|  |  |
| --- | --- |
| **Program / Project Name:** | Tips for Development |
| **Checklist / Template Completed by:** | TBD |
| **Date Completed:** | Click here to enter a date. |

**Tips for Development**

Revision: A.0

**Detailed Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev** | **Date** | **Editor(s)** | **Description of change** |
| A.0 | 24-MAY-2019 | Sang-Gu Kang | VirtualBox |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[1 Linux 5](#_Toc10108351)

[1.1 command 5](#_Toc10108352)

[1.1.1 linux system 정보 확인 5](#_Toc10108353)

[2 VirtualBox 6](#_Toc10108354)

[2.1 VirtualBox Installation 6](#_Toc10108355)

[2.2 Ubuntu installation 6](#_Toc10108356)

[2.2.1 Ubuntu 16.04 LTS 설치 6](#_Toc10108357)

[2.2.2 Ubuntu 16.04 LTS 한글 설정 6](#_Toc10108358)

[2.2.3 Ubuntu 16.04 LTS 소프트웨어 업데이트 6](#_Toc10108359)

[2.2.4 Ubuntu 16.04 LTS 공유폴더 설정 6](#_Toc10108360)

[2.2.5 주의사항 7](#_Toc10108361)

[2.3 Qt5 설치 7](#_Toc10108362)

[2.3.1 reference 7](#_Toc10108363)

[2.4 Development Environment(Ubuntu 14.01) 22](#_Toc10108364)

[2.5 가상 시스템 내보내기/가져오기 22](#_Toc10108365)

[2.5.1 내보내기 22](#_Toc10108366)

[2.6 Network Setup 27](#_Toc10108367)

[2.6.1 Current Setup 확인 27](#_Toc10108368)

[2.6.2 네트워크 설정 특징 29](#_Toc10108369)

[2.6.3 가상화 모드 설정 가능 Ethernet Card 32](#_Toc10108370)

[2.7 Serial Port Setup 32](#_Toc10108371)

[2.7.1 Window serial port setup check 32](#_Toc10108372)

[2.7.2 VirtualBox Serial Port Setup 34](#_Toc10108373)

[2.7.3 Serial Port Test using minicom 35](#_Toc10108374)

[3 Qt(linux) 39](#_Toc10108375)

[3.1 외부 Library 추가하기 39](#_Toc10108376)

# Linux

## command

### linux system 정보 확인

#### Serial Port 정보 확인하기

dmesg|grep tty

#### 32bit/64bit 확인

getconf LONG\_BIT

# VirtualBox

## VirtualBox Installation

아래 사이트 참조. 특히 해시값 확인

<https://wnw1005.tistory.com/97>

<https://makingrobot.tistory.com/11?category=801859>

## Ubuntu installation

아래 사이트를 참조로 설치한다.

### Ubuntu 16.04 LTS 설치

<https://makingrobot.tistory.com/13>

### Ubuntu 16.04 LTS 한글 설정

<https://makingrobot.tistory.com/14?category=801859>

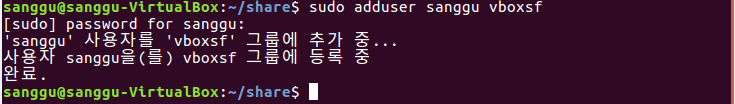
### Ubuntu 16.04 LTS 소프트웨어 업데이트

<https://makingrobot.tistory.com/15?category=801859>

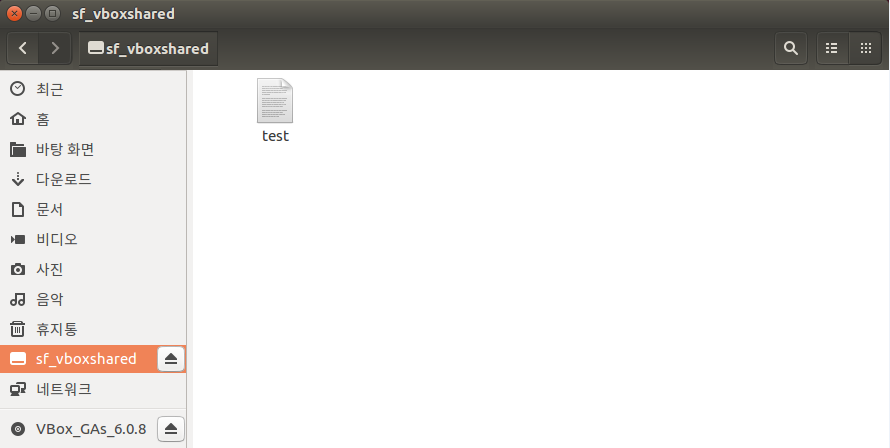
### Ubuntu 16.04 LTS 공유폴더 설정

<https://makingrobot.tistory.com/18?category=801859>

자동으로 공유폴더가 마운트 되도록 설정하기 위해서 로그인 계정에 vboxsf 그룹을 추가해야 한다.



그룹을 추가한 후에 재부팅하여 폴더를 열어보면 정상적으로 공유됨을 확인할 수 있다.



### 주의사항

공유폴더 설정이나, 전체 화면 구성을 위해서는 “게스트 확장CD 이미지 삽입”을 해야 하는데 설치 이후 바로 해주면 좋고, 한번만 해주면 된다.

## Qt5 설치

### reference

<https://webnautes.tistory.com/1120>

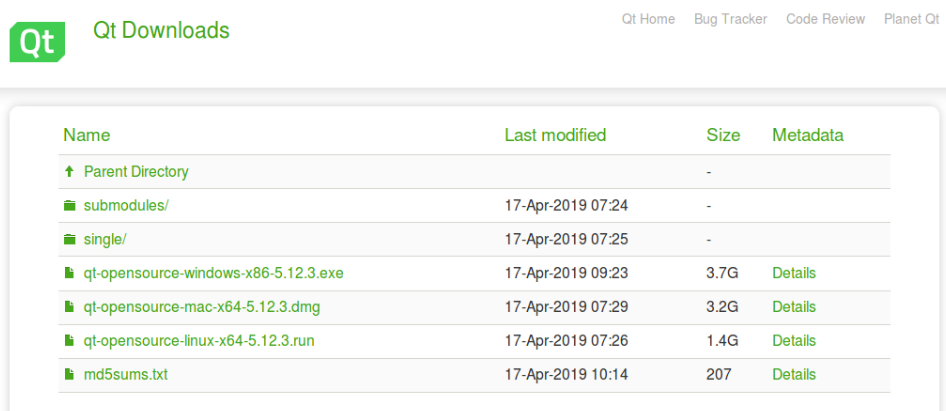
<https://wiki.qt.io/Install_Qt_5_on_Ubuntu>

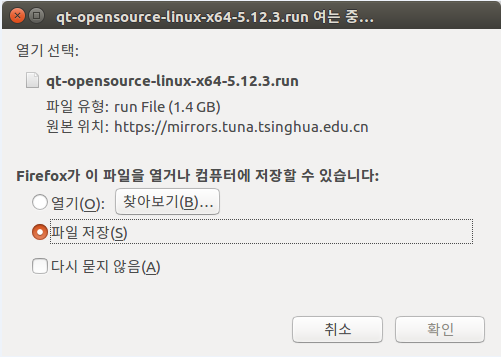
<https://m.blog.naver.com/PostView.nhn?blogId=yally23232&logNo=220795372958&proxyReferer=https%3A%2F%2Fwww.google.com%2F>

### download

적당한 장소에서 설치 파일을 다운로드 받는다.

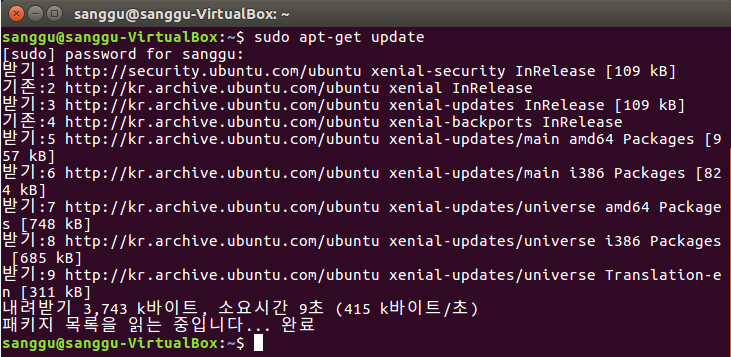
<https://download.qt.io/official_releases/qt/5.12/5.12.3/>



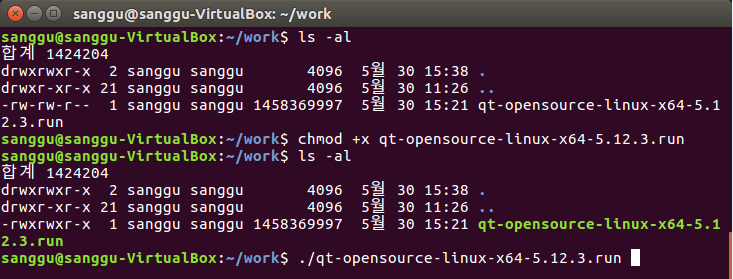


### install

패키지 리스트 업데이트



권한변경 및 설치



스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

Login 없이 skip해도 상관 없음.

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

스크린샷이(가) 표시된 사진

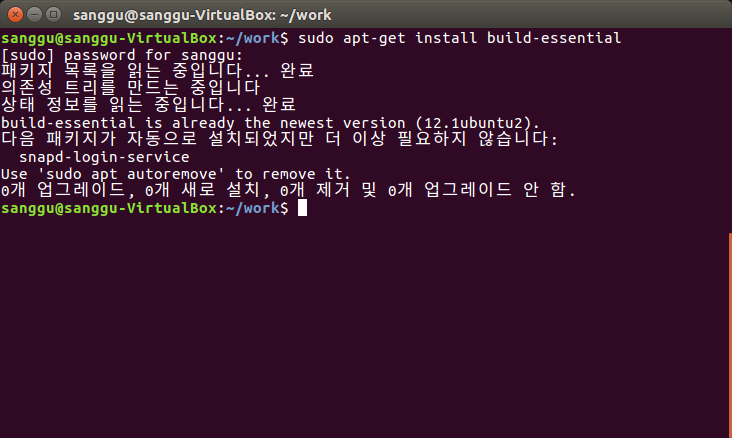
자동 생성된 설명

스크린샷이(가) 표시된 사진

자동 생성된 설명

### Install etc

C/C++ 컴파일러와 C/C++를 위한 라이브러리, 헤더파일, make같은 유틸리티 도구 등이 포함되어 있는 build-essential 패키지를 설치한다.

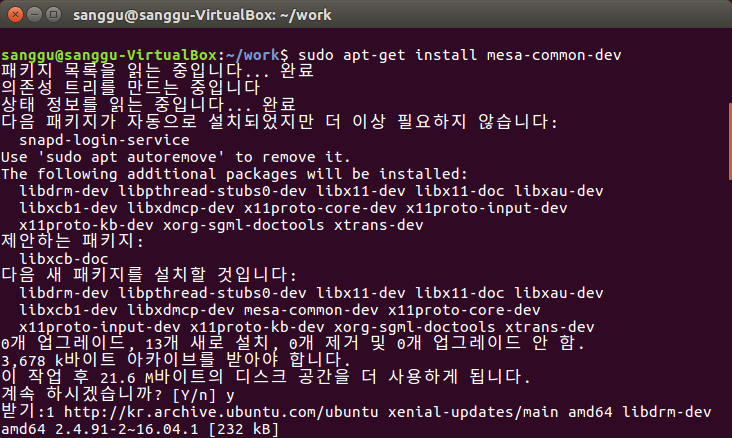


Generic font configuration library -runtime

스크린샷이(가) 표시된 사진

자동 생성된 설명

OpenGL libraries



스크린샷이(가) 표시된 사진

자동 생성된 설명

### 환경 추가

.bashrc 파일 수정

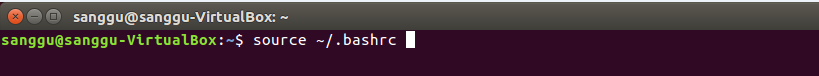


Editor로 열어서 맨 마지막 줄에 내용 추가

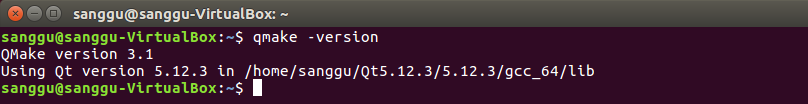
스크린샷이(가) 표시된 사진

자동 생성된 설명

패스 환경 추가한 것 적용



확인



### Simple Compile example

Qt Creator를 사용하지 않고 터미널에서 간단한 Qt 예제 코드를 만들고 컴파일하는 과정을 보여준다.

