POS SDK For Android
User's Manual_English

Shandong New Beiyang Information Technology Co., Ltd.

Contents

About this Manual

Aim of the Manual

This manual aims to introduce to customers how to use the sample program and APIs of POS SDK For Android.

Manual Contents

This manual is made up of the following sections:

1. Overview

This chapter describes searching and connecting to printer of WIFI, searching and connecting to printer of bluetooth, searcheing and connecting to printer of Ethernet, connecting to printer of USB and serial port, APIs founction, operating environment, files in the package and version record.

1.1 Function

● **WIFI/Bluetooth/USB/Serial Port/Ethernet**

The functions as follows:
- Seaching WIFI(USB and serial port does not have this function)
- Seaching Ethernet
- Connecting to printer
- Change printer port Settings
- Sending and writing datas
- Setting timeout,
- Record communication data
- Close port

● **API**

- Setting system function(Set communication module instance of WIFIPort, Initialize printer, Select print mode, Select paper type, Set the horizontal and vertical motion units, Query Status, Feed line, Cut paper, Download file, Open cashdrawer).Reset,print self-check page,change print IP address, change printer subnet mask, change printer gateway, change printer Dns, change printer timeout function.
- Text printing(Select an international character set and Code page, Set line height, Set character spacing, Set alignment mode, Select font type, White/Black reverse, Bold, Underline, Roration, Font Magnify, Bi-colour print, User defined character printing, Text raster printing).
- Image printing(8/24-dot single/double-density, Download image to RAM and print, Download image to Flash and print, Print raster image).
- Barcode 1D printing(UPC-A, UPC-E, EAN-8, EAN-13, Code39, Code93, ITF, Codabar, Code128).

- Barcode 2D printing(PDF417, QR, Maxicode, GS1 DataBar and GS1 composite barcode).
- Setting standard mode paramter(Set left margin , print area width , horizontal Starting Position).
- Setting page mode paramter(Set print area, Print direction, horizontal/vertical Starting Position, printing of page mode, clear buffer).

1.2 Operating Environment

● **Android Version**

Android Ver.4.0 or later

● **Android Device**

Android phone
Android tablets
Android development board

● **Printer**

POS series printers of SNBC

● **Interface**

WIFI, Bluetooth, USB, Serial Port, Ethernet

● **Development Environment**

JDK：Ver.1.6 or later
Android Studio：Ver.2.3.2 or later
Android SDK ：Ver.1.6 or later
ADT: Ver.20.0.3 or later

1.3 Contents in the Package

Files	Description
Two development kit	POSSDKForAndroid.jar libserial_port.so、 android.hardware.usb.host.xml
Project of sample program	POSSDKForAndroid.apk、

	POSSDKForAndroidDemo
User's Manual	POS SDK For Android User's Manual_Chinese.pdf POS SDK For Android User's Manual_English.pdf

1.4 Version Record

Version	Date	Description
V1.00	26/11/2013	Initial draft
V2.00	05/14/2014	Support ADT22.3
V2.02	12/12/2015	Software optimization
V2.03	1/26/2016	Interface optimization; Help document modification
V2.04	11/16/2016	Repair USB Port Error
V2.0.5.1	10/10/2018	Add imageStandardModeGrayPrint()
V2.0.5.2	01/05/2019	Add Ethernet connection support
V2.0.5.3	31/07/2019	Add network port configuration printer function
V2.0.6.1	03/09/2019	Change the data read and write function of the network port
V2.0.6.2	21/12/2019	Add imageCompressedRasterPrint()
V2.0.6.3	03/09/2020	Optimize USB broadcast

2. Sample Program

This chapter aims to describe how to use the sample program.

The sample program has the following functionality:

- Searching for printers of WIFI
- Searching for printers of Ethernet
- Set the IP address for the printer
- Set the gateway address for the printer
- Set the subnet for the printer
- Set the DNS for the printer
- Set the timeout for the printer
- Searching for printers of Bluetooth
- Input IP address
- Input the serial port and baud rate
- Open port
- Close port
- Select print mode
- Text printing
- Image download for RAM/Flash printing
- Barcode printing
- Barcode PDF417 printing
- Barcode QR printing
- Barcode GS1 Databar printing
- Text raster printing
- User defined character printing
- Printer status acquisition
- Feed line
- Cut paper
- Reset
- Print self check page

2.1 Start up the Sample Program

1) Extract the sample program zip file to a directory of your choosing.

2)Run Android Studio and select "Open an existing Android Studio project", as Fig.1
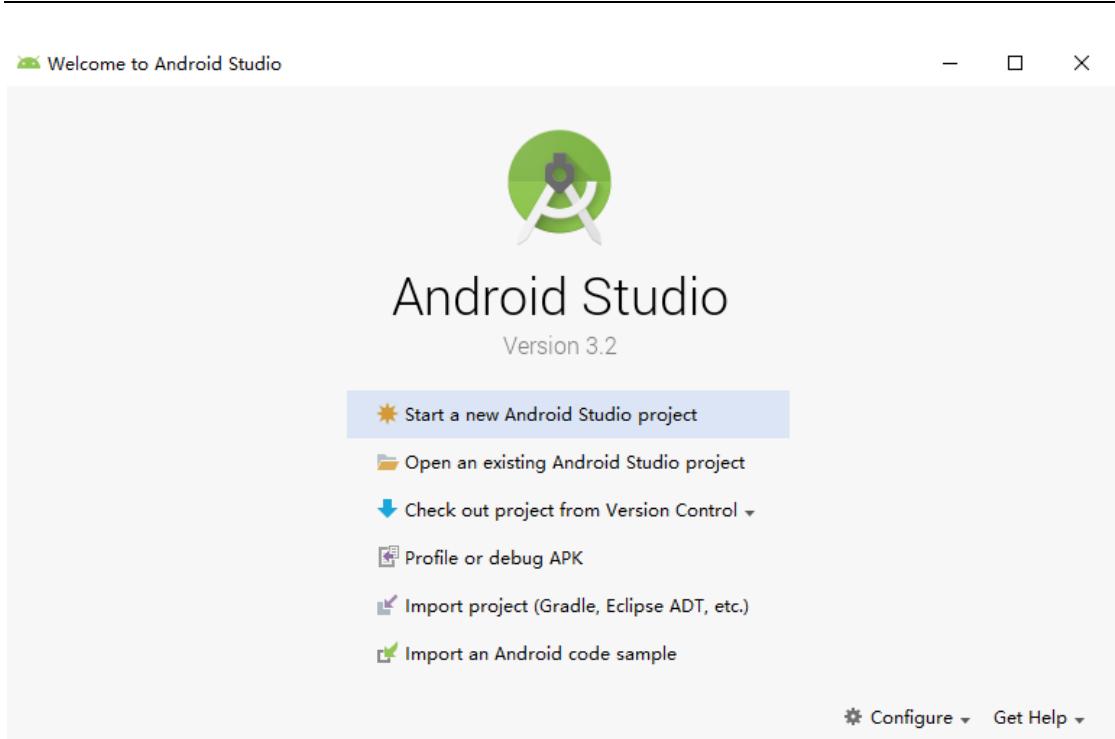
Figure-1

3) Slect the sample program POSSDKForAndroidDemo, as Fig.2 and Fig3
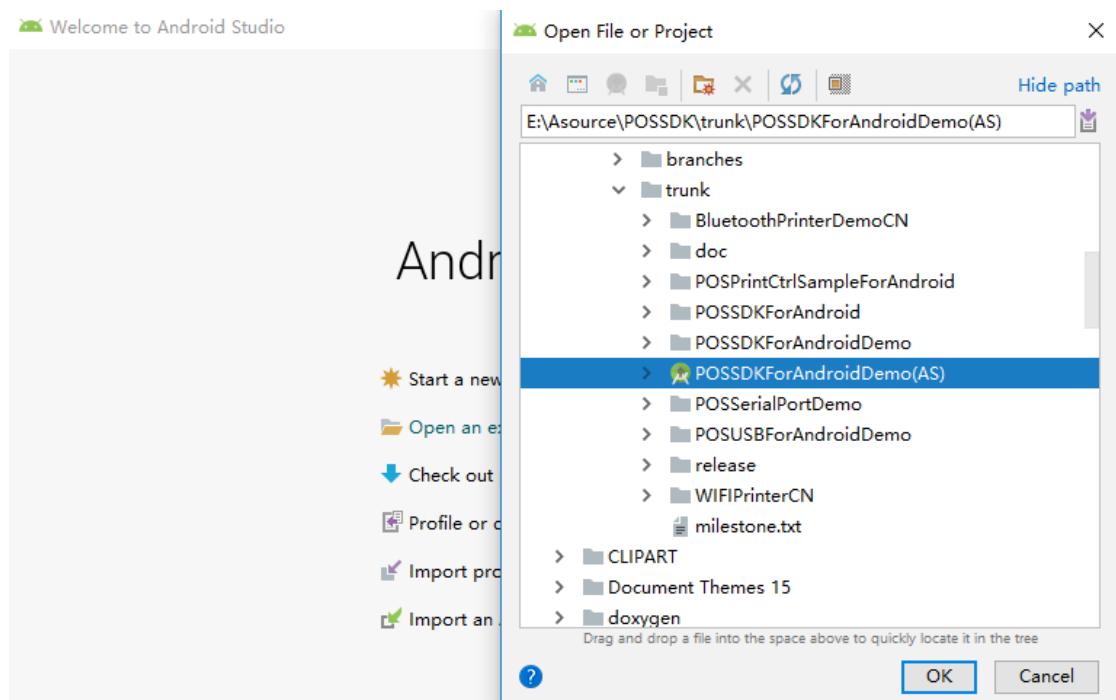
Figure-2
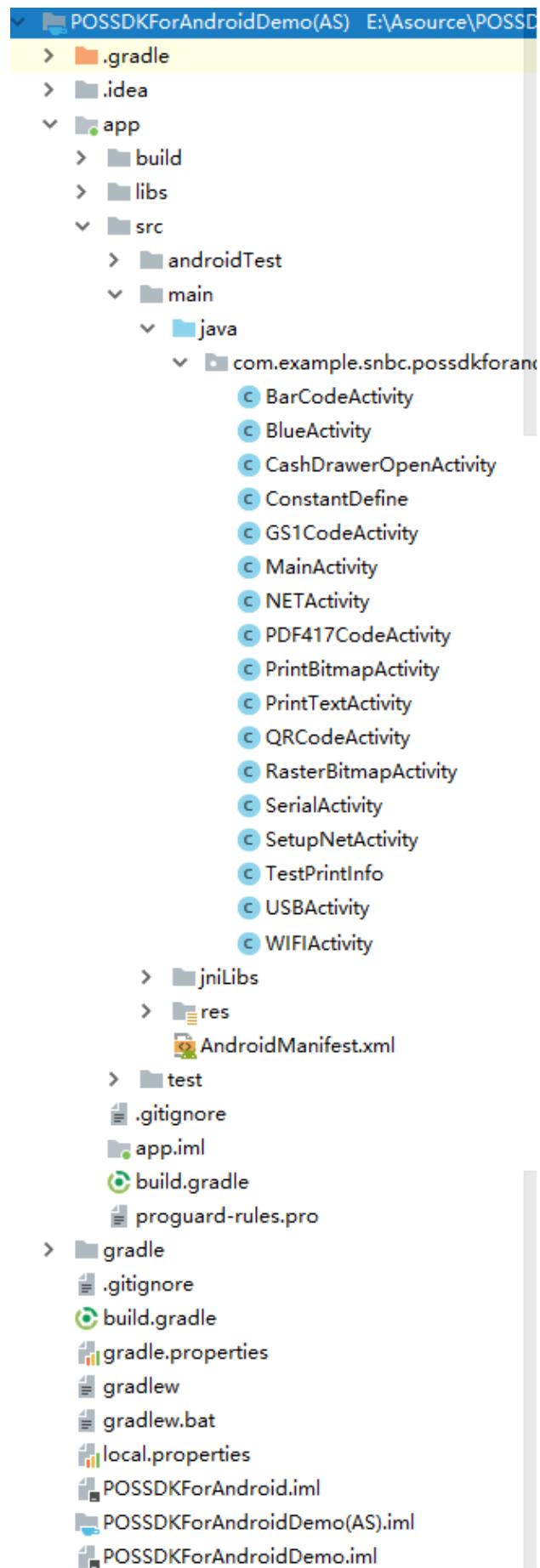
Figure-3

4) Copy "android.hardware.usb.host.xml" in package to the file "/system/etc/permissions" of Android device. Set the permissions of USB HOST so that Android device can control printers by USB.

5) Copy POSSDKForAndroid.apk in package to Android device, of course executable sample program can generate by installed directly.

Remarks:

Windows and Android can transfer files by the adb command. Such as:

File import command format：adb push   the path and file name of the file in Windows system   the path of the file need to be stored in the Android system;

File export command format：adb pull the path and file name of the file in Android system the path of the file need to be stored in the Windows system.

2.2 How to Use the Sample Program

● **Main Screen**

The main screen has the following functions:
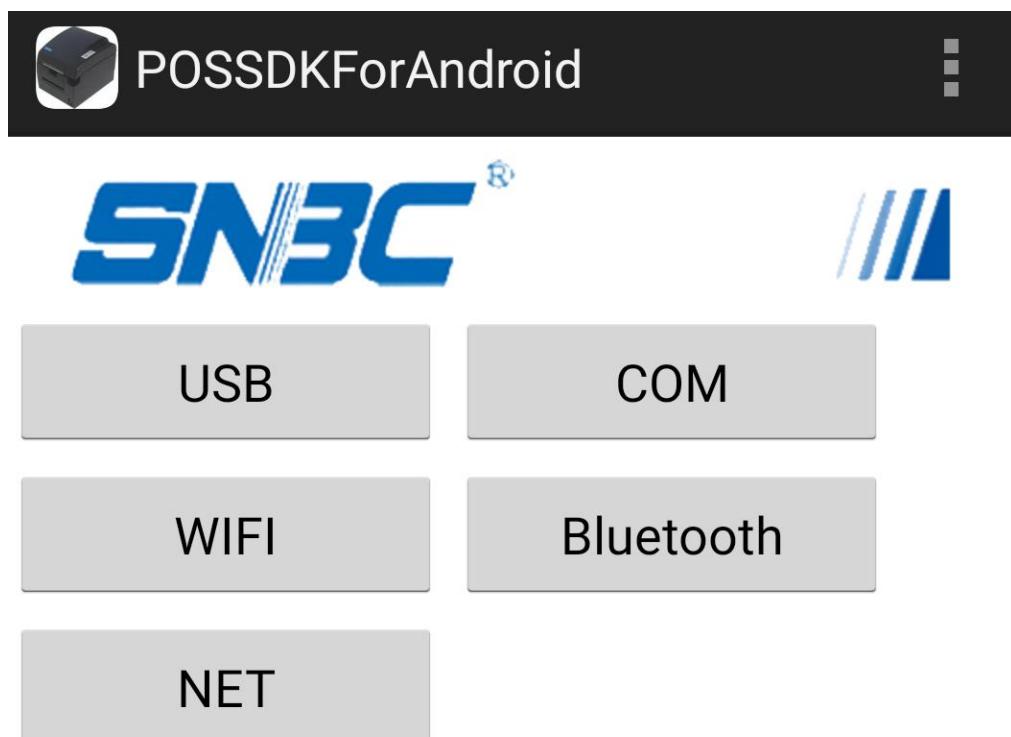
Execute the following processes:

