CASIO
V-T500/V-N500 Series
Quick Start Guide

(Version 1.10)

CASIO Computer Co., Ltd.

October 2014

Table of the Contents

1. Product Overview 4
   - 1.1 Features at a Glance 4
   - 1.2 Library Configuration 6
   - 1.3 Development Manuals 6
2. Prerequisites 7
   - 2.1 Skills Required 7
   - 2.2 Hardware Required 7
   - 2.3 Systems Required 7
   - 2.4 Software Required 8
3. Setting up the Development Environment 9
   - 3.1 Application Development 9
   - 3.2 Installing JDK 10
   - 3.3 Installing Android SDK (ADT Bundle) 13
   - 3.4 Installing Software Development Kit (“SDK”) 18
4. Connecting V-T500/V-N500 to PC 19
   - 4.1 Installing USB Driver 19
5. Setting Up the Development Environment 27
   - 5.1 Setting JDK 27
   - 5.2 Setting AVD 28
   - 5.3 Setting Android Project 30
   - 5.4 Setting Up the Debugging Environment 32
     - 5.4.1 Setting the Debug Configuration with Terminal 32
     - 5.4.2 Setting the Debug Configuration with the Emulator 33
     - 5.4.3 Debugging Applications 34
6. Emulator 37
   - 6.1 Software Required 37
   - 6.2 Starting Up the Emulator 37
     - 6.2.1 How to Start Up the Emulator 37
   - 6.3 Using the Emulator 40
     - 6.3.1 V- T500/V-N500 Emulator 40
7. Eclipse 41
   - 7.1 Developing with System Library 41
8. Kitting 60
   - 8.1 New Configuration of System 60
     - 8.1.1 Flow of New Configuration 60
     - 8.1.2 Design the system 60
     - 8.1.3 Set Environment 60
     - 8.1.4 Install Application Software 61
     - 8.1.5 Replicate Environment to Other Terminals 62
     - 8.1.6 Set Specific Environment of Each Terminal 65
9. Appendix Default Values of the Setting 67

1. Product Overview

This guide clearly and concisely sets out the information developers need to know to get started with CASIO **V-T500**/**V-N500** series terminals development.

1.1 Features at a Glance

The following is a brief overview of the features available on **V-T500**/**V-N500**series terminals. For further details on the hardware specifications, refer to **V-T500/V-N500Series Hardware Manual**.

Table 1.1 Features at a Glance

V-T500/V-N500		
CPU		OMAP 1.5 GHz Dual core
Memory	RAM	1 GB
	F-ROM	16 GB
Platform		Android 4.0
LCD	Model	IPS mode TFT color LCD
	Size	10.1 inch
	Pixel	1280 x 800 Wide XGA
	Color	16.7
	Viewing angle	80°
	Backlight	LED Backlight
LED		Red x 1, Green x 1 (For confirmation of charging/state of operation)
Touch panel		Capacitive touch panel with Multi-Touch Input :Pen-type digitizer
Button		Power button/function buttons A and B/ Volume button (UP/DOWN)/ Screen locking button (display rotation locking button)
Bluetooth® communication	Version	Bluetooth® Specification Ver.4.0+EDR
	Communication range	Approximately 5 meters(Varies depending on conditions of radio wave and environment.)
	Class	2 (Output 2.5 mW Range 10 m)
NFC	Model	ISO14443 Type A (Mifare)
		ISO14443 Type B
		Felica(JIS X 6319)
		ISO15693 (Tag)
	Antenna	Magnetic loop antenna
SIM	Standard	ISO 7816 IC Card Standard
	General specification	
SAM	Standard	ISO 7816 IC Card Standard
	General specification	
Wireless LAN	Standard specification	IEEE 802.11a/b/g/n
	Communication system	Spread Spectrum Communication System
	Modulation method	BPSK,QPSK,CCK,16QAM,64QAM

	Transfer rate	802.11a/g： up to 54Mbps 802.11b：up to 11Mbps 802.11n：up to 72Mbps
	Communication range	802.11b/g/n： 50m (in-door), 150 m (out-door) (2.4GHz band for /n) 802.11a/n ： 30m (in-door), 150 m (out-door) (5GHz band for /n)
	Output	802.11a：12.0dBm min. Typical 13.5dBm (at 54Mbps) 802.11b：17.0dBm mini. Typical 18dBm (at 11Mbps) 802.11g：12.0dBm mini. Typical 14dBm (at 54Mbps) 802.11n：11.0dBm min. Typical 12.5dBm (at 72Mbps)
GPS	General specifications	1575.42 MHz, C/A Code
	Positioning scheme	Self-ephemeris
SD card slot		Supporting SDHC/SDIO cards
Micro SD card slot		Supporting SDHC card
USB		USB host x 1 USB devicex 1
Camera	Effective pixels	Approx. 2M Pixels (2592×1944 pixels)
	Aperture	F2.8±5%
	Focal length	f=3.4mm±5%
	Focus distance	10cm to infinity
LED light		21,000mcd()
Speaker		Alarm, etc.
Microphone		Voice input
Power source	Main Battery	Lithium Ion Polymer Rechargeable Battery
	Sub-battery	1 x Lithium Ion Secondary Battery
Temperature		-20 to 50 ℃
Humidity		10 to 90% RH
Drop strength		1.0 m
Waterproof and dustproof performance		IP54-compliant
Size		269 mm x 190 mm x 14 mm

1.2 Library Configuration

The **V-T500/V-N500 Software Development Kit** ("**BDK**") provides various libraries including those listed in the table.

Table 1.2

Library	Description
System Library	Library that is used to control the system.
SAM Library	Library that is used to communicate with the SAM Card.
Authentication library	Library that is used to set the authentication function such as the password.
Camera scanning library	Library that is used to scan symbols with the camera.

The following table shows each file name of the jar files.

Table 1.3

Library	jar file
System Library	jp_casio_vx_framework_system.jar
SAM Library	jp_casio_vx_framework_sam.jar
Authentication library	jp_casio_vx_framework_authenticate.jar
Camera scanning library	jp_casio_vx_framework_camerascanner.jar

1.3 Development Manuals

The **V-T500/V-N500 Software Development Kit** ("**BDK**") provides various development reference manuals as described in the table below.

Table 1.4 **V-T500/V-N500** manual

Development Manual	Description
Quick Start Guide	This reference manual.
Hardware Manual	Reference manual that describes hardware specifications in detail on each dedicated option and V-T500/V-N500 handheld terminal.
Software Manual	Reference manual that describes software specifications in detail for all the software integrated in V-T500/V-N500 handheld terminal.

Table 1.5 Library manual

Development Manual	Description
System Library Manual	Reference manual that describes individual functions in detail for System Library.
SAM Library Manual	Reference manual that describes individual functions in detail for SAM Library.
Authentication Library Manual	Reference manual that describes individual functions in detail for Authentication Library.
Camera Scanning Library Manual	Reference manual that describes individual functions in detail for Camera Scanning Library.

2. Prerequisites

2.1 Skills Required

V-T500/**V-N500** application can be developed by the following languages:

- Java

The following skills or experience are also desirable:

- Android OS
- Development of Android applications
- Eclipse integrated development environment
- Some networking experience

2.2 Hardware Required

The following models and dedicated options are available.

Table 2.1 List of available model

Model no.	Remark
V-N500	
V-T500	

2.3 Systems Required

PC Operating System

One of following OS (in case of 32 bit) is required.

- Microsoft Windows Vista Service Pack 2 or later
- Microsoft Windows 7 Service Pack 1 or later (Professional/Ultimate)
- Microsoft Windows 8.1 (Professional)

One of following OS (in case of 64 bit) is required.

- Microsoft Windows 7 Service Pack 1 or later (Professional/Ultimate)
- Microsoft Windows 8.1 (Professional)

Main unit of the computer

The personal computer on which the OS above operates

Memory

Secure the necessary and adequate capacity for operating the OS above.
The necessary capacity varies according to the OS used. In either case, a capacity of 1 GB or more is recommended.

Hard disc capacity

10 GB or more free space is recommended.

2.4 Software Required

The Android development environment is required for developing the **V-T500**/**V-N500** application.

Table 2.2 Development platform

Programming Language	Detail
Java	Java SE Development Kit Eclipse IDE with built-in ADT Android SDK (API level 15)

Note:
Procedures may differ in the future by updating of Google.

3. Setting up the Development Environment

3.1 Application Development

This chapter explains what to set up for the development environment before starting your application development.

(1) Installing JDK/Android SDK/Eclipse/ADT

Install the development environment for the Android application on the PC.
For details, refer to Chapter 2.4Software Required.

(2) Installing CASIO SDK to PC

Install CASIO SDK on the PC.
For the installation method, refer to Chapter 3.6Installing Software Development Kit ("SDK").

(3) Connecting V-T500/V-N500 to PC

Connect **V-T500/V-N500** to PC.
For the connection method, refer to Chapter 4Connecting V-T500/V-N500 to PC.

(4) Preparing the Device Emulator

Prepare to use the CASIO **V-T500/V-N500** device emulator on the PC.
For the operation method, refer to Chapter 6Emulator.
If no device emulator is required, proceed to (5).

(5) Application Development

Now, the application development environment is set up and your development with the development platform can be started.
After an application has been developed, transfer it to the Device Emulator or an actual terminal of **V-T500/V-N500** via the tool of (3) to check the operability.
For the application development method and transferring your application, refer to Chapter 7Eclipse.

3.2 Installing JDK

Java SE JDK can be downloaded from the Java site. Access the Java site, and download the **Java SE JDK** package compatible with the Windows platform used.
The URL and the details of the Java site may differ from those given in this document because the site has been updated.
When downloading **Java SE JDK**, access the Java site from a search site, and follow the procedures explained on the site.
http://www.oracle.com/technetwork/java/javase/downloads/index.html

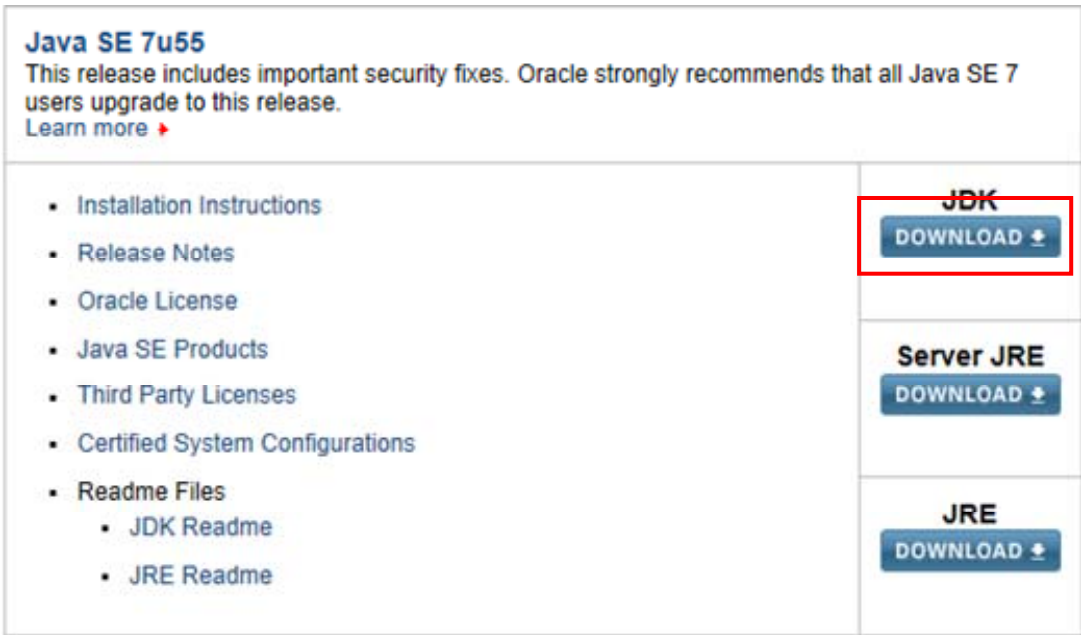


Figure 3.1

jdk-7u55-windows-i586.exe (32 Bit version) is downloaded in the example below.
Download the 64 bit version for the 64 bit version OS.

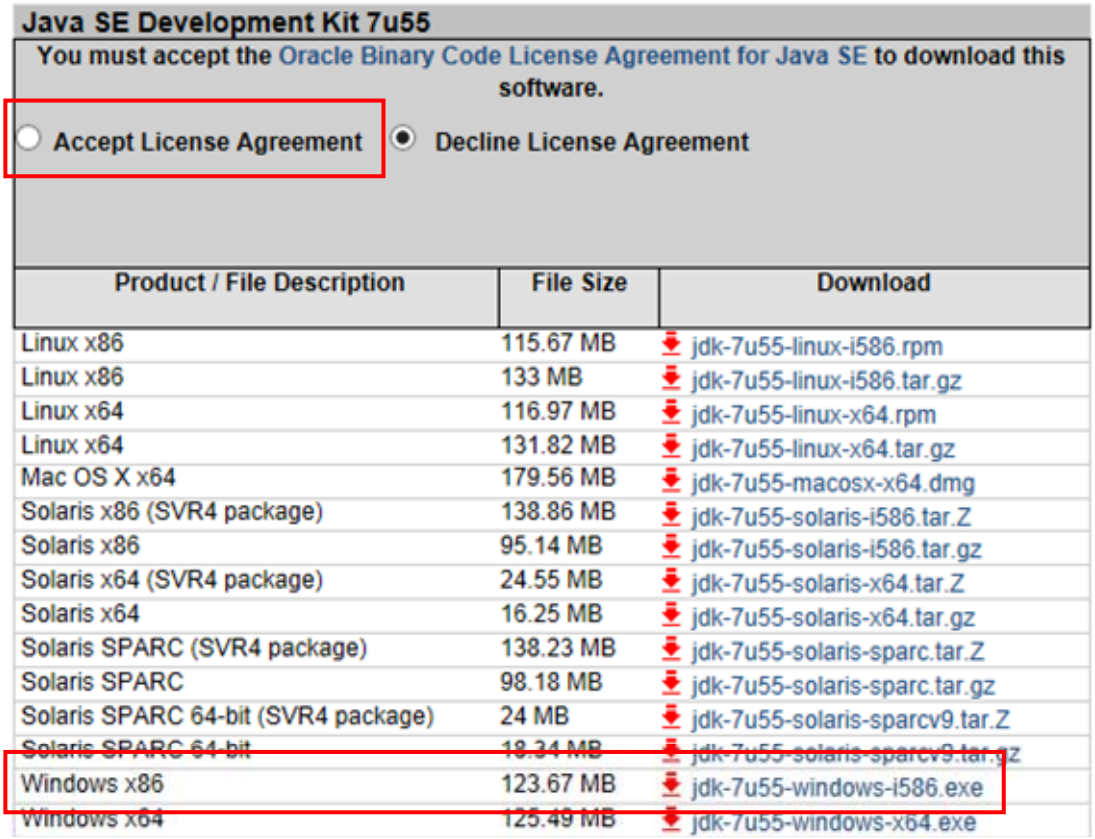


Figure 3.2

Start the downloaded file, and execute installation.
For installation, execute the downloaded file by a user who has administrative authority.
If you wish to execute it by a user who does not have administrative authority, right-click on the downloaded file on the explorer, select **Execute as the administrator** from the displayed menu, and execute it.
No changes are required during installation.

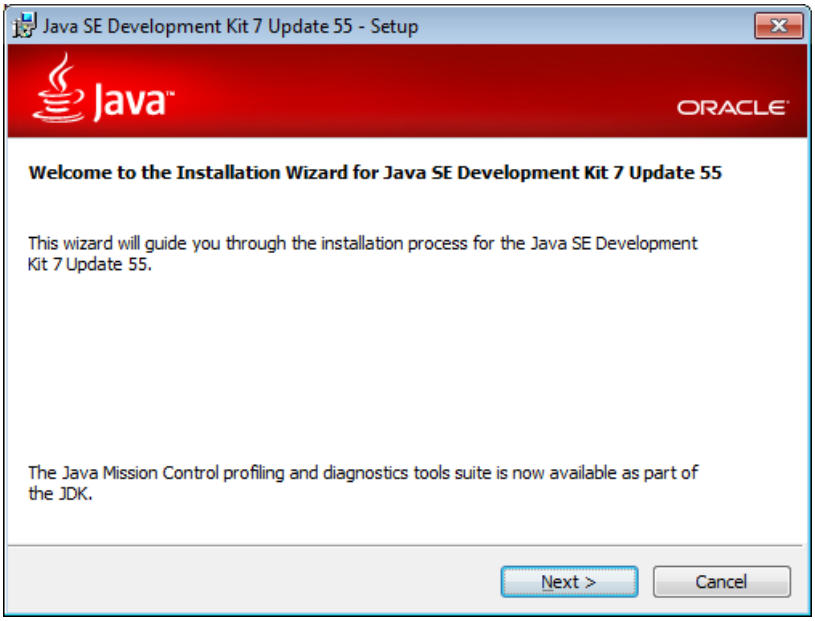


Figure 3.3

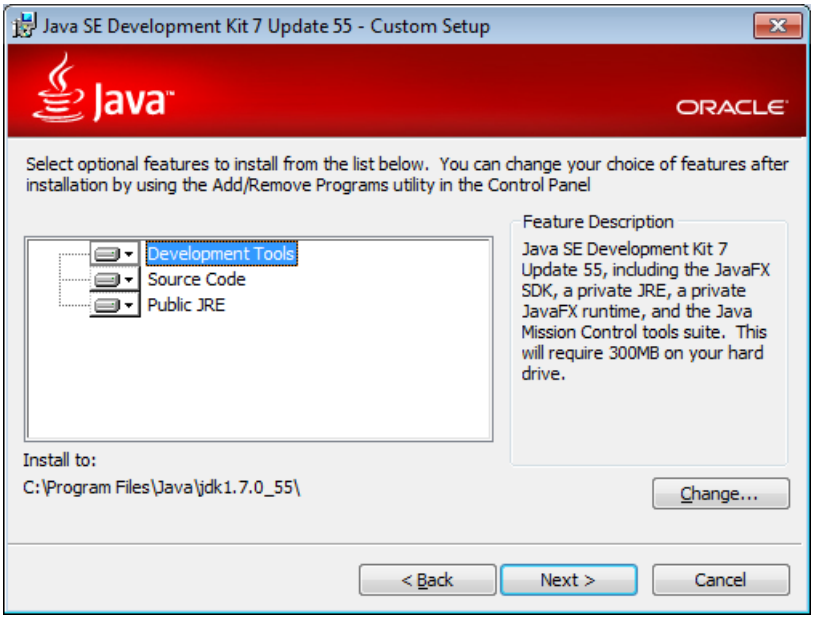


Figure 3.4

Figure 3.5

When the installation is finished, set the folder where JAVA is installed in the environment variable **JAVA_HOME**, and log in again. Pay attention because if you forget this setting, an error occurs when installing **Android SDK**.

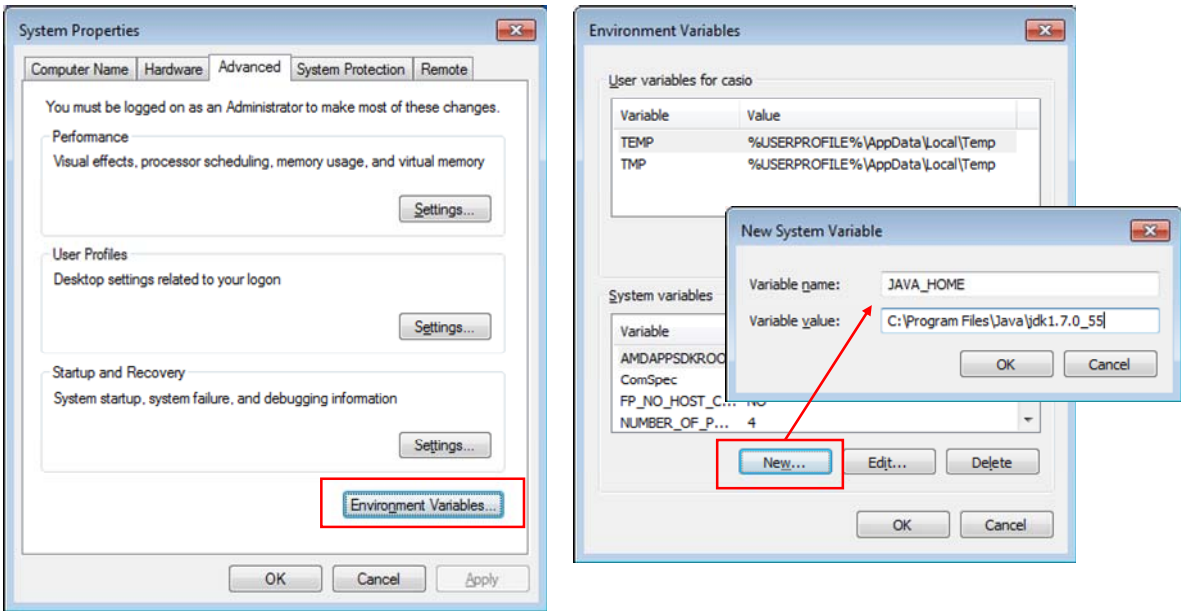


Figure 3.6

3.3 Installing Android SDK (ADT Bundle)

Android SDK (ADT Bundle) is available from the Android of Google site.
Please input the URL at browser, and download **Android SDK**.
http://developer.android.com/sdk/index.html?hl=sk

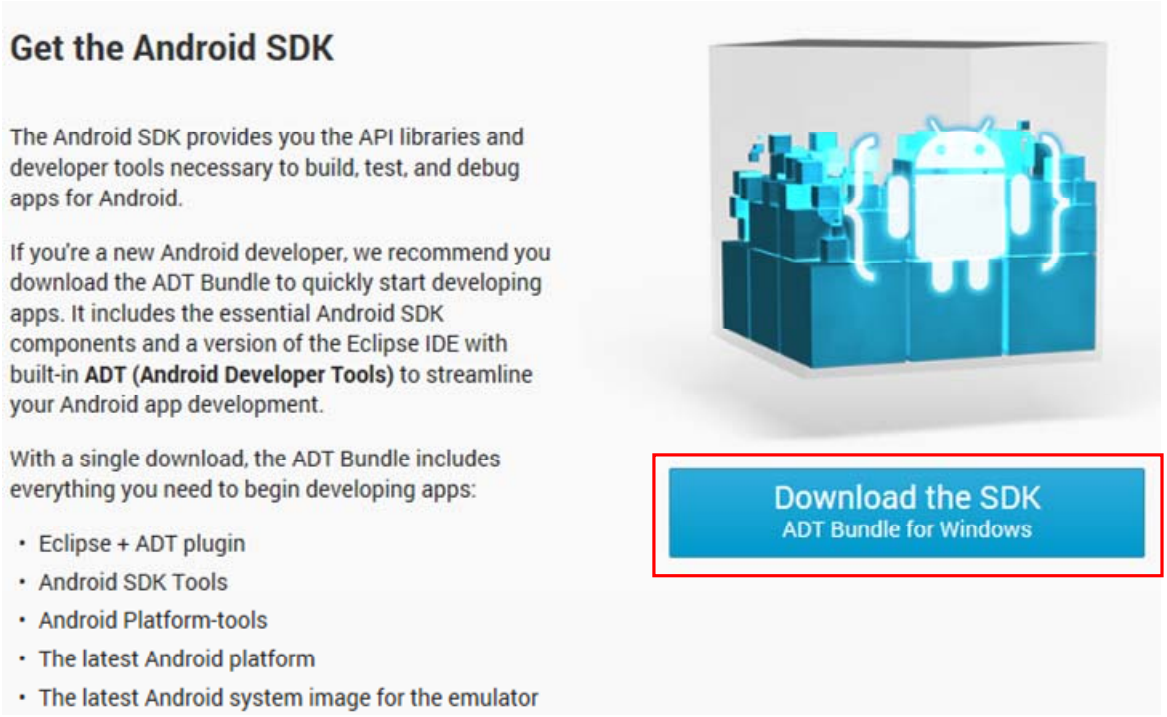


Figure 3.7

32 Bit version is downloaded in the example below.
Download the 64 bit version for the 64 bit version OS.

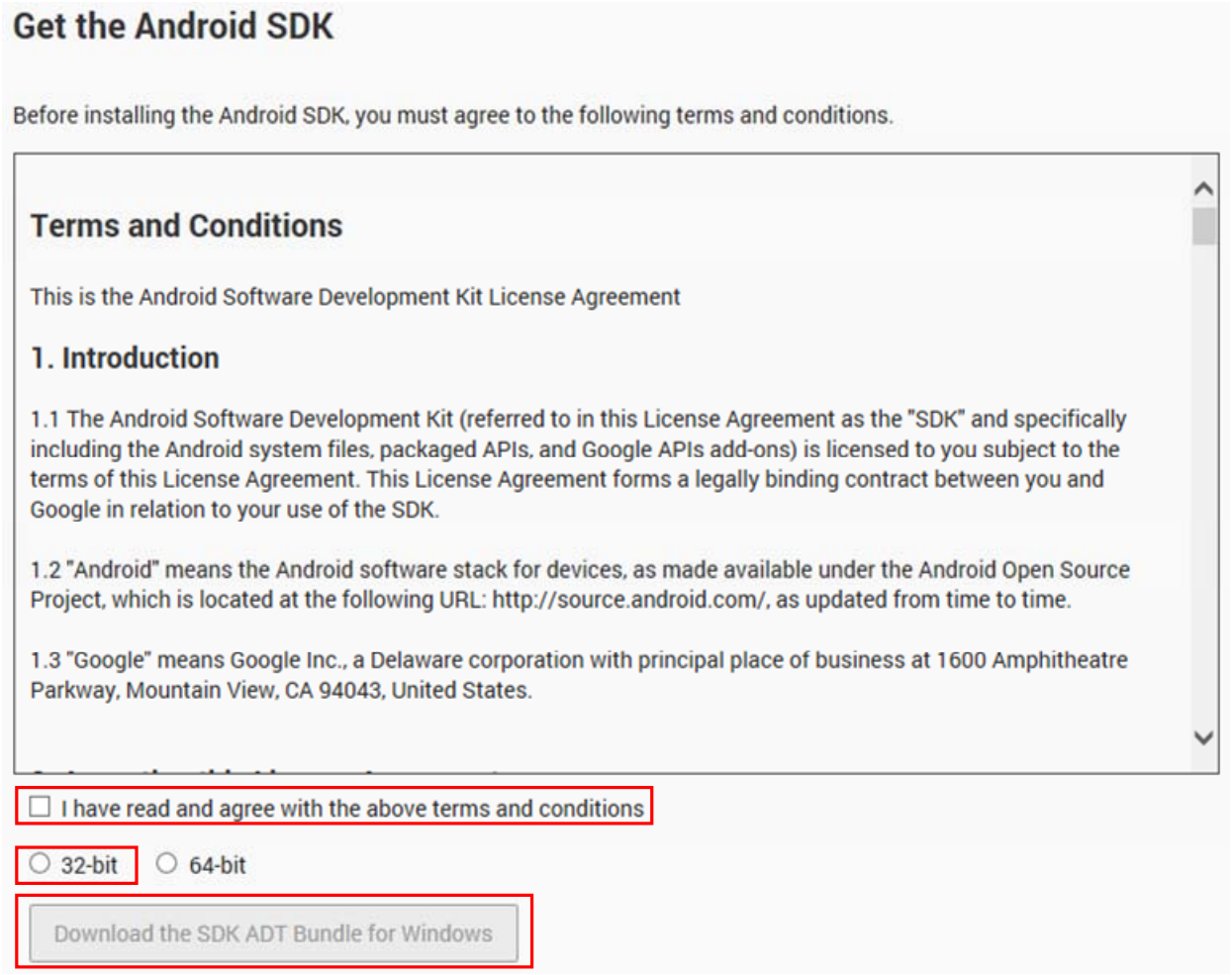


Figure 3.8

Unzip downloaded file into directory of your choice.

