
iMX.6 Processor Series

All-in-One Embedded Computer - EC21XX

User Manual



Copyright 2000 – 2014 IC Nexus Corporation. All rights reserved

Release Notes

Version	Release Date	Notes
1.0	August 2014	Initial release according to Lab

Disclaimer

This documentation is provided for use with IC Nexus products. No license to IC Nexus property right is granted. IC Nexus assumes no liability, provides no warranty either expressed or implied relating to the usage, or intellectual property right infringement that may result from its use.

IC Nexus provides this document without any warranty of any kind, expressed or implied, including, but not limited to, its particular purpose. IC Nexus may make changes to this document without notice.

Federal Communication Commission Interference Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

FCC Caution: To assure continued compliance, any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment. (Example - use only shielded interface cables when connecting to computer or peripheral devices).

FCC Radiation Exposure Statement

This equipment complies with FCC RF radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with a minimum distance of 20 centimeters between the radiator and your body.

This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

The antennas used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter.

Table of Contents

TABLE OF CONTENTS.....	4
1. PACKAGE CONTENTS.....	5
2. OVERVIEW.....	6
3. SETUP.....	7
3.1 Connecting the Debug Port to PC	7
3.2 Start Running	10
3.3 Connecting to the PC (for Android 4.2)	12
3.4 Update firmware via USB dongle (for Android 4.2)	22
4. RUNNING SOFTWARE	24
4.1 Android	24
4.2 Ubuntu Linux.....	29
5. TOUCH PANEL CALIBRATION.....	34
5.1 Android RTP re-calibrate procedure	34
5.2 Ubuntu RTP re-calibrate procedure.....	36

1. Package Contents

All-in-One embedded computer products include EC2107 and EC2110.

Item	Notes
EC21xx	
Power Adapter	
RS232 IDC Cable	Pin Header for DB9
CD	Software and User's Manual

2. Overview



Ubuntu enabled device showing ports on each side



Android enabled device with wireless antennas attached

3. Setup

3.1 Connecting the Debug Port to PC

During development, it is a good idea to connect using the debug port.

- Find the debug port on EC2107 / EC2110.

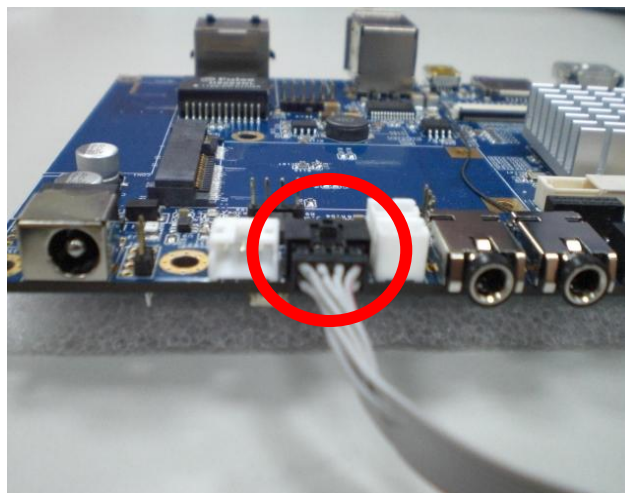


EC2107



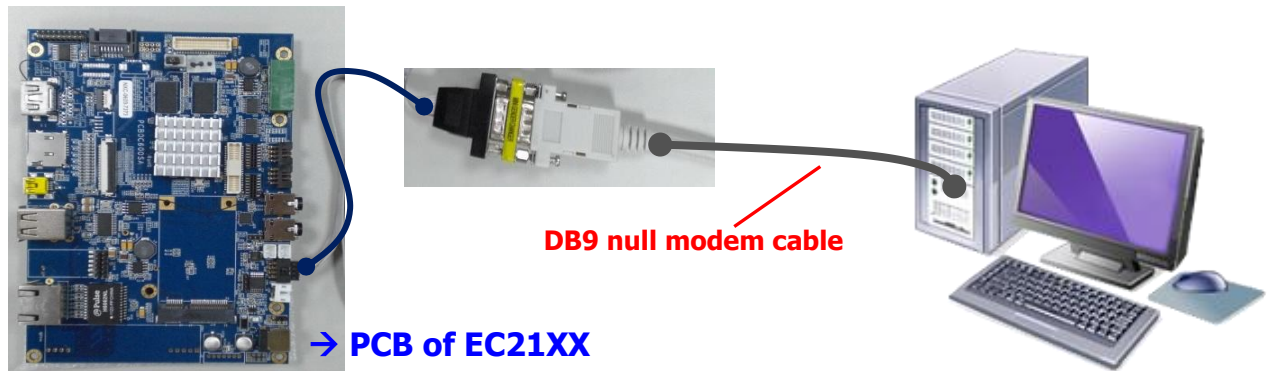
EC2110

Connect a RS232 IDC cable to the debug port

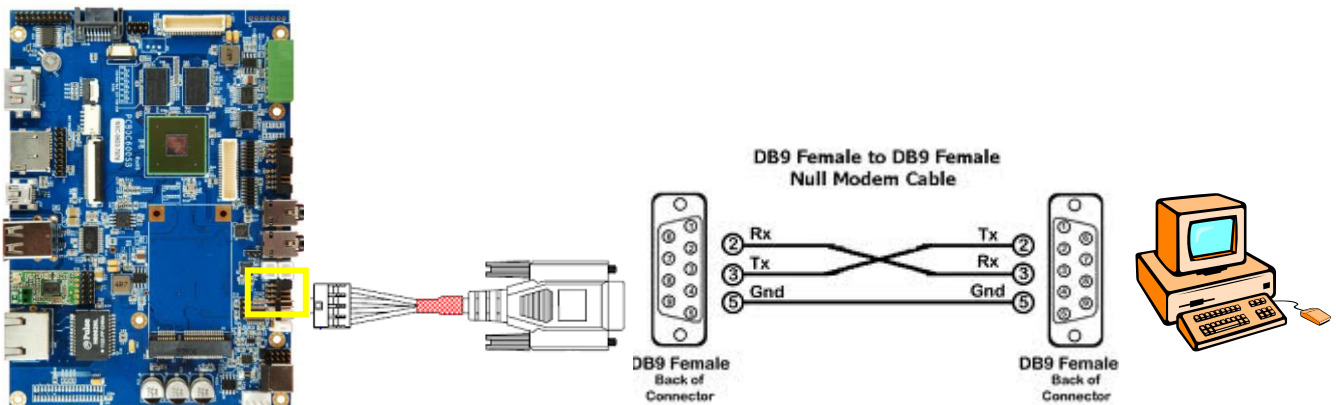


Connect to PC

Turn on the PC, run the terminal program, and open the COM port. We use **TeraTerm**. You can find this tool and the user guide on our wiki page online.



Console / Debug Port Connection Diagram



UART1 is dedicated as the debug port. UART1 default settings are **Baud Rate 115200, 8 data bits, no parity, 1 stop bit and no flow control.**

A DB9 **null modem cable** (or adapter) is required when you want to connect UART1 to a PC with terminal emulation software such as TeraTerm.

Pin Assignment: RS232 IDC Cable and Null Modem Adapter

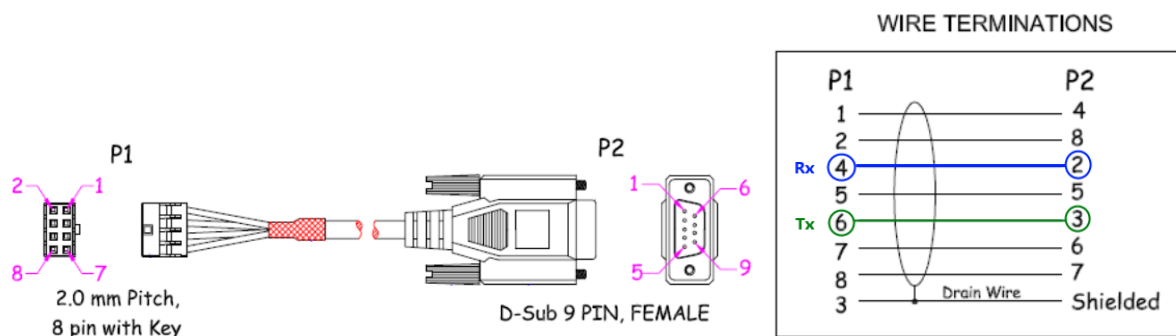
We prefer to use this kind of adapter if your DB9 cable is straight through cable. (Not cross-connected)

- ☐ Console box header (CN3 on PCB) pin assignment

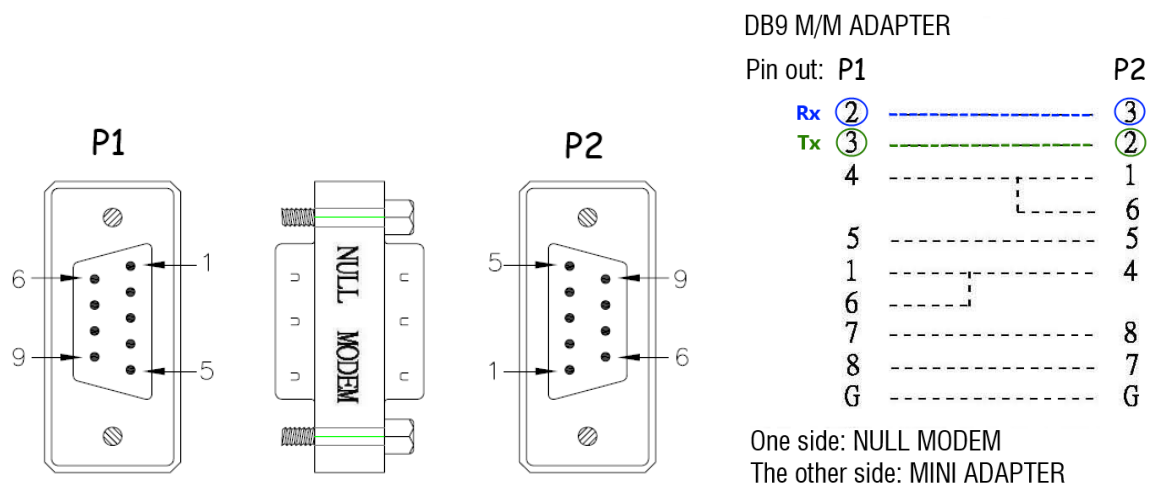


2 : CTS 6 : Tx
 4 : Rx 8 : RTS

- RS232 IDC cable: pin header for DB9



- Null modem adapter: male-to-male for DB9



3.2 Start Running

After connecting to the debug port, please power on the device to start.

1. Before logging into the system, you can enter into the U-Boot environment to check some variables. After powering on, quickly hit the **Enter** key within 3 seconds when you see the message below.

Hit any key to stop autoboot: 3

2. Type **'?'** or **'help'** to get all U-Boot commands and more details.
3. Type **printenv** on the U-Boot shell to see the current environmental variables

```
> printenv
```

4. Use the command below to modify an environment variable.

```
> setenv variable 'string value'
```

If you need to change your output display, do the following command to set the variable **"panel"**; otherwise, we do not recommend you doing this step.

- Android

```
//HDMI (1920 x 1080)
> setenv panel 'video=mxcfb0:dev=hdmi,1920x1080M@60,if=RGB24,bpp=32 video=mxcfb1:off cea'
//7" LCD-PT (800 x 480) with RTP
> setenv panel 'video=mxcfb0:dev=lcd,LCD-WVGA,if=RGB24,bpp=32 video=mxcfb1:off'
//7" LCD-AWT (800 x 480) with RTP
> setenv panel 'video=mxcfb0:dev=lcd,AWT-WVGA,if=RGB24,bpp=32 video=mxcfb1:off'
```

- Ubuntu

```
//HDMI (1920 x 1080)
> setenv panel 'video=mxcfb0:dev=hdmi,1920x1080M@60,if=RGB24 video=mxcfb1:off'
//7" LCD (800 x 480) with RTP
```

```
> setenv panel 'video=mxcfb0:dev=lcd,LCD-WVGA,if=RGB24 video=mxcfb1:off'  
//7" LCD-AWT (800 x 480) with RTP  
> setenv panel 'video=mxcfb0:dev=lcd,SEIKO-WVGA,if=RGB24 video=mxcfb1:off'
```

★ For other LCD / LVDS panels we support, please contact to our sales in order to get right string value.

5. Use the command **saveenv** to save the environment variables that you have modified. If not saved, then any changes to the variables will not persist after a restart.

```
> saveenv  
Saving Environment to SPI Flash...  
Erasing SPI flash...Writing to SPI flash.....SUCCESS  
  
done
```

6. Use the command **boot** to start the operating system.

```
> boot
```

3.3 Connecting to the PC (for Android 4.2)

For Android systems, connect to a Windows PC by following the steps below:

- ☐ Install Android SDK
- ☐ Connect to PC with USB OTG cable
- ☐ Install USB driver