Controller	Description
"USB"	Tap [USB] on the main screen. For details, refer to USB Communication.
"COM"	Tap [COM] on the main screen. For details, refer to COM Communication.
"WIFI"	Tap [WIFI] on the main screen. For details, refer to WIFI Communication.
"Bluetooth"	Tap [Bluetooth] on the main screen. For details, refer to Bluetooth Communication.
"NET"	Tap [NET] on the main screen. For details, erfer to NET Communication

● USB Communication
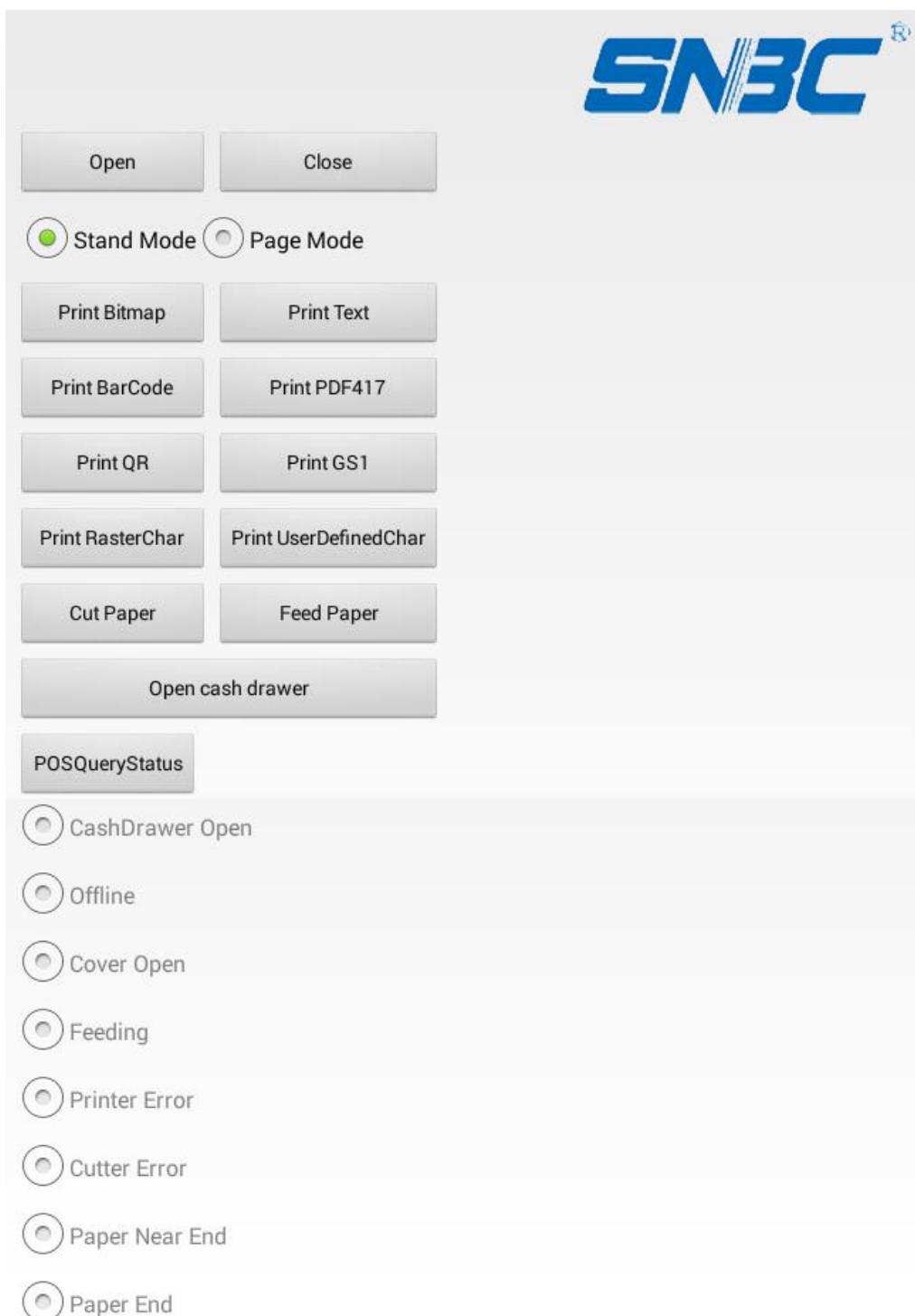
Controller	Description
"Open"	Tap [Open] to connect to printer.
"Close"	Tap [Close] to close port.

"Stand Mode"and"Page Mode"	Select print mode. Tap [Stand Mode] to enter standard mode. Tap [Page Mode] to enter page mode.
"Print Bitmap"	Tap [Print Bitmap], for details, refer to Image printing.
"Print Text"	Tap [Print Text], for details, refer to Text printing.
"Print BarCode"	Tap [Print BarCode], for details, refer to Barcode printing.
"Print PDF417"	Tap [Print PDF417], for details, refer to Barcode PDF417 printing.
"Print QR"	Tap [Print QR], for details, refer to Barcode QR printing.
"Print GS1"	Tap [Print GS1], for details, refer to Barcode GS1 Databar printing.
"Print RasterChar"	Tap [Print RasterChar], for details, refer to Text raster printing.
"Print UserDefinedChar"	Tap [Print UserDefinedChar].
"Cut Paper"	Tap [Cut Paper].
"Feed Paper"	Tap [Feed Paper].
"POSQueryStatus"	Tap [POSQueryStatus].

● **COM Communication**
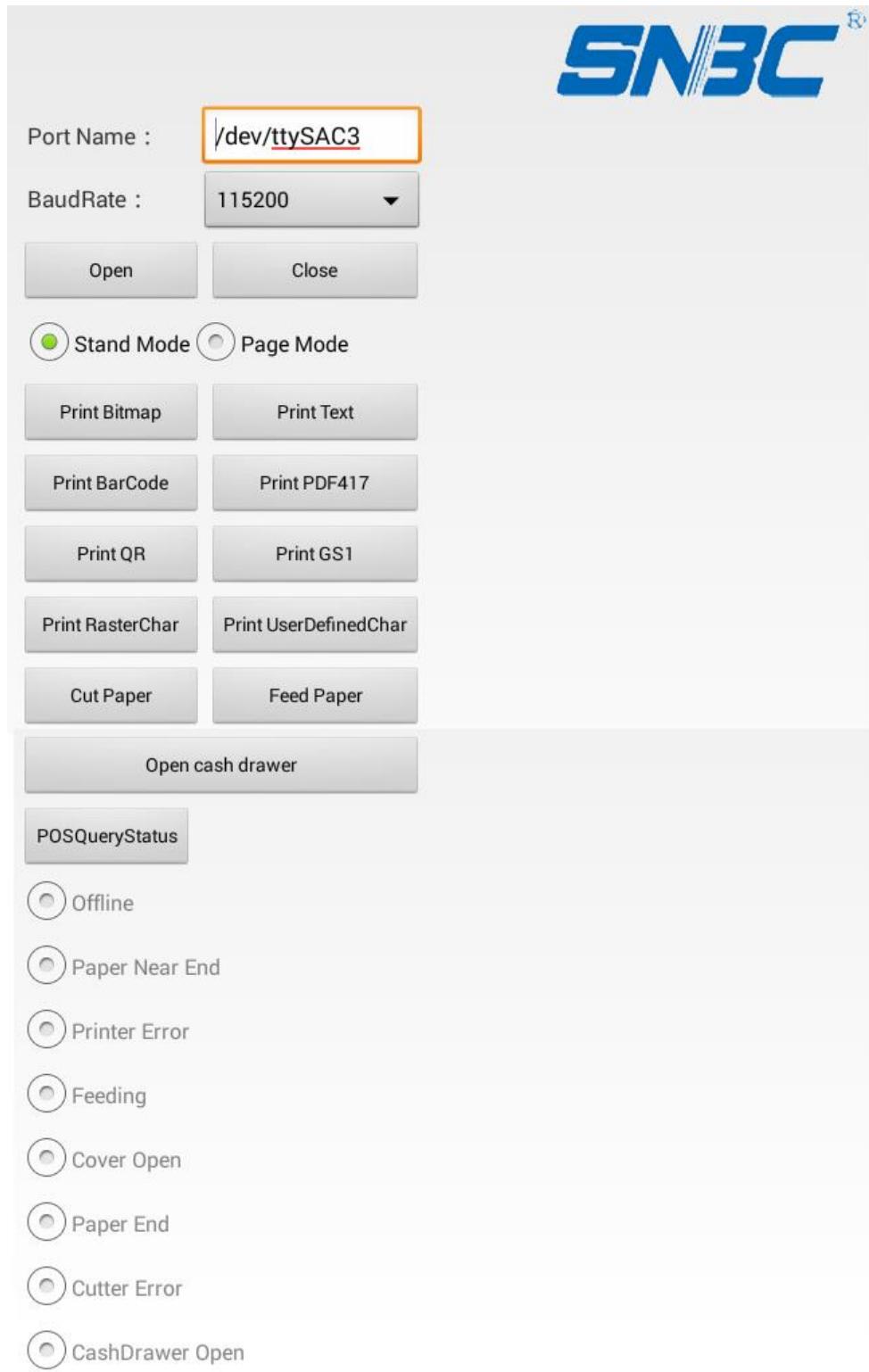
Ther

e are several following different controllers from USB screen:

Controller	Description
"Port Name"	Input port ID for device

"BaudRate"	Select baud rate.

● **WIFI Communication**
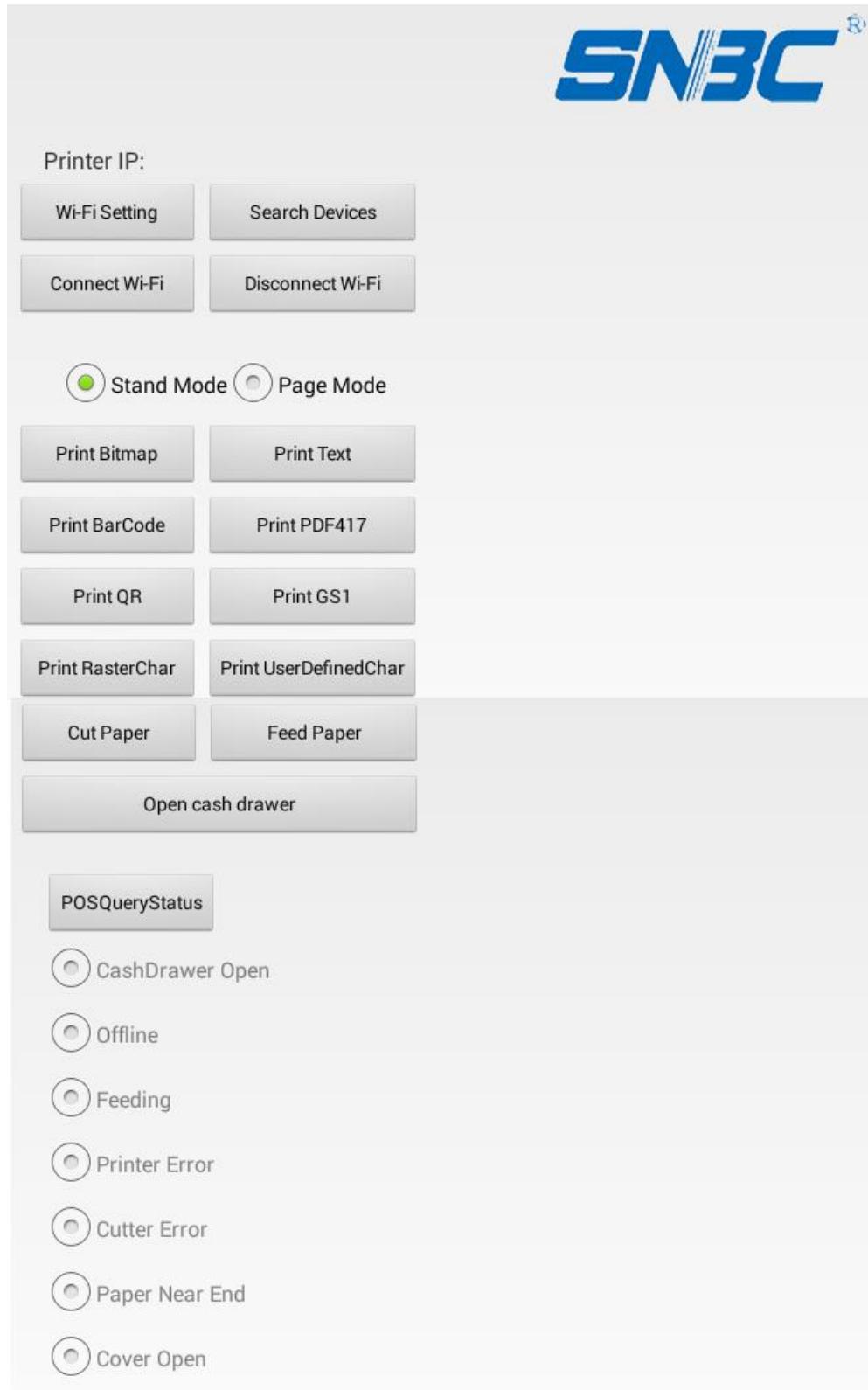
The

re are several following different controllers from USB screen:

Controller	Description
"Printer IP"	Input IP address.
"WI-FI Setting"	WIFI setting.
"Search Devices"	Search printer.
"Connect WI-FI"	Connect to port.
"Disconnect WI-FI"	Close port.

