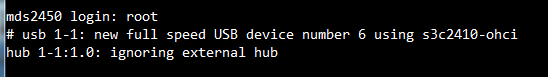
USB 허브를 이용해서 와이파이모듈과 카메라를 사용하고 싶다

우리가 사용하고싶은 USb허브, 와이파이모듈, 카메라 관련 소스는 이미다 있다.

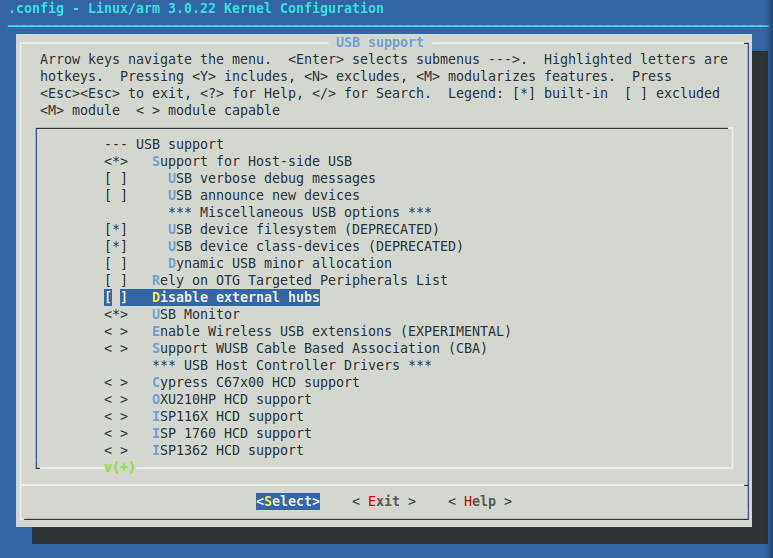
그래서 우선 이 소스들을 이용하여 구현을 먼저하고

해당 기능을 커널에서 제거한후 관련 소스를 고쳐서 우리만의 모듈을 만든 후 실행해보자

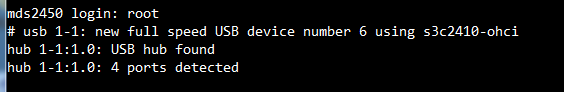
1. USB허브 인식
   1. 폰으로 찍은 보드와 허브연결한 사진
   2. 인식안됨. 커널의 menuconfig 수정요망



* 1. 커널폴더에서 make menuconfig
     1. Device Drivers ->USB support -> Disable external hubs 체크 해제



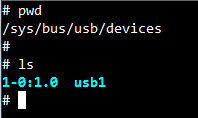
* 1. root@ip6:/work/linux/finished/linux-3.0.22-done# make (커널 이미지 재생성)
  2. root@ip6:/work/linux/finished/linux-3.0.22-done# cp arch/arm/boot/zImage /tftpboot(tftp로복사)
  3. 보드에서 임시테스트
     1. MDS2450# tftp c0008000 zImage
     2. MDS2450# bootm c0008000



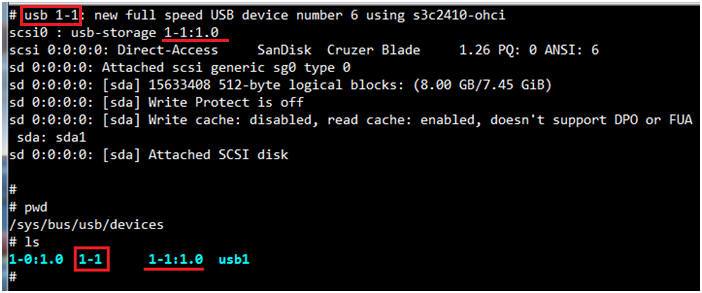
* 1. nand에 수정한 커널 이미지올리기
     1. MDS2450# tftpboot 32000000 zImage(재부팅 한 경우 다시 받기)
     2. MDS2450# nand erase 00080000 00500000
     3. MDS2450# nand write 32000000 00080000 00500000
     4. MDS2450# save
     5. MDS2450# boot

