

운영체제 및 실습
- HW09 -
라즈베리파이 크로스컴파일 테스트

제출일자	2017.05.31
분 반	03
학번/이름	201203409/임동현
	201302366/김규태
	2013024223/신종욱

```
u201302423@u201302423: ~  
u201302423@u201302423:~$ uname -a  
Linux u201302423 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux  
u201302423@u201302423:~$
```

리눅스 버전 환경이다.

모듈컴파일을 위한 Makefile, C파일 작성

```
u201302423@u201302423: ~/rpi/test_module  
GNU nano 2.5.3 File: Makefile  
  
obj-m := test_module.o  
KDIR := /home/u201302423/rpi/linux  
  
default:  
    make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- -C $(KDIR) M=$(shell pwd) modules  
clean:  
    make -C $(KDIR) M=$(shell pwd) clean  
  
[ Read 8 lines ]  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos  
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line
```

해당 메이크는 크로스 컴파일 환경에서 작동하기위해 작성된 것으로 보인다.

```
u201302423@u201302423: ~/rpi/test_module  
GNU nano 2.5.3 File: test_module.c  
  
#include <linux/init.h>  
#include <linux/module.h>  
#include <linux/kernel.h>  
  
MODULE_LICENSE("GPL");  
  
static int mod_init(void)  
{  
    printk( KERN_INFO "Hello rpi world!\n");  
    return 0;  
}  
  
static void mod_exit(void)  
{  
    printk( KERN_INFO "Bye rpi!\n");  
}  
  
module_init(mod_init);  
module_exit(mod_exit);  
  
[ Read 20 lines ]  
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos  
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^_ Go To Line
```

새 세션 등록 정보

범주(C):

- 연결
 - 사용자 인증
 - 로그인 프롬프트
 - 로그인 스크립트
 - SSH
 - 보안
 - 터널링
 - SFTP
 - TELNET
 - RLOGIN
 - SERIAL
 - 프록시
 - 연결 유지
 - 터미널
 - 키보드
 - VT 모드
 - 고급
 - 모양
 - 창
 - 고급
 - 추적
 - 로깅
 - 파일 전송
 - X/YMODEM
 - ZMODEM

연결

일반

이름(N): 새 세션

프로토콜(P): SSH

호스트(H): 192.168.1.3

포트 번호(O): 22

설명(D):

다시 연결

☐ 예기치 않게 연결이 끊겼을 때 자동으로 다시 연결(A)

간격(Y): 0 초 제한(L): 0 분

TCP 옵션

☐ 네이글 알고리즘을 사용(U)