● **Bluetooth Communication**
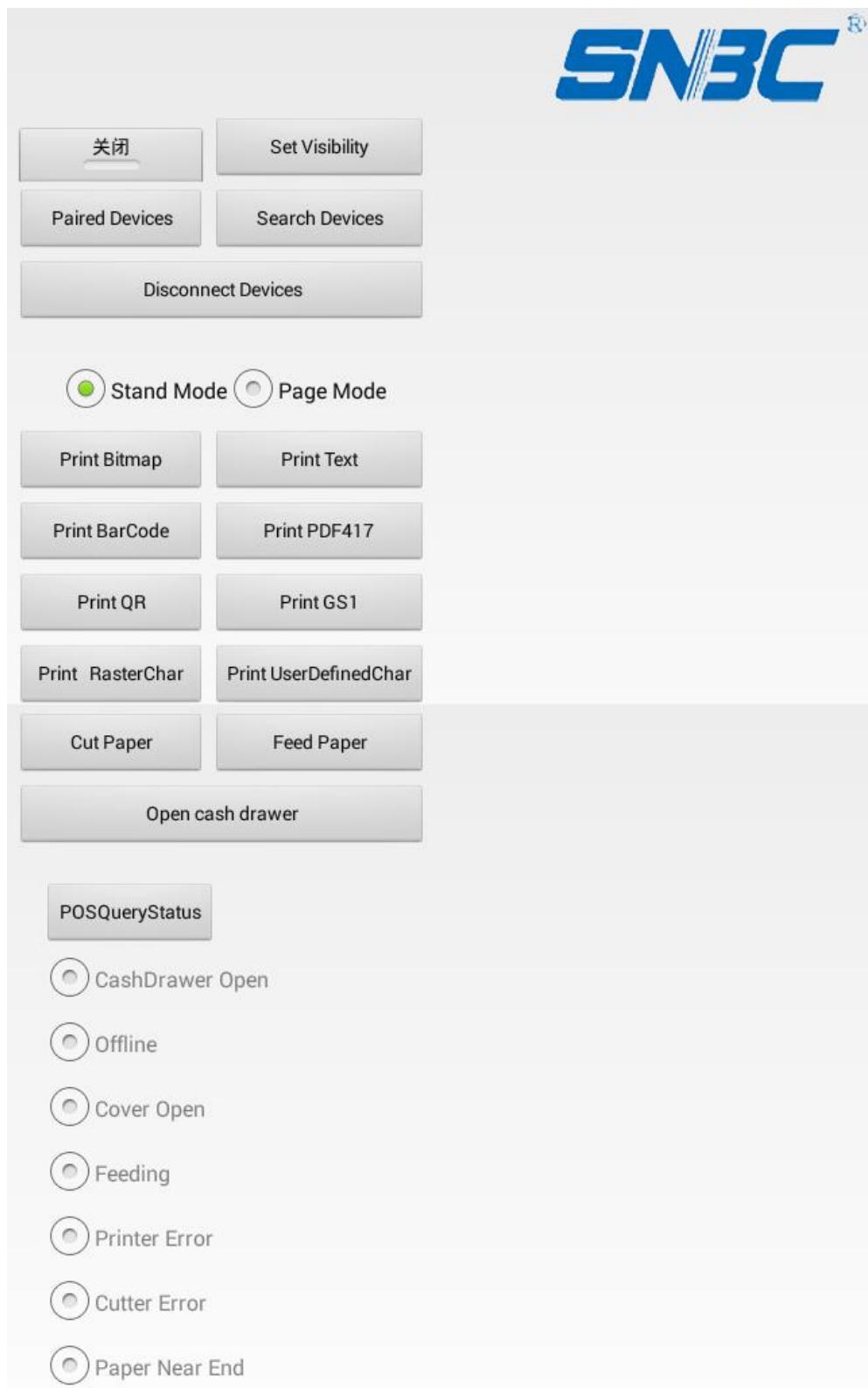
The re are several following different controllers from USB screen:

Controller	Description

"ON/OFF"	Turn on/off Bluetooth.
"Set Visibility"	Set device visibility.
"Paired Devices"	Search devices which have been paired.

● **Ethernet Communication**
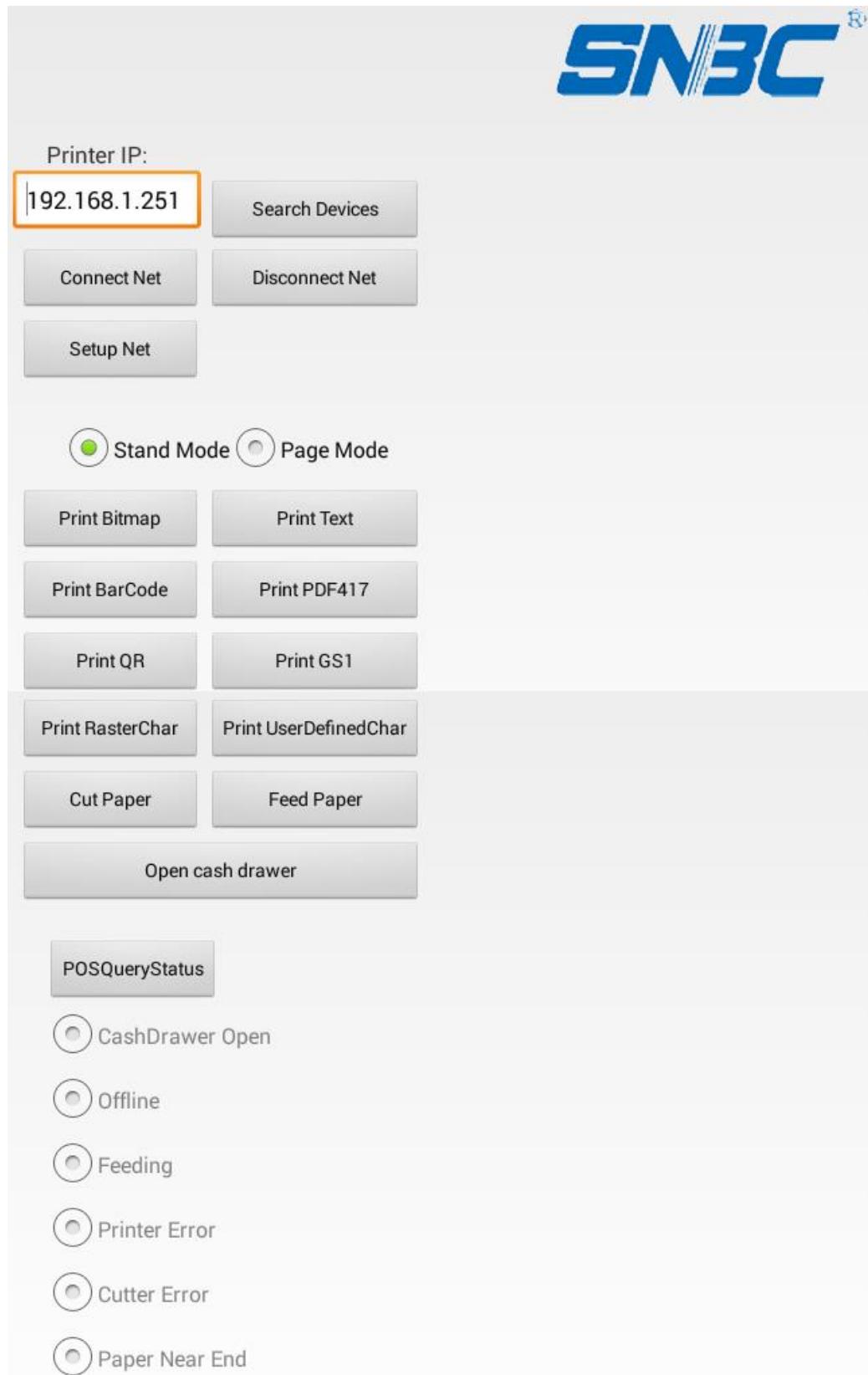
There are several following different controllers from USB screen:

"Printer IP" edit box	Enter the ip address of the printer.
"Search Devices"	Search printer.

"Connect NET"	Connect to port.
"Disconnect NET"	Close port.
"Setup Net"	Enter the screen to change the printer's network port Settings

●**Ethernet setting**

Tap [Setup Net], and you can enter ethernet setting screen:
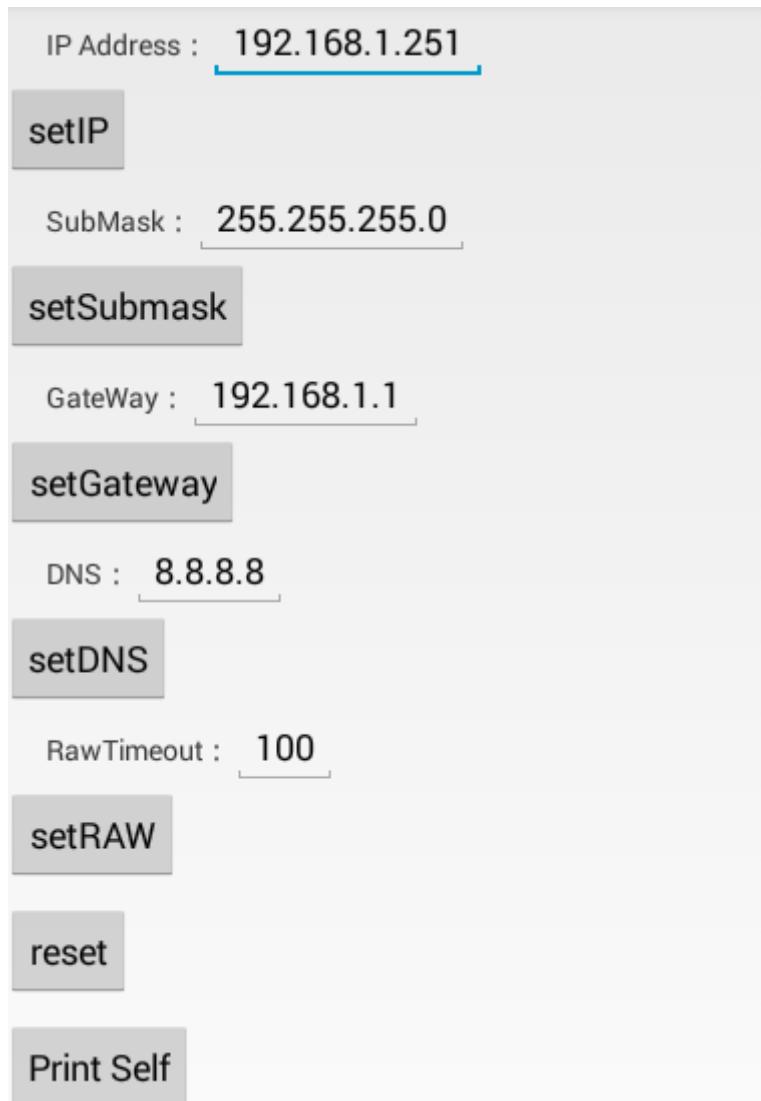
There are the following functions on "Setup Net" screen;

控件	描述
"IP Adress"	Enter the changed IP address of the printer
"setIp"	Change IP and print the self check page
"SubMask"	Enter the changed submask of the printer
"setSubMask" 按钮	Change submask

"GateWay"编辑框	Enter the changed gateway of the printer
"setGateWay"按钮	Change gateway
"DNS"编辑框	Enter the changed DNS of the printer
"setDNS"按钮	Change DNS
"RawTimeout"编辑框	Enter the changed RAW of the printer
"setRAW"按钮	Change RAW
"reset"按钮	Reset device
"Print Self"按钮	Print self-check page

● **Image printing**

Tap [Print Bitmap], and you can enter image printing screen:
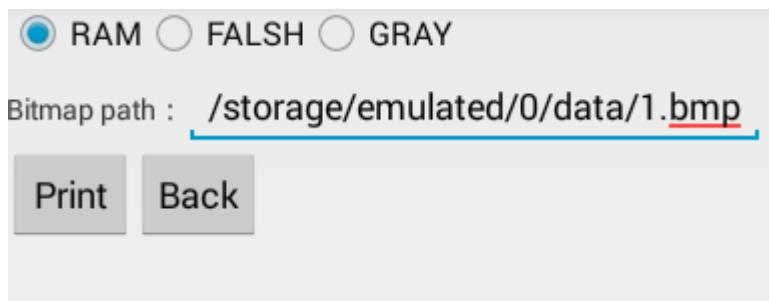
There are the following functions on "Print Bitmap" screen;

Controller	Description
Select"RAM"and"FALSH" and"GRAY"	RAM or Flash or GRAY image is selected to download.
"Bitmap path"	Input the image path or name. Remarks: If RAM and GRAY are specified, once a single image can be downloaded. If Flash is specified, multi-bitmaps can be downloaded, and they must be compared by "@", such as /data/bmp/1.bmp@2.bmp".
"Print"	Print image.
"Back"	Back to the upper screen.

● **Text printing**

Tap [Print Text], and you can enter text printing screen:
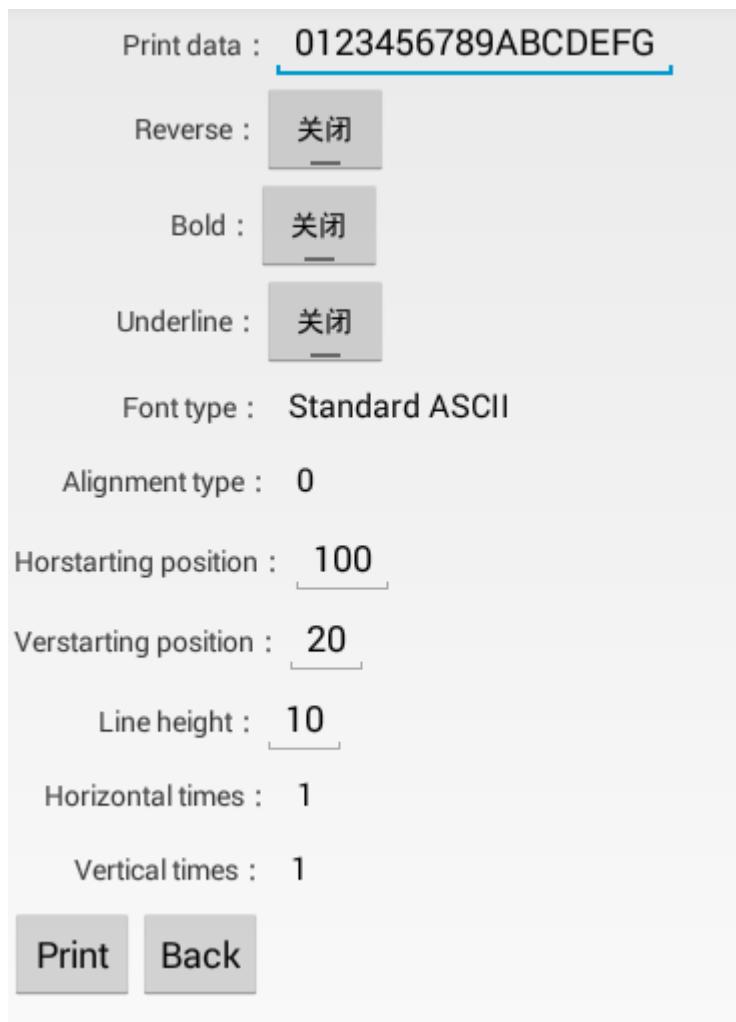
There are the following functions on "Print Text" screen:

Controller	Description
"Print data"	Enter a string to print.
"Reverse"	Turn white/black reverse mode on/off.
"Bold"	Turn bold on/off.
"Underline"	Turn underline on/off.
"Font type"	Select font type.
"Alignment type"	Select alignment type.
"Horstarting position"	Set horizontal starting position.
"Verstarting position"	Set vertical starting position.
"Line height"	Set line height.
"Horizontal times"	Select horizontal times.
"Vertical times"	Select vertical times.
"Print"	Print text.
"Back"	Back to the upper screen.

