

---

# **Android Guide**

## **For MVC100**

---



**MicroVision Co., Ltd.**

## Document Information

Version	1.0
File Name	MVC100_Android_Guide.doc
Date	2010. 2. 5
Satus	Working

## Revision History

Date	Version	Update Descriptions	Editor
2010. 2. 5.	V1.0	First Edition	Jongill Wee

Copyright © 2007 MicroVision Co.,Ltd. All rights reserved.

Published by MicroVision Co.,Ltd.

(☎) +82-2-3283-0101, (✉) sale@microvision.co.kr

<http://www.microvision.co.kr>, <http://www.mvtool.co.kr>

Room #610, Hanshin IT Tower 235, Guro3-dong, Guro-gu, Seoul, Korea.

# Contents

1. Environment 4/54
  - 1.1 Installing ubuntu 8.10 5/53
  - 1.2 Environment for development 9/53
    - 1.2.1 The way to login ROOT 9/53
    - 1.2.2 Installing package and update 9/53
    - 1.2.3 To setup Samba 10/53
  - 1.3 Installing Java 5 13/53
2. To setup GCC 14/53
  - 2.1 Unzip 14/53
3. To setup Bootloader 15/53
  - 3.1. Compile 15/53
4. To setup Kernel 17/53
  - 4.1. Compile 17/53
5. To Compile for Cupcake 20/53
6. Download 22/53
7. The way setup ADB for Application 36/53

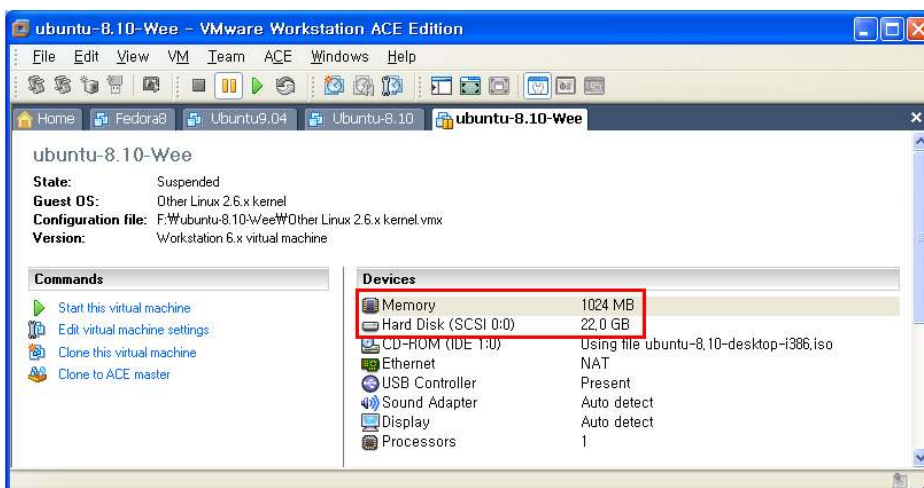
# 1. Environment

File	Name	Version
mvcl00_android_uboot.tar.gz	Bootloader	1.3.4
mvcl00_android_kernel.tar.gz	Kernel	2.6.27
C100_cupcake_RTM10_w_lib.tar.bz2	File System	1.5
4.3.1-eabi-armv6.tar.gz	GCC Compiler	4.3.1

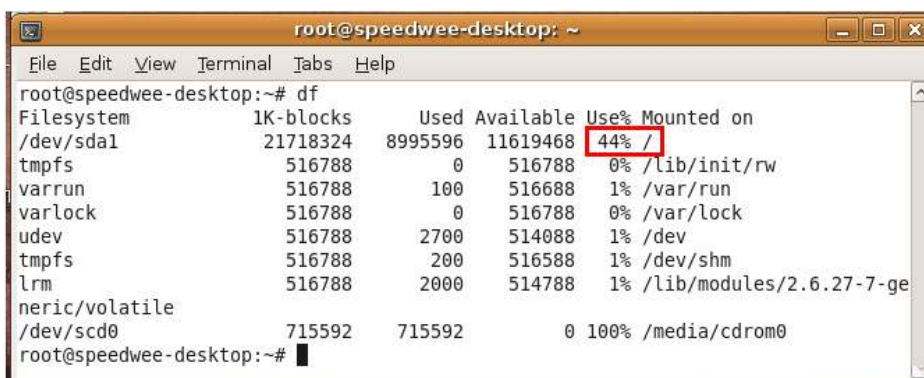
I tested base of ubuntu 8.10(Linux). (You have to use ubuntu 8.10)

You can download address <http://ftp.kaist.ac.kr/ubuntu-cd/8.10/>.

If you will install Linux through VMware, you would setup SDRAM Memory capacity 1024MB, HDD a minimum 20GB because **cupcake** has big capacity compiling than you thought.



Under picture, HDD is left capacity after compile, that was made HDD 22GB

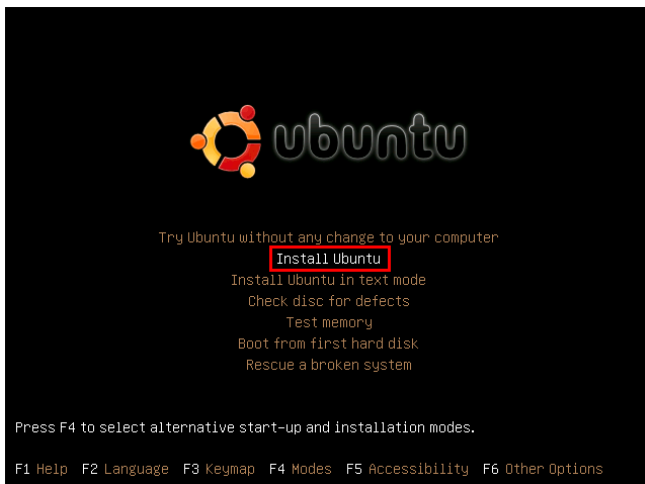


## 1.1 Installing ubuntu 8.10

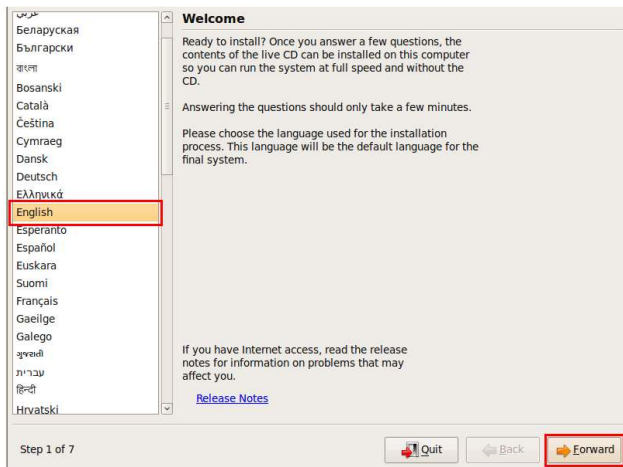
Select "Language"



Select "Install Ubuntu"



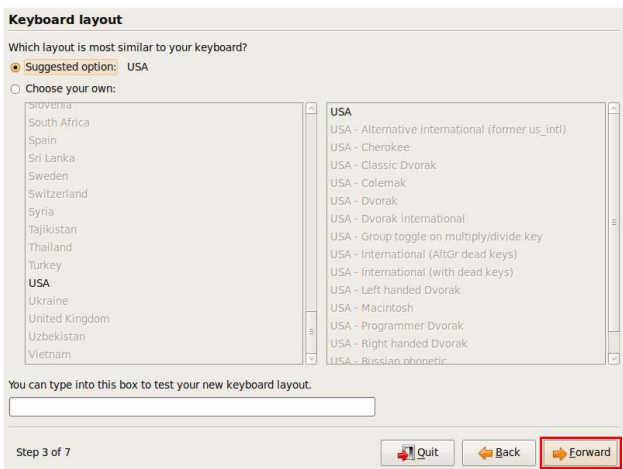
After select Language, click “Forward”



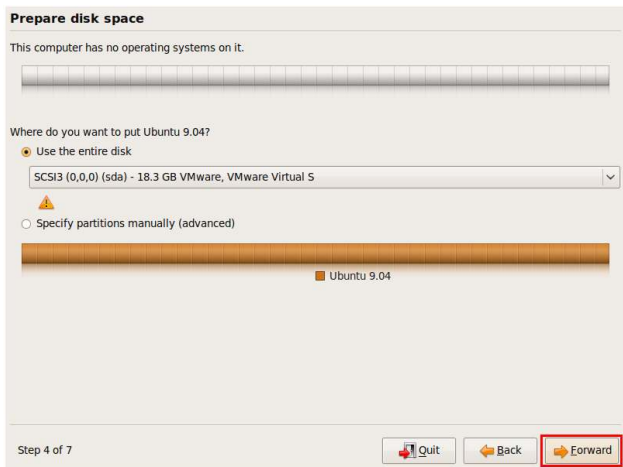
After select Time, click “Forward”



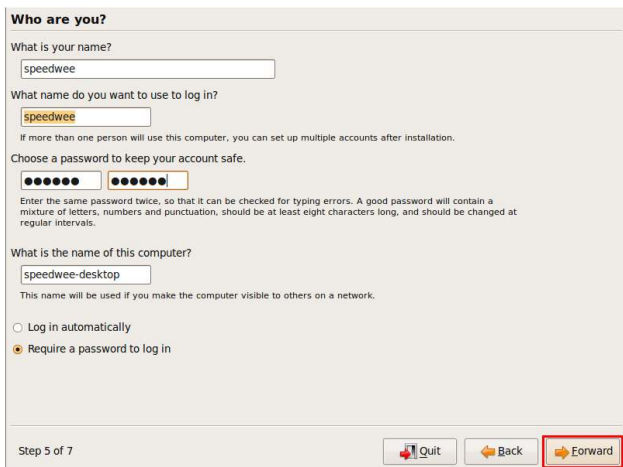
After select KeyBoard, select “Forward”



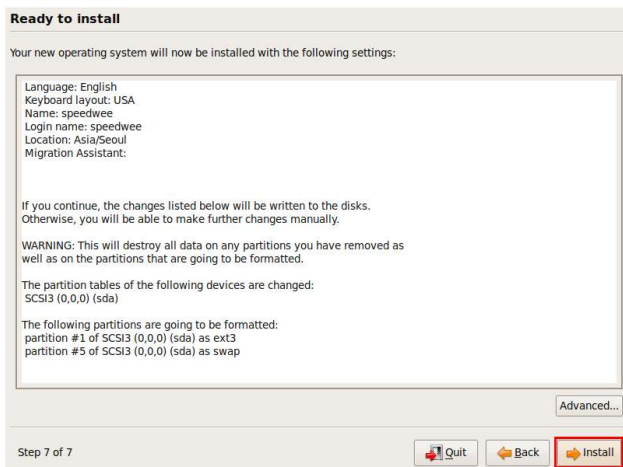
### Click "Forward"



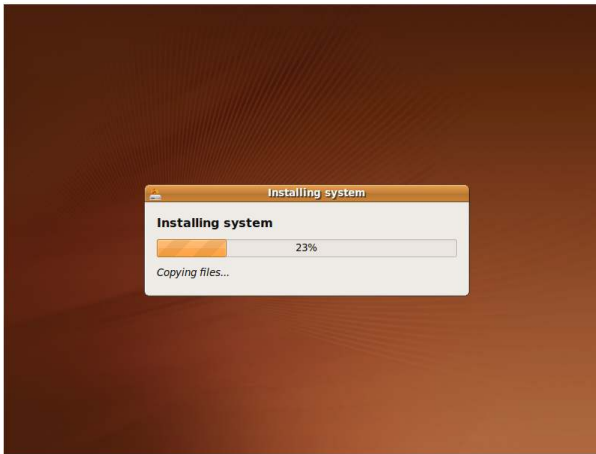
### After input ID and Password, select "Forward"



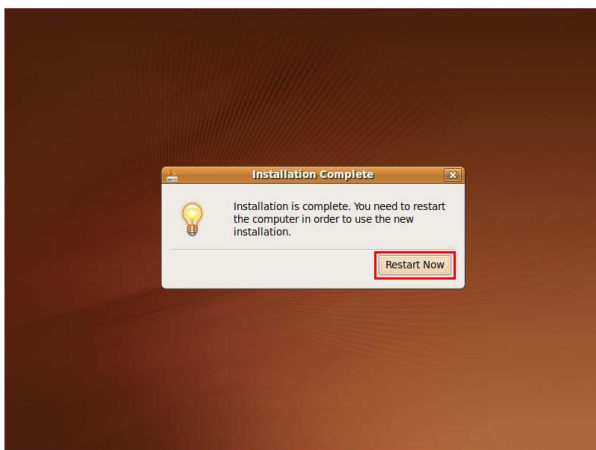
### Click "Install"



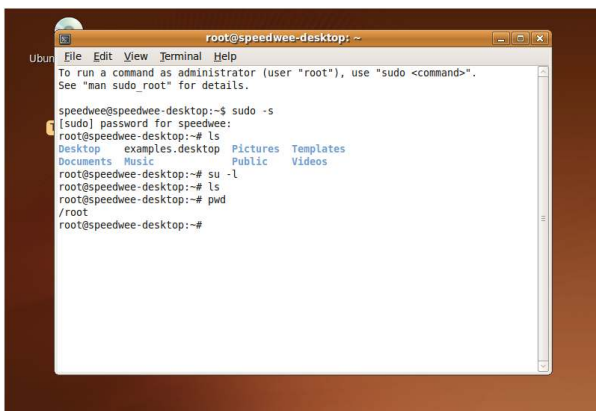
It is installing



Click "Restart Now"



It is that booted



## 1.2 Environment for development

### 1.2.1 The way to login ROOT

# sudo -s after run, fill out "Password"

A terminal window titled "root@speedwee-desktop: ~" is shown. The prompt is "speedwee@speedwee-desktop:~\$". The user enters "sudo -s", which is highlighted with a red box. The prompt changes to "[sudo] password for speedwee:". The user enters their password, and the prompt changes to "root@speedwee-desktop:~#". The user then enters "ls", and the terminal displays a directory listing: Desktop, Examples, mvc100, Public, Templates, ubuntu\_nfs.txt, Documents, Music, Pictures, sdk.txt, ubuntu\_env.txt, Videos. The prompt returns to "root@speedwee-desktop:~#".

```
root@speedwee-desktop: ~
File Edit View Terminal Tabs Help
speedwee@speedwee-desktop:~$ sudo -s
[sudo] password for speedwee:
root@speedwee-desktop:~# ls
Desktop  Examples  mvc100    Public  Templates  ubuntu_nfs.txt
Documents Music     Pictures  sdk.txt  ubuntu_env.txt  Videos
root@speedwee-desktop:~#
```

### 1.2.2 Installing package and update

```
# sudo apt-get install libqt3-headers libqt3-mt-dev build-essential
bin86 kernel-package libstdc++6
```

```
# sudo apt-get install python-dev python patch m4 make ccache perl
diffstat bitbake wget curl ftp cvs git subversion
```

```
# sudo apt-get install git tar bzip2 gzip unzip jade docbook docbook-utils
texinfo texi2html sec bison bc libc6-dev
```

```
# sudo apt-get upgrade
```

```
# sudo apt-get update
```

After installed, Rebooting you can move from Windows to VMware without "Ctrl + Alt"

### 1.2.3 To setup Samba

You felt uncomfortable installed VMware when you used TFTP, FTP. So you will setup Samba Server that is easy for you to copy files from windows to VMware.

Run for installing samba "sudo apt-get install samba"

```
# sudo apt-get install samba
```

Modify samba script

```
# sudo vi /etc/samba/smb.conf
```

```
[global]
```

```
workgroup = MVC100 <- modify 35 line
```

```
server string = %h server (For C100 Samba) <- modify 38 line
```

```
log file = /var/log/samba/log.%m <- modify 75 line
```

```
security = SHARE <- modify 99 line
```

```
encrypt password = yes <- add to 100 line
```

```
smb password file = /etc/samba/smbpasswd <- add to 101 line
```

```
client code page = 949 <- add to 102 line
```

```
encrypt passwords = yes <- modify 105 line
```

```
socket options = TCP_NODELAY SORCVUF=8192 SOSNDBUF=8192 <- modify 196  
line
```

```
[homes]
```

```
comment = Home Speedwee <- modify 238 line
```

```
browseable = yes <- modify 239 line
```

```
writable = yes <- add to 240 line
```

```
[public]
```

```
comment = MVC100 <- add to 263 line
```

```
path=/home/Speedwee <- add to 264 line
```

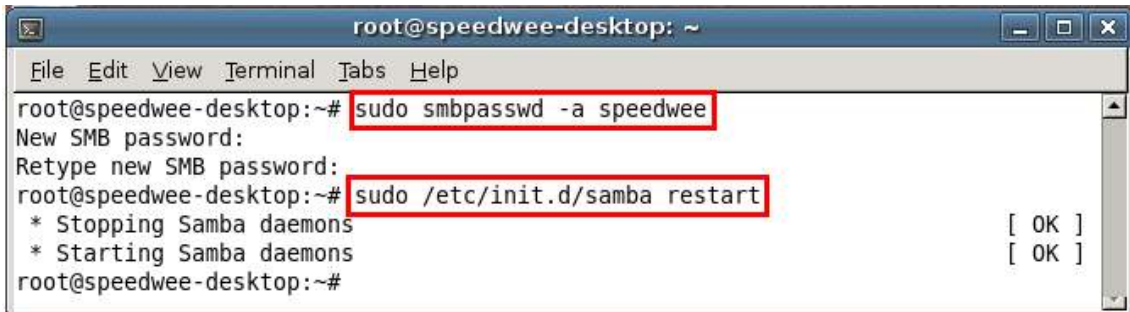
```
public = yes <- add to 265 line
```

```
writable = yes <- add to 66 line
```

After saved, run a command.