Simple.cpp 파일 생성



스크린샷이(가) 표시된 사진

자동 생성된 설명

qmake -project 명령으로 프로젝트 파일 생성

스크린샷이(가) 표시된 사진

자동 생성된 설명

Qt 프로젝트 파일이 생성될 때, qt 모듈이 자동으로 추가되지 않으므로 추가. 사용중인 qt 모듈을 찾아서 포함시키되 디렉토리 이름에서 Qt를 빼고 소문자로 변환한 widgets를 추가한다.

스크린샷이(가) 표시된 사진

자동 생성된 설명

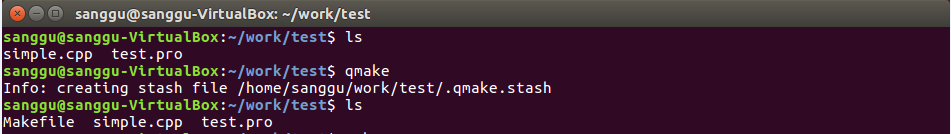


스크린샷이(가) 표시된 사진

자동 생성된 설명

qmake 명령으로 Makefile을 생성하고, make 명령으로 컴파일 및 링크를 실행.

생성된 test 실행 파일을 실행



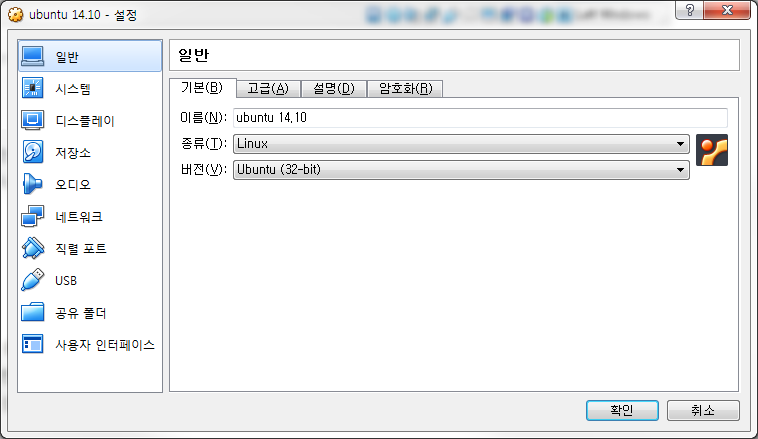
스크린샷이(가) 표시된 사진

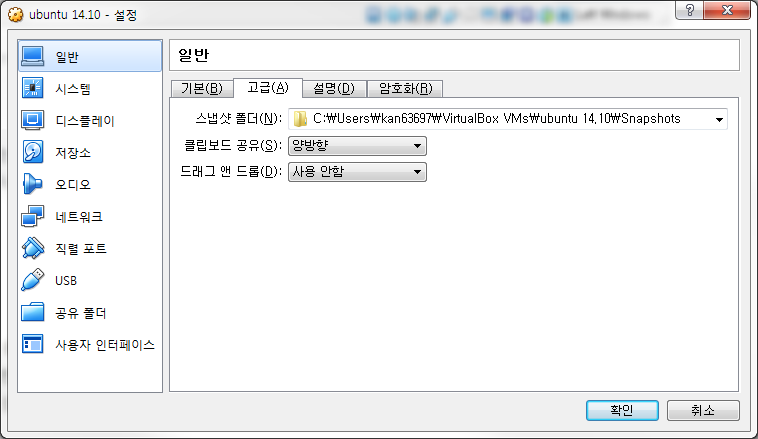
자동 생성된 설명

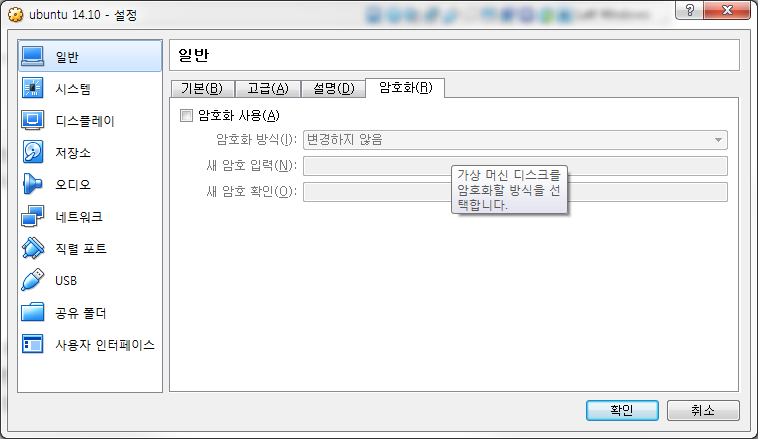
스크린샷이(가) 표시된 사진

자동 생성된 설명

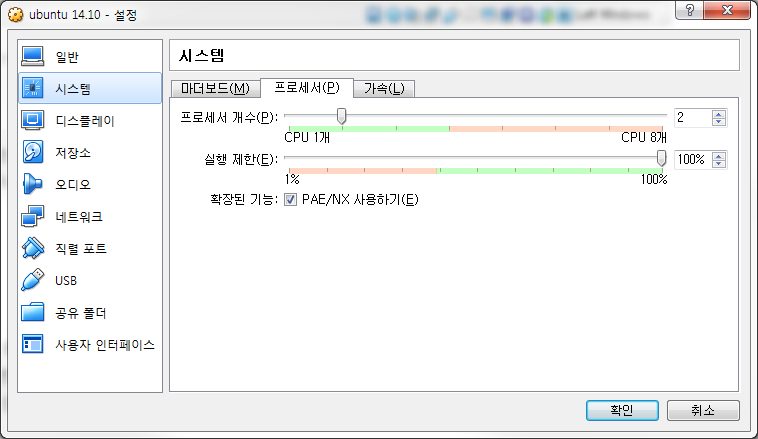
## Ubuntu14 설정 내용

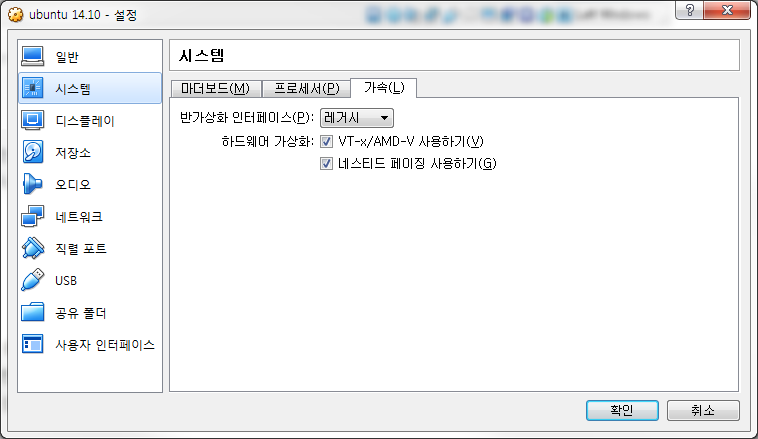




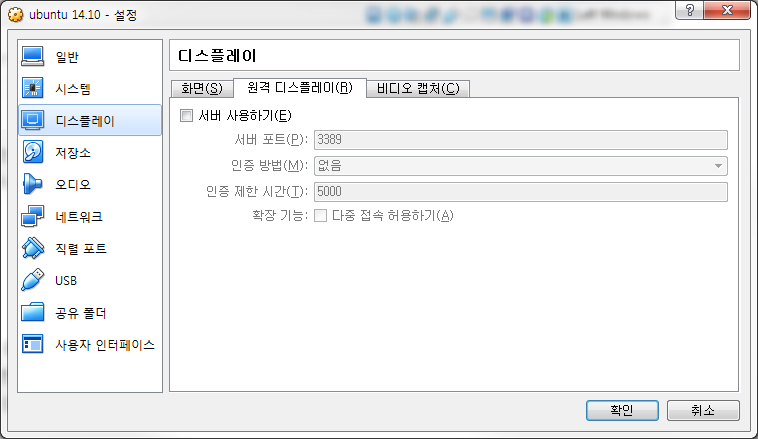


