# ZedBoard Lab 0
# Environment Setting

Chun-Chen Tu
timtu@umich.edu

# How to make your ZedBoard works?

- Non-OS environment : Xilinx EDK
  - Xilinx XPS
  - Xilinx SDK
- OS environment:
  - Xilinx XPS, SDK
  - Unix-based environment for compiling kernel, driver … etc.
- This slide contains instruction of setting up Unix-based environment and cross compiling tools.

# Outline

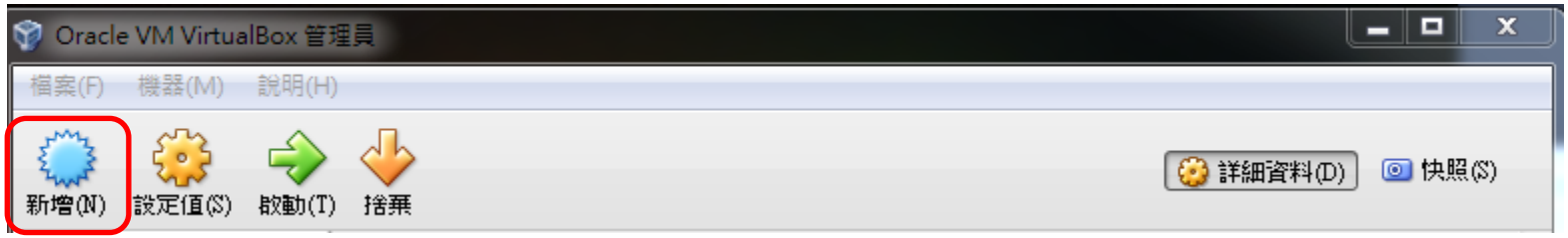
- VirtualBox
  - Install and set up a guest machine.
  - Shared folder, NFS.
- Cross Compiling toolchain

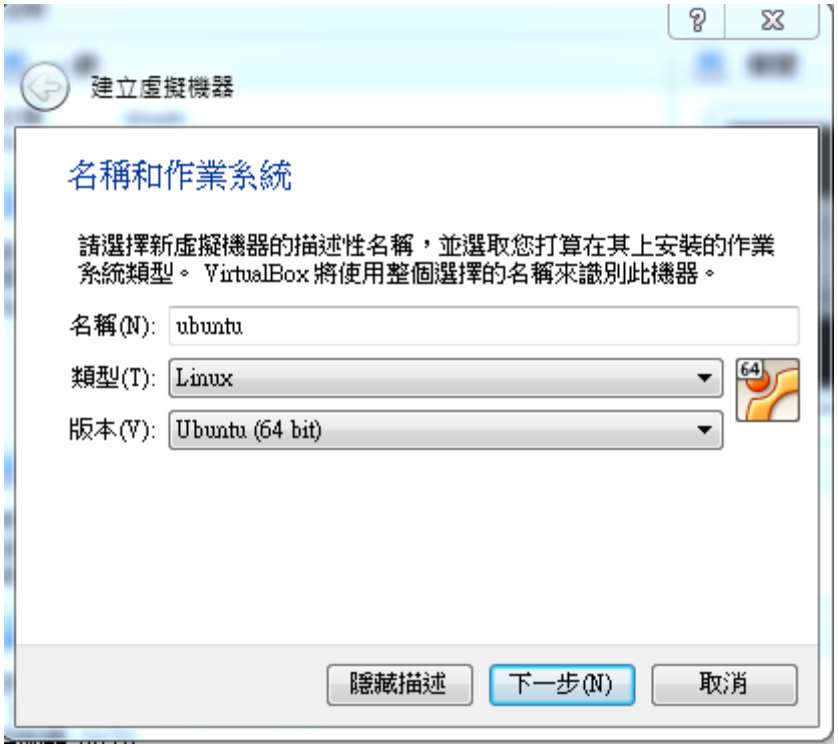
# Install VirtualBox

- https://www.virtualbox.org/wiki/Downloads
- Download and install VirtualBox.

# Create new guest machine

- We use Ubuntu 12.04 server version.
  - Download the image through: http://www.ubuntu.com/download
- Activate VirtualBox. And create a guest machine.