# `sudo smbpasswd -a "Your Wants name"`

# `sudo /etc/init.d/samba restart`



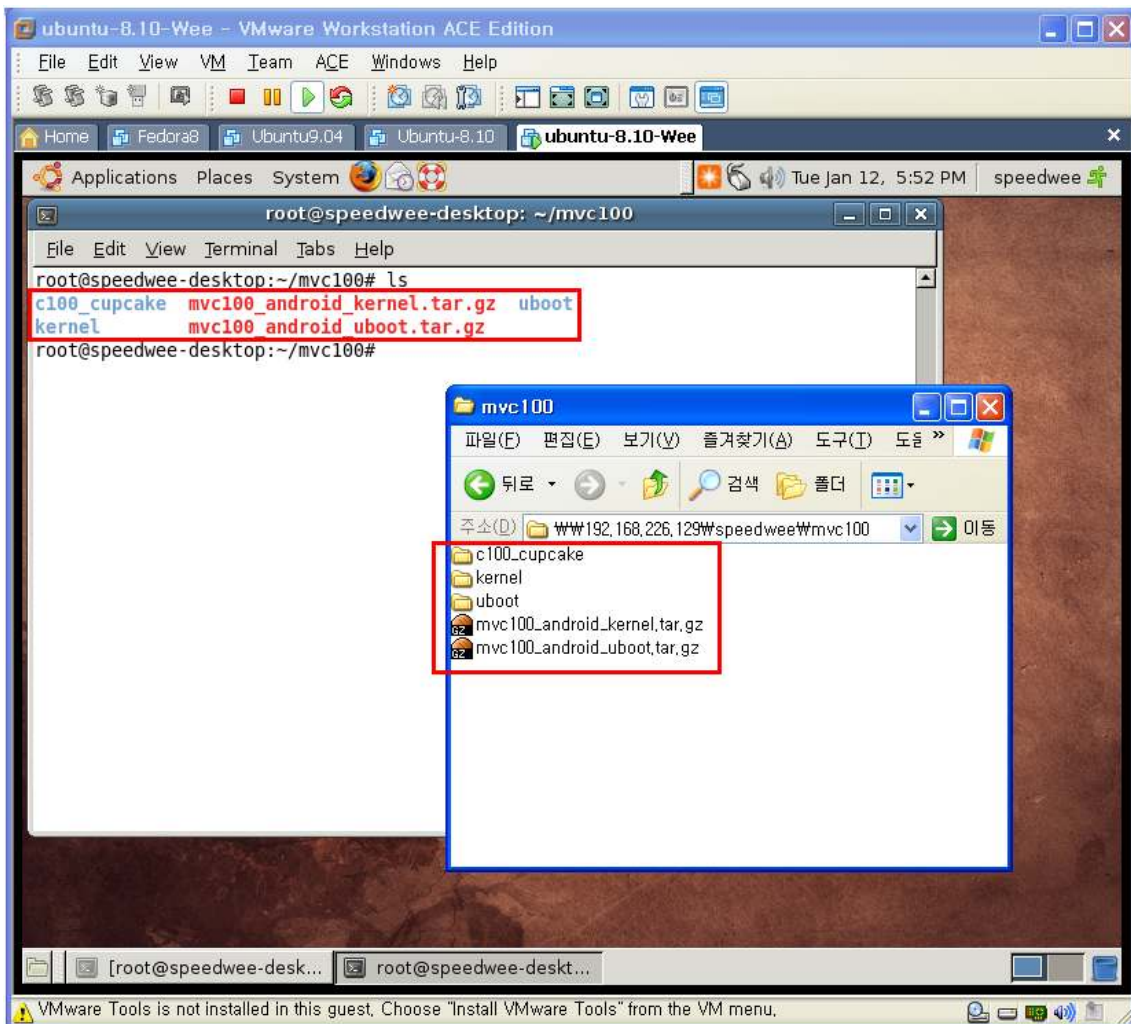
Fill in Linux IP address Windows on My Computer.

“`\\192.168.226.129\speedwee`”

“`\\xxx.xxx.xxx.xxx\WID`”



After filled, it appears new windows so fill in ID, Password.



It is share from linux to windows xp

Now you can copy files mouse dragging linux to windows xp

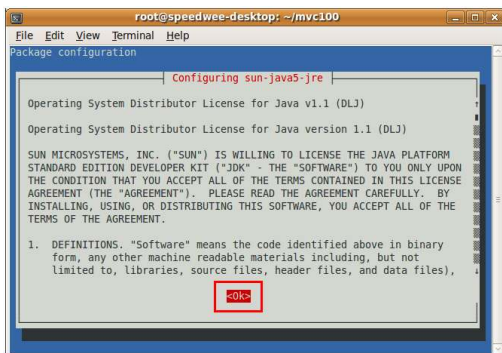
### 1.3 Installing Java 5

I recommend a JAVA5 because you installed JABA6. maybe error when compiling Cupcake.

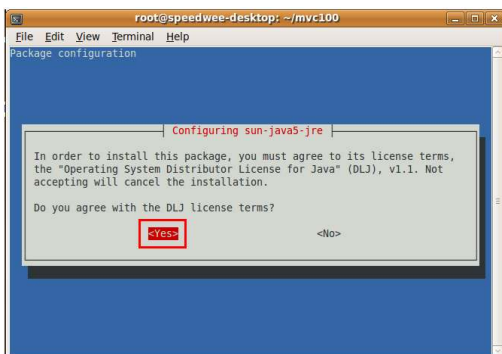
Install packages order for development Android.

```
# sudo apt-get install git-core gnupg sun-java5-jdk flex bison gperf  
libsdl-dev libesd0-dev libwxgtk2.6-dev build-essential zip curl  
libncurses5-dev zlibg-dev
```

Under picture appear, use "Tab" select "<Ok>".



Select "<Yes>".



```
# sudo apt-get install valgrind  
# apt-get install lib32readline5-dev  
# update-java-alternatives -s java-1.5.0-sun
```

## 2. To setup GCC

### 2.1 Unzip

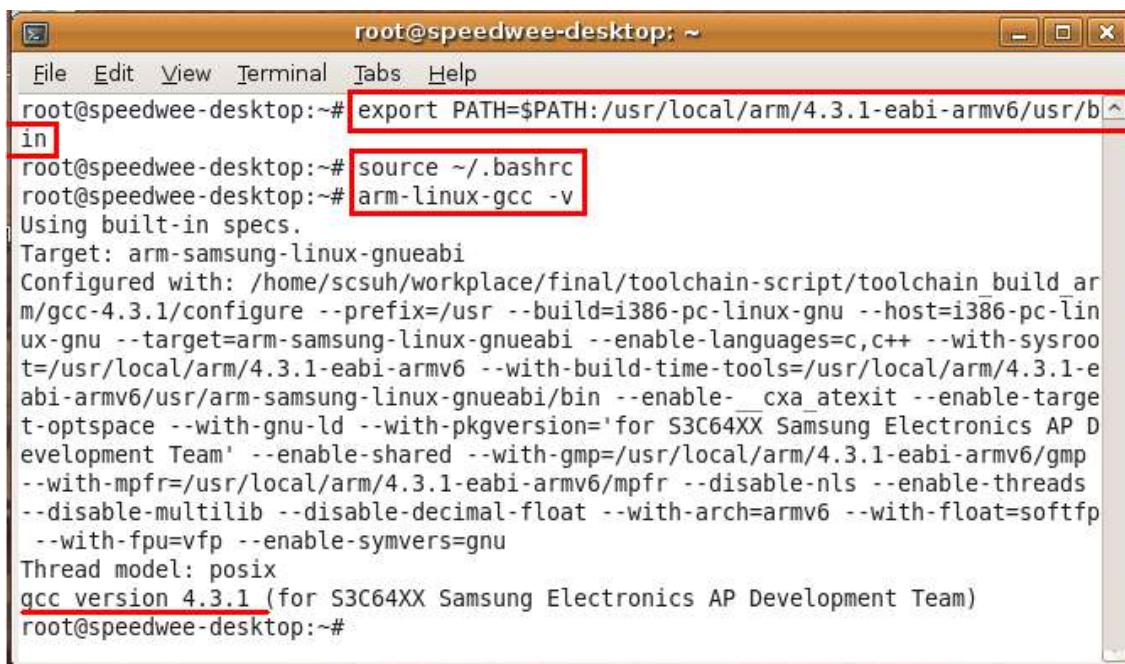
```
# mkdir /usr/local/arm
# mv 4.3.1-eabi-armv6.tar.bz2 /usr/local/arm
# cd /usr/local/arm
# tar jxvf 4.3.1-eabi-armv6.tar.bz2
```



```
root@speedwee-desktop: /usr/local/arm
File Edit View Terminal Tabs Help
root@speedwee-desktop:~/mvc100# mkdir /usr/local/arm
root@speedwee-desktop:~/mvc100# mv 4.3.1-eabi-armv6.tar.bz2 /usr/local/arm/
root@speedwee-desktop:~/mvc100# cd /usr/local/arm/
root@speedwee-desktop:~/local/arm# tar jxvf 4.3.1-eabi-armv6.tar.bz2
```

### Setup PATH

```
# cd
# export PATH=$PATH:/usr/local/arm/4.3.1-eabi-armv6/usr/bin
# source ~/.bashrc : Apply environment
# arm-linux-gcc -v : To see GCC Version
```



```
root@speedwee-desktop: ~
File Edit View Terminal Tabs Help
root@speedwee-desktop:~# export PATH=$PATH:/usr/local/arm/4.3.1-eabi-armv6/usr/b
in
root@speedwee-desktop:~# source ~/.bashrc
root@speedwee-desktop:~# arm-linux-gcc -v
Using built-in specs.
Target: arm-samsung-linux-gnueabi
Configured with: /home/scsuh/workplace/final/toolchain-script/toolchain build ar
m/gcc-4.3.1/configure --prefix=/usr --build=i386-pc-linux-gnu --host=i386-pc-lin
ux-gnu --target=arm-samsung-linux-gnueabi --enable-languages=c,c++ --with-sysroo
t=/usr/local/arm/4.3.1-eabi-armv6 --with-build-time-tools=/usr/local/arm/4.3.1-e
abi-armv6/usr/arm-samsung-linux-gnueabi/bin --enable-__cxa_atexit --enable-targe
t-optspace --with-gnu-ld --with-pkgversion='for S3C64XX Samsung Electronics AP D
evelopment Team' --enable-shared --with-gmp=/usr/local/arm/4.3.1-eabi-armv6/gmp
--with-mpfr=/usr/local/arm/4.3.1-eabi-armv6/mpfr --disable-nls --enable-threads
--disable-multilib --disable-decimal-float --with-arch=armv6 --with-float=softfp
--with-fpu=vfp --enable-symvers=gnu
Thread model: posix
gcc version 4.3.1 (for S3C64XX Samsung Electronics AP Development Team)
root@speedwee-desktop:~#
```

## 3. To setup Bootloader

### 3.1. Compile

Unzip as command.

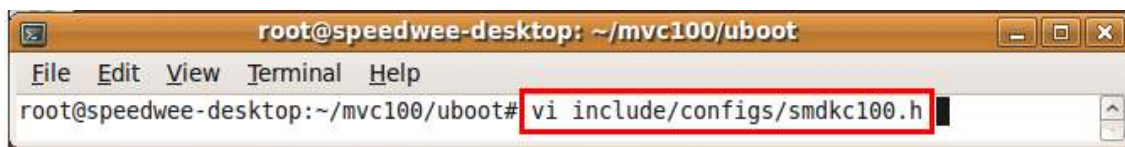
```
# tar xvf mvcl00_android_uboot.tar.gz
```



```
root@speedwee-desktop: ~/mvc100
File Edit View Terminal Help
root@speedwee-desktop:~/mvc100# tar xvf mvcl00_android_uboot.tar.gz
```

To setup IP

```
# vi include/configs/smdkc100.h
```



```
root@speedwee-desktop: ~/mvc100/uboot
File Edit View Terminal Help
root@speedwee-desktop:~/mvc100/uboot# vi include/configs/smdkc100.h
```

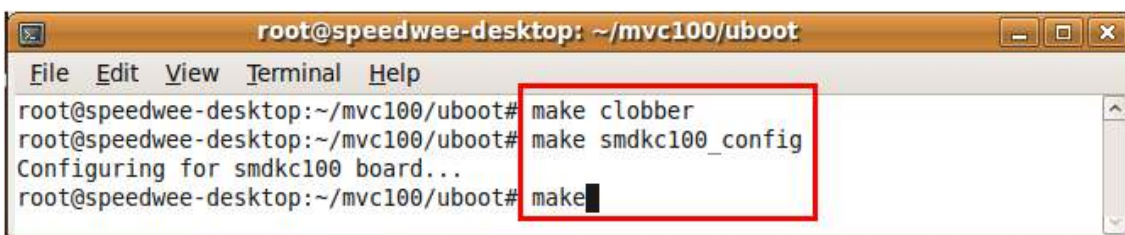
```
#define CONFIG_ETHADDR      MAC ADDRESS
#define CONFIG_NETMASK      SUBNETMASK
#define CONFIG_IPADDR       Target Board
#define CONFIG_SERVERIP     TFTP Server PC
```

The compile is order.

```
# make clobber
```

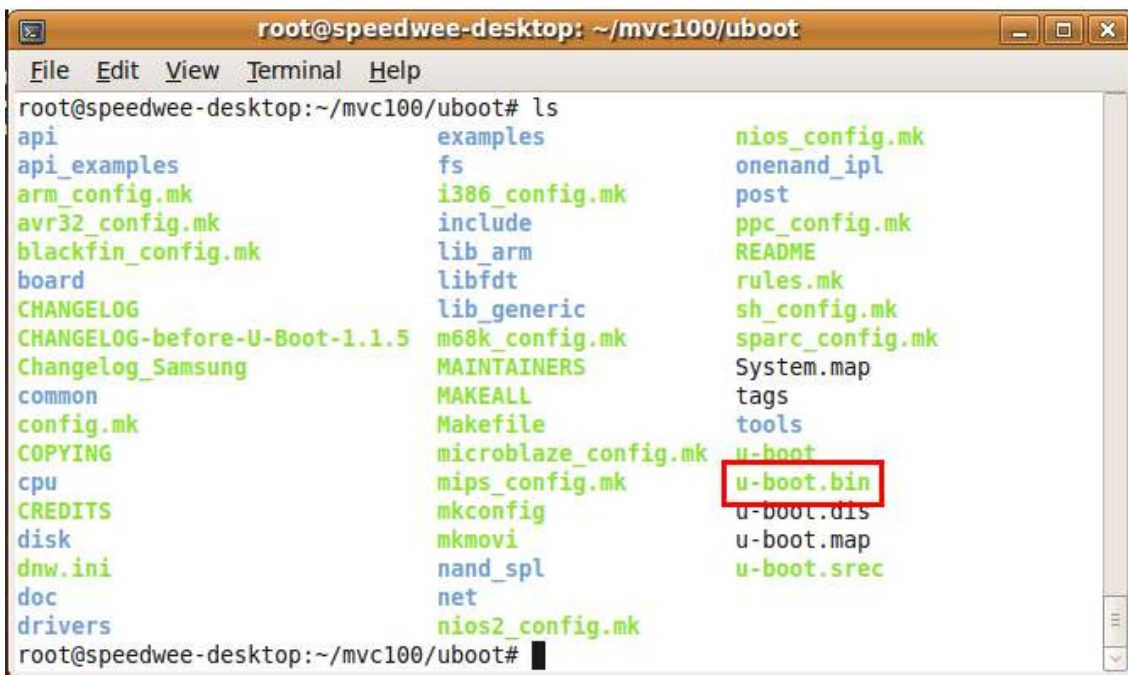
```
# make smdkc100_config
```

```
# make
```



```
root@speedwee-desktop: ~/mvc100/uboot
File Edit View Terminal Help
root@speedwee-desktop:~/mvc100/uboot# make clobber
root@speedwee-desktop:~/mvc100/uboot# make smdkc100_config
Configuring for smdkc100 board...
root@speedwee-desktop:~/mvc100/uboot# make
```

You can see “u-boot.bin” in /uboot.



## 4. To setup Kernel

### 4.1. Compile

Unzip as a command.

```
$ tar xvf mvcl00_android_kernel.tar.gz
```



As under picture, Run vi editor ".cross\_compile" and then modify GCC PATH.

```
# vi .cross_compile
```

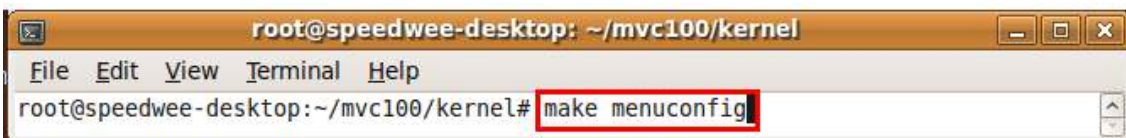


```
/usr/local/arm/4.3.1-eabi-armv6/usr/bin/arm-linux-
```

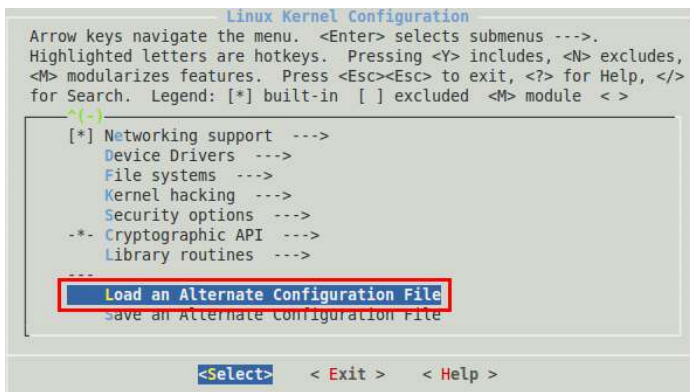
Save after made sure

Run a command for compiling.

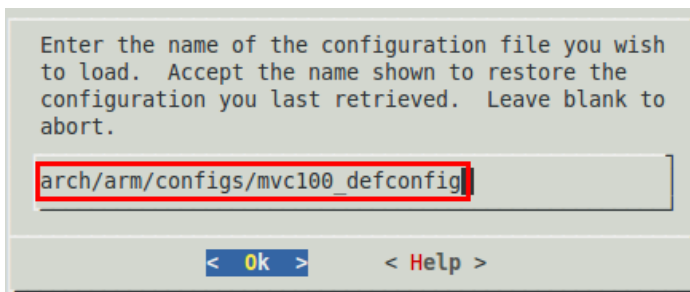
```
# make menuconfig
```



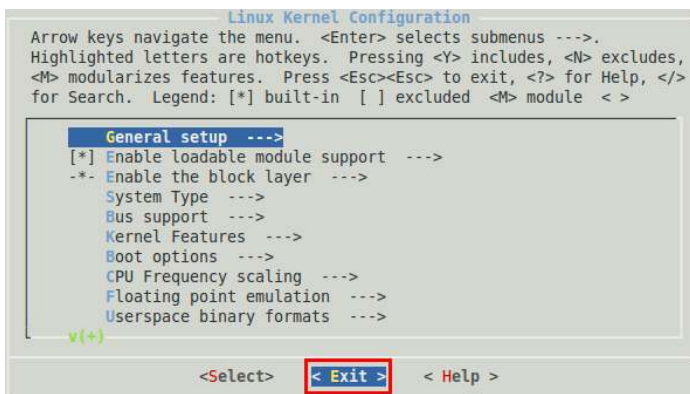
Select "Load an Alternate Configuration File".