보드에 USB장치(Sandisk USB메모리) 삽입했을경우 USB장치의 정보 보기

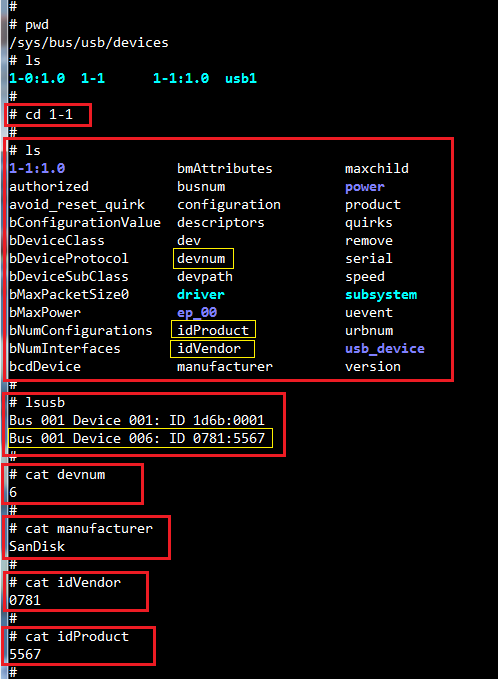
보드에 USB장치 삽입 안했을경우



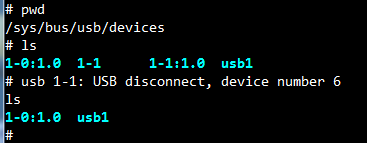
보드에 USB장치(Sandisk USB메모리) 삽입했을경우



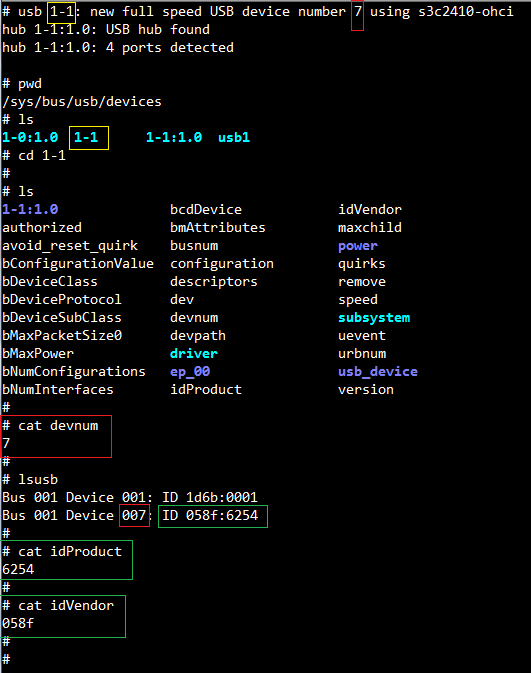
삽입된 USB 장치에 대한 정보



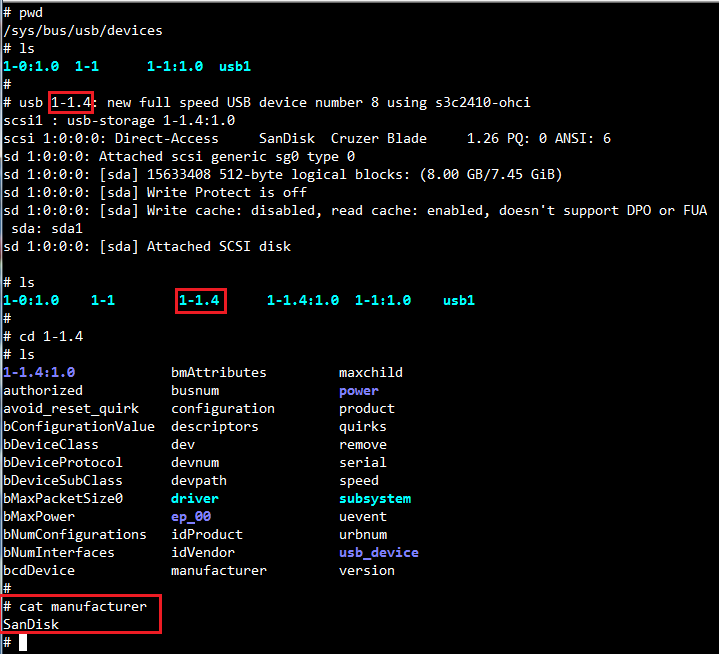
장치 제거했을경우



USB 허브 장착했을 경우

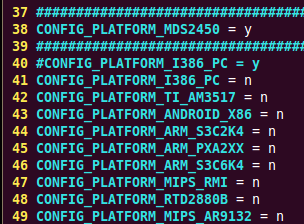


USB허브 삽입된 상태에서 USB디스크를 허브에 삽입



iptime에서 받은 u100mini 디바이스 드라이버 makefile수정