● **Barcode printing**

Tap [Print BarCode], and you can enter barcode printing screen:

Print data : 12345678901

BarCode type : UPC-A

Module width : 3

Barcode height : 100

HRI type : FontTypeStandardASCII

HRI position : 1
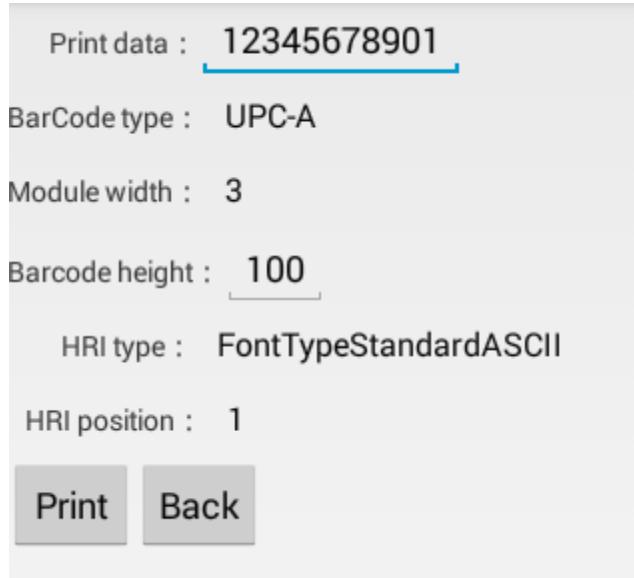
Print   Back

There are the following functions on "Print BarCode" screen:

Controller	Description
"Print data"	Input the barcode data.
"BarCode type"	Select barcode type.
"Module width"	Select module width.
"Barcode height"	Set barcode height.
"HRI type"	Select HRI type.
"HRI position"	Select HRI position
"Print"	Print barcode.
"Back"	Back to the upper screen.

● **Barcode PDF417 printing**

Tap [Print PDF417], and you can enter barcode PDF417 printing screen:
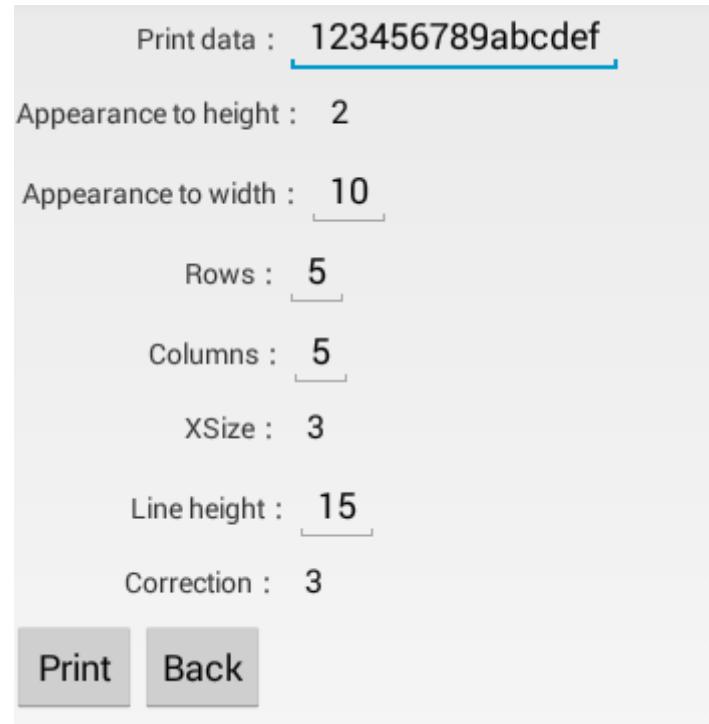
There are the following functions on "Print PDF417" screen:

Controller	Description
"Print data"	Input PDF417 barcode data.
"Appearance to height"	Set appearance to height for PDF417
"Appearance to width"	Set appearance to width for PDF417
"Rows"	Set the number of rows
"Columns"	Set the number of columns
"XSize"	Select XSize
"Line height"	Set line height.
"Correction"	Select correction grade.
"Print"	Print PDF417.
"Back"	Back to the upper screen.

● **Barcode QR printing**

Tap [Print QR], and you can enter barcode QR printing screen:
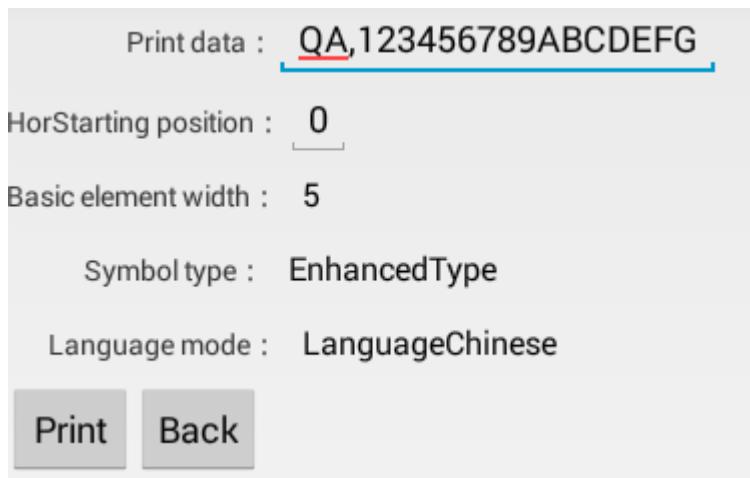
There are the following functions on "Print QR" screen:

Controller	Description
"Print data"	Enter QR barcode data.
"Horstarting position"	Set horizontal starting position
"Basic element width"	Select basic element width
"Symbol type"	Select symbol type.
"Language mode"	Select language mode
"Print"	Print QR.
"Back"	Back to the upper screen.

● **Barcode GS1 Databar printing**

Tap [Print GS1], and you can enter barcode GS1 printing screen:

Print data : 12345678901234

Barcode type : GS1DataBar Omnidirectional

Basic element width : 3

Barcode height : 50

Basic element height : 2

Separator height : 1

Segment height : 6

HRI type : 1

AI : 0

Print   Back

There are the following functions on "Print GS1" screen:

Controller	Description
"Print data"	Input GS1 barcode data.
"Barcode type"	Select GS1 barcode type.
"Basic element width"	Select basic element width.
"Barcode height"	The height of the DataBar, Stacked, stacked omnidirectional, expanded stacked barcode indicate the height of each line of barcode.
"Basic element height"	Select the basic element height of the 2D barcode in the composite barcode.
"Separator height"	Select the height of the separator.
"Segment height"	Set the number of segments of each line of barcode. Only in expanded stacked barcode should this parameter be set.
"HRI type"	Select the content of the note character
"AI"	Whether to use AI
"Print"	Print GS1.
"Back"	Back to the upper screen.

● **Text raster printing**

Tap [Print RasterChar], and you can enter raster text printing screen:

Print data : SNBC

Paint size : 50

Font bold : 1
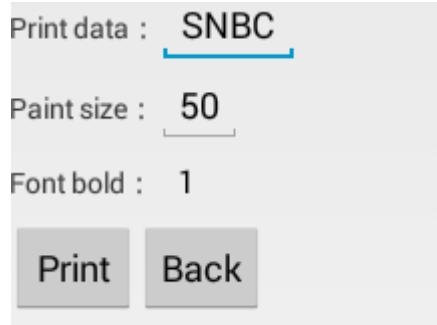
Print   Back

There are the following functions on "Print RasterChar" screen:

Controller	Description
"Print Data"	Enter a string to print.
"PaintSize"	Set canvas size.
"FontBold"	Set the degree of thickness of characters.
"Print"	Print raster text.
"Back"	Back to the upper screen.

3. Programming Guide

This chapter describes how to write programs in the application development using POS SDK For Android.

3.1 Connect to Printer's Port

● **Using USB to Connect**

Copy "android.hardware.usb.host.xml" in package to the file "/system/etc/permissions" of Android device. Set the permissions of USB HOST so that Android device can control printers by USB.

● **Using WIFI to Connect**

Make sure both the Android device and the printer set correct parameters (SSID, encryption, password, etc.), and connect to network correctly.And in Ad-Hoc mode,make sure the WIFI of the android device connect the printer.If the Android devices connected to the printer time over 3S, Please use the Android device static address, Setting the same segment of the IP address and gateway address at the same of the printer,Setting the domain name 1 and the domain name 2 is 0.0.0.0.

● **Using Bluetooth to Connect**

Make sure that Android device and printer have been paired, and make sure the Bluetooth of Android device has been turned on.

● **Using COM to Connect**

Make sure that Android device did not be occupied by the system console. Please input the right port ID and baud rate. Ensure that Android device and printer have the same baud rate. In addition to ensure the COM parameters of EEPROM are the default settings. Such as,handshake protocol is hard, data bit is 8, check bit is 0, stop bit is 1.

●**Using Ethernet to Connect**

Make sure both the Android device and the printer connect to network correctly. And in Ad-Hocmode make sure the IP of the android device connect the printer. If the Android devices connected to the printer time over 3S, Please use the Android device static address, Setting the same segment of the IP address and gateway address at the

same of the printer,Setting the domain name 1 and the domain name 2 is 0.0.0.0.

3.2 Using SDK

This part describes how to using the JAR package of POS serial printers in Android.

● **How to import JAR package**

It's easy to import JAR package. It's to put JAR package into the libs founder of the project. If this catalog did not exist, you can create one by yourself. You can right-click and select "Refresh", and then, you can see JAR package has been imported to libs. If you can not see the JAR package, you can re-import.

● **How to import JNI DLL of COM**

For USB, WIFI,Ethernet and bluetooth, only need to use the JAR package. However, it's also need libserialport.so DLL except for the JAR package for COM.

DLL and JARpackage import are the same. That is, It's OK to put the .so file into the libs catalog.

● **How to Use API methods in JAR Package**

If you want to use API methods in JAR package, you need import the class and create the project firstly. The parent class is POSInterfaceAPI, and the child classes are POSUSBAPI (for USB), POSWIFIAPI (for WIFI), POSBluetoothAPI (for bluetooth) and POSSerialAPI (for COM) and POSNETAPI (for Ethernet).

```
import POSAPI.POSInterfaceAPI;
import POSAPI.POSWIFIAPI;
import POSAPI.POSUSBAPI;
import POSAPI.POSBluetoothAPI;
import POSAPI.POSSerialAPI;
import POSAPI.POSNETAPI;

//Create project for USB
POSInterfaceAPI  interface_usb = new POSUSBAPI(this);
//Create project for WIFI
POSInterfaceAPI interface_wifi = new POSWIFIAPI();
//Create project for Bluetooth
POSInterfaceAPI  interface_blue=
POSBluetoothAPI.getInstance(Activity.this);
//Create project for COM
POSInterfaceAPI  interface_com = new POSSerialAPI();
//创建网口对象
```

```
POSInterfaceAPI interface_net = new POSNETAPI();
```

After creating the project, API methods can be called.

● How to Use API methods of POSSDK Class in JAR Package

If you want to use API methods of POSSDK class in JAR package, you need import the POSSDK class and create the project whose class name as POSSDK firstly.

```
import POSSDK.POSSDK;
//Create the project of USB for the POSSDK class
POSSDK pos_sdk = new POSSDK(interface_usb);
```

● Example of methods

Search printer of WIFI

```
private static final int SearchPortMAX = 10;
private SearchPortInfo port_info[] = new SearchPortInfo[SearchPortMAX];
private int sch_prt_num = 0;
for(i = 0; i < SearchPortMAX; i++){
    port_info[i] = new SearchPortInfo();
}
sch_prt_num = interface_wifi.WIFISearchPort(port_info, SearchPortMAX);
```

Search printer of Ethernet

```
private static final int SearchPortMAX = 10;
private SearchPortInfo port_info[] = new SearchPortInfo[SearchPortMAX];
private int sch_prt_num = 0;
for(i = 0; i < SearchPortMAX; i++){
    port_info[i] = new SearchPortInfo();
}
sch_prt_num = interface_net.NetSearchPort(port_info, SearchPortMAX);
```

Connect to port for USB

```
error_code = interface_usb.OpenDevice();//connect to printer device of normal
error_code = interface_usb.OpenDevice(5455,5455);//connect to printer device by vid and pid
```

Connect to printer's port for WIFI

```
private static final int POSPORT = 9100;
private static final int STATEPORT = 4000;
private static String POSIP = "192.168.1.210";
error_code = interface_wifi.OpenDevice(POSIP, POSPORT);
```

Connect to printer's port for Ethernet

```
private static final int POSPORT = 9100;
```

```
private static final int STATEPORT = 4000;
private static String POSIP = "192.168.1.210";
error_code = interface_net.OpenDevice(POSIP, POSPORT);
```

Connect to printer's port for Bluetooth

```
String address = "00:1B:35:07:16:AC";
error_code = interface_blue.OpenDevice(address);
```

Connect to printer's port for COM

```
String port_name = "/dev/ttySAC3";
int baud_rate = 115200;
error_code = interface_com.OpenDevice(new File(port_name),baud_rate);
```

Close port

```
error_code = interface_wifi.CloseDevice();
interface_wifi = null;
```

Set paramters for standard mode

```
private static final int PRINT_MODE_STANDARD = 0;
private static final int PRINT_MODE_PAGE = 1;
error_code = pos_wifi.systemSelectPrintMode(PRINT_MODE_STANDARD);
```

Set paramters for page mode

```
error_code = pos_wifi.systemSelectPrintMode(PRINT_MODE_PAGE);
error_code = pos_sdk.pageModeSetPrintArea(0,0,640,500,0);
error_code = pos_sdk.pageModeSetStartingPosition(20,200);
```

Text printing

```
String txtbuf = "123456789";
byte []send_buf = txtbuf.getBytes("GB18030");
error_code = pos_sdk.textPrint(send_buf, send_buf.length);
send_buf = null;
```

Raster text printing

```
String txt = "SNBC";
c_image = pos_sdk.imageCreateRasterBitmap(txt,50,1);
error_code = pos_sdk.imageStandardModeRasterPrint(c_image,640);
```

Print user defined character

```
String path =  "/data/bmp/u1.bmp" + "@" + "/data/bmp/u2.bmp" + "@" +
               "/data/bmp/u3.bmp";
FileInputStream temp_stream = null;
String sigPaths[] = path.split("@");
int image_num = sigPaths.length;
Bitmap cg_image[] = new Bitmap[image_num];
int i = 0;
for(i = 0; i < image_num; i++){
    try {
```

```
                temp_stream = new FileInputStream(sigPaths[i]);
        } catch (FileNotFoundException e) {
                        e.printStackTrace();
        }
   if(temp_stream == null){
                        return;
                }
    cg_image[i] = BitmapFactory.decodeStream(temp_stream);
    temp_stream = null;
 }
 error_code = pos_sdk.textUserDefinedCharacterEnable(1);
 if(error_code != POS_SUCCESS){
                        return;
                }
error_code = pos_sdk.textUserDefinedCharacterDefine(3, 12, 48, 50,
cg_image);
if(error_code != POS_SUCCESS){
                        return;
                }
error_code = pos_sdk.textSelectFontMagnifyTimes(2,2);
error_code = pos_sdk.textUserDefinedCharacterCancel(48);
error_code = pos_sdk.textUserDefinedCharacterCancel(49);
error_code = pos_sdk.textUserDefinedCharacterCancel(50);
```

Print barcode

```
String pszBuffer = "012345678912";
error_code =
pos_sdk.barcodePrint1Dimension(pszBuffer,pszBuffer.length(),BarcodeUP
C A,4, 100,0,1);
```

Print PDF417

```
String str = "123456789";
int data_size = 0;
try {
        data_size=str.getBytes("GB18030").length;
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    }
error_code = pos_sdk.barcodePrintPDF417(str,
data_size,2,10,5,5,3,10,3);
```

Print QR

```
String str = "123456789";
int data_size = 0;
try {
        data_size=str.getBytes("GB18030").length;
```

```
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();

    }
error code = pos sdk.barcodePrintQR(str,data size, 0, 5, 1, 0);
```