Add the paths related to **Android SDK** to **PATH** of the system environment variable.
Add paths to tools under the **The folder Android SDK is installed**, and **platform-tools** under the **The folder Android SDK is installed**.

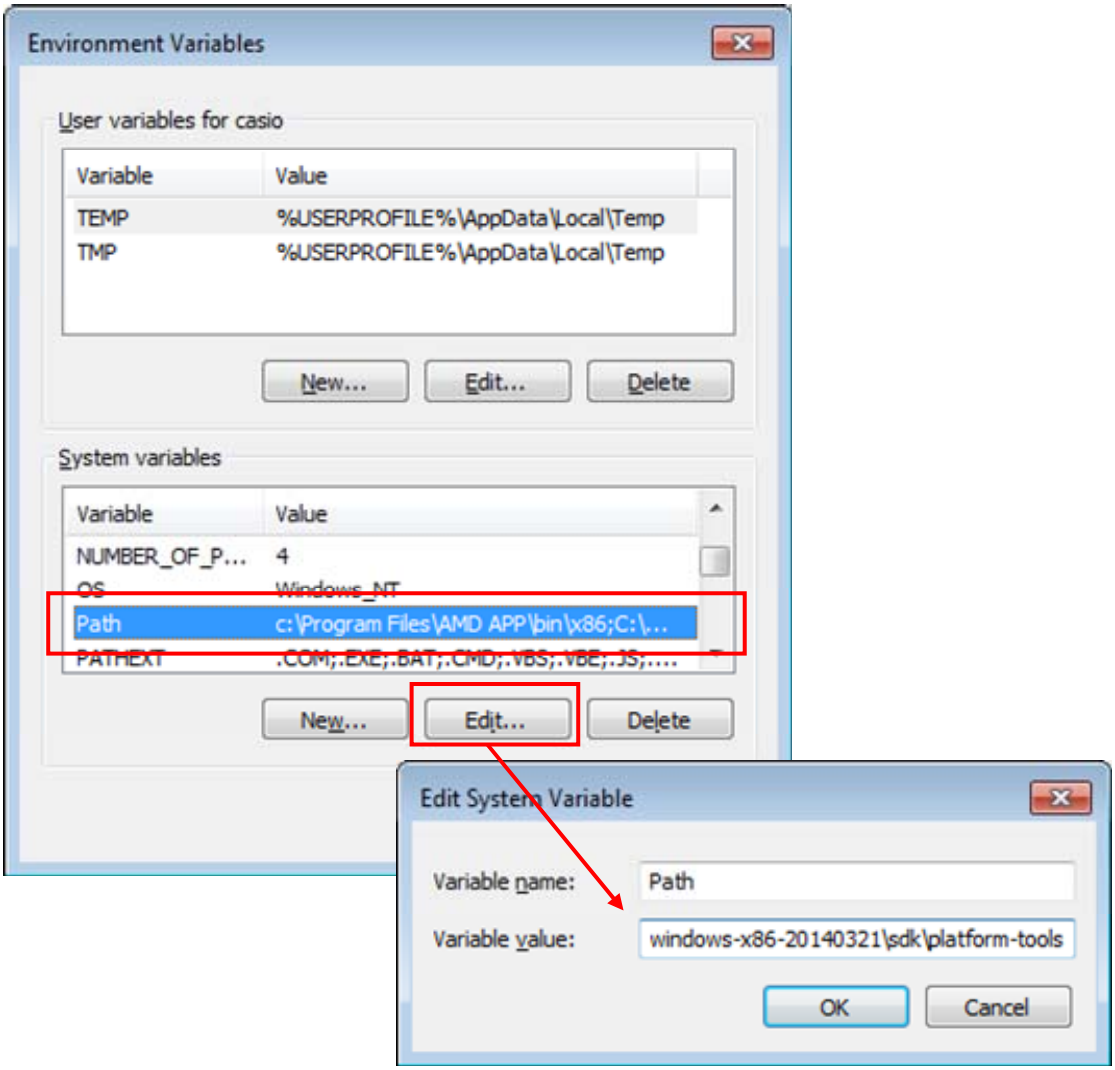


Figure 3.9

Execute **SDK Manager** from the menu of **Android SDK Tools** by a user who has administrative authority.
If you wish to execute it by a user who does not have administrative authority, right-click on **SDK Manager** in the menu of **Android SDK Tools**, and select **Execute as the administrator** from the displayed menu, and execute it.
Select **Reload** from the Package menu.

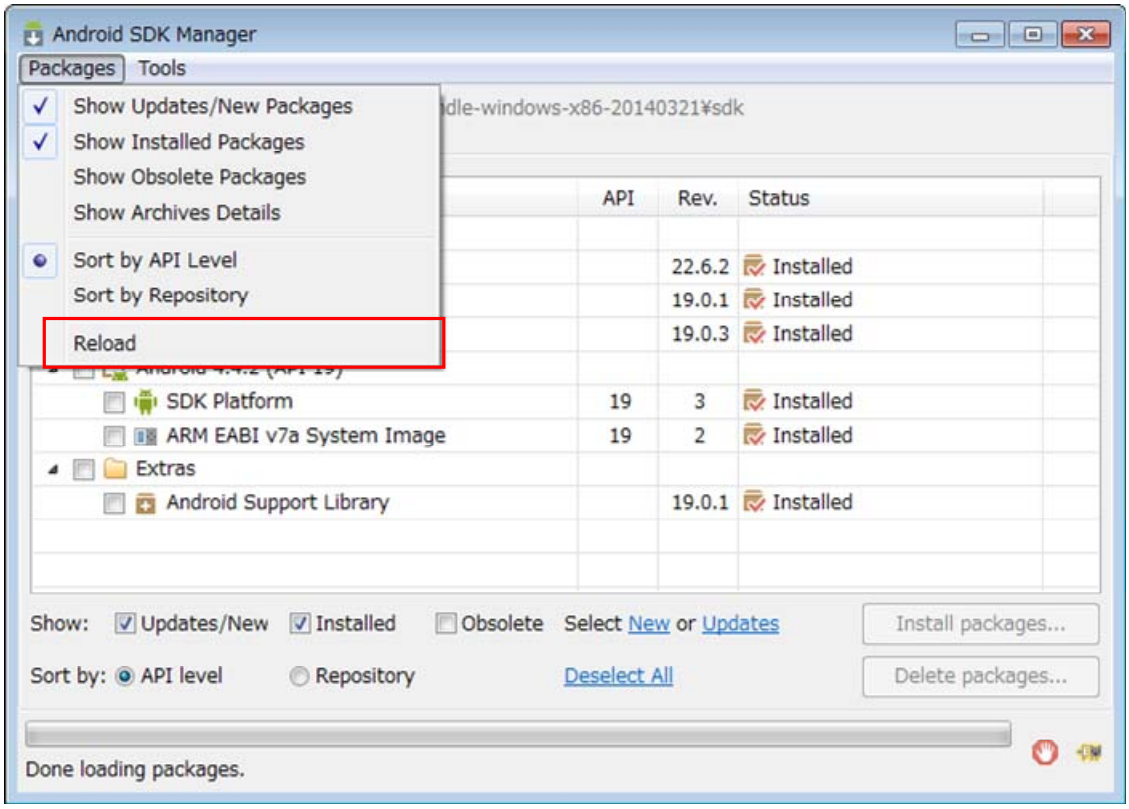


Figure 3.10

The list of packages is updated to the latest version.
Check **Android 4.0.3(API15)** and **Google USB driver** in **Extras**, then install.
Please do not check **Android SDK Tools** in **Tools**.
Please pay attention because Android 4.0.3 (API15) is not checked by default.

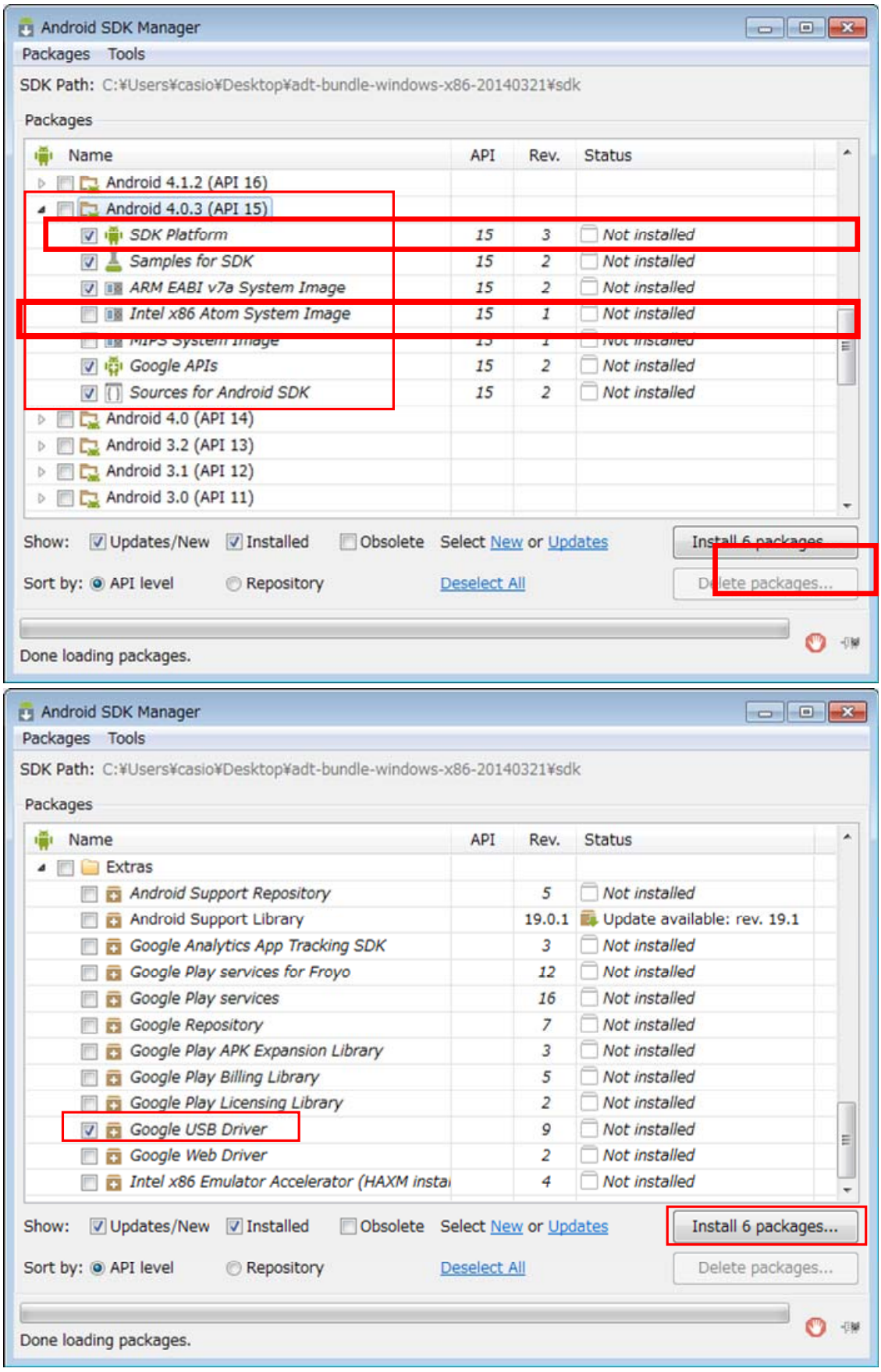

Android SDK Manager
Packages Tools
SDK Path: C:¥Users¥casio¥Desktop¥adt-bundle-windows-x86-20140321¥sdk
Packages
Name API Rev. Status
Android 4.1.2 (API 16)
Android 4.0.3 (API 15)
SDK Platform 15 3 Not installed
Samples for SDK 15 2 Not installed
ARM EABI v7a System Image 15 2 Not installed
Intel x86 Atom System Image 15 1 Not installed
Google APIs 15 2 Not installed
Sources for Android SDK 15 2 Not installed
Android 4.0 (API 14)
Android 3.2 (API 13)
Android 3.1 (API 12)
Android 3.0 (API 11)
Show: Updates/New Installed Obsolete Select New or Updates
Sort by: API level Repository Deselect All
Done loading packages.
Android SDK Manager
Packages Tools
SDK Path: C:¥Users¥casio¥Desktop¥adt-bundle-windows-x86-20140321¥sdk
Packages
Name API Rev. Status
Extras
Android Support Repository 5 Not installed
Android Support Library 19.0.1 Update available: rev. 19.1
Google Analytics App Tracking SDK 3 Not installed
Google Play services for Froyo 12 Not installed
Google Play services 16 Not installed
Google Repository 7 Not installed
Google Play APK Expansion Library 3 Not installed
Google Play Billing Library 5 Not installed
Google Play Licensing Library 2 Not installed
Google USB Driver 9 Not installed
Google Web Driver 2 Not installed
Intel x86 Emulator Accelerator (HAXM instal 4 Not installed
Show: Updates/New Installed Obsolete Select New or Updates
Install 6 packages...
Sort by: API level Repository Deselect All
Delete packages...
Done loading packages.

Figure 3.11

3.4 Installing Software Development Kit (“SDK”)

This chapter explains about installing SDK and construct Android SDK.

1. Install SDK as add-on for **Android SDK**.
2. Navigate to “**package**” directory of SDK.
3. Copy “**V-T500**” directory under “**package**” directory of SDK.
4. Paste “**V-T500**” directory copied in above 3 into “**add-ons**” directory under the root directory of **Android SDK**.

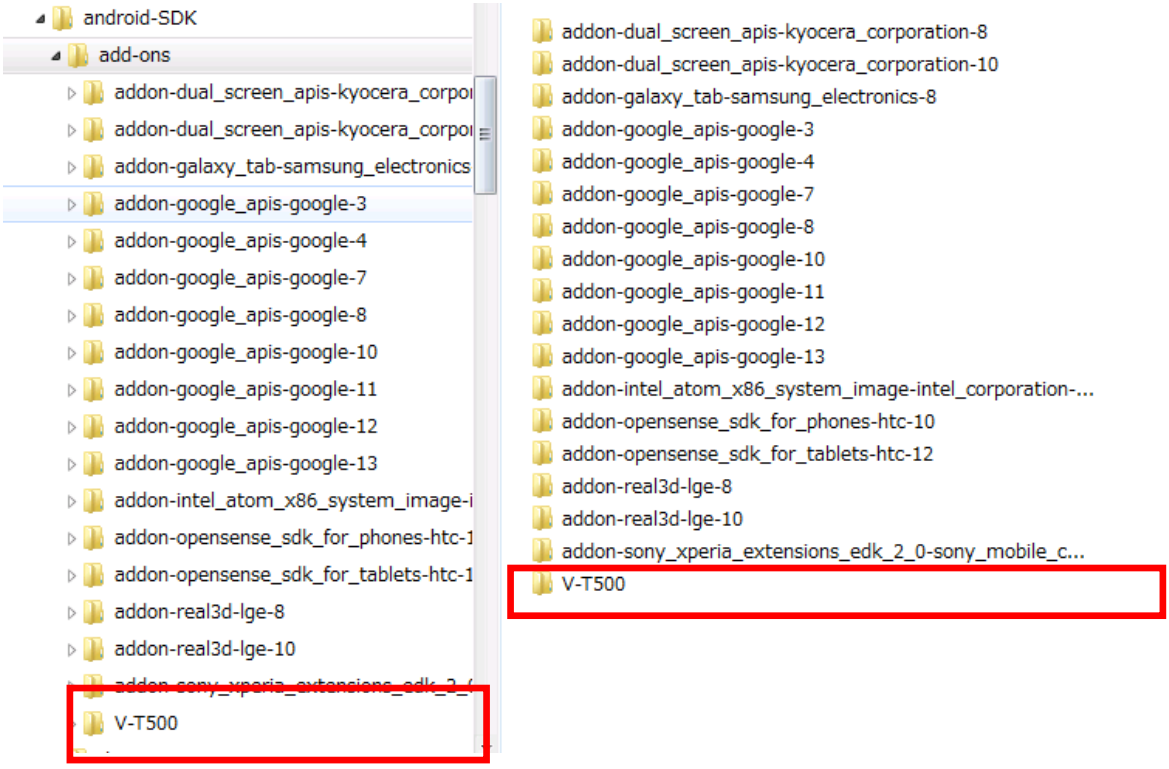


Figure 3.12

4. Connecting V-T500/V-N500 to PC

4.1 Installing USB Driver

Install the USB driver to connect the PC and the **V-T500/V-N500**.

Only the configuration file of the USB driver is included in the **V-T500/V-N500** development kit.
The USB driver is included in the **Android SDK**, which was installed in Chapter 3.3Installing Android SDK.

■ In case of Windows Vista and Windows 7

First, create a folder for the driver (e.g. "C:\V-T500Driver").
Next, search a file named **android_winsub.inf** in the installed folder of **Android SDK**.

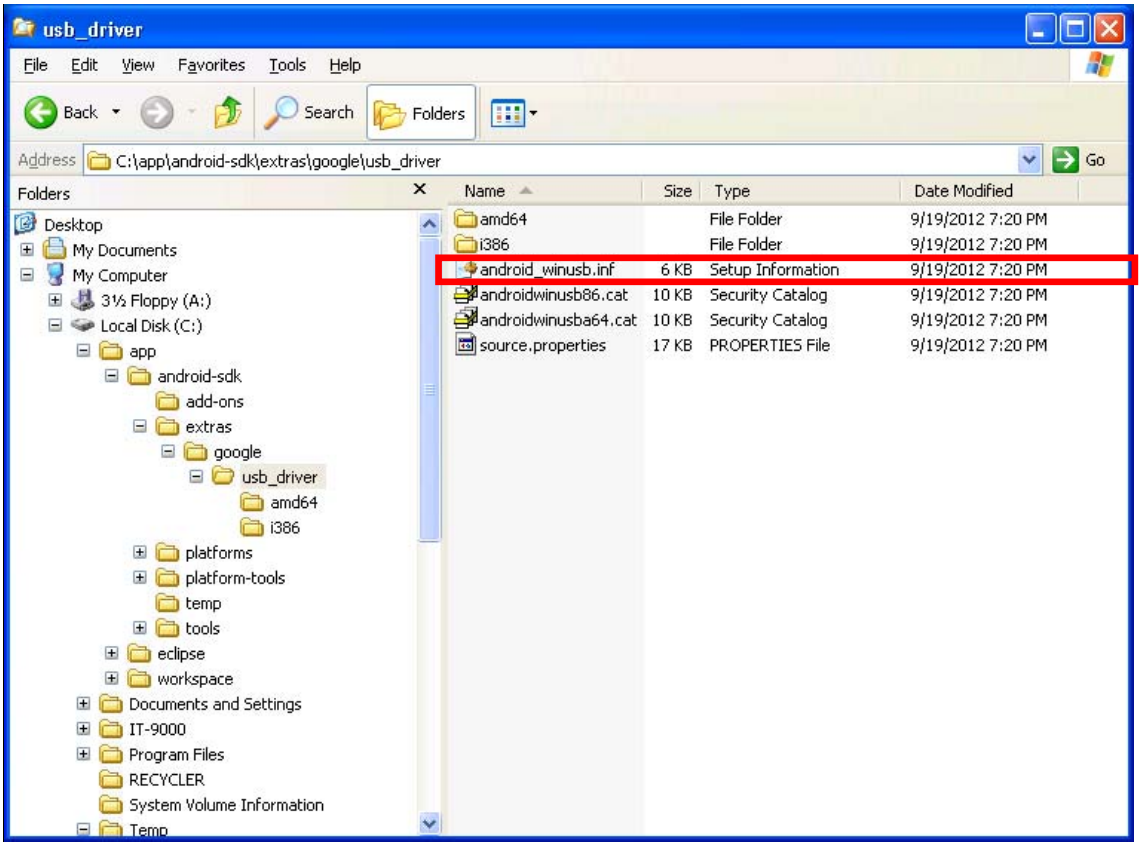


Figure 4.1

When the file **android_winsub.inf** is found, copy it along with the sub folders in the folder for the driver that you created at the beginning.
Then, copy the following file from the CD.
V-T500/V-N500 development kit.\V-T500\driver\ android_winsub.inf

Paste the file in the same folder for the driver as above by overwriting.

When the mini-B USB port of **V-T500/V-N500** and the USB port of the PC are connected by cable, the connection is automatically detected by the OS, and the following dialog is displayed. Follow the instructions on the screen.

Select **No. not this time**, and press **Next**. > button.

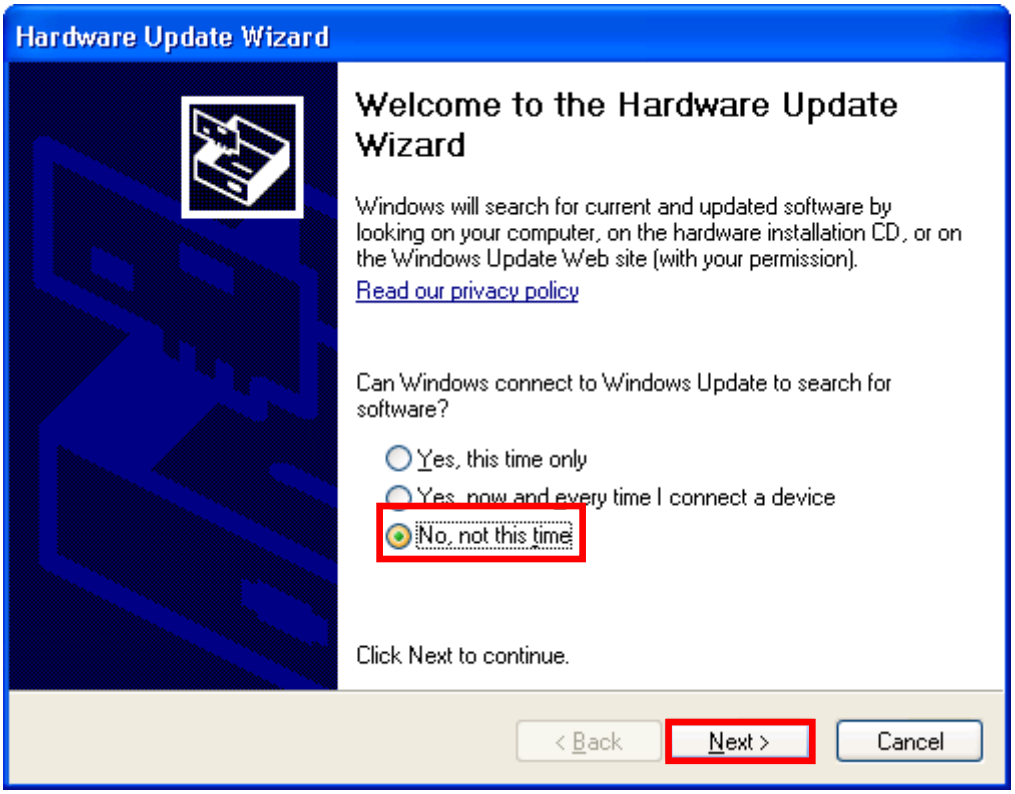


Figure 4.2

Select **Install from a list or specific location [Advanced]** and press **Next >** button.