확인 취소

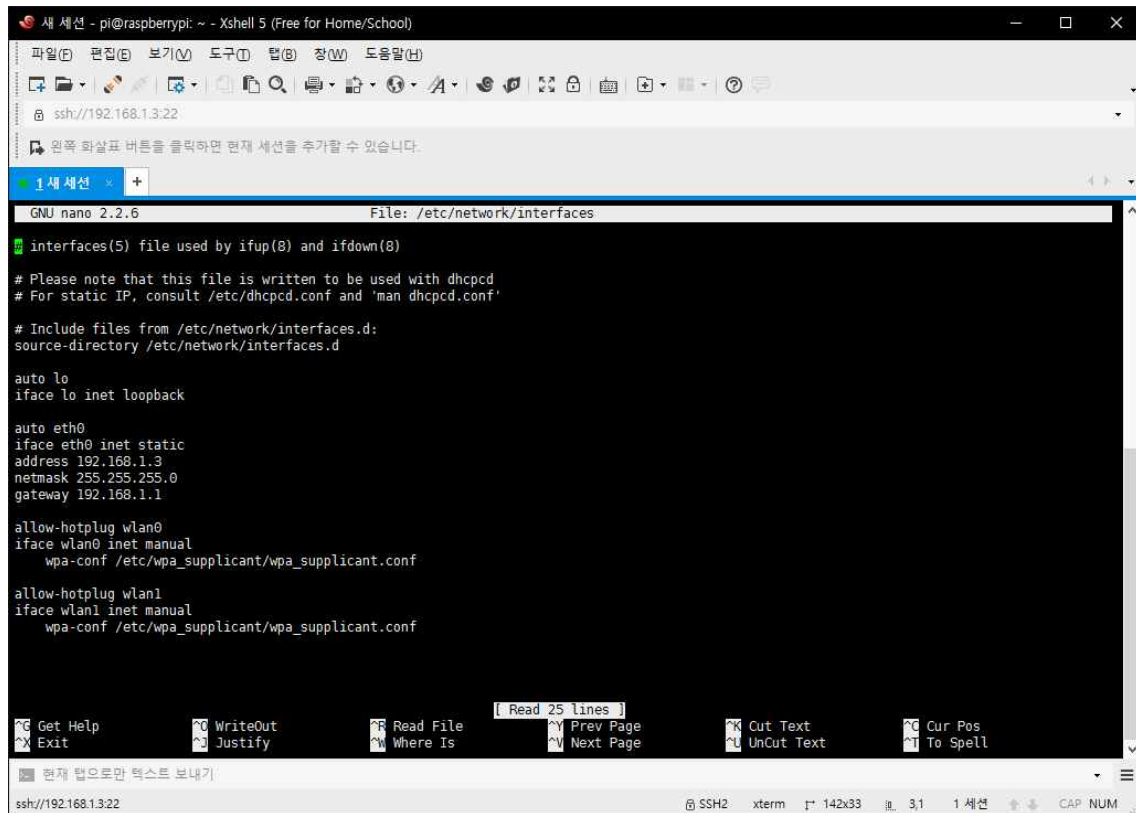
Xshell를 이용하여서 라즈베리파이와 연결

```

새 세션 - pi@raspberrypi: ~ - Xshell 5 (Free for Home/School)
파일(F) 편집(E) 보기(V) 도구(D) 탭(T) 창(W) 도움말(H)
ssh://192.168.1.3:22
원격 화살표 버튼을 클릭하면 현재 세션을 추가할 수 있습니다.
1 새 세션
pi@raspberrypi:~ $ sudo modprobe config
pi@raspberrypi:~ $ sudo zcat /proc/config.gz > config
pi@raspberrypi:~ $ ls
config Desktop Documents Downloads Music Pictures Public python_games Templates test_module.ko Videos
pi@raspberrypi:~ $ uname -a
Linux raspberrypi 4.4.50-v7+ #970 SMP Mon Feb 20 19:18:29 GMT 2017 armv7l GNU/Linux
pi@raspberrypi:~ $
  
```

config파일을 만들어주는 이유는 현재 라즈베리파이의 설정을 저장한 정보를 리눅스로 넘겨주기위함이다 현재버전은 4.4.50버전

현재 라즈베리파이 아이피 설정 확인



The screenshot shows a terminal window titled "새 세션 - pi@raspberrypi: ~ - Xshell 5 (Free for Home/School)". The terminal is running the nano text editor on the file `/etc/network/interfaces`. The file content is as follows:

```
interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.3
netmask 255.255.255.0
gateway 192.168.1.1

allow-hotplug wlan0
iface wlan0 inet manual
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

allow-hotplug wlan1
iface wlan1 inet manual
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

The terminal window includes a menu bar with options like "파일(F)", "편집(E)", "보기(V)", "도구(T)", "탭(T)", "창(W)", and "도움말(H)". At the bottom, there is a status bar showing "ssh://192.168.1.3:22" and window details like "SSH2", "xterm", "142x33", "3,1", "1 세션", and "CAP NUM".