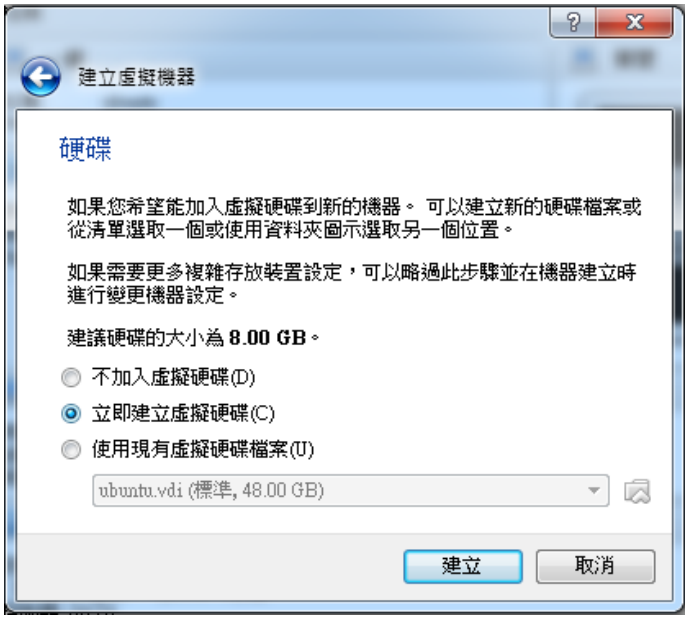


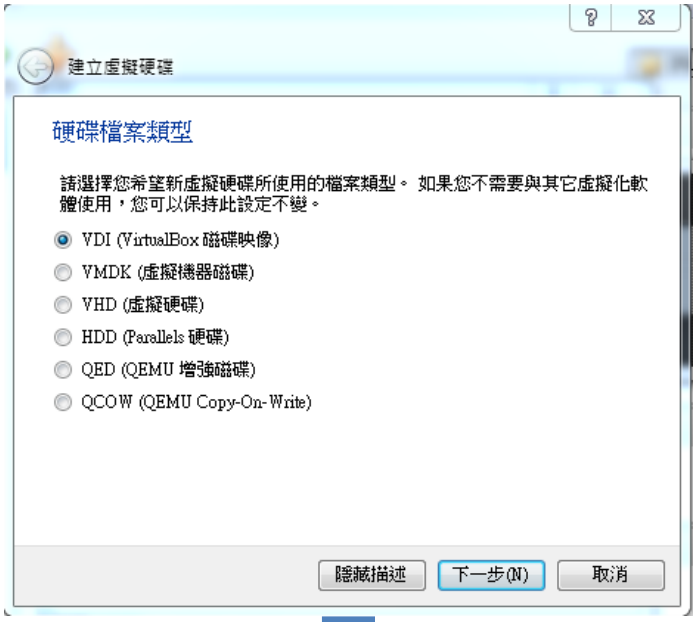

Enter your machine name and select the right configuration.

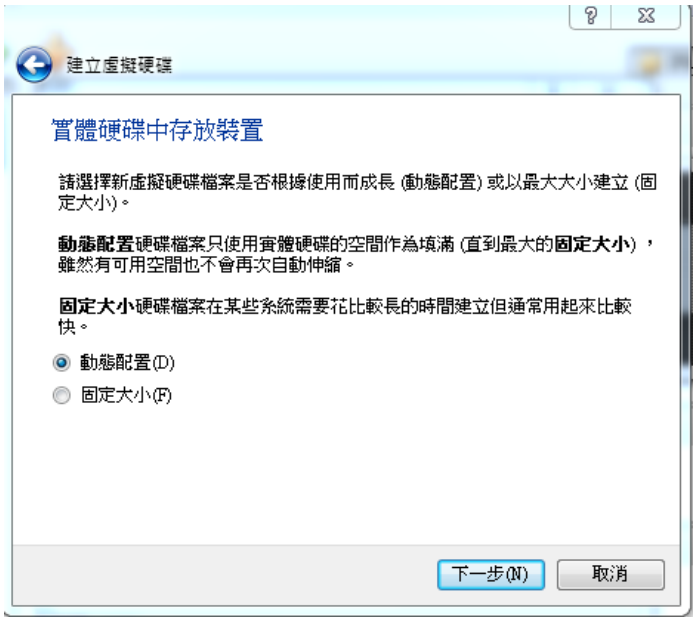
(Note: sometimes you cannot find 64-bit option. It is related to your CPU's. Check the virtualization related options in the BIOS setting.)


Allocate the memory. Better larger than 1GB

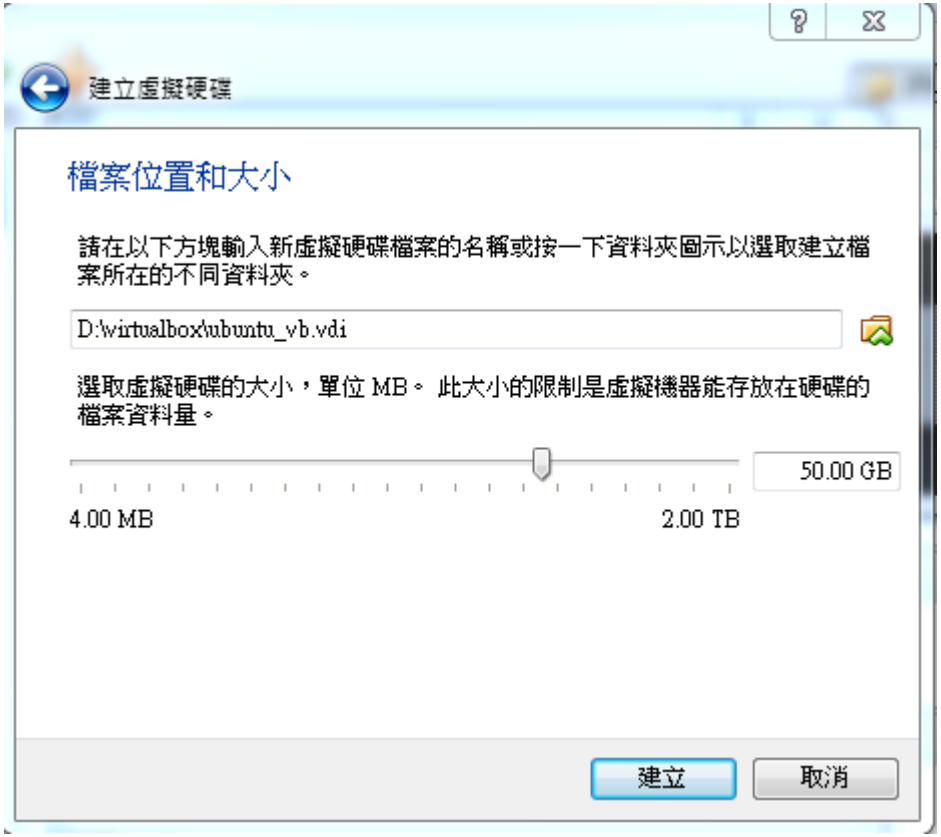
(Note: The allocated memory will be occupied as you activated VirtualBox. Carefully adjust the figure.)




