

Mango-IMX6Q Marvell8787

Wifi 드라이버 포팅 가이드

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

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

Crazy Embedded Laboratory

Document History

Revision	Date	Change note
Init	2016-06-22	전종인

1.	Marvell8787 wifi 모듈 펌웨어 다운로드 하기.....	4
2.	Wifi 인식 시키기	5
3.	커널에서 드라이버 수정.....	8
3.1.	파일 시스템에 wifi 드라이버 포함하기	9
4.	Wifi 구동하기	10
5.	문제점 수정	13
5.1.	Failed to get firmware mrvl/sd8787_uapsta.bin.....	13
5.2.	Calling CRDA to update world regulatory domain 문제	13

1. Marvell8787 wifi 모듈 펌웨어 다운로드 하기

펌웨어 다운로드 하기

https://kernel.googlesource.com/pub/scm/linux/kernel/git/firmware/linux-firmware.git/+/_/dca884016afa9f954baa69e3e28b8f2aab3b6921/mrvl/

[kernel](#) / [pub](#) / [scm](#) / [linux](#) / [kernel](#) / [git](#) / [firmware](#) / [linux-firmware.git](#) / [dca884016afa9f954baa69e3e28b8f2aab3b6921](#) / [.](#) / [mrvl](#)

tree: 8c42c5d21e0f54d123841e9ca259e1300badeba3 [\[path history\]](#) [\[tgz\]](#)

- [pcie8897_uapsta.bin](#)
- [sd8688.bin](#)
- [sd8688_helper.bin](#)
- [sd8787_uapsta.bin](#)
- [sd8797_uapsta.bin](#)
- [sd8801_uapsta.bin](#)
- [sd8887_uapsta.bin](#)
- [sd8897_uapsta.bin](#)
- [usb8766_uapsta.bin](#)
- [usb8797_uapsta.bin](#)
- [usb8801_uapsta.bin](#)
- [usb8897_uapsta.bin](#)

받은 후 압축을 풉니다.

sd8787_uapsta.bin 파일을

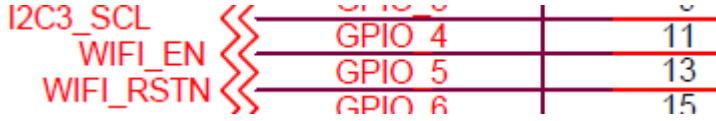
커널 소스에 firmware/mrvl 디렉토리에 복사를 합니다.

```
$mkdir mrvl
```

```
$ ls firmware/mrvl  
sd8787_uapsta.bin
```

2. Wifi 인식 시키기

회로도를 보면 아래와 같습니다.



WIFI_EN <-> GPIO4 ,

WIFI_RSTN <-> GPIO_5

연결이 되어 있습니다.

http://cache.nxp.com/files/32bit/doc/ref_manual/IMX6DQRM.pdf?fpsp=1&WT_TYPE=Reference%20Manuals&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation&fileExt=.pdf

datasheet를 참조하면

GPIO_5	ALT0	ESAI_TX2_RX3	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO05
	ALT2	KEY_ROW7	PUS - 100K_OHM_PU	
	ALT3	CCM_CLKO1	PUE - PULL	
	ALT5	GPIO1_IO05	PKE - ENABLED	
	ALT6	I2C3_SCL	ODE - DISABLED	

Table continues on the next page...

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

Freescale Semiconductor, Inc.

273

Overview

Table 4-1. Pin Assignments (continued)

Pad Name	Mode	Signal	Pad Settings	Pad/Group Registers
	ALT7	ARM_EVENTI	SPEED - MEDIUM DSE - 40_OHM SRE - SLOW	