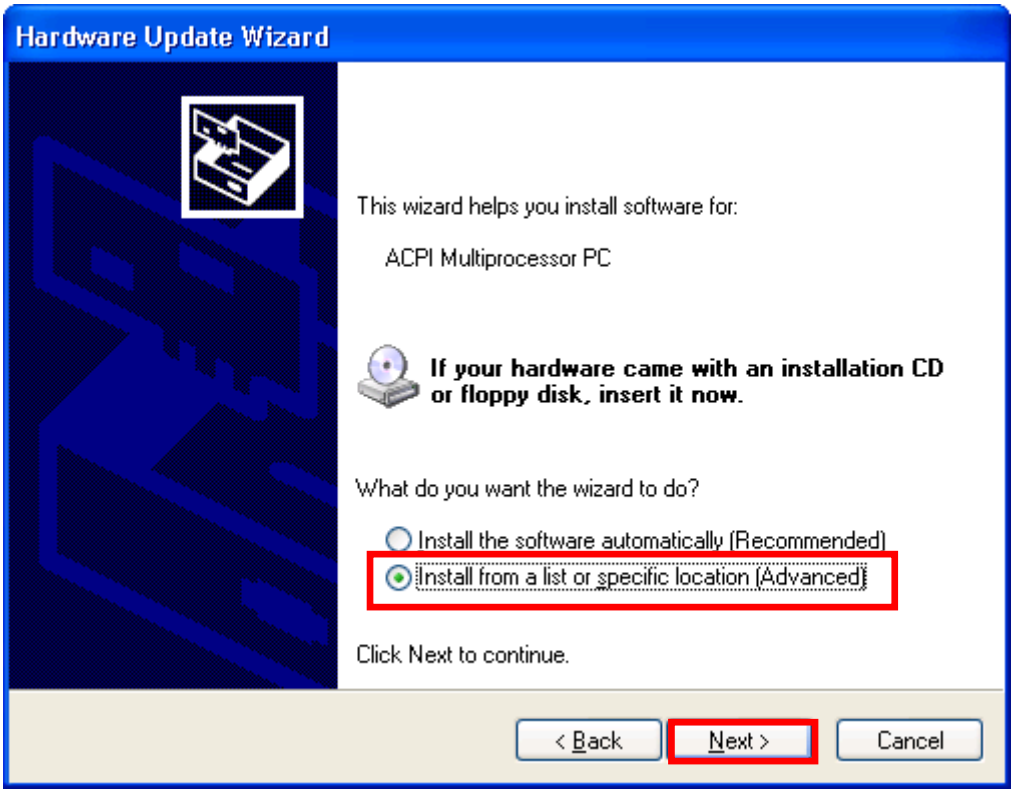


Figure 4.3

Check **Include this location in the search:**, and specify the folder for the driver that you created previously.

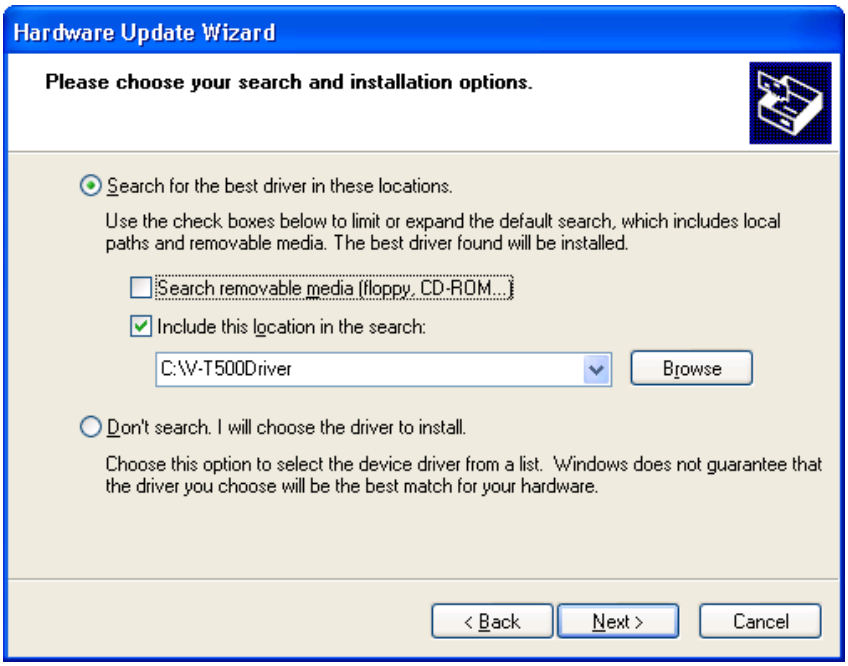


Figure 4.4

Installation is started.

When the dialog below is displayed, installation is finished. Press **Finish** button, and close the dialog.

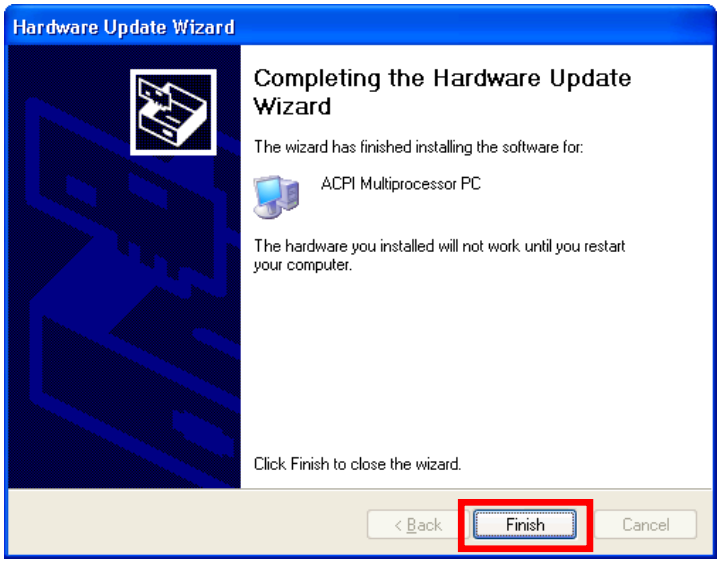


Figure 4.5

For confirmation, open the command prompt, and enter the following command: **adb devices**

Figure 4.6

If the message above is displayed, the installation of the driver has succeeded.

■ In case of Windows 8.1

First, create a folder for the driver (e.g. "C:\V-T500Driver").
Next, search a file named **android_winsub.inf** in the installed folder of **Android SDK**.

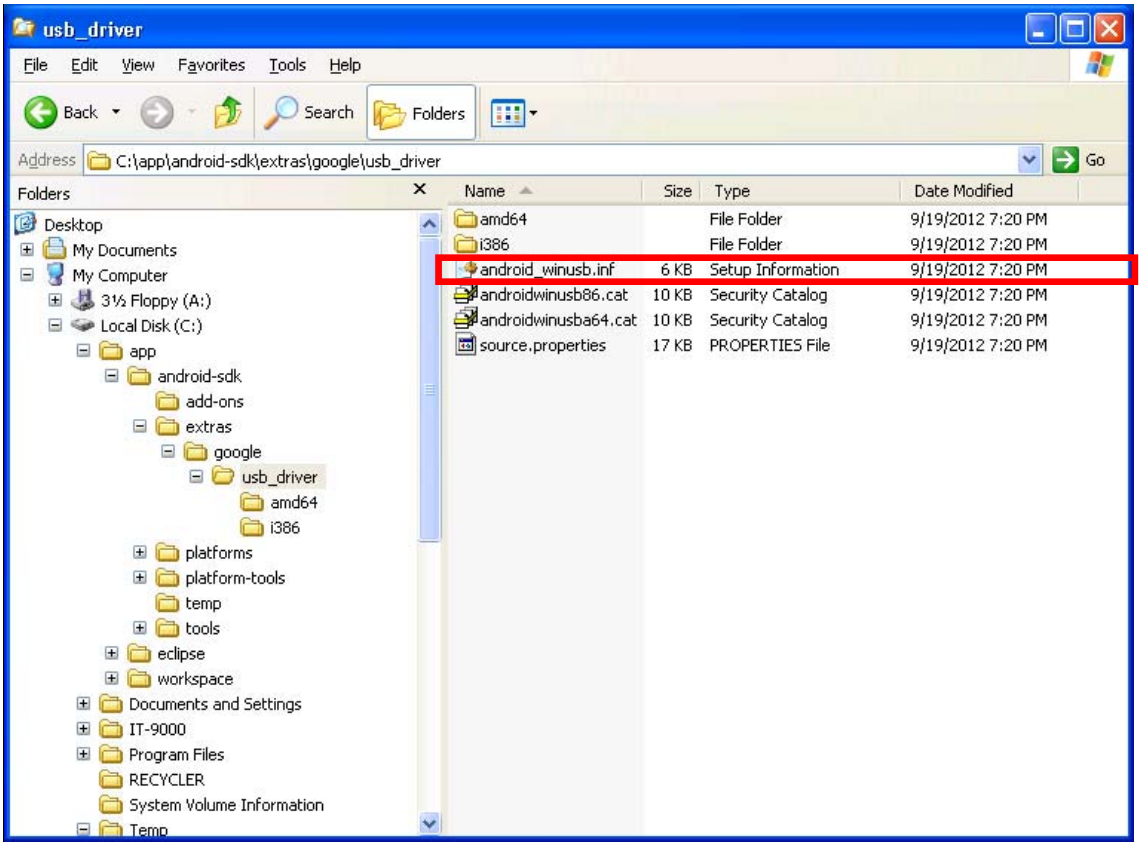


Figure 4.7

When the file **android_winsub.inf** is found, copy it along with the sub folders in the folder for the driver that you created at the beginning.
Then, copy following files from the CD.

V-T500/V-N500 development kit.\V-T500\drv-win8\android_winsub.inf
V-T500/V-N500 development kit.\V-T500\drv-win8\androidwinusb86.cat
V-T500/V-N500 development kit.\V-T500\drv-win8\androidwinusba64.cat

Paste these files in the same folder for the driver as above by overwriting.

Connect the mini-B USB port of **V-T500/V-N500** and the USB port of the PC by a cable.
The following menu will be displayed if start button is right-clicked. Select **Device Manager**.

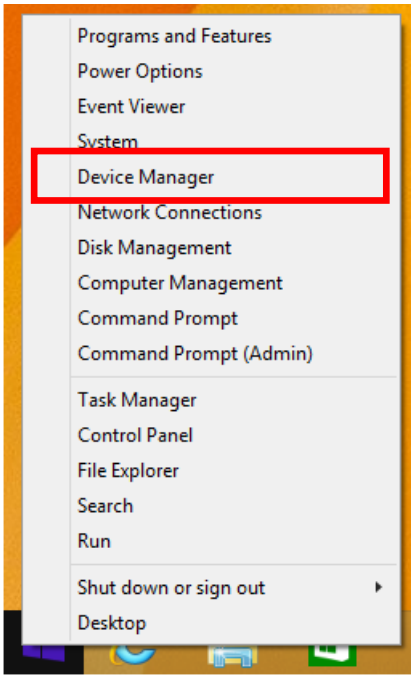


Figure 4.8

As shown in the following figure, the connected device name is displayed.

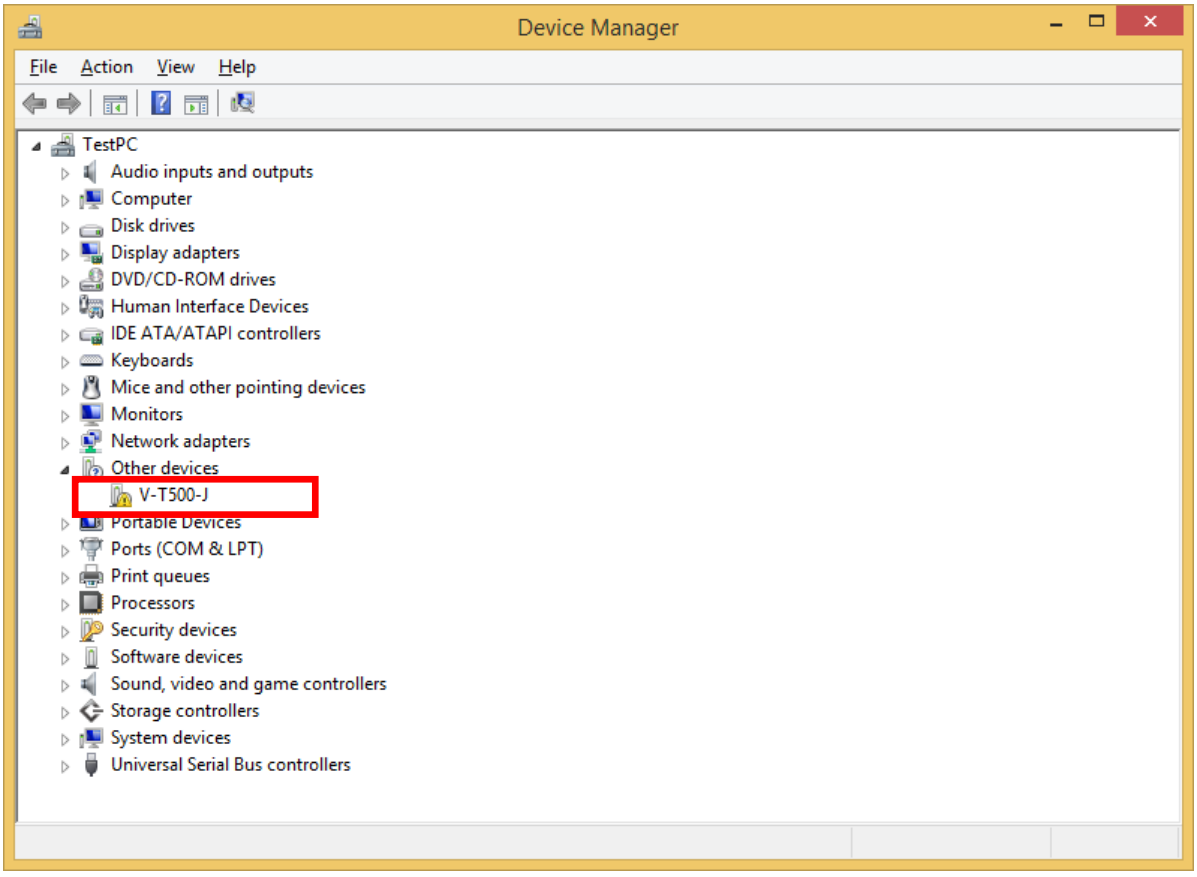


Figure 4.9

The following menu will be displayed if the device name is right-clicked. Select **Update Driver Software**.

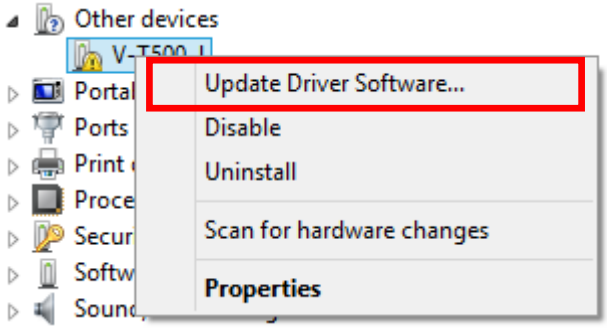


Figure 4.10

Select **Browse my computer for driver software**.

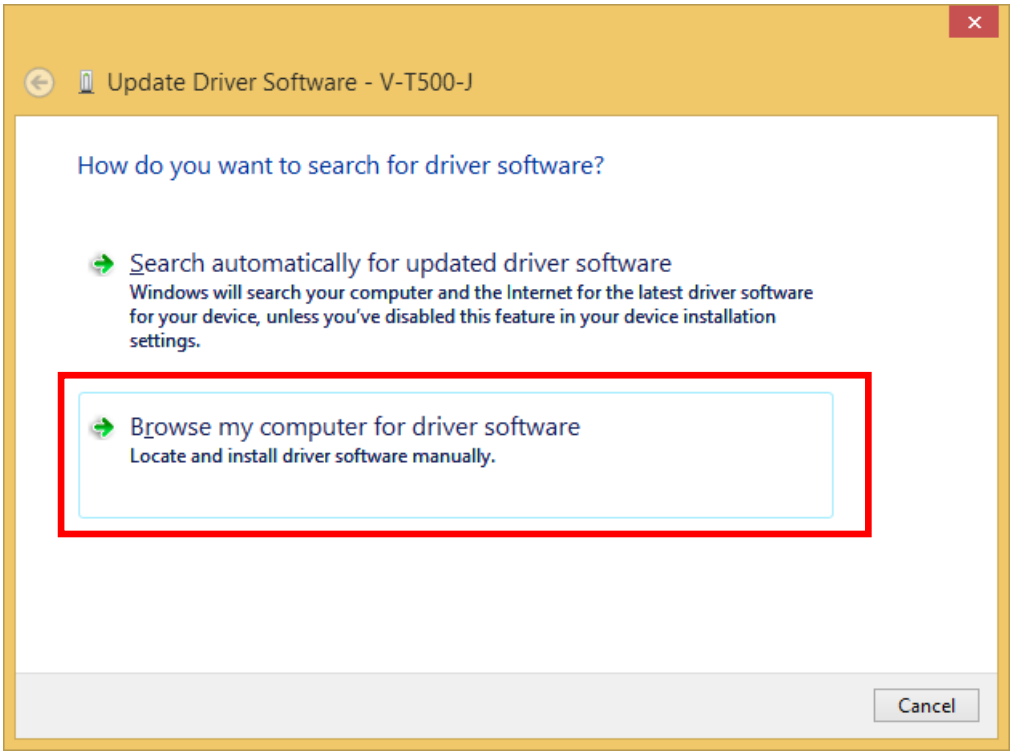


Figure 4.11

Specify the folder for the driver that you created previously, and Press **Next** button.

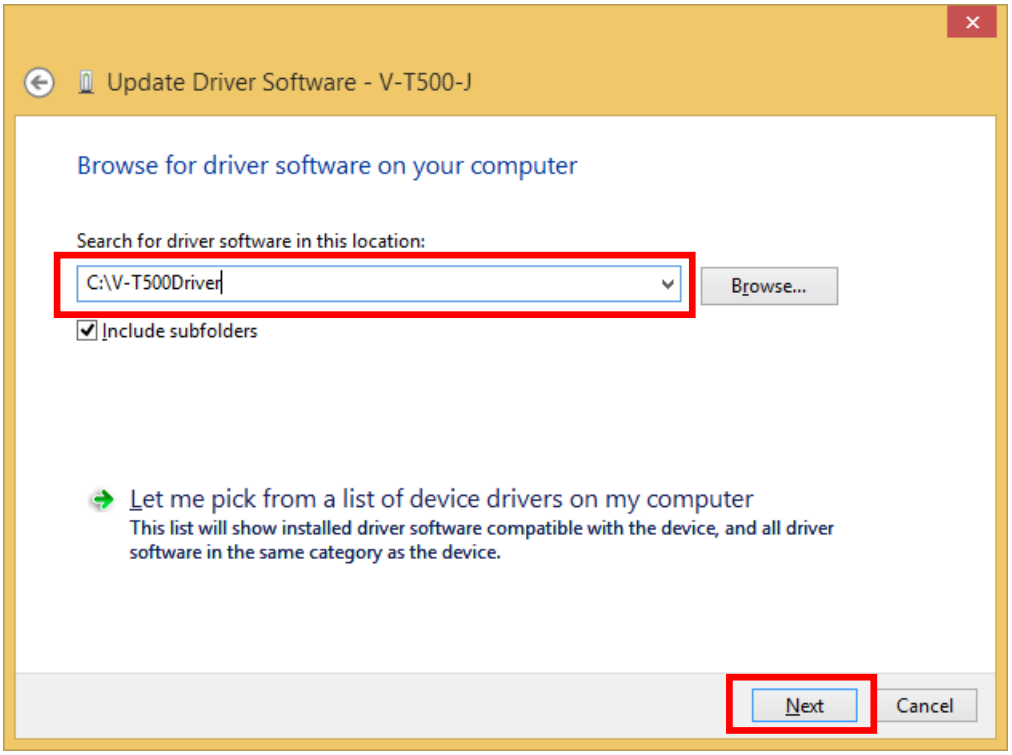


Figure 4.12

The dialog of the following figure is displayed. Select **Install** button.

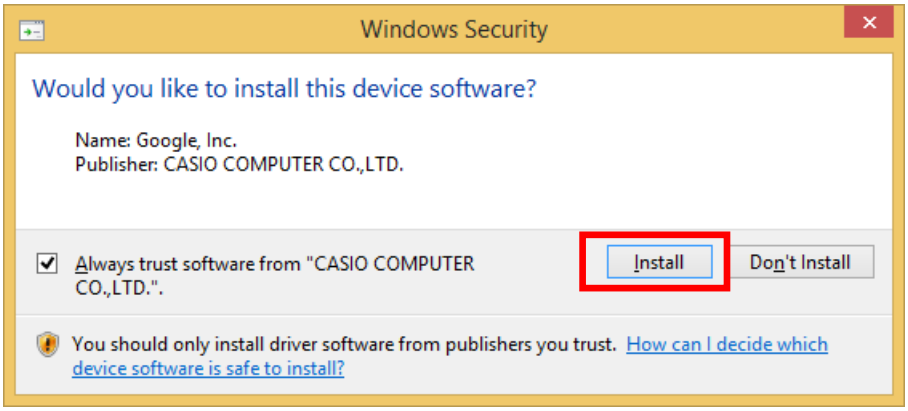


Figure 4.13

When the dialog below is displayed, installation is finished. Press **Close** button, and close the dialog.

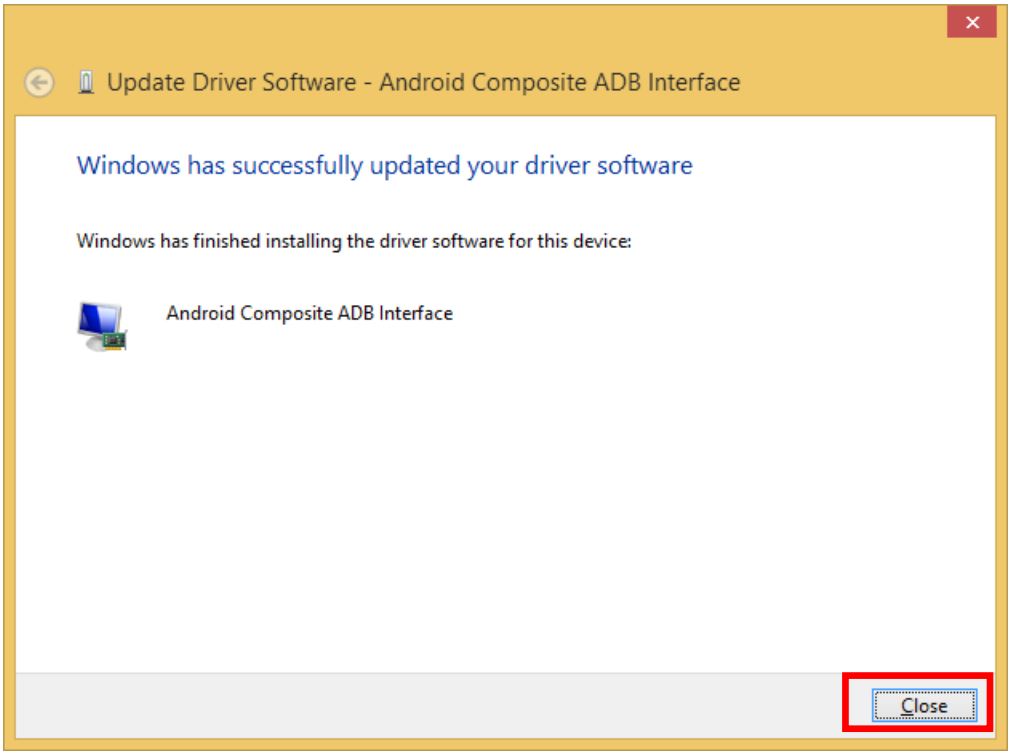


Figure 4.14

For confirmation, open the command prompt, and enter the following command: **adb devices**

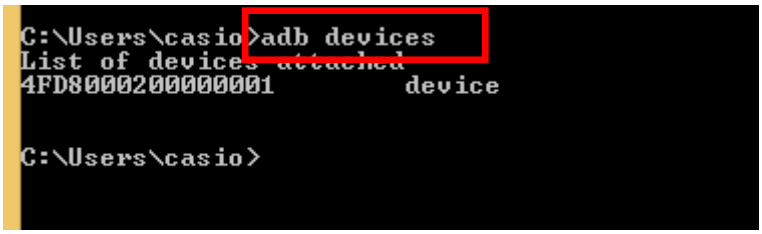


Figure 4.15

If the message above is displayed, the installation of the driver has succeeded.

5. Setting Up the Development Environment

5.1 Setting JDK

Set the JDK installation version.

Start up Eclipse, and select **Preferences** of the **Window** menu.

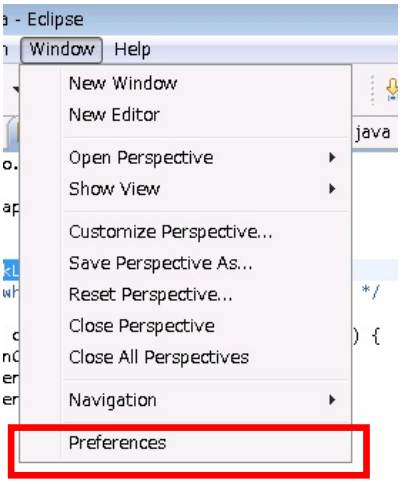


Figure 5.1

Select **Java** → **Compiler**, and set the level to **1.6**.