Print Maxicode

```
String str = "123456789";
int data_size = 0;
try {
        data_size=str.getBytes("GB18030").length;
    } catch (UnsupportedEncodingException e) {
            e.printStackTrace();

    }
error code = pos sdk.barcodePrintMaxicode (str,data size);
```

Print GS1 DataBar and GS1 composite barcode

```
String str = "123456789";
int data_size = 0;
try {
        data_size=str.getBytes("GB18030").length;
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();

    }
error_code = pos_sdk.barcodePrintGS1DataBar(str, data_size, 1,3,50, 5,
3, 6, 1, 0);
```

Image print

```
String str = "/data/bmp/Look.bmp";
 try {
        temp_stream = new FileInputStream(str);
    } catch (FileNotFoundException e) {
                e.printStackTrace();

    }
    if(temp_stream == null){
         return;
      }
image = BitmapFactory.decodeStream(temp_stream);
error_code = pos_sdk.imageStandardModePrint (image, 33, 0, 640);
```

RAM/Flash image download and print

RAM

```
error_code = pos_sdk.imageDownloadToPrinterRAM(2, image, 640);
error_code = pos_sdk.imageRAMPrint(2,0);
```

Flash

```
String path =  "/data/bmp/Look.bmp" + "@" + "/data/bmp/s.bmp" + "@" +
             "/data/bmp/1.bmp";
```

```java
String sigPaths[] = path.split("@");
int image_num = sigPaths.length;
Bitmap cg_image[] = new Bitmap[image_num];
int i = 0;
for(i = 0; i < image_num; i++){
    try {
            temp_stream = new FileInputStream(sigPaths[i]);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
    if(temp_stream == null) {
            return;
    }
    cg_image[i] = BitmapFactory.decodeStream(temp_stream);
    temp_stream = null;
}
error_code = pos_sdk.imageDownloadToPrinterFlash(image_num, cg_image,
640);
 if(error_code != POS_SUCCESS){
                    return;
            }
error_code = pos_sdk.imageFlashPrint(1, 0);
error_code = pos_sdk.imageFlashPrint(2, 0);
error_code = pos_sdk.imageFlashPrint(3, 0);
for(i = 0; i < image_num; i++){
        cg_image[i].recycle();
 }
```

Raster image print

```java
error_code = pos_sdk.imageStandardModeRasterPrint(image,640);
```

Download file

```java
String str = "/data/bmp/BTP-R980.JK";
error_code = pos_sdk.systemDownloadFile(str,5000);
if(error_code == POS_SUCCESS){
    System.out.println("Download file success!");
    return error_code;
}else{
    System.out.println("Download file fail!");
}
```

Query Status

```java
final int QueryStatusSize=4;
byte StatusBuffer[] = new byte[QueryStatusSize];
error_code =
pos_sdk.systemQueryStatus(StatusBuffer,QueryStatusSize,1);
```

Feed line

```
error_code = pos_sdk.systemFeedLine(3);
```

Cut paper

```
error_code = pos_sdk.systemCutPaper(65, 0);
```

Change the IP address of the printer

```
strMac = "001341 0EAAA8";
String strIpAddress = "192.168.1.251";
error_code = interface_net.SetIpAddress(strMac, strIpAddress);
```

Change the subnet mask for the printer

```
strMac = "001341 0EAAA8";
strSubMask = "255.255.255.0";
error_code = interface_net. SetSubMask(strMac, strSubMask);
```

Change the gateway for the printer

```
strMac = "001341 0EAAA8";
strGateWay = "192.168.1.1";
error_code = interface_net.SetGateWay(strMac, strGateWay);
```

Change the dns for the printer

```
strMac = "001341 0EAAA8";
strDns = "8.8.8.8";
error_code = interface_net.SetDnsAddress(strMac, strDns);
```

Change the raw for the printer

```
strMac = "001341 0EAAA8";
strRaw = "100";
error_code = interface_net.SetRawTime(strMac, strRaw);
```

Reset

```
strMac = "001341 0EAAA8";
error_code = pos_sdk.resetDevice(strMac);
```

Print self-check page

```
error_code = pos_sdk.printSelftest();
```

4. API Reference

This chapter describes the API provided in the POS SDK for Android.

4.1 Interface Reference

API	描述
OpenDevice	Connect to printer's port for USB.
OpenDevice：Vid, Pid	Connet to printer which is specified Vid and Pid for USB
OpenDevice：device, baudrate	Connect to printer for COM
OpenDevice：dstName, dstPort	Connect to printer for WIFI and Ethernet
OpenDevice：macAddress	Connect to printer for Bluetooth
CloseDevice	Close connecting.
WIFISearchPort	Search printers.
NETSearchPort	Search printers
SetIpAddress	Change the IP address of the printer
SetSubMask	Change the subnet mask for the printer
SetGateWay	Change the gateway for the printer
SetDnsAddress	Change the DNS for the printer
SetRawTime	Change the RAW for the printer
WriteBuffer	Send data to a port.
ReadBuffer	Receive data from device.
recordCommunicationData	Record communication data.
LogTrace	Record log.

●OpenDevice

Connect to printer's port for USB.

Method

- public int **OpenDevice**()

Return

Return Value	Description

| POS_SUCCESS | Connect successfully. |
| ERR_PROCESSING | Connecting failed. |

Example

See also: <u>Connect to port for USB</u>.

●OpenDevice：Vid, Pid

Connet to printer which is specified Vid and Pid for USB.

Method

- public int **OpenDevice** (int Vid,int Pid)

Parameter

- **Vid**   The Vid of printer.
- **Pid**   The Pid of printer.

- **Return**

Return Value	Description
POS_SUCCESS	Connect successfully.
ERR_PROCESSING	Connecting failed.

Example

See also: <u>Connect to port for USB</u>.

●OpenDevice：device, baudrate

Connect to printer for COM.

Method

- public int **OpenDevice**(File device,int baudrate)

Parameter

- **device**   Serial number, such as /dev/ttySAC3.
- **baudrate**   Baud rate. The printer and the development board serial port baud rate must be the same.

- **Return**

Return Value	Description
POS_SUCCESS	Connect successfully.
ERR_PROCESSING	Connecting failed.

Example

See also: <u>Connect to printer's port for COM</u>.

●OpenDevice：dstName, dstPort

Connect to printer for WIFI.

Method

- public int **OpenDevice**(String dstName, int dstPort)

Parameter

- **dstName**   IP address of printer, Such as 192.168.1.200.
- **dstPort**   Port ID. 9100 or 4000

- **Return**

Return Value	Description
POS_SUCCESS	Connect successfully.
ERR_PROCESSING	Connecting failed.

Example

See also: <u>Connect to printer's port for WIFI</u>.

●OpenDevice：macAddress

Connect to printer for Bluetooth.

Method

- public int **OpenDevice**(String macAddress)

Parameter

- **macAddress**   Mac address of printer, Such as 00:1B:35:07:16:AC.

- **Return**

Return Value	Description

| POS_SUCCESS | Connect successfully. |
| ERR_PROCESSING | Connecting failed. |

Example

See also: Connect to printer's port for Bluetooth.

●CloseDevice

Close connecting.

Method

- public int **CloseDevice**()

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PROCESSING	Processing was failed.

Example

See also: Close port.

●WIFISearchPort

Search printers.

Method

- public int **WIFISearchPort**(SearchPortInfo port_info[],int printer_num_max)

Parameter

- **Searchport_info**   The message of printers which were searched.
- **printer_num_max**   The maximum information which were required to search.

Return

Return Value	Description
The number of printers which were searched	Search successfully.
0	No printer was searched.

The printer message list, as follows:

@ **public class SearchPortInfo**

```
{
    private String Allinfo;     //all message
    private String Prt_name;    //device name
    private String IP_address;  //IP address
    private String MAC_address; // MAC address
}
```

Example

See also: <u>Search printer of WIFI</u>.

●NETSearchPort

Search printers.

Method

- public int **NETSearchPort**(SearchPortInfo port_info[],int printer_num_max)

Parameter

- **Searchport_info**   The message of printers which were searched.

- **printer_num_max**   The maximum information which were required to search.

Return

Return Value	Description
The number of printers which were searched	Search successfully.
0	No printer was searched.

The printer message list, as follows:

```
@ public class SearchPortInfo
{
    private String Allinfo;     //all message
    private String Prt_name;    //device name
    private String IP_address;  //IP address
    private String MAC_address; // MAC address
}
```

Example

See also: <u>Search printer of Ethernet.</u>

●SetIpAddress

Change the IP address of the printer.

37

Method

     - public int **SetIpAddress**(String strMac, String strIpAddress)

Parameter

- **strMac**   Mac address of printer
- **strIpAddress** Changed IP address

Remarks

  a) The format of the strMac parameter is "****** *****",Such as "001341 0EAAA8".

  b) The parameter strIpAddress is 4 bytes, each with a range of 0-255, such as "192.168.1.251".

  c) Call systemResetDevice interface reset or restart printer can only take effect.

Return

返回值	情况
POS_SUCCESS	successful
ERR_PROCESSING	failure
ERR_PARAM	Parameter error

示例代码

See also: Change the IP address of the printer

●SetSubMask

   Change the subnet mask for the printer.

Method

     - public int **SetSubMask**(String strMac, String strSubMask)

Parameter

- **strMac**   Mac address of printer
- **strSubMask** Changed subnet mask

Remarks

  a) The format of the strMac parameter is "****** *****",Such as "001341 0EAAA8".

  b) The parameter strSubMask is usually represented by the same dotted decimal as

the IP address format, such as "255.255.255.0".

c) Call systemResetDevice interface reset or restart printer can only take effect.

Return

返回值	情况
POS_SUCCESS	successful
ERR_PROCESSING	failure
ERR_PARAM	Parameter error

示例代码

See also: Change the subnet mask for the printer

●SetGateWay

Change the gateway for the printer.

Method

- public int **SetGateWay(**String strMac, String strGateWay**)**

Parameter

- **strMac**   Mac address of printer
- **strGateWay** Changed gateway

Remarks

a) The format of the strMac parameter is "****** *****",Such as "001341 0EAAA8".

b) The parameter strGateWay is usually represented by the same dotted decimal as the IP address format, such as "255.255.255.0".

c) Call systemResetDevice interface reset or restart printer can only take effect.

Return

返回值	情况
POS_SUCCESS	successful
ERR_PROCESSING	failure
ERR_PARAM	Parameter error

示例代码

See also: Change the gateway for the printer

●SetDnsAddress

Change the DNS for the printer.

Method

    - public int **SetDnsAddress**(String strMac, String strDnsAddress)

Parameter

- **strMac**   Mac address of printer
- **strDnsAddress** Changed DNS

Remarks

   a) The format of the strMac parameter is "****** *****",Such as "001341 0EAAA8".

   b) The parameter strDnsAddress is usually represented by the same dotted decimal as the IP address format, such as "255.255.255.0".

   c) Call systemResetDevice interface reset or restart printer can only take effect.

Return

返回值	情况
POS_SUCCESS	successful
ERR_PROCESSING	failure
ERR_PARAM	Parameter error

示例代码

See also: Change the dns for the printer

●SetRawTime

Change the RAW for the printer.

Method

    - public int **SetRawTime**(String strMac, int rawTime)

Parameter

- **strMac**   Mac address of printer
- **strSubMask** Changed RAW

Remarks

   a) The format of the strMac parameter is "****** *****",Such as "001341

0EAAA8".

b) The rawTime parameter is an integer of 0 or greater than 0.

c) Call <u>systemResetDevice</u> interface reset or restart printer can only take effect, the printer automatically closes the port when no data is issued after the timeout.

Return

返回值	情况
POS_SUCCESS	successful
ERR_PROCESSING	failure
ERR_PARAM	Parameter error

示例代码

See also: <u>Change the raw for the printer</u>

●WriteBuffer

Send data to a port. When the data is More than 4096 bytes, packet disassemble packet is 4096 bytes.

Method

- public int **WriteBuffer** (byte[] WriteBuffer,int OffsetSize,int nBytesToWrite, int WriteTimeOut)

Parameter

- **WriteBuffer**   The sending data buffer. It stores data to be sent.
- **OffsetSize**   Specify the offset value from the top of WriteBuffer.
- **nBytesToWrite**   The number of bytes which would be sent.
- **WriteTimeOut**   The timeout of writing data (ms).

Return

Return Value	Description
The bytes of sending data	Processing was successful.
The bytes of sending data the actual	Sending part of data
0	Sending failed.

Example

pszCommand [2] = {0x1b,0x40}; nReturn = WriteBuffer(pszCommand,0,2,WRITETIMEOUT);

●ReadBuffer

Receive data from device.

Method

- public int **ReadBuffer** (byte[] ReadBuffer,int OffsetSize,int nBytesToRead,int ReadTimeOut)

Parameter

- **ReadBuffer**   The receiving data buffer for storing received data.
- **OffsetSize**   Specify the offset value from the top of ReadBuffer.
- **nBytesToRead**   The number of bytes which would be received.
- **ReadTimeOut**   The timeout of receiving data (ms).

Return

Return Value	Description
The bytes of reading data	Processing was successful.
0	Reading failed.

Example

```
byte pointBuffer[4];
nReturn = ReadBuffer(pointBuffer,0, pointBuffer.length,10000);
```

● recordCommunicationDataEnable

Record communication data,the file size does not exceed 5M.

Method

- public int **recordCommunicationData**(Context contexts,int IsRecord,String FileName)

Parameter

- contexts   Directory of application
- IsRecord   Whether to record communication data or not
- FileName   The file name recorded.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.

ERR_PROCESSING	Processing was failed.

Example

recordCommunicationData(MainActivity.this,0x01,data_file_name);

●LogTrace

Record log, the file size does not exceed 5M.And record the time log for system standard GMT time.If the showing time of android devices is different from the standard GMT time,the time of recording is different from the showing time of android devices.

Method

-public int **LogTrace**(Context contexts,int IsRecord,String FileName)

Parameter

- contexts   Directory of application
- IsRecord   Whether to record log or not
- FileName   The file name recorded.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PROCESSING	Processing was failed.