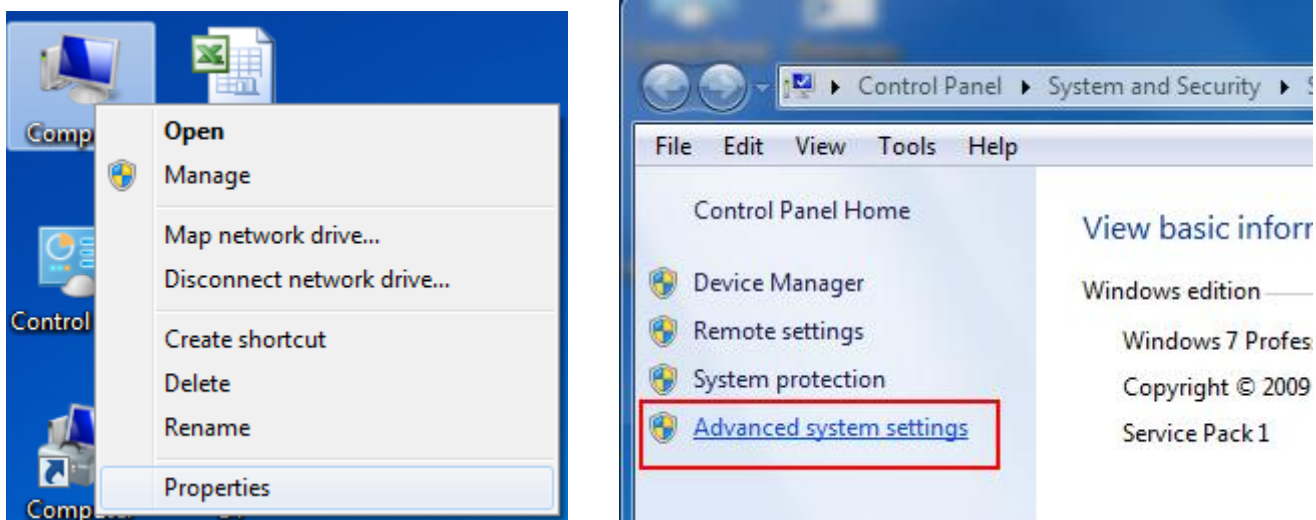
3.3.1 Installing the Android SDK

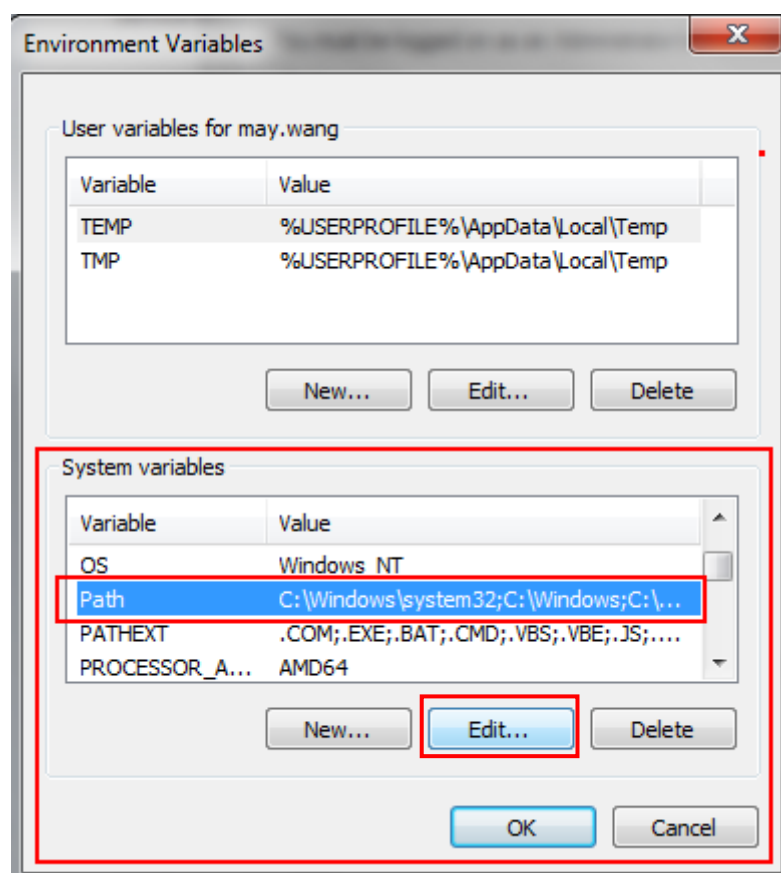
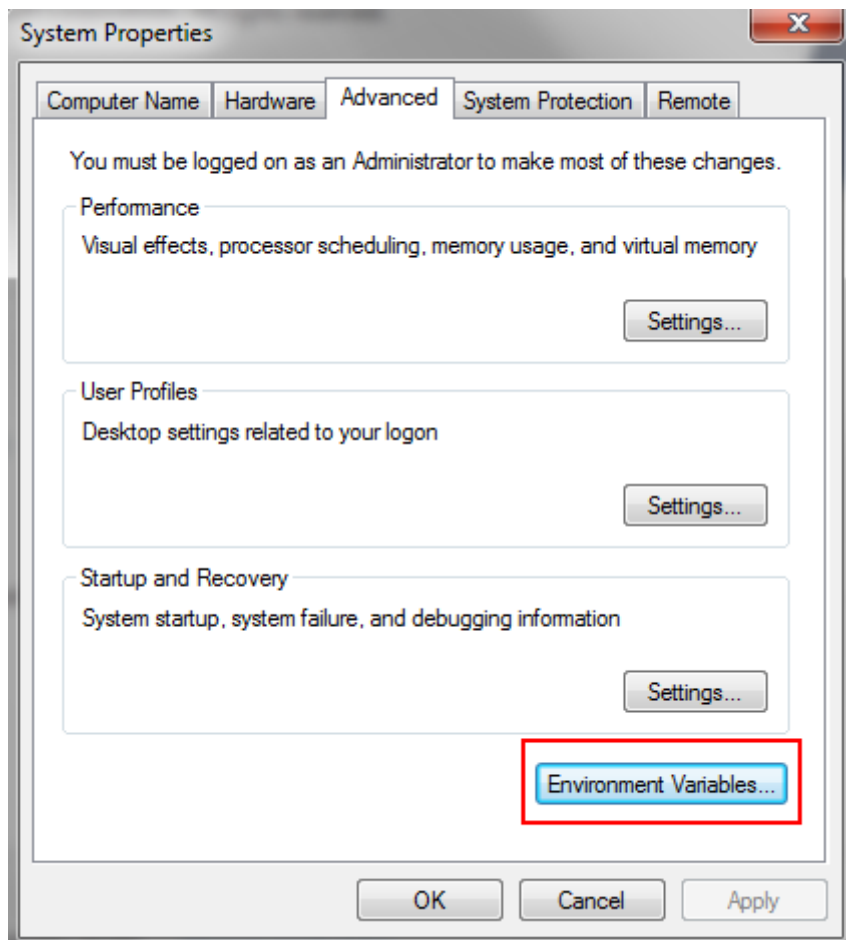
This chapter is intended for developers to quickly setup an EC series and know how to launch EC series demo applications from a host PC via a USB interface. The host PC requires a Windows system (for example Windows XP or 7) and at least a few gigabytes free disk space. The first time you launch a demo app from the host PC, you will have to install a driver into the host PC.

Follow the steps below to install the driver and make the hardware connection:

1. Insert the software DVD into the host PC and find the Android SDK folder. Copy to the host PC in a folder named **<SDK>**
2. Add the **<SDK>** folders to the **path** environment variable of the host PC:
; <SDK>\tools; <SDK>\platform-tools
3. The following figures illustrate the steps to add a path on a Windows 7 PC:
Computer > Properties > Advanced system settings > Environment Variables... > Path > Edit...

Screenshots for Windows 7

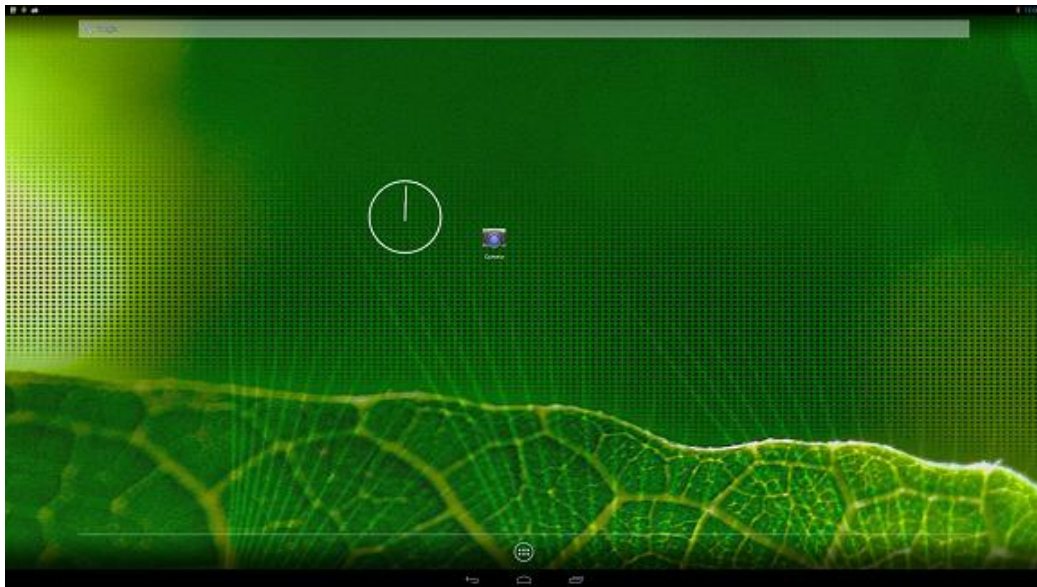




3.3.2 Connecting to the PC with a USB OTG cable

Connect the power adapter to an EC21XX power jack and plug the power adapter to an AC outlet.

Wait for the EC21XX to boot up and show a standard desktop on the LCD screen. A standard Android desktop is similar to the following figure.



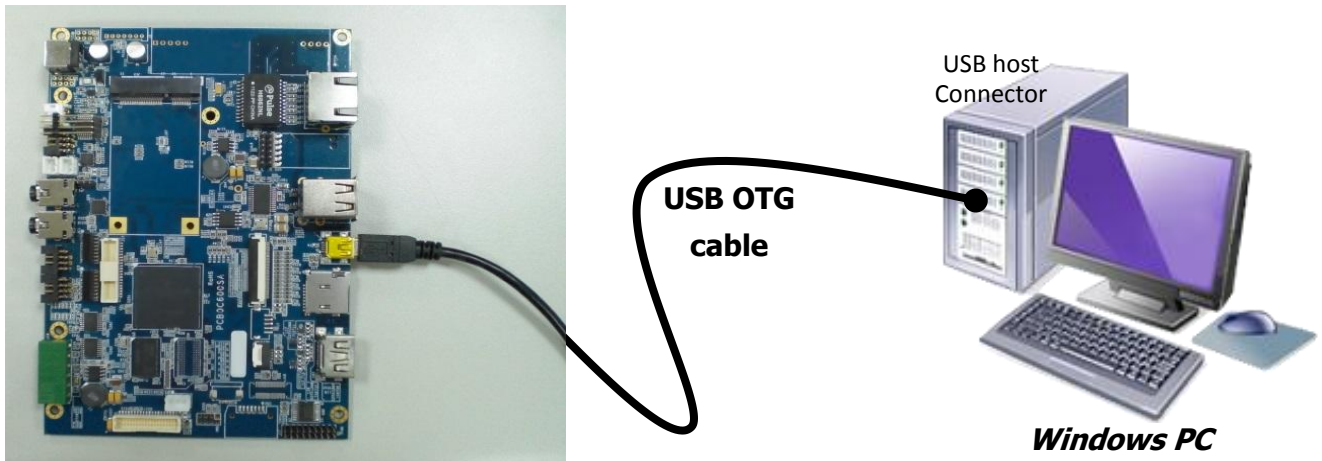
Connect a USB cable to EC21XX mini-USB connector.



Connect the USB cable to a USB port on the host PC.



The following diagram shows a USB connected EC series and a Windows PC.



3.3.3 Installing the USB Driver (Android ADB Interface)

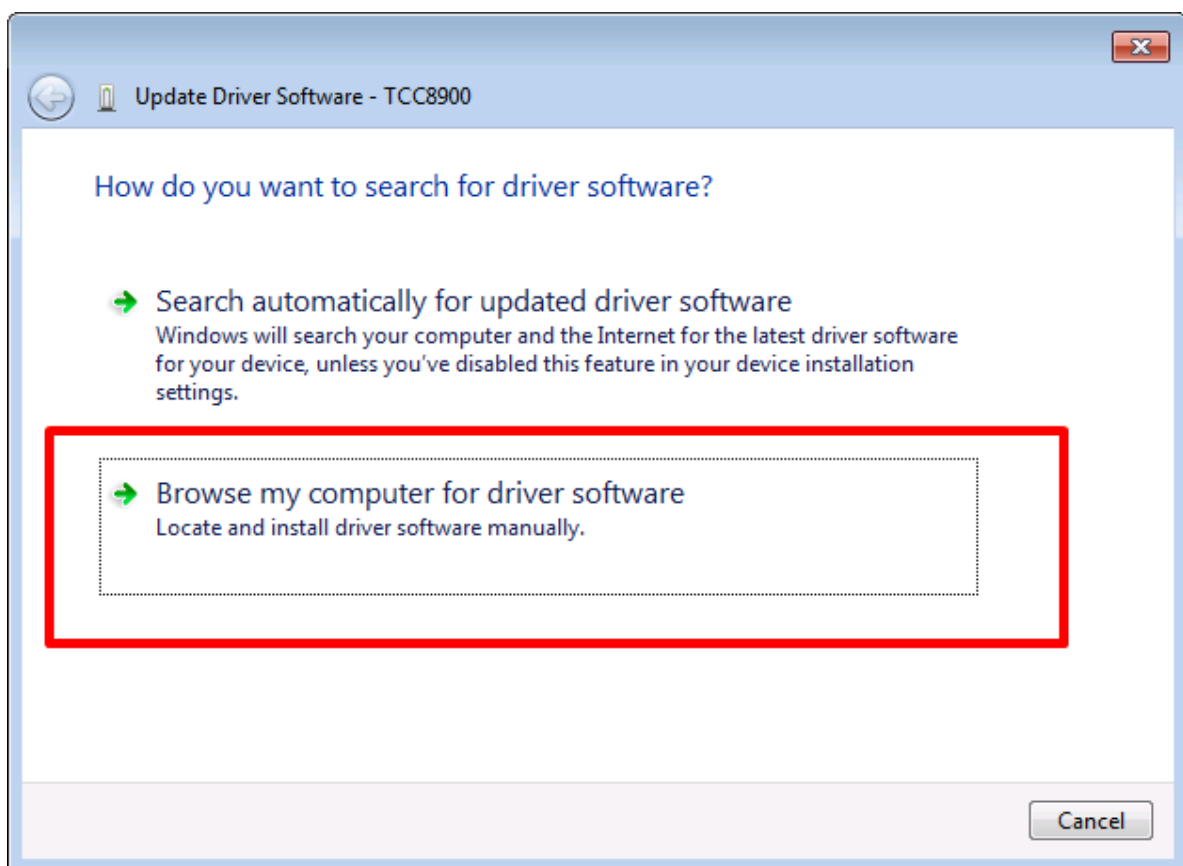
When first connected, the host PC will prompt you about detecting an unknown USB device and ask you to install a driver. Choose to install software from a specific location.

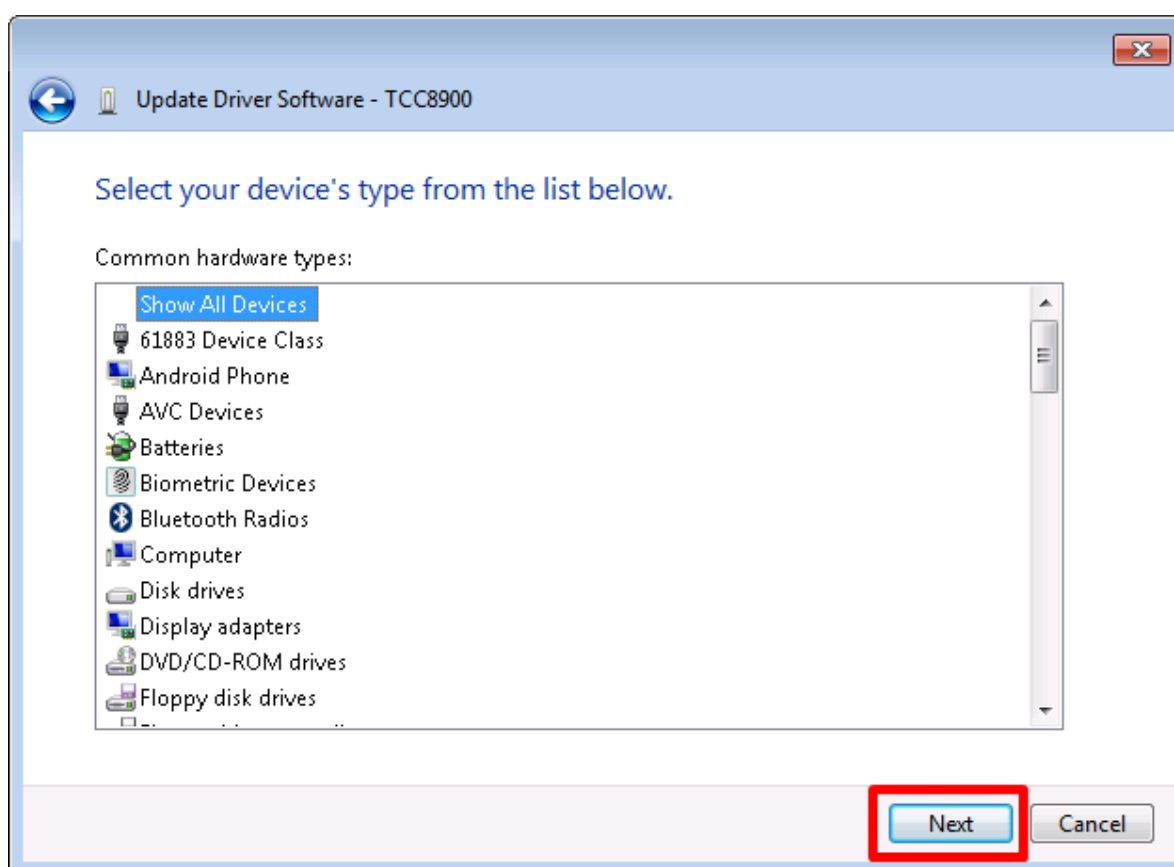
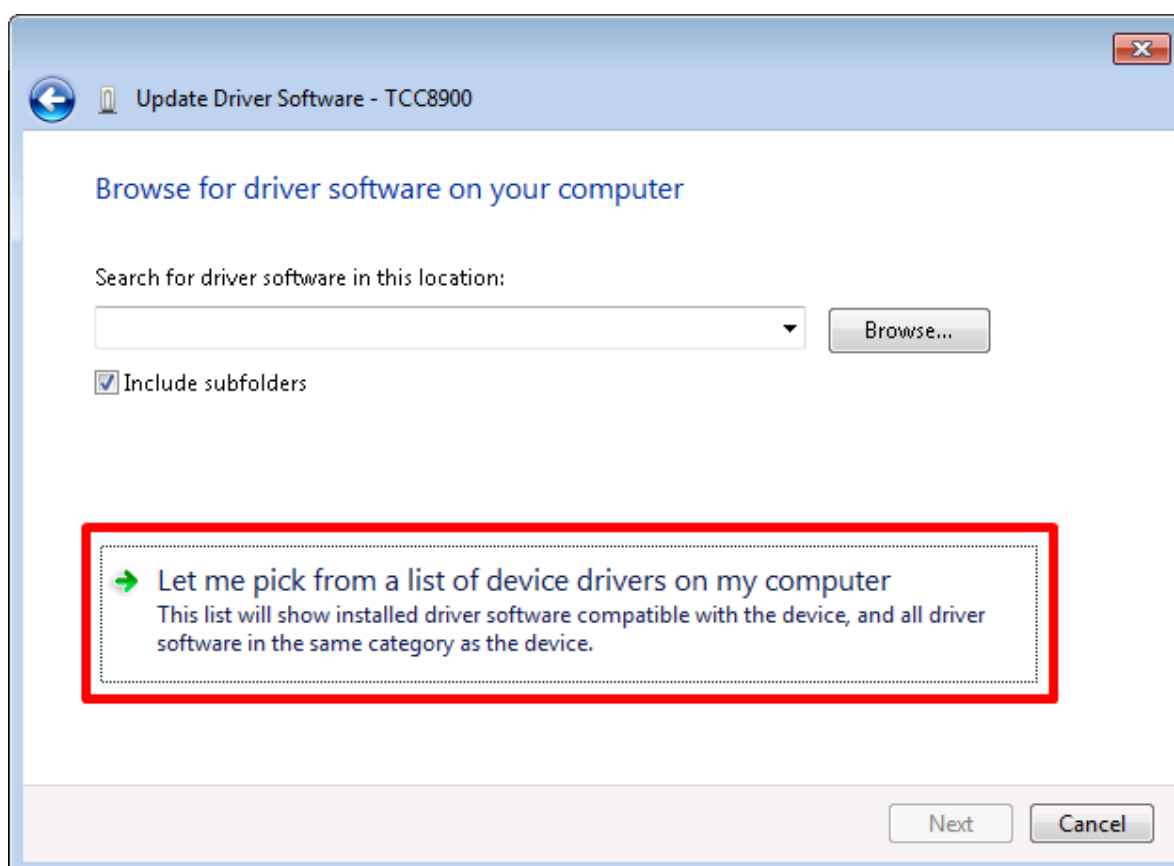
Find the USB driver in the software DVD and copy it to the host PC. Add the path of Android USB driver as the search path for the wizard:

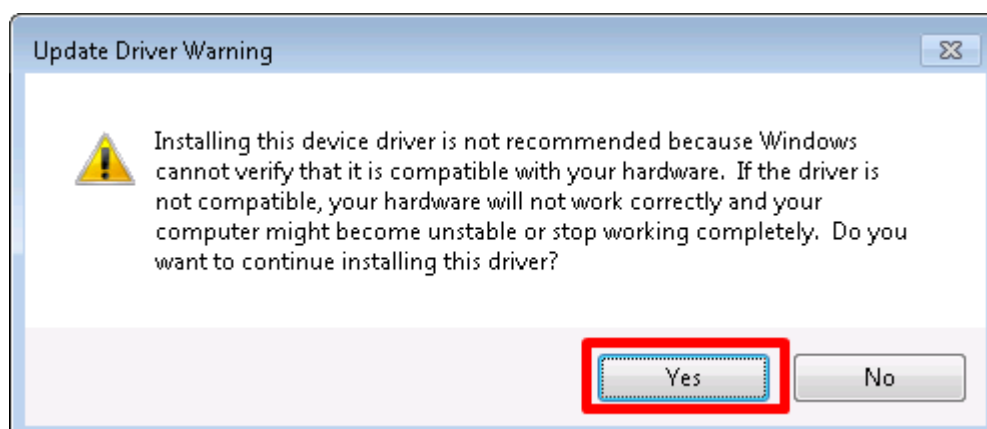
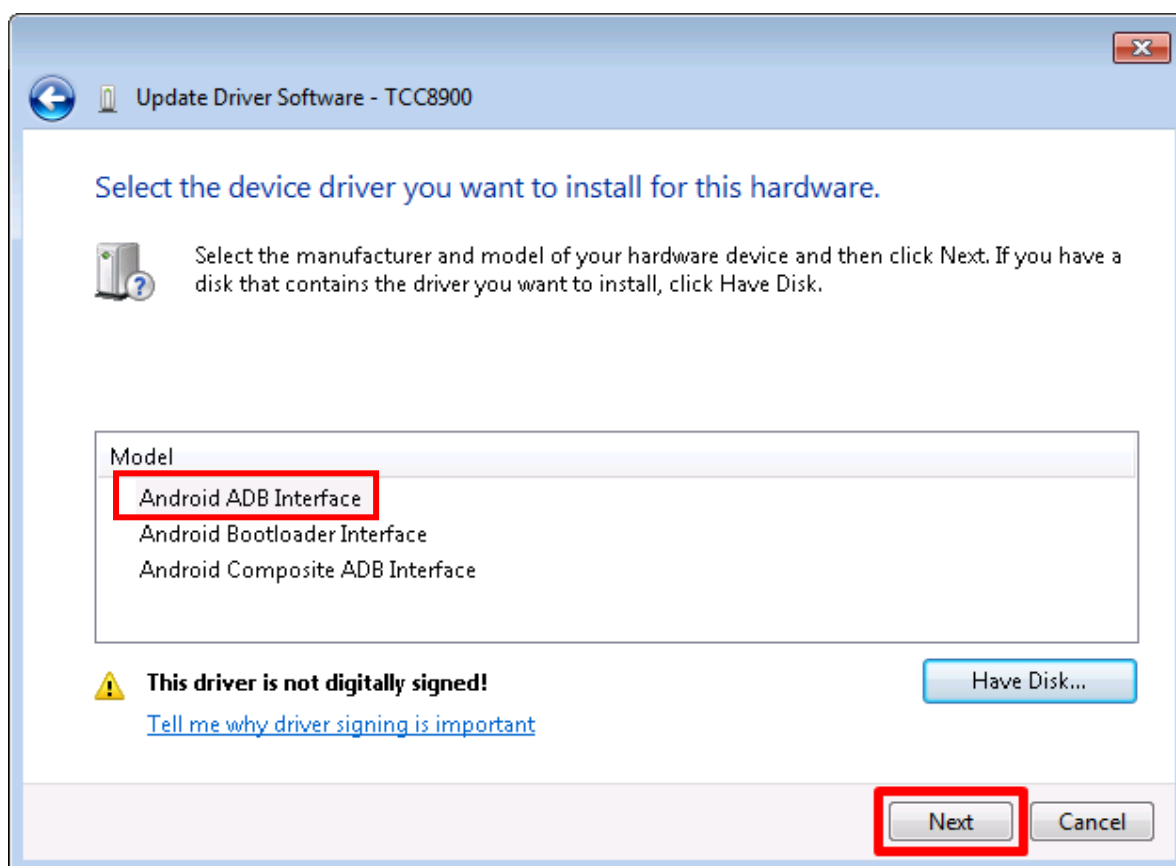
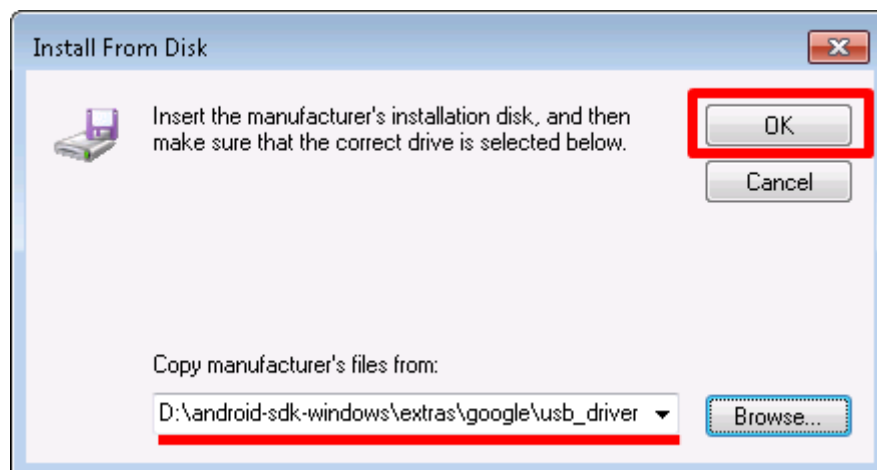
;D:\android-sdk-windows\extras\google\usb_driver

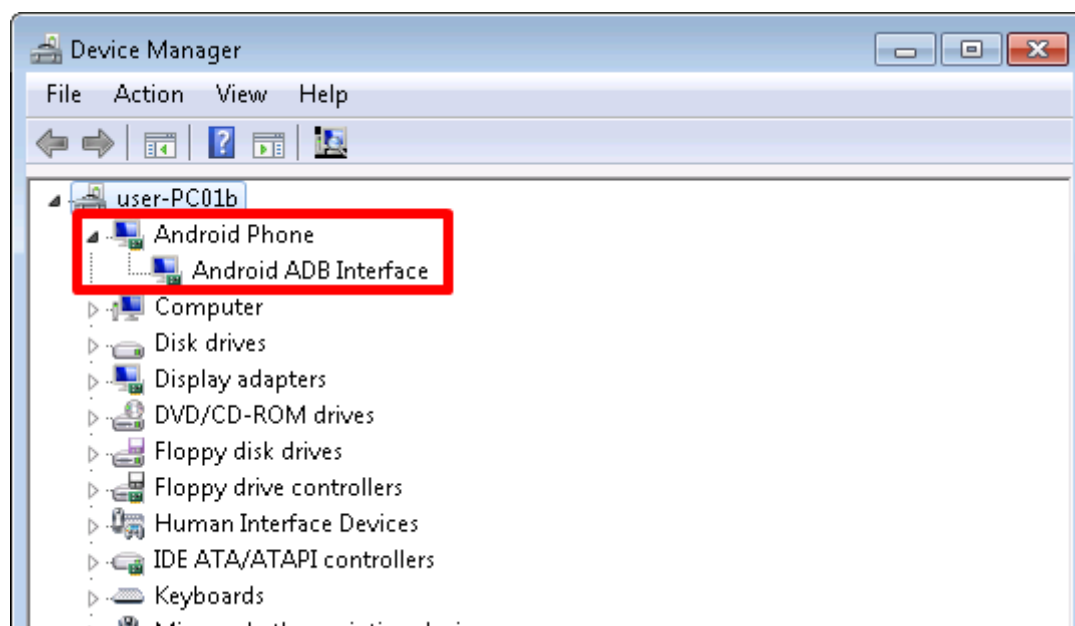
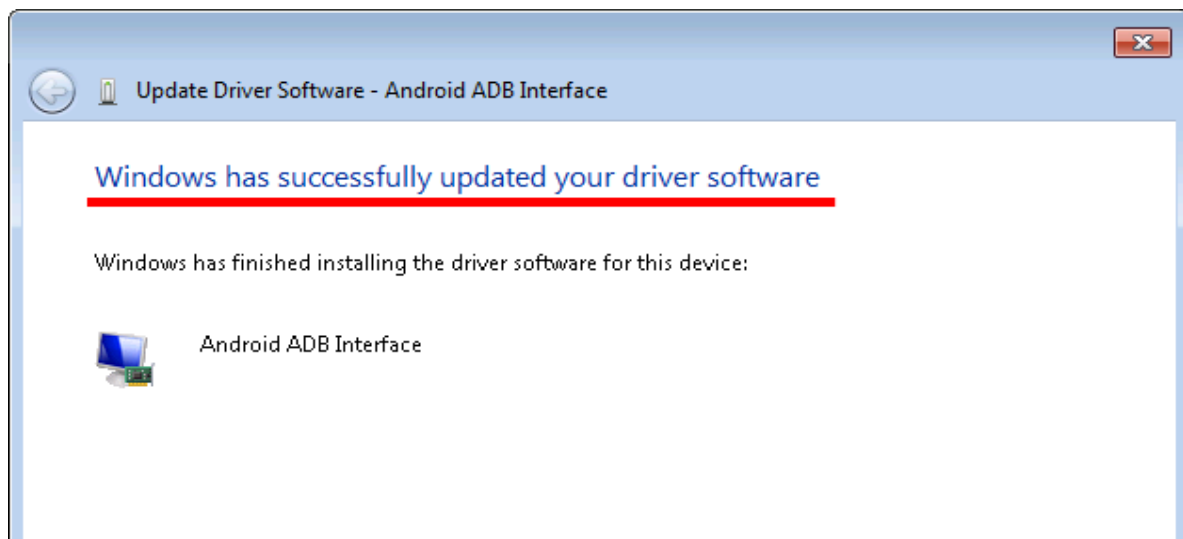
After the driver is successfully installed, you will find an “Android Phone” with the Android ADB interface in Device Manager.

Screenshots for Windows 7

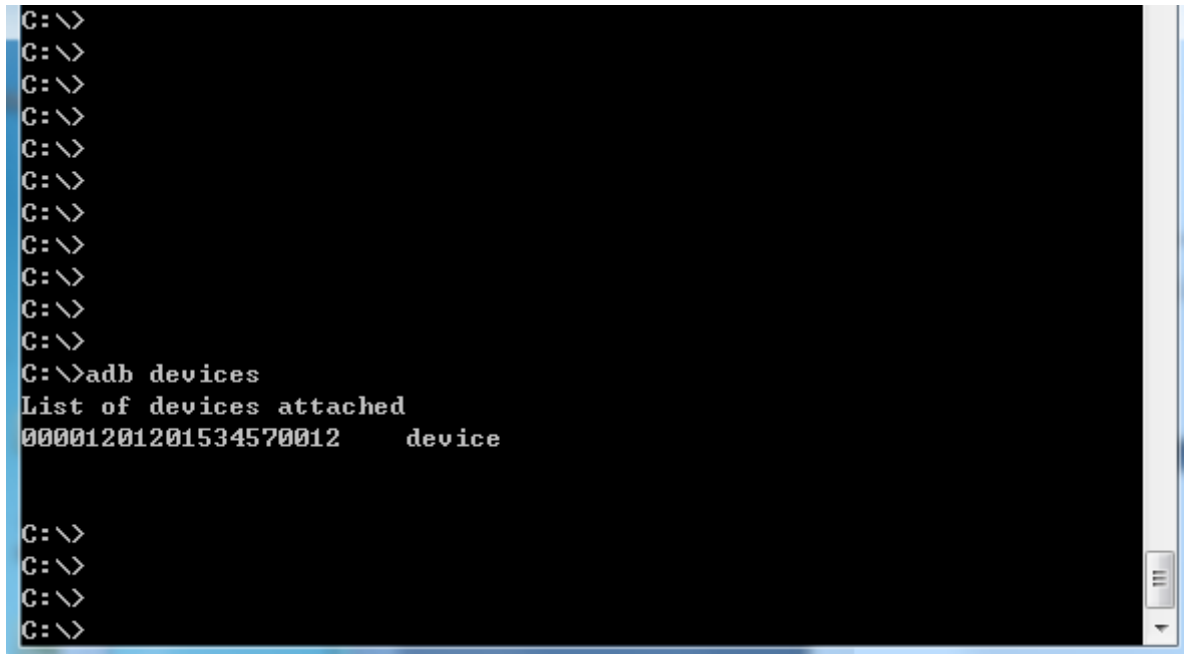








To verify whether or not the driver is correctly installed, you can type `adb devices` at the Windows command prompt. The attached EC21XXdevice will be listed with the device ID. If it does not show any attached devices, repeat the previous steps to install the driver again.

A screenshot of a Windows command prompt window. The window has a black background with white text. The prompt is 'C:\>'. The user has entered the command 'adb devices'. The output shows 'List of devices attached' followed by a single line: '00001201201534570012 device'. Below this, there are four more 'C:\>' prompts without any further input or output.

```
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>adb devices
List of devices attached
00001201201534570012    device

C:\>
C:\>
C:\>
C:\>
```

Now you can use the **adb** command at the command line to manage your connected device. This allows you to copy files/directories to or from the device, run a remote shell, install files/apps to the device, and can even run applications you develop directly through an emulator.

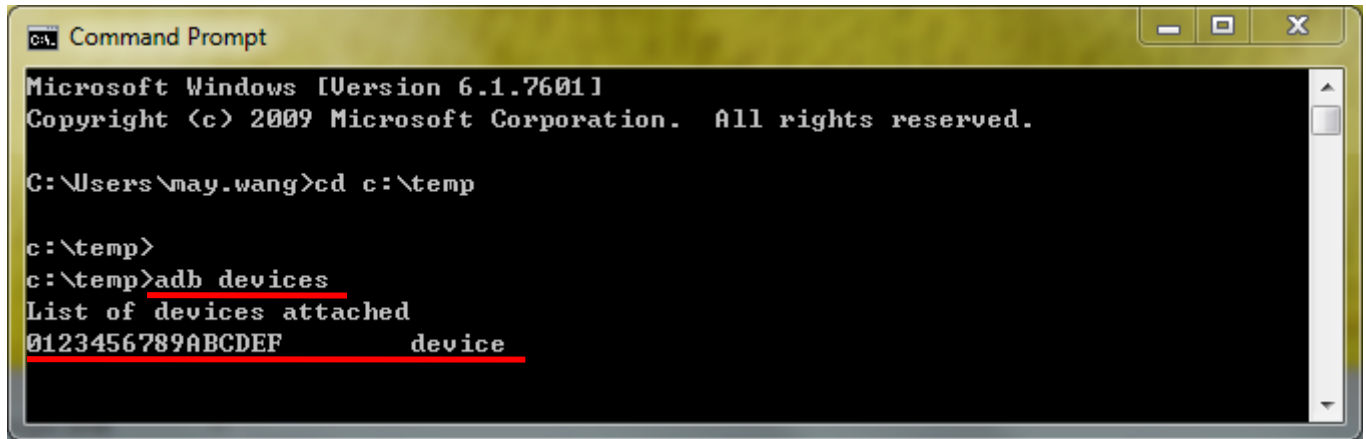
We list some basic ADB command options here. [This link](#) has more details.

<code>adb push <local> <remote></code>	Copy file/dir to device
<code>adb pull <remote> [<local>]</code>	Copy file/dir from device
<code>adb sync [<directory>]</code>	Copy host->device only if changed (-l means list but don't copy)
<code>adb shell</code>	Run remote shell interactively

3.3.4 Installing Extra Apps

This part shows you how to install extra apps from the PC to the connected device.