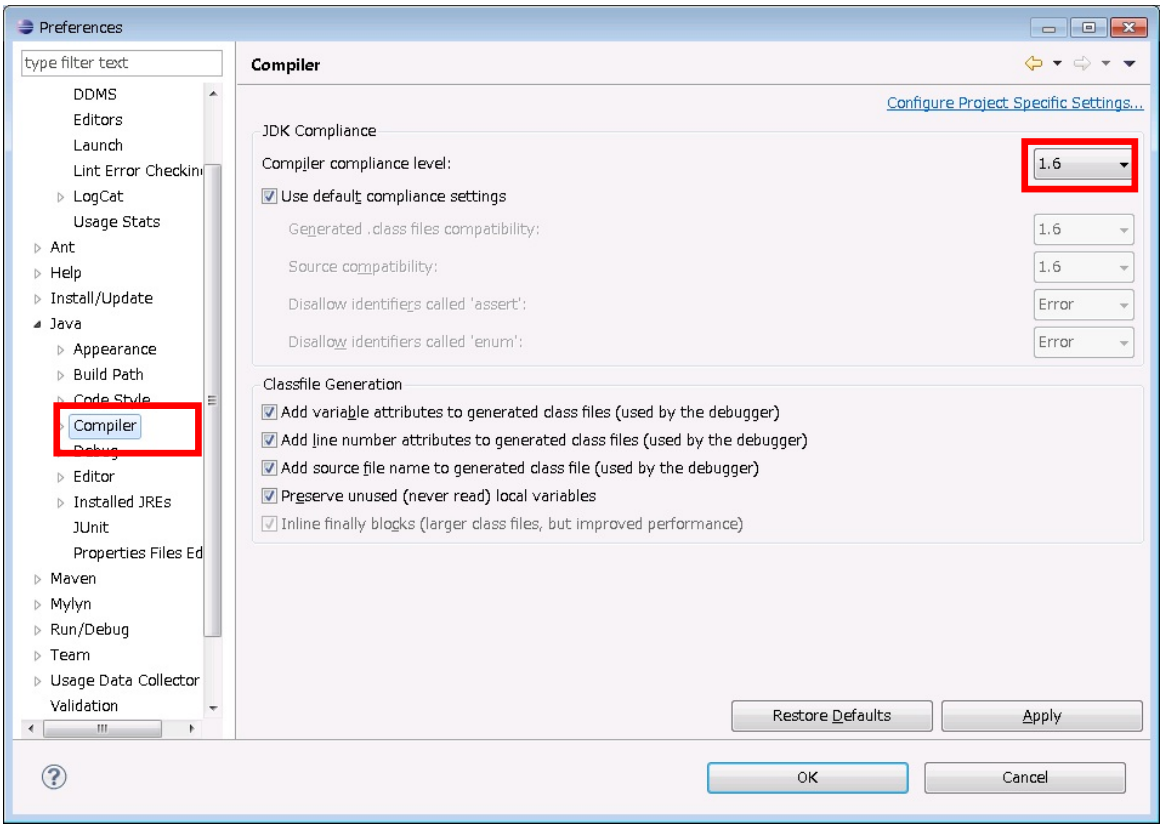


Figure 5.2

5.2 Setting AVD

Set the **AVD** (Android Virtual Device).

Start up **Eclipse**, and select **AVD Manager**.

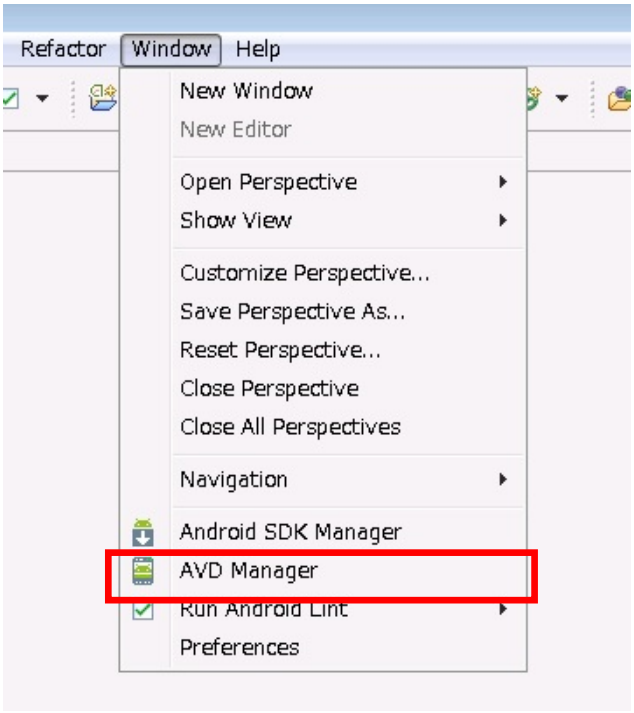


Figure 5.3

Press the **New...** button, and create a new AVD.

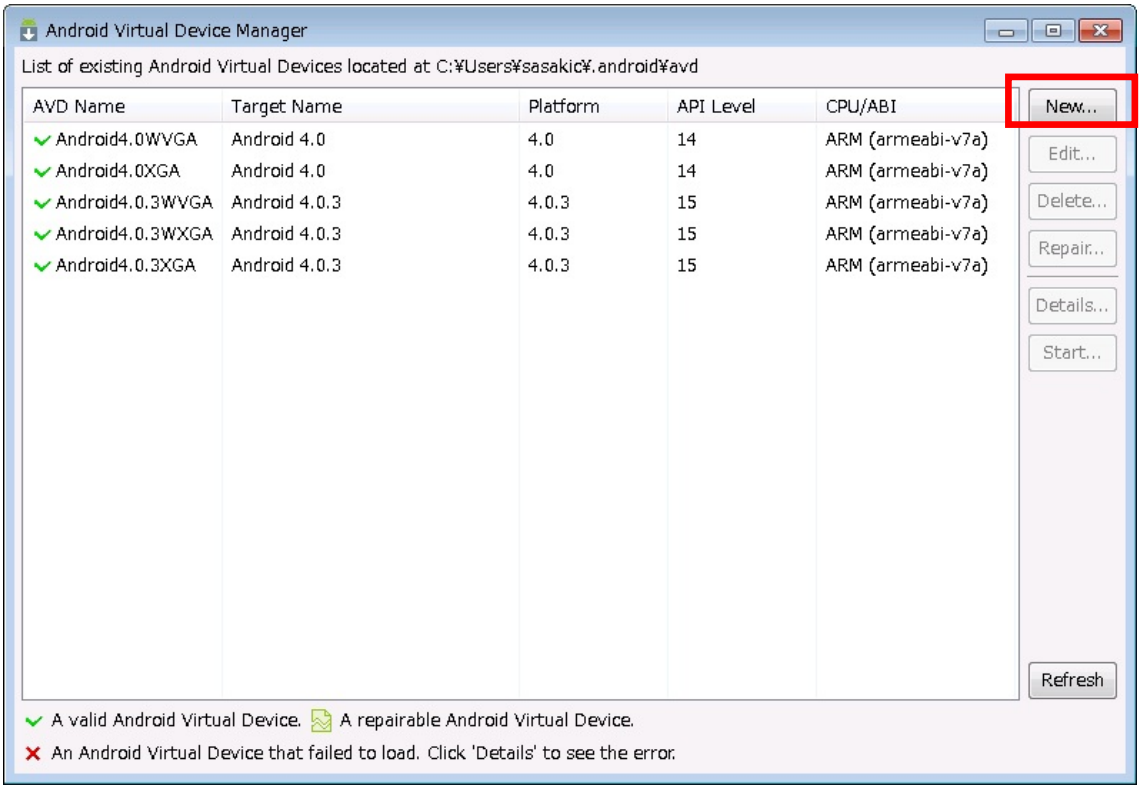


Figure 5.4

Enter "**CASIO_V_T500**" in **Name**, and select "**CASIO V-T500/V-N500(CASIO COMPUTER CO., LTD.) - API Level 15**" at Target, and press the **Create AVD** button.

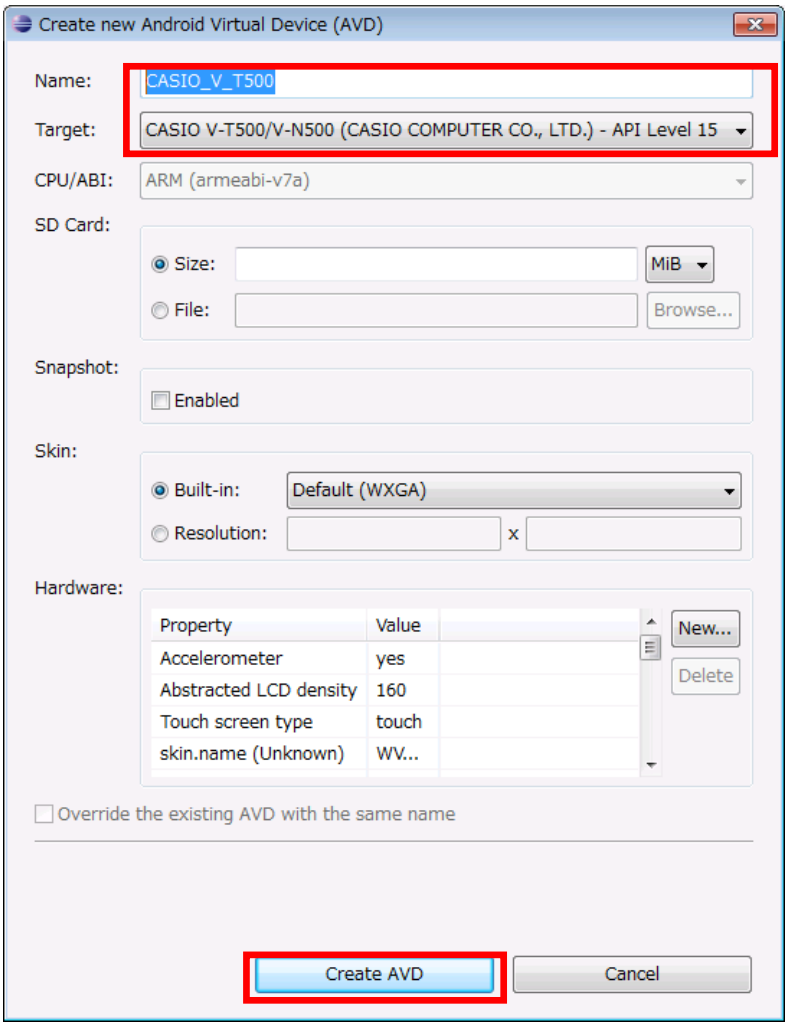


Figure 5.5

Return to the Android Virtual Device Manager screen, confirm that "**CASIO_V_T500**" is added, and close the dialog to finish.

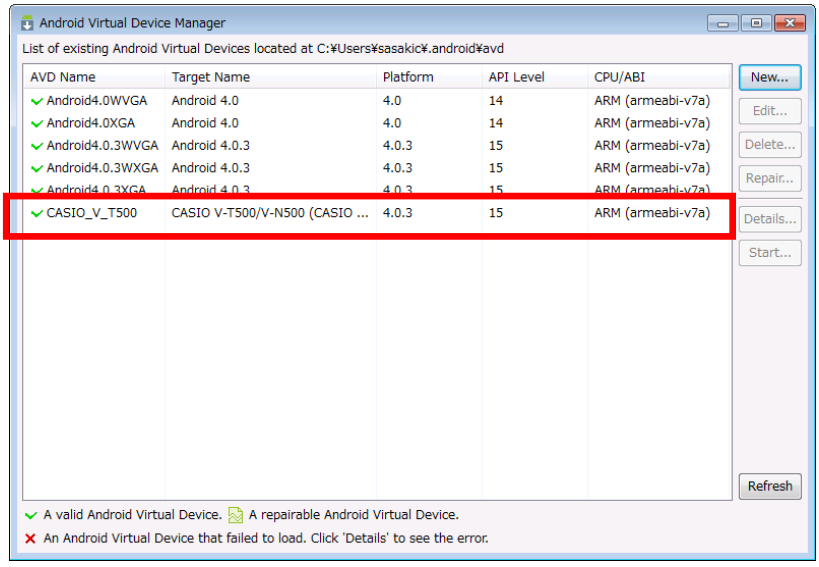


Figure 5.6

5.3 Setting Android Project

Set the Android Project of the application created by **Eclipse**.

Select and right-click Android Project, and select **Properties**.
Select **Android** in the displayed Properties dialog, and select **CASIO V-T500/V-N500** at **Target Name**.

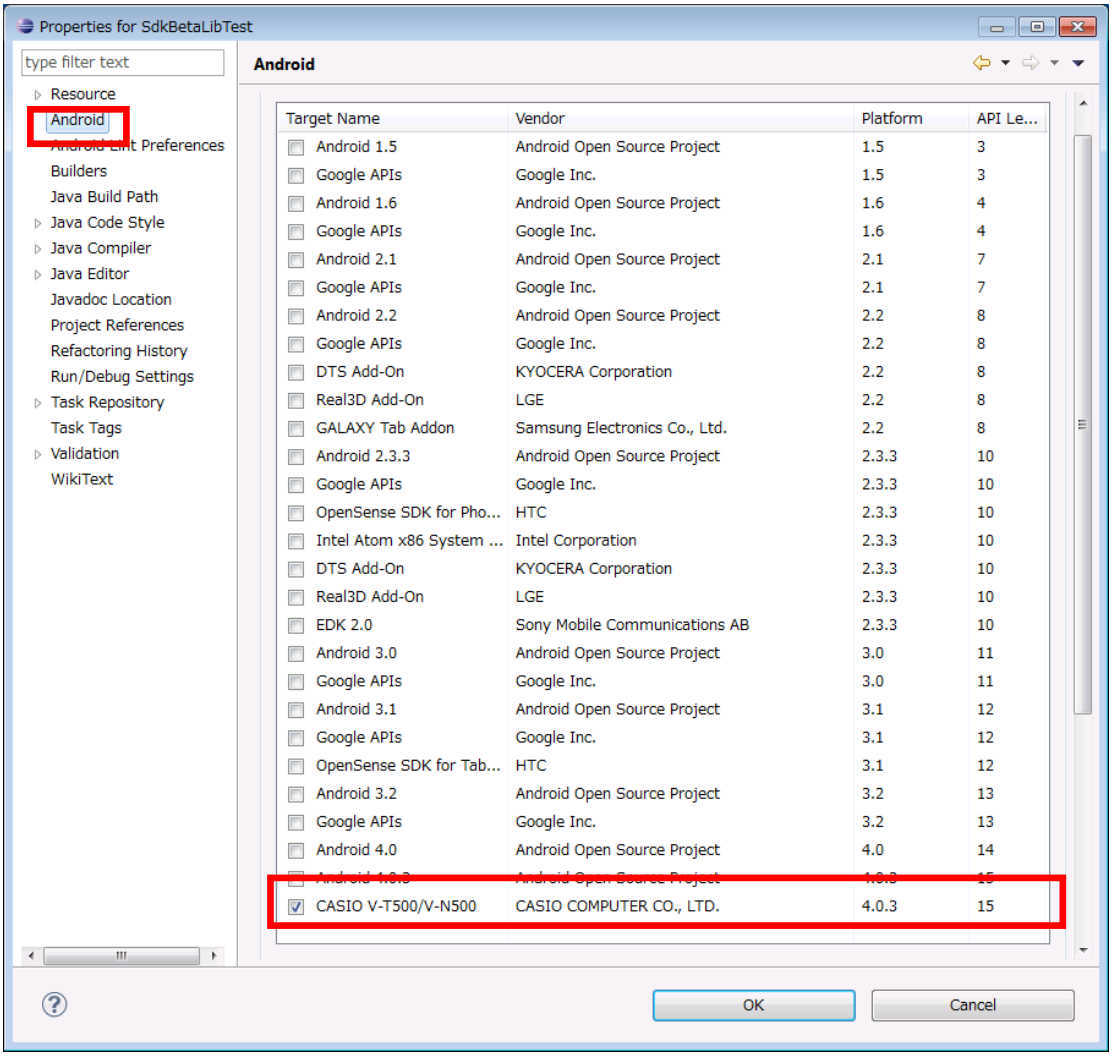


Figure 5.7

Next, select **Java Build Path**.
Press **Add External JARs...**button in the **Libraries** tab, and display the file selection dialog.

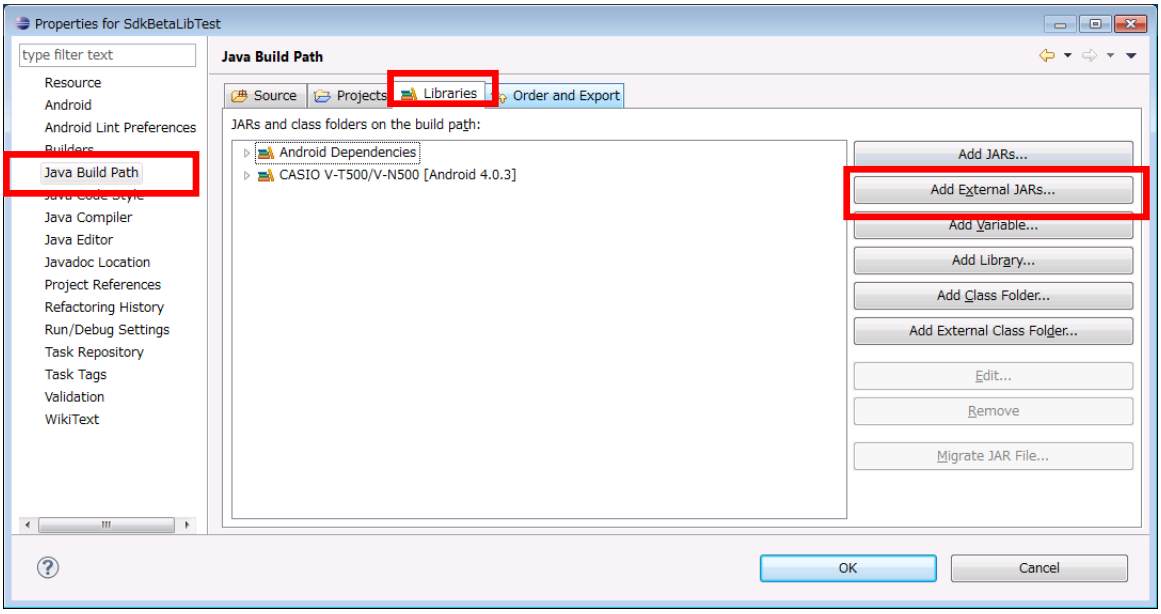


Figure 5.8

Open the **libs** folder under the SDK, and select all files.

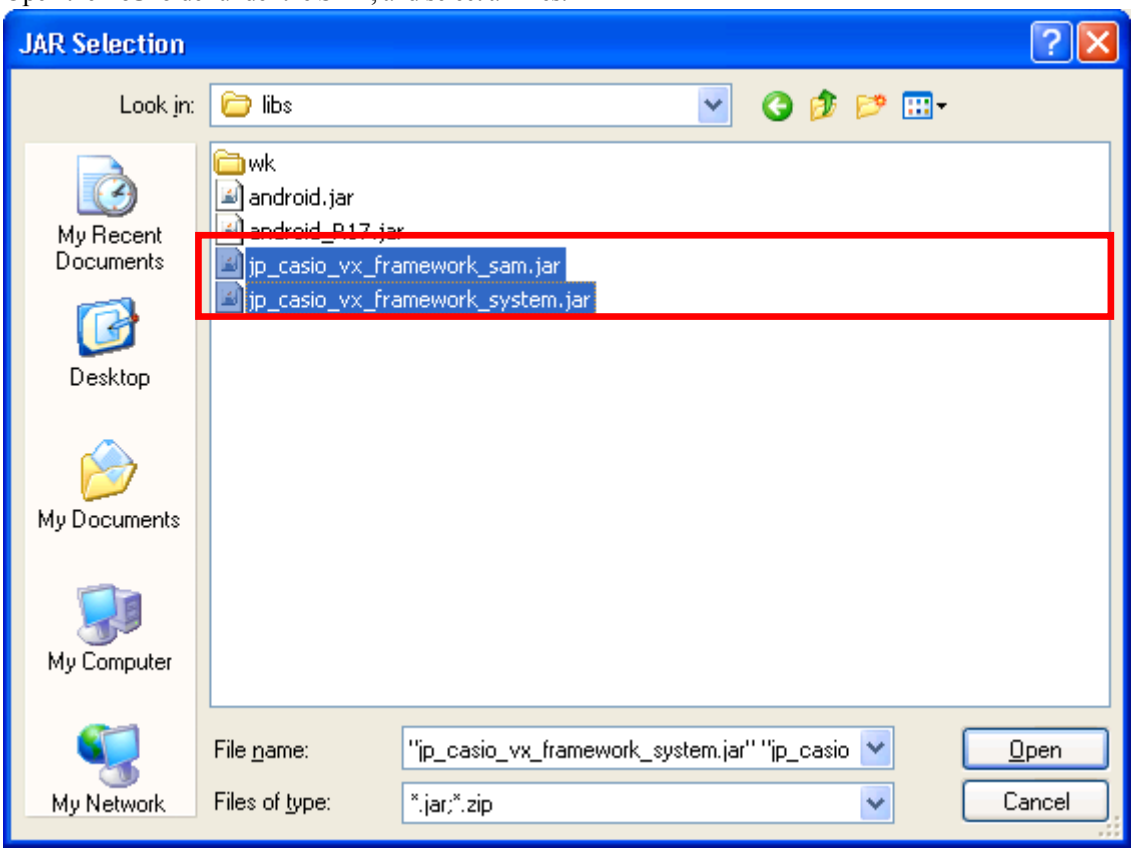


Figure 5.9

Next, open the **Order and Export** tab, check all the added files, and press **OK** button to finish.

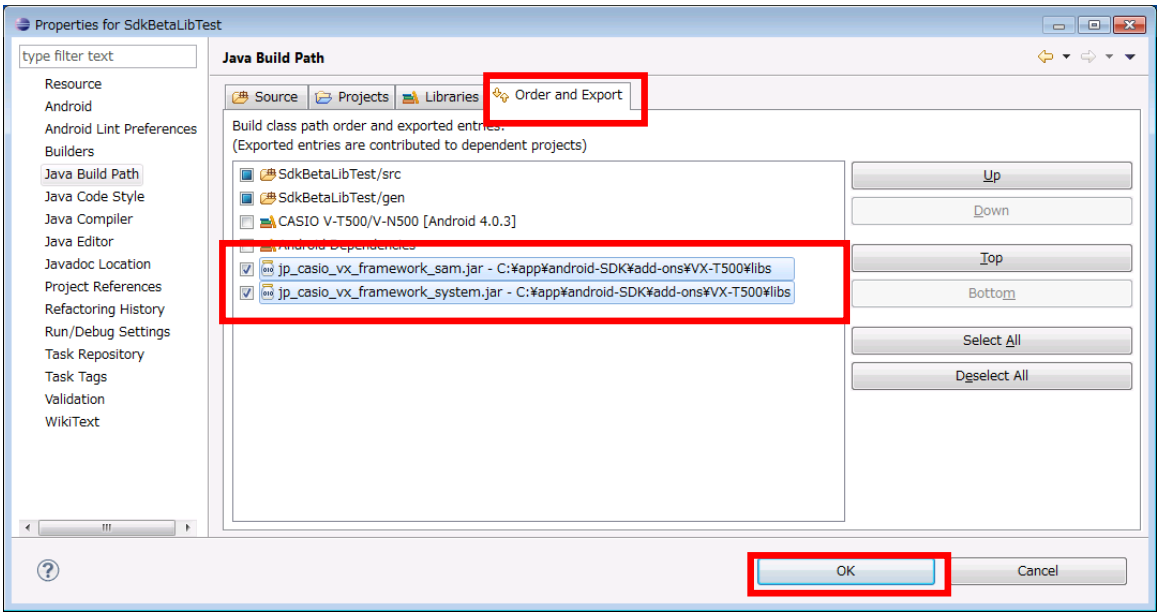


Figure 5.10

5.4 Setting Up the Debugging Environment

Set the debugging environment of the application.
Note: For the procedure to create the application, refer to Chapter 7Eclipse.

Use the actual terminal of **V-T500/V-N500** or the emulator for debugging.

5.4.1 Setting the Debug Configuration with Terminal

Select **Debug Configurations…** from the **Run** menu of **Eclipse**.

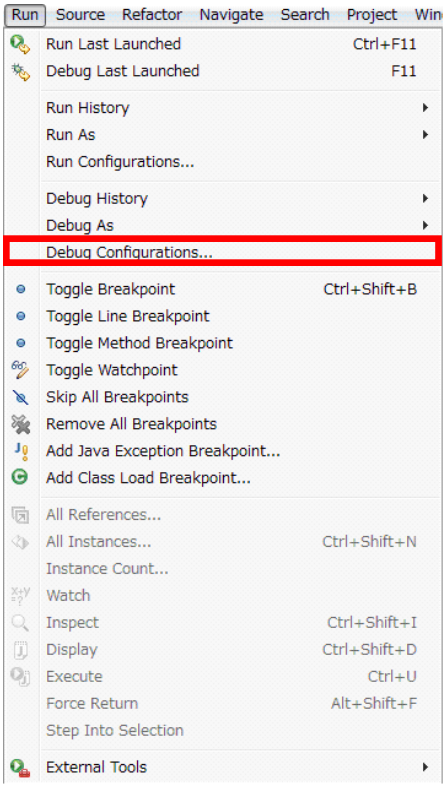


Figure 5.11

Select the **Android** tab, and enter the debugging project name in **Project**.

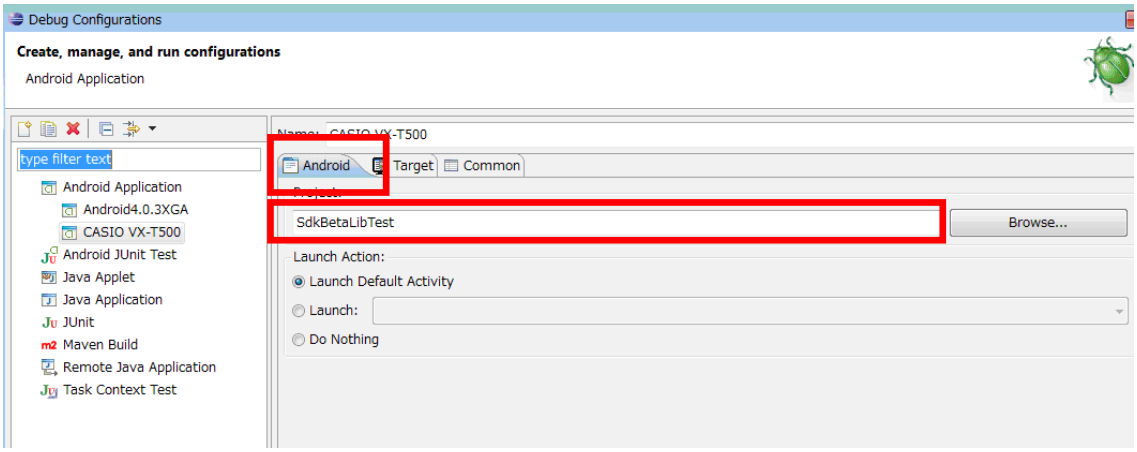


Figure 5.12

Switch to the **Target** tab, select **Always prompt to pick device**, and close the dialog with the **Close** button.