Use the default, Next!




Choose the place to store your Ubuntu files.

30GB will be enough.


Double click the machine you just created.

Select the Ubuntu image you just downloaded.

The installation will start. For more detail, please refer to the slide of **Ubuntu server 12.04 installation.**

# VirtualBox Settings

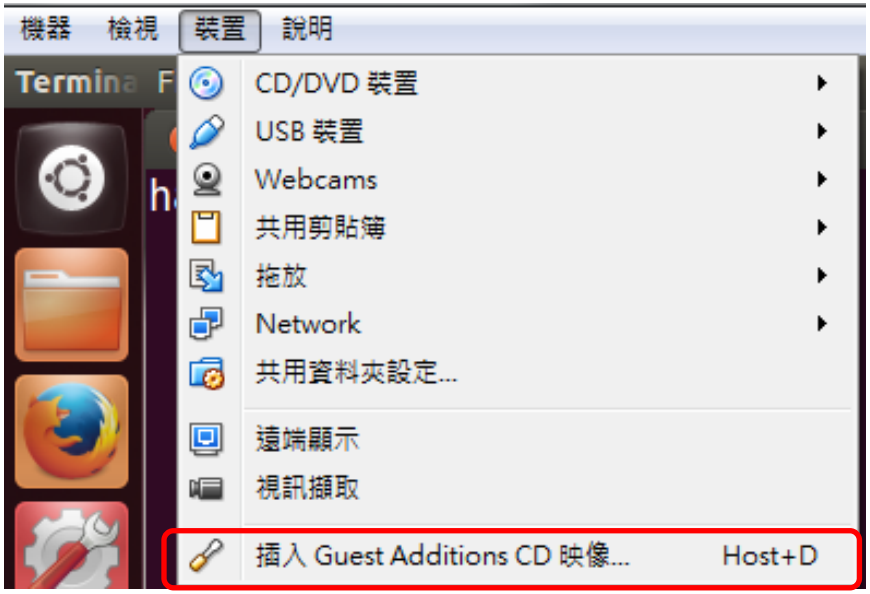
- VirtualBox Guest Additions
  - Provide multiple display resolution options.
  - Shared folder with host and guest machines
- Use the devices of USB
- NFS settings
- Cross Compile toolchain

# VirtualBox Guest Additions

For the first booting, you will find you cannot adjust the display resolution. Thus, you need Guest Additions.

Boot up the machine and select **Install Guest Additions**. And follow the step to install it.