After installing the Android ADB interface, connect the EC21XX device to the PC and verify that the device is attached.



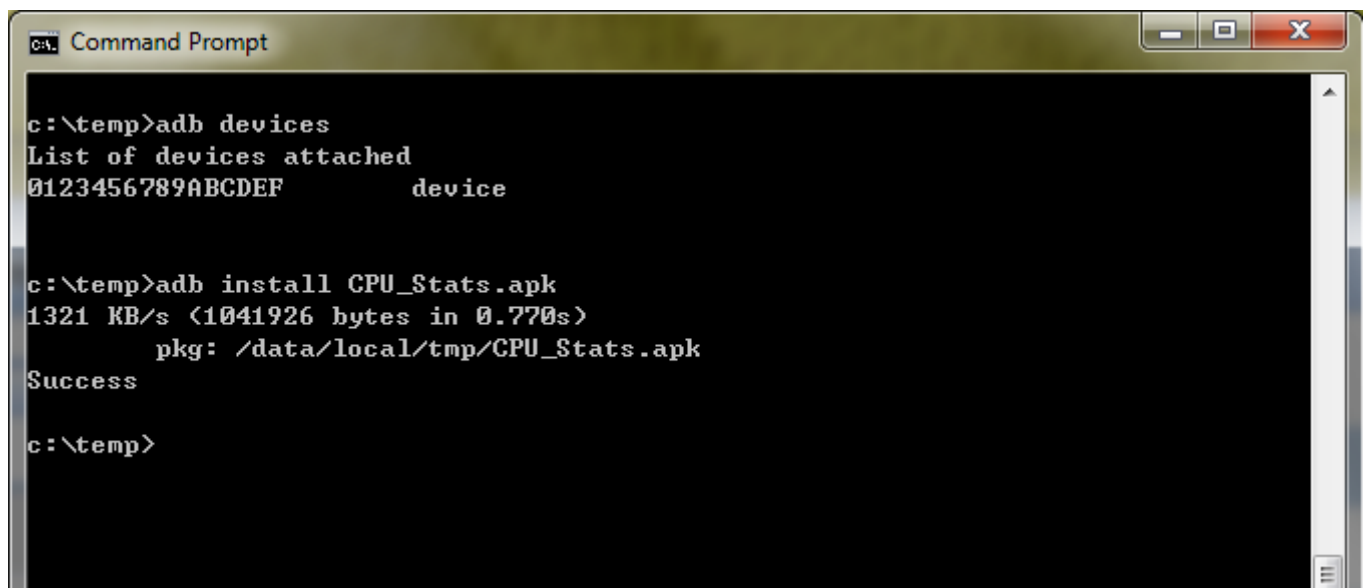
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\may.wang>cd c:\temp

c:\temp>
c:\temp>adb devices
List of devices attached
0123456789ABCDEF      device
```

The install file should be an **.apk** file. Type the command:

adb install <APK_file>



```
c:\temp>adb devices
List of devices attached
0123456789ABCDEF      device

c:\temp>adb install CPU_Stats.apk
1321 KB/s (1041926 bytes in 0.770s)
    pkg: /data/local/tmp/CPU_Stats.apk
Success