WIFI_RSTN은 GPIO1_5 GPIO로 설정

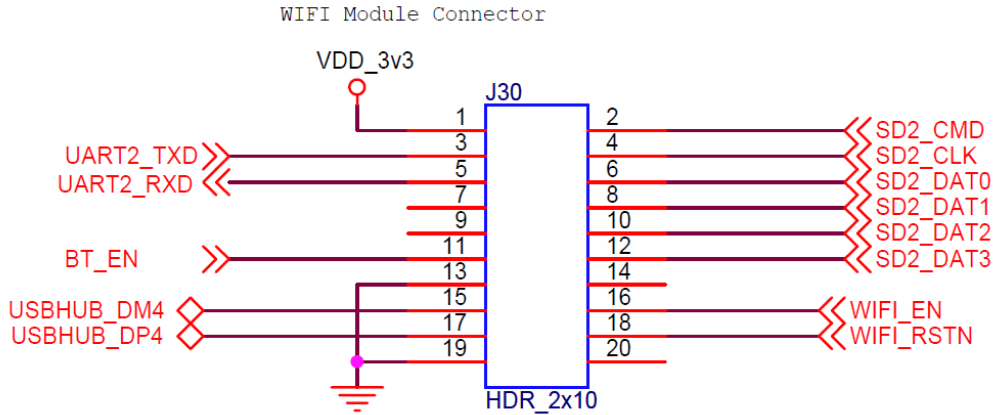
GPIO_4	ALT0	ESAI_TX_HF_CLK	HYS - ENABLED	SW_PAD_CTL_PAD_GPIO04
	ALT2	KEY_COL7	PUS - 100K_OHM_PU	
	ALT5	GPIO1_IO04	PUE - PULL	
	ALT6	SD2_CD_B	PKE - ENABLED	
			ODE - DISABLED	
			SPEED - MEDIUM	
			DSE - 40_OHM	
			SRE - SLOW	

WIFI_EN <-> GPIO4<->GPIO1_4 GPIO로 설정

"arch/arm/boot/dts/imx6qdl-sabresd.dtsi" 에 추가

```
mango_wifi_reset: mango-wifi-reset {
    compatible = "gpio-reset";
    reset-gpios = <&gpio1 5 GPIO_ACTIVE_LOW>;
    reset-delay-us = <100000>;
    initially-in-reset;
    #reset-cells = <0>;
};
```

```
&usdhc2 {
    pinctrl-names = "default";
    pinctrl-0 = <&pinctrl_usdhc2_1>;
    #if 0
    cd-gpios = <&gpio2 2 0>;
    wp-gpios = <&gpio2 3 0>;
    #endif
    resets = <&mango_wifi_reset>;
    enable-gpio = <&gpio1 4 0>;
    no-1-8-v;
    keep-power-in-suspend;
    enable-sdio-wakeup;
    status = "okay";
};
```



SDIO 2번 채널에 WiFi 모듈을 연결을 할 수 있도록 회로가 되어 있습니다.

"drivers/mmc/host/sdhci-esdhc-imx.c" 파일에 추가

```
sdhci_esdhc_imx_probe_dt(struct platform_device *pdev,
                        struct esdhc_platform_data *boarddata) 함수에 내용 추가

device_reset(&pdev->dev);

enable_gpio = of_get_named_gpio(np, "enable-gpio", 0);
if (gpio_is_valid(enable_gpio))
{
    status = devm_gpio_request_one(&pdev->dev,enable_gpio,
                                GPIOF_OUT_INIT_HIGH,NULL);

    if(status < 0 )
    {
        dev_err(&pdev->dev, "Wifi EN Request Fail status %d\n"
                ,status);

        return status;
    }
}

return 0;
```

커널 컴파일 후 커널 로그에서 아래와 같이 디버깅 메시지가 나오면, 인식이 된 것입니다.

```
mmc2: SDHCI controller on 2198000.usdhc [2198000.usdhc] using ADMA
mmc1: new high speed SDIO card at address 0001
```

3. 커널에서 드라이버 수정

```
CONFIG_MWIFIEX_SDIO=m
CONFIG_MWIFIEX=y
CONFIG_WIRELESS_EXT=y
CONFIG_WEXT_CORE=y
CONFIG_WEXT_PROC=y
CONFIG_WEXT_SPY=y
CONFIG_WEXT_PRIV=y
CONFIG_CFG80211=y
CONFIG_CFG80211_WEXT=y
CONFIG_LIB80211=y
```

컴파일을 하면 커널 object 파일이 만들어 집니다.