# VirtualBox Guest Additions




Insert Guest Additions. The content will automatically pop up. If not, type

sudo mount –t iso9660 /dev/cdrom /media/cdrom

cd /media/cdrom

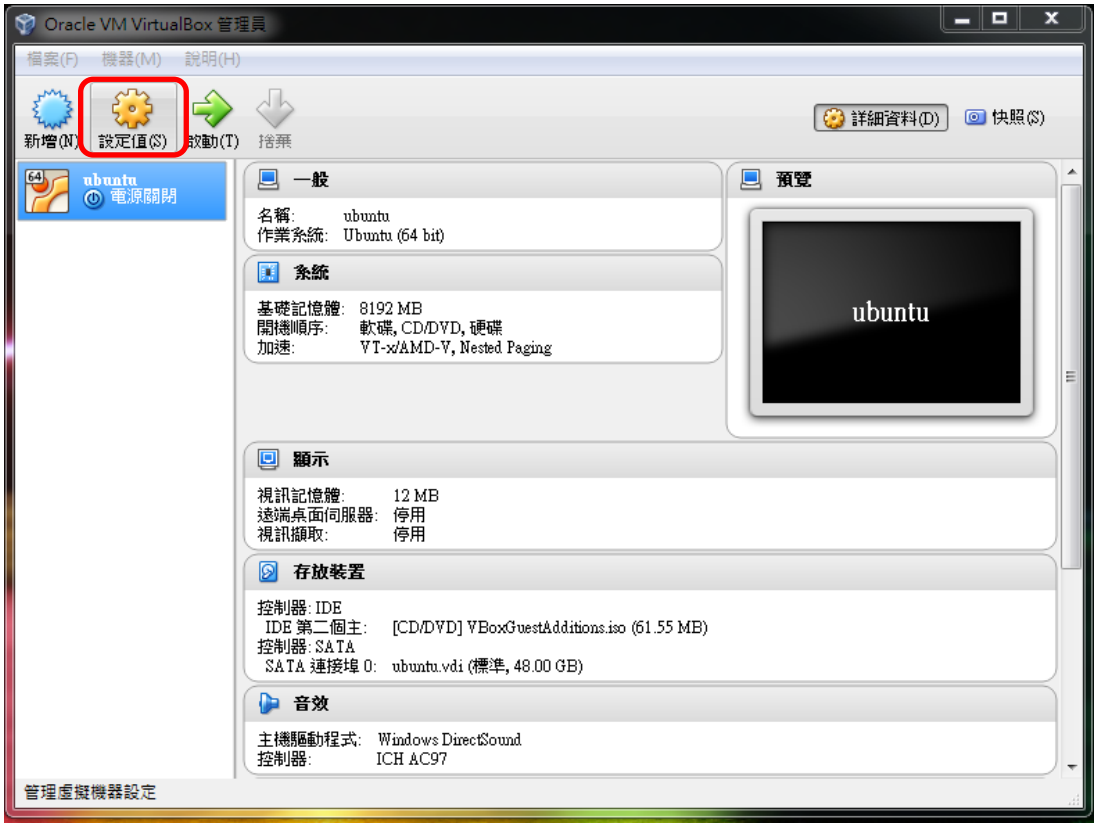
sudo ./VBoxLinuxAdditions.run

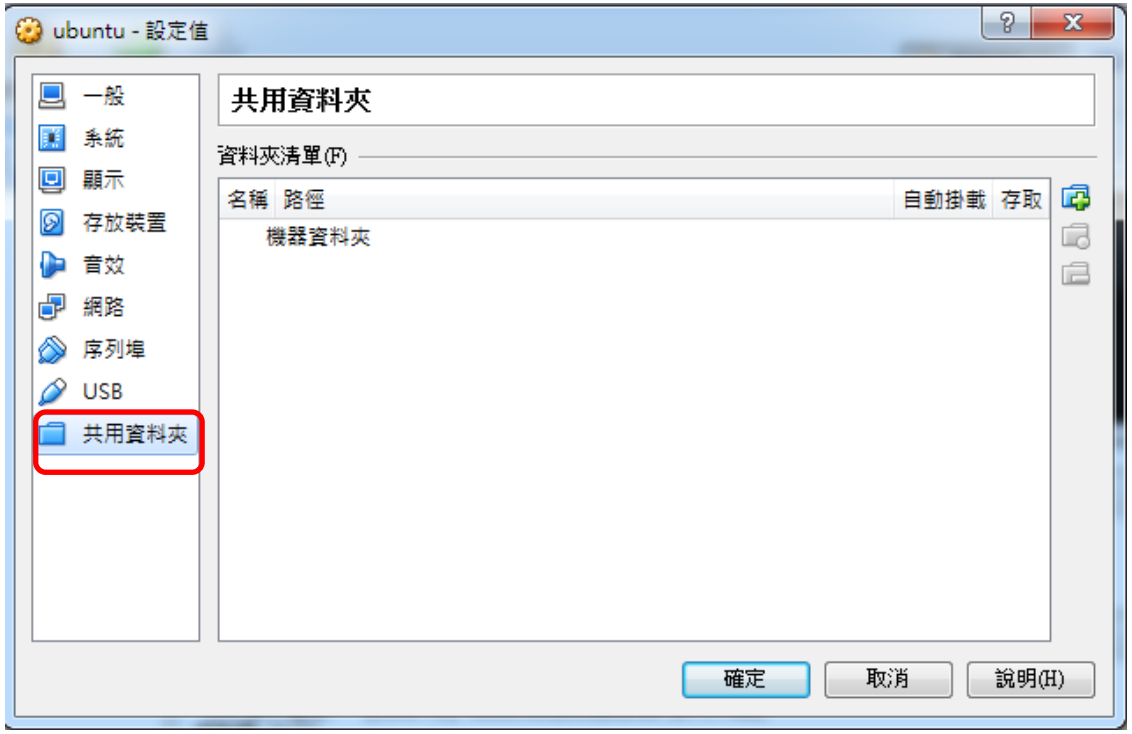
And then restart the machine.

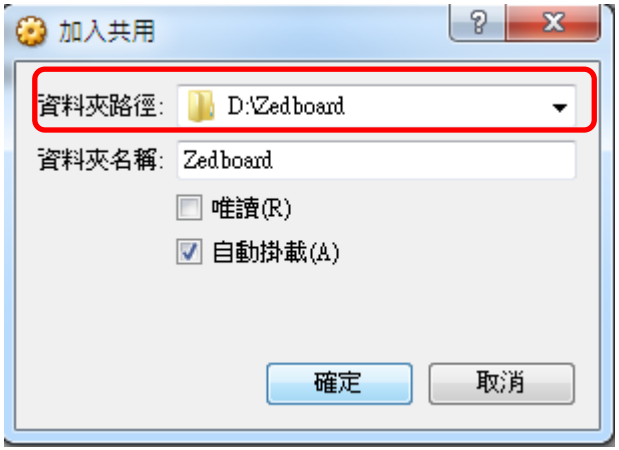
# Shared folders

In many cases, you need to exchange files between host and guest machine. Share folder will make it easy.