```
u201302423@u201302423: ~/rpi/linux
u201302423@u201302423:~/rpi$ sudo scp pi@192.168.1.3:/home/pi/config ~/rpi/config
[sudo] password for u201302423:
pi@192.168.1.3's password:
config 100% 131KB 131.2KB/s 00:00
u201302423@u201302423:~/rpi$ ls
config linux Module7.symvers test_module tools
u201302423@u201302423:~/rpi$ cd linux
u201302423@u201302423:~/rpi/linux$ cp ../config .config
u201302423@u201302423:~/rpi/linux$ sudo make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- oldconfig
scripts/kconfig/conf --oldconfig Kconfig
#
# configuration written to .config
#
u201302423@u201302423:~/rpi/linux$ make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- modules_prepare
scripts/kconfig/conf --silentoldconfig Kconfig
CHK include/config/kernel.release
CHK include/generated/uapi/linux/version.h
CHK include/generated/utsrelease.h
make[1]: 'include/generated/mach-types.h' is up to date.
CHK include/generated/bounds.h
CHK include/generated/timeconst.h
CHK include/generated/asm-offsets.h
CALL scripts/checksyscalls.sh
u201302423@u201302423:~/rpi/linux$
```

scp pi로 라즈베리파이에있는 config 파일을 가져온다
라즈베리가 가진 설정을 리눅스에서 컴파일이 설정과 맞추주기 위함이다
그리고 linux폴더에 그 config파일을 저장한뒤 커널컴파일과 모듈컴파일을 진행을 한다

```
u201302423@u201302423: ~/rpi
u201302423@u201302423:~/rpi/linux$ clear
u201302423@u201302423:~/rpi/linux$ cp ../Module7.symvers Module.symvers
u201302423@u201302423:~/rpi/linux$ ls
arch CREDITS firmware ipc lib Module.symvers samples tools
block crypto fs Kbuild MAINTAINERS net scripts usr
certs Documentation include Kconfig Makefile README security virt
COPYING drivers init kernel mm REPORTING-BUGS sound
```

그리고 모듈테스트시 버전을 확인하기위한 다운받은 심볼테이블인 Module.symvers파일을 옮겨준다

```
u201302423@u201302423: ~/rpi/test_module
u201302423@u201302423:~/rpi/test_module$ sudo make
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- -C /home/u201302423/rpi/linux M=/home/u201302423/rpi/test_module modules
make[1]: Entering directory '/home/u201302423/rpi/linux'
Building modules, stage 2.
MODPOST 1 modules
make[1]: Leaving directory '/home/u201302423/rpi/linux'
u201302423@u201302423:~/rpi/test_module$ sudo scp test_module.ko pi@192.168.1.3:/home/pi/test_module.ko
pi@192.168.1.3's password:
test_module.ko
100% 3460 3.4KB/s 00:00
u201302423@u201302423:~/rpi/test_module$
```

make로 크로스 컴파일을 해서 test_module.ko를 넘겨준다음

sudo insmod 커맨드를 통해 test_module.ko를 라즈베리 파이 커널에 적재하려고 하였으나 잘 되지 않았다. dmesg 커맨드를 통해 커널 메시지를 살펴봤다.