Load "arch/arm/configs/mvc100\_defconfig".

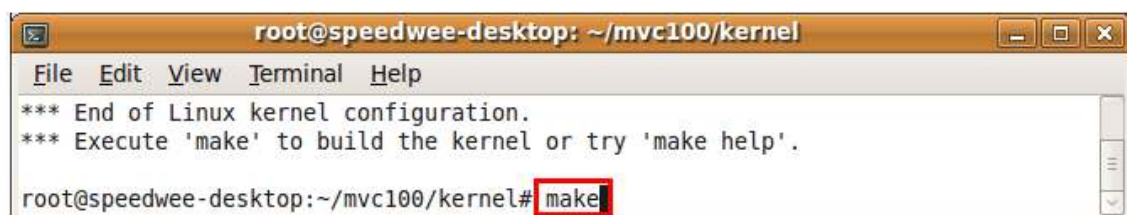


After make sure and then exit.

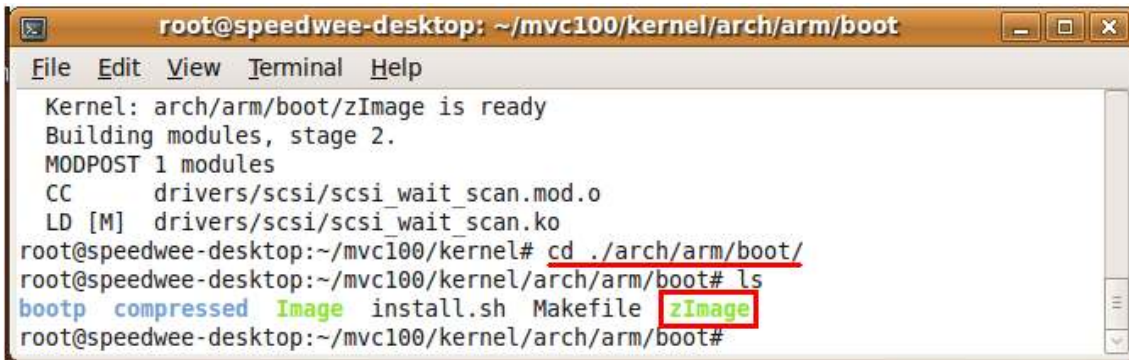


Run "make"

# make



You can see “zImage” in kernel/arch/arm/boot.



```
root@speedwee-desktop: ~/mvc100/kernel/arch/arm/boot
File Edit View Terminal Help
Kernel: arch/arm/boot/zImage is ready
Building modules, stage 2.
MODPOST 1 modules
CC      drivers/scsi/scsi_wait_scan.mod.o
LD [M]  drivers/scsi/scsi_wait_scan.ko
root@speedwee-desktop:~/mvc100/kernel# cd ./arch/arm/boot/
root@speedwee-desktop:~/mvc100/kernel/arch/arm/boot# ls
bootp compressed Image install.sh Makefile zImage
root@speedwee-desktop:~/mvc100/kernel/arch/arm/boot#
```

## 5. To Compile for Cupcake

Unzip as command.

```
# tar jxvf C100_cupcake_RT10_w_lib.tar.bz2
```



```
root@speedwee-desktop: ~/mvc100/c100_cupcake
File Edit View Terminal Tabs Help
root@speedwee-desktop:~/mvc100/c100_cupcake# tar jxvf C100_cupcake_RT10_w_lib.t
ar.bz2
```

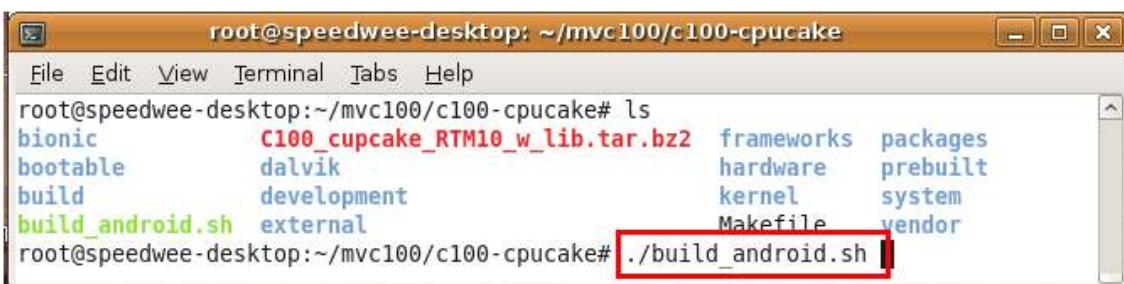
Make folder “/tftpboot” before compile.

```
# mkdir /tftpboot
```

```
# chmod 777 /tftpboot
```

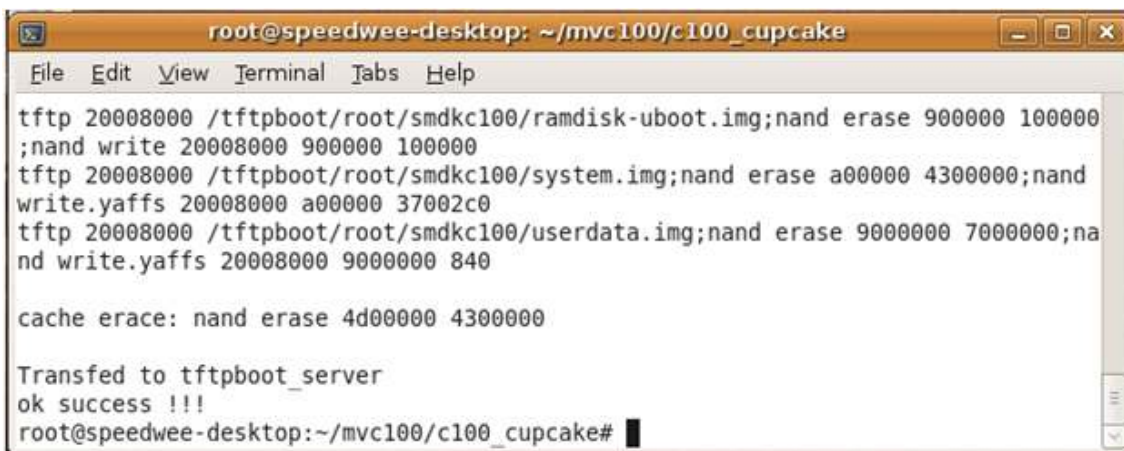
Run “build\_android.sh” for compiling

```
# ./build_android.sh
```



```
root@speedwee-desktop: ~/mvc100/c100-cupcake
File Edit View Terminal Tabs Help
root@speedwee-desktop:~/mvc100/c100-cupcake# ls
bionic          C100_cupcake_RT10_w_lib.tar.bz2  frameworks  packages
bootable       dalvik                          hardware    prebuilt
build          development                      kernel      system
build_android.sh external                          Makefile    vendor
root@speedwee-desktop:~/mvc100/c100-cupcake# ./build_android.sh
```

Roughly 5 hour that spends compile time.



```
root@speedwee-desktop: ~/mvc100/c100_cupcake
File Edit View Terminal Tabs Help
tftp 20008000 /tftpboot/root/smdkc100/ramdisk-uboot.img;nand erase 900000 100000
;nand write 20008000 900000 100000
tftp 20008000 /tftpboot/root/smdkc100/system.img;nand erase a00000 4300000;nand
write.yaffs 20008000 a00000 37002c0
tftp 20008000 /tftpboot/root/smdkc100/userdata.img;nand erase 9000000 7000000;na
nd write.yaffs 20008000 9000000 840

cache erase: nand erase 4d00000 4300000

Transfed to tftpboot_server
ok success !!!
root@speedwee-desktop:~/mvc100/c100_cupcake#
```

Done

If it will be error 64 line, you would install “uboot-mkimage”.

```
# apt-get install uboot-mkimage
```

You can see images in /out/target/product/smdk100 and in /tftpboot/root

```

root@speedwee-desktop: ~/mvc100/c100_cupcake/out/target/product/smi
File Edit View Terminal Tabs Help

ramdisk-uboot.img -> /tftpboot/root/smdkc100
system.img -> /tftpboot/root/smdkc100
userdata.img -> /tftpboot/root/smdkc100

[[[[[[[ u-boot commands for fusing ]]]]]]]

tftp 20008000 /tftpboot/root/smdkc100/ramdisk-uboot.img;nand erase 900000 100000
;nand write 20008000 900000 100000
tftp 20008000 /tftpboot/root/smdkc100/system.img;nand erase a00000 4300000;nand
write.yaffs 20008000 a00000 37002c0
tftp 20008000 /tftpboot/root/smdkc100/userdata.img;nand erase 9000000 7000000;na
nd write.yaffs 20008000 9000000 840

cache erace: nand erase 4d00000 4300000

Transfed to tftpboot_server
ok success !!!
root@speedwee-desktop:~/mvc100/c100_cupcake# cd out/target/product/smdkc100/
root@speedwee-desktop:~/mvc100/c100_cupcake/out/target/product/smdkc100# ls
android-info.txt  installed-files.txt      ramdisk.img  system
clean_steps.mk   obj                       root         system.img
data              previous_build_config.mk symbols       userdata.img
root@speedwee-desktop:~/mvc100/c100_cupcake/out/target/product/smdkc100#

```

```

root@speedwee-desktop: /tftpboot/root/smdkc100
File Edit View Terminal Help

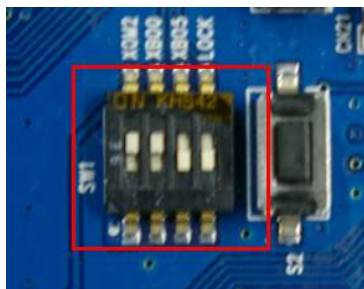
root@speedwee-desktop:/tftpboot/root/smdkc100# pwd
/tftpboot/root/smdkc100
root@speedwee-desktop:/tftpboot/root/smdkc100# ls
ramdisk-uboot.img system.img userdata.img
root@speedwee-desktop:/tftpboot/root/smdkc100#

```

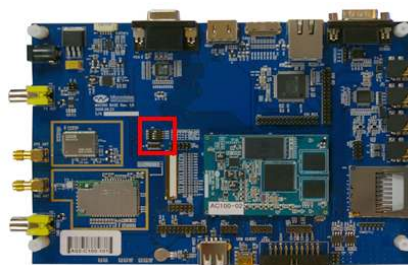
## 6. Download

As I introduced in the document MVC100\_Operating.pdf, S5PC100 has in the CPU 32KB ROM and 96KB SRAM so that can program without JTAG emulator by itself.

Please set up as under picture Boot mode. This is Boot mode which can download to mDDR through USB Cable

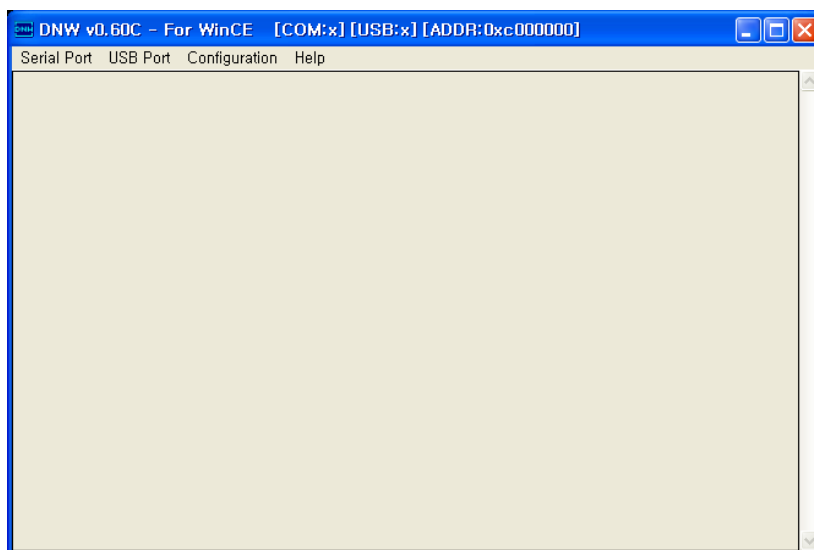


USB Boot mode



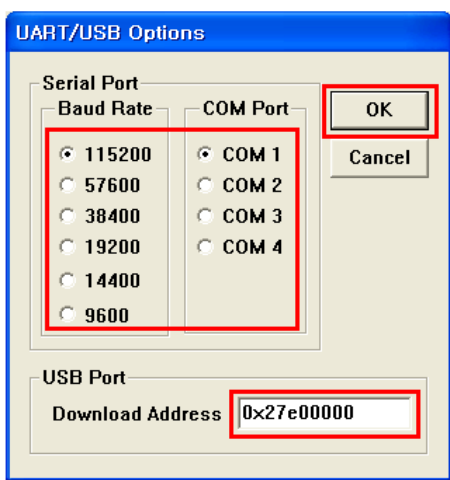
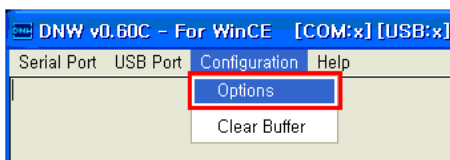
Back Switch position

Run DNW.exe in WTool\DNW v0.60C

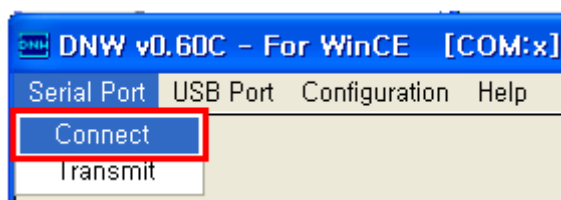


DNW

On the Configuration menu, click Options to set the UART/USB options. The following window appears on your screen. Select Baud Rate and COM Port as shown in figure "UART/USB options", enter the download address as 0x27e00000 and then click OK button.

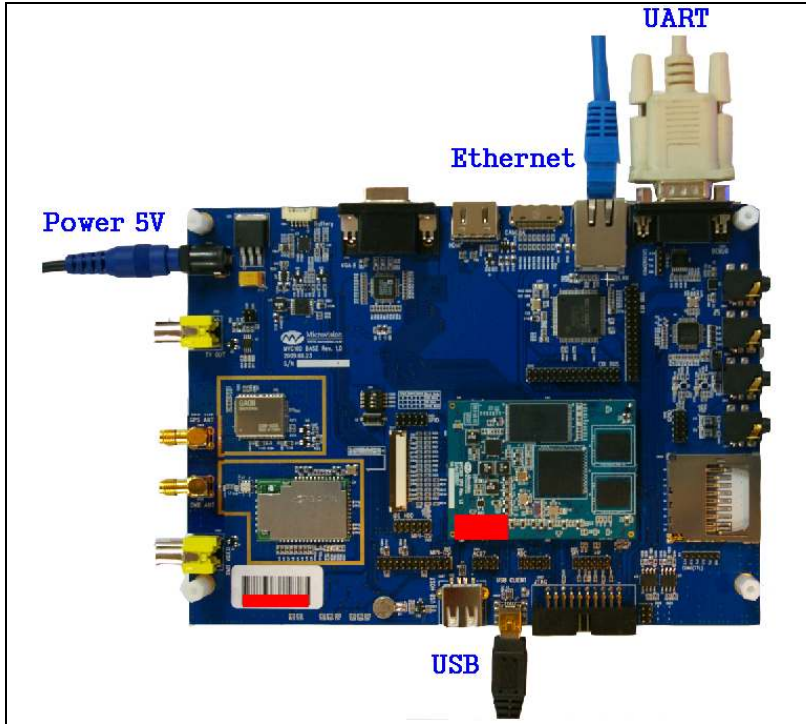


On the Serial Port menu, click Connect. Switch ON the reference board and then press any key and then install the USB driver in WSRWCWTool\DNW v0.60C driver directory.



Done

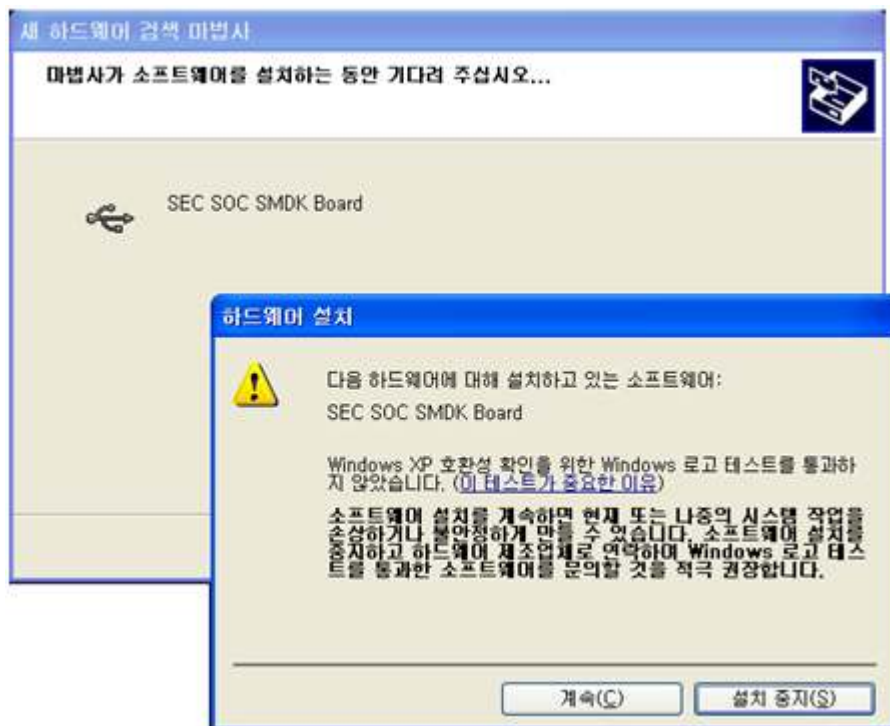
Connect Power 5V with Board and then USB 2.0 OTG Device with your Host PC to download through USB and also RS232 for monitoring.