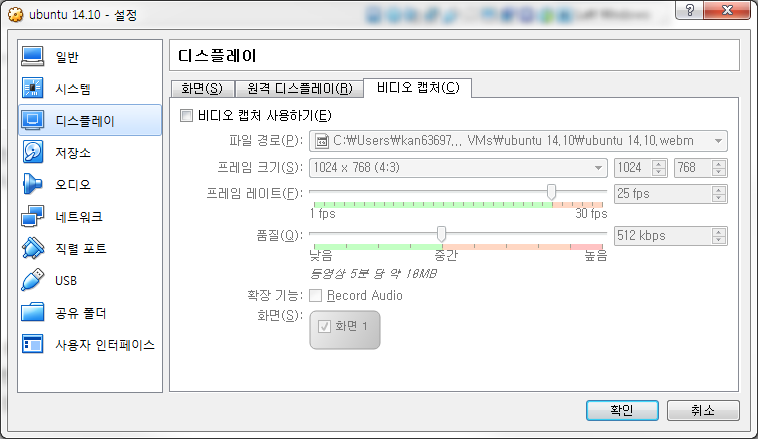


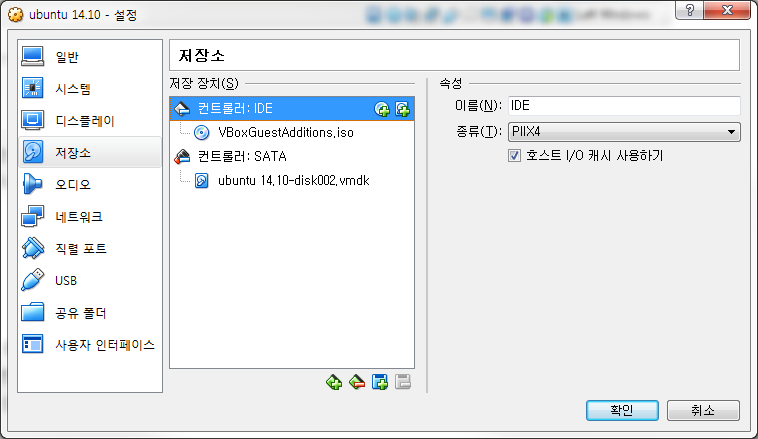


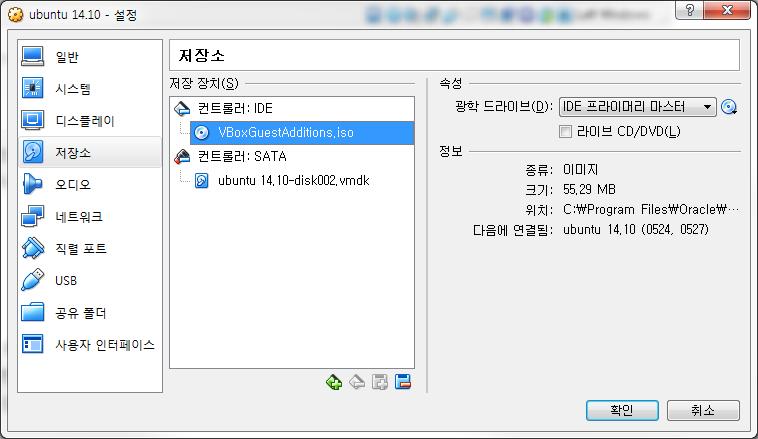


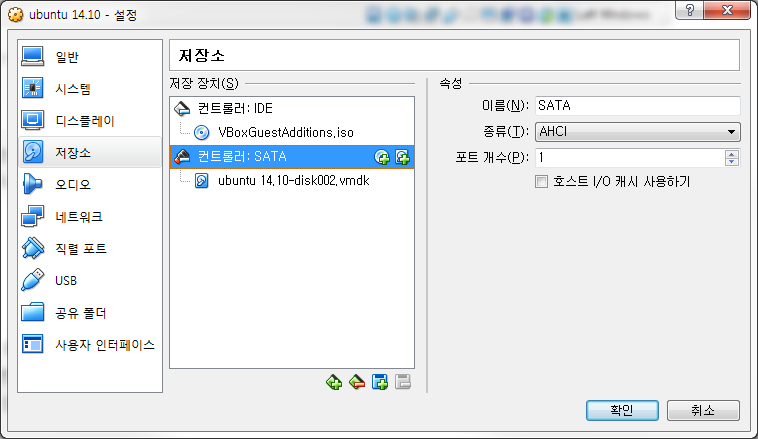


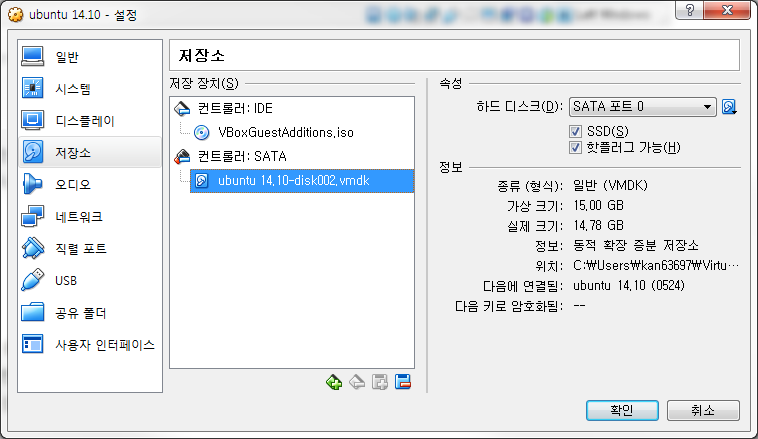


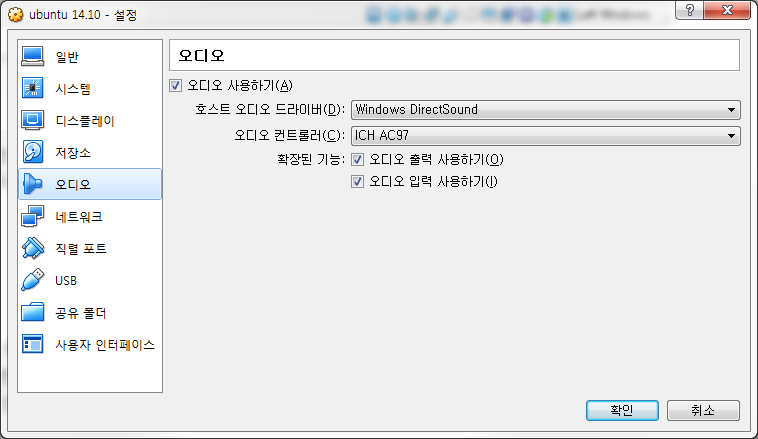


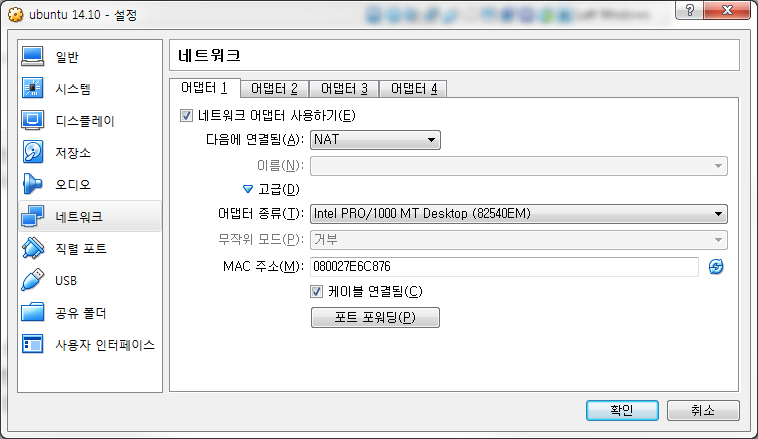




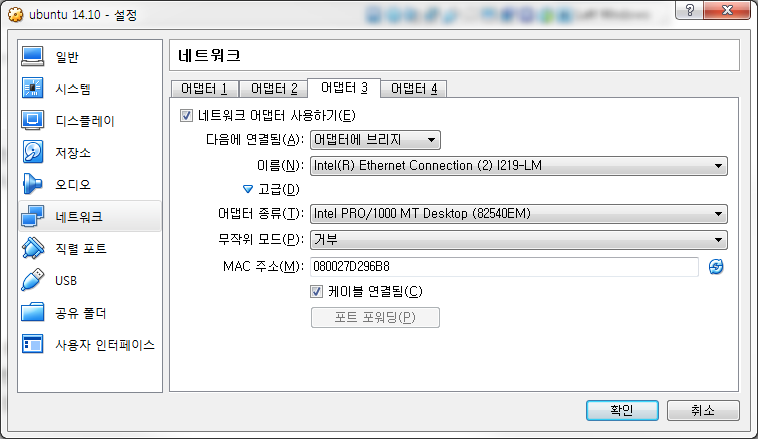


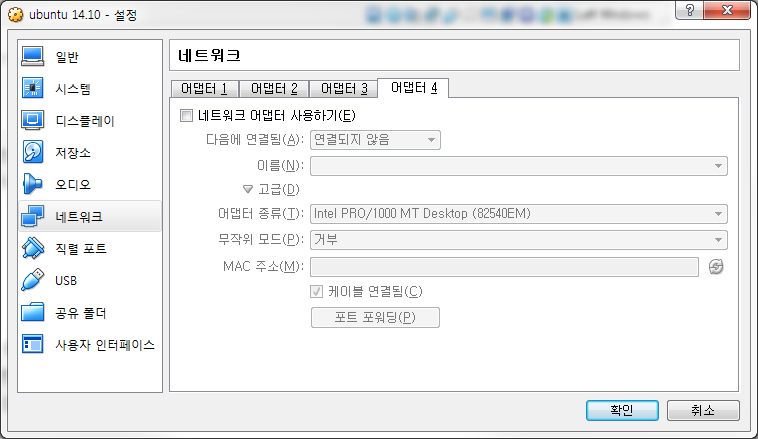


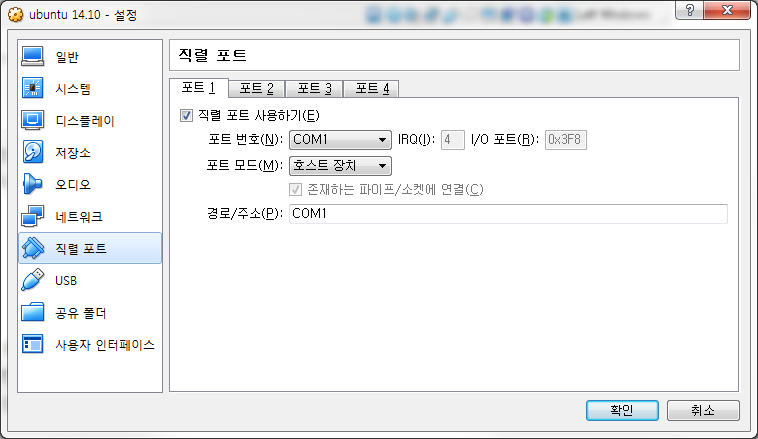


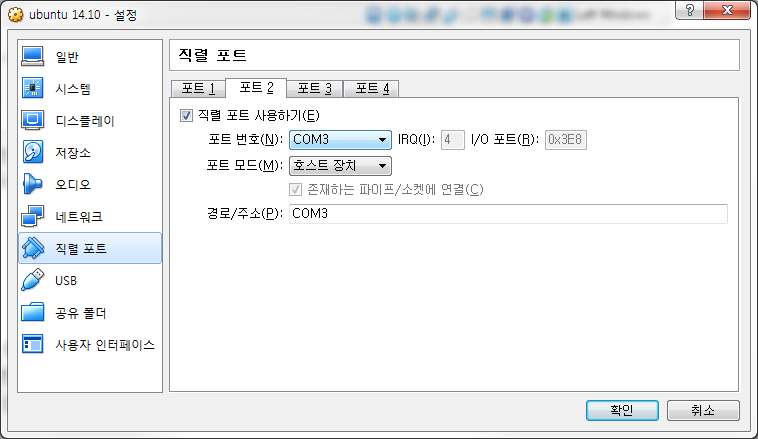


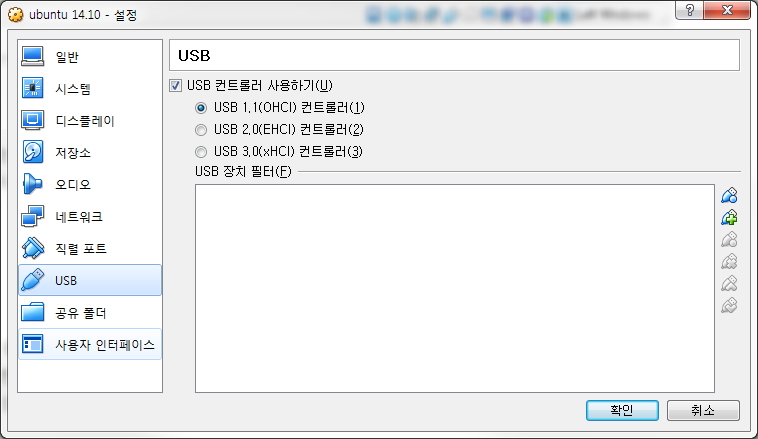


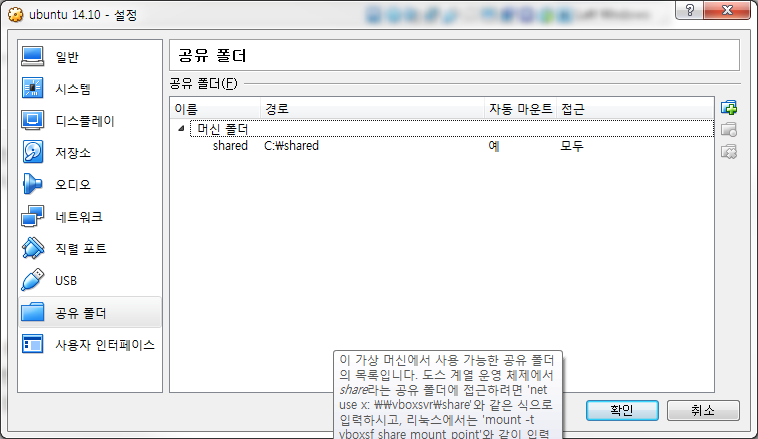


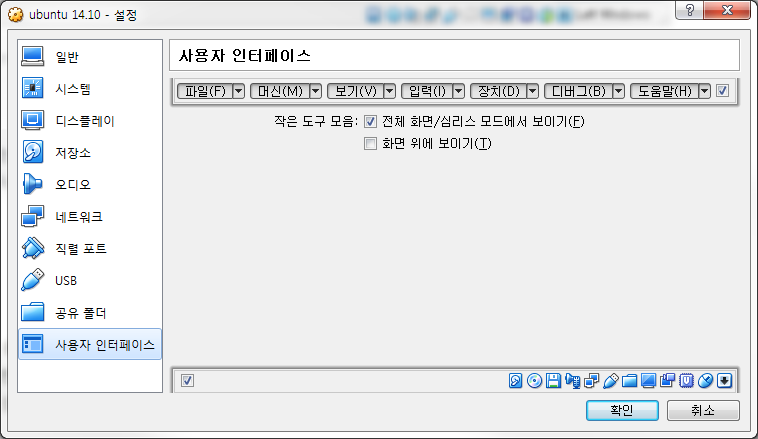












## Development Environment(Ubuntu 14.01)

id: jdsu

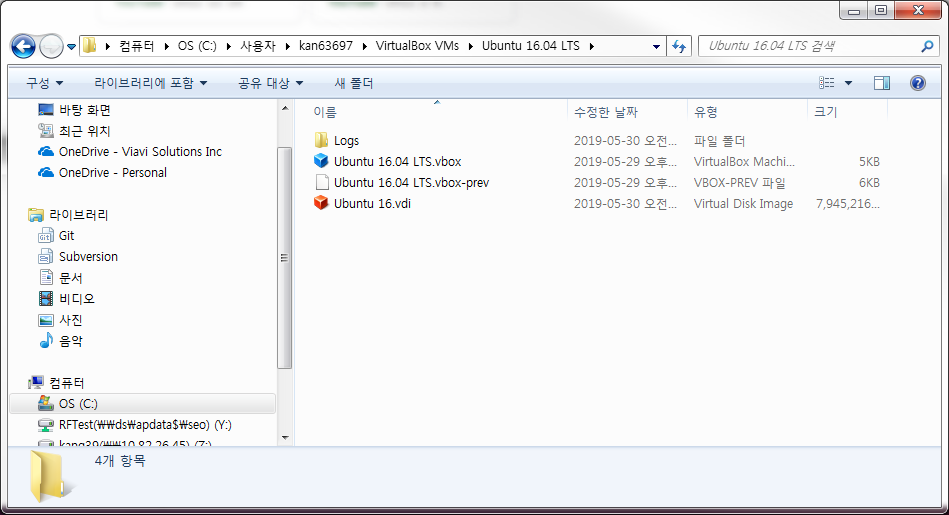
pw: genc0mm