```
[ 282.352191] smst95xx 1-1.1:1.0 eth0: link up, 100Mbps, full-duplex, lpa 0xC0E1
[ 2211.720541] test module: disagrees about version of symbol module_layout
pi@raspberrypi:~ $
```

disagrees about version of symbol module_layout 라는 오류가 발생하고 있는 것을 확인하였다. Module7.symvers 파일에서 module_layout 항목을 확인해보았으나

0xf2265113	debugfs_attr_write	vmlinux	EXPORT_SYMBOL_GPL
0x7cdcd32d	seq_release	vmlinux	EXPORT_SYMBOL
0x7caab8a1	insert_inode_locked	vmlinux	EXPORT_SYMBOL
0x1866cec2	ring_buffer_size	vmlinux	EXPORT_SYMBOL_GPL
0x838b13e7	ring_buffer_free	vmlinux	EXPORT_SYMBOL_GPL
0xbca7617b	module_layout	vmlinux	EXPORT_SYMBOL
0x5c5a1b16	tick_broadcast_control	vmlinux	EXPORT_SYMBOL_GPL
0xc8b507b7	vchiq_shutdown	vmlinux	EXPORT_SYMBOL
0x216d759a	mmiocpy	vmlinux	EXPORT_SYMBOL

무엇을 어떻게 수정해야 할지 알 수 없었다.

disagrees about version of symbol module_layout을 키워드로 검색을 통해 문제점을 해결하려 애썼고 다음과 같은 사실을 알게 되었다.

리눅스 커널은 버전뿐만 아니라 컴파일 옵션에 따라서도 변하는 자료구조를 포함한다. 그 결과, 컴파일할 때는 커널 소스의 헤더파일뿐만 아니라 커널 컴파일 과정에서 생성된 헤더파일도 필요하다. 단순히 커널 소스의 압축을 푸는 것만으로는 부족하다.

CONFIG_MODVERSIONS 옵션을 통해 빌드된 커널은, 버전 번호는 다를 수 있으나, 자료구조의 레이아웃은 반드시 같아야만 한다. 이 옵션은 우분투 커널에서 활성화되어 있다. 이 옵션을 사용한다면, 모듈은 (헤더뿐만이 아니라) 적절한 Module.symvers 파일에 준해서 컴파일되어야 한다.

Module.symvers 파일의 중요성에 대해 알게 되었다. 제공되는 Module.symvers 파일이 현재의 최신버전이므로, 그 버전에 상응하는 raspbian이 필요할 것 같다.

우리가 받은 Module.symvers가 어떤지 확인하기 위해



..	
Module.symvers	kernel: Bump to 4.9.30