Example

LogTrace(MainActivity.this,0x01,log_file_name);

4.2 POS SDK API Reference

Prefix	API	Description
Prefix as system (methods about system)	systemDownloadFile	Download File.
	systemReset	Initialize printer, clear data in print buffer and set print mode to the default mode when powered on.
	systemResetDevice	Reset
	systemSelectPrintMode	Select the print mode.
	systemSelectPaperType	Select the paper type.

	systemSetMotionUnit	Set the horizontal and vertical motion units.
	systemQueryStatus	Query the printer's status.
	systemFeedLine	Print and feed line.
	systemCutPaper	Select cut paper mode and cut paper.
	systemPrintSelftest	Print self-check page
Prefix as cashdrawer	cashdrawerOpen	Output the cash drawer control pulse to specified connector pin.
Prefix as text (methods about text)	textSelectCharSetAndCodePage	Select an international character set and Code page.
	textSetLineHeight	Set line height.
	textSetCharacterSpace	Set character spacing.
	textStandardModeAlignment	Align all the data in one line to the specified position (Standard mode).
	textStandardModeUpsideDown	Turn on/off upside-down printing mode.
	textPrint	Print text.
	textSelectFontMagnifyTimes	Select character size.
	textStandardModeRotate	Rorate integer times 90 degree.
	textSelectFont	Select character font and font style.
	textEnterOrQuitColorPrint	Enter/ Quit bi-colour print mode.
	textSetColorPrint	Set the printing color
	textUserDefinedCharacterEnable	User-defined character is enable/disable.
	textUserDefinedCharacterDefine	Define user-defined character.
	textUserDefinedCharacterCancel	Text cancel font user-defined of char code.
Prefix as image (methods about image)	imageCreateRasterBitmap	Text raster print.
	imageStandardModePrint	Image print in standard mode.
	imageDownloadToPrinterRAM	Download images to RAM.
	imageRAMPrint	Print image which downloaded to RAM.
	imageDownloadToPrinterFlash	Download images to Flash.
	imageFlashPrint	Print image which downloaded to Flash.

	imageStandardModeRasterPrint	Print raster image in standard mode.
	imageCompressedRasterPrint	Print compressed image in standard mode.
	imageStandardModeGrayPrint	Print gray image in standard mode.
Prefix as barcode (methods about barcode)	barcodePrint1Dimension	Print 1Dimension barcode.
	barcodePrintQR	Print barcode QR.
	barcodePrintPDF417	Set barcodePDF417 size and print PDF417.
	barcodePrintMaxicode	Print Maxicode.
	barcodePrintGS1DataBar	Print GS1 DataBar and GS1 composite barcode.
Prefix as standardMode (methods about standard mode)	standardModeSetPrintAreaWidth	Set print area postion and width in standard mode.
	standardModeSetStartingPosition	Set horizontal starting position in standard mode.
Prefix as pageMode (methods about page mode)	pageModeSetStartingPosition	Set horizontal and vertical starting position in page mode.
	pageModeSetPrintArea	Set print area in page mode.
	pageModePrint	Print data in page mode
	pageModeClearBuffer	Delete all the print data in current area.

● **systemDownloadFile**

Download file.

Method

- public int **systemDownloadFile**(String FileName,int TimeOut)

Parameter

- **FileName**   The file name of the file which would be download.
- **TimeOut**   The timeout of downloading file.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_COMMUNICATE	Processing was failed.
ERR_PARAM	The file name is wrong or data is null.

Example

```
String str = "/data/bmp/BTP-R980.JK";
error_code = pos_sdk.systemDownloadFile(str,5000);
```

● **systemReset**

Initialize printer, clear data in print buffer and set print mode to the default mode when powered on.

Method

- public int **systemReset**()

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PROCESSING	Processing was failed.

Example

```
pos_sdk.systemReset();
```

● **systemResetDevice**

Reset.

Method

- public int **systemResetDevice**(String strMac)

Parameter

- **strMac** Mac address of printer

Remarks

a) The format of the strMac parameter is "****** *****".Such as "001341 0EAAA8".

b) This interface takes effect when the printer is configured on the network port. It

is not supported in other cases.

Return

Return Value	Description
POS_SUCCESS	Successful.
ERR_PROCESSING	Failure
ERR_PARAM	Parameter error

Example

See also:Reset

● systemSelectPrintMode

Select the print mode(This command sets the print position to the beginning of the line).

Method

- public int **systemSelectPrintMode**(int Mode)

Parameter

- **Mode**   Print mode

PrintMode Set Value	Description
PRINT_MODE_STANDARD	Selsct standard mode.
PRINT_MODE_PAGE	Select page mode
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_SELECT_PRINT_MODE	The print mode selecting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.systemSelectPrintMode(PRINT_MODE_STANDARD);// Standard mode
error_code = pos_sdk.systemSelectPrintMode(PRINT_MODE_PAGE);// Page mode

● systemSelectPaperType

Select the paper type.

Method

   - public int **systemSelectPaperType**(int PaperType)

Parameter

• **PaperType**   Paper type

PaperType Set Value	Description
PaperTypeCoutinuous	Coutinuous paper
PaperTypeMarked	Marked paper
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_SELECT_PAPER_TYPE	The paper type selecting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_wifi. systemSelectPaperType (PaperTypeCoutinuous);// Coutinuous paper

error_code = pos_wifi. systemSelectPaperType (PaperTypeMarked);// Marked paper

● **systemSetMotionUnit**

   Set the horizontal and vertical motion units.

Method

   - public int **systemSetMotionUnit**(int HorizontalUnit,int VerticalUnit)

Parameter

• **HorizontalUnit**   Horizontal unit

HorizontalUnit Set Value	Description
0-255	Legal value
Other values	Invalid parameter

• **VerticalUnit**   Vertical unit

HorizontalUnit Set Value	Description
0-255	Legal value
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_SET_MOTION_UNIT	The motion uint selecting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.systemSetMotionUnit(203,203);

● **systemQueryStatus**

Query the printer's status.

Method

- public int **systemQueryStatus**(byte[] QueryStatusBuffer,int ReadSize,int port_type)

Parameter

- **QueryStatusBuffer**   The buffer for storing the printer's status.
- **ReadSize**   The number of bytes which would be read.
- **Port_type**   The type of port.

Port_type Set Value	Description
1	USB
2	COM
3	WIFI
4	Bluetooth
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_QUERY_STATUS	The printer's status querying failed.
ERR_PARAM	An invalid parameter was passed

Example

final int QueryStatusSize=4;
byte StatusBuffer[] = new byte[QueryStatusSize];
error_code = pos_sdk.systemQueryStatus(StatusBuffer,QueryStatusSize,1);

● **systemFeedLine**

Print and feed line.

Method

- public int **systemFeedLine**(int LineNum)

Parameter

• **LineNum**   The number of feed line

LineNum Set Value	Description
0-255	Legal value
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_FEED_LINE	Feed line failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.systemFeedLine(5);

● **systemCutPaper**

Select cut paper mode and cut paper.

Method

- public int **systemCutPaper**(int CutMode, int FeedDistance)

Parameter

• **CutMode**   Feed paper distance

LineNum Set Value	Description
CutFullImmdediately	Full cut
CutPartImmdediately	Part cut
CutPartAfterFeed	Feed line and part cut
Other values	Invalid parameter

• **FeedDistance**   The distance of feed line

FeedDistance Set Value	Description
0-255	Legal value
Other values	Invalid parameter

Remarks

a) If the parameter CutMode is specified CutFullImmdediately or CutPartImmdediately, the parameter FeedDistance will be ignored.

b) If the parameter CutMode is specified CutPartAfterFeed, the printer will feed FeedDistance distance and cut.

c) The parameter FeedDistance will be ignored at the Mark Paper mode. The printer will find the mark and cut.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_SYSTEM_CUT_PAPER	The paper cutting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.systemCutPaper(CutPartAfterFeed,80);

● **systemPrintSelftest**

Print self-check page.

Method

- public int **systemPrintSelftest**

Parameter

Return

Return Value	Description
POS_SUCCESS	Successful.
ERR_PROCESSING	Failure

Example

See alos: <u>Print self-check page</u>

● cashdrawerOpen

Output the cash drawer control pulse to specified connector pin.

Method

- public int **cashdrawerOpen**(int CashdrawerID, int PulseOnTimes, int
PulseOffTimes)

Parameter

- **CashdrawerID**   Connector pin of cashdrawer

CashdrawerID Set Value	Description
0	Drawer kick-out connector pin 2
1	Drawer kick-out connector pin 5
Other values	Invalid parameter

- **PulseOnTimes**   The pulse ON time

PulseOnTimes Set Value	Description
0-255	Legal value
Other values	Invalid parameter

- **PulseOffTimes**   The pulse Off time

PulseOffTimes Set Value	Description
0-255	Legal value
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_CASH_DRAWER_OPEN	The cashdrawer opening failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.cashdrawerOpen(0,100,100);

● textSelectCharSetAndCodePage

Select an international character set and code page.

Method

- public int **textSelectCharSetAndCodePage**(int CharSet,int CodePage)

Parameter

- **CharSet**   International character set

CharSet Set Value	Description
CharacterSetUSA	U.S.A
CharacterSetFrance	France
CharacterSetGermany	Germany
CharacterSetUK	U.K
CharacterSetDenmark_I	Denmark I
CharacterSetSweden	Sweden
CharacterSetItaly	Italy
CharacterSetSpain_I	Spain I
CharacterSetJapan	Japan
CharacterSetNorway	Norway
CharacterSetDenmark_II	Denmark II
CharacterSetSpain_II	Spain II
CharacterSetLatin_America	Latin America
CharacterSetKorea	Korea
Other values	Invalid parameter

- **CodePage**   Code page

CodePage Set Value	Description
0	PC437
1	Katakana
2	PC850
3	PC860
4	PC863
5	PC865
16	WPC1252
17	PC866
18	PC852
19	PC858
12	PC857
13	771
14	Hebrew1

15	Hebrew2
21	Thai1
22	Thai2
23	Thai3
24	Thai4
25	Thai5
26	Thai6
27	Frasi
28	864[Arabic]
29	737[Greek]
32	1254[Turkish]
33	862[hebrew]
34	1251[Cyrillic]
35	1253[Greek]
36	855[Cyrillic]
37	774[Lithuanian]
38	928[Greek]
39	775[Baltic]
40	772[Lithuanian]
41	Hebrew3
42	851[Greek]
43	869[Greek]
44	1257[Baltic]
45	1250[Latin-2]
46	1255
47	1256[Arabic]
64	3840 (IBM-Russian)
65	3841 (Gost)
66	3843 (Polish)
67	3844 (CS2)
68	3845 (Hungarian)
69	3846 (Turkish)
70	3847 (Brazil-ABNT)
71	3848 (Brazil-ABICOMP)

72	1001 (Arabic)
73	2001 (Lithuanian-KBL)
74	3001 (Estonian-1)
75	3002 (Estonian-2)
76	3011 (Latvian-1)
77	3012 (Latvian-2)
78	3021 (Bulgarian)
79	3041 (Maltese)
80	8859
81	Persia

Remarks

a) Several kinds of printers may not support all code page types.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_SELECT_CHAR_SET	The char set selecting failed.
ERR_TEXT_SELECT_CODE_PAGE	The code page selecting failed.
ERR_PARAM	An invalid parameter was passed.

Example

```
String txtbuf = "123456ABCDabcd";
error_code = pos_sdk.textSelectCharSetAndCodePage(CharacterSetUSA,0);
error_code = pos_sdk.textPrint(txtbuf.getBytes(),txtbuf.getBytes().length);
```

● textSetLineHeight

Set line height.

Method

- public int **textSetLineHeight**(int Height)

Parameter

- **Height**   Line height

Height Set Value	Description
0-255	Legal value
Other values	Invalid parameter

Remarks

    a) If the parameter Height is 0, the printer's the height of one row will be set to default.

    b) If the parameter Height is less than the height of the character, the printer will set its the height of one row to the height of the character.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_SET_LINE_HEIGHT	The line height setting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textSetLineHeight(34);

● **textSetCharacterSpace**

    Set character spacing.

Method

    - public int **textSetCharacterSpace**(int LeftSpace,int RightSpace,int Mode)

Parameter

- **LeftSpace**   The character spacing for the right side of the character to inch (When mode = ChineseCharacterMode, LeftSpace must be legal.)

LeftSpaceSet Value	Description
0-255	Legal value
Other values	Invalid parameter

- **RightSpace**   The character spacing for the right side of the character to inch.

RightSpaceSet Value	Description
0-255	Legal value
Other values	Invalid parameter

- **Mode**   Character mode

ModeSet Value	Description
ChineseCharacterMode	Chinese character mode
EnglishCharacterMode	English character mode
Other values	Invalid parameter

Remarks

　　a) If the parameter Mode is specified ChineseCharacterMode, both the parameter LeftSpace and RightSpace must be legal. If the parameter Mode is specified EnglishCharacterMode, it's OK as long as the parameter RightSpace legal and LeftSpace will be ignored.

　　b) Only ChineseCharacterMode or EnglishCharacterMode can individually be changed by calling this method for one time.If you want to change both ChineseCharacterMode and EnglishCharacterMode,you should call this method for two times.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_SET_CHARACTER_SPACE	The character space setting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textSetCharacterSpace(10,50,EnglishCharacterMode);

● **textStandardModeAlignment**

　　Align all the data in one line to the specified position in standard mode.

Method

　　- public int **textStandardModeAlignment**(int Alignment)

Parameter

● **Alignment**　　Alignment mode of text

Alignment　　Set Value	Description
TextAlignmentLeft	Left justification
TextAlignmentCenter	Centering
TextAlignmentRight	Right justification
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.

ERR_TEXT_STANDARD_MODE_ALIGNMENT	The alignment mode selecting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textStandardModeAlignment(TextAlignmentLeft);

● textStandardModeUpsideDown

Turn on/off upside-down printing mode (This command sets the print position to the beginning of the line in standarad mode).

Method

- public int **textStandardModeUpsideDown**(int UpsideDown)

Parameter

• **UpsideDown** Upside-down printing mode is turned off or on.

UpsideDown Set Value	Description
1	Upside-down
0	Normal
Other	An invalid parameter was passed.

Remarks

a) Calling textStandardModeUpsideDown: FontStyleUpsideDown has the same print result as calling textStandardModeRotate: RotatePrint180.

b) When you calling textStandardModeUpsideDown: FontStyleUpsideDown first and then calling textStandardModeRotate: RotatePrint180,the result is as same as solely calling textStandardModeRotate: RotatePrint180. Reverse the call sequence, the result is unpredictable.

c) Suggestion not calling textStandardModeUpsideDown and textStandardModeRotate at the same.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_STANDARD_MODE_UPSIDEDOWN	Upside-down failed.
ERR_PARAM	An invalid parameter was passed.

Example

```
error_code = pos_sdk.textStandardModeUpsideDown(1);
```

● textPrint

Print text .

Method

- public int **textPrint**(byte[] Buffer,int BytesOfBuffer)

Parameter

- **Buffer**   The unsigned char data will be printed.
- **BytesOfBuffer**   The length of data

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_PRINT	The text printing failed.
ERR_PARAM	Buffer is null

Example

```
See also: Text printing
```

● textSelectFontMagnifyTimes

Select character size.

Method

- public int **textSelectFontMagnifyTimes**(int HorizontalTimes,int VerticalTimes)

Parameter

- **HorizontalTimes**   Horizontal times

HorizontalTimes Set Value	Description
1-6	The legal value of HorizontalTimes
Other values	Invalid Parameter

- **VerticalTimes**   Vertical times

VerticalTimes Set Value	Description
1-6	The legal value of VerticalTimes

Other values	Invalid Parameter

Remarks

    a) In standard mode, the vertical direction is the paper feed direction, and the horizontal direction is perpendicular to the paper feed direction. However, when character orientation changes in 90 °clockwise-rotation, the relationship between vertical and horizontal directions is reversed.

    b) In page mode, vertical and horizontal directions are based on the direction of print area.

    c) When characters in one line are enlarged to different sizes, all the characters are aligned at the baseline.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_SELECT_MAGNIFY_TIMES	Magnify times selecting was failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textSelectFontMagnifyTimes(2,3);

● **textStandardModeRotate**

    Rorate integer times 90 degree.