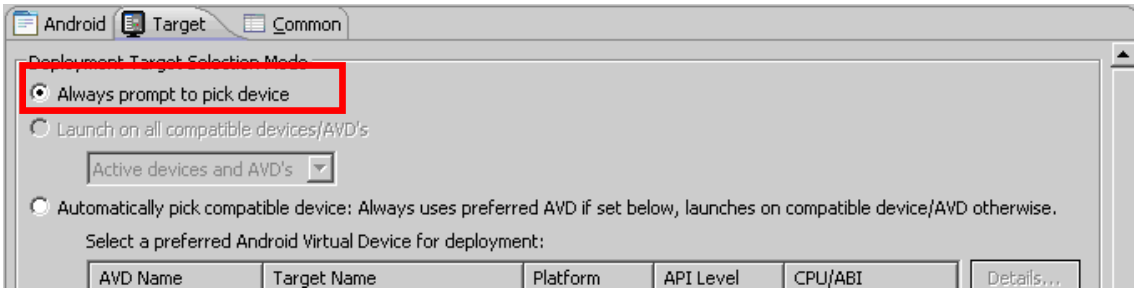


Figure 5.13

Set the USB drier as described in the Chapter 4Connecting V-T500/V-N500 to PC before connecting the actual terminal of **V-T500/V-N500** and the PC.

5.4.2 Setting the Debug Configuration with the Emulator

Select **Debug Configurations** from the **Run** menu of Eclipse in the same way as the actual terminal of **V-T500/V-N500**.
Enter the details of **Android** tab, the debugging project name, in the same way as the actual terminal of **V-T500/V-N500**.

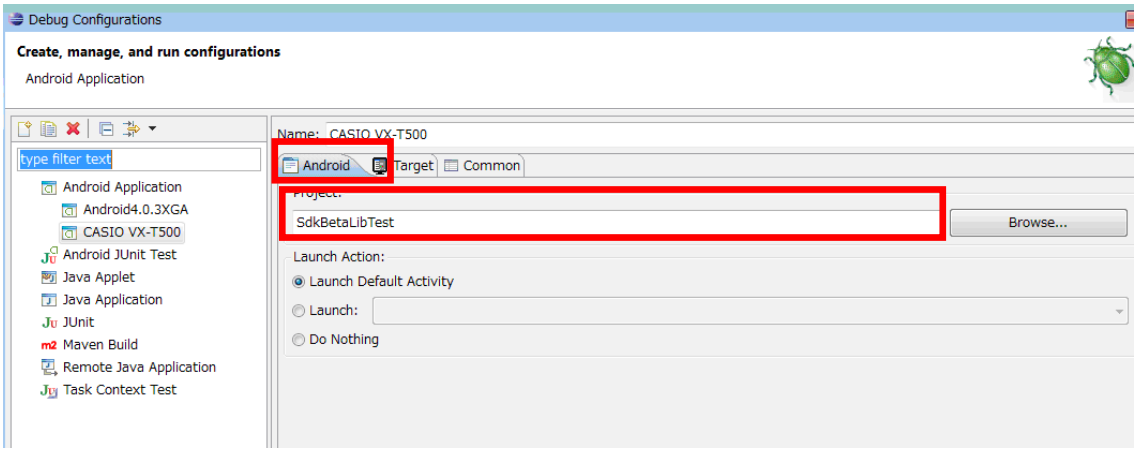


Figure 5.14

Next, select the **Target** tab.
Select **Automatically pick compatible device:~**, and check **CASIO_V_T500** in the check box below.

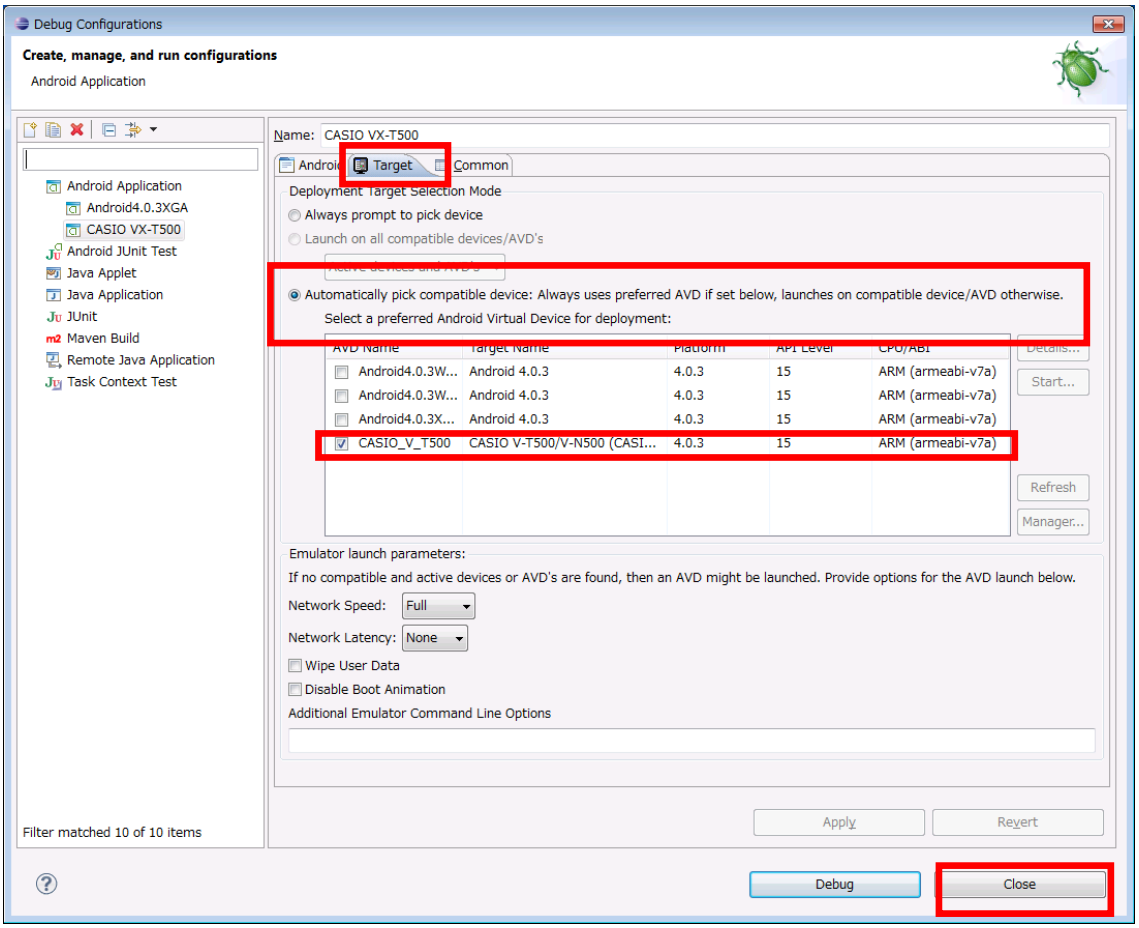


Figure 5.15

Press **Close** button, and close the dialog.

5.4.3 Debugging Applications

Basic operation of debugging

The debugging method is the same as the normal debugging operation of **Eclipse**.
Select **Debug** from the **Run** menu of **Eclipse**, or press F11 to start debugging.

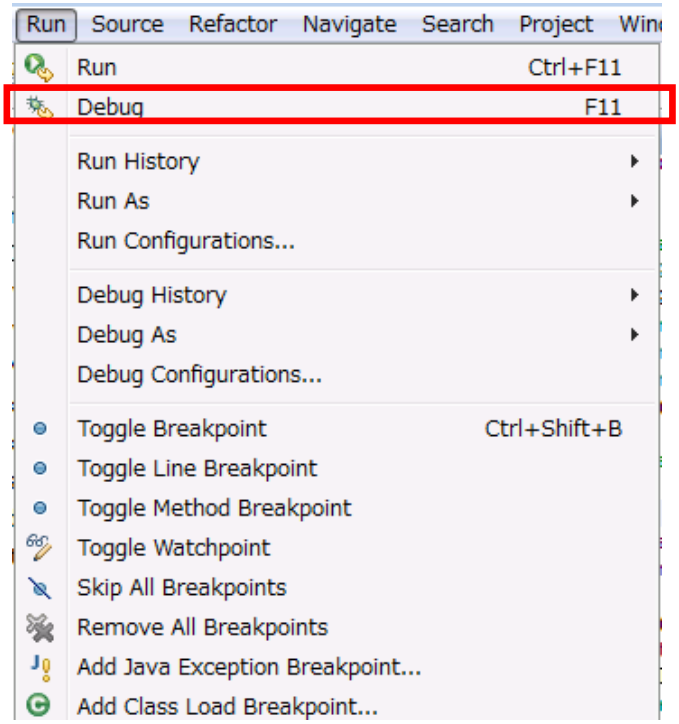


Figure 5.16

Debugging by the actual terminal (V-T500/V-N500)

Before debugging, connect the actual terminal to the PC with a USB cable.
Set the USB drier as described in the Chapter 4Connecting V-T500/V-N500 to PC before connecting the actual terminal and the PC.

When **Debug** is selected from the **Run** menu, the screen for selecting the device as shown below is displayed.

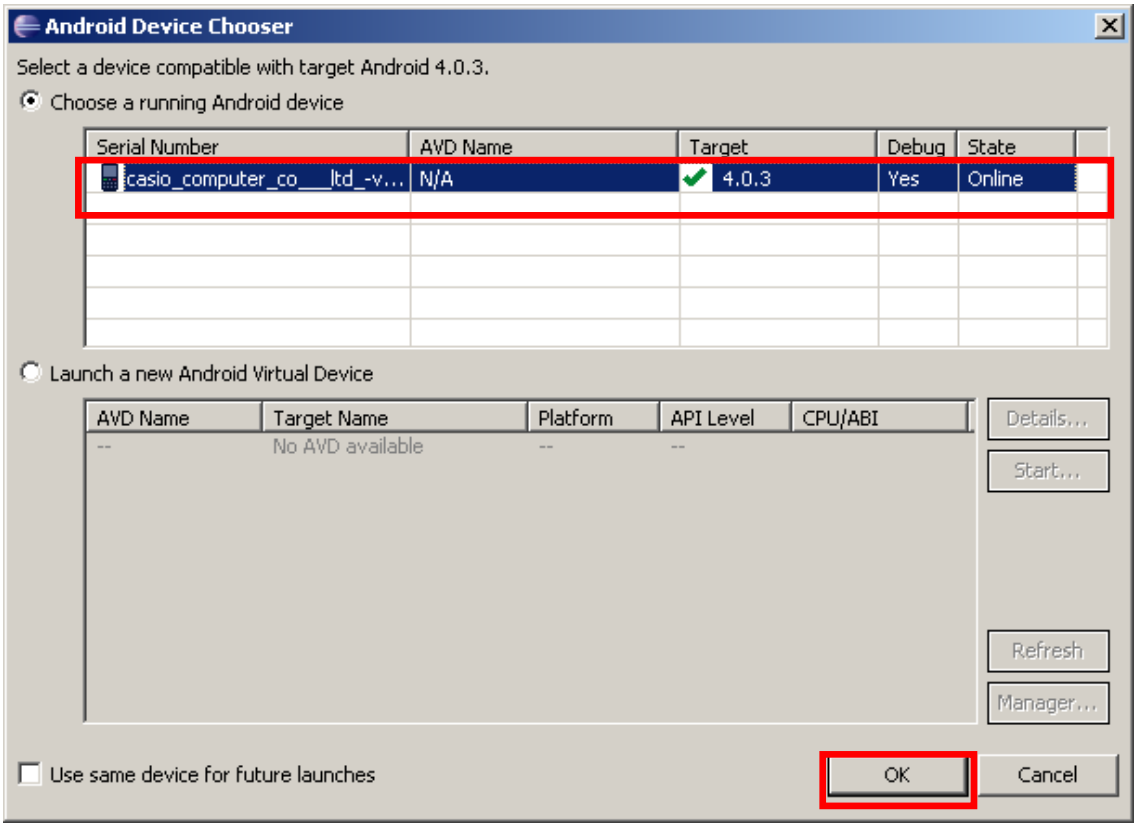


Figure 5.17

Select the target device, and press the **OK** button to start debugging.
It is possible to execute debugging by stepping, etc. on the actual terminal by setting breakpoints in the source code.

Precautions for debugging

- It is not possible to execute debugging when the debugging application is installed on the actual terminal of **V-T500**/**V-N500**. In such a case, uninstall the debugging application on the actual terminal of **V-T500**/**V-N500** before executing debugging.

Debugging with the emulator

When **Debug** is selected, debugging starts on the emulator if an existing emulator is operating, and by starting the emulator newly if no emulator is operating.
It is possible to execute debugging by stepping, etc. on the **V-T500**/**V-N500** emulator by setting breakpoints in the source code.

6. Emulator

The emulator provides an "**environment where basic operation check of the application and source level debugging are possible without the actual terminal of V-N500**" to the application developer.

6.1 Software Required

It is possible to use the emulator if you have built the development environment explained above.

6.2 Starting Up the Emulator

6.2.1 How to Start Up the Emulator

Start **AVD Manager** from the **Windows** menu of **Eclipse**.

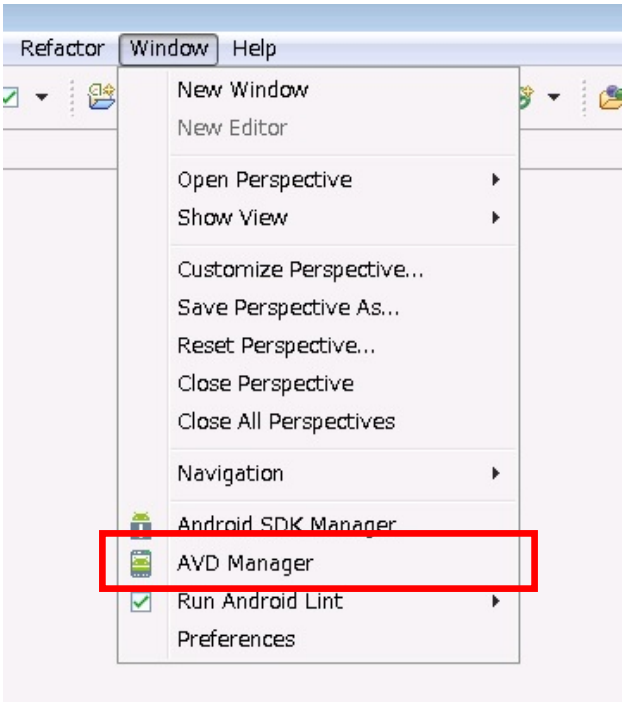


Figure 6.1

Select **CASIO_V_T500** and press the **Start** button.

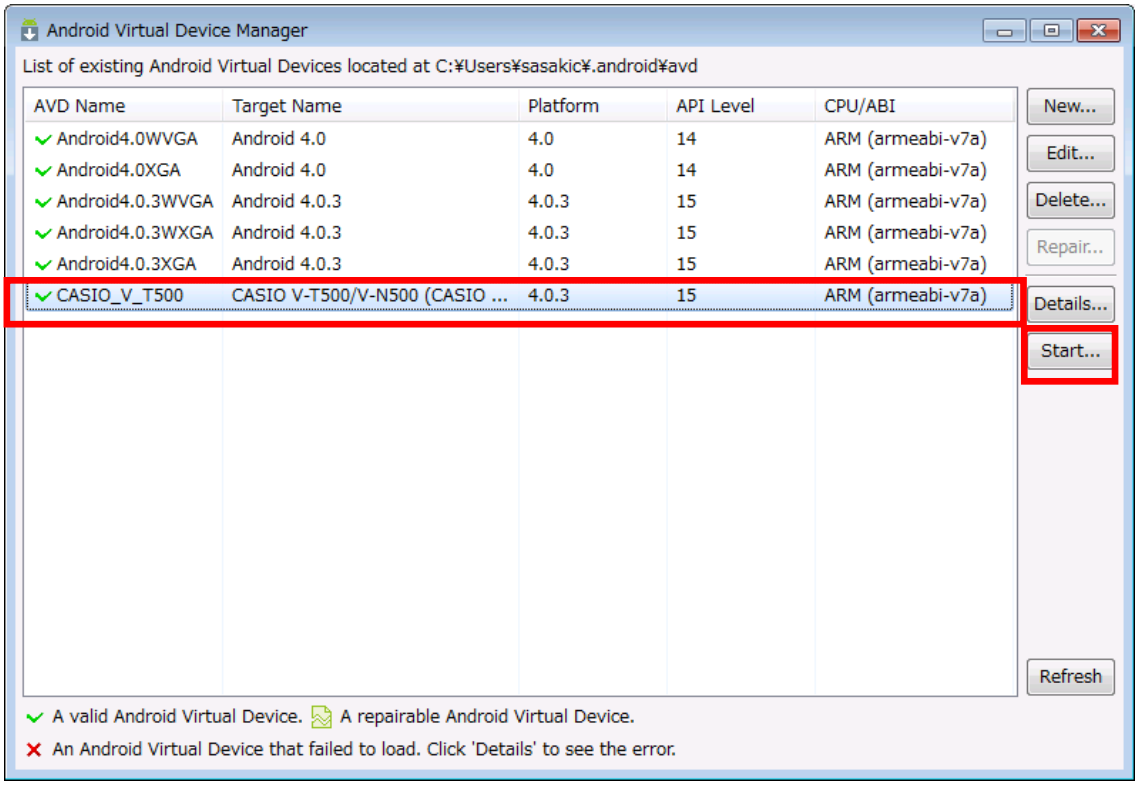


Figure 6.2

The dialog below is displayed. Press **Launch** as is.

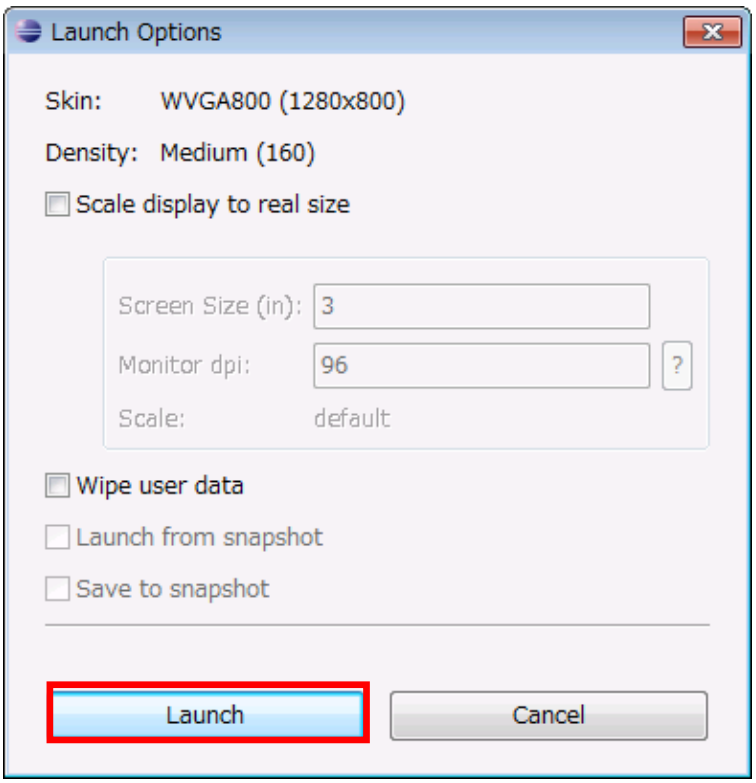


Figure 6.3

The dialog below is displayed, and the process of start up is displayed.

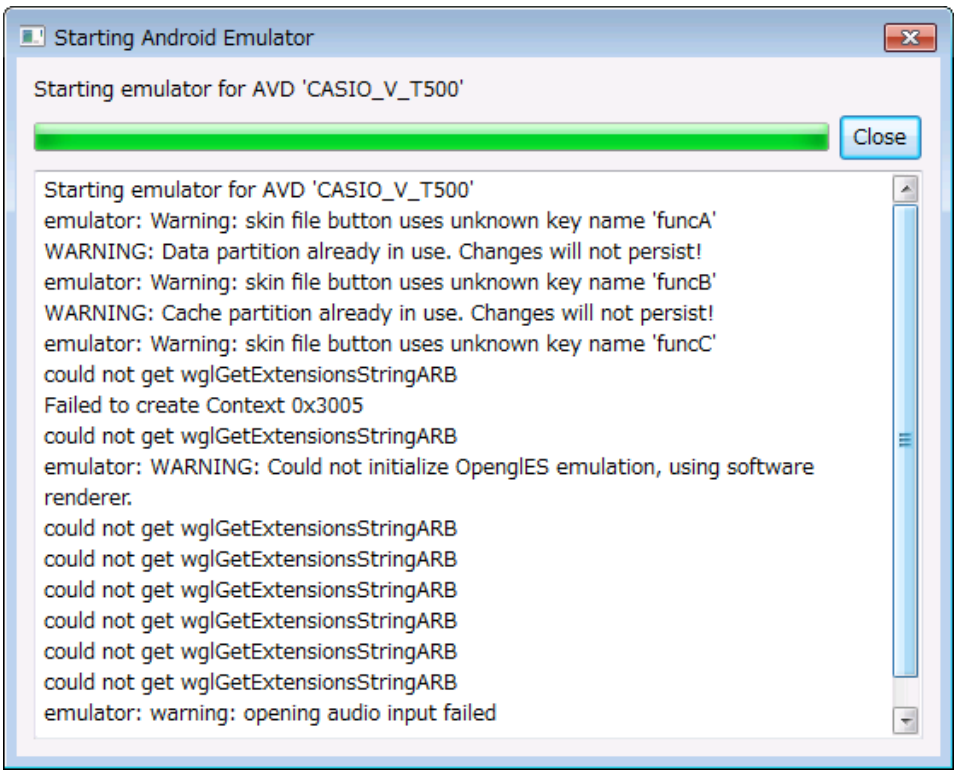


Figure 6.4

The emulator starts.

Figure 6.5

6.3 Using the Emulator

6.3.1 V- T500/V-N500 Emulator

Key input function

The key input function equivalent of the actual terminal (**V-T500**/**V-N500)** is incorporated.
It is possible to use key input by clicking the keys on the screen with a mouse.

Function to save the state

The **V-T500**/**V-N500** emulator does not have a function to save the state when it finishes.
The clock and the file system of the emulator are initialized when it starts up.

7. Eclipse

The application development procedure using **Eclipse** as the development environment is explained in this chapter.

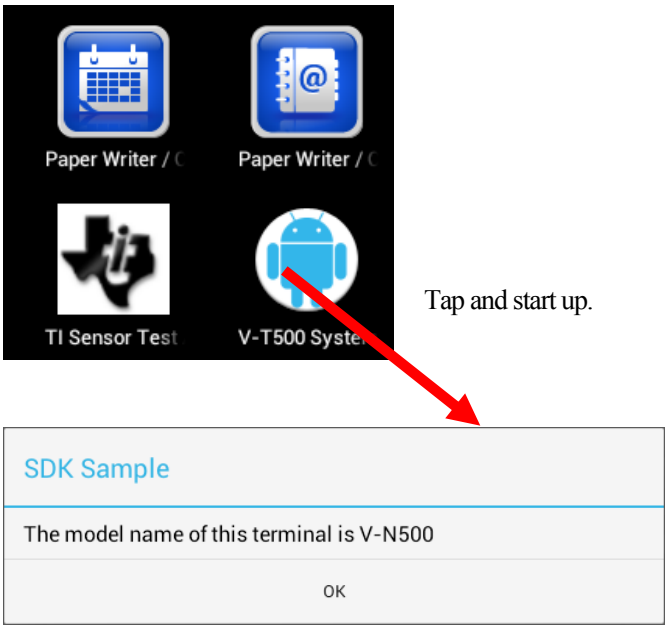
The example is an application that beeps a large-sound buzzer once for a second using the system library. Create this application in a Java programming language.

For the list of libraries provided with the **V-T500/V-N500** basic development kit, refer to the Chapter 1.2 Library Configuration.
For the details of the system library, refer to **System Library Manual**.

The example program of this chapter does not operate because the emulator does not have a large sound buzzer. Check the operation on the actual terminal of **V-T500/V-N500**.

7.1 Developing with System Library

In this section, the GUI application that gets the model name of terminal in Java is explained.



The confirmation dialog is displayed when it operates normally.