Close you guest machine and click on settings.






Click on the Shared folder tab and set the folder on host you want to share with guest.

After booting your guest machine, mount the shared folder.

sudo mount –t vboxsf Zedboard ~/win

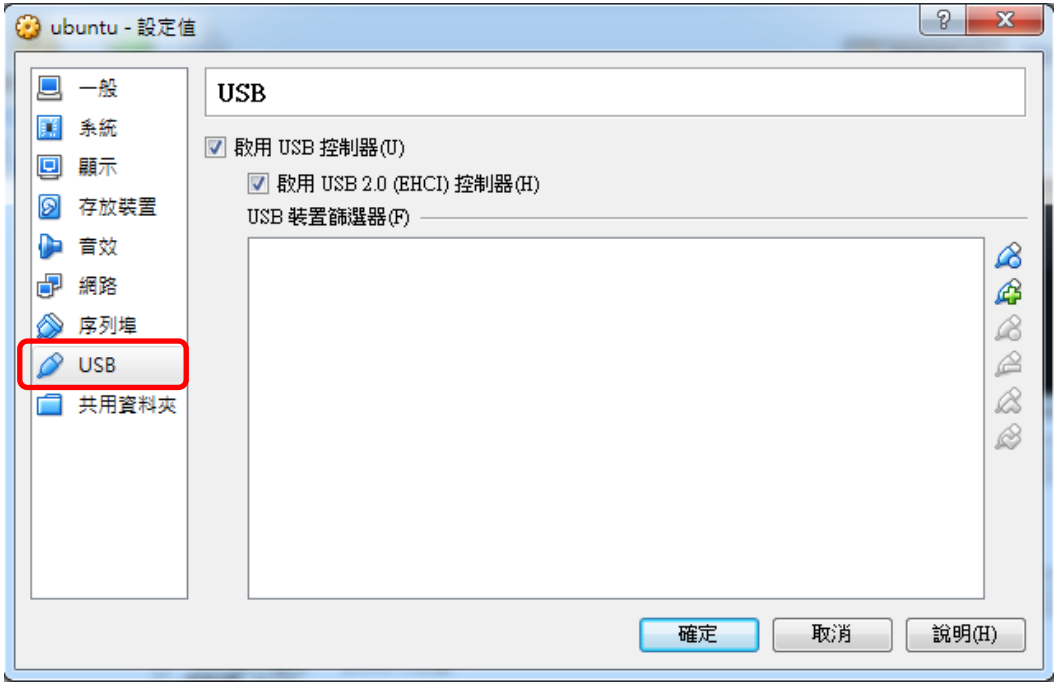
You can now access to the files.

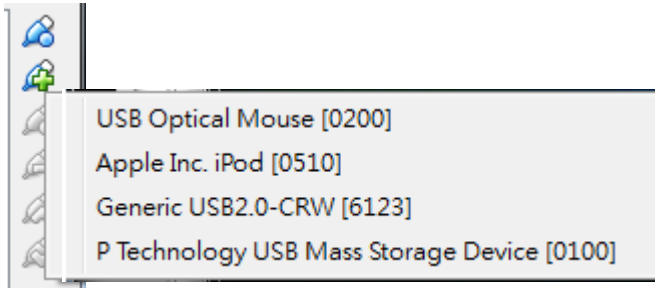
check this if you cannot mount shared folder.

# USB devices

Guest machines cannot access to the USB devices at the beginning. For example, if you plug in an USB storage, the guest machine is not able to access data from your storage. You need to make some configuration.

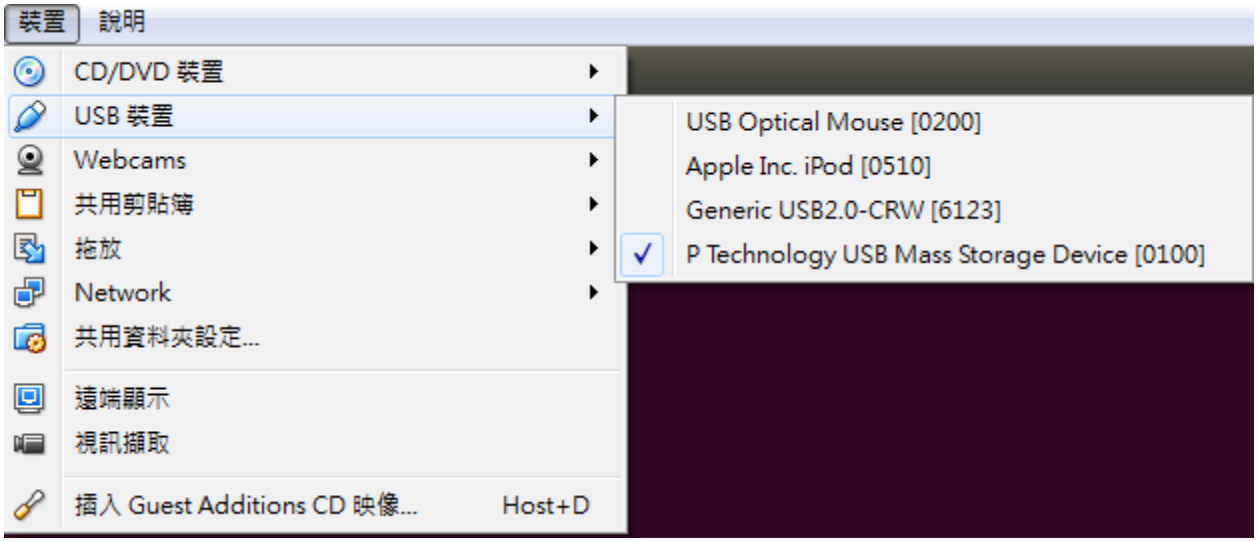
Again, shutdown your guest machine and click on settings and then the USB tab.