Method

- public int **textStandardModeRotate**(int Rotate)

Parameter

• **Rotate**   The degree of rotation

Rotate Set Value	Description
RotatePrintNormal	Nomal
RotatePrintR90	Turn 90 °clockwise rotation
RotatePrint180	180 °rotation
RotatePrintL90	Turn 90 °anticlockwise rotation
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_STANDARD_MODE_ROTATE	Roration setting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textStandardModeRotate(RotatePrintR90);// Turn 90 ° clockwise rotation

● **textSelectFont**

Select character font and font style.

Method

- public int **textSelectFont**(int FontType, int FontStyle)

Parameter

● **FontType**   Font type

FontType Set Value	Description
FontTypeStandardASCII	Standard ASCII
FontTypeCompressedASCII	Compressed ASCII
FontTypeUserDefined	User defined character
FontTypeChinese	Chinese character
Other values	Invalid parameter

● **FontStyle**   Font style

FontStyle Set Value	Description
FontStyleReverse	Reverse
FontStyleBold	Bold
FontStyleUpsideDown	UpsideDown
FontStyleUnderlineOneDotThick	One dot thick underline
FontStyleUnderlineTwoDotThick	Two dot thick underline

Remarks

a) The printer cannot underline when white/black reverse mode is enable.

b) The printer cannot underline characters which were clockwise rotated 90 or 270 degree.

c) The print result may be the same for several kinds of printers printing Chinese characters when the paramter FontStyle is specified FontStyleUnderlineOneDotThick

and FontStyleUnderlineTwoDotThick.

  d)You can set association of font type.Such as, setting the reverse and bold,the value is "FontStyleReverse| FontStyleBold".

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_SELECT_FONT_TYPE	The font type selecting failed.
ERR_TEXT_SET_FONT_STYLE_REVERSE	Reverse failed.
ERR_TEXT_SET_FONT_STYLE_BOLD	Bold failed.
ERR_TEXT_SET_FONT_STYLE_UNDERLINE	Underline failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.textSelectFont(FontTypeStandardASCII, FontStyleReverse| FontStyleBold);

● **textEnterOrQuitColorPrint**

  Enter/Quit bi-colour print mode.

Method

- public int **textEnterOrQuitColorPrint**(int ColorPrint)

- **ColorPrint** Font style

ColorPrint Set Value	Description
0	Quit bi-colour print mode
1	Enter bi-colour print mode
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_ENTER_QUIT_COLOR_PRINT	Failed to enter bi-color printing.
ERR_PARAM	An invalid parameter was passed.

Example

```
error_code = pos_sdk. textEnterOrQuitColorPrint(1);
```

● **textSetColorPrint**

Set the printting color.

Method

- public int **textSetColorPrint**(int Color)

• **Color** Font style

ColorPrint Set Value	Description
0	Select the color 1
1	Select the color 2
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_ENTER_QUIT_COLOR_PRINT	Failed to enter bi-color printing.
ERR_PARAM	An invalid parameter was passed.

Example

```
error_code = pos_sdk. textEnterOrQuitColorPrint(1);
```

● **textUserDefinedCharacterEnable**

User-defined character is enable/disable.

Method
   - public int **textUserDefinedCharacterEnable**(int Enable)

Parameter

• **Enable** Font user-defined enable or not

Enable Set Value	Description
FontUserDefinedDisable	Font User Defined disable
FontUserDefinedEnable	Font User Defined Enable
Other values	Invalid parameter

Remarks

a) All user-defined characters can be turn on/off.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_FONT_USER_DEFINED_ENABLE	User-defined enable failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: <u>Print user defined character</u>.

● **textUserDefinedCharacterDefine**

Define user-defined character.

Method

- public int **textUserDefinedCharacterDefine**(int BytesOfHeight,int DotsOfWidth, int StartingCode,int EndingCode, Bitmap []image)

Parameter

- **BytesOfHeight**   Bytes of height
- **DotsOfWidth**   Dots of width
- **StartingCode**   Starting char code
- **EndingCode**   Ending char code
- **CharacterData**   The data of user defined character downloaded

Legal value of every paramter:

Parameter	Legal Value
BytesOfHeight	3
DotsOfWidth	9 or 12
StartingCode	32-127
EndingCode	32-127

Remarks

a) The parameter BytesOfHeight must be 3.

b) The image of user defined characters must be 9*17 or 12*24.

c) The font style of white/black reverse,underline,character space,line

height,alignment mode and roratation can affect printing user defined characters.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_FONT_USER_DEFINED	User-defined character defining failed.
ERR_PARAM	Invalid parameter or Data does not match the number

Example

See also: Print user defined character.

● textUserDefinedCharacterCancel

Text cancel font user-defined of CharCode.

Method

- public int **textUserDefinedCharacterCancel**(int CharCode)

Parameter

- **CharCode**   The char code of cancel character

CharCode Set Value	Description
32-127	Legal value
Other values	Invalid parameter

Remarks

a) This method cancels the user defined character which specified by CharCode.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_TEXT_FONT_USER_DEFINED_CANCEL	User-defined character canceling failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: Print user defined character.

● imageCreateRasterBitmap

Create raster bitmap.

Method

   - public Bitmap **imageCreateRasterBitmap**(String printText, int textSize, int bold)

Parameter

- **PrintText**   The character which would be printed.
- **textSize**   The canvas size.

bold Set Value	Description
0-200	Legal value
Other values	Invalid parameter

- **bold**   The degree of thickness.

bold Set Value	Description
1-5	Legal value
Other values	Invalid parameter

Return

Return Value	Description
The project of Bitmap class	Processing was successful.
null	Failed.

Example

See also: Text raster printing.

● imageStandardModePrint

Image print (standard mode).

Method

- public int **imageStandardModePrint**(Bitmap image,int SingleDoubleFlag,int StartHorPos,int PrinterWidth)

Parameter

- **Image**   The image would be print.
- **SingleDoubleFlag**   Image print mode

SingleDoubleFlag Set Value	Description
SingleDensity_8	8-dot single-density
DoubleDensity_8	8-dot double-density
SingleDensity_24	24-dot single-density
DoubleDensity_24	24-dot double-density
Other values	Invalid parameter

- **StartHorPos**   The starting position.
- **PrinterWidth**   Printer width

PrinterWidth Set Value	Description
0	The image can not be zoom
64-65535	The legal value of printer width,and the image be normally zoom according to printer width
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_DOWNLOAD_AND_PRINT	Image downloading and printing failed.
ERR_PARAM	Invalid parameter was passed or the image data was null

Example

See also: image print.

● **imageDownloadToPrinterRAM**

Download images to RAM.

Method

- public int **imageDownloadToPrinterRAM**(int ImageID, Bitmap image,int PrinterWidth)

Parameter

- **ImageID**   The ID of download

ImageID Set Value	Description
0-7	Legal value

Other values	Invalid parameter

- **Image**　　　　　The image would be Download.
- **PrinterWidth**　　Printer width

PrinterWidth　Set Value	Description
0	The image can not be zoom
64-65535	The legal value of printer width,and the image be normally zoom according to printer width
Other values	Invalid parameter

Remarks

a) You can only download a sigle image and specify the ImageID.

b) Downloaded bitmaps will be cleared when turning power off.

Return

Return Value	Description
POS_SUCCESS	Processing was successful
ERR_IMAGE_DOWNLOAD_RAM	RAM images downloading failed
ERR_PROCESSING	Scaling the image failed
ERR_PARAM	Invalid parameter was passed.

Example

See also: RAM/Flash image download and print.

● imageRAMPrint

Print images which have been downloaded to RAM.

Method

- public int **imageRAMPrint**(int ImageID,int Mode)

Parameter

- **ImageID**　　　The ID of image which has been downloaded to RAM.

ImageID　Set Value	Description
0-7	Legal value
Other values	Invalid parameter

- **Mode**　　　Print mode

Mode　Set Value	Description

NormalMode	Normal size
Double_width	Double width
Double_height	Double height
Quadruple	Quadruple(both width and height was double)
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_RAM_PRINT	RAM image printing failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: RAM/Flash image download and print.

● imageDownloadToPrinterFlash

Download images to Flash.

Method

- public int **imageDownloadToPrinterFlash**(int image_num, Bitmap image[],int PrinterWidth)

Parameter

- **image_num**   The number of images which would be downloaded to Flash.
- **image**   The image data.
- **PrinterWidth**   Printer width.

PrinterWidth Set Value	Description
0	The image can not be zoom
64-65535	The legal value of printer width,and the image be normally zoom according to printer width
Other values	Invalid parameter

Remarks

a) Every downloading will clear the bitmap lastly downloaded into Flash.

b) When multi-bitmaps would be downloaded, if one is not successful, remove images have been downloaded.

c) The height of the downloaded image is not larger than 2040.

d) Downloaded bitmaps will not be cleared when turning power off.

e) Multi-bitmaps must be compared by "@" such as

"SNBC.bmp@Jpg.jpg@face.PNG".

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_DOWNLOAD_FLASH	Flash images downloading failed.
ERR_PROCESSING	Scaling the image failed
ERR_PARAM	1. The ImageArray was null. 2. The height of downloaded image is larger than 2040. 3. The image data was null

Example

See also: RAM/Flash image download and print.

● **imageFlashPrint**

Print images which have been downloaded to Flash.

Method

- public int **imageFlashPrint**(int ImageID,int Mode)

Parameter

- **ImageID**   The ID of image which has been downloaded to Flash.

ImageID Set Value	Description
1-255	Legal value
Other values	Invalid parameter

- **Mode**   Print mode

Mode Set Value	Description
NormalMode	Normal size
Double_width	Double width
Double_height	Double height
Quadruple	Quadruple(both width and height was double)
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_FLASH_PRINT	Flash image printing failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: RAM/Flash image download and print.

● imageStandardModeRasterPrint

Print raster image in standard mode. In preparing the data sending 1B 40 commands to grating print,but this commands not affect other operations.

Method

- public int **imageStandardModeRasterPrint**(Bitmap image, int PrinterWidth)

Parameter

- **Image** The image would be print.
- **PrinterWidth** Printer width

PrinterWidth Set Value	Description
0	The image can not be zoom
64-2040	The legal value of printer width,and the image be normally zoom according to printer width
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_STANDARD_MODE_RASTER_PRINT	The image raster printing failed.
ERR_PROCESSING	Scaling the image failed.
ERR_PARAM	The image was null.

Example

See also: Raster image print.

● imageCompressedRasterPrint

Print compressed image in standard mode.

71

Method

    - public int **imageCompressedRasterPrint** (Bitmap image, int PrinterWidth)

Parameter

- **Image**   The image would be print.

- **PrinterWidth**   Printer width

PrinterWidth Set Value	Description
0	The image can not be zoom
64-2040	The legal value of printer width,and the image be normally zoom according to printer width
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_STANDARD_MODE_RASTER_PRINT	The image raster printing failed.
ERR_PROCESSING	Scaling the image failed.
ERR_PARAM	The image was null.

Example

```
error_code = pos_sdk. imageCompressedRasterPrint(image,640);
```

● **imageStandardModeGrayPrint**

    Print gray image in standard mode.

Method

    - public int imageStandardModeGrayPrint (Bitmap image, int PrinterWidth)

Parameter

- **Image**   The image would be print.

- **PrinterWidth**   Printer width

PrinterWidth Set Value	Description
0	The image can not be zoom
64-2040	The legal value of printer width,and the image be

	normally zoom according to printer width
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_IMAGE_STANDARD_MODE_GRAY_PRINT	The image gray printing failed.
ERR_PROCESSING	Scaling the image failed.
ERR_PARAM	The image was null.

Example

```
error_code = pos_sdk.imageStandardModeGrayPrint(image,640);
```

● barcodePrint1Dimension

Print 1Dimension barcode.

Method

- public int **barcodePrint1Dimension**(String pszBuffer,int DataLength,int nType, int nWidthX,int nHeight, int nHriFontType, int nHriFontPosition)

Parameter

- **pszBuffer**   Barcode data
- **DataLength**   The data length of barcode data
- **nType**   Barcode type

nType Set Value	Description	Data length Set Value
BarcodeUPC_A	UPC-A	11 -12
BarcodeUPC_E	UPC-E	11-12
BarcodeJAN13orEAN13	EAN13	12-13
BarcodeJAN8orEAN8	EAN-8	7-8
BarcodeCODE39	Code39	1-255
BarcodeITF	Interleaved 2 of 5	1-255
BarcodeCODABAR	CodaBar	1-255
BarcodeCODE93	Code93	1-255
BarcodeCODE128	Code128	2-255

Other values	Invalid parameter	

- **nWidthX**　　Barcode module width

nWidthX　Set Value	Description
2-6	Legal value
Other values	Invalid parameter

- **nHeight**　　Barcode height

nHeight Set Value	Description
1-255	Legal value
Other values	Invalid parameter

- **nHriFontType**　　Hri font type

nHriFontType　Set Value	Description
FontTypeStandardASCII	Standard ASCII
FontTypeCompressedASCII	Compressed ASCII
Other values	Invalid parameter

- **nHriFontPosition**　　The position of Hri font

nHriFontPosition　Set Value	Description
HRINone	HRI can not be printed
HRIAbove	Above the barcode.
HRIBelow	Below the barcode.
HRIAboveAndBelow	Both above and below the barcode.
Other values	Invalid parameter

Remarks

a) See also: Appendix C. Barcode and Appendix D. Code 128

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_BARCODE_PRINT_1D	Barcode type selecting failed.
ERR_BARCODE_SELECT_MODULE_WIDTH	Module width selecting failed
ERR_BARCODE_SELECT_BARCODE_HEIGHT	Barcode height selecting failed
ERR_BARCODE_SELECT_HRI_FONT_TYPE	Hri font type selecting failed

ERR_BARCODE_SELECT_HRI_FONT_POSITION	Hri position selecting failed
ERR_PARAM	Invalid parameter

Example

```
String str_data = "123456789012"
int data_lengt = str_data.getBytes().length;
error_code = pos_sdk barcodePrint1Dimension(str_data, data_lengt,BarcodeUPC_A,3,100,0,1);
```

● **barcodePrintQR**

Set parameter and print barcode QR.

Method

- public int **barcodePrintQR**(String pszBuffer,int DataLength,int nOrgx,int BasicElementWidth,int SymbolType,int LanguageMode)

Parameter

- **pszBuffer**   BarcodeQR data
- **DataLength**   The length of QR data
- **nOrgx**   The starting position
- **BasicElementWidth**   Basic element width

BasicElementWidth Set Value	Description
1-255	Legal value
Other values	Invalid parameter

- **SymbolType**   Symbole type

SymbolType Set Value	Description
OriginalType	Original type
EnhancedType	Enhance type(This type is suggested to be used)
Other values	Invalid parameter

- **LanguageMode**   Language mode

LanguageMode Set Value	Description
LanguageChinese	Chinese
LanguageJapanese	Japanese
Other values	Invalid parameter

Remarks

a) When the barcode outside the print area by setting barcode data and basic

element width, the printer can not barcode.

    b) Recommended to use EnhancedType.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_BARCODE_QR_SET_PARAM	The parameters of QR setting failed.
ERR_BARCODE_PRINT_2D	Barcode 2D type selecting to print failed.
ERR_PROCESSING	Barcode data of QR sending failed.
ERR_PARAM	Invalid parameter

Example

See also: Print QR.