각종 backup file 위치: C:\works\VMTemp\ubuntu14

## 가상 시스템 내보내기/가져오기

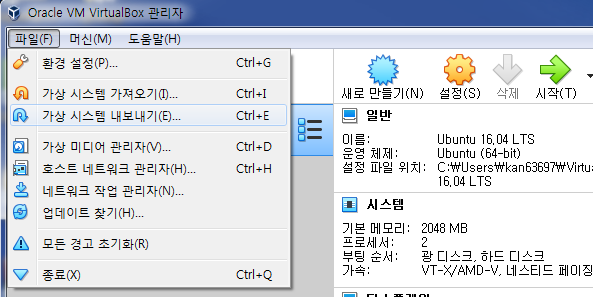
### 내보내기

VirtualBox의 VM 파일이 저장된 폴더에 가보면 아래와 같이 다수의 파일들로 구성되어 있다.

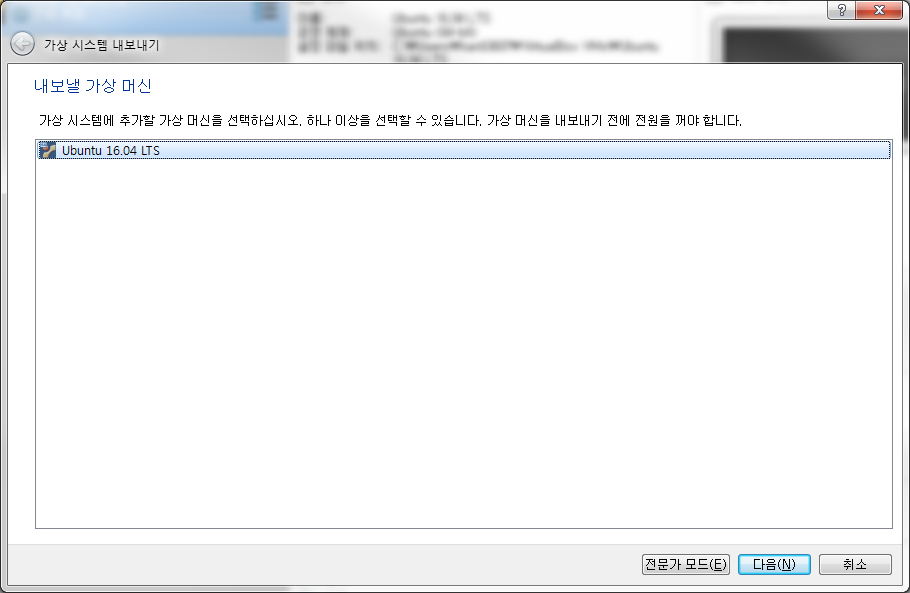


이런 각종 파일들을 하나의 ova(Open Virtualization Format Archive)라는 확장자의 파일로 추출해 내게 되고 이는 다른 PC에 그대로 적용할 수 있다.

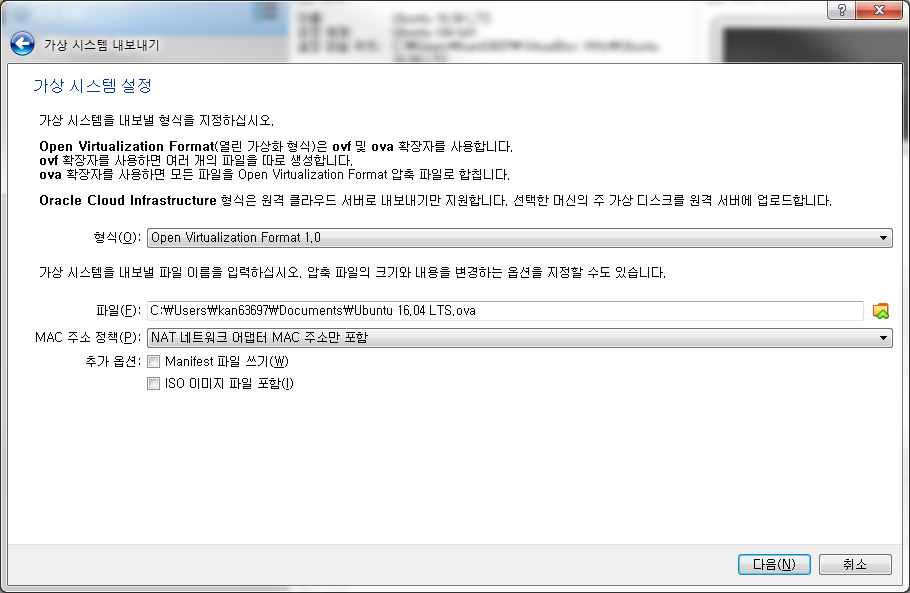
1. 파일 > 가상 시스템 내보내기 선택



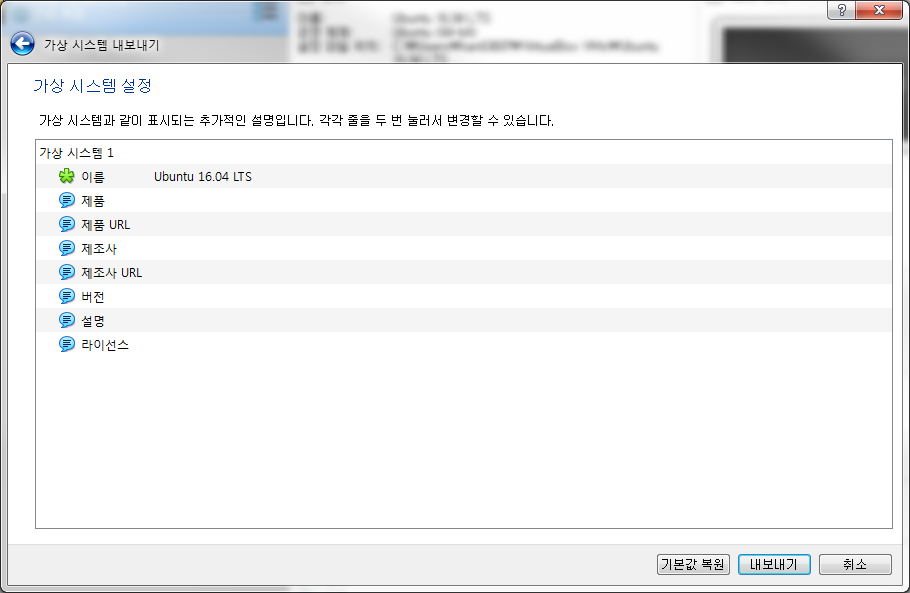
2. 내보내기 할 가상머신을 선택한다. 하나 이상을 선택할 수 있다. 가상 머신을 내보내기 전에 전원을 꺼야 한다.



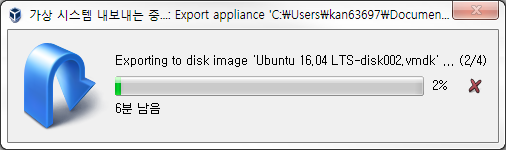
3. 내보내게 될 위치 및 형식을 선택한다. 형식에 대해 잘 모른다면 그냥 기본으로 선택하면 된다.



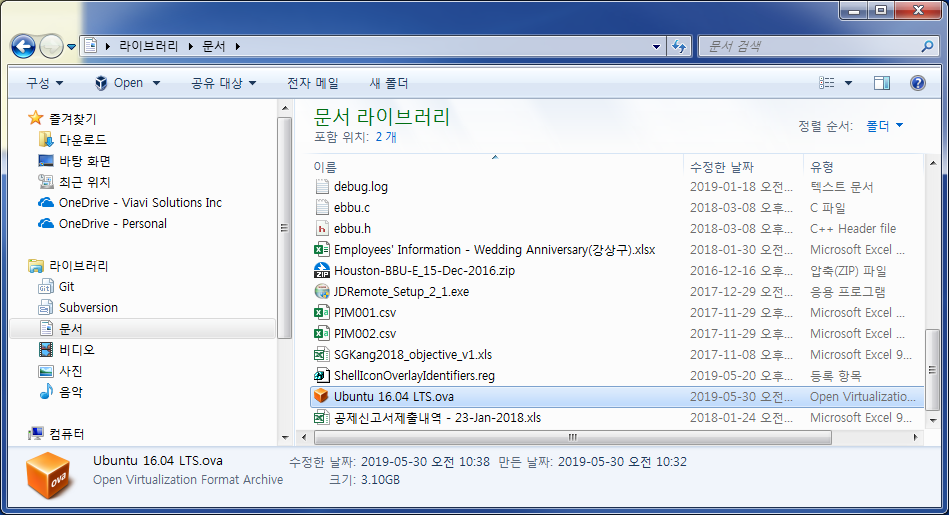
4. 추가적인 정보를 입력해도 좋지만 필요 없다면 내보내기 버튼을 누른다.



5. 아래와 같이 하나의 ova 파일로 추출이 진행된다.