root@ip6:/work/drive/RTL8188C\_8192C\_8192D\_USB\_linux\_v3.3.2\_3192.20120103/driver/rtl8188C\_8192C\_8192D\_usb\_linux\_v3.3.2\_3192.20120103# vi Makefile





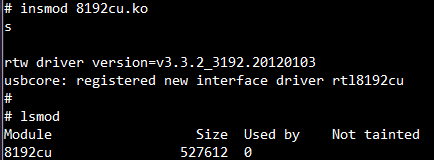
36번째 line에 CONFIG\_PLATFORM\_MDS2450추가

232번째 line에 platform추가하기

make 🡺 ko file생성

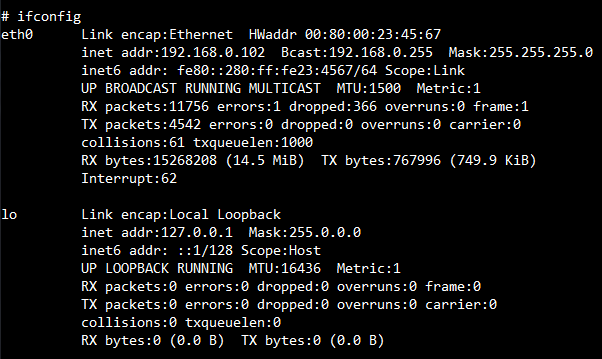
cp 8192cu.ko /nfsroot/root/

MDS2450# insmod 8192cu.ko



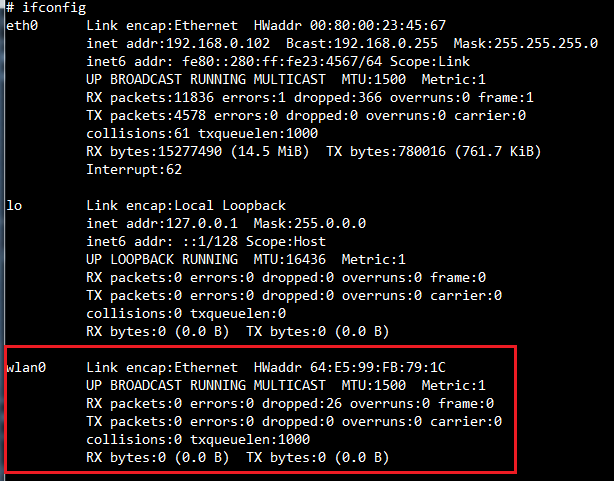
그후 와이파이 동글을 usb포트에 삽입하면 시스템 메시지가 쭉 출력된다.

이상태에서 ifconfig를 하면 네트워크가 잡혀있지 않다.



