Lab 1

Embedded Linux Development Environment
Multiple OS in a PC

- VMware

VMware Workstation Player - VMware Products

Try VMware Workstation Player

VMware Workstation Player builds on the industry leading foundation of Workstation Pro, and delivers a streamlined user interface for creating and running operating systems and applications in a virtual machine.

The free version is available for non-commercial, personal and home use. We also encourage students and non-profit organizations to benefit from this offering.

Commercial organizations require paid licenses to use Workstation Player.

Need a more advanced virtualization solution? Check out Workstation Pro for Windows or Workstation Pro for Linux.
Multiple OS in a PC

- Oracle Virtual Box

### VirtualBox

#### Download VirtualBox

Here, you will find links to VirtualBox binaries and its source code.

#### VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

- **VirtualBox 5.1.12 platform packages.** The binaries are released under the terms of the GPL version 2.
  - Windows hosts
  - OS X hosts
  - Linux distributions
  - Solaris hosts

- **VirtualBox 5.1.12 Oracle VM VirtualBox Extension Pack** All supported platforms
  Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP, disk encryption, NVMe and PXE boot for Intel cards
  The Extension Pack binaries are released under the VirtualBox Personal Use and Evaluation License (PUEL).

  Please install the extension pack with the same version as your installed version of VirtualBox.
  If you are using **VirtualBox 5.0.30**, please download the extension pack here.

- **VirtualBox 5.1.12 Software Developer Kit (SDK)** All platforms
- Debian
- Ubuntu
- Linux Mint
- Redhat
- CentOS
- Fedora
Development Environment

Host: Windows PC
- VMware player
- Linux Host

Target: Embedded Linux
- Embedded Linux kernel
- Application program
- Booting
- Flash Memory

Ethernet
- 192.168.0.2
- 192.168.0.120
- Bridged

Serial Port

192.168.0.60
USB to Ethernet Adapter
이더넷 9 속성

네트워킹

연결에 사용할 장치:
ASIX AX88772 USB2.0 to Fast Ethernet Adapter #4

이 연결에 다음 항목 사용(오):
- Microsoft Networks용 클라이언트
- VMware Bridge Protocol
- Microsoft 네트워크용 파일 및 프린터 공유
- QoS 패킷 스케줄러
- Reliable Multicast Protocol
- 인터넷 프로토콜 버전 4(TCP/IPV4)
- Microsoft 네트워크 어댑터 멀티플렉서 프로토콜
- Microsoft USB 프로토콜 드라이버

설정(N)...  제거(U)  속성(R)

설명
Transmission Control Protocol/인터넷 프로토콜입니다. 기본적인 광역 네트워크 프로토콜로, 다양한 연결된 네트워크에서 통신을 제공합니다.
네트워크가 IP 자동 설정 기능을 지원하면 IP 설정이 자동으로 할당되도록 할 수 있습니다. 지원하지 않으면, 네트워크 관리자에게 적절한 IP 설정값을 문의해야 합니다.

○ 자동으로 IP 주소 받기 (O)
○ 다음 IP 주소 사용 (S):
  
  IP 주소 (I): 192.168.0.2
  서브넷 마스크 (U): 255.255.255.0
  기본 게이트웨이 (D): . . .

○ 자동으로 DNS 서버 주소 받기 (B)
○ 다음 DNS 서버 주소 사용 (E):
  기본 설정 DNS 서버 (P): . . .
  보조 DNS 서버 (A): . . .

☐ 끝날 때 설정 유효성 검사 (L)

[그림: 네트워크 설정 화면]
VMware Player
VMware Player
VMware Player

- Select **Ubuntu1204_64** and play
- Or double click **Ubuntu1204_64.vmx** in the folder D:/Ubuntu1204_64
- ID: control password:
- ID: root password: control
- 마우스 포인터가 안 보일 경우에는 Ctrl-Alt
Open Terminal

- **pwd**: print working directory
- **ls**: list directory contents
New Terminal

- Right button click
Check IP address

- `ifconfig`
Open serial terminal

- Open **Windows SmarTTY**
Turn ON power switch
Power on and boot the target

```
Starting logging: OK
Populating /dev using udev: [ 5.238703] udevd[1831]: starting version 182
done
Initializing random number generator... done.
Starting system message bus: done
Starting network...
[ 5.745413] eth0: Freescale FEC PHY driver [Generic PHY] (mii_bus:phy_addr=1:01, irq=-1)
[ 5.755049] ADDRCONF(NETDEV_UP): eth0: link is not ready
Starting Network Interface Plugging Daemon: eth0.
welcome to Achro i.MX6Q
[ 5.848091] flexcan imx6q-flexcan.0: writing ctrl=0x01232004
[ 6.038596] ERROR: v4l2 capture: slave not found!
[ 6.038605] ERROR: v4l2 capture: slave not found!
[ 8.743605] PHY: 1:01 - Link is Up - 100/Full
[ 8.748568] ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
mount: mounting 192.168.0.120:/nfsroot on /mnt/nfs failed: No route to host
[root@ACHEL ~]#
```
Virtual Machine Network Setting