As soon as connect power, PC is aware of USB Driver



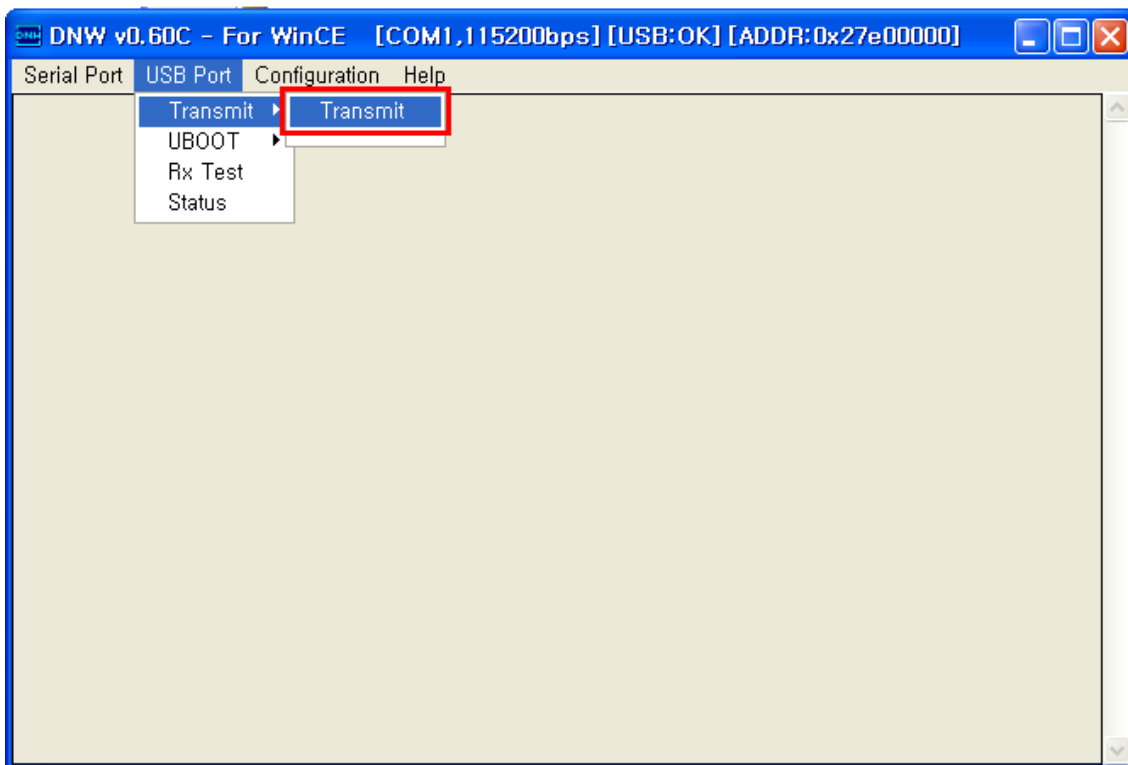
Installing in CD W\TOOLS\WMDKC100 USB Driver USB Driver



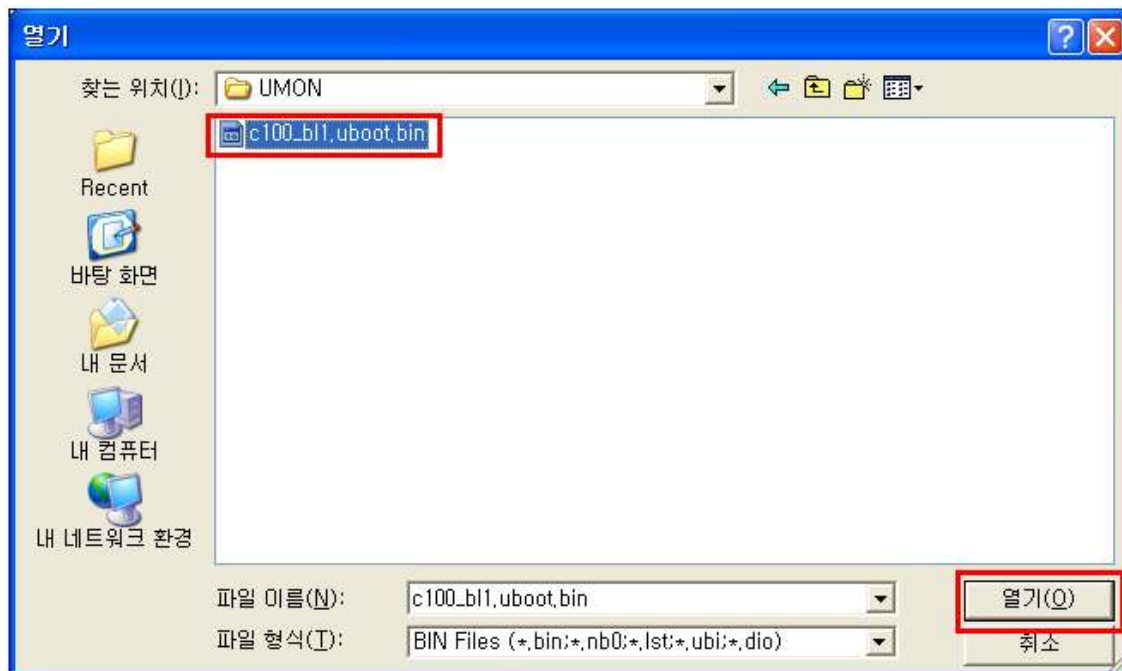
Success installed output message "USB OK"



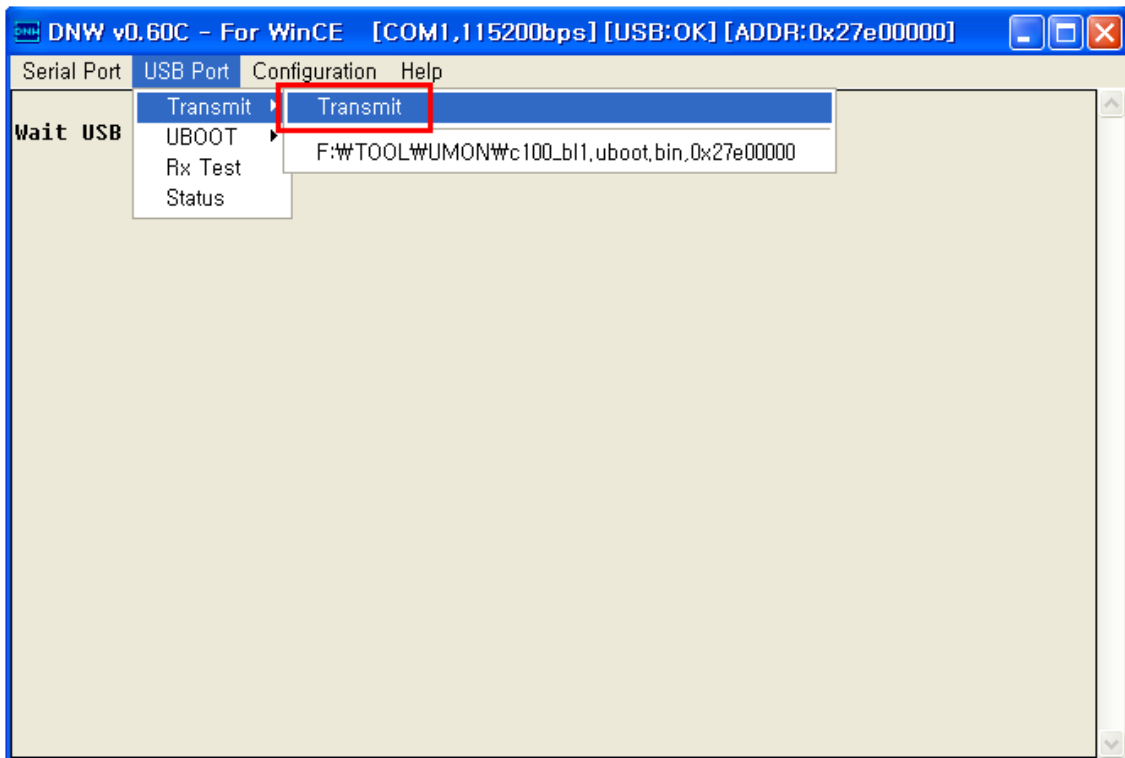
Select USB Port -> Transmit -> Transmit for loading "c100\_b11.uboot.bin".



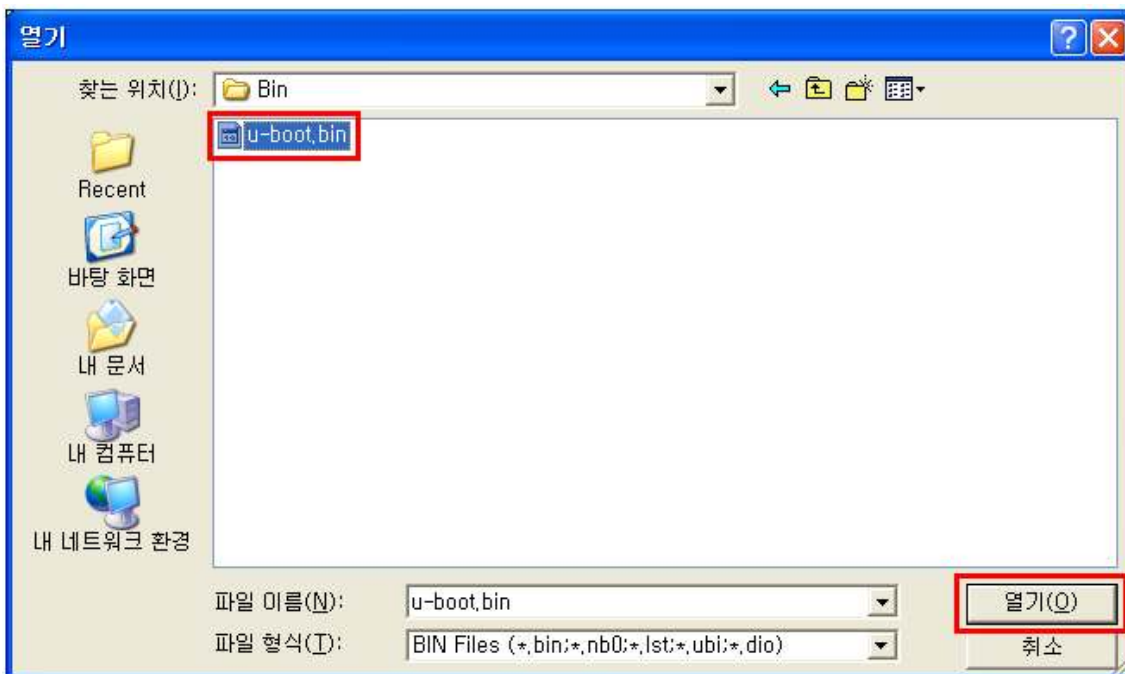
Open in CD /TOOL/UMOIN "c100\_b11.uboot.bin".



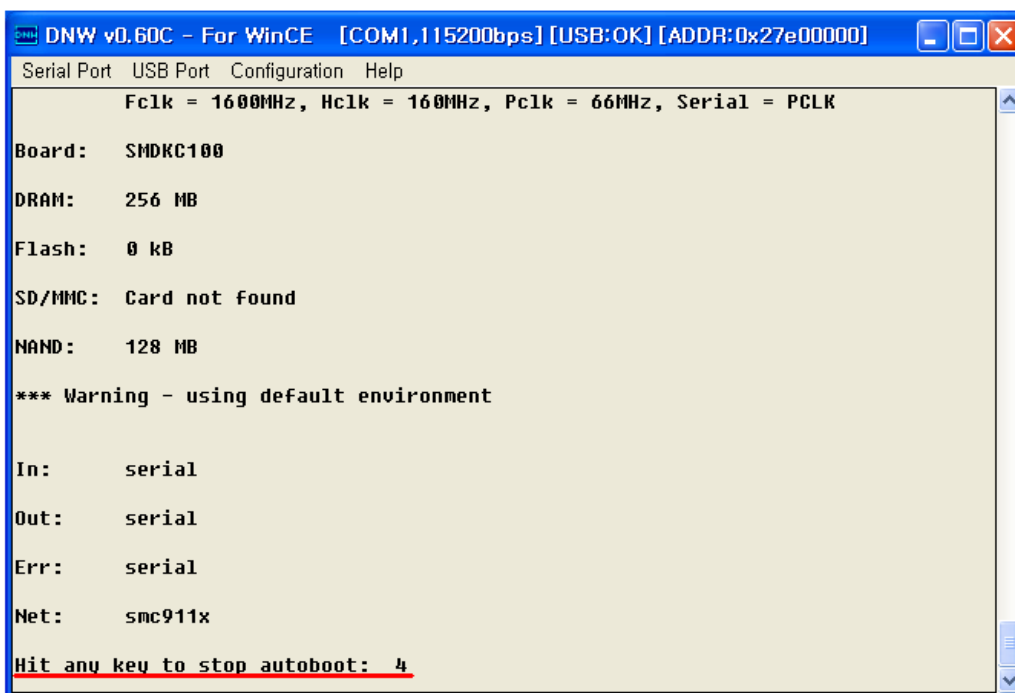
Select USB Port -> Transmit -> Transmit for running "u-boot.bin"



Open in CD SRCWBin "u-boot.bin".



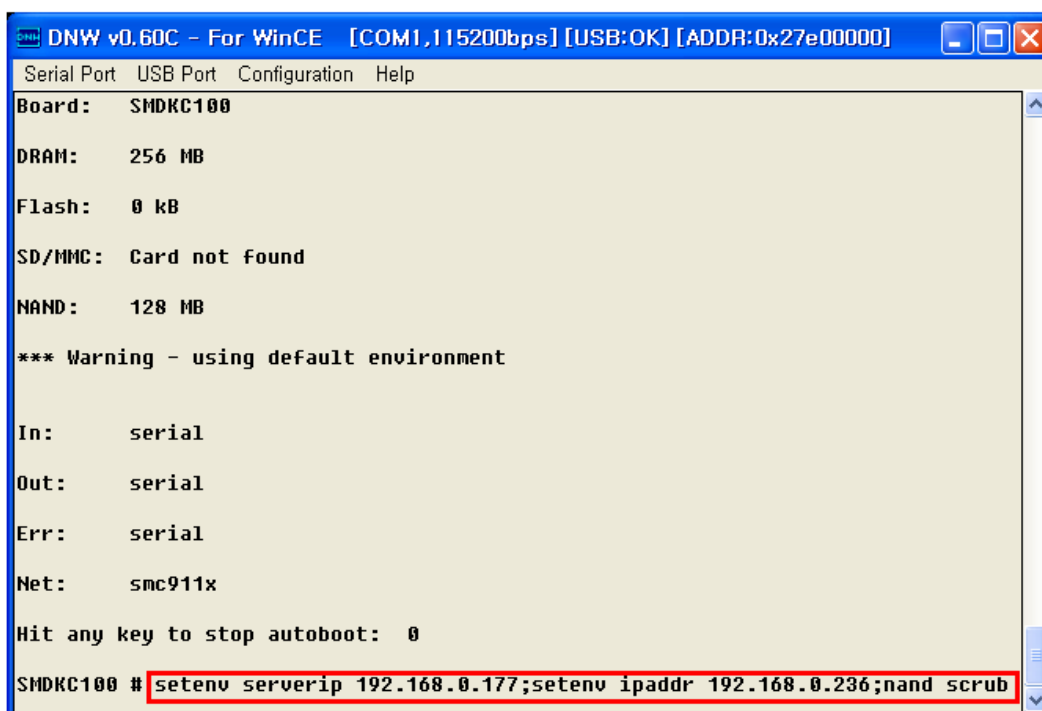
As soon as U-boot.bin download is over, the following messages appear in the DNW window. Please hit the SPACE BAR key to view.



```
DNW v0.60C - For WinCE [COM1,115200bps] [USB:OK] [ADDR:0x27e00000]
Serial Port  USB Port  Configuration  Help
      Fc1k = 1600MHz, Hc1k = 160MHz, Pc1k = 66MHz, Serial = PCLK
Board:  SMDKC100
DRAM:   256 MB
Flash:  0 kB
SD/MMC: Card not found
NAND:   128 MB
*** Warning - using default environment
In:     serial
Out:    serial
Err:    serial
Net:    smc911x
Hit any key to stop autoboot:  4
```

Fill in IP address for TFTP.