MDS2450# ifconfig wlan0 up

: Where wlan0 is the name of your wireless device (this is most often the default). The above command will bring your wireless device up so it is ready to use



접속할 수 있는 와이파이 검색 : iwlist wlan0 scan

: This command is to scan for any wireless access points with the command:

Ssid 잡기 : iwconfig wlan0 essid "EmbeddesSW"

이제 IP잡아야하는데 dhcpcd명령어가없다.. busybox 업데이트해보자

경로 = /work/rootfs/busybox-1.23.2 (폴더없으면 압축풀기)

Make menuconfig

Networking Utilities -> Support for external dhcp clients(NEW) 체크

Makefile 수정

CROSS\_COMPILE ?= 이부분에 arm-none-linux-gnueabi- 추가

Make

안됨

다시 busybox menuconfig -> Networking Utilities -> udhcp client for DHCPv6(udhcpc6)

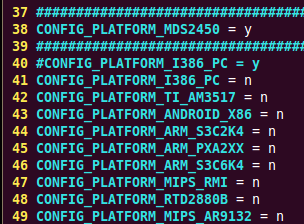
* Select IP address based on client MAC
* Enable ‘-P port’ option for udhcpd and udhcpc

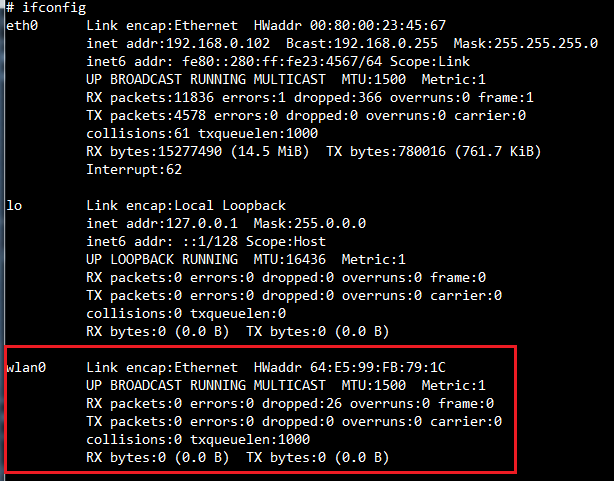
IP주소를 못잡음…..

처음부터 다시시작.

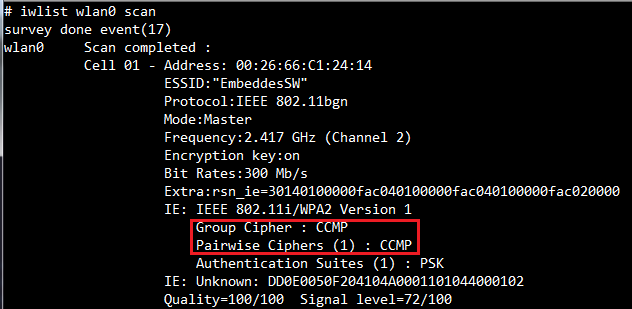
**와이파이 모듈 살리기**

**(USB허브 인식 포함)**

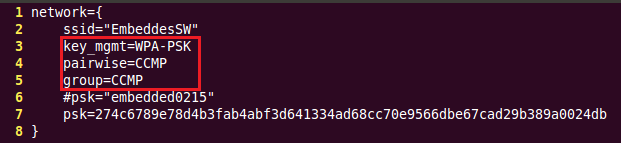
1. 드라이버가 없어서 와이파이 모듈홈페이지(iptime) 에서 리눅스용 디바이스드라이버 소스 다운로드
2. Makefile에 우리가 사용하는 플랫폼에 대한 정보가 없어서 추가.(경로설정)
   1. 
   2. 
3. Make 🡺 8192cu.ko 모듈생성
4. MDS2450# insmod 8192cu.ko
5. MDS2450# ifconfig wlan0 up (무선랜 장치 활성화)