- Virtual machine settings
**Network Adapter**

**Virtual Machine Settings**

<table>
<thead>
<tr>
<th>Device</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>1 GB</td>
</tr>
<tr>
<td>Processors</td>
<td>1</td>
</tr>
<tr>
<td>Hard Disk (SCSI)</td>
<td>30 GB</td>
</tr>
<tr>
<td>CD/DVD (SATA)</td>
<td>Auto detect</td>
</tr>
<tr>
<td>Network Adapter</td>
<td>Bridged (Automatic)</td>
</tr>
<tr>
<td>USB Controller</td>
<td>Present</td>
</tr>
<tr>
<td>Sound Card</td>
<td>Auto detect</td>
</tr>
<tr>
<td>Printer</td>
<td>Present</td>
</tr>
<tr>
<td>Display</td>
<td>Auto detect</td>
</tr>
</tbody>
</table>

**Device status**
- Connected
- Connect at power on

**Network connection**
- Bridged: Connected directly to the physical network
  - Replicate physical network connection state
  - Configure Adapters
- NAT: Used to share the host’s IP address
- Host-only: A private network shared with the host
- Custom: Specific virtual network
  - VMnet0
- LAN segment:

**OK** | **Cancel** | **Help**
Configure Adapters

Virtual Machine Settings

Automatic Bridging Settings

Select the host network adapter(s) you want to automatically bridge:

- [x] ASIX AX88772 USB2.0 to Fast Ethernet Adapter #4
- [ ] Realtek PCIe GBE Family Controller
- [ ] Qualcomm Atheros QCA9377 Wireless Network Adapter
- [ ] Bluetooth Device (Personal Area Network)

Device status
- [x] Connected
- [x] Connect at power on

Network connection
- [ ] Bridged: Connected directly to the physical network
- [x] Replicate physical network connection state
- [ ] NAT: Used to share the host's IP address
- Check if USB Ethernet adapter is connected to HOST!
System Settings
IP addresses

- Windows: 192.168.0.2
- Ubuntu1204: 192.168.0.120
- Target: 192.168.0.60
IP addresses

![Network configuration interface showing IP addresses](image.png)

- **Hardware Address**: 00:50:FC:BC:E6:74
- **IP Address**: 192.168.0.120
- **Subnet Mask**: 255.255.255.0
- **Default Route**: 192.168.0.1
- **DNS**: 8.8.8.8

**Editing Wired connection 1**
- **Connection name**: Wired connection 1
- **Method**: Manual
- **Addresses**:
  - Address: 192.168.0.120, Netmask: 255.255.255.0, Gateway: 192.168.0.1
  - DNS servers: 8.8.8.8
Check target IP address
Check host-target connection

```bash
control@lab-pc2:~$ ping 192.168.0.60
PING 192.168.0.60 (192.168.0.60) 56(84) bytes of data.
64 bytes from 192.168.0.60: icmp_req=1 ttl=64 time=0.603 ms
64 bytes from 192.168.0.60: icmp_req=2 ttl=64 time=0.311 ms
64 bytes from 192.168.0.60: icmp_req=3 ttl=64 time=1.20 ms
^C
--- 192.168.0.60 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.311/0.706/1.205/0.372 ms
control@lab-pc2:~$ 
```

```bash
[root@ACHRO ~]# ping 192.168.0.120
PING 192.168.0.120 (192.168.0.120): 56 data bytes
64 bytes from 192.168.0.120: seq=0 ttl=64 time=0.607 ms
64 bytes from 192.168.0.120: seq=1 ttl=64 time=0.488 ms
64 bytes from 192.168.0.120: seq=2 ttl=64 time=0.434 ms
64 bytes from 192.168.0.120: seq=3 ttl=64 time=0.448 ms

--- 192.168.0.120 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/mdev = 0.434/0.494/0.607 ms
[root@ACHRO ~]# 
```
Make working directory

- `mkdir`: make directory
- `cd`: change directory
- `vi hello.c`
Press ‘a’ key to start to enter
Start typing
To finish typing press ‘esc’ key

```c
#include <stdio.h>

void main(void)
{
    printf("Hello\n");
}
```

"hello.c" [New File]
To save and quit press `:wq` and Enter
Compile and run

- gcc hello.c
- arm-none-linux-gnueabi-gcc hello.c
Mount NFS (network file system)