c:\temp>
```

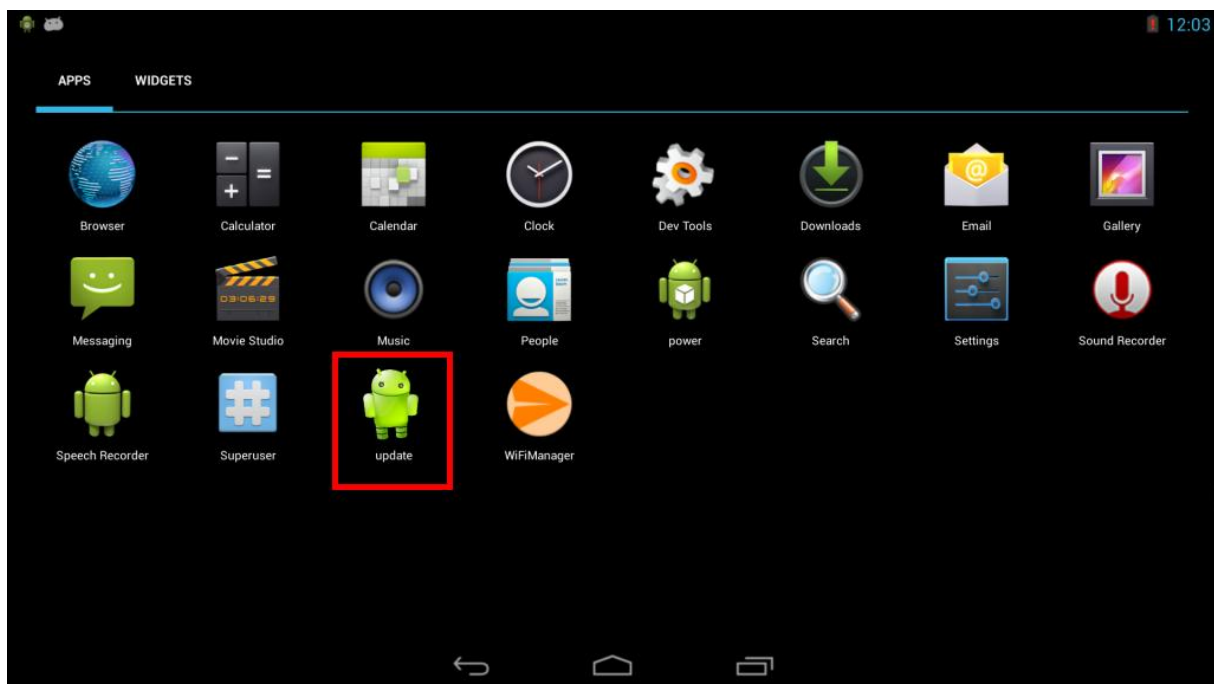
When it shows Success, your app has installed successful.

3.4 Update firmware via USB dongle (for Android 4.2)

This section shows you how to update the firmware easily via USB dongle. You only need to prepare the necessary image file and script file to achieve it.

- Hardware preparation:
 1. Empty USB mass storage device (USB dongle) or empty MicroSD card.
- Software preparation:
 1. The script file
(This can be found in your resource CD which is attached to the shipment and the files are under the folder: \Tool\Android\System Update\)
 2. Latest firmware image file.
(This can be found in your resource CD which is attached to the shipment and the files are under the folder: \Binary Images\)

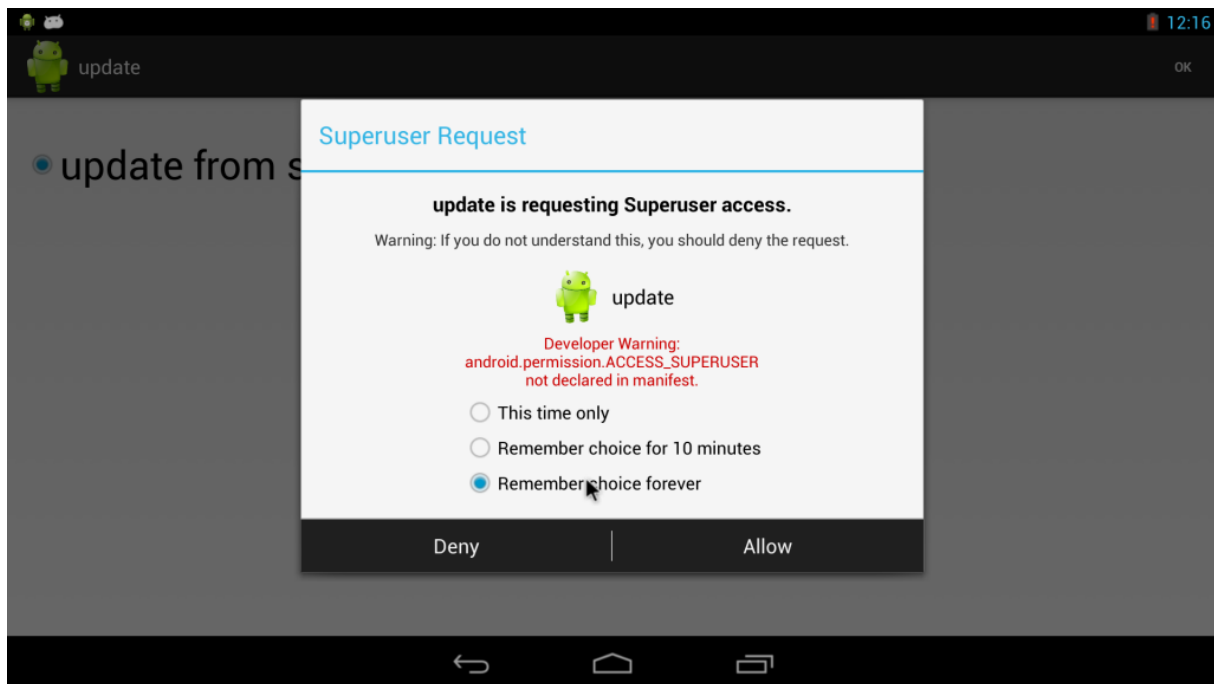
In your system application programs collection, there is an APP named “update” which is for you updating the system firmware more easily.



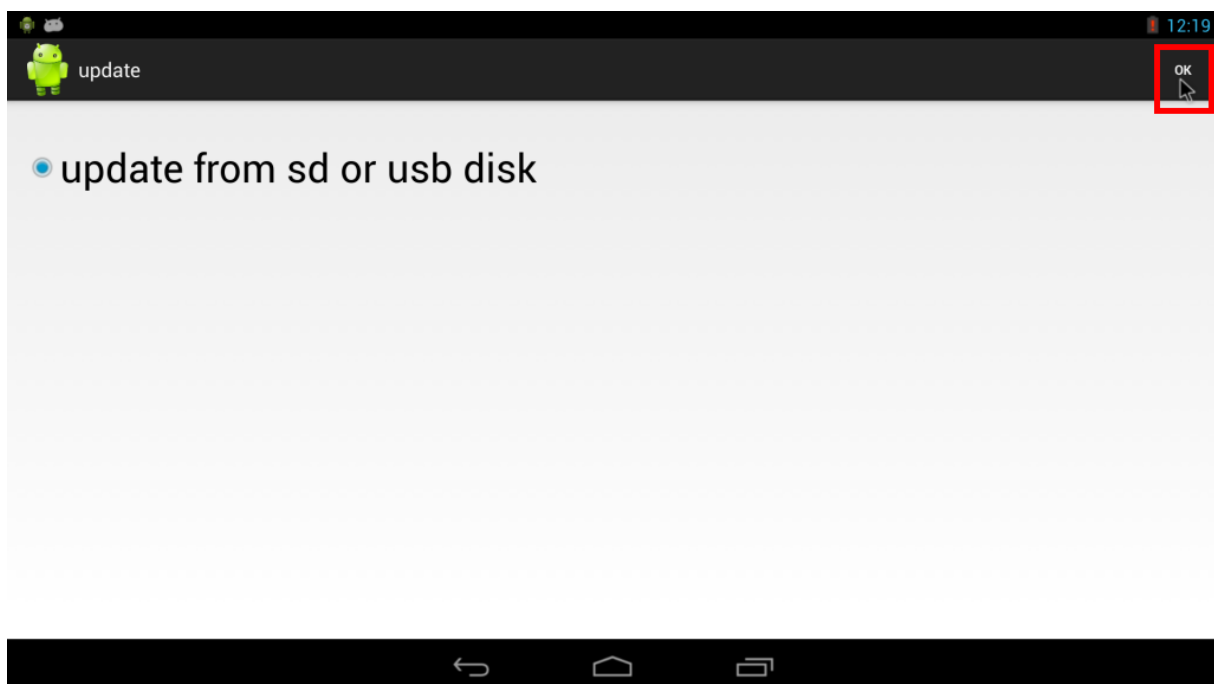
Step 1: copy the system update files(\Tool\Android\System Update\) into the empty USB mass storage or microSD card.

Step 2: copy the latest image files(\Binary Images\) into the empty USB mass storage or MicroSD card.

Step 3: Execute the “update” app and the superuser authorization dialog will pop-up, select the “**remember choice forever**” and click the button “**Allow**”.



Step 4: Insert the USB mass storage or MicroSD card and press “OK”.