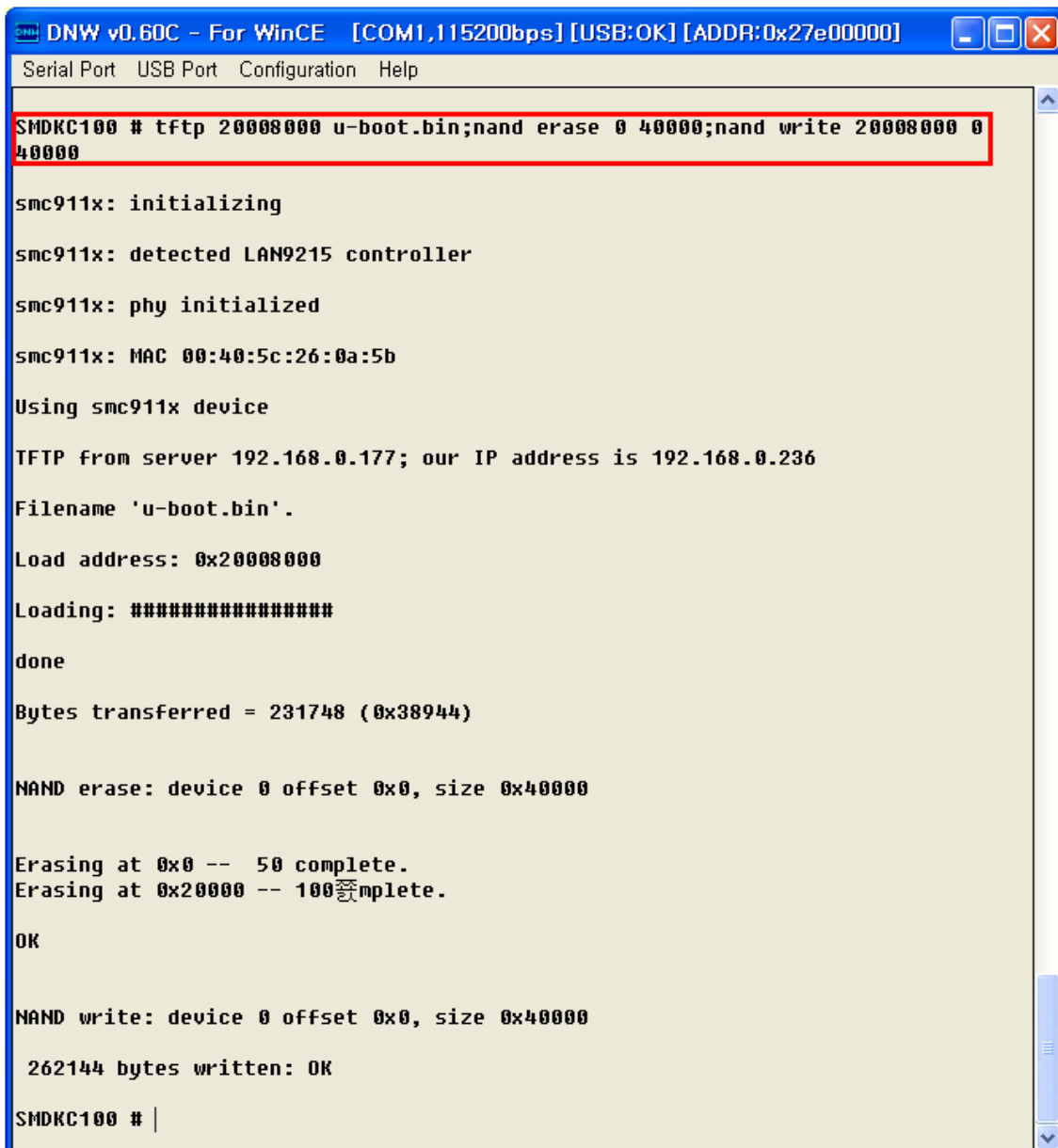
```
# setenv serverip 192.168.0.177;setenv ipaddr 192.168.0.236;nand scrub
-> enter "y"
```



```
DNW v0.60C - For WinCE [COM1,115200bps] [USB:OK] [ADDR:0x27e00000]
Serial Port  USB Port  Configuration  Help
Board:  SMDKC100
DRAM:   256 MB
Flash:  0 kB
SD/MMC: Card not found
NAND:   128 MB
*** Warning - using default environment
In:     serial
Out:    serial
Err:    serial
Net:    smc911x
Hit any key to stop autoboot:  0
SMDKC100 # setenv serverip 192.168.0.177;setenv ipaddr 192.168.0.236;nand scrub
```

Fill in commands for programming "u-boot.bin".

```
# tftp 20008000 u-boot.bin;nand erase 0 40000;nand write 20008000 0 40000
```



The screenshot shows a terminal window titled "DNW v0.60C - For WinCE [COM1,115200bps] [USB:OK] [ADDR:0x27e00000]". The terminal output shows the execution of the command: `SMDKC100 # tftp 20008000 u-boot.bin;nand erase 0 40000;nand write 20008000 0 40000`. The output includes initialization messages for the smc911x network controller, TFTP transfer details (server 192.168.0.177, IP 192.168.0.236), and successful completion of NAND erase and write operations.

```
Serial Port USB Port Configuration Help
SMDKC100 # tftp 20008000 u-boot.bin;nand erase 0 40000;nand write 20008000 0
40000

smc911x: initializing
smc911x: detected LAN9215 controller
smc911x: phy initialized
smc911x: MAC 00:40:5c:26:0a:5b
Using smc911x device
TFTP from server 192.168.0.177; our IP address is 192.168.0.236
Filename 'u-boot.bin'.
Load address: 0x20008000
Loading: #####
done
Bytes transferred = 231748 (0x38944)

NAND erase: device 0 offset 0x0, size 0x40000

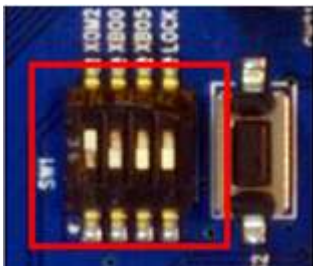
Erasing at 0x0 -- 50 complete.
Erasing at 0x20000 -- 100 complete.

OK

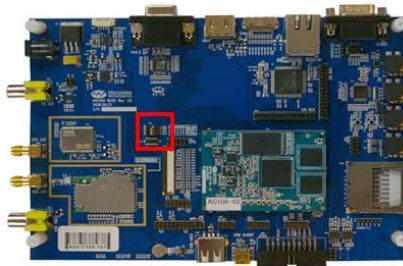
NAND write: device 0 offset 0x0, size 0x40000

262144 bytes written: OK
SMDKC100 # |
```

NAND was programmed 0x0 u-boot. Now change NAND Flash boot mode for programming kernel, file system.



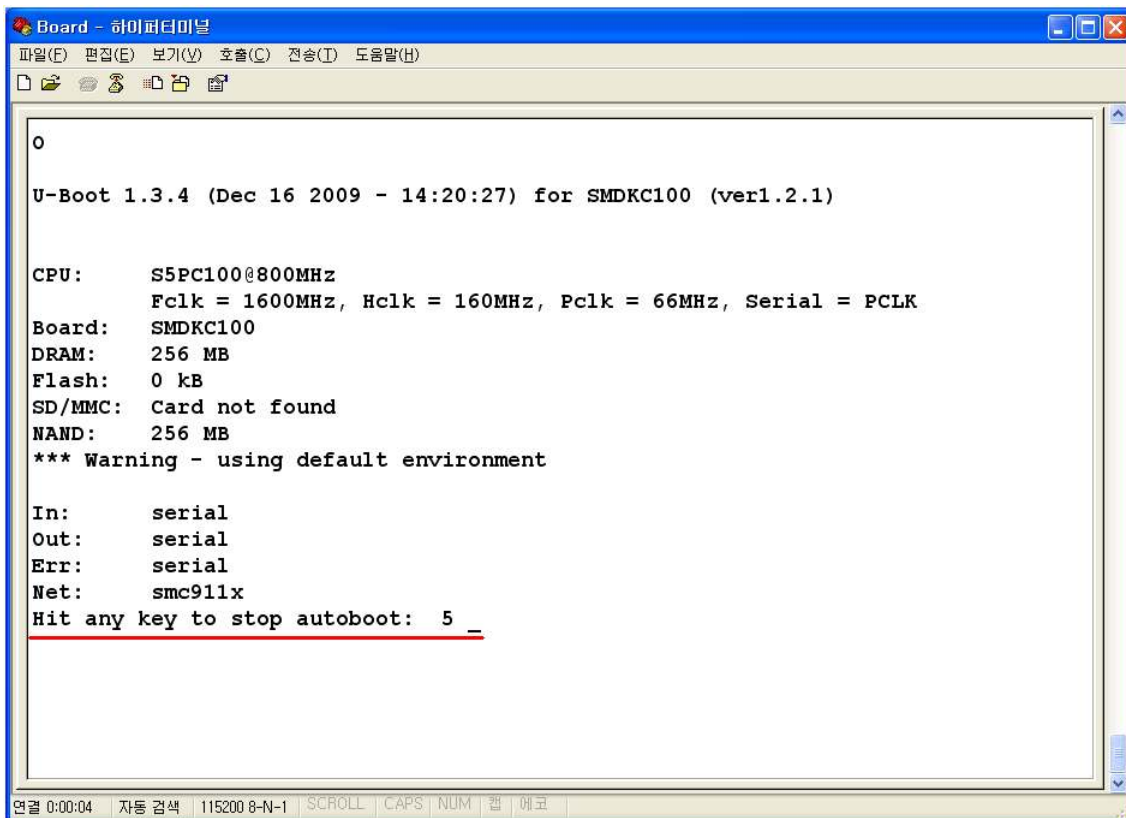
NAND boot mode



Back Reset Switch position

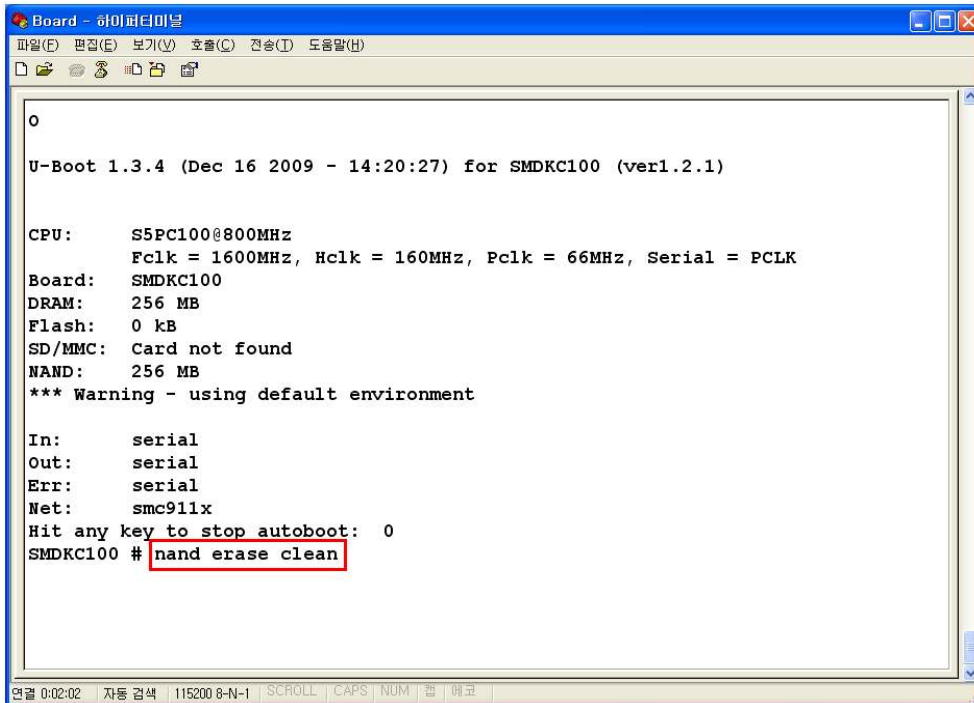
DNW can't copy text so I recommended "HyperTerminal"

As soon as U-boot.bin download is over, the following messages appear in the DNW window. Please hit the SPACE BAR key to view.



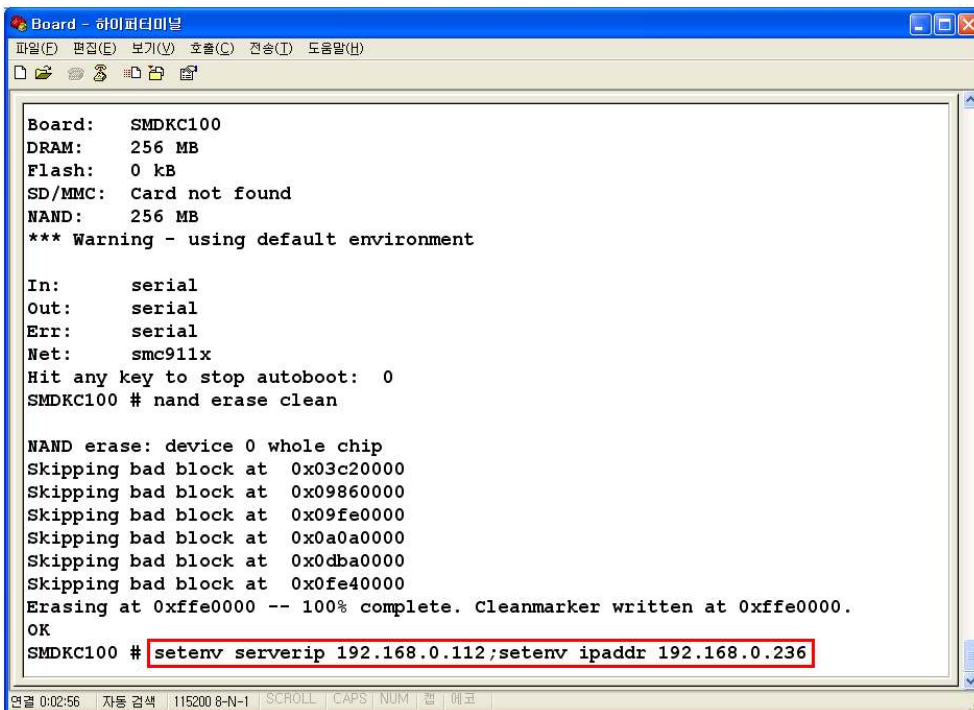
Use a command for the delete because Android different from bootloader position.

# nand erase clean

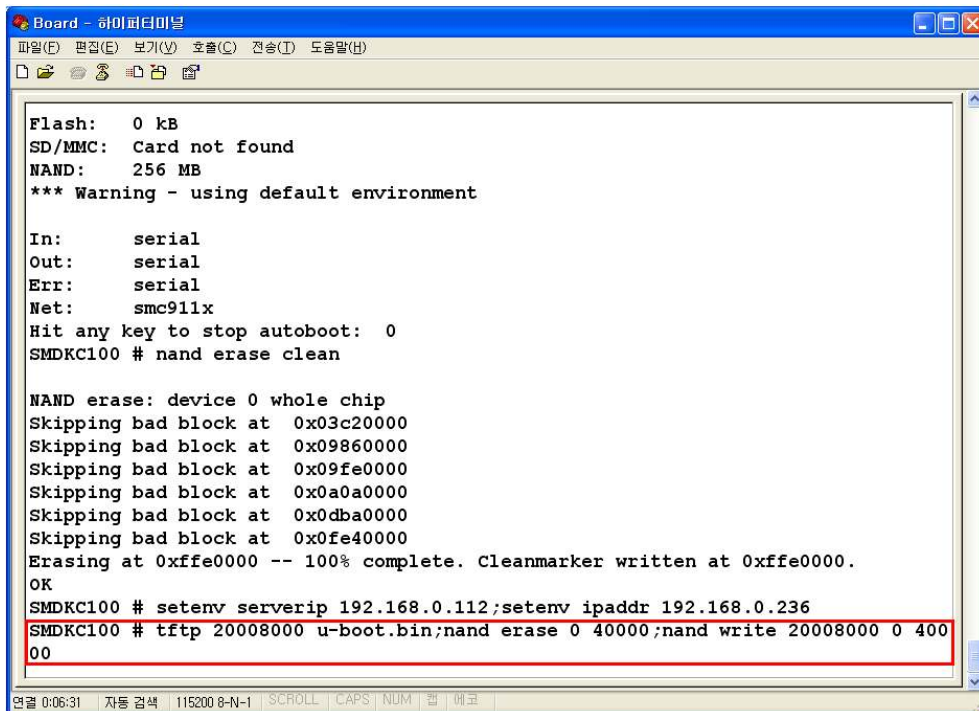


Setup IP address for Tftp.

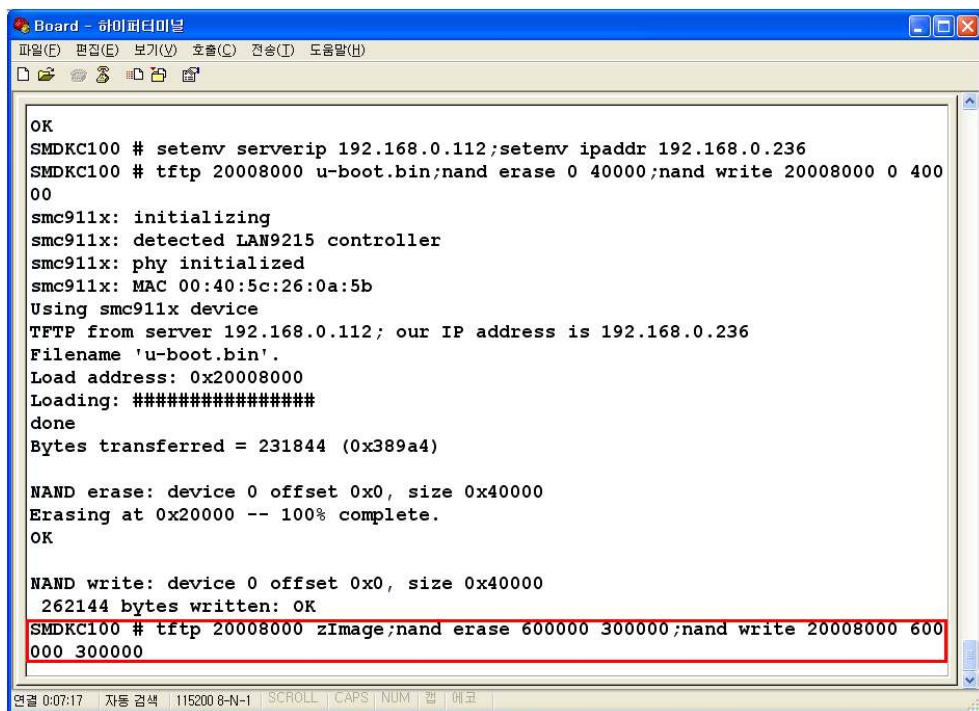
# setenv serverip 192.168.0.112;setenv ipaddr 192.168.0.236



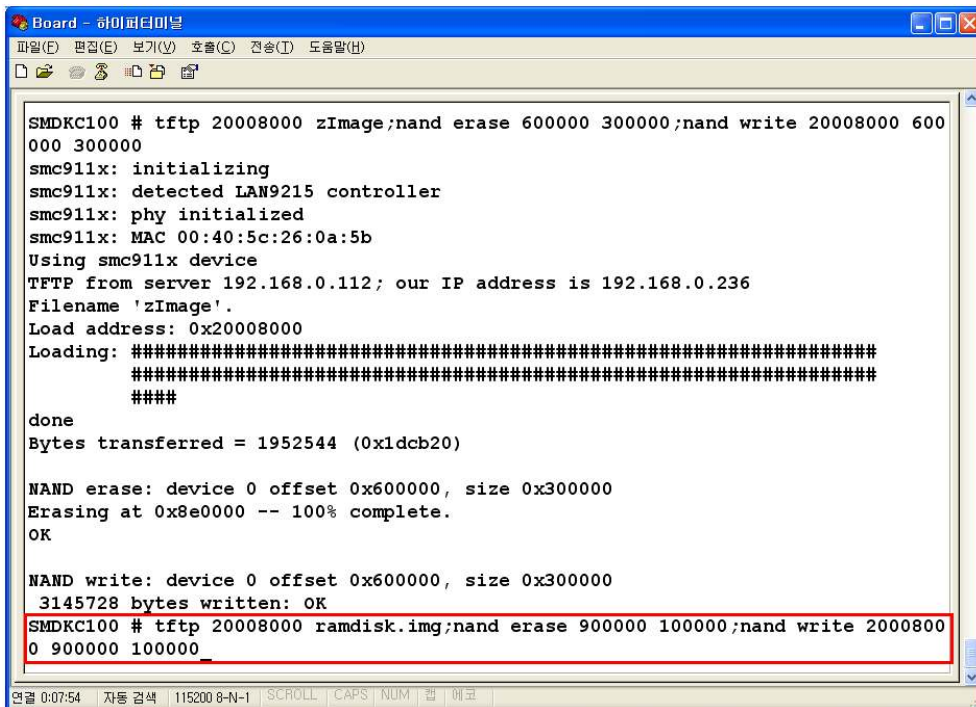
```
# tftp 20008000 u-boot.bin;nand erase 0 40000;nand write 20008000 0 40000
```