- `mount -t nfs 192.168.0.120:/nfsroot /mnt/nfs -o rw,rsize=4096,nolock`

```
[root@ACHRO ~]# cd /etc/init.d
[root@ACHRO init.d]# cat rcS
```

```
ip link set can0 type can bitrate 1000000
ifconfig can0 up
mount -t nfs 192.168.0.120:/nfsroot /mnt/nfs -o rw,rsize=4096,nolock
telnetd
telnetd
[root@ACHRO init.d]# cd /mnt/nfs
[root@ACHRO nfs]# ls
    test.txt
[root@ACHRO nfs]#
```
/nfsroot
Run hello.c in target
Run `hello.c` in target
Run hello.c in target
Run hello.c in target
Run `hello.c` in target
```make
all: app install

app: hello.c
    arm-none-linux-gnueabi-gcc -o hello hello.c

install:
    cp hello /nfsroot
```
- ‘vi hello.c’ to start editing
- Move the cursor using arrow keys or ‘h,j,k,l’ keys
- Press ‘i’ to insert
- To finish typing press ‘esc’ key
- To save ‘:w’ and Enter
- To quit ‘:q’ and Enter
- To delete a character press ‘x’
- To append a character ‘a’
- To copy a line ‘yy’
- To paste a line ‘p’
- To delete and copy a line ‘dd’
- To delete and copy multiple lines ‘#dd’, for example ‘5dd’ to delete and copy 5 lines
- To insert a line ‘o’
- To cancel the last action ‘u’
- Line number `:set nu`
- Replace a pattern `:%s/old/new/`
- Quit without saving `:q!`
- To copy from line number #1 to line number #2 to the next line of #3 `:`#1,#2co#3`
- To move from #1 to #2 to the next line of #3 `:`#1,#2m#3`
- To delete from #1 to #2 `:`#1,#2d`
Text Editor
```c
#include <stdio.h>

void main(void)
{
    printf("Hello World\n");
}
```
Basic Linux Commands

- **cd**: change directory
- **mkdir**: make a new directory
- **rmdir**: remove a directory, Directory including files or sub-directories cannot be deleted
- **cp**: copy one or more files to another location
- **rm**: remove files and directories
- **ls**: list files
Workspace Switcher
Shut Down

- Do not just close the VMware window
Exercise

- Hello World를 10번 프린트 하는 프로그램 을 vi 에디터를 사용하여 편집한 후, linux host와 target에서 각각 실행해 보십시오.
File Sharing with Windows

- Edit virtual machine settings
Virtual Machine Settings

![Virtual Machine Settings in VMware Workstation 12 Player](image)
Folder sharing

Shared folders expose your files to programs in the virtual machine. This may put your computer and your data at risk. Only enable shared folders if you trust the virtual machine with your data.

- Disabled
- Always enabled
- Enabled until next power off or suspend

Folders

<table>
<thead>
<tr>
<th>Name</th>
<th>Host Path</th>
</tr>
</thead>
</table>

Add...  Remove  Properties
Folder sharing

Welcome to the Add Shared Folder Wizard

This wizard will guide you through the steps of adding a new shared folder to your virtual machine.
Folder sharing
Folder sharing

Add Shared Folder Wizard

Name the Shared Folder
What would you like to call this shared folder?

Host path
D: \W

Name
D
Folder sharing

The image shows a Virtual Machine Settings window with an Add Shared Folder Wizard. The wizard is used to specify shared folder attributes, including:

- **Enable this share**
- **Read-only**

These attributes control the scope of the shared folder and whether access is allowed for read or read-write operations.
Folder sharing

Shared folders expose your files to programs in the virtual machine. This may put your computer and your data at risk. Only enable shared folders if you trust the virtual machine with your data.

- Disabled
- Always enabled
- Enabled until next power off or suspend

Folders

<table>
<thead>
<tr>
<th>Name</th>
<th>Host Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>D:\</td>
</tr>
<tr>
<td>E</td>
<td>E:\</td>
</tr>
</tbody>
</table>
Folder sharing

```
control@lab-pc2:~/mnt/hgfs
control@lab-pc2:~$ cd /mnt
control@lab-pc2:/mnt$ ls
hgfs
control@lab-pc2:/mnt$ cd hgfs
control@lab-pc2:/mnt/hgfs$ ls
D   E
control@lab-pc2:/mnt/hgfs$
```
USB memory

A USB device is about to be unplugged from the host and connected to this virtual machine. It will first be stopped to enable safe removal. With some devices, the host may display the message "The device can now safely be removed."

- Do not show this message again

OK  Cancel
USB memory