Step 5: The system will be restart and update the firmware after finished the process.

Step 6: Unplug the power adaptor to reboot the device, then the device is accomplished the system update procedure.

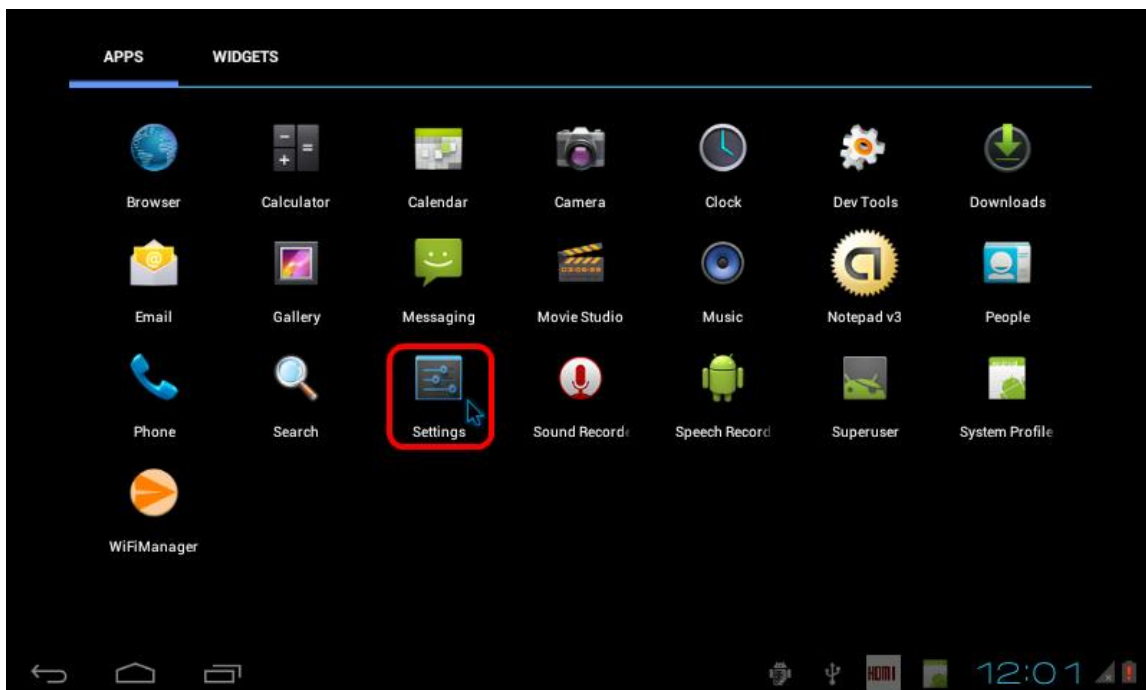
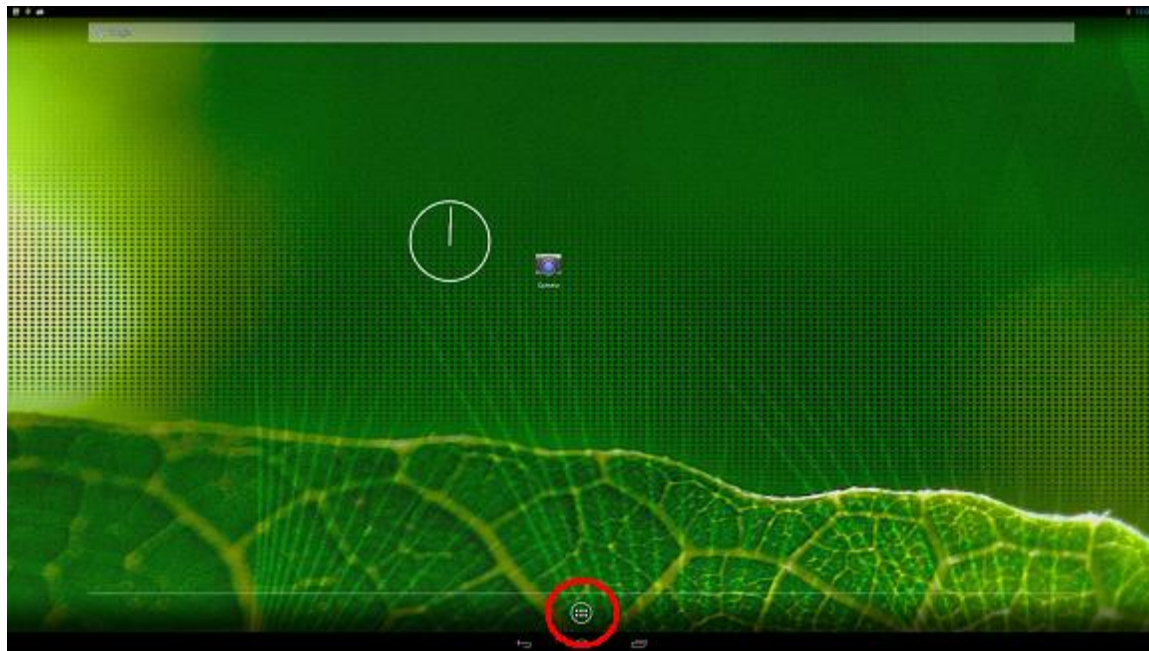
4. Running Software

4.1 Android

4.1.1 Settings

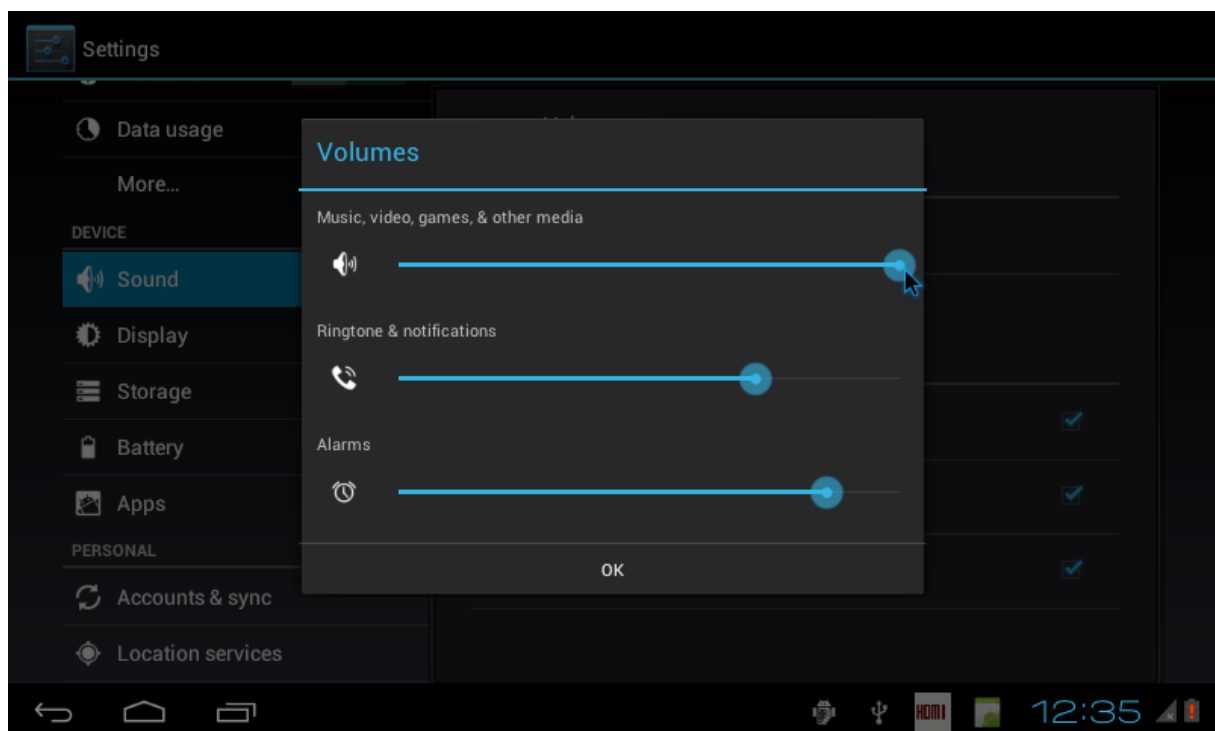
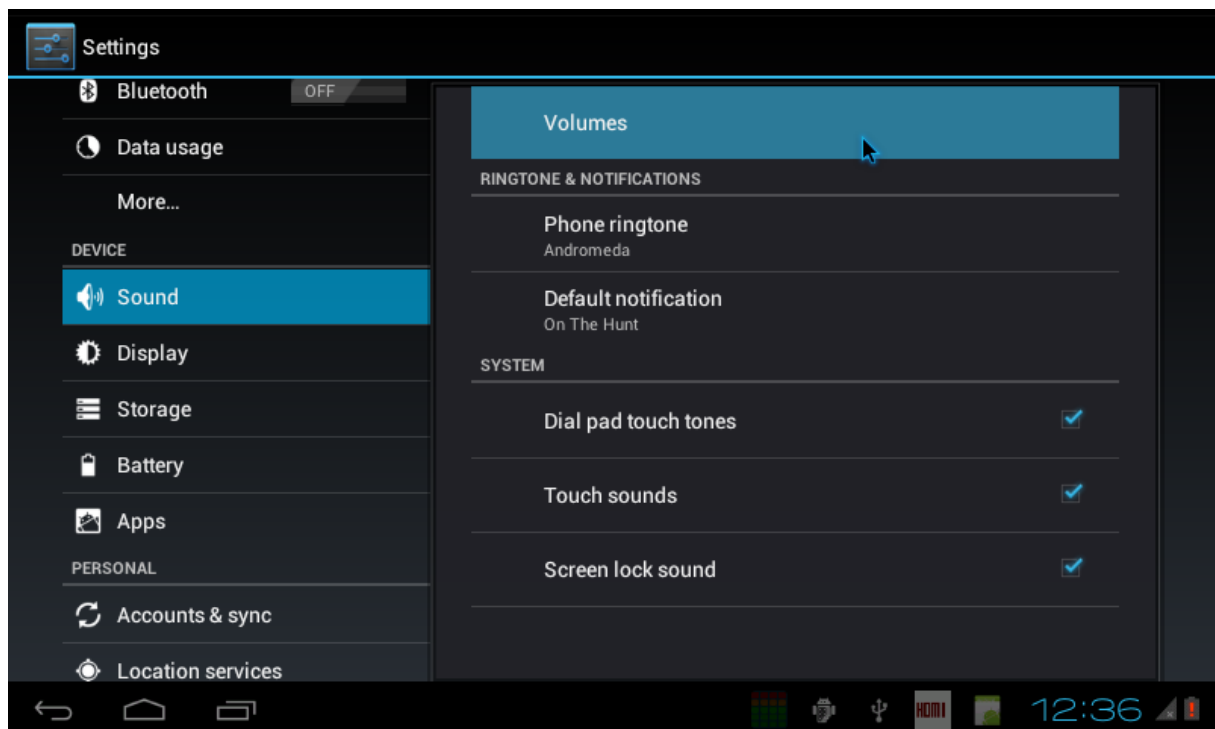


Click the app drawer icon .



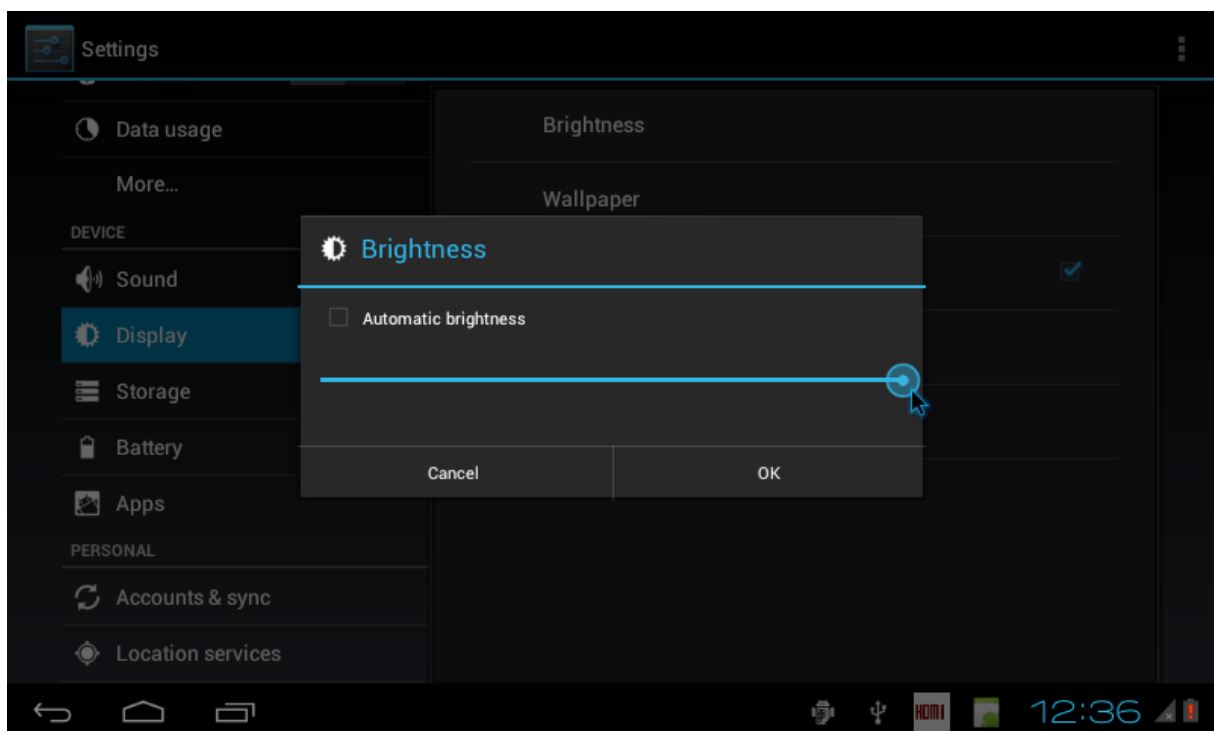
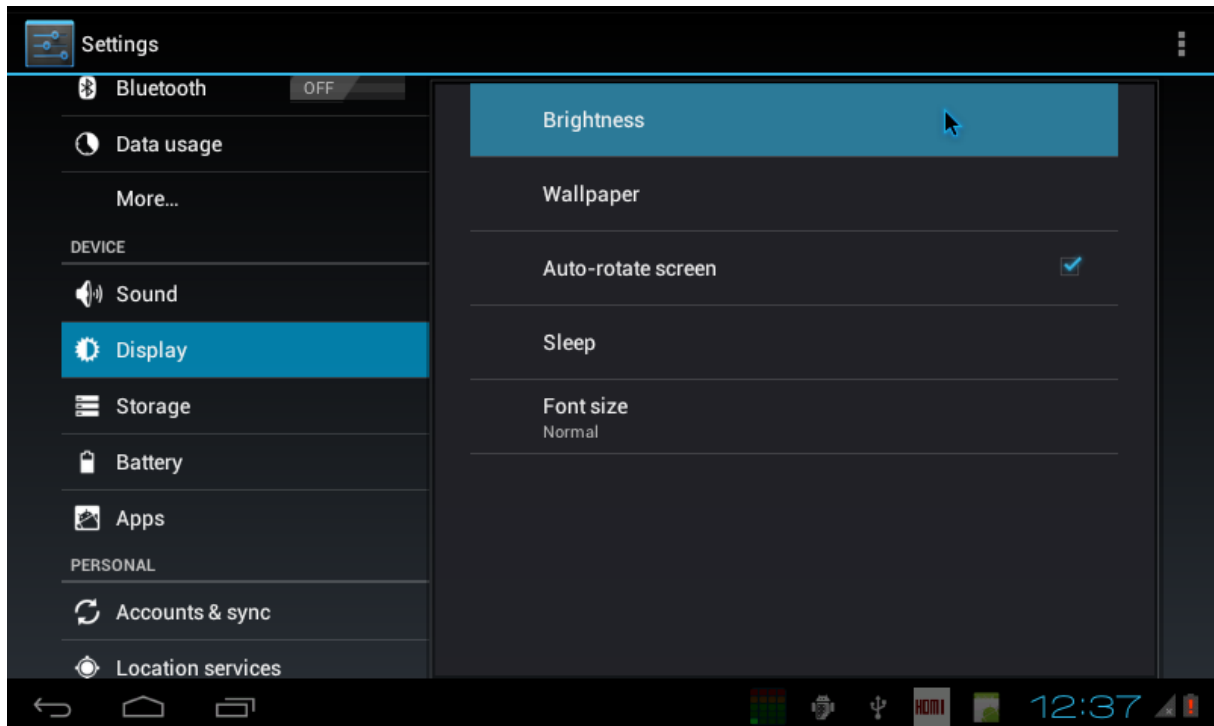
Sound

Adjust the volume



Display

Adjust the brightness, if supported by the panel.

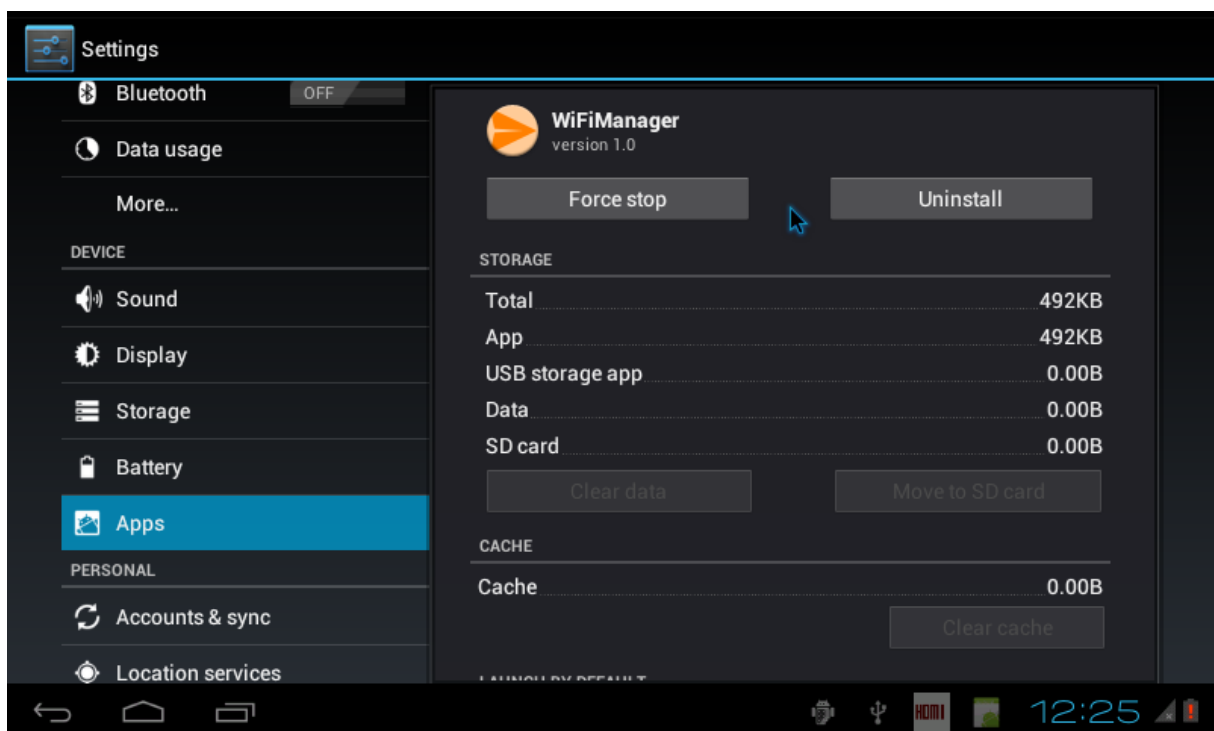
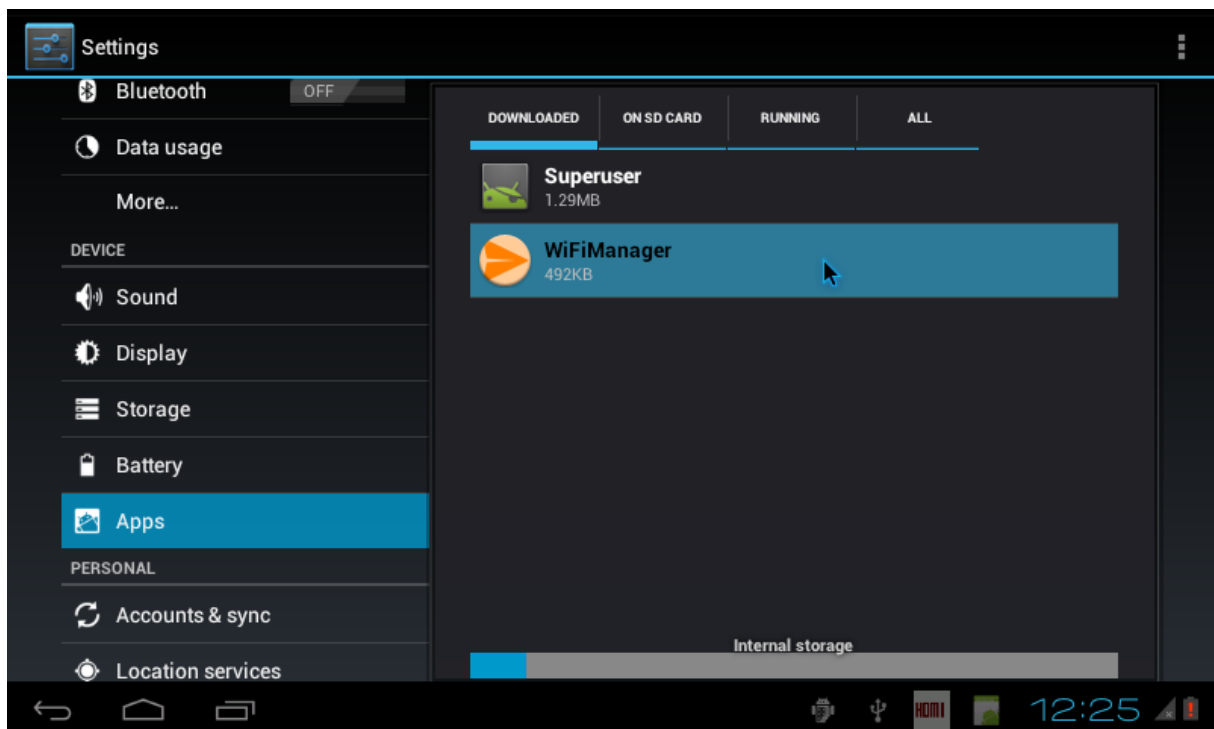


Note: In our Android system, like most of normal Android, there is the sleep mode for power manages.

You can set the time out of sleep mode in Settings > Display > Sleep. The minimum is 15 s. When time is up, the screen will be dim.

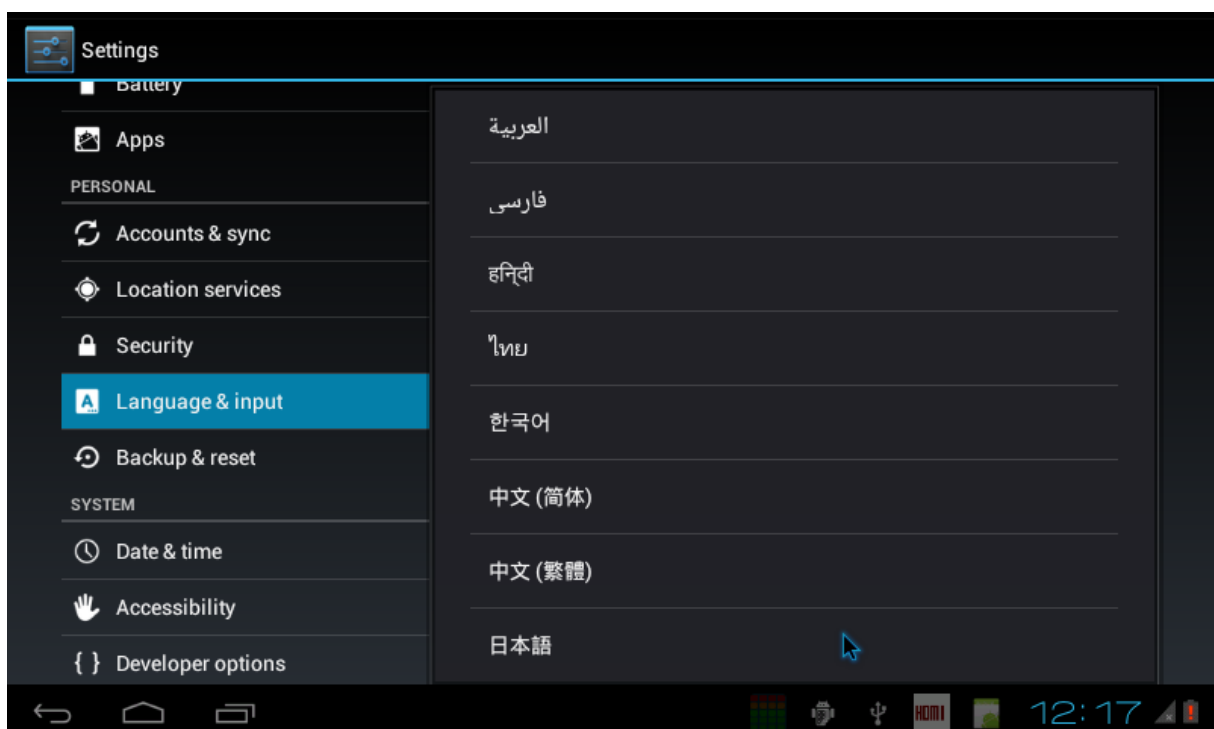
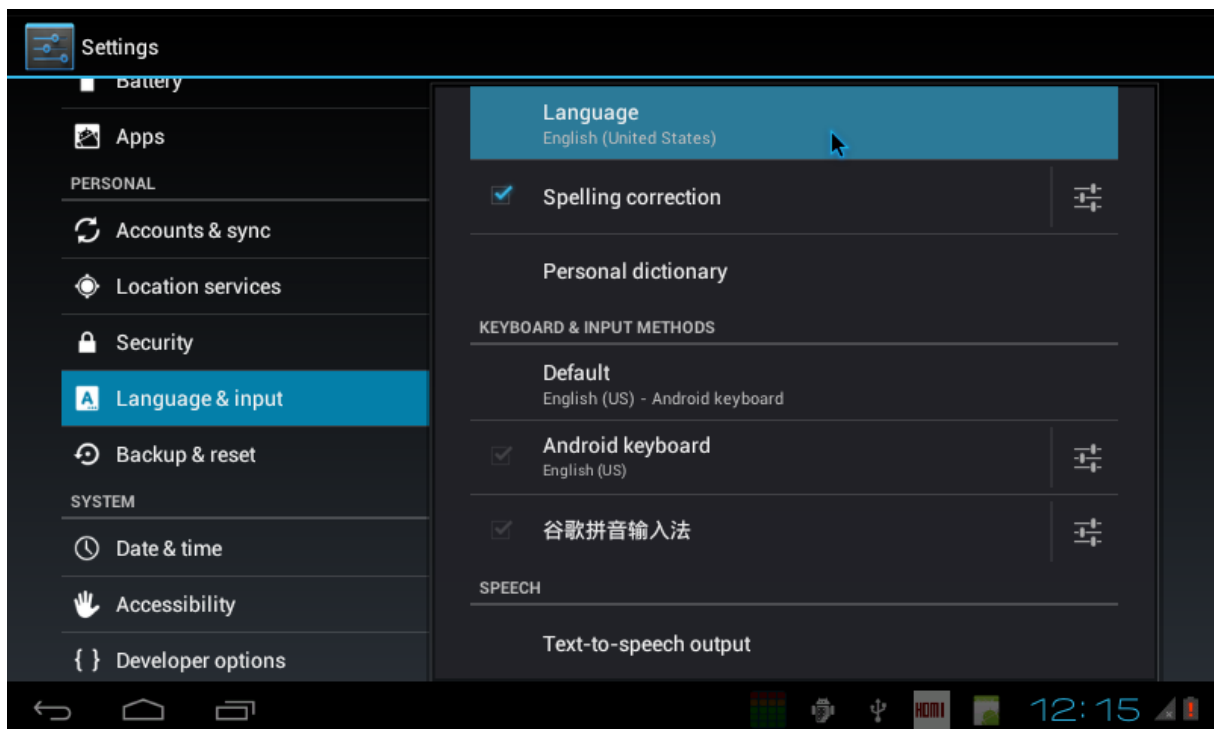
Apps

Manage all apps. You can force-stop or uninstall an app that you have installed.



Language & Input

You can change the UI display language and the default input methods.

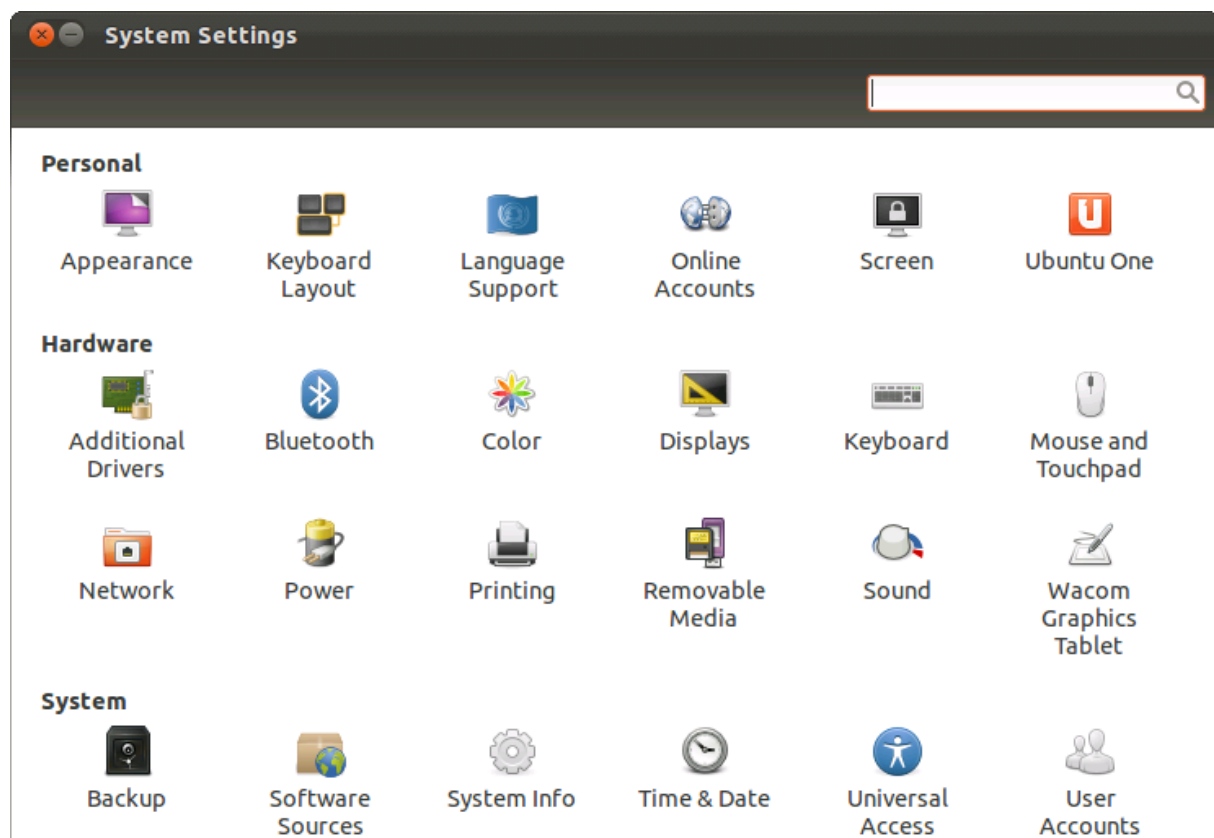
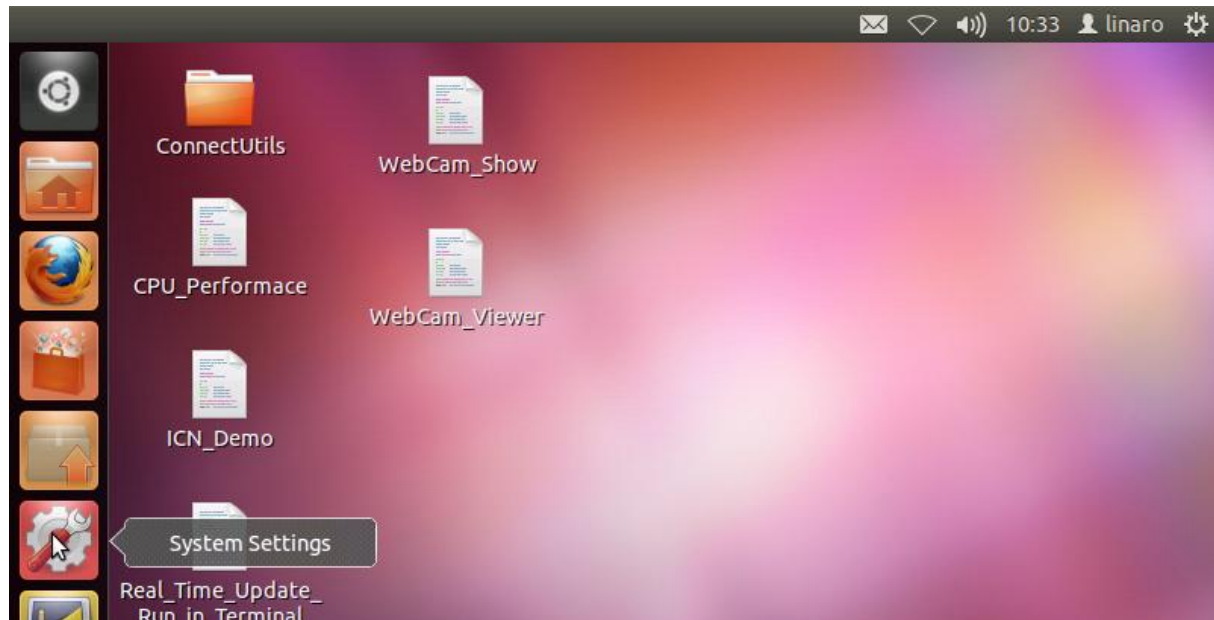


4.2 Ubuntu Linux

Root Password: *linaro*

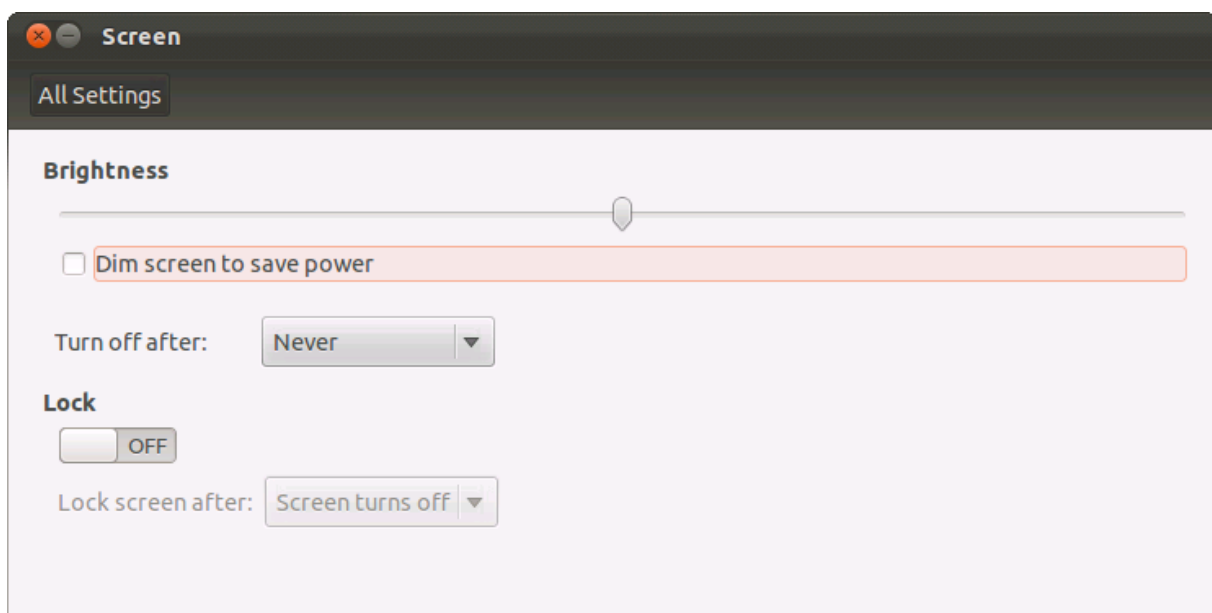
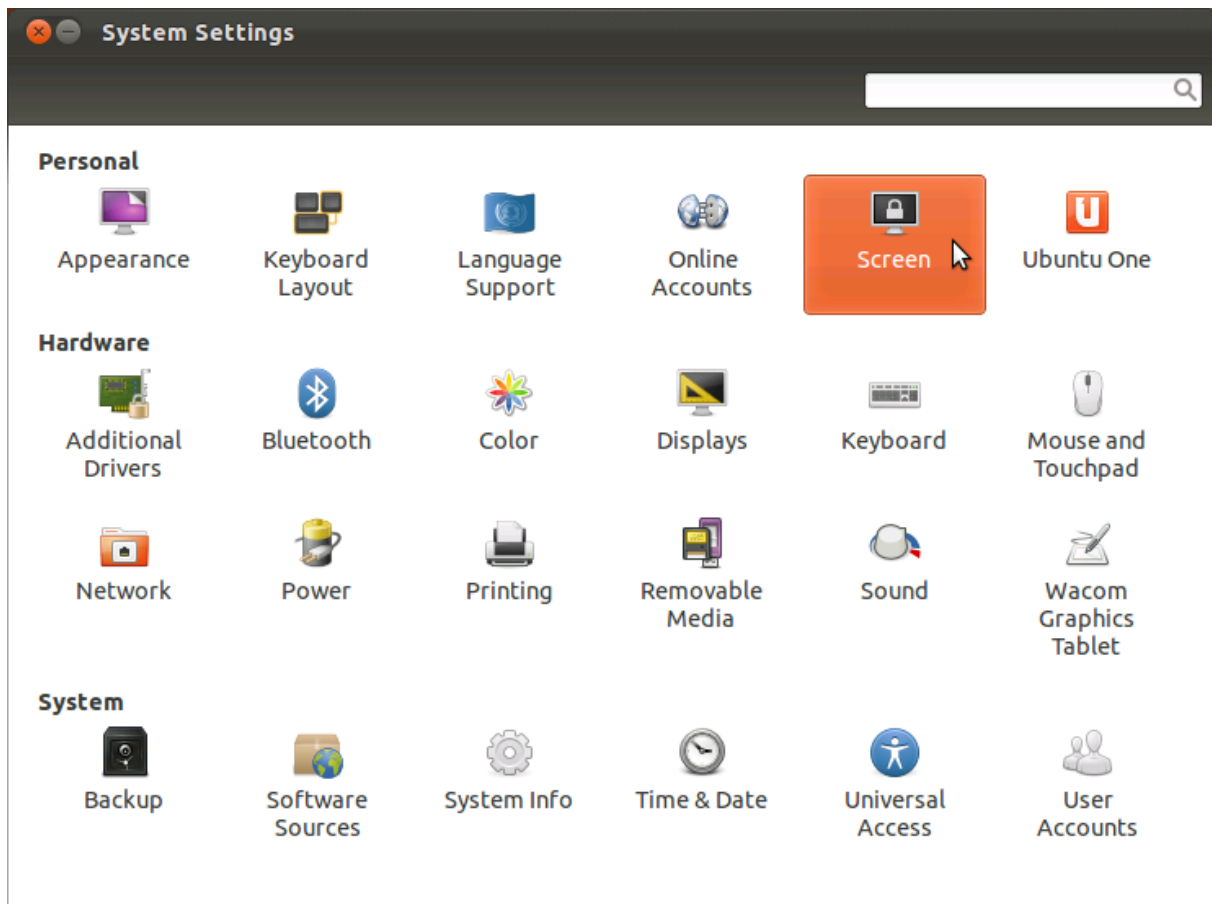