Module7.symvers	kernel: Bump to 4.9.30
System.map	kernel: Bump to 4.9.30
System7.map	kernel: Bump to 4.9.30
dt-blob.dts	kernel: Bump to 4.9.14
git_hash	kernel: Bump to 4.9.30
uname_string	kernel: Bump to 4.9.30
uname_string7	kernel: Bump to 4.9.30

git hub에서 확인결과 4.9버전을 위한 파일이다 하지만 현재 라즈베리파이는 4.4.50버전이다 그래서 서로 컴파일시 symbol version을 비교하면 달라서 컴파일거부가 일어난다.

해결방안으로 생각해본게 현재 4.4버전에 맞는 파일을 github에서 구해서 커널에 알맞게 수정해주는 방법과 라즈베리파이를 최신버전으로 업데이트 하는 방법을 생각해봤다.

버전을 낮추어 맞추려면 커널을 건드려야하는 부분이 많아서 상당히 복잡했다.

예를 들어 커널마다 magic number라는게 존재해서

리눅스는 /etc/magic이라는 파일을 읽어 파일의 타입이 무엇인지 표현하는데

이것을 이용하여 커널 데이터 구조를 보호하는 시스템인걸 알았다.

(<https://01.org/linuxgraphics/gfx-docs/drm/process/magic-number.html>에서

매직넘버의 개념을 이해하는데 도움을 받았다)

이 방법보단 최신 리눅스에 맞는 최신 라즈베리파일을 설치하는 것이 가장 쉽고 빠른 방법이라고 생각해서 실천하였다.

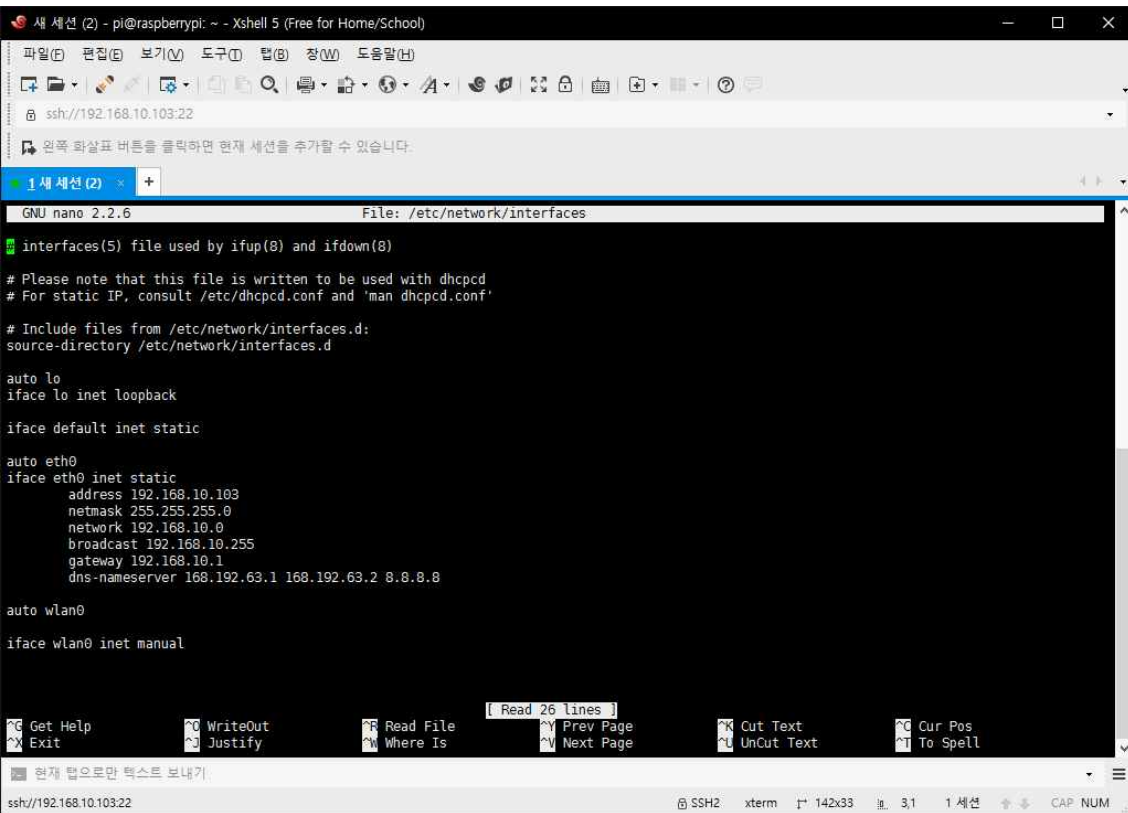
업데이트 하는 apt-get-update, upgrade 명령을 실행 하기 위해서는 라즈베리파이가 인터넷에 연결된 상태여야한다.

일단 공유기에 라즈베리파이를 연결한상태로 진행하였다.

동적 IP 할당 목록				
번호	호스트이름	IP 주소	MAC 주소	할당 정보
1	DESKTOP-L1JHFV3	192.168.10.100	E0:94:67:89:EA:D1	23:35:59
2	android-e7b82ca6e2b37c78	192.168.10.101	30:75:12:B2:C3:C9	23:34:48
3	DESKTOP-L1JHFV3	192.168.10.102	80:FA:5B:29:82:F1	23:36:53
4	raspberrypi	192.168.10.103	B8:27:EB:13:E3:5F	12:30:35

연결시 자동으로 공유기에서 IP를 할당해 주었는데 공유기의 내부네트워크 정보로 192.168.10.103을 할당한걸 알게되었다.

이 IP를 이용해서 Xshell로 새롭게 192.168.10.103으로 연결한뒤 IP를 고정으로 셋팅하였다.