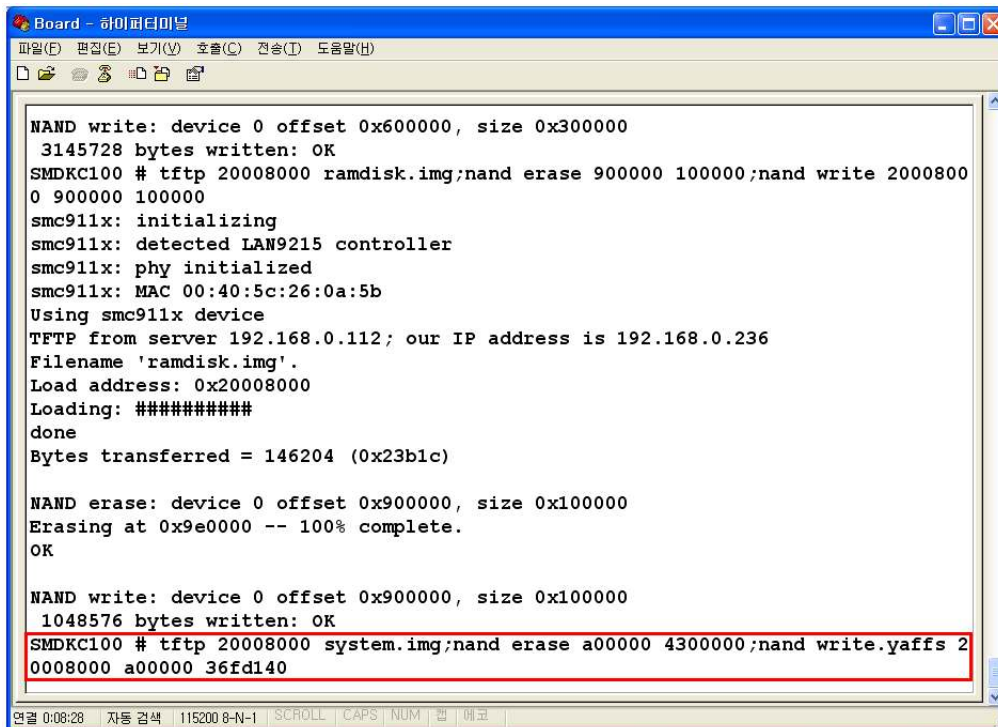
```
# tftp 20008000 zImage;nand erase 60000 300000;nand write 20008000 60000 300000
```



```
# tftp 20008000 ramdisk.img;nand erase 900000 100000;nand write 20008000 900000 100000
```

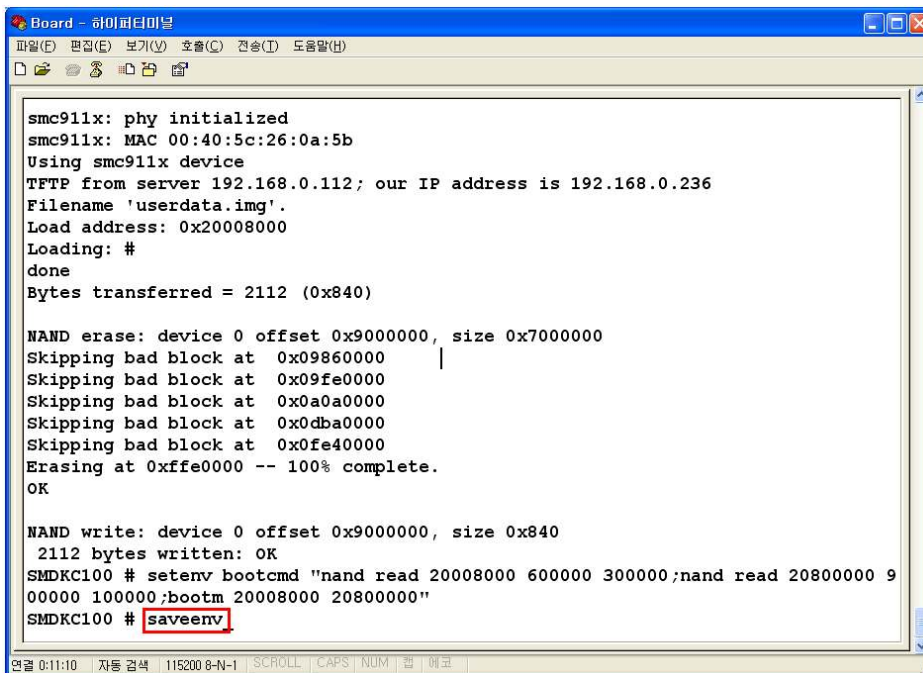


```
# tftp 20008000 system.img;nand erase a00000 430000;nand write.yaffs 20008000 a00000 36fd140
```

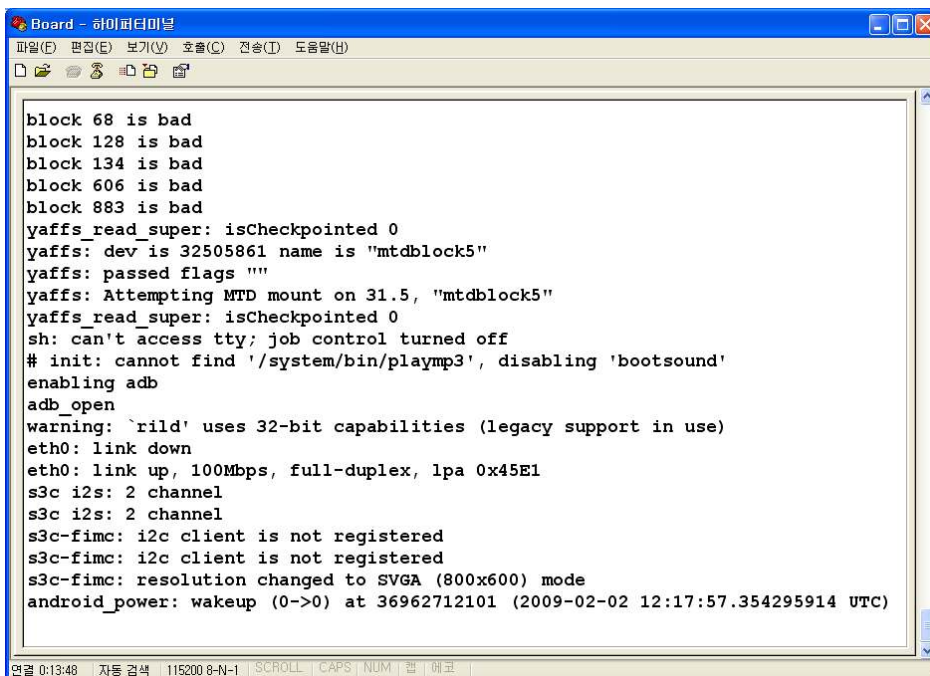




# saveenv



Reboot



Done

## 7. The way setup ADB for Application.

Eclipse download site

(download "Eclipse IDE for Java EE Developers")

<http://www.eclipse.org/downloads/>

Android SDK download site

(download "android-sdk\_r04-windows.zip")

<http://developer.android.com/sdk/index.html>

JAVA JDK download site (download Version 6 .)

[http://java.sun.com/javase/downloads/index\\_jdk5.jsp](http://java.sun.com/javase/downloads/index_jdk5.jsp)

You can see in CD /ADB included package.

/ AndroidBSPTools (USB Driver)

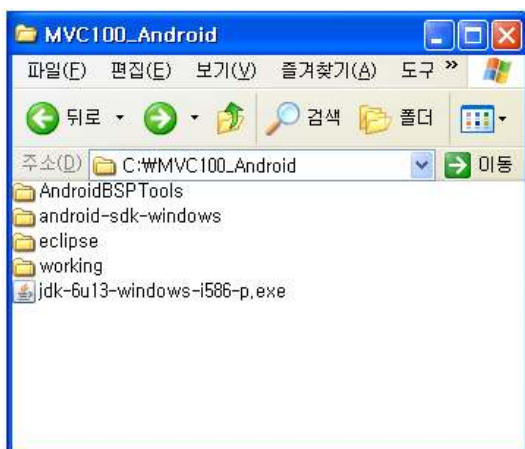
/ eclipse (eclipse program)

/ Java6-JDK (Java JDK6)

/ SDK (Android SDK)

You need to install Java6-JDK and USB but you don't have to install eclipse and SDK only run.

Example

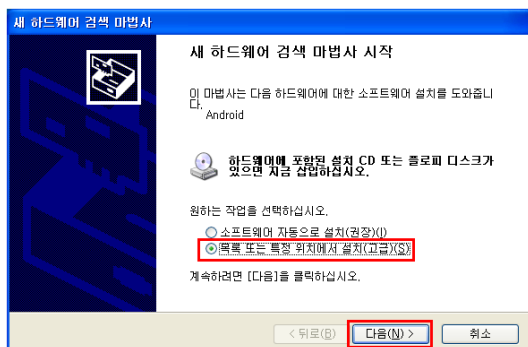


### To install USB Driver.

After booted, connect USB cable PC with Board so that can see message. (You have to check on board, settings -> Application -> Development -> USB debubbing so you can use the ADB.)

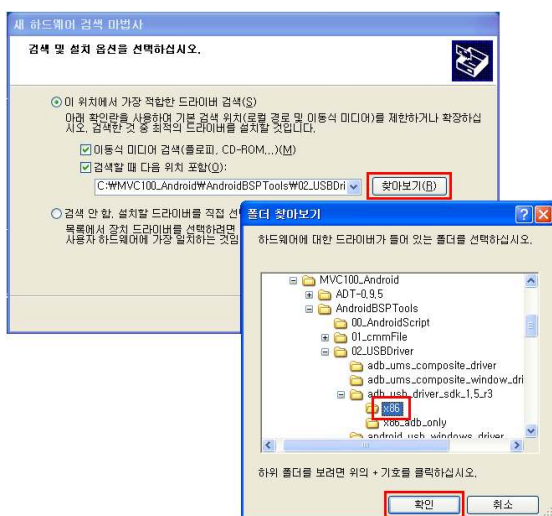


Click "NEXT".

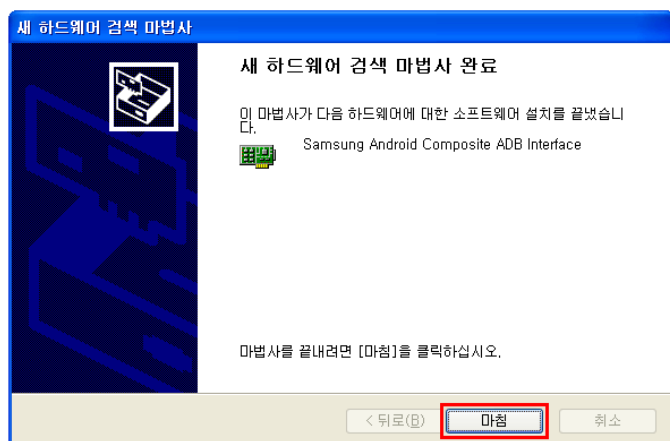
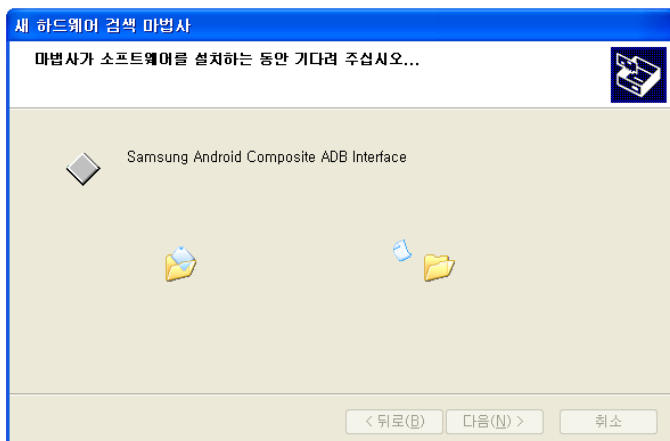
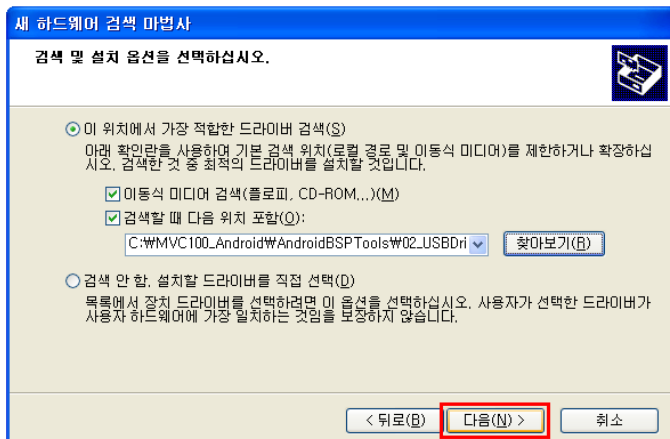


You can see USB Driver in CD

₩ADB₩AndroidBSPTools₩02\_USBDriver₩adb\_usb\_driver\_sdk\_1.5\_r3₩x86.

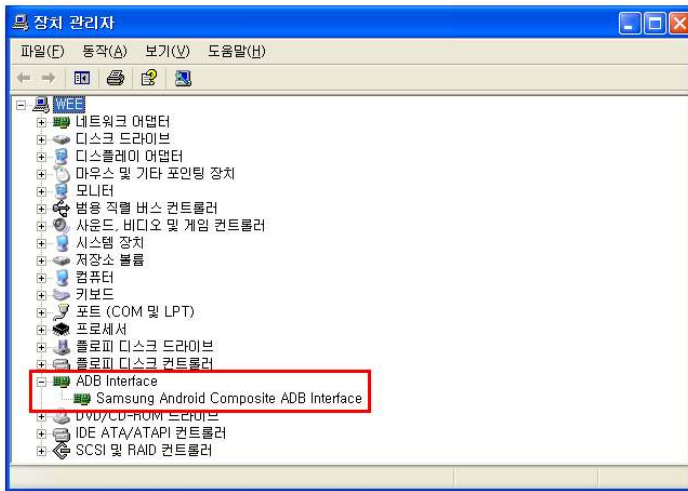


Click "NEXT".



Done

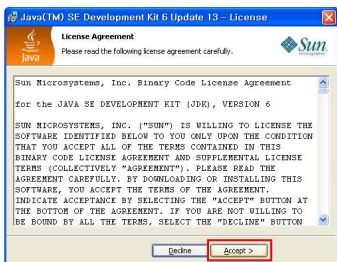
You can see the ADB.



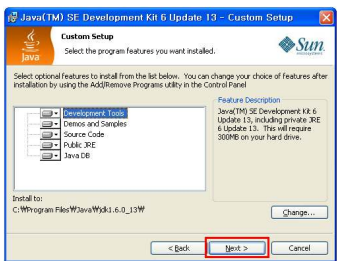
## To install JAVA JDK6

Run “jdk-6u13-windows-i586-p.exe” .

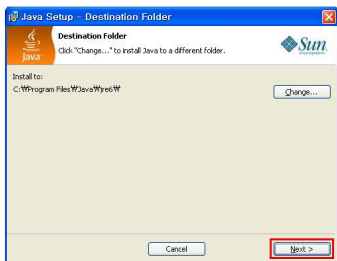
Click “Accept”



Click “NEXT”



Click “NEXT”



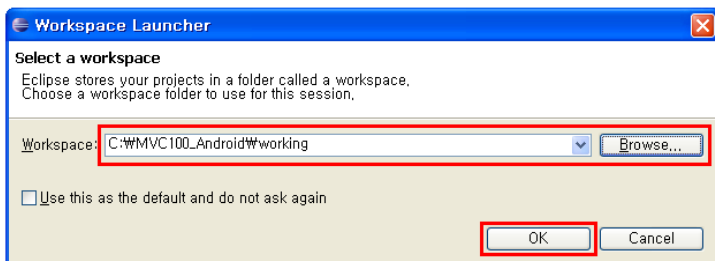
Click “Finish”



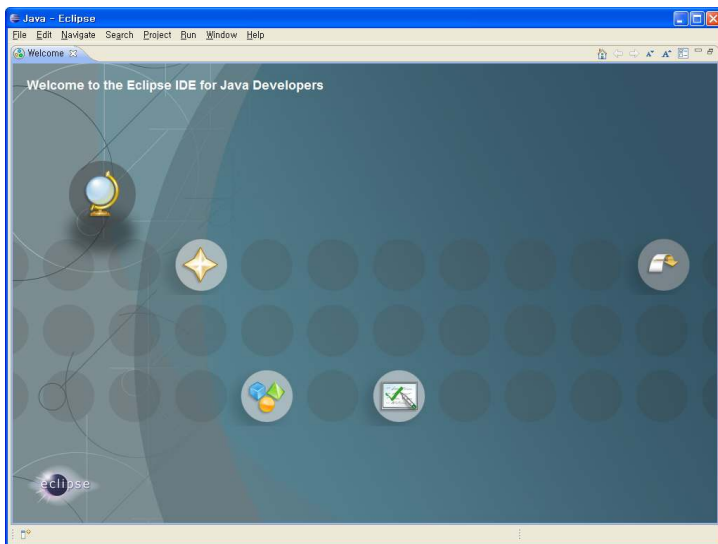
### To setup Eclipse, Install SDK

Eclipse is in CD, I copy to C:\MVC100\_AndroidW.  
Run eclipse.exe.

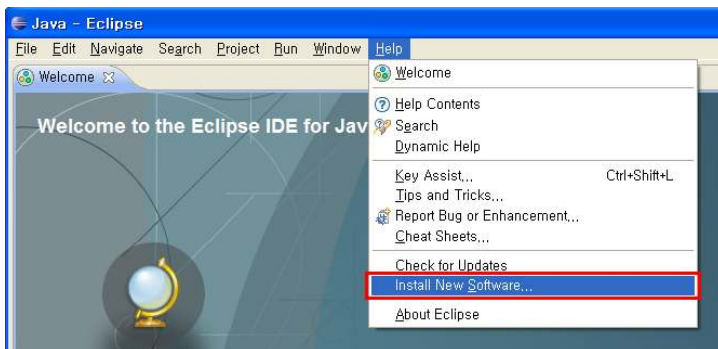
Setup PATH "Workspace".