Figure 7.1

1. Create a new Android application on **Eclipse**.

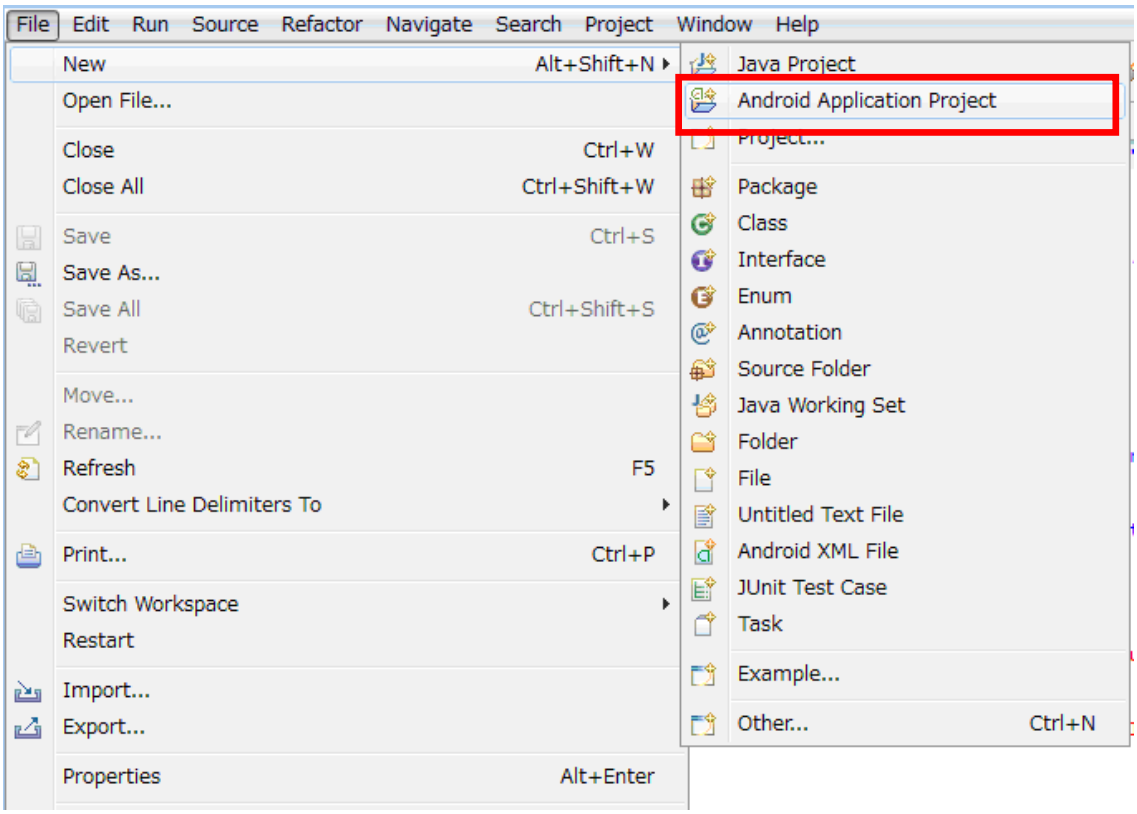


Figure 7.2

The application name is the name of the application created. This name is displayed on the menu of Android. It is named **SDKSample1** in this example.

The project name is the name of the project created in the workspace of **Eclipse**.
Two or more projects of the same name cannot exist in the workspace. The project name is **SDKSample1** in this example.

The package name is the name to avoid collision of the class names of Java during execution. Since it is desirable to have a unique name in the world, the domain name of the developer is normally used.
Note that the hierarchy of the package becomes lower in the order of left → right with Java, which is opposite the domain description by URL.
It is named **jp.casio.vx.framwork.sdksample1** in this example.

All the other settings are default in this example.

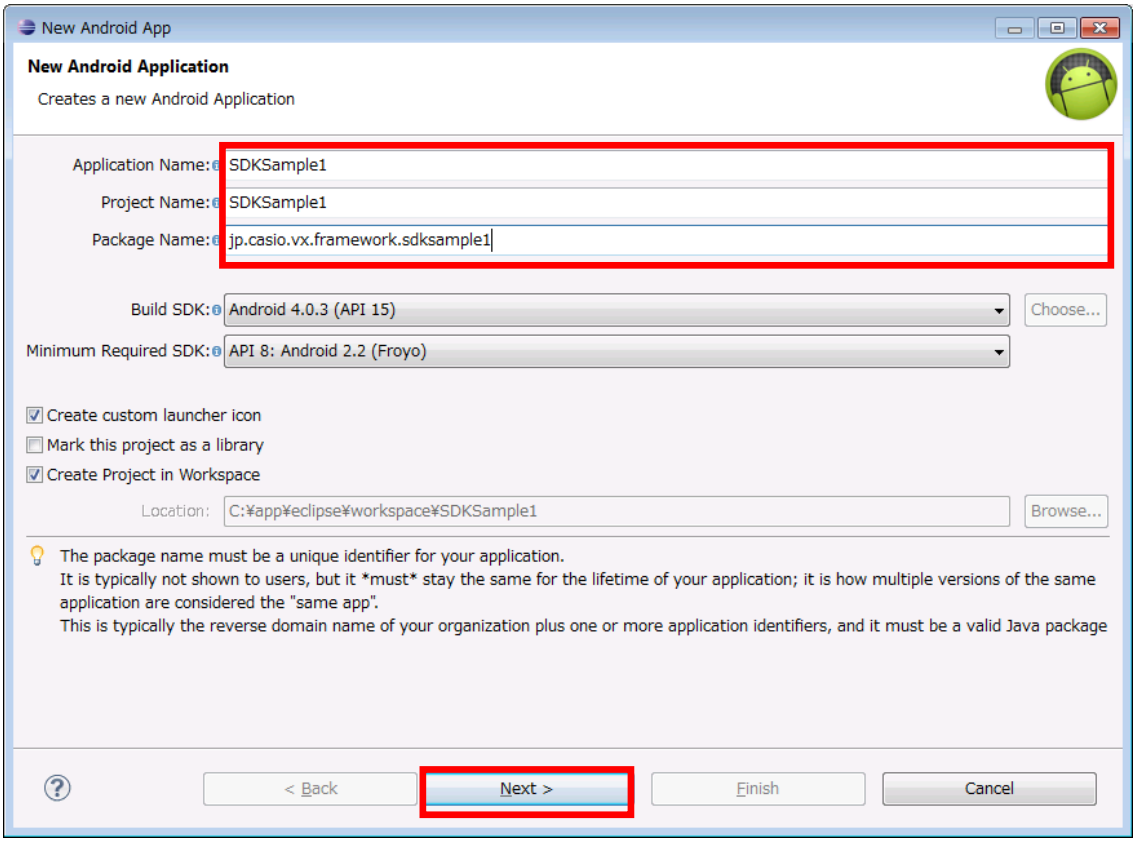


Figure 7.3

Next, set the launcher icon. Select the typical Android icon in this example.

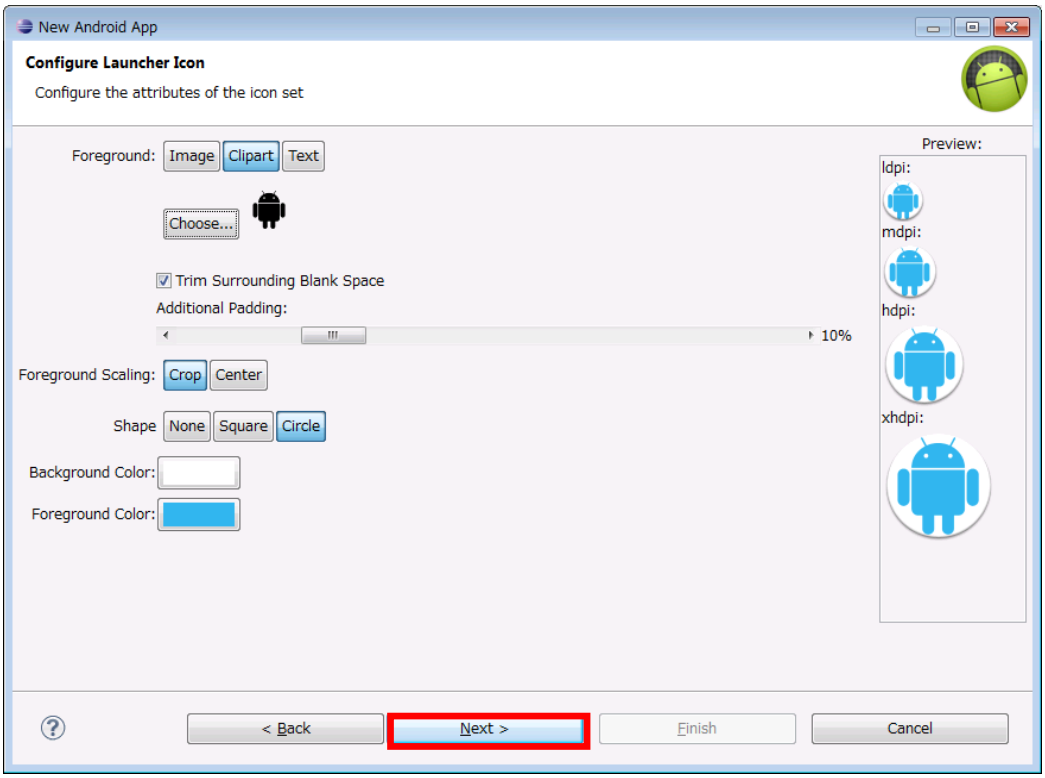


Figure 7.4

Next, set the activity.
Activity is one of the Java classes in specific. It works as units of the user interface of the Android application,

and display, response to events, etc. are included.
The title is changed to **V-T500 System Lib sample** in this example. The title is displayed as the screen title when executing an application or the menu title.

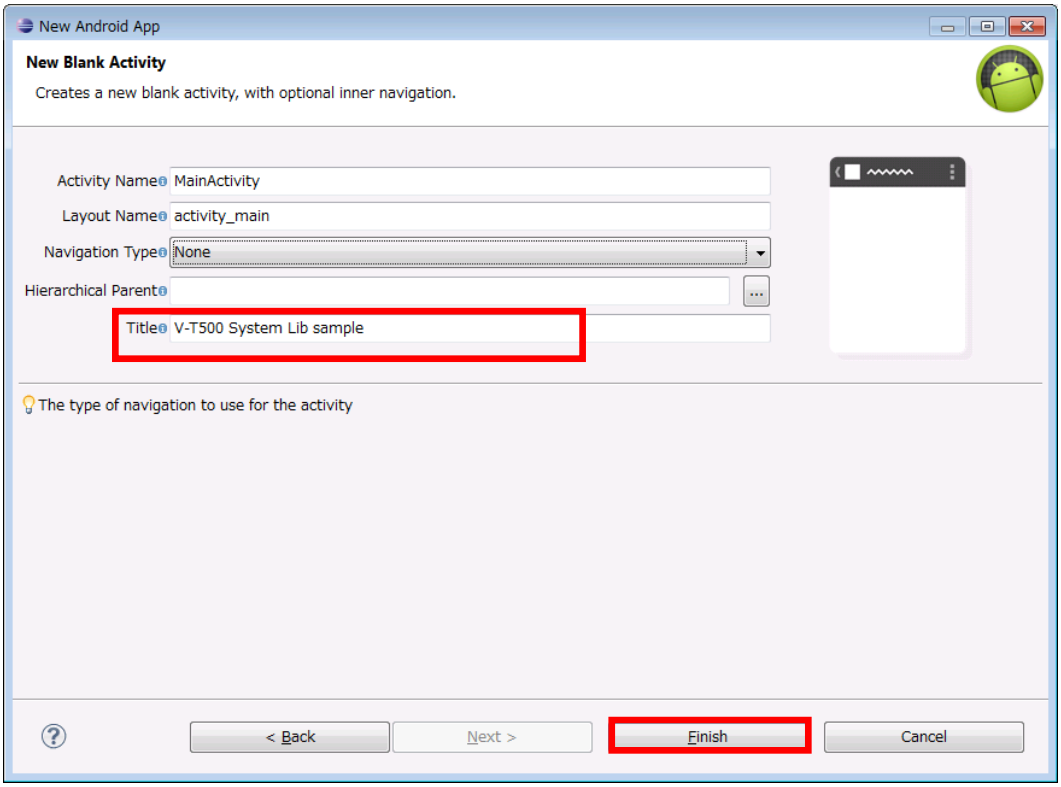
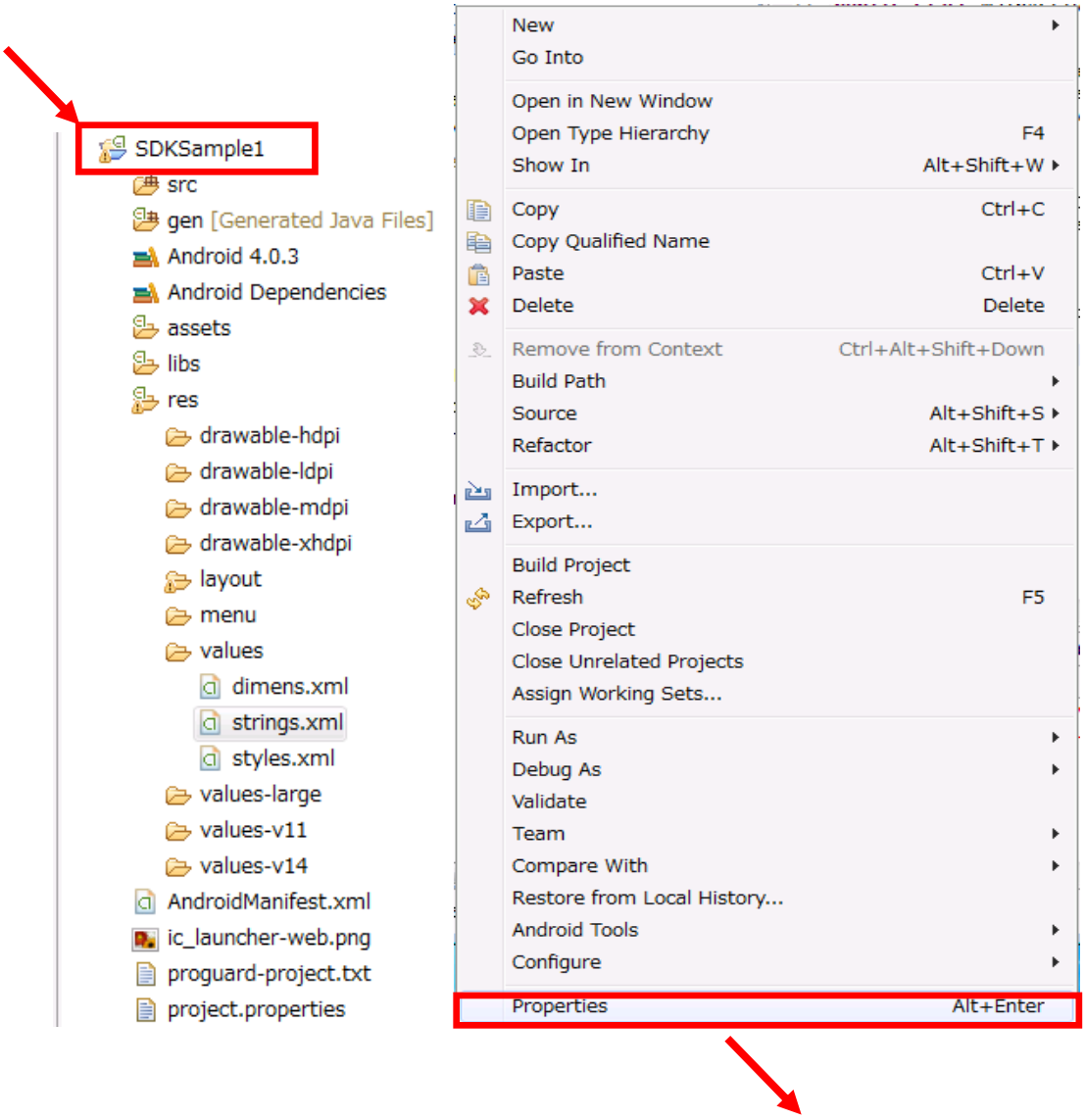


Figure 7.5

2. Set the Android project.

When a new project is created, follow the procedure described in Chapter 5.3Product Overview first. If this process is omitted, an error occurs in a subsequent process.



For the operations after this, refer to the Chapter 5.3Setting Android Project

Figure 7.6

3. Create a screen.

When a project is created, the edit display as shown below is displayed at first. It becomes the screen of the application.
First, delete the text "**Hello world!**" displayed at the center.
Next, drag & drop the grey icon which indicates **Button** from **Form Widgets** in **Palette** on the left into the screen.

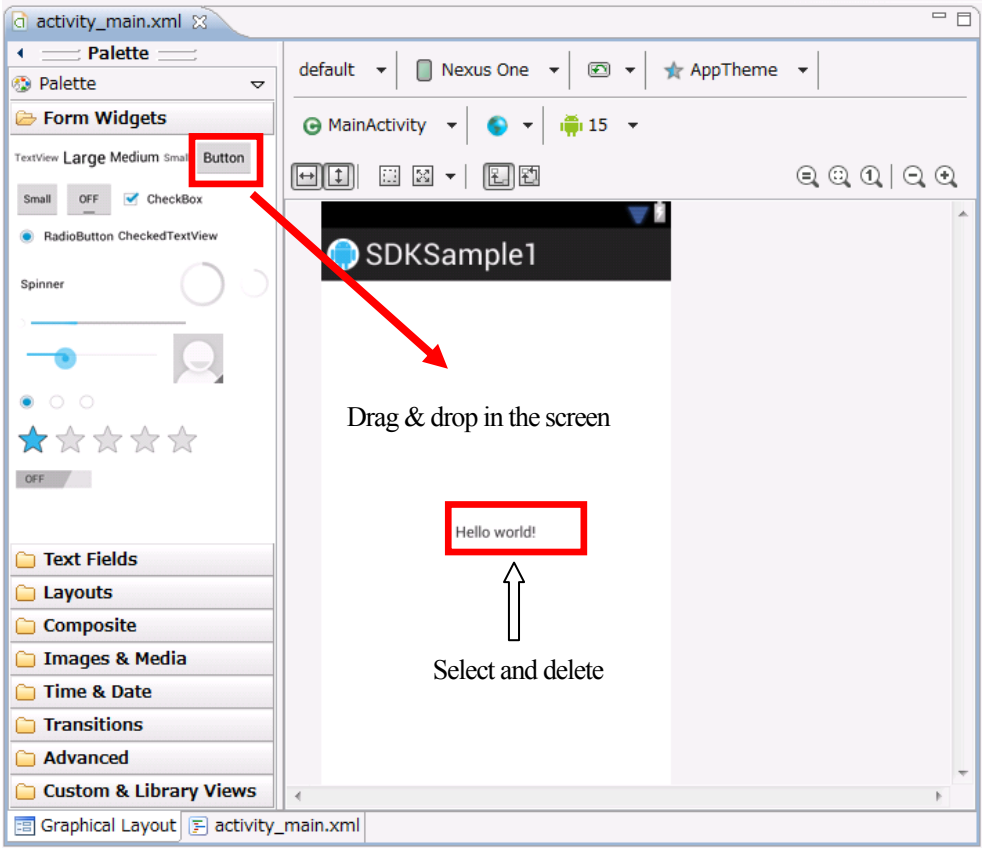


Figure 7.7

Paste the button on the screen, and adjust the size and the position.

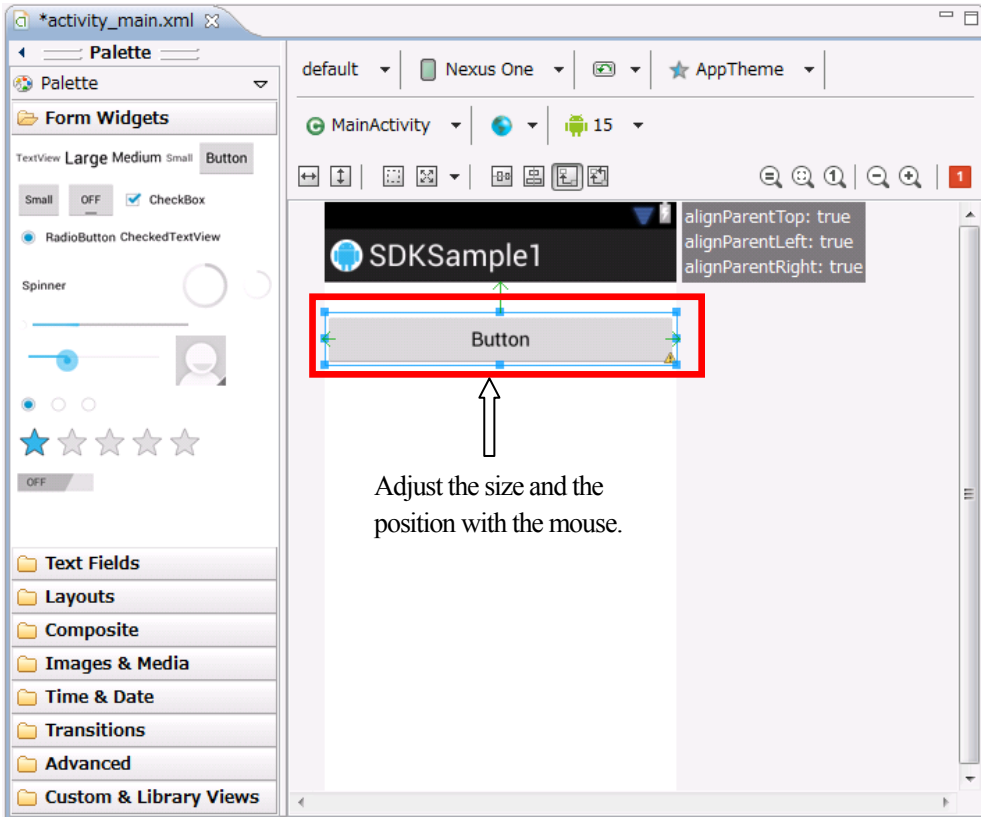


Figure 7.8

Next, create a resource of the character strings.
Open the project of **SDKSample1** from **Package Explorer** on the left side of the **Eclipse** screen.
When you point res → values, the file "**strings.xml**" appears as shown in the figure below, then double-click on it.

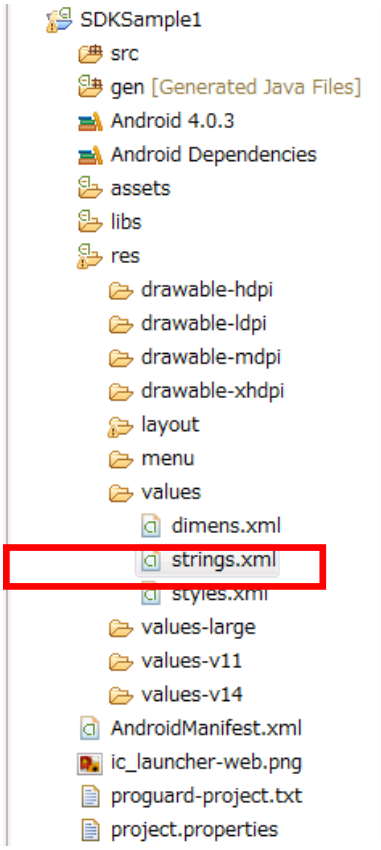


Figure 7.9

The resource editing dialog as shown in the figure below is displayed.
First, select unnecessary **hello_world**, and press the **Remove** button.

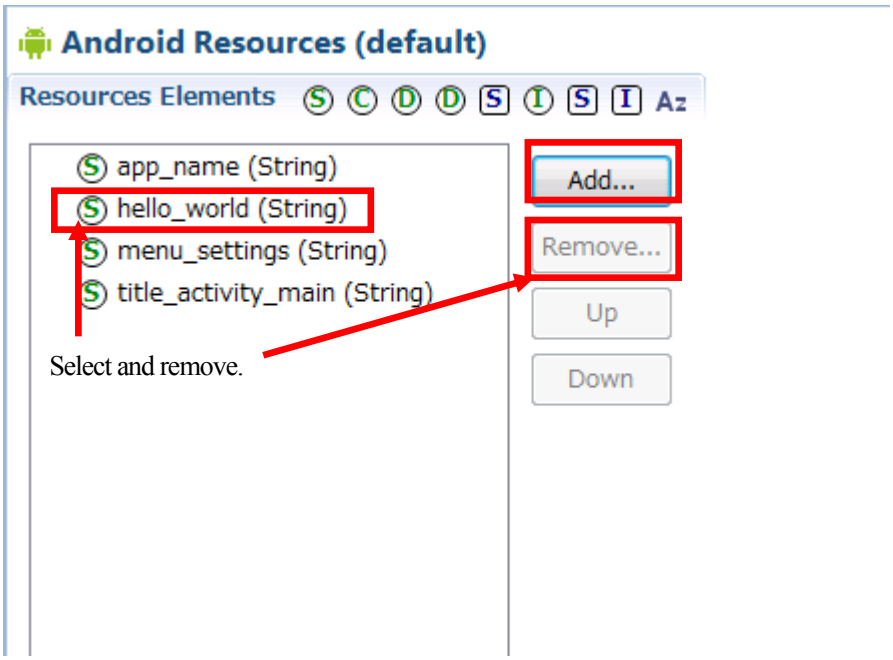


Figure 7.10

Next, press the **Add** button.
The dialog below is displayed. Select **String**.

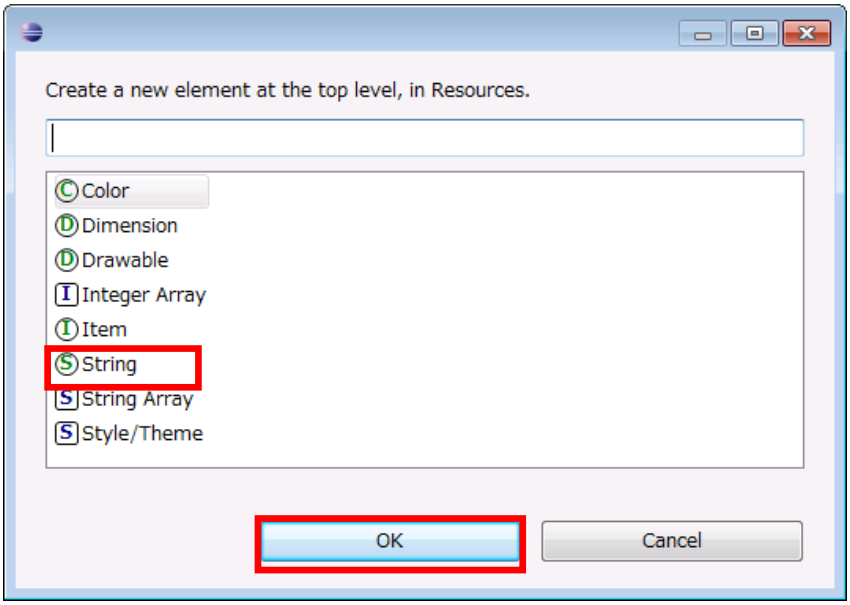


Figure 7.11

Set the name and the value in the new string resource.
The name "**button_title**" and the value "**Get model name!**" are set in this example.