```
GNU nano 2.2.6 File: /etc/network/interfaces
interfaces(5) file used by ifup(8) and ifdown(8)
# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

iface default inet static

auto eth0
iface eth0 inet static
    address 192.168.10.103
    netmask 255.255.255.0
    network 192.168.10.0
    broadcast 192.168.10.255
    gateway 192.168.10.1
    dns-nameserver 168.192.63.1 168.192.63.2 8.8.8.8

auto wlan0
iface wlan0 inet manual
```

IP주소와 네트워크 브로드캐스트 게이트웨이를 아이피에 맞게 바꿔주었다.

그냥 IP주소만 입력한다면 내부망(노트북) 핑이 보내지는데 외부망으로 핑을 보낼시 안 보내져서 조교님에게 물어보니 DNS서버를 설정하라고 하셔서

DNS-nameserver를 지정했다 앞에 두 개는 KT의 공개 DNS서버이고 마지막은 구글의 공개 DNS 서버이다

/etc/resolv.conf를 수정하여도 DNS서버가 바뀌는데 이 파일은 리부트할때마다 초기화하여서 네트워크 인터페이스에서 지정하였다.

인터넷연결이 완료되었으니 업데이트후 업그레이드를 하였다.

```
pi@raspberrypi:~ $ sudo apt-get update
Get:1 http://mirrordirector.raspbian.org jessie InRelease [14.9 kB]
Get:2 http://archive.raspberrypi.org jessie InRelease [22.9 kB]
Get:3 http://mirrordirector.raspbian.org jessie/main armhf Packages [9,534 kB]
Get:4 http://archive.raspberrypi.org jessie/main armhf Packages [163 kB]
Get:5 http://archive.raspberrypi.org jessie/ui armhf Packages [57.9 kB]
Get:6 http://mirrordirector.raspbian.org jessie/contrib armhf Packages [43.3 kB]
Get:7 http://mirrordirector.raspbian.org jessie/non-free armhf Packages [84.2 kB]
Get:8 http://mirrordirector.raspbian.org jessie/rpi armhf Packages [1,356 B]
Ign http://archive.raspberrypi.org jessie/main Translation-en_US
Ign http://archive.raspberrypi.org jessie/main Translation-en
Ign http://archive.raspberrypi.org jessie/ui Translation-en_US
Ign http://archive.raspberrypi.org jessie/ui Translation-en
Ign http://mirrordirector.raspbian.org jessie/contrib Translation-en_US
Ign http://mirrordirector.raspbian.org jessie/contrib Translation-en
Ign http://mirrordirector.raspbian.org jessie/main Translation-en_US
Ign http://mirrordirector.raspbian.org jessie/main Translation-en
Ign http://mirrordirector.raspbian.org jessie/non-free Translation-en_US
Ign http://mirrordirector.raspbian.org jessie/non-free Translation-en
Ign http://mirrordirector.raspbian.org jessie/rpi Translation-en_US
Ign http://mirrordirector.raspbian.org jessie/rpi Translation-en
Fetched 9,921 kB in 18s (531 kB/s)
Reading package lists... Done
```