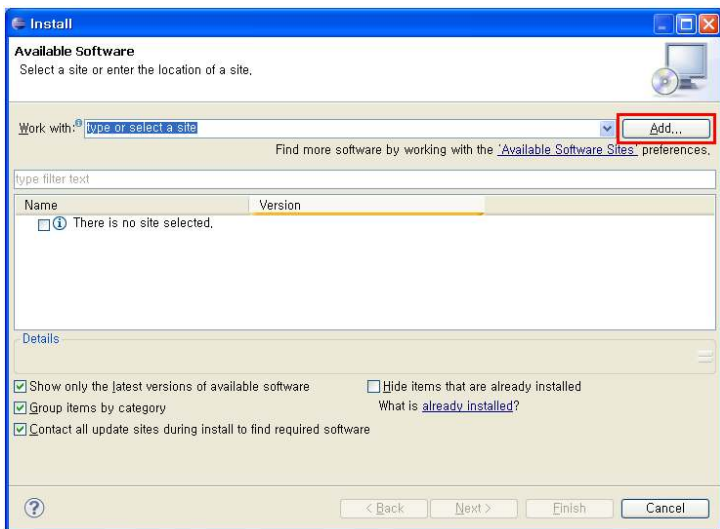
### Eclipse



Click Help -> Install New Software .

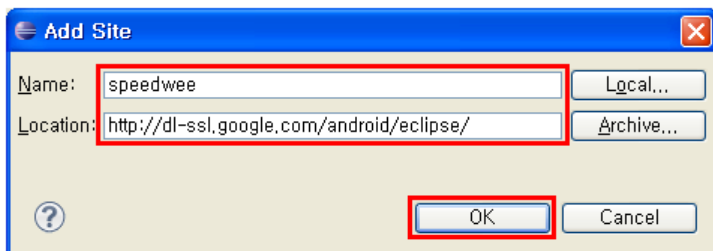


Select "Add".

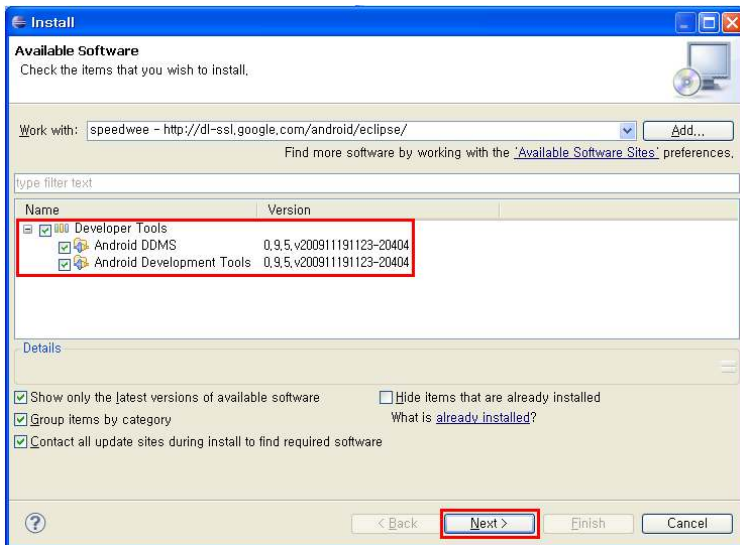


Fill in as address.

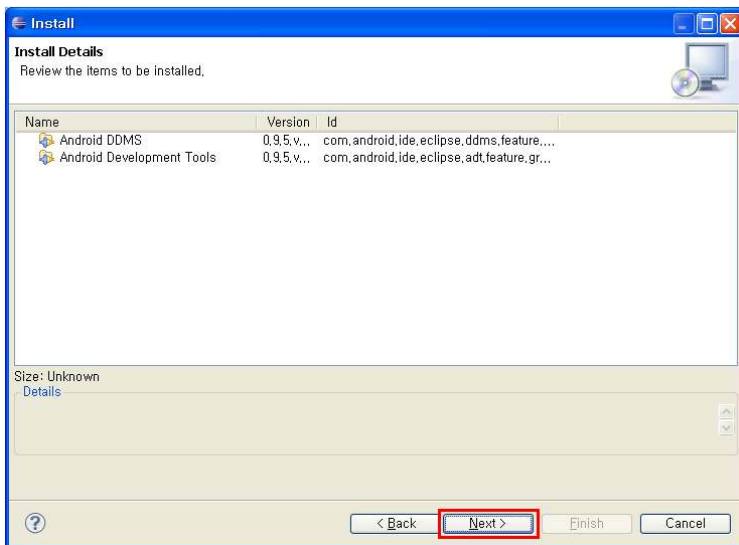
<http://dl-ssl.google.com/android/eclipse/>



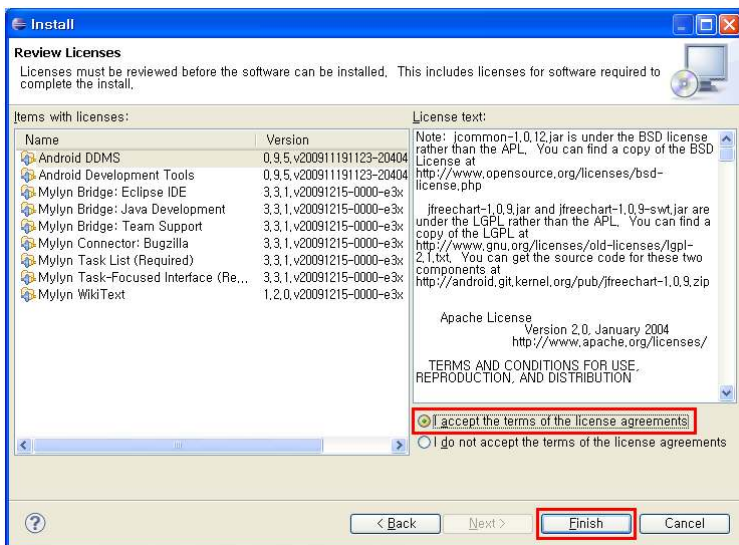
As under picture, check and then click "NEXT".



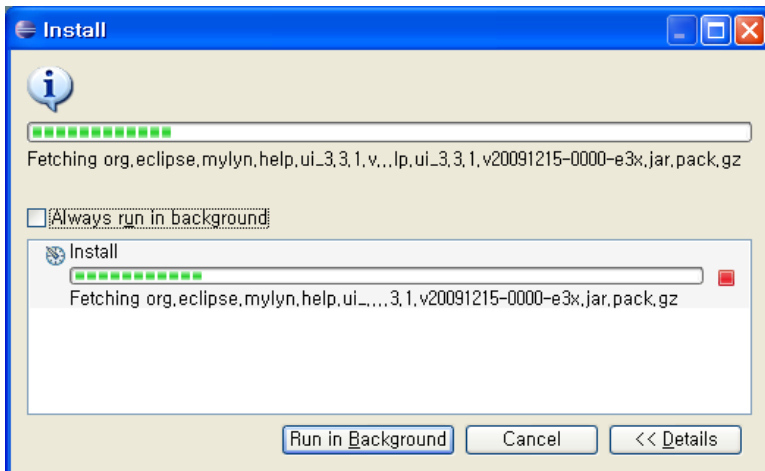
Click "NEXT".



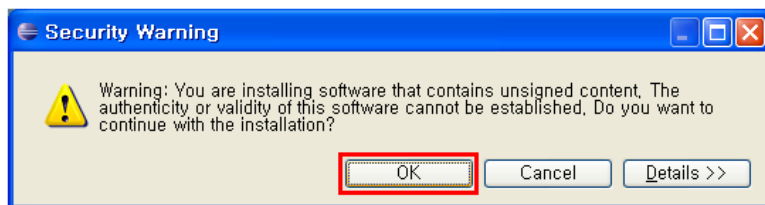
Click "Finish".



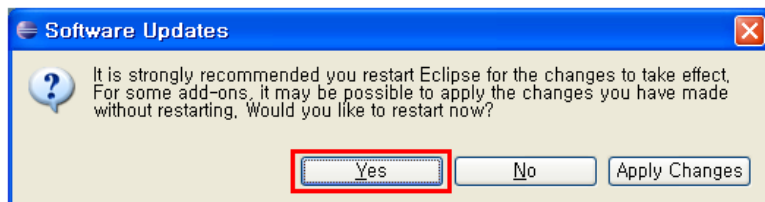
It is installing.



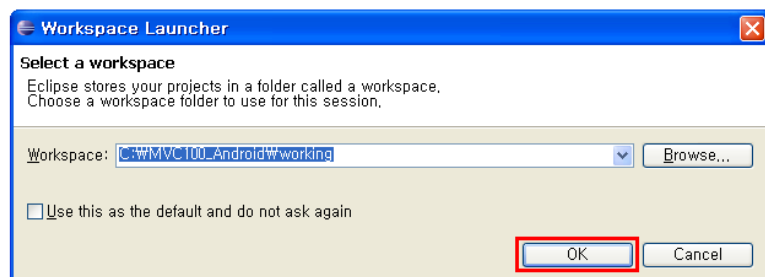
Click "OK".



Click "Yes".



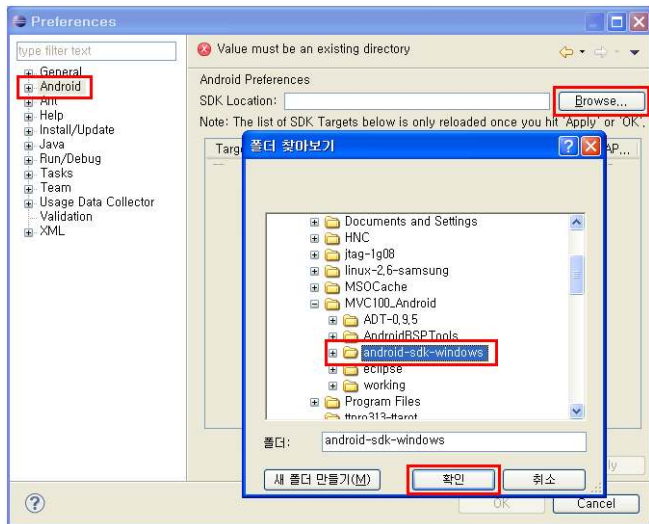
Click "OK".



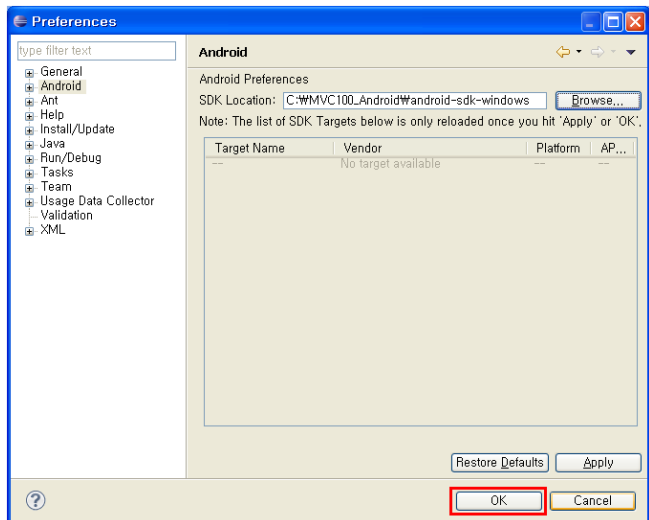
Select Windows->Preferences.



Select Android and then click "Browse", Assign PATH SDK.



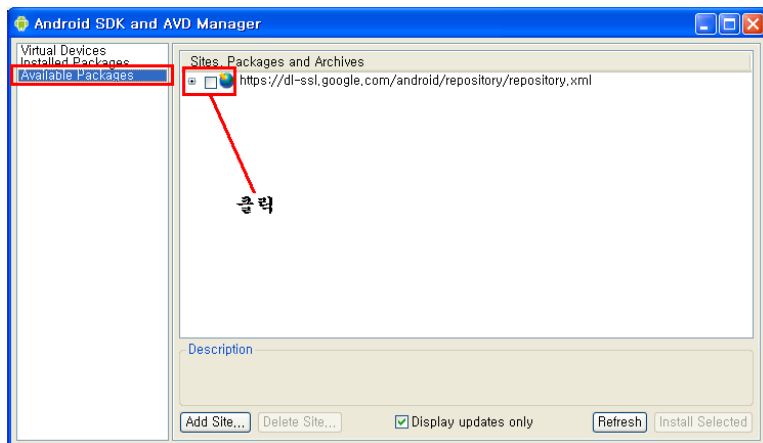
Click "OK".



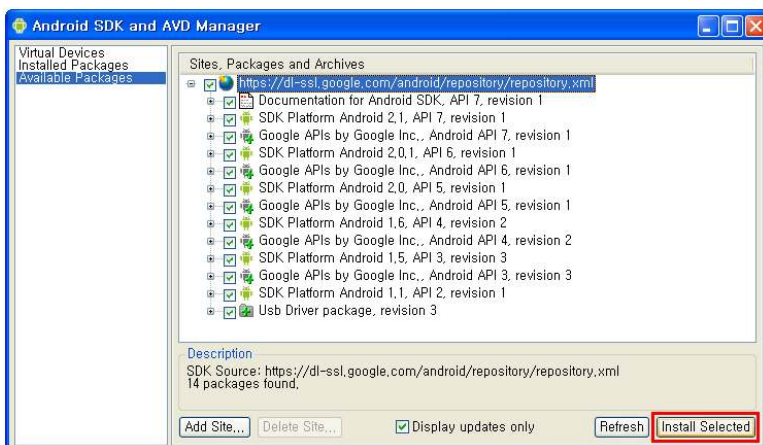
Select Windows -> Android SDK and AVD Manager.



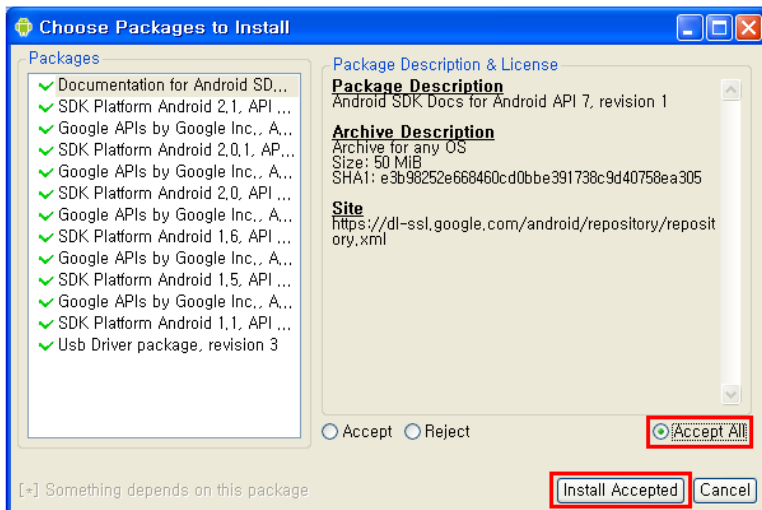
Select Android Packages and then click it.



Click "Install Selected".

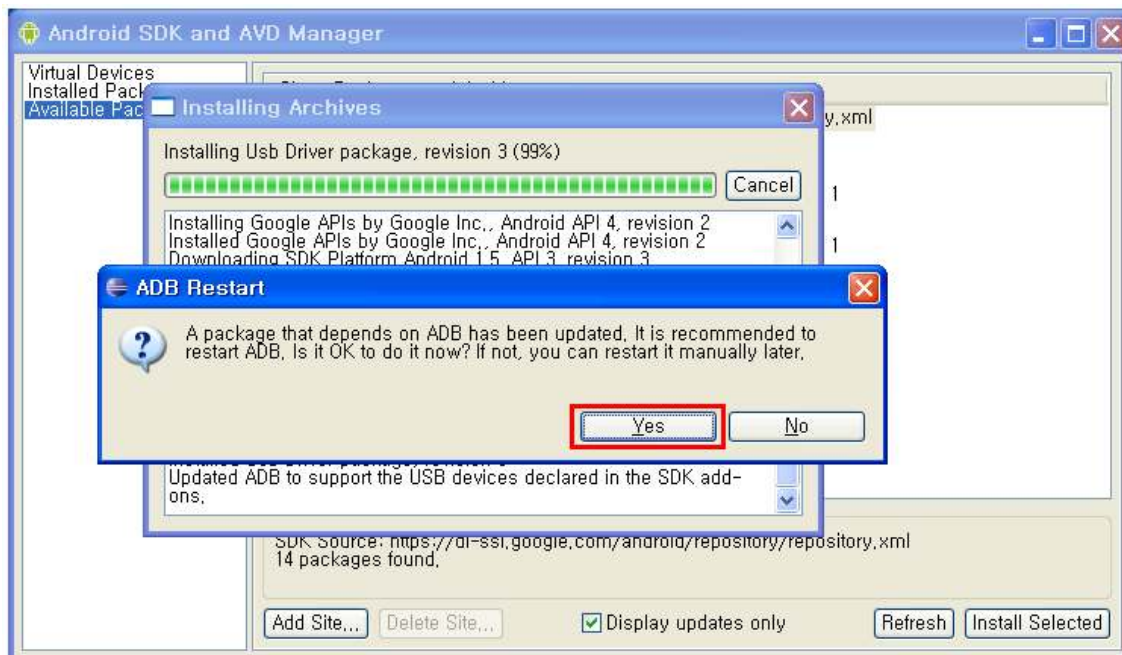


Select "Accept All" and then click "Install Accepted".

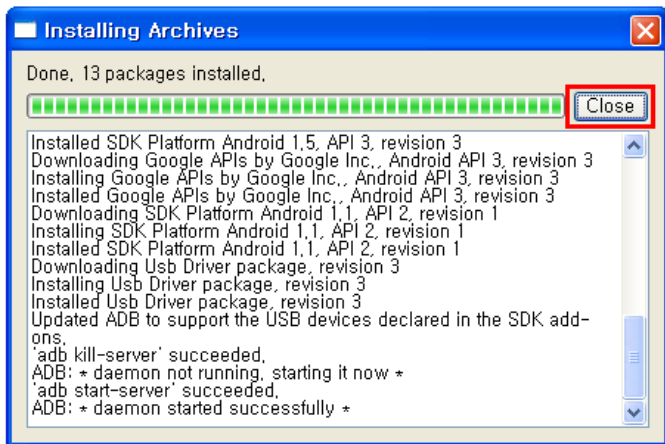


It middle of installing message, click "OK".