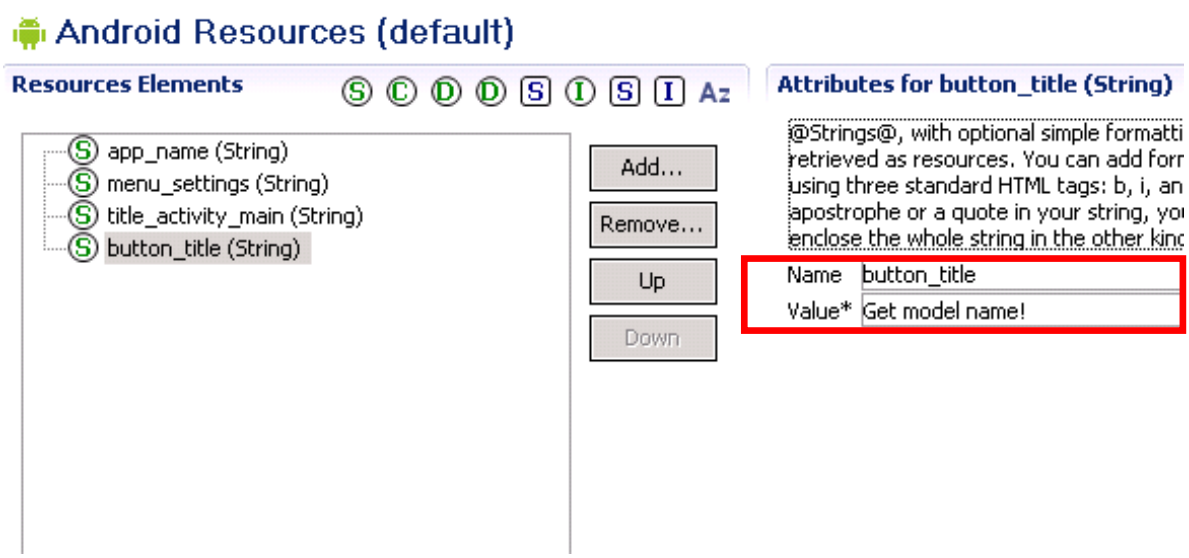


Figure 7.12

When editing is finished, select **Save** from the **File** menu, and save the change.

Change the details of the text displayed on the button in the **Text** in **Properties** at the lower right while selecting the button.
Enter "**@string/button_title**" in this example. When it is entered, the display on the button changes to "**Get model name!**".
It indicates the text resource "**button_title**" that you have just created.

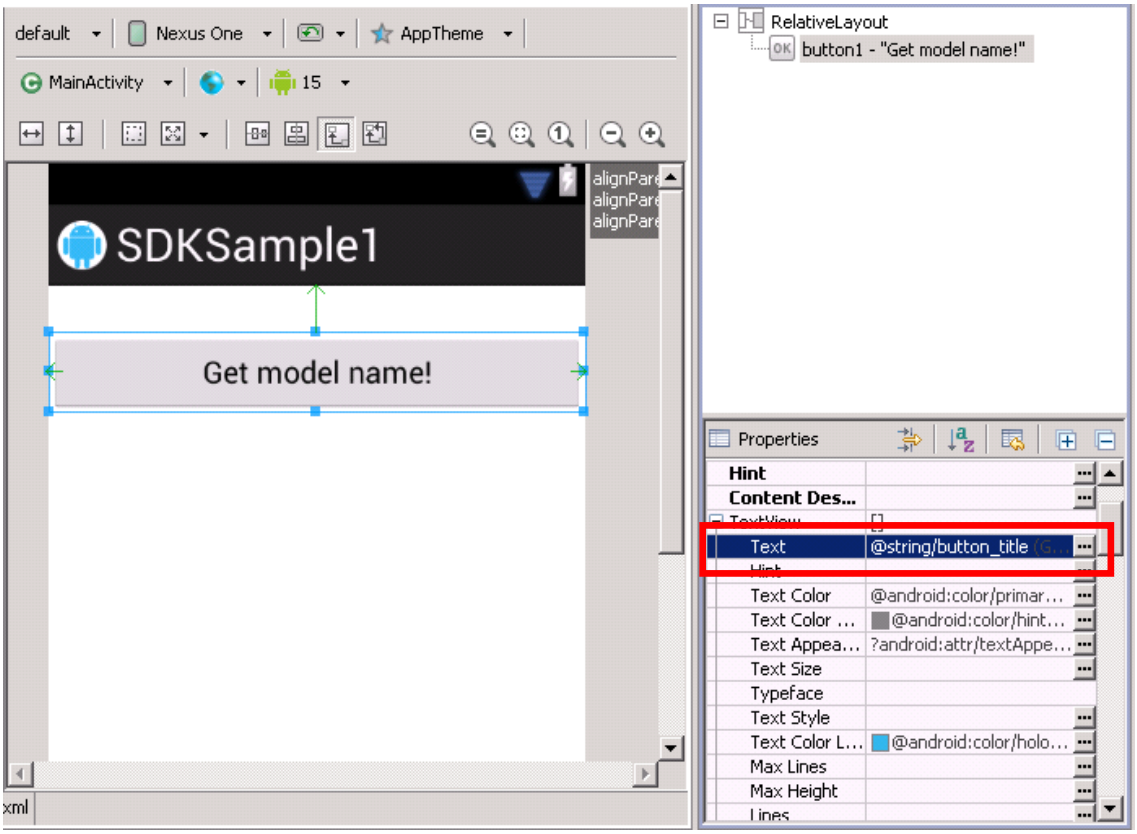


Figure 7.13

Next, enter "**onClick**" in **On Click** in **Properties** in the same way.
It is the name of the method in the activity that is executed when the button is pressed. (It is created later.)

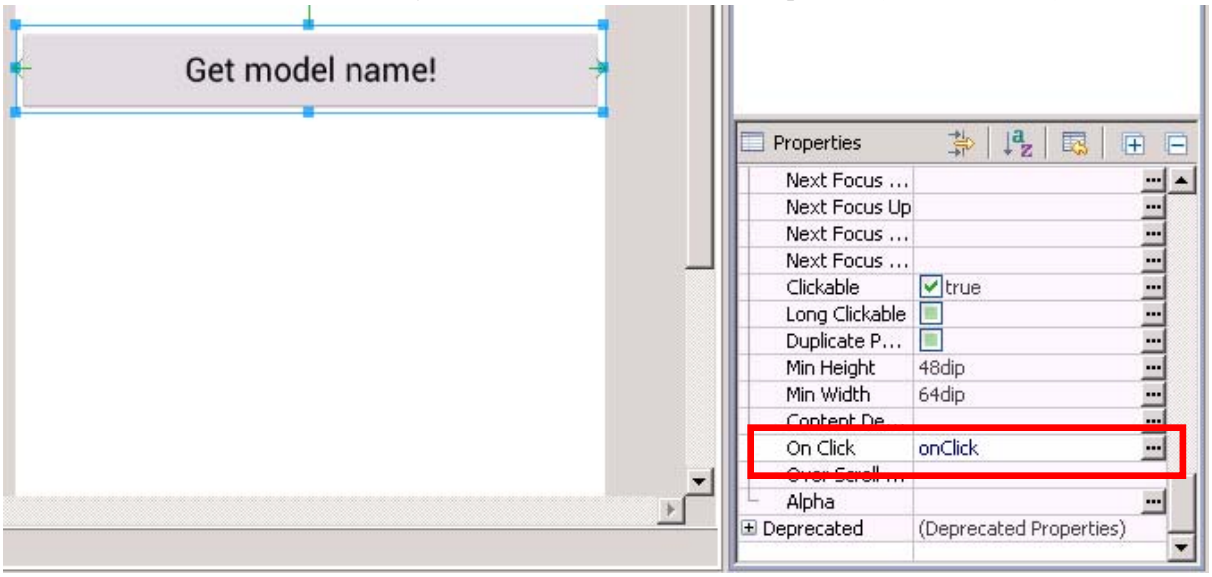


Figure 7.14

4. Create a program.

Next, open the source of the activity.

Open in the order of **SDKSample1** → **src** → **jp.casio.vx.framework.sdksample1** from **Package Explorer**, and

double-click **MainActivity.java**.

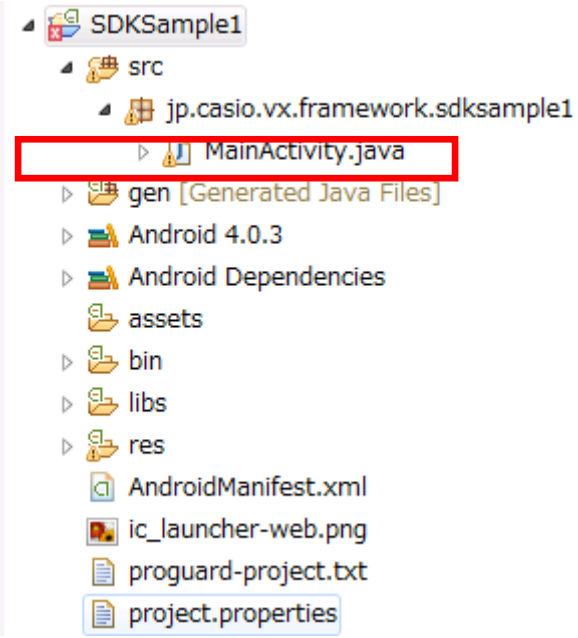


Figure 7.15

Correct **MainActivity.java** as shown below.

(For the specifications of the **System library** currently used, refer to the manual of the library.)

```
package jp.casio.vx.framework.sdksample1;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.view.Menu;
import android.content.Context;
import android.view.View;
import jp.casio.vx.framework.system.Api;

public class MainActivity extends Activity implements
        View.OnClickListener        {

 @Override
 public void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_main);
 }

 @Override
 public boolean onCreateOptionsMenu(Menu menu) {
 getMenuInflater().inflate(R.menu.activity_main, menu);
```

```
    return true;
    }

    public void onClick(View view)
    {
            jp.casio.vx.framework.system.Api sysapi
                                                    = new jp.casio.vx.framework.system.Api(this);
                    //Get model name of terminal
                    String modelName = sysapi.getModelNameString();

                    if(modelName.length() > 0) {
                            showDialog(this,"SDK Sample",
                                                    "The model name of this terminal is "+ modelName);
                    }else {
                            showDialog(this,"SDK Sample","Can't get model name.");
                    }
    }

    private static void showDialog(Context context, String title, String text)
    {
            AlertDialog.Builder ad=new AlertDialog.Builder(context);
            ad.setTitle(title);
            ad.setMessage(text);
            ad.setPositiveButton("OK", null);
            ad.show();
    }
}
```

Save the source by **Save All** of the **File** menu after entering the source.

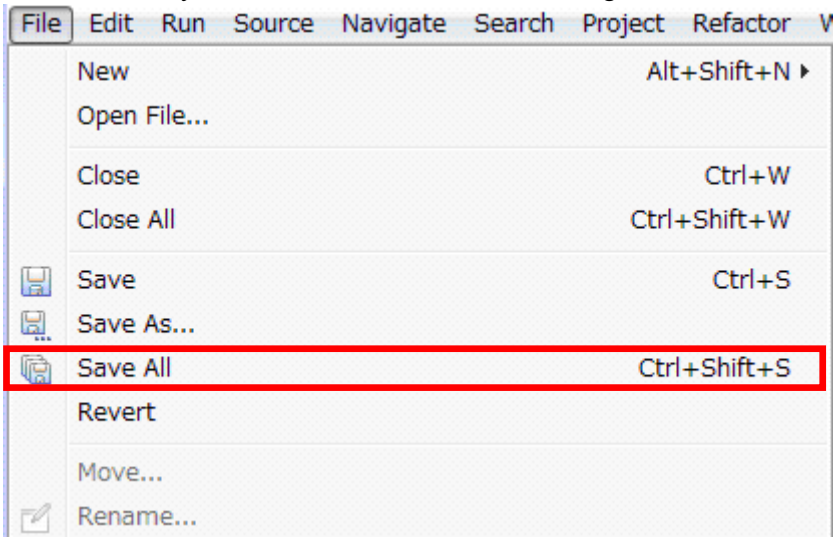


Figure 7.16

After saving the source, execute build. When any error occurs, correct as necessary.

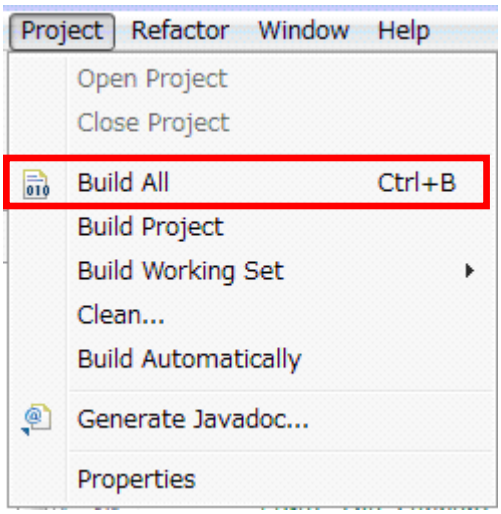


Figure 7.17

After the build, debug as necessary. For the details of debugging, refer to Chapter 5.4Product Overview.
And since the beep of the buzzer is the main function in this sample, it is recommended to debug on the actual terminal of **V-T500/V-N500**.

5. Incorporate the application into the terminal.

Fight-click **SDKSample1** on the **Package Explorer**, and select **Android Tools → Export Signed Application Package**.

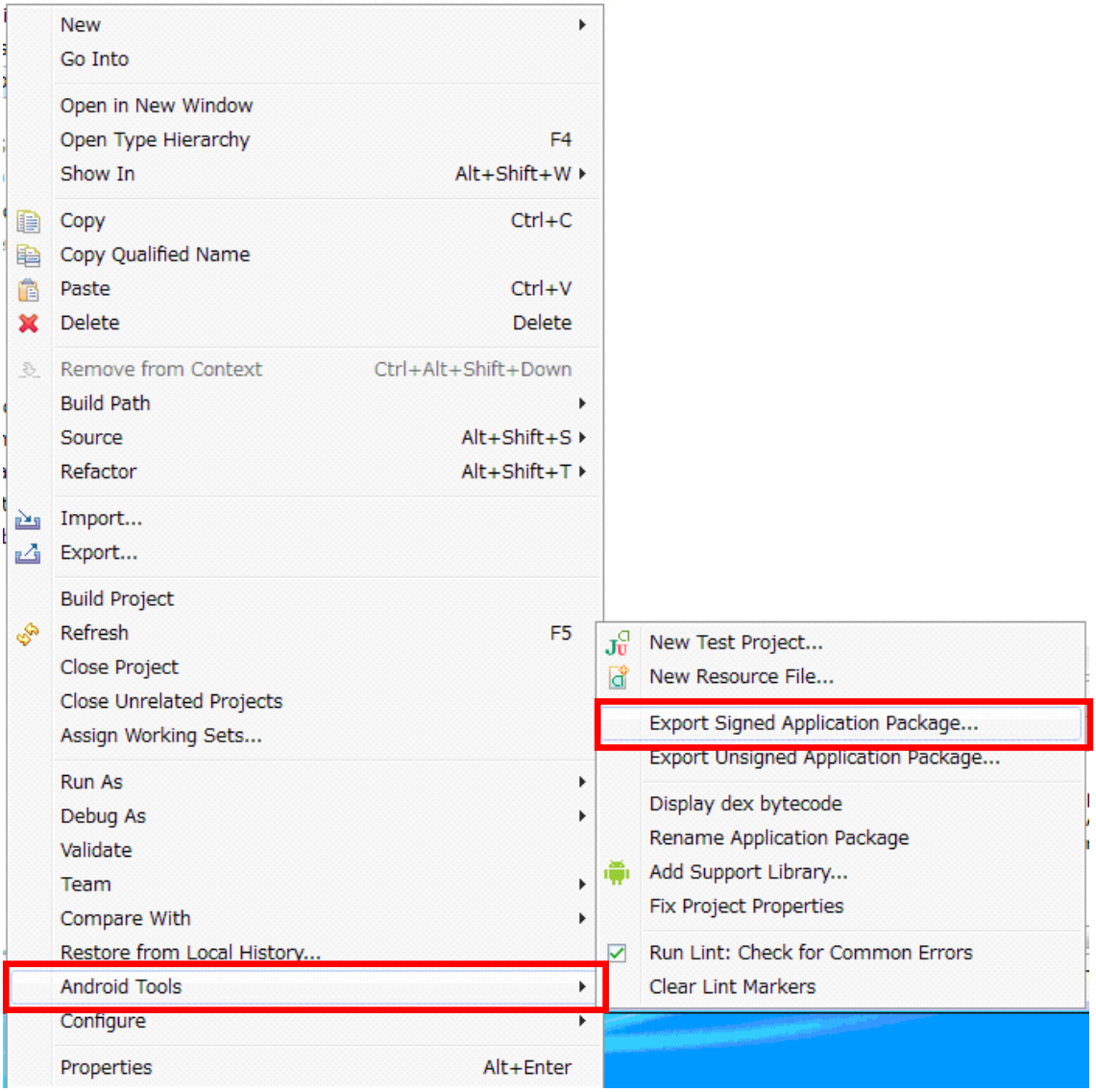


Figure 7.18

The dialog below is displayed. Press **Next >**, and proceed to the next step.

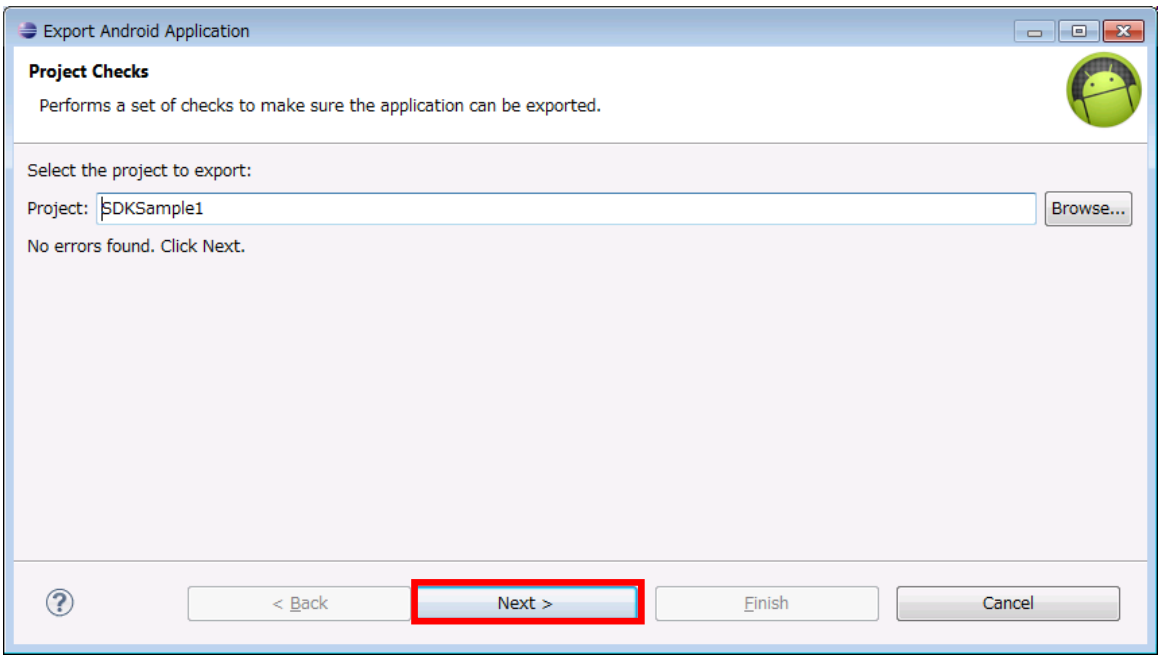


Figure 7.19

The dialog below is displayed.
This is setting of the certificate file to add a signature to Android applications.
The certificate file can be created here or select an existing one.
When you create a new one, set a password of at least 6 letters.

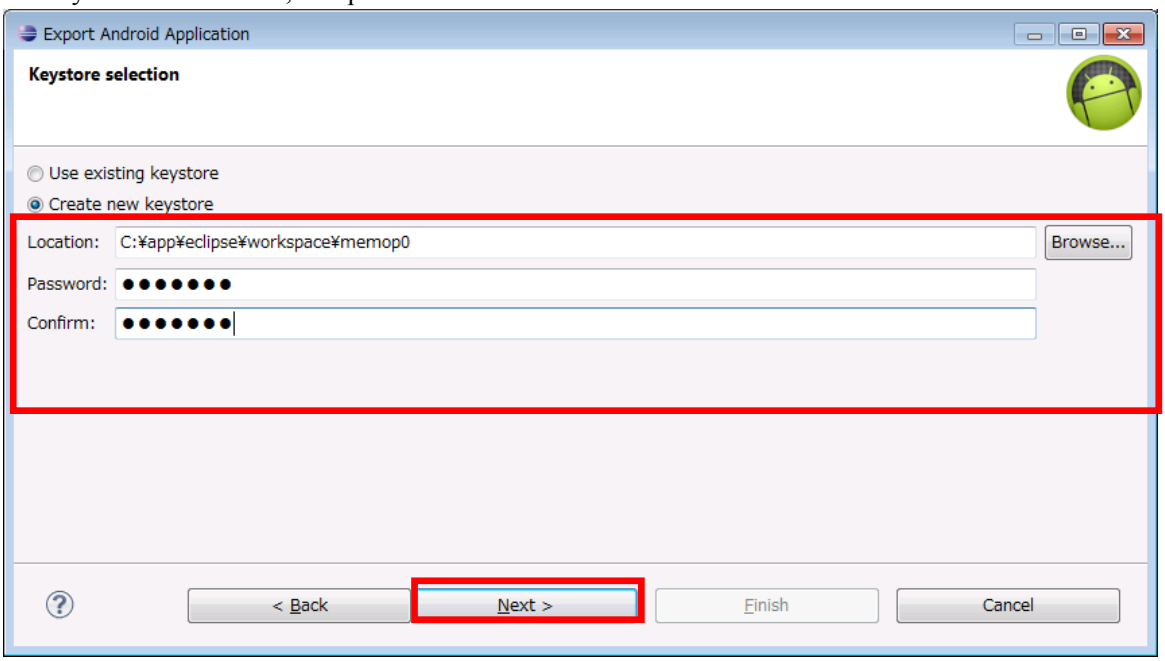


Figure 7.20

When you create a new key, the screen below is displayed.
Input all items.

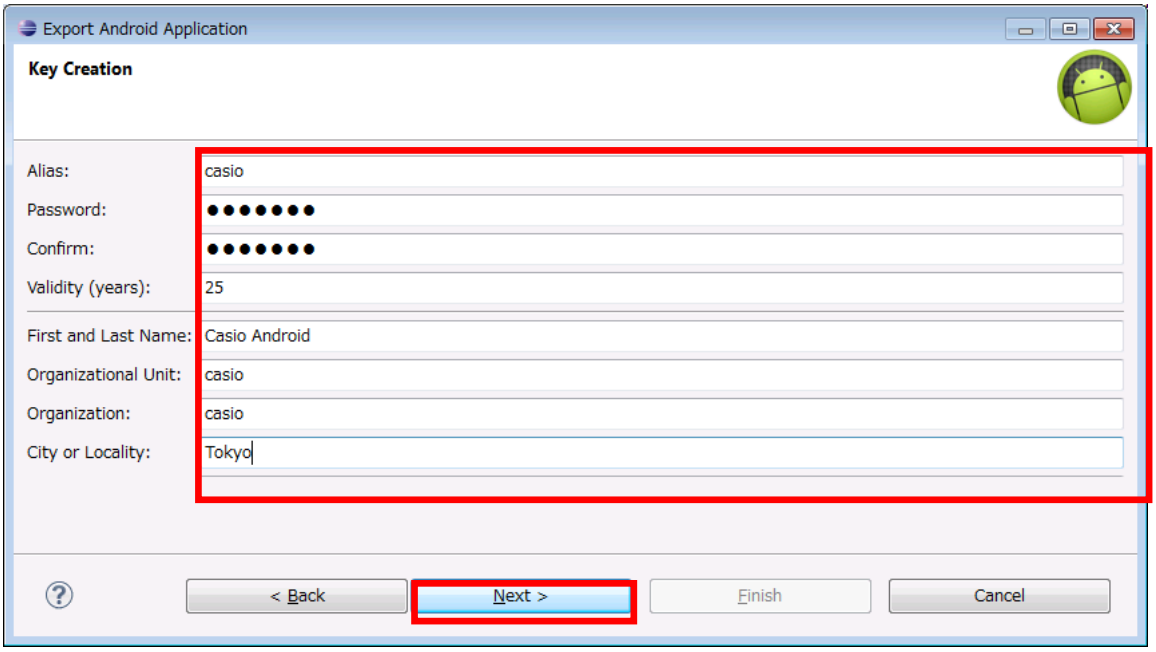


Figure 7.21

Table 7.1

Item	Description
Alias	Alias of the signature.
Password	Password.
Confirm	Enter the password for confirmation.
Validity(years)	Specify the expiration in years. Specify 25 or more.
First and Last Name	The name of the creator.
Organization Unit	The name of the organization unit.
Organization	The name of the organization.
City of Locality	The name of the city.

Then, the dialog below is displayed.
Check the details of the **Destination APK file**, and finish with **Finish** button.