4.2.1 System Settings

You can reach settings and information for personal, hardware, and system here.



Screen

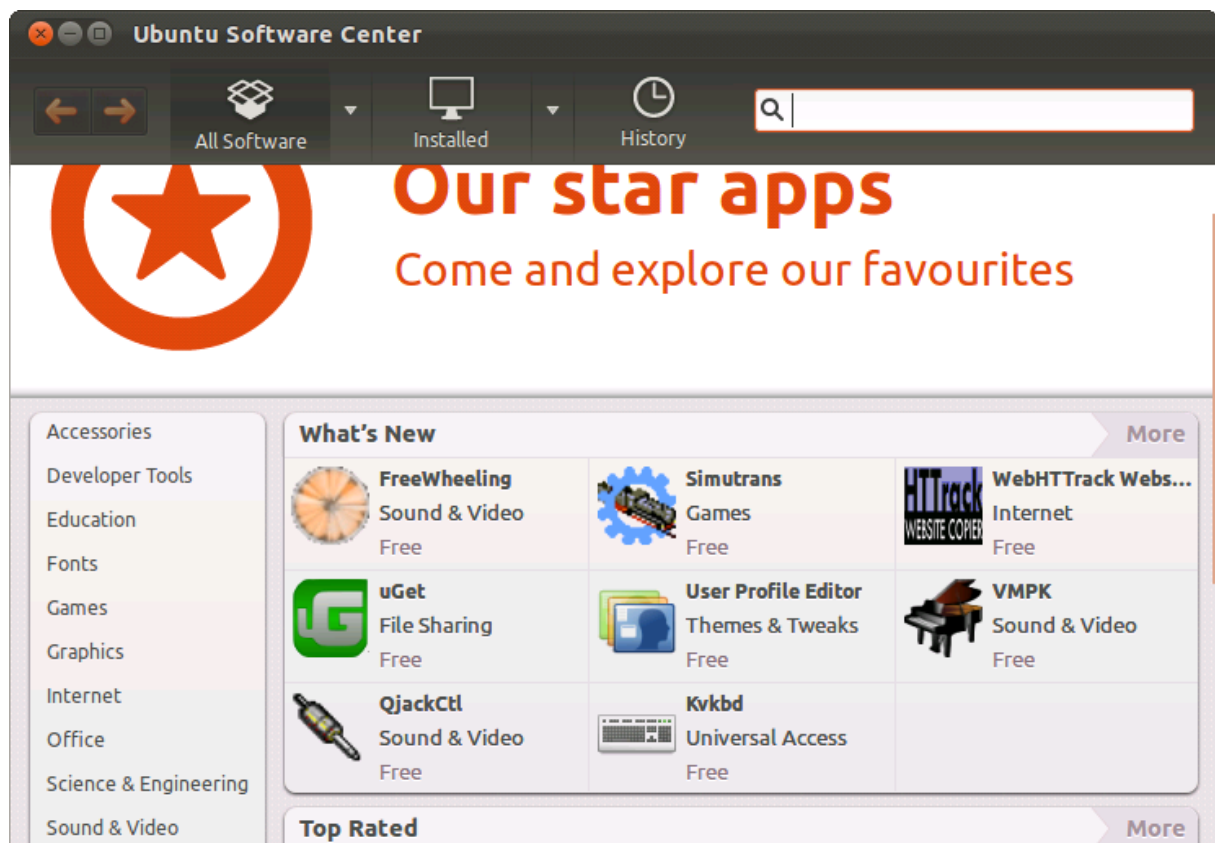
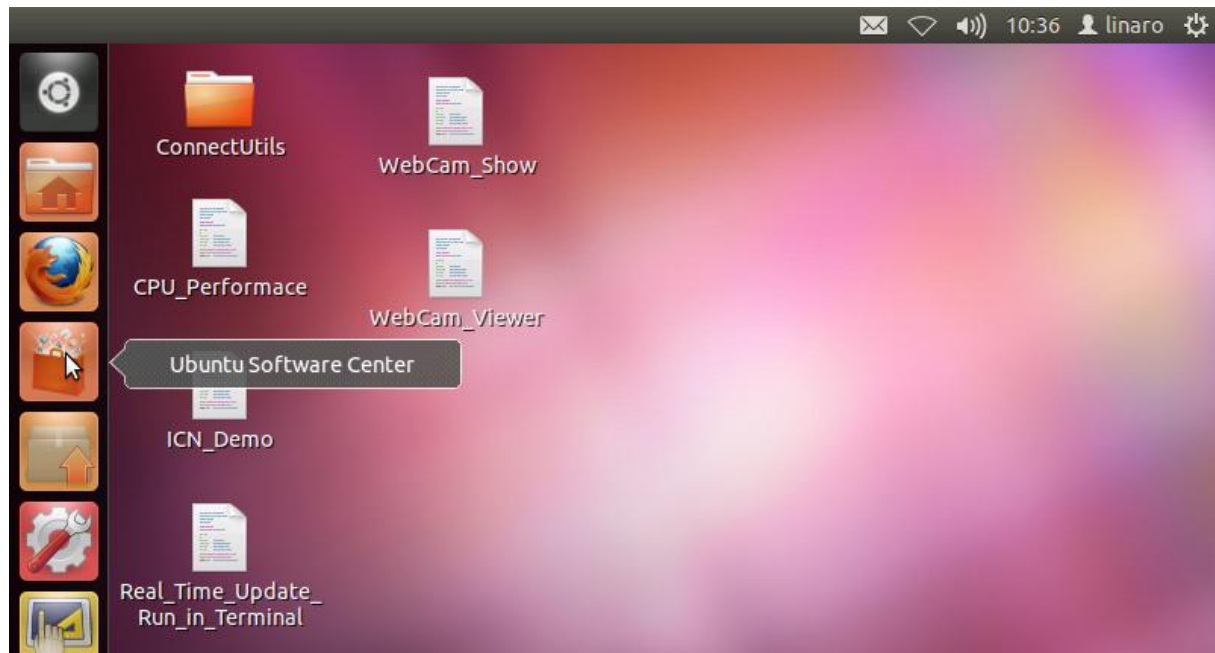
If you want to disable screen lock or disable turn off/dim screen, go to the **Screen** option to change the settings.



Note: In our Ubuntu system which release after 2014/5/20, we remove dim screen function. So after you set like above, the screen would not turn off.

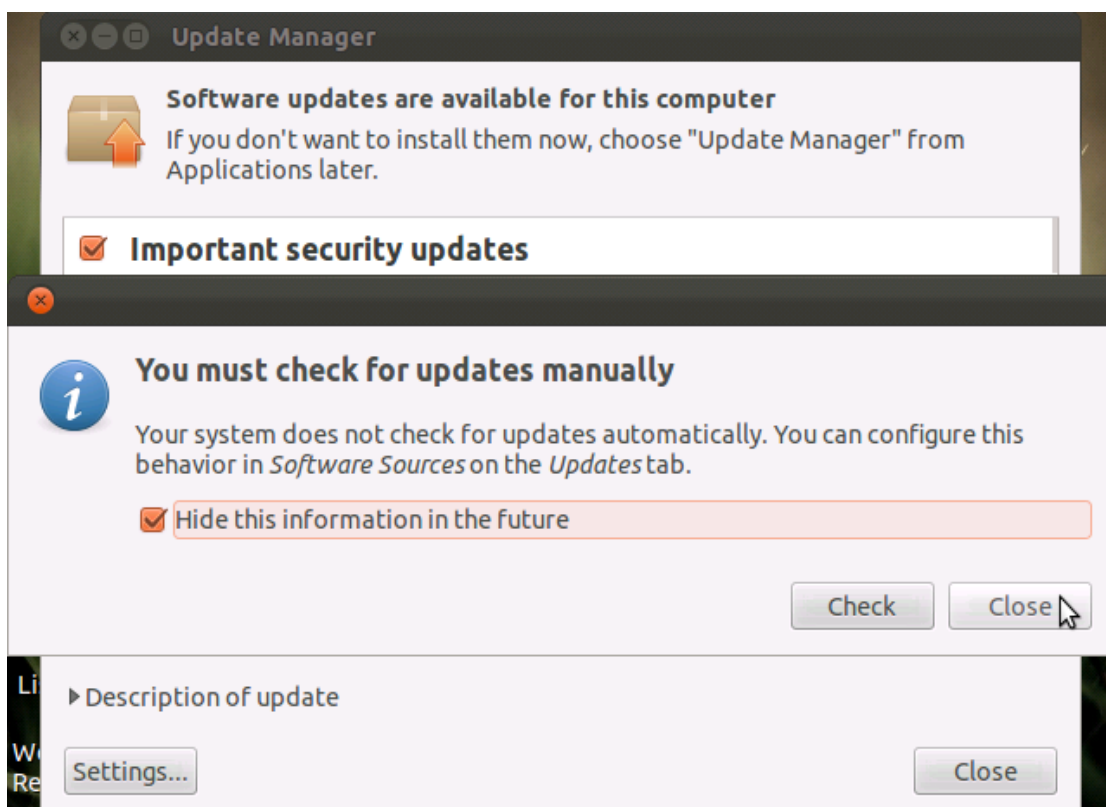
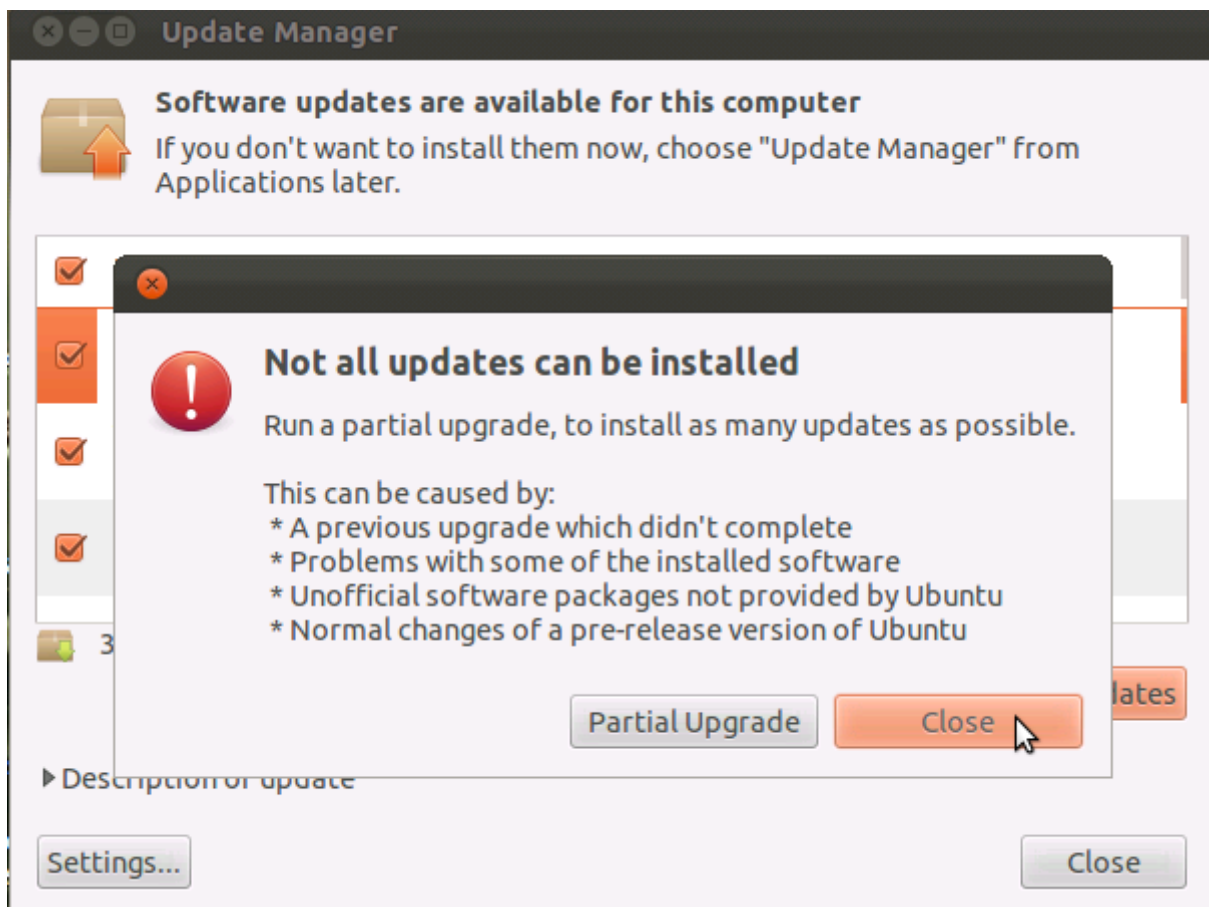
4.2.2 Software Center

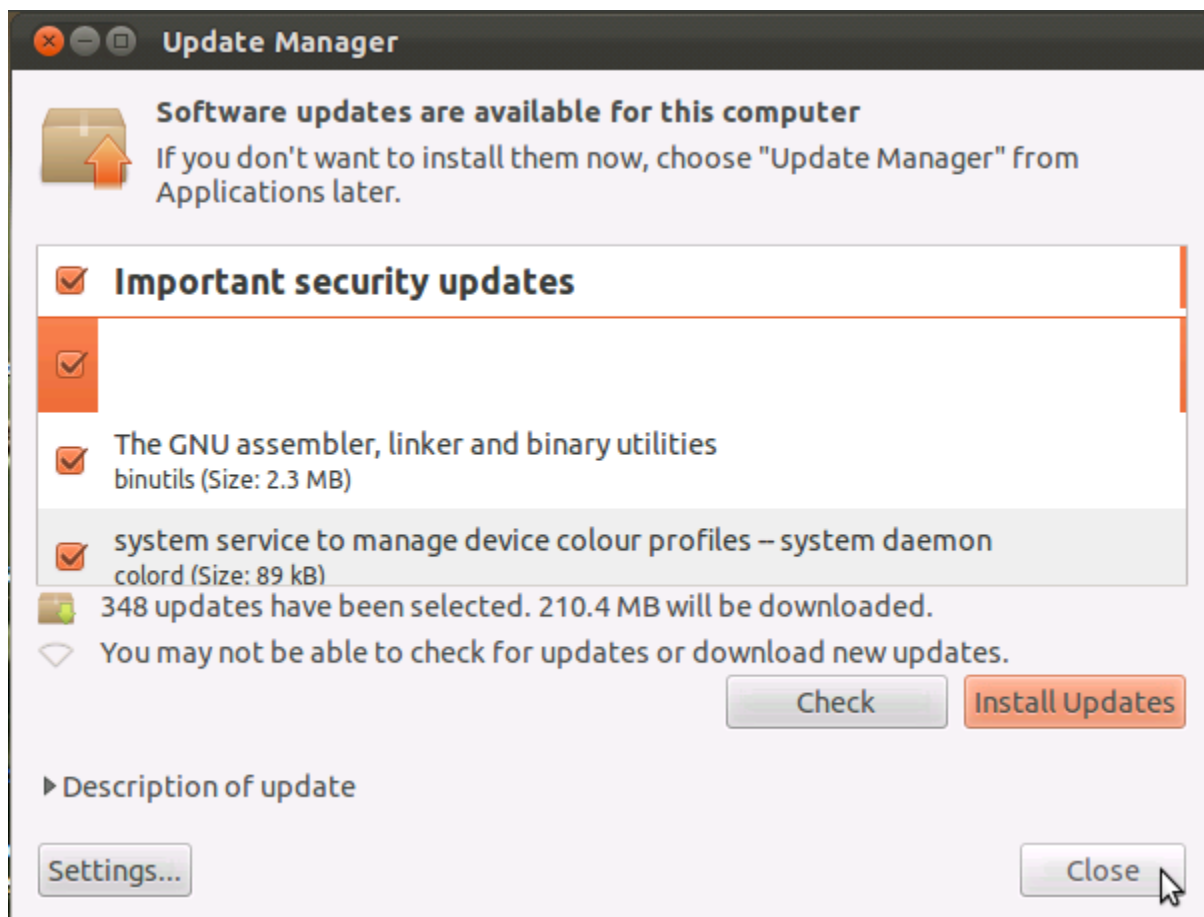
You can use the Ubuntu Software Center to install extra software.



4.2.3 Update Manager

If the Update Manager for Ubuntu appears, **please close it**. Do not use it to update the system. We do not offer any support should you use the system update manager.





5. Touch Panel Calibration

5.1 Android RTP re-calibrate procedure

If you want or need to calibrate the touch for any reason such as:

- Touch is not accurate
- You have changed output
- It is your first boot after you updated firmware and set output.

Use the commands to calibrate after devices boot.

When device power on, please notice the messages from console show '**adb_open**', while the message occurred, quick input or paste character "**stop**" and press "Enter" key to stop the system booting process. Further then input below commands:

```
adb_bind_config

adb_open

# stop

//now the system would stop and display stays at penguin icon

# ts_calibrator

//now there would be a white '⊕' at left, it is calibration point
```

Here is an example of the console message.

```
init: using deprecated syntax for specifying property 'ro.product.model', use ${name} instead
init: using deprecated syntax for specifying property 'ro.serialno', use ${name} instead
init: cannot find '/system/etc/install-recovery.sh', disabling 'flash_recovery'
root@android:/ # rtk_btusb: Realtek Bluetooth USB driver ver 1.0.20130524
usbcore: registered new interface driver rtk_btusb
USB Serial support registered for Vizzini USB serial port
usbcore: registered new interface driver vizzini
USB Driver for Vizzini USB serial port: v.1.1
android_usb: already disabled
android_usb: already disabled
mtp_bind_config
adb_bind_config
adb_open
stop
root@android:/ #
```

When the calibration point '⊕' shows, please quickly and accurately touch the cross on panel. Be careful that this calibrator is with timeout function. If it is timeout, just input **ts_calibrator** again.

After touching several crosses, there will appear a green and red small square. Touch the left green one a while. It is accurate confirmation function; all blue crosses should be within the green square. If not, the calibrator will let you calibrate again automatically.

Finally, reset power or keyin **start** to continue system.

start

5.2 Ubuntu RTP re-calibrate procedure

5.2.1 Ubuntu 11.10

For the Ubuntu 11.10 version, if you want to re-calibrate the touch function, please kindly key in following commands on the console terminal after system booted:

```
# rm -f /etc/pointercal
```

```
//delete calibration file
```

```
# shutdown -r now
```

```
//then reboot
```

5.2.2 Ubuntu 12.04

For the Ubuntu 12.04 version, if you want to re-calibrate the touch accuracy, please kindly follow below steps to accomplish the calibration:

1. First, connected keyboard, mouse and Internet (Ethernet port).
2. Open Ubuntu terminal window which on the left hand side shortcut list.
3. On the terminal window, please input below command line.

```
~$ xinput_calibrator --device ak4183ts
```

Note

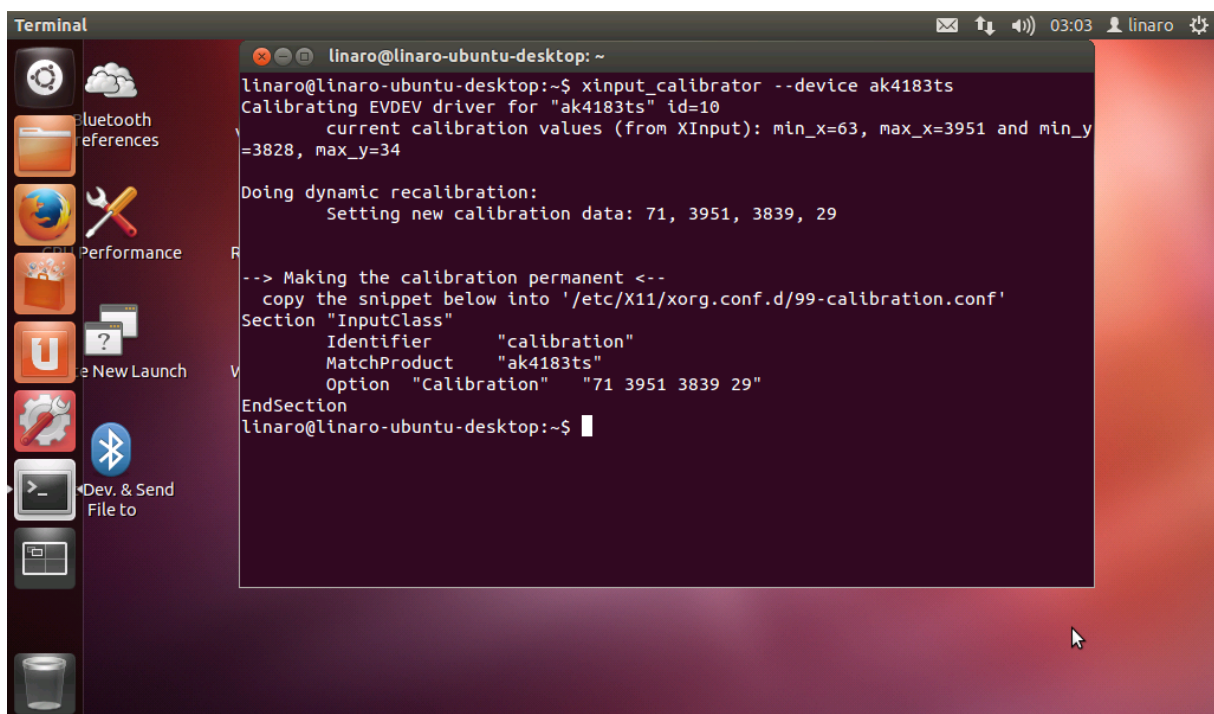
If you can't do [xinput_calibrator], please try to get the app first as following below step:

On the terminal screen, please input the command line which in red.

```
~$ sudo apt-get install xinput-calibrator
```

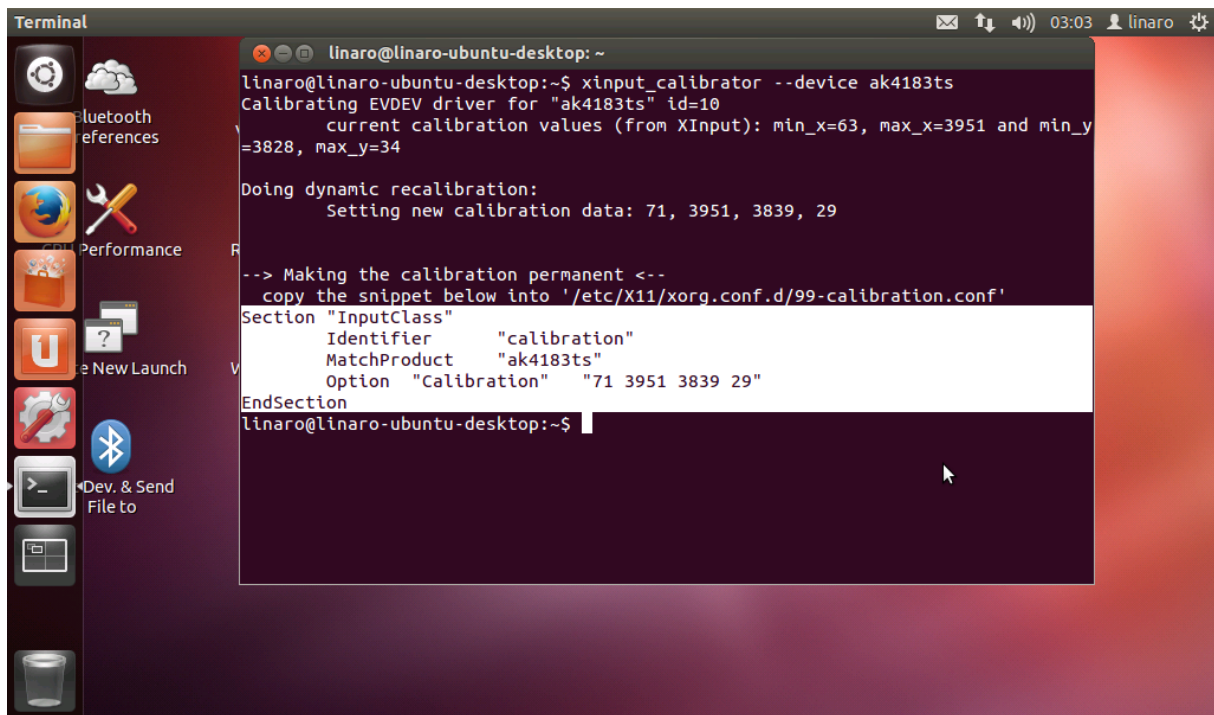
And input password: **linaro**

4. Now you can start calibrating. When calibration is successful, program will be closed automatically and showed some information.



5. Now, test the touch function is functional or not. If not, please go to step 3 and calibrate again.

- Copy the content between [Section "InputClass"] and [EndSection] (refer below picture in white block)



```
linaro@linaro-ubuntu-desktop:~$ xinput_calibrator --device ak4183ts
Calibrating EVDEV driver for "ak4183ts" id=10
current calibration values (from XInput): min_x=63, max_x=3951 and min_y
=3828, max_y=34

Doing dynamic recalibration:
Setting new calibration data: 71, 3951, 3839, 29

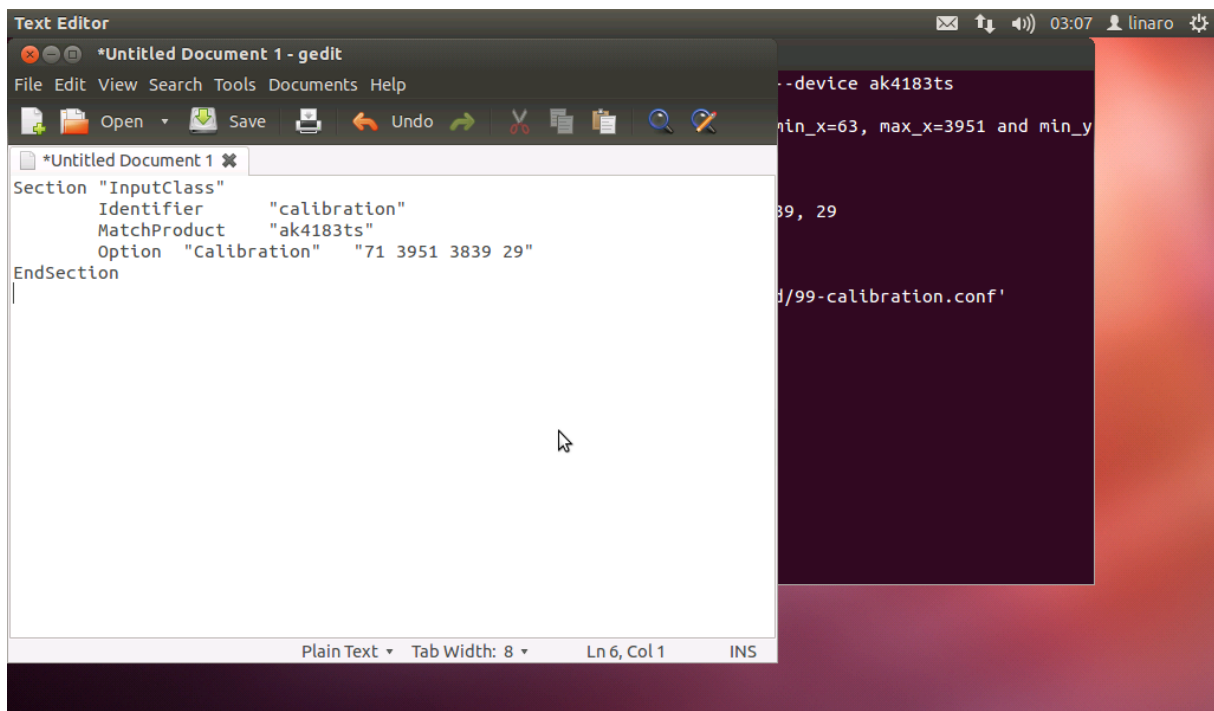
--> Making the calibration permanent <--
copy the snippet below into '/etc/X11/xorg.conf.d/99-calibration.conf'
Section "InputClass"
    Identifier      "calibration"
    MatchProduct    "ak4183ts"
    Option "Calibration" "71 3951 3839 29"
EndSection
linaro@linaro-ubuntu-desktop:~$
```

- On terminal screen, please input

~\$ **sudo gedit**

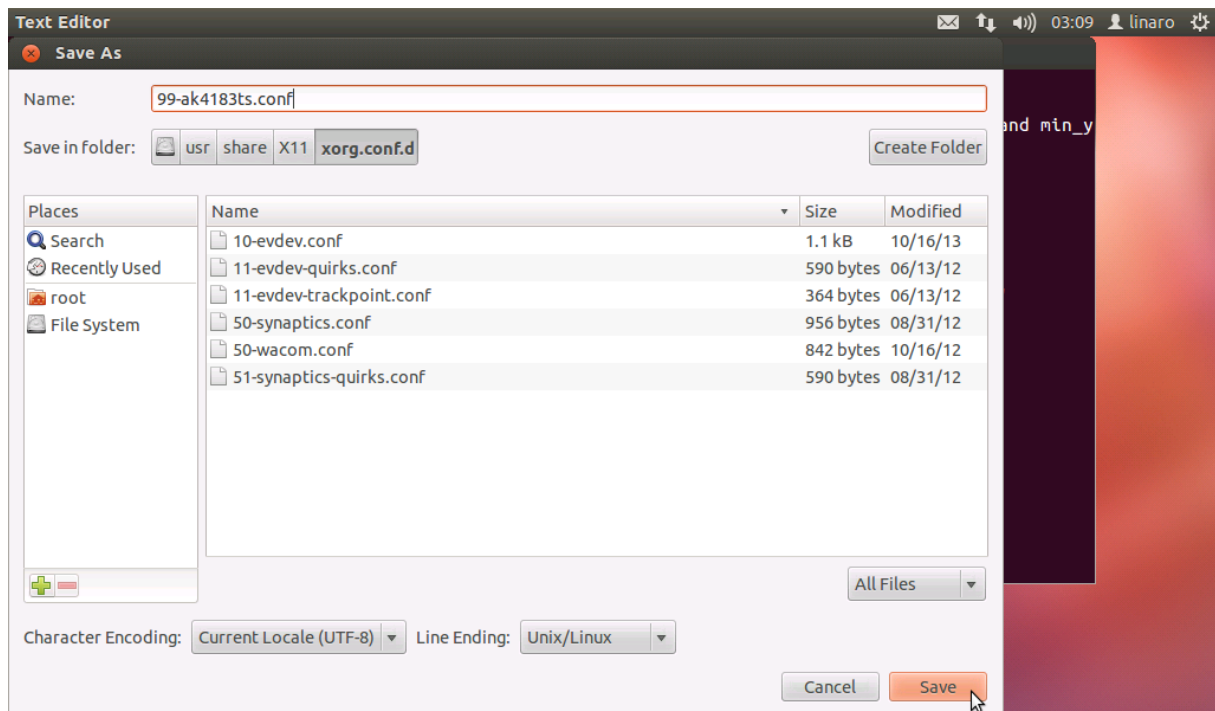
And input password: **linaro**

- Paste the content what you copied just now in the gedit.



```
*Untitled Document 1 - gedit
File Edit View Search Tools Documents Help
Open Save Undo Cut Copy Paste Find
*Untitled Document 1
Section "InputClass"
    Identifier      "calibration"
    MatchProduct    "ak4183ts"
    Option "Calibration" "71 3951 3839 29"
EndSection
Plain Text Tab Width: 8 Ln 6, Col 1 INS
```

9. Save the file under the path `/usr/share/X11/xorg.conf.d/`
And named the file as `99-ak4183ts.conf`
Then, close gedit.



10. Finally, reboot to check whether touch is correct or not.