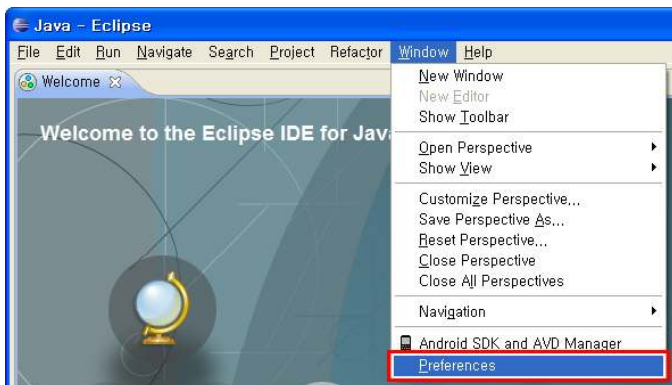
(It spends All Package time 40 hour.)



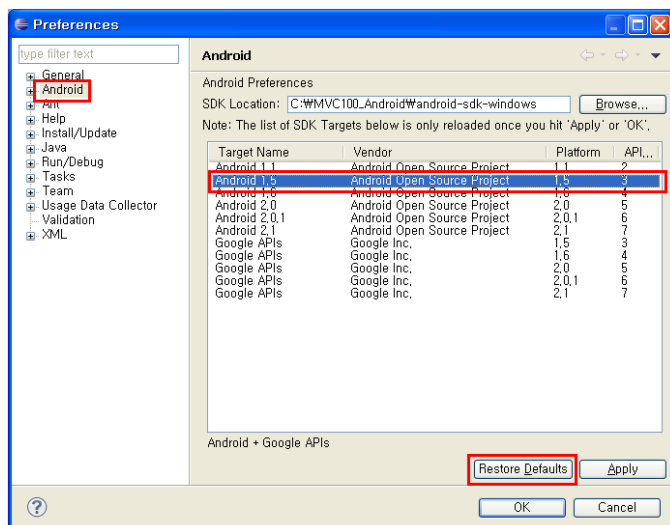
Click "Close".



Select Windows->Preferences .



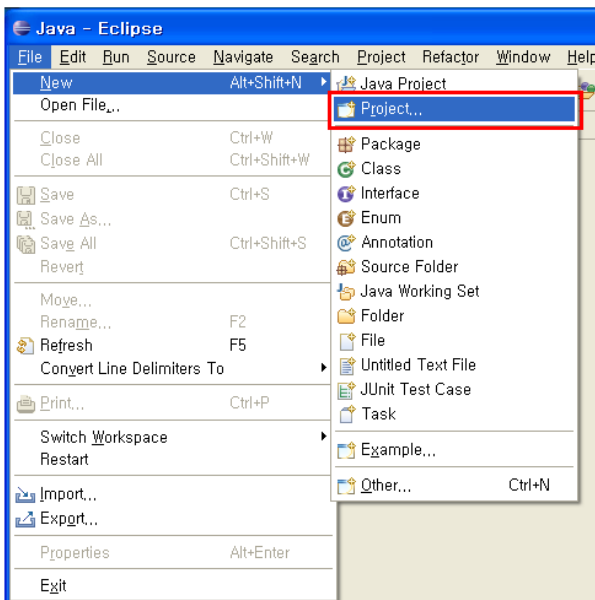
Select Android and then click "Restore Defaults".  
(You don't have to do it.)



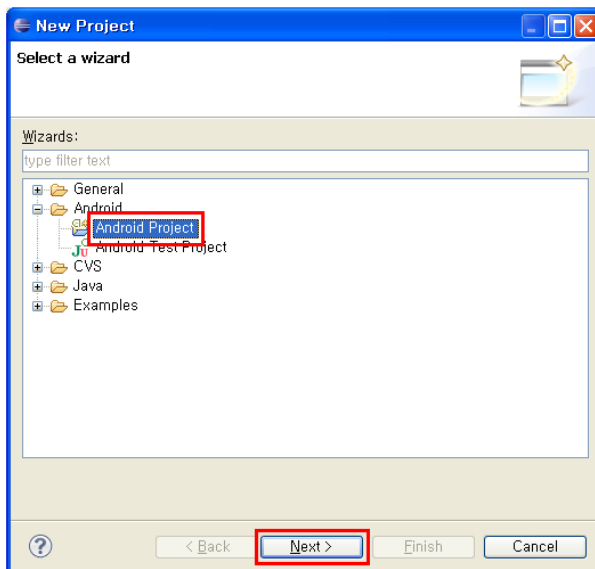
Rerun Eclipse. It has to connect USB cable PC with MVC100.

Make New project Eclipse.

Select New->Project



Select "Android Project" and then Click "NEXT".



Project name : xxx

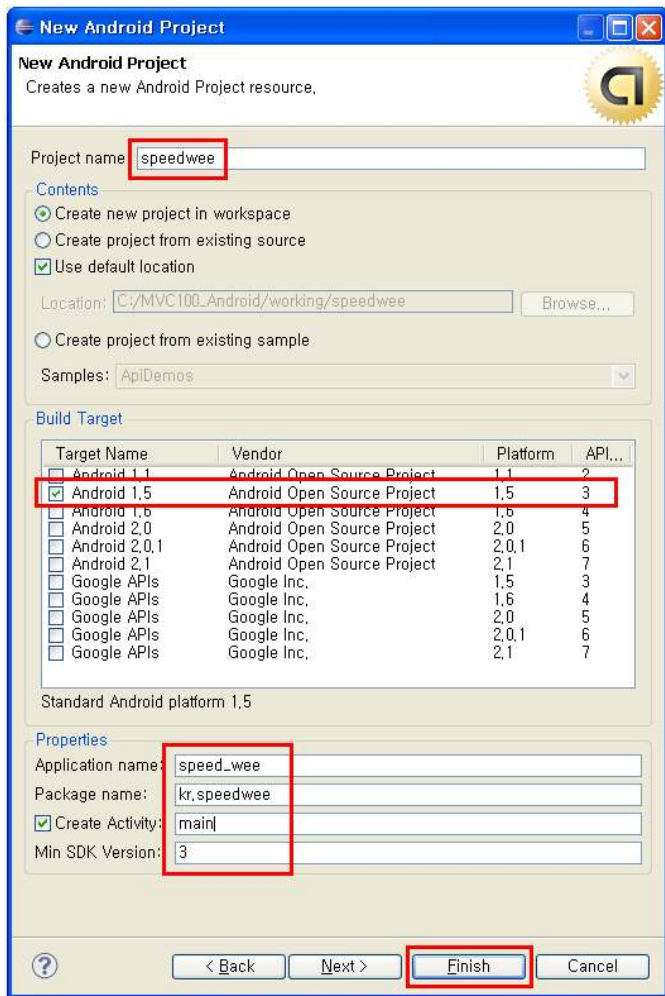
Select Android 1.5

Application name : xxx

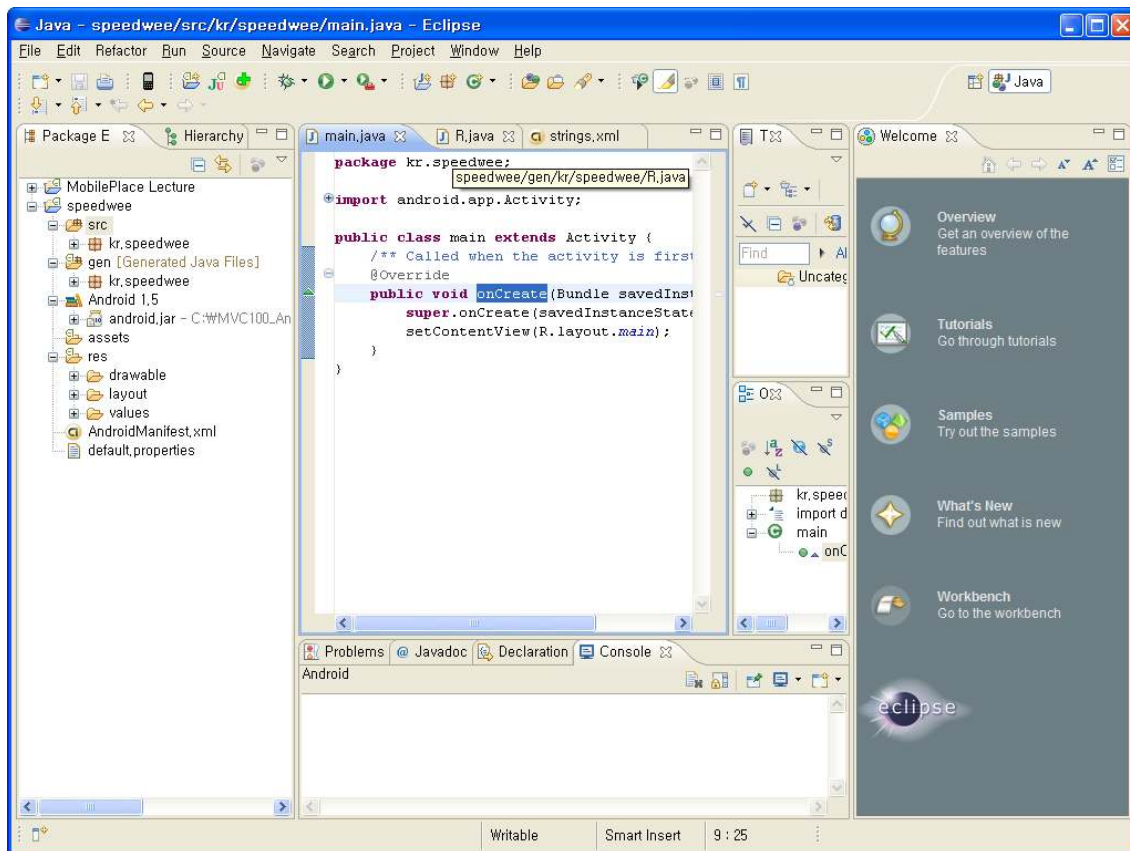
Package name : kr.xxx

Create Activity : fill in "main"

Click "Finish".



Done



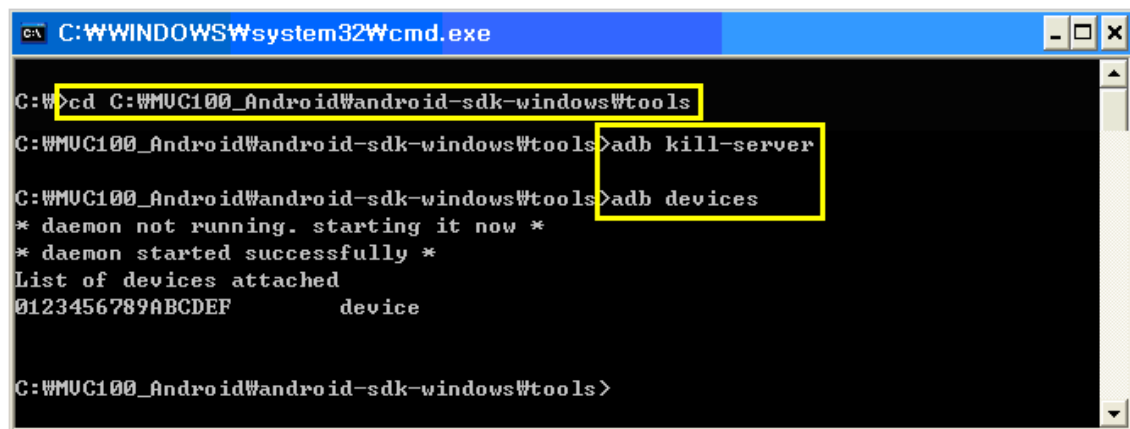
Run in Windows start -> run -> cmd and then run as adb

The order

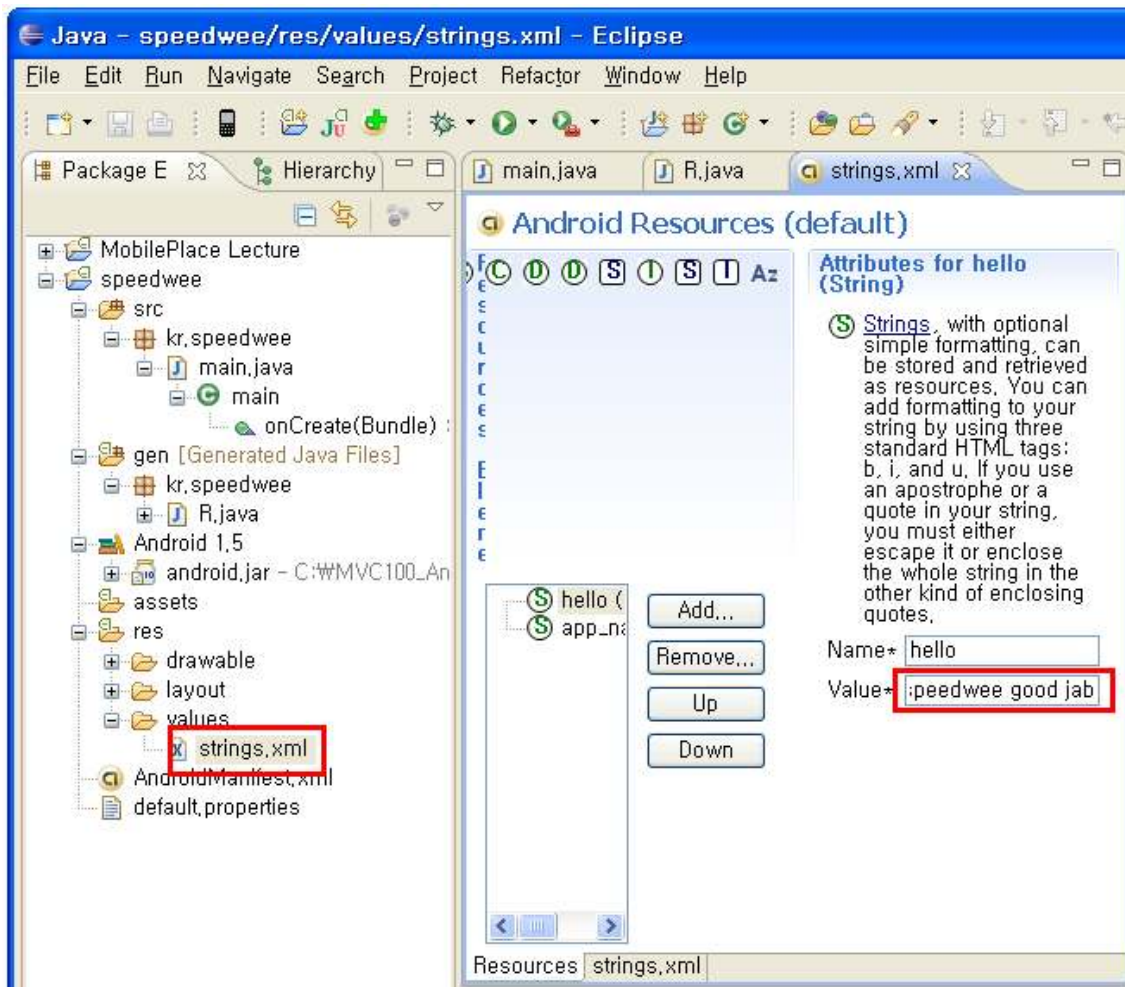
```
cd %windir%\system32\cmd
```

```
adb kill-server
```

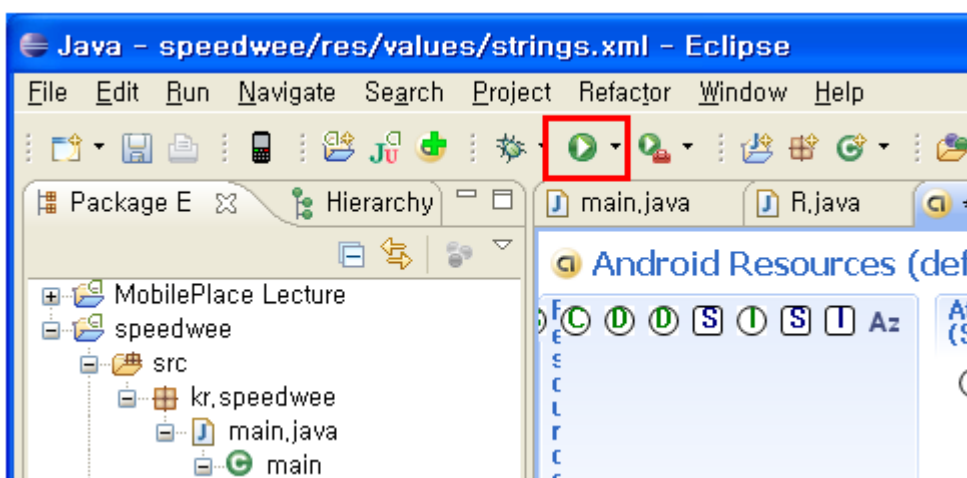
```
adb devices
```



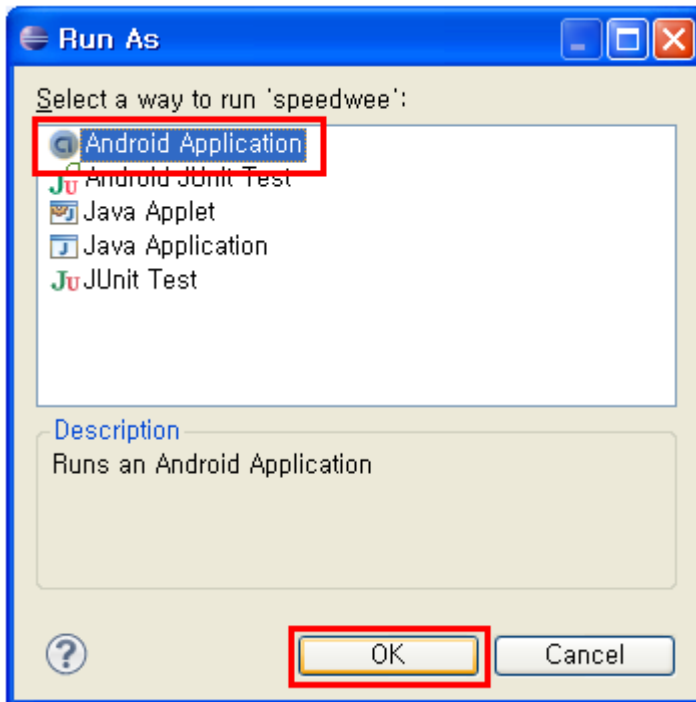
write something.



Click "a icon"



Select "Android Application" and then click "OK".



Done