\_p9.0.0\_1.0.0-ga\_tools.tar.gz 파일을 다운로드 받는다.

압축을 해제하면 어떻게 사용하는지 확인 해 보면, 리눅스 PC에서 사용해야하는 shell 형태로 되어 있다. 좀 더 살펴 보자. 이전에는 mfgtools를 사용했는데 변경 되었다.

#### 4.8.1. 스크립트로 이용하여 이미지 Write하기

안드로이드 빌드 완료 후 Micro SD 카드를 Linux PC에 삽입한다.

#### \$ cd \${ANDROID\_BUILD}

\$ sudo ./device/fsl/common/tools/fsl-sdcard-partition.sh -f imx6q -D out/target/product/sabresd\_6dq/ /dev/sdb

이미지가 Write가 된다.

#### 4.8.2. 소소제목