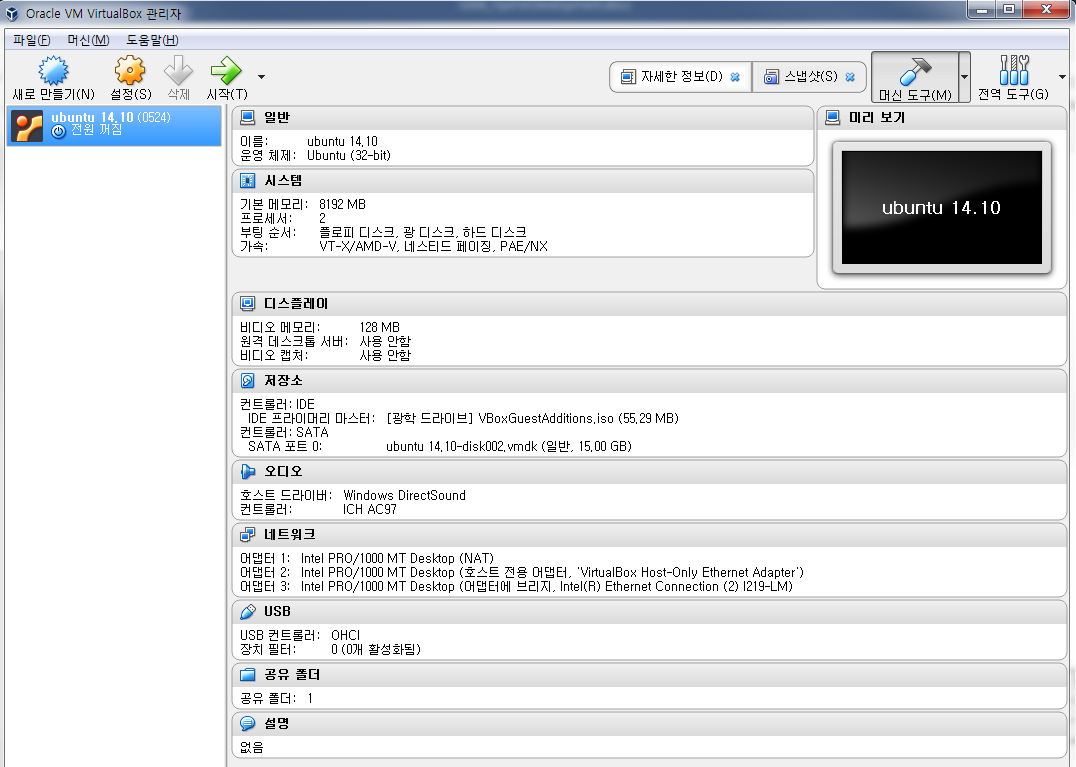


6. 추출이 완료되면 저장 위치에 하나의 파일이 생성된다.

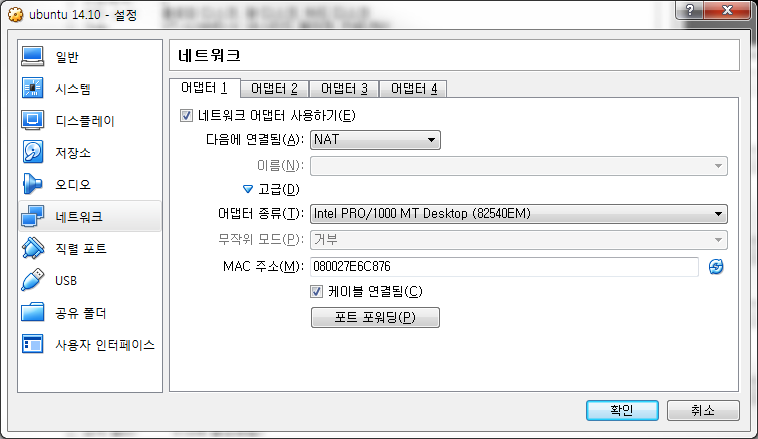


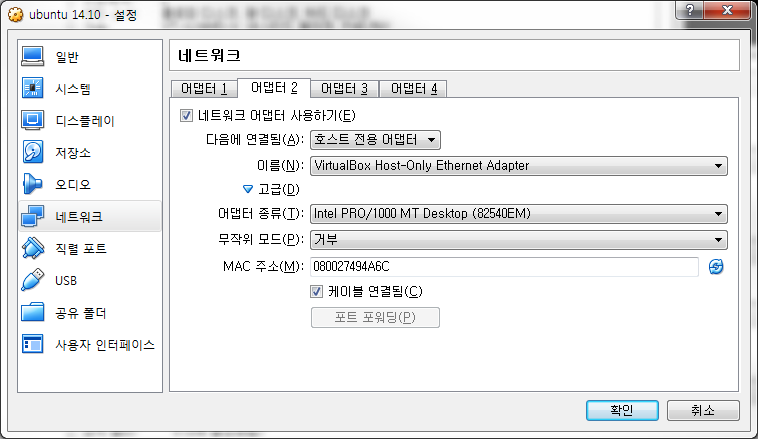
## Network Setup

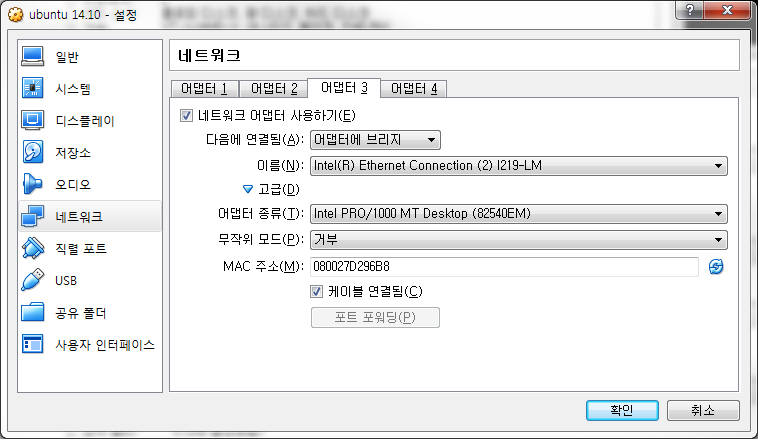
### Current Setup 확인



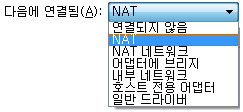








### 네트워크 설정 특징



#### 연결되지 않음

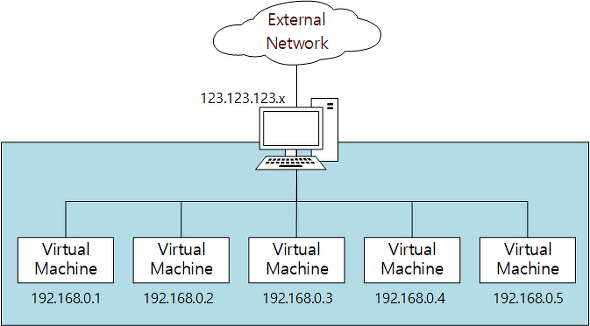
네트워크 카드는 존재하지만 마치 케이블을 꼽지 않은 것과 같이 연결하지 않은 것 처럼 동작하는 것.

#### NAT(Network Address Translation)

* 가상머신 내부 네트워크 -> Host PC 외부 네트워크 단방향 통신 가능
* Host PC 내의 가상 머신 간의 통신 불가능

기본 선택 사항. 가상 머신 내에서 단순히 인터넷 사용이나 파일 다운로드와 같은 용도로만 사용할 경우 설정이 가능하다. 가상 머신 내에서는 자체 사설 IP 주소를 사용하지만 실제 외부로 나갈 때는 Host PC가 가진 IP 주소를 바꿔 달고 나가게 된다. Host PC의 외부 네트워크에서는 내부의 가상 머신에 접근 불가능하다.

즉, 아래 그림과 같이 NAT 모드일 경우 192.168.0.1의 주소를 사용하는 가상 머신은 외부 네트워크와 통신할 때 123.123.123.x 주소를 달고 외부와 통신하게 된다. 반면 192.168.0.x 의 주소를 가진 다른 가상 머신들과는 통신이 불가능하다.

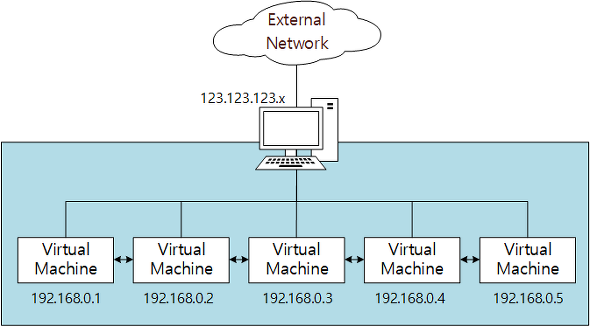


#### NAT Network

* 가상머신 내부 네트워크 -> Host PC 외부 네트워크 단방향 통신 가능
* Host PC 내의 가상 머신 간의 통신 가능

해당 모드는 NAT와 거의 동일하지만 Host PC 내의 가상 머신끼리 통신이 가능하다는 점이 다르다. 가상 머신 간에 상호 통신이 필요한 작업이 있다면 해당 모드를 사용해야 한다.

NAT Network의 경우는 아래 그림과 같이 NAT와 다 동일하지만 192.168.0.x의 주소를 가진 다른 가상 머신들과도 통신이 가능하다는 점이 다르다.



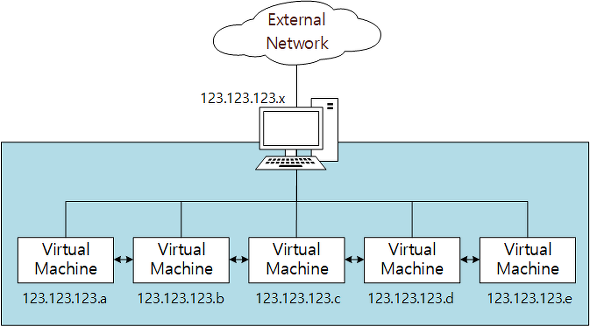
#### 브리지 어댑터(Bridged Adapter)

* Host PC와 동등한 수준의 네트워크 구성
* 추가의 IP 할당이 필요함

해당 모드는 가상 머신의 네트워크를 Host PC와 동등 수준으로 구성할 수 있게 해준다. 즉, 예를 들어 Host PC의 네트워크가 192.168.0.2로 설정되어 있다면 가상 머신의 네트워크 또한 192.168.0.4 정도로 같은 네트워크로 설정할 수 있는 것이다.

실제 물리적인 네트워크 인터페이스는 Host PC의 Ehternet 카드 하나밖에 없다. 그렇다면 어떻게 Host PC와 가상 머신의 네트워크를 구분해 내는 것일까? VirtualBox의 Manual에 따르면 소위 “net filter”라 불리는 Host PC의 network device driver를 사용하여 구분해 낸다고 한다. 해당 net filter를 통해 가상 머신은 자신의 data를 Host PC의 네트워크 카드로부터 분리해내고 삽입하기도 한다.

브리지 어댑터의 경우 아래 그림과 같이 물리적으로는 Host PC 내에 존재하지만 Host PC의 주소와 동등한 수준의 별도의 주소를 가지고 외부와 통신한다.

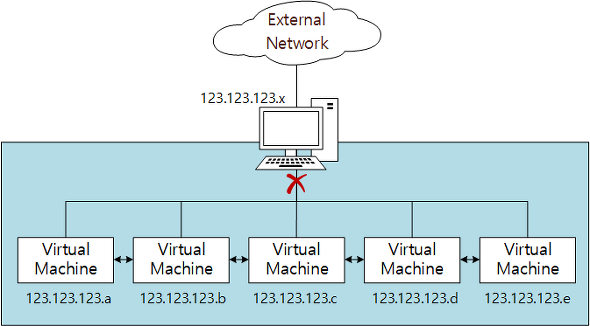


#### 내부 네트워크 (Internal Network)

* Host PC와 독립적인 내부 네트워크
* 같은 Host PC 내의 가상 머신 간 연결이 가능

해당 모드는 브리지 어댑터와 많은 부분에서 유사하지만 가장 큰 차이점으로 외부와 통신이 되지 않는다는 점이 있다. 즉, Host PC의 네트워크와 분리되어 있고, Host PC 내 가상머신 간에만 상호 연결하여 통신이 가능하다. Host PC의 물리적인 네트워크 인터페이스와 연결되어 있지 않다.