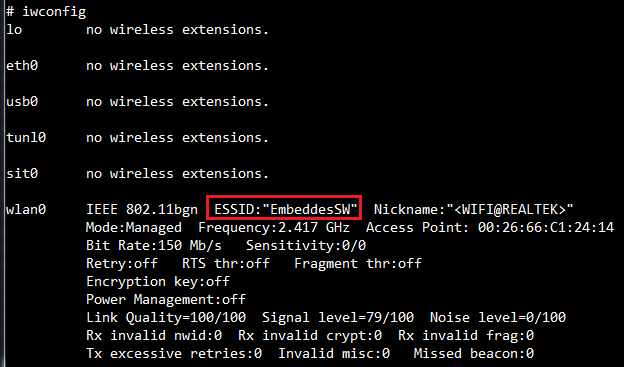
1. MDS2450# iwlist wlan0 scanning (연결 가능한 무선공유기 찾기)



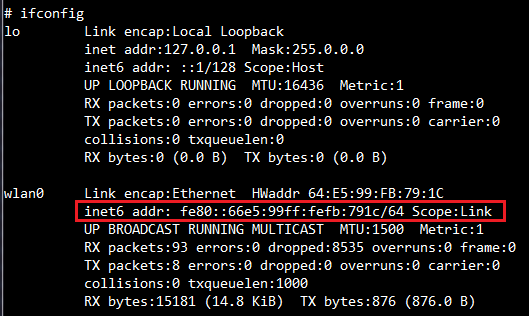
1. Wpa-supplicant명령을 사용하려면 wpa\_supplicant.conf가 필요하다. 이를 위해 우선 psk키를 생성한다.
   1. 그런데 psk키를 생성하는 명령어는 wpa\_passphrase 라는 명령어인데 이명령어는 우리가 사용한 busyBox에 없다. 그래서 서버 ubuntu에서 이 명령을 수행해서 wpa\_supplicant.conf를 생성한다.
   2. $ wpa\_passphrase ESSID PASSWD > EmbeddesSW.conf (파일명이 중요하진 않다.)
   3. 생성된 EmbeddesSW.conf 파일에서 추가 정보 입력



1. 유선 랜장치를 중지
   1. ifconfig eth0 down
2. ESSID를 받은 네트워크에 접속
   1. iwconfig wlan0 essid “essid”
   2. wpa\_supplicant -iwlan0 -c EmbeddesSW.conf &
   3. iwconfig명령으로 할당이 됬는지 확인.



* 1. ifconfig 명령으로 할당이 됬는지 확인. 아직 IP주소는 잡히지 않았다.



1. IP주소 할당(고정
   1. MDS2450# ifconfig wlan0 192.168.0.88 netmask 255.255.255.0 broadcast 192.168.0.255
   2. MDS2450# route add default gw 192.168.0.1 wlan0
   3. MDS2450# echo nameserver 192.168.0.1 > /etc/resolv.conf
2. 테스트
   1. MDS2450# ping [www.google.com](http://www.google.com)

**와이파이 모듈 살리는 과정에서 가장어려웠던점**

1. **공유기까지 접속후 IP주소를 할당해도 ping test가 되지 않았다.**

개발과정에서 “8.유선랜 장치 중지” 라는게 있다. 저희는 개발환경이 NFS였기에 랜선사용이 필수였다. 그래서 유선랜 장치 중지를 안하고 test를 했었다. 여기서 하루이상을 소비했다. 아무리 해도 안되서 혹시나 해서 NFS가 아닌 램디스크를 사용해 보고자 했다. 그런데 컴파일한 공유기 모듈의 용량이 7M를 넘었다. 그래서인지 커널을 부팅하는데 커널패닉이 일어났다. 다운받는 램의 주소도 바꿔보았지만 해결이 안되서 YAFFS2를 사용하기로 했다. YAFFS2로 설정하고 랜선을 뽑고 위의 과정을 동일하게 수행했고(ifconfig eth0 down 수행X) 마침내 성공.

Ping test를 하는데 유선랜과 무선이 살아있으면 우선적으로 유선랜포트로 ping이 가는 것을 알수 있었다.