Click on the USB with plus icon and add the device to the filter. Make sure it is checked.




As you boot up, you can check that the USB storage is activate. Once your host machine want to access the USB storage, just click on the device and your host machine can recognize the USB device.

# NFS server

This is not necessary for developing on ZedBoard. But with NFS, exchanging between machines will become more easier.

Setting up NFS server is easy. However, since our Ubuntu runs on VirtualBox, there are some additional setting for NFS.

Install the NFS server packages.

sudo apt-get install nfs-kernel-server

Create a folder for NFS(my username is hadoop)

mkdir /home/hadoop/NFS

NFS configuration

sudo vim /etc/exports

/home/hadoop/NFS *(rw,sync,no_root_squash,insecure)

Note that the insecure option is not necessary. But sometimes the NFS server will complain about the security issue and will result in fail mounting. Try this option if you cannot mount NFS successfully.

Restart NFS

sudo /etc/init.d/nfs-kernel-server restart

sudo /etc/init.d/portmap restart

Test

mkdir test

sudo mount 10.0.2.15:/home/hadoop/NFS test

# NFS with fixed port

By default, NFS server will use random port for NFS service. But we need a fixed port for VirtualBox for port forwarding. Thus we need to modify some configuration.

sudo vim /etc/default/nfs-common

STATDOPTS="-p 32765 –o 32766"

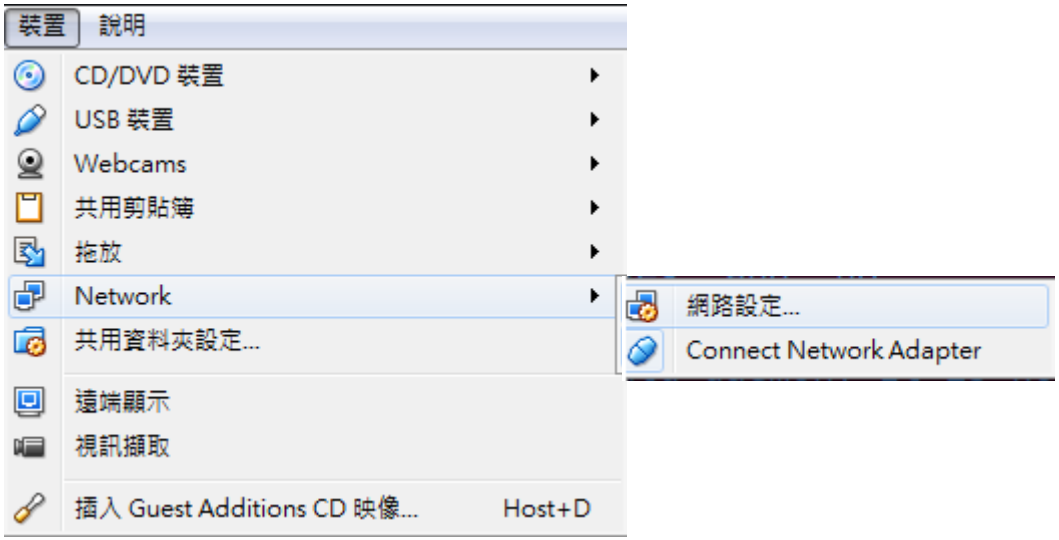
sudo vim /etc/default/nfs-kernel-server

RPCMOUNTDOPTS="-p 32767"