```
Removing 'diversion of /boot/overlays/pwm.dtbo to /usr/share/rpikernelhack/overlays/pwm.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/qca7000.dtbo to /usr/share/rpikernelhack/overlays/qca7000.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/raspidac3.dtbo to /usr/share/rpikernelhack/overlays/raspidac3.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-backlight.dtbo to /usr/share/rpikernelhack/overlays/rpi-backlight.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-cirrus-ws102.dtbo to /usr/share/rpikernelhack/overlays/rpi-cirrus-ws102.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-dac.dtbo to /usr/share/rpikernelhack/overlays/rpi-dac.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-display.dtbo to /usr/share/rpikernelhack/overlays/rpi-display.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-ft5406.dtbo to /usr/share/rpikernelhack/overlays/rpi-ft5406.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-proto.dtbo to /usr/share/rpikernelhack/overlays/rpi-proto.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-sense.dtbo to /usr/share/rpikernelhack/overlays/rpi-sense.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rpi-tv.dtbo to /usr/share/rpikernelhack/overlays/rpi-tv.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/rra-digidacl-wm8741-audio.dtbo to /usr/share/rpikernelhack/overlays/rra-digidacl-wm8741-audio.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sc16is750-i2c.dtbo to /usr/share/rpikernelhack/overlays/sc16is750-i2c.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sc16is752-sp11.dtbo to /usr/share/rpikernelhack/overlays/sc16is752-sp11.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sdhost.dtbo to /usr/share/rpikernelhack/overlays/sdhost.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sdio-lbit.dtbo to /usr/share/rpikernelhack/overlays/sdio-lbit.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sdio.dtbo to /usr/share/rpikernelhack/overlays/sdio.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/sdtweak.dtbo to /usr/share/rpikernelhack/overlays/sdtweak.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/smi-dev.dtbo to /usr/share/rpikernelhack/overlays/smi-dev.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/smi-nand.dtbo to /usr/share/rpikernelhack/overlays/smi-nand.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/smi.dtbo to /usr/share/rpikernelhack/overlays/smi.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi-gpio35-39.dtbo to /usr/share/rpikernelhack/overlays/spi-gpio35-39.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi-rtc.dtbo to /usr/share/rpikernelhack/overlays/spi-rtc.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi0-cs.dtbo to /usr/share/rpikernelhack/overlays/spi0-cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi0-hw-cs.dtbo to /usr/share/rpikernelhack/overlays/spi0-hw-cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi1-1cs.dtbo to /usr/share/rpikernelhack/overlays/spi1-1cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi1-2cs.dtbo to /usr/share/rpikernelhack/overlays/spi1-2cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi1-3cs.dtbo to /usr/share/rpikernelhack/overlays/spi1-3cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi2-1cs.dtbo to /usr/share/rpikernelhack/overlays/spi2-1cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi2-2cs.dtbo to /usr/share/rpikernelhack/overlays/spi2-2cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/spi2-3cs.dtbo to /usr/share/rpikernelhack/overlays/spi2-3cs.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/tinyld35.dtbo to /usr/share/rpikernelhack/overlays/tinyld35.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/uart1.dtbo to /usr/share/rpikernelhack/overlays/uart1.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/vc4-fkms-v3d.dtbo to /usr/share/rpikernelhack/overlays/vc4-fkms-v3d.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/vc4-kms-v3d.dtbo to /usr/share/rpikernelhack/overlays/vc4-kms-v3d.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/vq666.dtbo to /usr/share/rpikernelhack/overlays/vq666.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/wl-gpio-pullup.dtbo to /usr/share/rpikernelhack/overlays/wl-gpio-pullup.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/wl-gpio.dtbo to /usr/share/rpikernelhack/overlays/wl-gpio.dtbo by rpikernelhack'
Removing 'diversion of /boot/overlays/wittypi.dtbo to /usr/share/rpikernelhack/overlays/wittypi.dtbo by rpikernelhack'
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 4.9.24+ /boot/kernel.img
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 4.9.24+ /boot/kernel.img
run-parts: executing /etc/kernel/postinst.d/apt-auto-removal 4.9.24-v7+ /boot/kernel7.img
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 4.9.24-v7+ /boot/kernel7.img
Setting up libraspberrypi-bin (1.20170427-1) ...
Setting up raspberrypi-ui-modes (1.20170503) ...
Installing new version of config file /etc/xdg/openbox/lxde-pi-rc.xml ...
Backing up old config files...
/home/pi/.config/openbox/lxde-pi-rc.xml
Processing triggers for libc-bin (2.19-18+deb8u9) ...
Processing triggers for initramfs-tools (0.120+deb8u3) ...
Processing triggers for ca-certificates (20141019+deb8u3) ...
Updating certificates in /etc/ssl/certs... 0 added, 0 removed: done.
Running hooks in /etc/ca-certificates/update.d....done.
pi@raspberrypi:~ $
```

업데이트후 업그레이드 완료

라즈베리파이의 커널 버전확인을 하였다니 4.9버전으로 업그레이드 된걸 확인하였다.
우리의 예상이 맞다면 Config파일을 다시생성후 넘겨준후 커널컴파일과 모듈컴파일을
다시한뒤 make를 한다면 잘될 것이다.

앞서 했던거와 같이 config 파일을 만들고 한번 확인해보았다.

전과 달리 4.9버전으로 바뀐 config 파일을 확인할 수 있었다.

이전 config는 지우고 새로운 config를 다시 우분투로 가져왔다

그 후 리눅스 폴더로 접근후 cp ../config .cofing 명령어를 실행해 리눅스폴더에
.config 파일을 생성해서 이제 모드 최신버전 라즈베리파이의 .config 파일을 가지게 되었고
그런다음 앞서 했던거와 같이 커널컴파일과 모듈컴파일을 진행해주었다.
moudle.symbor는 깃허브에서 다운받은게 최신버전에 맞춰졌기 때문에 크로스 컴파일에서
오류가 생기지 않을 것이다.