모듈로 컴파일 방법

```
$ vi build_kernel
수정
all*)
    echo make -j$CPU_JOB_NUM uImage
    make -j$CPU_JOB_NUM uImage
    make -j$CPU_JOB_NUM modules
```

컴파일 결과

```
$ ls drivers/net/wireless/mwifiex/
11n_aggr.c      cfg80211.o  ie.o      mwifiex_sdio.ko
```

sd8787_uapsta.bin 파일 커널에 포함해서 컴파일 하기

```
Generic Driver Options->

(mrvl/sd8787_uapsta.bin) External firmware blobs to build into the kernel binary      x x
  x x                (firmware) Firmware blobs root directory (NEW)
```

```
CONFIG_EXTRA_FIRMWARE="sd8787_uapsta.bin"
```

```
CONFIG_EXTRA_FIRMWARE_DIR="firmware"
```

컴파일 결과 아래와 같이 커널 이미지에 포함이 된 것을 볼 수 있습니다.


```
$ ls firmware/sd8787_uapsta.bin*
```

```
firmware/sd8787_uapsta.bin  firmware/sd8787_uapsta.bin.gen.S  firmware/sd8787_uapsta.bin.gen.o
```

```
arch/arm/boot/dts/imx6qdl.dtsi
```

```
usdhc2 {
    pinctrl_usdhc2_1: usdhc2grp-1 {
        fsl,pins = <
            MX6QDL_PAD_SD2_CMD_SD2_CMD    0x17059
            MX6QDL_PAD_SD2_CLK_SD2_CLK    0x10059
            MX6QDL_PAD_SD2_DAT0_SD2_DATA0 0x17059
            MX6QDL_PAD_SD2_DAT1_SD2_DATA1 0x17059
            MX6QDL_PAD_SD2_DAT2_SD2_DATA2 0x17059
            MX6QDL_PAD_SD2_DAT3_SD2_DATA3 0x17059
            MX6QDL_PAD_NANDF_D4_SD2_DATA4 0x17059
            MX6QDL_PAD_NANDF_D5_SD2_DATA5 0x17059
            MX6QDL_PAD_NANDF_D6_SD2_DATA6 0x17059
            MX6QDL_PAD_NANDF_D7_SD2_DATA7 0x17059
        >;
    };

    pinctrl_usdhc2_2: usdhc2grp-2 {
        fsl,pins = <
            MX6QDL_PAD_SD2_CMD_SD2_CMD    0x17059
            MX6QDL_PAD_SD2_CLK_SD2_CLK    0x10059
            MX6QDL_PAD_SD2_DAT0_SD2_DATA0 0x17059
            MX6QDL_PAD_SD2_DAT1_SD2_DATA1 0x17059
            MX6QDL_PAD_SD2_DAT2_SD2_DATA2 0x17059
            MX6QDL_PAD_SD2_DAT3_SD2_DATA3 0x17059
        >;
    };
};
```

만들어진 파일을 파일 시스템에 복사를 합니다.

3.1. 파일 시스템에 wifi 드라이버 포함하기

KO 파일을 image 폴더에 복사를 합니다.

```
$ cp drivers/net/wireless/mwifiex/mwifiex_sdio.ko ../image
```

Micro SD 카드를 리눅스 PC에 삽입을 합니다.

```
$ mkdir rootfs
$ dmesg | tail
[13882821.951030] sdg: sdg1 sdg2 sdg3 sdg4
[13882828.847001] EXT4-fs (sdg2): mounted filesystem with ordered data mode. Opts: (null)
[13882860.785736] EXT4-fs (sdg3): mounted filesystem with ordered data mode. Opts: (null)
[13882866.357704] sdg: detected capacity change from 8068792320 to 0
[13903180.931517] sd 228:0:0:0: [sdg] 15628288 512-byte logical blocks: (8.00 GB/7.45 GiB)
[13903180.933074] sd 228:0:0:0: [sdg] No Caching mode page present
[13903180.933077] sd 228:0:0:0: [sdg] Assuming drive cache: write through
[13903180.935187] sd 228:0:0:0: [sdg] No Caching mode page present
[13903180.935190] sd 228:0:0:0: [sdg] Assuming drive cache: write through
[13903180.935799] sdg: sdg1 sdg2

$ sudo mount /dev/sdg2 rootfs
$ sudo cp mwifiex_sdio.ko rootfs/home/root/
$ sudo umount /dev/sdg2
```

mwifiex_sdio.ko 파일을 root 디렉토리에 복사를 합니다.

Micro SD 카드를 보드에 삽입하고 부팅을 합니다.