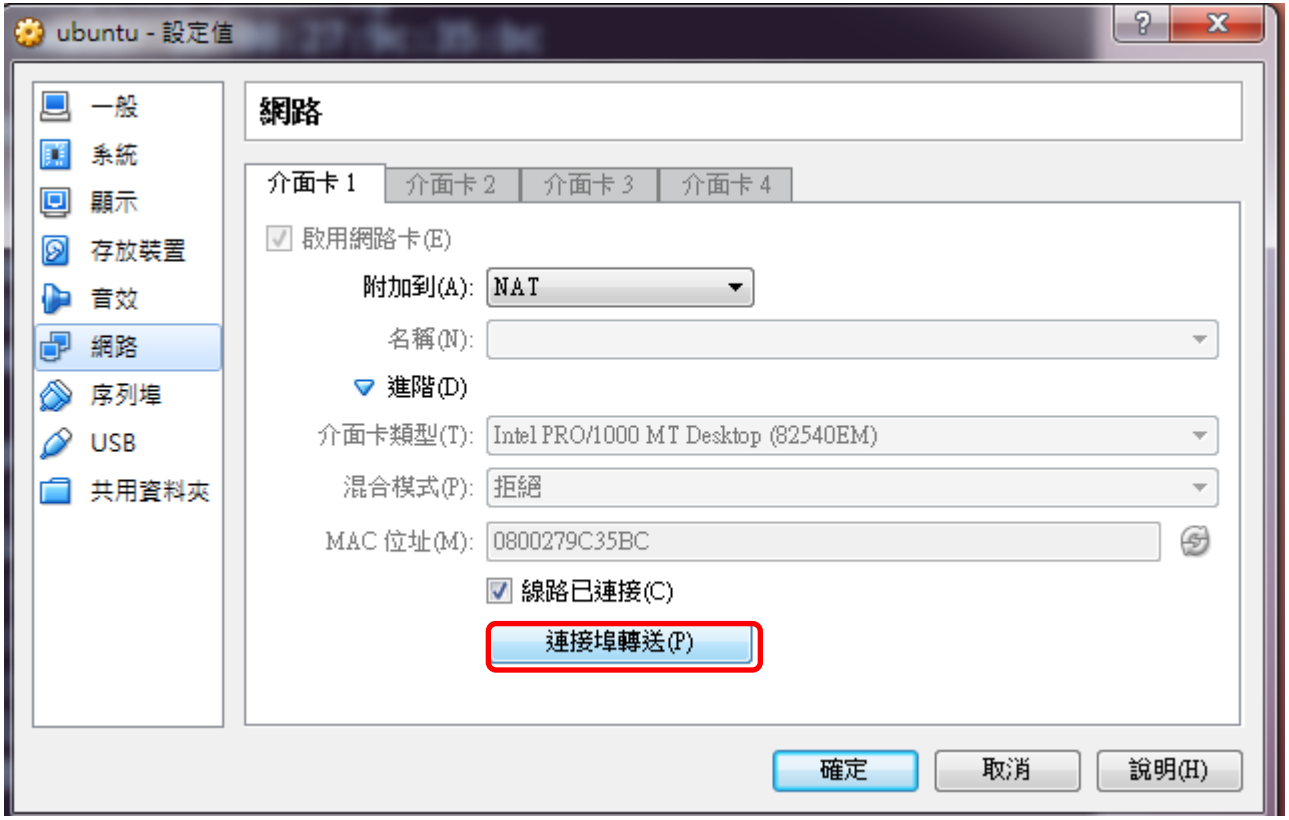
# Port forwarding of VirtualBox

```
eth0      Link encap:Ethernet  HWaddr 08:00:27:9c:35:bc
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9c:35bc/64 Scope:Link
```

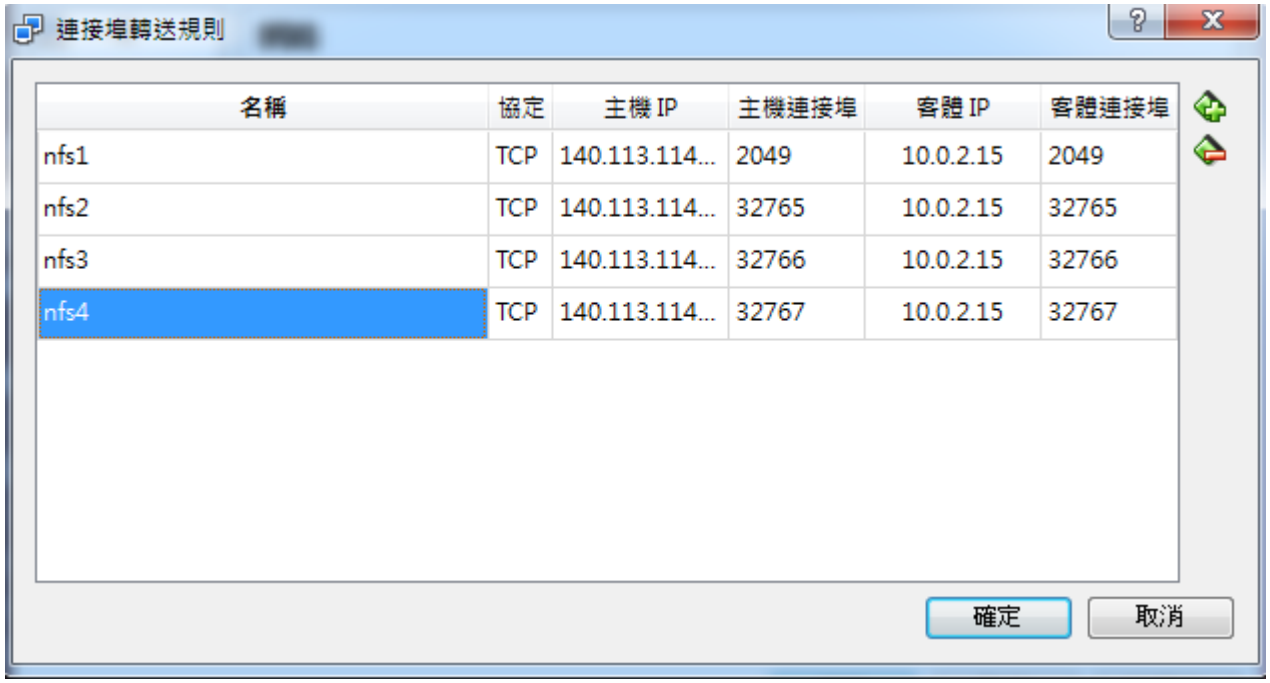
Check your guest ip: 10.0.2.15




Open the network setting page.