아래 그림과 같이 내부 네트워크는 Host PC 네트워크와 분리되어 있고, 가상 머신 간에만 통신이 가능하다.

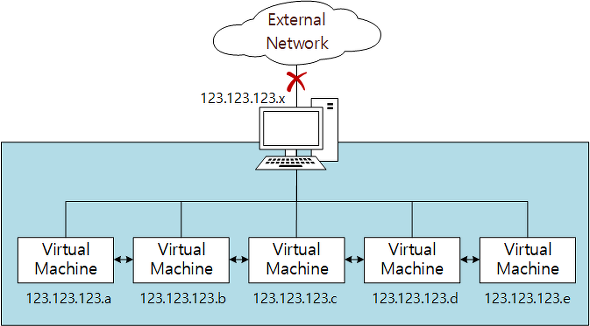


#### 호스트 전용 어댑터(Host-only Adapter)

* Host PC를 포함한 내부 네트워크(외부 네트워크와는 단절)
* 같은 Host PC 내의 가상머신 간 연결이 가능

해당 모드는 “브리지 어댑터” 모드와 같이 Host PC와 내부 가상머신들과 통신이 가능하고, “내부 네트워크” 모드와 같이 외부 네트워크와는 통신이 불가능하다. Host PC의 물리 네트워크 어댑터 외에 별도의 Loopback 어댑터를 생성하여 해당 어댑터를 통해 통신한다.

호스트 전용 어댑터의 경우는 아래 그림과 같이 외부 네트워크가 단절되어 있다.



### 가상화 모드 설정 가능 Ethernet Card

* AMD PCNet PCI II (Am79C970A)
* AMD PCNet FAST III (Am79C973, the default)
* Intel PRO/1000 MT Desktop (82540EM)
* Intel PRO/1000 T Server (82543GC)
* Intel PRO/1000 MT Server (82543EM)
* Paravirtualized network adapter (virtio-net)

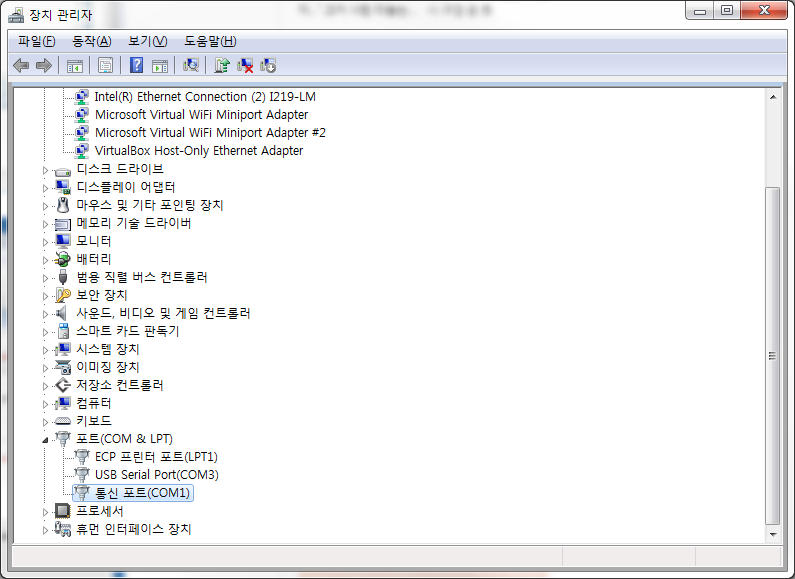
## Serial Port Setup

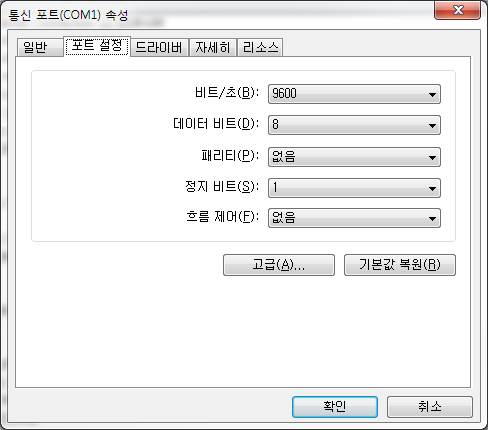
### Window serial port setup check

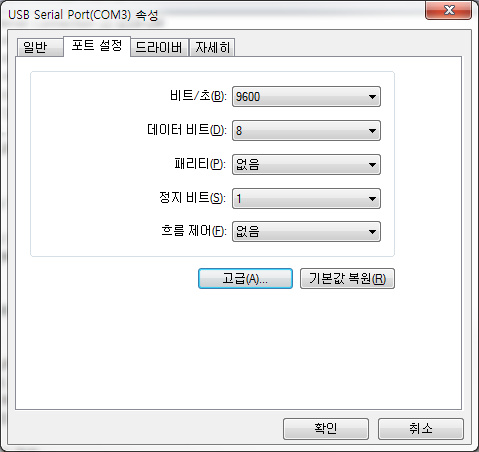
제어판 > 장치 관리자 선택

현재 노트북 기본 제공 COM1과 RS-485통신을 위한 USB Serial Port COM3가 연결되어 있다.

각각의 포트에 대한 속성에서 포트 설정을 확인해 둔다.

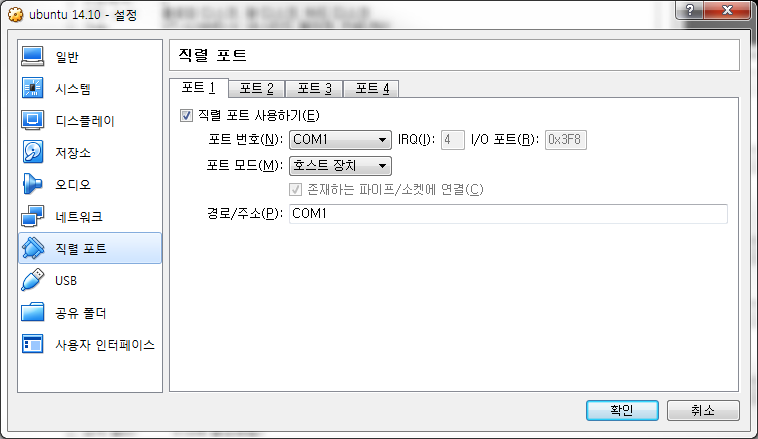


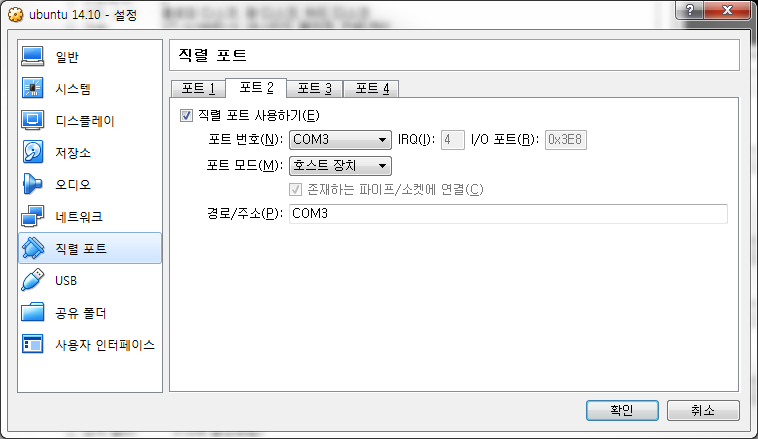




### VirtualBox Serial Port Setup

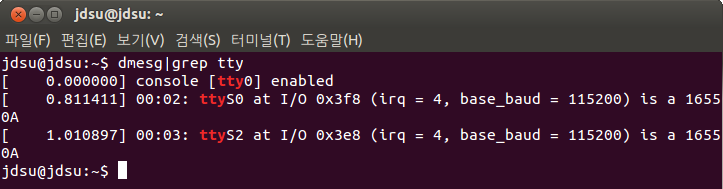
설정에서 직렬포트를 선택하고 설정값을 입력한다.



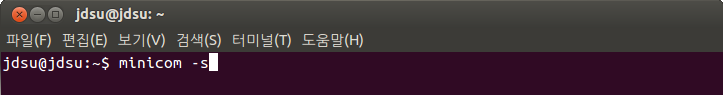


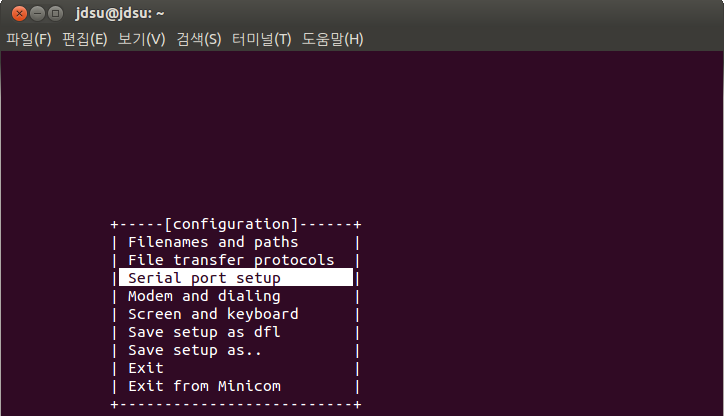
### Serial Port Test using minicom

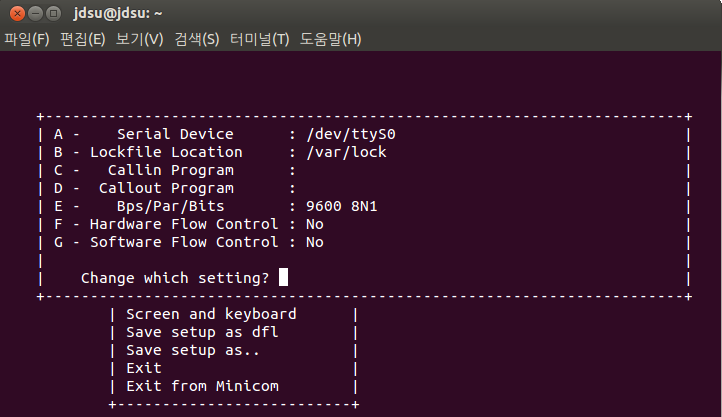
현재 연결된 serial port 상태 확인



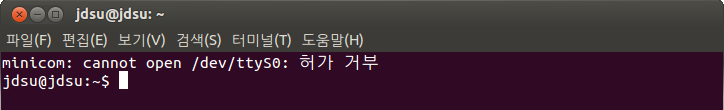
minicom 이용하기. 쉘에서 minicom -s 명령을 실행하여 Serial Port 설정을 한다.







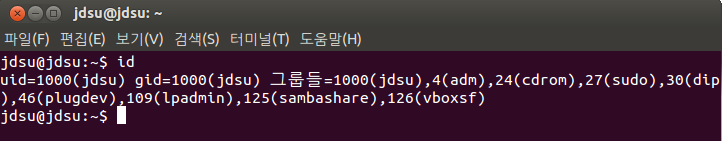
설정을 하였으나 permission denied가 발생하여 접근이 거부되는 경우가 있다.

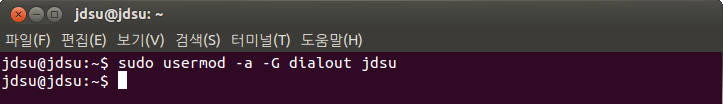


접근이 거부되면, 아래와 같이 설정한다.

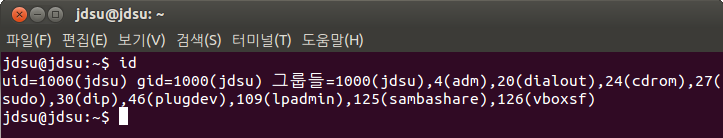
아래 4개의 스크린샷이 주요 설정 과정이다. 나머지는 minicom 설정이다.