4. Wifi 구동하기

부팅 후

```
imx6qsabresd login: root
root@imx6qsabresd:~#
```

```
root@imx6qsabresd:~# insmod mwifiex_sdio.ko
root@imx6qsabresd:~# mwifiex_sdio mmc1:0001:1: WLAN FW is active
mwifiex_sdio mmc1:0001:1: ignoring F/W country code US
mwifiex_sdio mmc1:0001:1: driver_version = mwifiex 1.0 (14.66.35.p52)
IPv6: ADDRCONF(NETDEV_CHANGE): mlan0: link becomes ready
```

```
root@imx6qsabresd:~# ifconfig -a
```

```

eth0    Link encap:Ethernet  HWaddr 7E:6A:A1:34:C2:A1
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING  MTU:65536  Metric:1
        RX packets:4 errors:0 dropped:0 overruns:0 frame:0
        TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:280 (280.0 B)  TX bytes:280 (280.0 B)

mlan0   Link encap:Ethernet  HWaddr AC:3F:A4:4F:9B:78
        inet6 addr: fe80::ae3f:a4ff:fe4f:9b78/64 Scope:Link
        UP BROADCAST MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:6 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

```

```

root@imx6qsabresd:~# iwlist mlan0 scan | grep CRZ
        ESSID:"CRZ_icanjji"
        ESSID:"CRZ_ybkim"
        ESSID:"CRZ-TOM"

```

```

root@imx6qsabresd:~# ifconfig mlan0 up
root@imx6qsabresd:~# iwconfig mlan0 essid CRZ_icanjji
root@imx6qsabresd:~# iwconfig
lo      no wireless extensions.

mlan0   IEEE 802.11bgn  ESSID:"CRZ_icanjji"
        Mode:Managed  Frequency:2.412 GHz  Access Point: 00:26:66:1A:55:AC

```

```
Bit Rate=150 Mb/s
Retry long limit:7   RTS thr:off   Fragment thr:off
Encryption key:off
Power Management:on
Link Quality=33/70   Signal level=-77 dBm
Rx invalid nwid:0   Rx invalid crypt:0   Rx invalid frag:0
Tx excessive retries:0   Invalid misc:0   Missed beacon:0
```

```
eth0    no wireless extensions.
```

```
tunl0    no wireless extensions.
```

```
root@imx6qsabresd:~# udhcpc -imlan0
udhcpc (v1.22.1) started
Sending discover...
Sending select for 192.168.100.2...
Lease of 192.168.100.2 obtained, lease time 7200
/etc/udhcpc.d/50default: Adding DNS 168.126.63.1
/etc/udhcpc.d/50default: Adding DNS 168.126.63.2
```

Iperf 테스트

```
root@imx6qsabresd:~# iperf -c 192.168.100.12 -i 1 -t 10
-----
Client connecting to 192.168.100.12, TCP port 5001
TCP window size: 20.7 KByte (default)
-----
[ 3] local 192.168.100.2 port 56055 connected with 192.168.100.12 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3] 0.0- 1.0 sec    384 KBytes    3.15 Mbits/sec
[ 3] 1.0- 2.0 sec    1.25 MBytes   10.5 Mbits/sec
[ 3] 2.0- 3.0 sec    1.25 MBytes   10.5 Mbits/sec
[ 3] 3.0- 4.0 sec    1.38 MBytes   11.5 Mbits/sec
[ 3] 4.0- 5.0 sec    640 KBytes    5.24 Mbits/sec
[ 3] 5.0- 6.0 sec    512 KBytes    4.19 Mbits/sec
[ 3] 6.0- 7.0 sec    512 KBytes    4.19 Mbits/sec
[ 3] 7.0- 8.0 sec    384 KBytes    3.15 Mbits/sec
[ 3] 8.0- 9.0 sec    768 KBytes    6.29 Mbits/sec
```

```
[ 3] 9.0-10.0 sec  1.25 MBytes  10.5 Mbits/sec
[ 3] 0.0-10.1 sec  8.38 MBytes  6.96 Mbits/sec
```

5. 문제점 수정

5.1. Failed to get firmware mrvl/sd8787_uapsta.bin

아래와 같이 에러 발생 시

```
root@imx6qsabresd:~# insmod mwifiex_sdio.ko
mwifiex_sdio mmc1:0001:1: Failed to get firmware mrvl/sd8787_uapsta.bin
```

```
$ cd firmware
```

```
$ mkdir mrvl
```

```
$ cp sd8787_uapsta.bin mrvl/
```

하고 다시 커널을 컴파일 한다.

5.2. Calling CRDA to update world regulatory domain 문제

```
root@imx6qsabresd:~# cfg80211: Calling CRDA for country: KR
cfg80211: Calling CRDA to update world regulatory domain
```

```
CONFIG_LIB80211=y
```