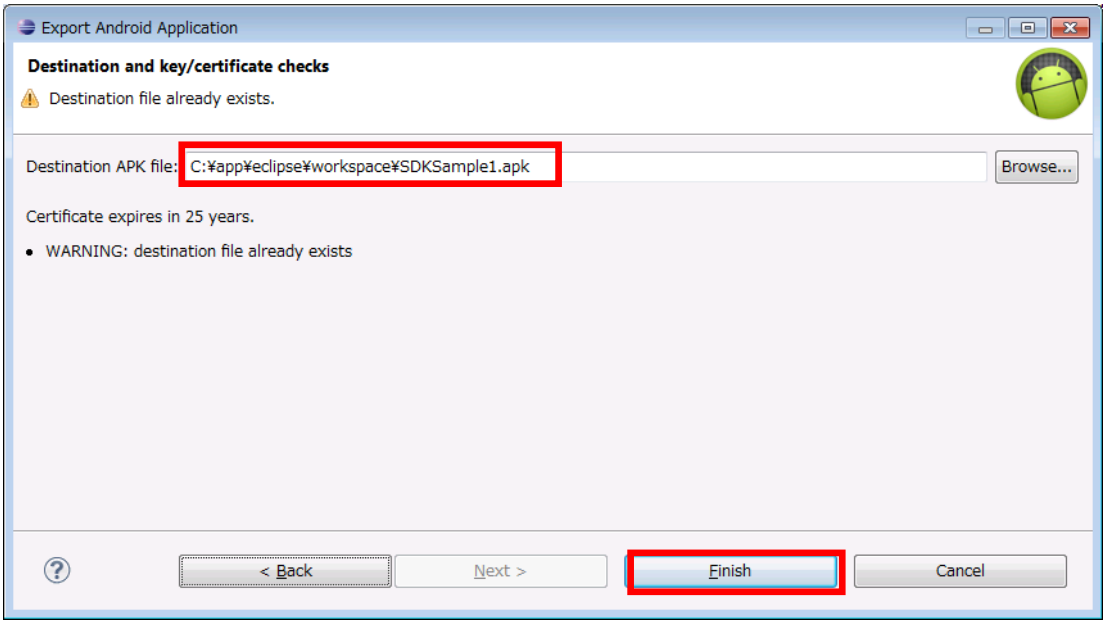


Figure 7.22

The package file (apk file) of the application is created.

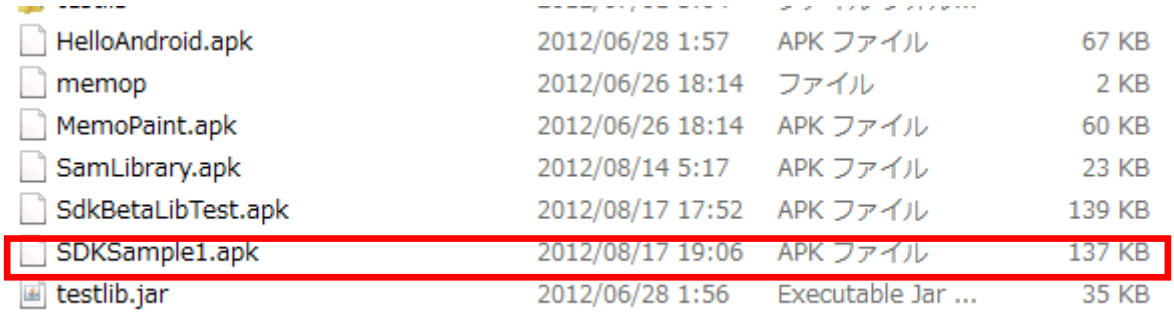


Figure 7.23

There are two methods of installing this file on the actual terminal of **V-T500/V-N500**.

(1) Copy in the SD card, and install on the actual terminal of **V-T500/V-N500**.

Copy the created apk file from the PC to the SC card.

Set the SD card in the SD card slot of the actual terminal of **V-T500/V-N500**, and close the SD card cover tightly.

Next, start up the **File Manager** on the actual terminal of **V-T500/V-N500**.

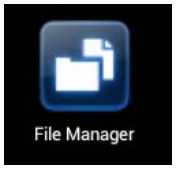


Figure 7.24

Memory card is displayed as shown below, and tap on it.

If **Memory card** does not appear, mount the SD card from **Storage** of the **Setting** tool.

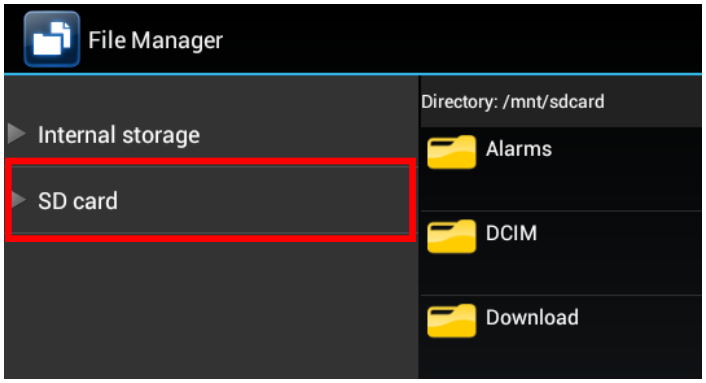


Figure 7.25

The copied apk file is in the memory card. Tap on it, too.

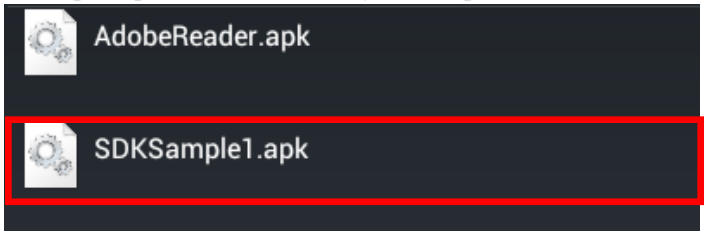


Figure 7.26

The dialog below is displayed. Select **Install** button.

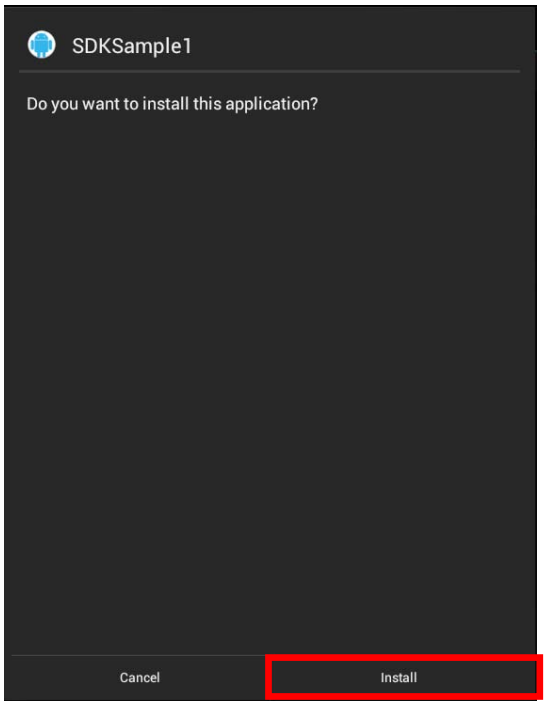


Figure 7.27

Installation of the application is finished.

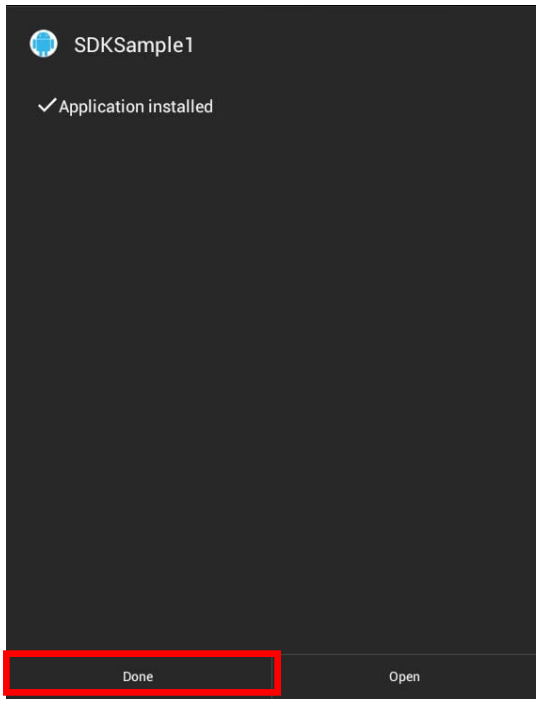


Figure 7.28

After this, the application can be executed on the actual terminal of **V-T500**/**V-N500**.
Execute the application from the menu, and check the operation.

(2) Install from the PC side using the **ADB** command.
First, connect the terminal and the PC using a USB cable. (The mini B USB connector comes to the terminal side.)

For details of connection, refer to Chapter 4Product Overview.

Open the command prompt, and move the current folder to the location where the apk file exist.
The installation command is as shown below.
adb install -r SDKSample1.apk

The result of execution becomes as follows:

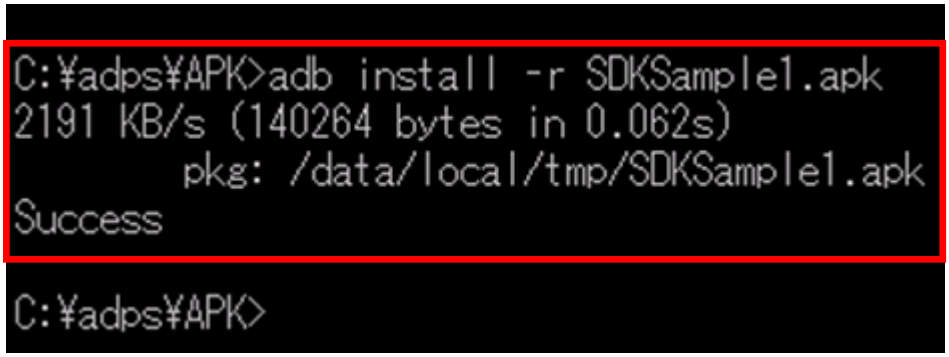


Figure 7.29

After this, the application can be executed on the terminal.
Execute the application from the menu, and check the operation.

8. Kitting

Installation of the application on the terminal, initial settings and update of the OS are described in this section.

8.1 New Configuration of System

The methods of making new settings and installing the application on the terminal are described.

8.1.1 Flow of New Configuration

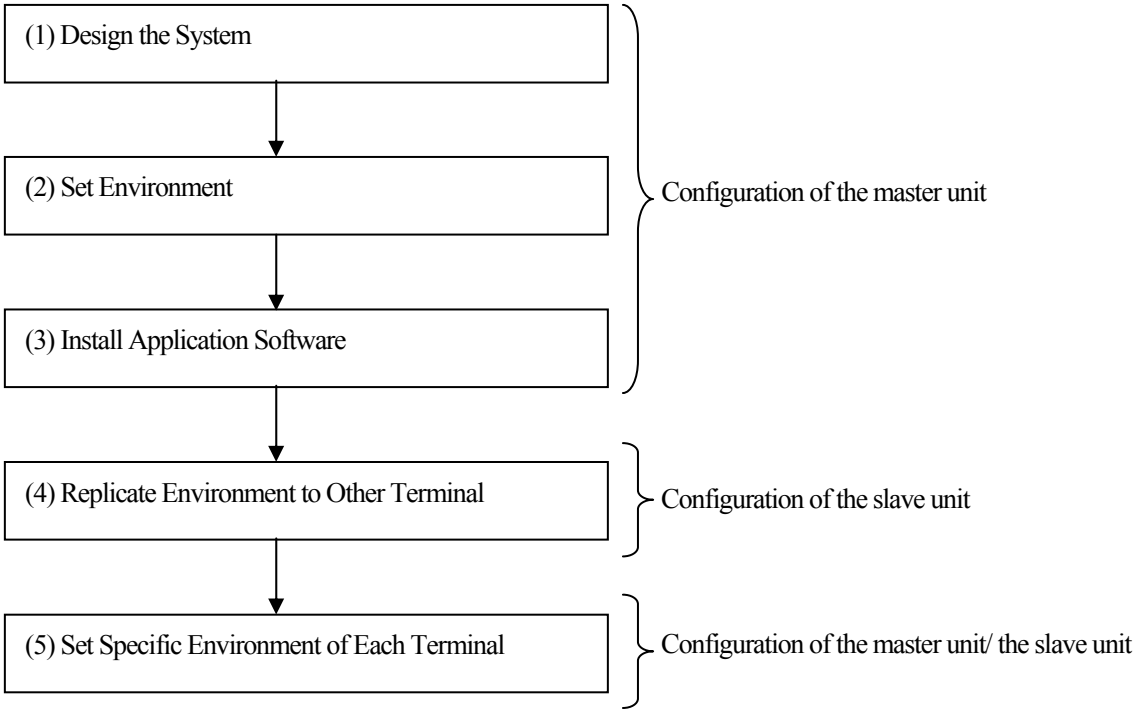


Figure 8.1

8.1.2 Design the system

Depending on the usage of the terminal, design the system to be required and application software to install.
It's recommended to design system with following points of view.

- Design for all terminals
- Design for each terminal

8.1.3 Set Environment

Make environment setting of the terminal.
There settings for the environment setting of the terminal are as follows:
1) Environment setting by the **Settings** tool
2) Setting the security for the administrator
3) Setting the security for general users
4) Setting the **Job Menu**

The settings above are all arbitrary settings.
Make setting as necessary.

1) Environment setting by the **Settings** tool
Basically, they can be used with the preset values.
For the setting of some items such as wireless setting, set according to the usage.
The preset values of the terminal are described in Appendix.

2) Setting the security for the administrator
Set the password for the administrator.
The password for the administrator is used for authentication for using a secure function.
It is also possible to set whether the user data should be deleted when authentication fails.
This setting is made using the **Security Settings** tool.

3) Setting the security for general users
Set the authentication information (user/password) for general users.
Multiple authentication information can be registered.
The authentication information is used for log-in authentication when using a terminal and administration of **Job Menu**.
It is also possible to set whether the user data should be deleted when authentication fails.
This setting is made using the **Account Edit** tool.

4) Setting the **Job Menu**
This is the menu displayed instead of the Android standard home application.
It is possible to configure the menu configuration per user by associating with the general user mentioned above.
Create the definition of the menu using the tool (menu editor) that operates on the PC (Windows).

8.1.4 Install Application Software

This section explains about the method to install the application on the terminal and configure the file used on the application.

1) Registration method

The method of registering the application and the configuration file to the terminal is described below.
■ Registration using the flash memory

2) Registration using the flash memory

■ Facilities required
External storage medium: SD card or USB memory

■ Procedure
(1) Save the application (apk file) and the configuration file in the external storage medium.

(2) Set the external storage medium on the terminal.

(3) Start up the **File Manager**.
If the external storage medium set on the terminal does not appear on the **File Manager**, open the **Storage** menu of the **Settings** tool, and mount the external storage medium.

(4) Install the application (apk file).
Tap the application (apk file) on the file manager, and install according to the displayed messages.

(5) Install the configuration file.
Press and hold the configuration file on the **File Manager**.
When it is held long, the **Edit** menu is displayed.
Select **Copy** from the **Edit** menu.
Move to the folder to install by the file manager.
When it is moved to the folder to install, select the **Paste** menu.

8.1.5 Replicate Environment to Other Terminals

The method of replicating the configured environment on another terminal (slave unit) after configuring a master unit is described in this section.
Note that it is necessary to acquire a license for each copy of software when replicating the software.

1) Replication method

When replicating a configured environment of a terminal to another terminal can be conducted in the following method.
It is possible to register in any method.

■ Replication of the environment using the **Backup/Restore** tool
■ Replication of the environment using the **Copy Devices** tool between master units

2) Details of replication

The details that can be replicated by the methods above are as listed below.

Details that can be replicated

(1) Environment setting set by the **Setting** tool
(2) Security setting for the administrator
(3) Security setting for general users
(4) **Job menu** setting
(5) The installed applications and configuration files (under the /data folder)
Note: It's possible to replicate only by using **Copy Devices** tool .

3) Replication of the environment using Backup/Restore

■ Facilities required
External storage medium: SD card or micro SD card

■ Procedure
(1) Set an external storage medium on the master unit (replication source).
Open the **Storage** menu of the **Settings** tool, and confirm that the external storage medium is mounted.
If the external storage medium is not mounted, mount it.

(2) Start up the **Backup/Restore** tool of the master unit (replication source)
The operation of the **Backup/Restore** tool on the master unit (replication source) is described below.

(3) Select the external storage medium to save the backup data in **Option setting**.

(4) Check **All items** in **Backup**.

(5) Tap **Backup the selected items** in **Backup**.
The backup data is stored in the "**/Backup**" folder in the external storage medium.
It is not necessary to create "**/Backup**" in the external storage medium in advance.
If there is already a "**/Backup**" folder in the external storage medium, confirm that there is no file under the folder before executing this operation.

(6) When backup is completed, finish the **Backup/Restore** tool.

The description above is the operation of the **Backup/Restore** tool on the master unit (replication source).

(7) Remove the external storage medium from the master unit (replication source).
Open the **Storage** menu of the **Settings** tool, and implement unmount of the external storage medium.
If the external storage medium is removed from the terminal without performing unmount, the external storage medium may be damaged. Make sure to implement unmount.
Remove the external storage medium from the master unit (replication source).

(8) Initialize the slave unit (replication destination).
Delete the applications and setting files installed on the slave unit (replication destination), and return the storage to the initial state.
Open the **Backup & reset** menu of the **Settings** tool, and execute **Factory data reset**..

(9) Set an external storage medium on the slave unit (replication destination).
Open the **Storage** menu of the **Settings** tool, and confirm that the external storage medium is mounted.
If the external storage medium is not mounted, mount it.

(10) Start up the **Backup/Restore** tool of the slave unit (replication destination).

The operation of the Backup/Restore tool on the slave unit (replication destination) is described below.

(11) In **Option setting** menu of **Backup/Restore** tool , select the external storage medium where the backup data is stored.

(12) Check **Check All** in **Restore** menu.

(13) Tap **Restore selected items** button in **Restore** menu.

(14) When restore is completed, finish the **Backup/Restore** tool.

The description above is the operation of the **Backup/Restore** tool on the slave unit (replication destination).

(15) Remove the storage medium from the slave unit (replication destination)
Open the **Storage** menu of the **Settings** tool, and implement unmount of the external storage medium.

If the external storage medium is removed from the terminal without performing unmount, the external storage medium may be damaged. Make sure to implement unmount.
Remove the external storage medium from the slave unit (replication destination).

(16) Restart the slave unit (replication destination).
Press and hold the power supply button long, and shut down the slave unit (replication destination).
Press and hold the power supply button, and start up the slave unit (replication destination).

The replication of the environment on another terminal is completed.

4) Replication of the environment using the Copy Devices tool between master units

■ Facilities required
External storage medium: SD card or micro SD card

The external storage medium above is used for saving the backup data of the master unit (replication source).
If the backup data is stored in the built-in memory internal storage) of the master unit (replication source), the external storage medium is not necessary.

The built-in memory and the external storage medium are called "**storage**" in the following explanation.
If each of the built-in memory and the external storage medium should be mentioned clearly, it is described as is.

The destination to transfer the backup data of the slave unit (replication destination) is the same storage as the master unit (replication source).
If an SD card or a micro SD card is used as storage for the backup data, it is necessary to set (mounted) on the terminal in advance on both the master unit (replication source) and the slave unit (replication destination).

■ Procedure
(1) Start up the **Backup/Restore** tool of the master unit (replication source).

The operation of the **Backup/Restore** tool on the master unit (replication source) is described below.

(2) Select the storage to save the backup data in **Option setting**.

(3) Check **All items** in **Backup**.

(4) Tap **Backup selected items** in **Backup** menu.
The backup data is stored in the "**/Backup**" folder in the storage.
It is not necessary to create "**/Backup**" in the storage in advance.
If there is already a "**/Backup**" folder in the storage, the backup data is deleted.

(5) When backup is completed, finish the **Backup/Restore** tool.

The description above is the operation of the **Backup/Restore** tool on the master unit (replication source).

(6) Start up the **Settings** tool on the master unit (replication source), and enable WiFi-Direct in the **More…** menu.

(7) Start up the **Copy Devices** tool on the master units (replication source).

(8) Select **2. Sender mode** of the **Copy Devices** tool on the master units (replication source).

(9) Start up the **Settings** tool on the slave unit (replication destination), and enable WiFi-Direct in the **More…** menu.

(10) Start up the **Copy Devices** tool on the slave unit (replication destination).

(11) Select **1. Receiver mode** of the **Copy Devices** tool on the slave unit (replication destination).

Repeat Steps (9), (10) and (11) for all slave units (replication destinations).

(12) Select **Start** of the **Copy Devices** tool on the slave unit (replication destination).
In the first attempt, make sure to select **Start** of the **Copy Devices** tool on the slave unit (replication destination) before selecting **Start** of the **Copy Devices** tool on the master unit (replication source).
When **Start** is selected and replication starts, the background color of the **Copy Devices** tool on slave units (replication destination) becomes green.

(13) Select **Start** of the **Copy Devices** tool on the master unit (replication source).
When **Start** is selected and replication starts, the background color of the **Copy Devices** tool on the master units (replication source) becomes blue.

Repeat Step (12) for all slave units (replication destinations).
It's possible to add slave unit (replication destination) during replication.

(14) Execute restore of the backup data on the slave unit (replication destination), and restart it.
When it's completed to transfer the backup data to the slave unit (replication destination), the background color of the **Copy Devices** tool on the slave unit (replication destination) becomes orange.
Then, the dialog to inform the completion of receipt and ask to execute restore and reset is appeared.
Select **Yes** in this dialog.

(15) Finish the **Copy Devices** tool on the master units of the master unit (replication source).
When replication of the environment on the prepared slave units (replication destinations) is completed, select **Finish** in the **Copy Devices** tool on the master unit, and finish the **Copy Devices** tool on the master unit.

The replication of the environment to other terminals is completed.

■ Precautions
If there is any other device that is the group owner of WiFi-Direct nearby, the slave unit (replication destination) recognizes it as the master unit (replication source), and does not operate correctly.
When using the **Copy Devices** tool, confirm there is no such device nearby before use.

8.1.6 Set Specific Environment of Each Terminal

When configuration of the terminal common environment is finished, perform each terminal-specific environment.

The environments that may be set separately on each terminal are as follows:
There are some environments that have to be set additionally according to the operation and the usage other than the following:

■ Mail setting
■ WAN setting
■ WLAN (in case of fixed IP)
■ Security setting for the administrator
■ Security setting for general users

9. Appendix Default Values of the Setting

The initial values of the setting are described below.

■ WIRELESS & NETWORKS

The connection to the network and the devices using Wi-Fi, Bluetooth, mobile, network and USB connection is set and managed.

It is also possible to set the connection between the tablet and a virtual private network (VPN), connect to the Internet with another device via the data communication of the tablet, and turn off all wireless communication by switching to in airplane mode.

Table 9.1

Item		Default Value
Wi-Fi		OFF
Bluetooth		OFF
Data usage		ON
More…	Airplane mode	OFF
	VPN	-
	Tethering & portable hotspot	-
	NFC	ON
	Android beam	OFF
	Wi-Fi Direct	OFF
	Wi-Fi Direct (Set up peer-to-peer connectivity)	-
	Mobile networks	-

■ DEVICE

It is possible to set the sound and the display, and check the state of the storage, the battery and the applications.

The grayed-out items are used for confirmation of the state, and they are not the setting items.

Table 9.2

Item		Default Values
Sound	Volumes	Prescribed value
	Default notification	Pixie Dust
	Vibrate and ring	OFF
	Touch sounds	ON
	Screen lock sound	ON
	Vibrate on touch	ON
Display	Brightness	Prescribed value
	Wallpaper	Prescribed value
	Auto-rotate screen	ON
	Sleep	After 1 minutes of inactivity
	Font size	Normal
Storage		
Battery		
Apps		

■ PERSONAL

The settings related to usage of users and how to use such as various security settings are performed.
The grayed-out items are used for confirmation of the state, and they are not the setting items.

Table 9.3

Item		Default Value
Accounts & sync		ON
Locatioin services	Google's location service	OFF
	GPS satelites	ON
Security	Screen lock	Slide
	Owner info	-
	Encrypt tablet	-
	Set up SIM card lock	Lock SIM card
	Make passwords visible	ON
	Device administrators	
	Unknown sources	OFF
	Trusted credentials	
	Install from SD card	
	Clear credentials	
Language and input	Language	Japanese (Japanese model) English (Non Japanese model)
	Spelling correctioin	ON
	Personal dictionary	-
	Android keyboard	Japanese IME (Japanese model) Android keyboard (Non Japanese model)
	Text-to-speech output	Pico TTS (ON)
	Pointer speed	Prescribed value
Backup and reset	Initialization of the data	

■ SYSTEM

The settings of date and time and the settings for developers are performed.

The grayed-out items are used for confirmation of the state, and they are not the setting items.

Table 9.4

Item		Default Value
Date & time	Automatic date & time	ON
	Automatic time zone	ON
	Set date	Impossible to change
	Set time	Impossible to change
	Select time zone	Impossible to change
	Use 24-hour format	ON
	Select date format	Regional
Accessibility	Large text	OFF
	Auto-rotate screen	ON
	Speak password	OFF
	Touch & hold delay	Short
	Install web scripts	Not allowed
Developer options	USB debugging	ON
	Development device ID	
	Stay awake	OFF
	Allow mock locations	OFF
	HDCP checking	-
	Desktop backup password	OFF
	Strict mode enabled	OFF
	Pointer location	OFF
	Show touches	OFF
	Show screen updates	OFF
	Show CPU usage	OFF
	Force GPU rendering	OFF
	Window animation scale	Animation scale 1x
	Transition animation	Animation scale 1x
	Don't keep activities	OFF
	Background process limit	Standard limit
	Show all ANRs	OFF