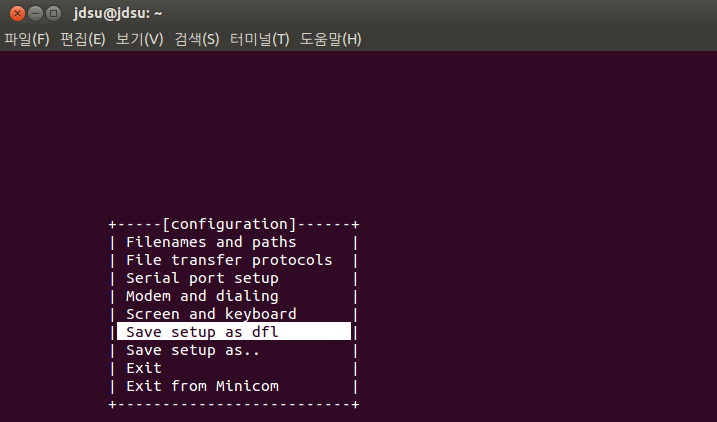




재부팅 후 id를 확인하면 dialout이 추가된 걸 볼 수 있다.



Default 설정 값이 저장되지 않으면 minicom을 root permission으로 실행한다. #sudo minicom -s



minicom을 실행하고 serial cable의 2번, 3번 을 임시로 연결한 후 키보드로 타이핑을 하게 되면 해당 글자가 보인다. 2,3번 연결을 해제하면 키보드 타이핑에도 반응하지 않는다. 이것으로 전송된 데이터가 루프백으로 되돌아오는 시험을 성공할 수 있다.