Click port forwarding.



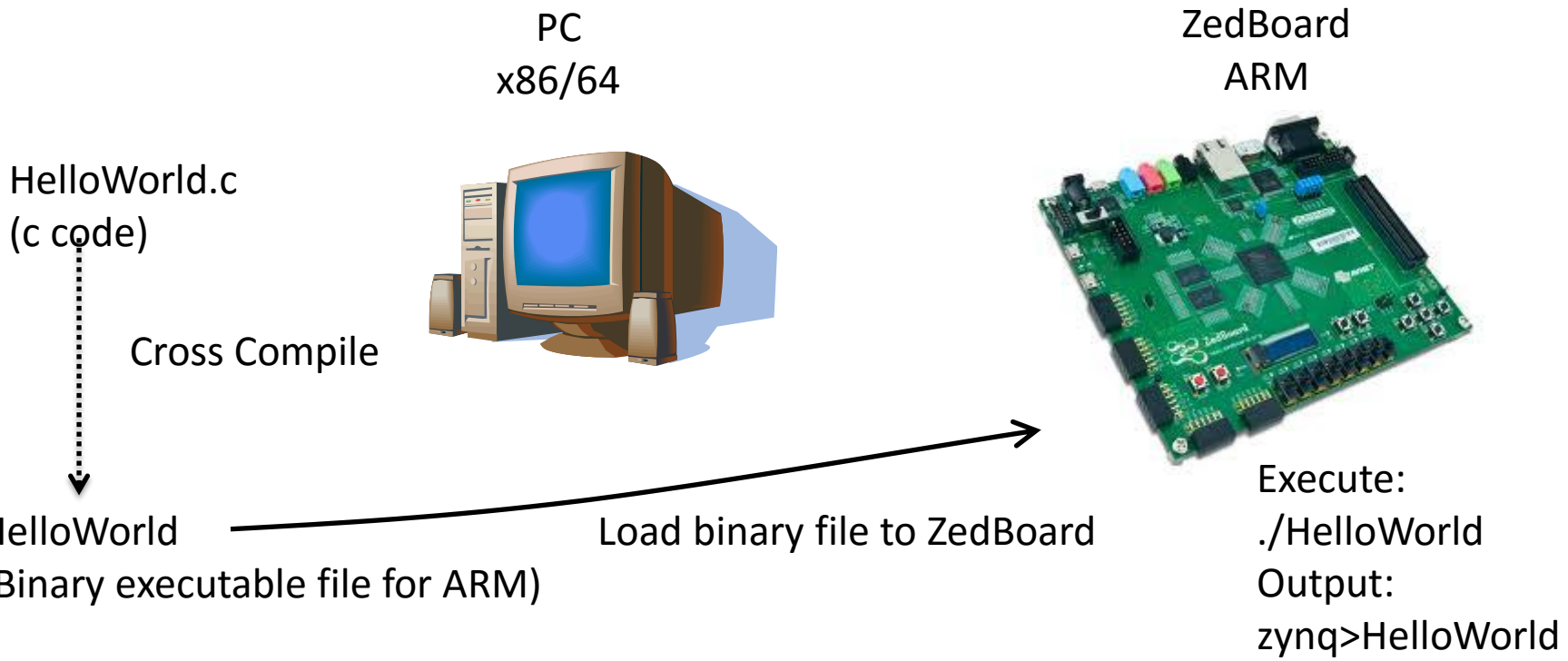

Add port forwarding rules.

Test it by

sudo mount –t nfs –o nolock,proto=tcp,port=2049,mountport=32767 <*IP*>:/home/hadoop/NFS temp

# Cross Compile Toolchain

- Why do we need this?
  - ZedBoard adopts ARM architecture. If we want our problems run on ZedBoard, we need compiler targeting on ARM architecture.



# Fetch and set toolchain

- You can find the toolchain in the CD under directory CodeSourcery or FTP ARM_CrossCompile.tar.gz
  - You can install toolchain automatically using CD. The installation will automatically help you set up environment variable.
  - If you use the file on the FTP, you should set up environment manually. ( I have modify the folder name to make it shorter)

    tar xvf CrossCompile.tar.gz

    mv CrossCompile ~/

    vim ~/.bashrc

    export PATH=$PATH:/home/hadoop/CrossCompile/bin

    export CROSS_COMPILE=arm-xilinx-linux-gnueabi-

    source ~/.bashrc

  Note: The cross-compiler is with 32-bit format. For 64-bit system to recognize it you need to install 32-bit library by:

    sudo apt-get install ia32-libs