```
u201302423@u201302423: ~/rpi/module_test
u201302423@u201302423:~/rpi/module_test$ sudo make
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- -C /home/u201302423/rpi/linux M=/home/u201302423/rpi/module_test modules
make[1]: Entering directory '/home/u201302423/rpi/linux'
  CC [M]  /home/u201302423/rpi/module_test/test_module.o
Building modules, stage 2.
MODPOST 1 modules
  CC      /home/u201302423/rpi/module_test/test_module.mod.o
  LD [M]  /home/u201302423/rpi/module_test/test_module.ko
make[1]: Leaving directory '/home/u201302423/rpi/linux'
u201302423@u201302423:~/rpi/module_test$ sudo scp test_module.ko pi@192.168.10.103:/home/pi/test_module.ko
pi@192.168.10.103's password:
test_module.ko                                100% 3460      3.4KB/s   00:00
u201302423@u201302423:~/rpi/module_test$
```

sudo make를 해준다음 해당 test_module.ko를 바뀐 IP주소로 복사 해주었다.

```
새 세션 (2) - pi@raspberrypi: ~ - Xshell 5 (Free for Home/School)
파일(F) 편집(E) 보기(V) 도구(D) 탭(T) 장(W) 도움말(H)
ssh://192.168.10.103:22
원쪽 화살표 버튼을 클릭하면 현재 세션을 추가할 수 있습니다.
1 세션 (2)
pi@raspberrypi:~ $ ls
config Desktop Documents Downloads Music oldconffiles Pictures Public python_games Templates test_module.ko Videos
pi@raspberrypi:~ $ sudo insmod test_module.ko
pi@raspberrypi:~ $ dmesg | tail -5
[ 12.828164] Bluetooth: BNEP (Ethernet Emulation) ver 1.3
[ 12.828173] Bluetooth: BNEP filters: protocol multicast
[ 12.828188] Bluetooth: BNEP socket layer initialized
[ 36.152039] test_module: loading out-of-tree module taints kernel.
[ 36.152524] Hello rpi World!
pi@raspberrypi:~ $ sudo rmmod test_module
pi@raspberrypi:~ $ dmesg | tail -5
[ 12.828173] Bluetooth: BNEP filters: protocol multicast
[ 12.828188] Bluetooth: BNEP socket layer initialized
[ 36.152039] test_module: loading out-of-tree module taints kernel.
[ 36.152524] Hello rpi World!
[ 43.730773] Bye rpi!
pi@raspberrypi:~ $
```

Xshell에서 test_module.ko가 생긴걸 확인하였고

그후 모듈을 적재한뒤 메시지를 확인하였다. Hello rpi World가 잘 출력되었고
모듈제거시 Bye rpi도 잘 출력되었다.