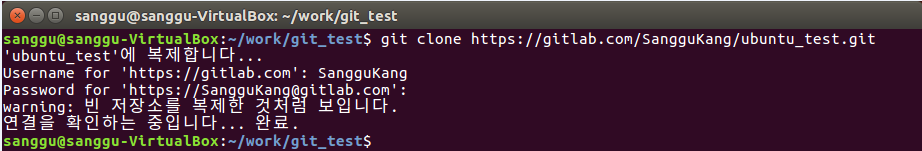
## Git

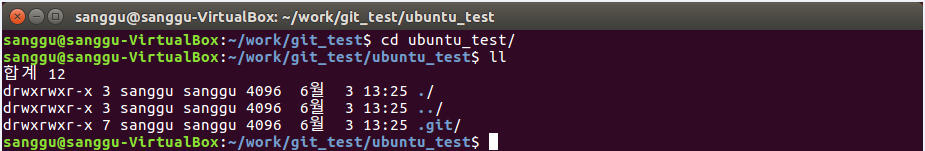
### Git 설치

sudo apt-get install git

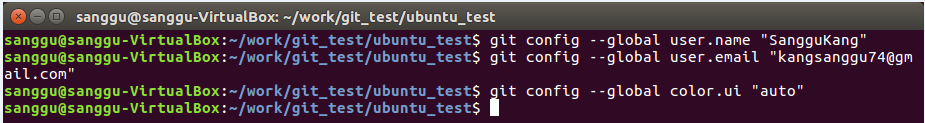
git version

### Git Hosting





### Git 설정



## 숨김 파일 보기

Ctrl+h를 누르거나 편집>기본설정>숨김/백업 파일 보이기 체크

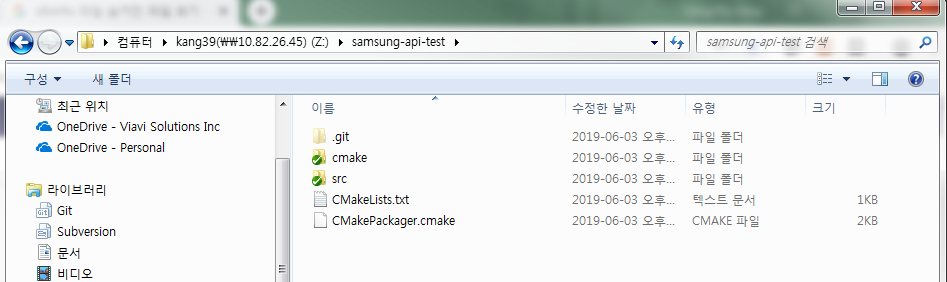
스크린샷이(가) 표시된 사진

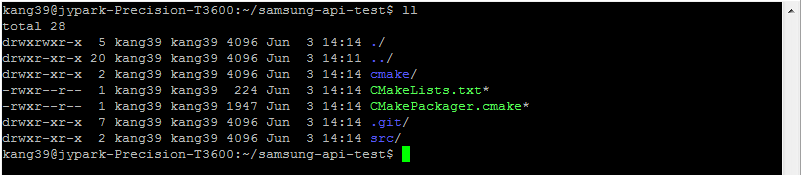
자동 생성된 설명

# Cross Compile

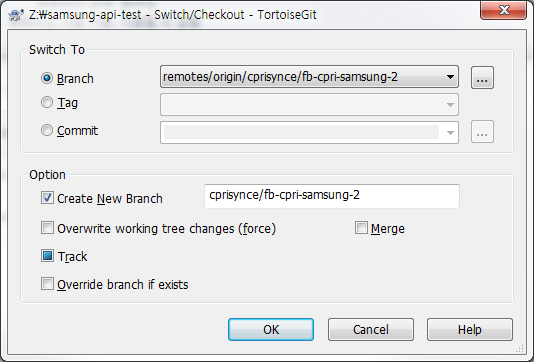
## LKII Server

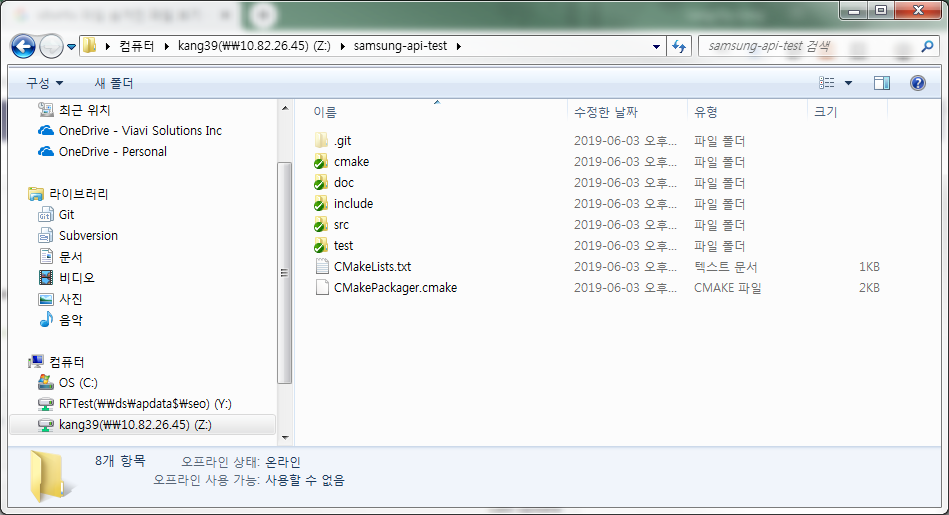
### Clone from GitLab

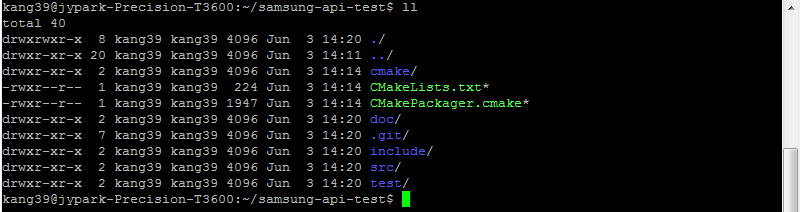




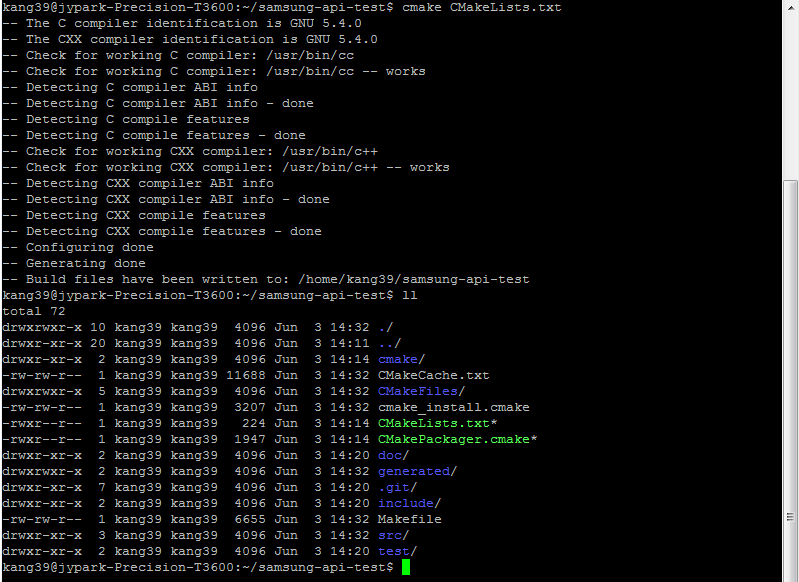
### Move to branch

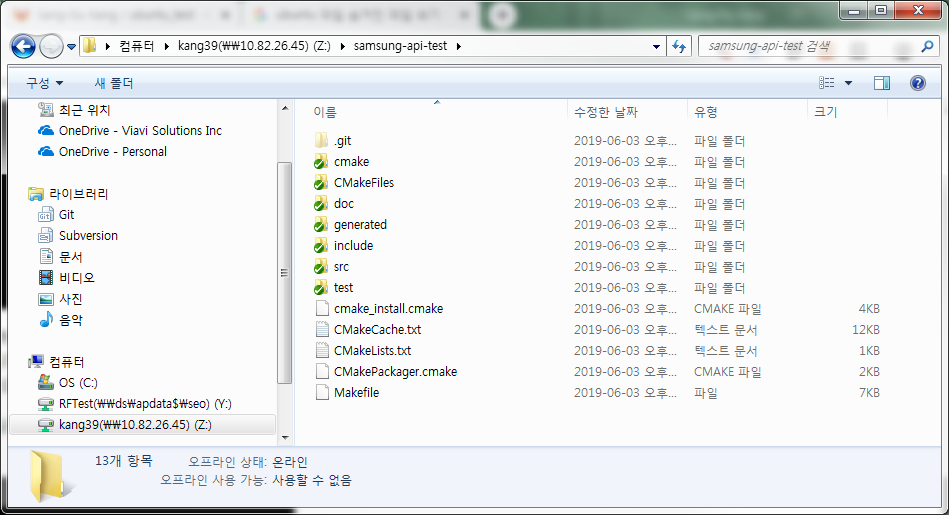




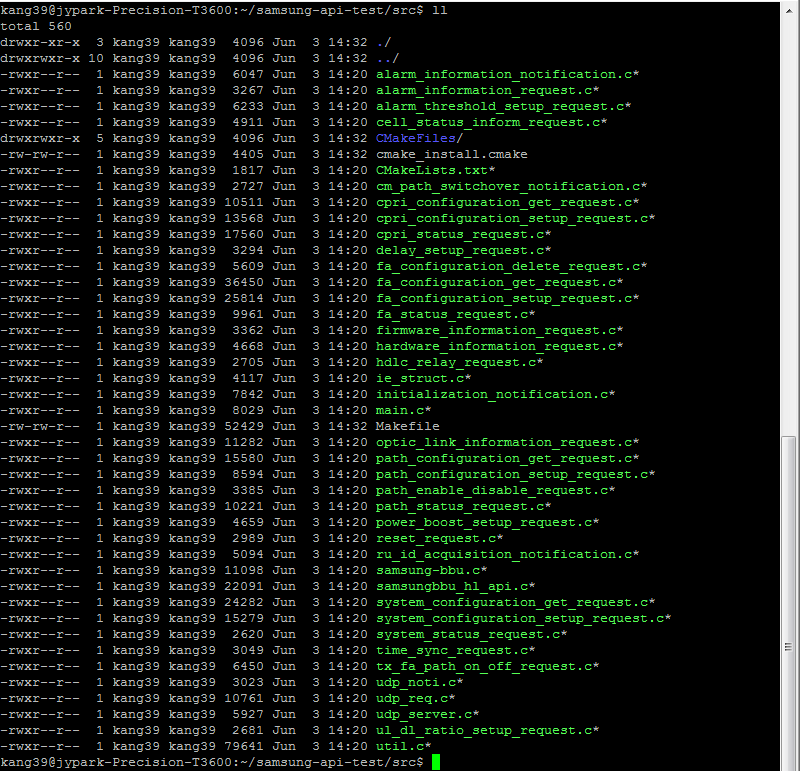


### CMake, make

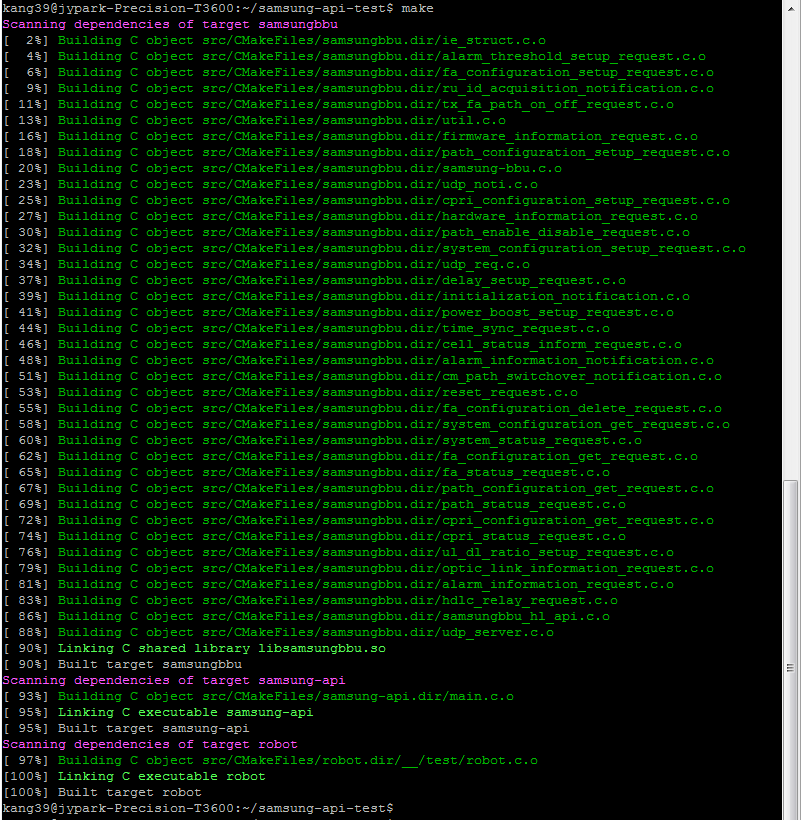


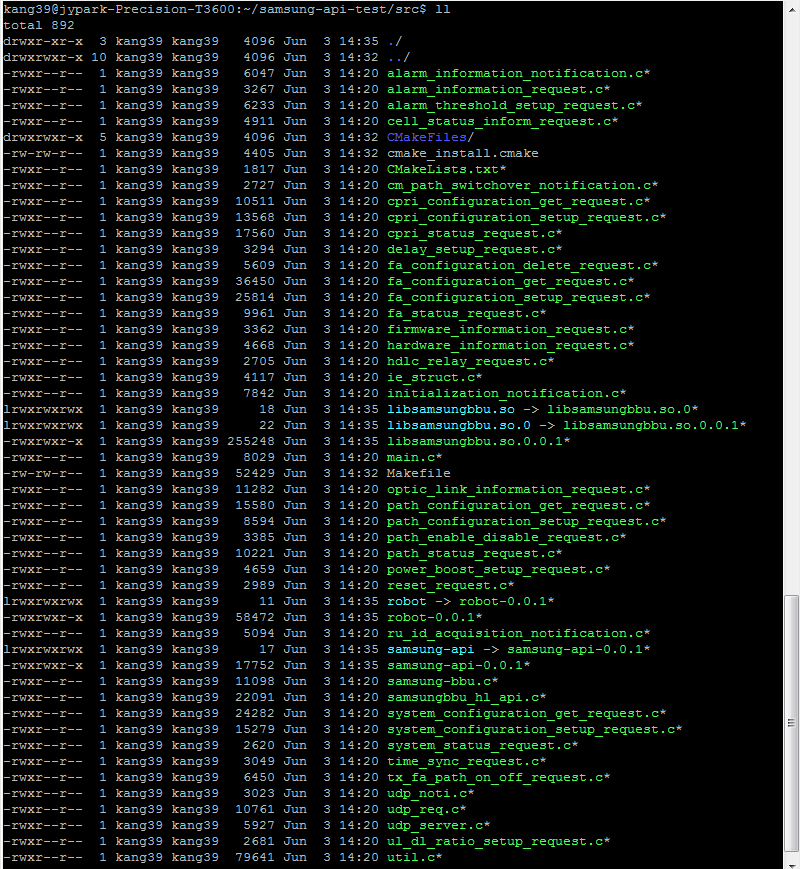


make 전



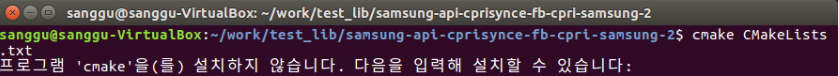
make 후

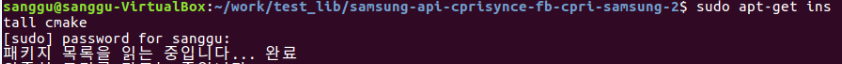




## Ubuntu X86\_64

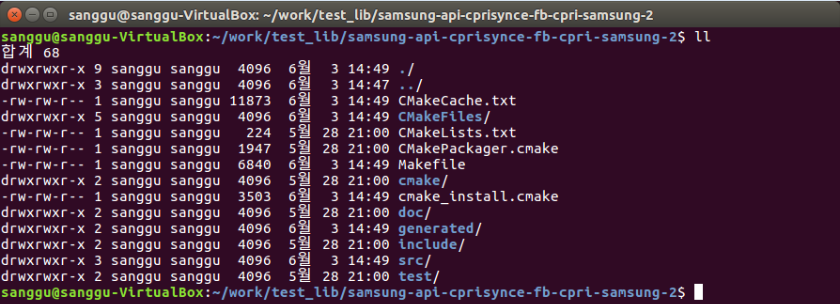
### Cmake 설치

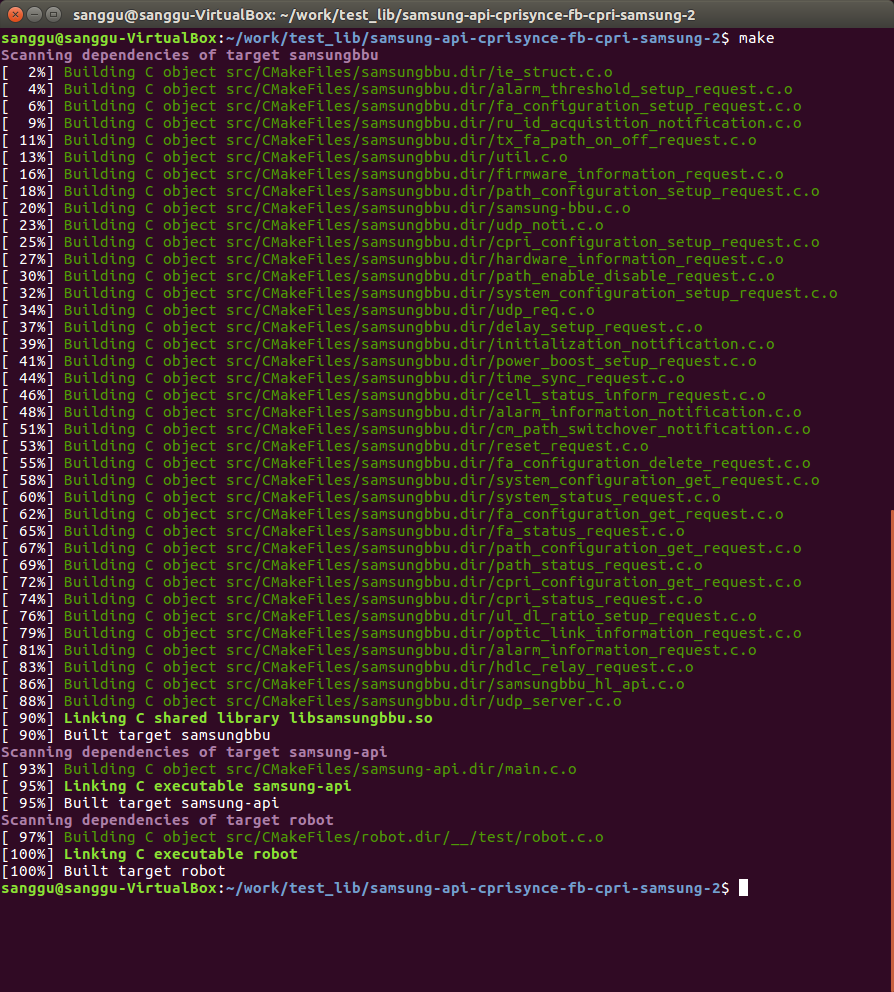




### CMake, make









# Qt(linux)

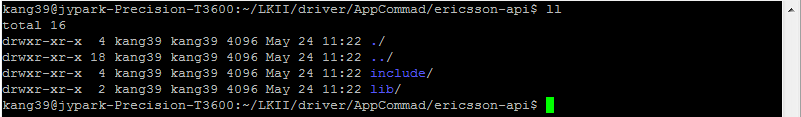
## LKII using library

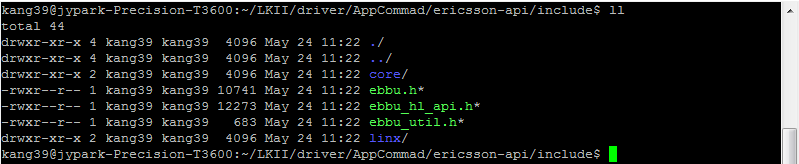
### LK2.pro

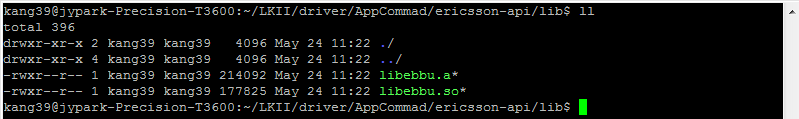




### LKII Source driver







## 외부 Library 추가하기

구글에서 검색함 -> qt에 c library

<https://codeday.me/ko/qa/20190307/19538.html>

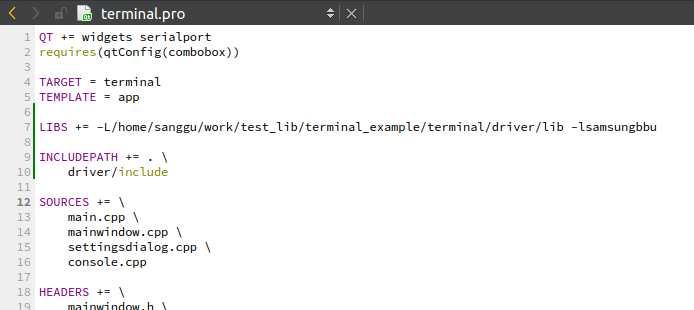
구글에서 검색함 -> c library 만들기

<https://swstar.tistory.com/96>

### Directory and source



### Terminal.pro



### Include terminal program