● barcodePrintPDF417

Set barcodePDF417 size and print PDF417.

Method

- public int **barcodePrintPDF417**(String pszBuffer,int DataLength,

    int AppearanceToHeight,int AppearanceToWidth,int RowsNumber,

int ColumnsNumber,int Xsize, int LineHeight,int nCorrectGrade)

Parameter

- **pszBuffer**   Barcode PDF417 data
- **DataLength**   The length of PDF417 data
- **AppearanceToHeight**   Appearance to height
- **AppearanceToWidth**   Appearance to width
- **RowNumber**   The number of rows
- **ColumnNumber**   The number of columns
- **XSize**   XSize
- **LineHeight**   Line height
- **CorrectionGrade**   Correction grade

The legal values of every paramter, as follow:

Parameter	Legal value
AppearanceToHeight	1-10
AppearanceToWidth	1-100

RowNumber	3-90
ColumnNumber	1-30
XSize	1-7
LineHeight	2-25
CorrectionGrade	0-8

Remarks

a) When the barcode data outside range, the barcode can not be printed.

b) When the barcode size outside the print area, the printer can not barcode.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_BARCODE_PDF417_SET_SIZE	PDF417 size setting failed.
ERR_BARCODE_PDF417_SELECT_CORRECTION_GRADE	PDF417 correction grade selecting failed
ERR_BARCODE_PRINT_2D	Barcode 2D type selecting to print failed.
ERR_PROCESSING	Barcode data of PDF417 sending failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: Print PDF417.

● **barcodePrintMaxicode**

Print barcode Maxicode.

Method

- public int **barcodePrintMaxicode**(String pszBuffer, int DataLength)

Parameter

- **Data**   Maxicode data
- **DataLength**   The length of Maxicode data

Remarks

a) When the barcode data outside range, the barcode can not be printed.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_BARCODE_PRINT_2D	Barcode 2D type selecting to print failed.
ERR_PROCESSING	Barcode data of Maxicode sending failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: Print Maxicode.

● barcodePrintGS1DataBar

Set GS1 DataBar and GS1 composite barcode parameter and print barcode, Whether GS1 barcode is separate or composite barcode is distinguished by data

separator"|" If there is "|".in the programmed data, it is composite barcode; otherwise,

it is separate DataBar..

Method

- public int **barcodePrintGS1DataBar**(String pszBuffer,int DataLength,int BarcodeType,int BasicElementWidth,int BarcodeHeight,int BasicElementHeight,int SeparatorHeight,int SegmentHeight,int HRI,int AI)

Parameter

- **Data**   Barcode data
- **DataLength**   The length of GS1 barcode data
- **BarcodeType**   Barcode type

BarcodeType Set Value	Description
GS1DataBarOmnidirectional	GS1DataBar Omnidirectional
GS1DataBarTruncated	GS1DataBar Truncated
GS1DataBarStacked	GS1 DataBar Stacked
GS1DataBarStackedOmnidirectional	GS1 DataBar Stacked Omnidirectiona
GS1DataBarLimited	GS1 DataBar Limited
GS1DataBarExpanded	GS1 DataBar Expanded
GS1DataBarExpandedStacked	GS1 DataBar ExpandedStacked
Other values	Invalid parameter

- **BasicElementWidth**   Basic element width

BasicElementWidth Set Value	Description
1-6	Legal value
Other values	Invalid parameter

- **BarcodeHeight**   The height of the DataBar, Stacked, stacked omnidirectional, expanded stacked barcode indicate the height of each line of barcode.

BarcodeHeight Set Value	Description
2-250	Legal value
Other values	Invalid parameter

- **BasicElementHeight**   The basic element height of the 2D barcode in the composite barcode

BasicElementHeight Set Value	Description
1-10	Legal value
Other values	Invalid parameter

- **SeparatorHeight**   The height of the separator. This parameter should be set in DataBar composite barcode or separate stacked, stacked omnidirectional, expanded stacked barcodes.

SeparatorHeight Set Value	Description
1-10	Legal value
Other values	Invalid parameter

- **SegmentNumber**   The number of segments of each line of barcode. Only in expanded stacked barcode should this parameter be set.

SegmentNumber Set Value	Description
2-20	The legal value of separate expanded stacked barcodes
4-20	The legal value of composite expanded stacked barcodes
Other values	Invalid parameter

- **HRI**   The content of the note character

HRI Set Value	Description
DataBarAnd2DHri	DataBar and 2D in composite barcode DataBar only in separate barcode
Only DataBarHri	Print DataBar in composite or separate barcode

Only2DHri	Print 2D in composite barcode, no print in separate barcode
NoHri	No note character
Other values	Invalid parameter

- **AI**   whether to use AI (use identifier): 0 indicates to not use AI; 1 indicates to use AI.

Remarks

    a) Several kinds of printers may not support all GS1 barcode type.

    b) When the barcode data outside range, the barcode can not be printed.

    c) When the barcode size outside the print area, the printer can not barcode.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_BARCODE_GS1DATABAR_SET_PARAM	Parameters of GS1 setting failed.
ERR_BARCODE_PRINT_2D	Barcode 2D type selecting to print failed.
ERR_PROCESSING	Barcode data of GS1 sending failed.
ERR_PARAM	An invalid parameter was passed.

Example

See also: Print GS1 DataBar and GS1 composite barcode.

● standardModeSetPrintAreaWidth

    Set left margin and width in standard mode (This command sets the print position to the beginning of the line).

Method

    - public int **standardModeSetPrintAreaWidth**(int LeftMargin,int Width)

Parameter

- **LeftMargin**   Left margin

LeftMargin Set Value	Description

0-65535	Legal value
Other values	Invalid parameter

- **Width**　　　　Print area width

Width　Set Value	Description
0-65535	Legal value
Other values	Invalid parameter

Remarks

a) This method can not affect printing user defined characters, the method of imageStandardModeRasterPrint and textStandardModeRasterPrint.

b) For page mode, this method is invalid.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_STANDARD_MODE_SET_PRINT AREA_WIDTH	Print area width for standard mode setting failed.
ERR_STANDARD_MODE_SET_LEFT_ MARGIN	Left margin for standard mode setting failed.
ERR_PARAM	An invalid parameter was passed.

Example

eror_code = pos_sdk. standardModeSetPrintAreaWidth (100,500);

● **standardModeSetStartingPosition**

Horizontal Starting Position in standard mode.

Method

- public int **standardModeSetStartingPosition**(int X)

Parameter

- **X**　　　Horizontal starting position for Standard mode

Distance　Set Value	Description
0-65535	Legal value
Other values	Invalid parameter

Remarks

a) This method can not affect user defined characters printing, image raster printing and text raster printing.

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_STANDARD_MODE_SET_HORIZONT AL_STARTING_POSITION	Horizontal starting position for standard mode setting failed.
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk. standardModeSetStartingPosition (100);

● **pageModeSetStartingPosition**

Set horizontal and vertical starting position in page mode.

Method

- public int **pageModeSetStartingPosition**(int X,int Y)

Parameter

- **X**   Horizontal starting position

X Set Value	Description
0-65535	Legal value
Other values	Invalid parameter

- **Y**   Vertical starting position

Y Set Value	Description
0-65535	Legal value
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_STANDARD_MODE_SET_HORIZONTA L_STARTING_POSITION	Horizontal starting position for standard mode setting failed
ERR_PAGE_MODE_SET_VERTICAL_STARTI NG_POSITION	Vertical starting position for page mode setting failed

ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.pageModeSetStartingPosition(203,203);

● pageModeSetPrintArea

Set print area in page mode.

Method

- public int **pageModeSetPrintArea**(int X,int Y, int AreaWidth,int AreaHeight,int Direction)

Parameter

- **X**   Horizontal starting position
- **Y**   Vertical starting position
- **AreaWidth**   Area width
- **AreaHeight**   Area height

X/Y/AreaWidth/AreaHeight Set Value	Description
0-65535	Legal value
Other values	Invalid parameter

- **Direction**   Print direction

Direction Set Value	Description
LeftToRight	Left to right
BottomToTop	Bottom to top
RightToLeft	Right to left
TopToBottom	Top to bottom
Other values	Invalid parameter

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PAGE_MODE_SET_PRINT_AREA	Print area for page mode setting failed
ERR_PAGE_MODE_SET_PRINT_DIRECTION	Print direction for page mode setting failed
ERR_PARAM	An invalid parameter was passed.

Example

error_code = pos_sdk.pageModeSetPrintArea(100,0,400,1000,LeftToRight);

● **pageModePrint**

Print data in page mode.

Method

- public int **pageModePrint**()

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PAGE_MODE_PRINT	page mode printing failed.

Example

error_code = pos_sdk.pageModePrint();

● **pageModeClearBuffer**

In page mode, delete all the print data in current area.

Method

- public int **pageModeClearBuffer**()

Return

Return Value	Description
POS_SUCCESS	Processing was successful.
ERR_PAGE_MODE_CLEAR_BUFFER	Page mode clearing buffer failed.

Example

error_code = pos_sdk.pageModeClearBuffer();

5. Appendix

Appendix A. List of Error Code

Error Code	Description
POS_SUCCESS	Processing was successful.
ERR_PARAM	Invalid parameter was passed.
ERR_SYSTEM_RESET	Printer resetting failed.
ERR_SYSTEM_SELECT_PRINT_MODE	The print mode selecting failed.
ERR_SYSTEM_SELECT_PAPER_TYPE	The paper type selecting failed.
ERR_SYSTEM_SET_MOTION_UNIT	The motion uint selecting failed.
ERR_SYSTEM_QUERY_STATUS	The printer's status querying failed.
ERR_SYSTEM_FEED_LINE	Feed line failed.
ERR_SYSTEM_CUT_PAPER	The paper cutting failed.
ERR_CASH_DRAWER_OPEN	The cashdrawer opening failed.
ERR_TEXT_SELECT_CHAR_SET	The char set selecting failed.
ERR_TEXT_SELECT_CODE_PAGE	The code page selecting failed.
ERR_TEXT_SET_LINE_HEIGHT	The line height setting failed.
ERR_TEXT_SET_CHARACTER_SPACE	The character space setting failed.
ERR_TEXT_STANDARD_MODE_ALIGNMENT	The alignment mode selecting failed.
ERR_TEXT_SELECT_FONT_TYPE	The font type selecting failed.
ERR_TEXT_SET_FONT_STYLE_REVERSE	Reverse failed.
ERR_TEXT_SET_FONT_STYLE_BOLD	Bold failed.
ERR_TEXT_SET_FONT_STYLE_UNDERLINE	Underline failed.
ERR_TEXT_STANDARD_MODE_UPSIDEDOWN	Upside-down failed.
ERR_TEXT_SELECT_MAGNIFY_TIMES	Magnify times selecting was failed.
ERR_TEXT_STANDARD_MODE_ROTATE	Roration setting failed.
ERR_TEXT_ENTER_QUIT_COLOR_PRINT	Failed to enter /cancel bi-color printing.
ERR_TEXT_SET_COLOR_PRINT	Color setting failed.
ERR_TEXT_FONT_USER_DEFINED_ENA	User-defined enable failed.

BLE	
ERR_TEXT_FONT_USER_DEFINED	User-defined character defining failed.
ERR_TEXT_FONT_USER_DEFINED_CANCEL	User-defined character canceling failed.
ERR_TEXT_PRINT	The text printing failed.
ERR_IMAGE_DOWNLOAD_AND_PRINT	Image downloading and printing failed.
ERR_IMAGE_DOWNLOAD_RAM	RAM images downloading failed.
ERR_IMAGE_RAM_PRINT	RAM image printing failed.
ERR_IMAGE_DOWNLOAD_FLASH	Flash images downloading failed.
ERR_IMAGE_FLASH_PRINT	Flash image printing failed.
ERR_IMAGE_STANDARD_MODE_RASTER_PRINT	The image raster printing failed.
ERR_IMAGE_STANDARD_MODE_GRAY_PRINT	The image gray printing failed.
ERR_STANDARD_MODE_SET_PRINTAREA_WIDTH	Print area width for standard mode setting failed
ERR_STANDARD_MODE_SET_LEFT_MARGIN	Left margin for standard mode setting failed
ERR_STANDARD_MODE_SET_HORIZONTAL_STARTING_POSITION	Horizontal starting position for standard mode setting failed.
ERR_PAGE_MODE_SET_VERTICAL_STARTING_POSITION	Vertical starting position for page mode setting failed
ERR_PAGE_MODE_SET_PRINT_AREA	Print area for page mode setting failed
ERR_PAGE_MODE_SET_PRINT_DIRECTION	Print direction for page mode setting failed
ERR_PAGE_MODE_PRINT	Page mode printing failed.
ERR_PAGE_MODE_CLEAR_BUFFER	Page mode clearing buffer failed.
ERR_BARCODE_PRINT_1D	Barcode type selecting failed.
ERR_BARCODE_PRINT_2D	Barcode 2D type selecting to print failed.
ERR_BARCODE_SELECT_MODULE_WID	Module width selecting failed.

TH	
ERR_BARCODE_SELECT_BARCODE_HEIGHT	Barcode height selecting failed.
ERR_BARCODE_SELECT_HRI_FONT_TYPE	Hri font type selecting failed.
ERR_BARCODE_SELECT_HRI_FONT_POSITION	Hri position selecting failed.
ERR_BARCODE_QR_SET_PARAM	The parameters of QR setting failed.
ERR_BARCODE_PDF417_SELECT_CORRECTION_GRADE	PDF417 correction grade selecting failed
ERR_BARCODE_PDF417_SET_SIZE	PDF417 size setting failed.
ERR_BARCODE_GS1DATABAR_SET_PARAM	Parameters of GS1 setting failed.

Appendix B. Barcode

The data length and character set of barcode type,as follows:

Barcode Type	Data length	ASCII	Remarks
UPC-A	11 ~ 12	48 ~ 57	
UPC-E	11 ~ 12	48 ~ 57	The first character must be 0.
JAN13 (EAN13)	12 ~ 13	48 ~ 57	
JAN 8 (EAN8)	7 ~ 8	48 ~ 57	
CODE39	1 ~ 255	45 ~ 57, 65 ~ 90, 32, 36, 37,43	
ITF	1 ~ 255	48 ~ 57	
CODABAR	1 ~ 255	48 ~ 57 65 ~ 68, 36, 43,45,46,47 58	The beginning code and ending code must be one of character A, B, C, D. The ending code can be replaced by T, E, *,N.
CODE93	1 ~ 255	0 ~ 127	
CODE128	2 ~ 255	0 ~ 127	You must specify the

			character ser before barcode data.
PDF417	1 ~ 255	0 ~ 255	
QRCODE	4 ~ 255	0 ~ 255	
MAXICODE	1 ~ 138	48 ~ 57,65 ~ 90	
GS1	1 ~ 255	It depands on GS1 barcode type,see also: the following Table.	

The data length and character set of GS1, as follows:

Parameter	Barcode type	Character set	Data length
1	GS1DataBar Omnidirectional	Number 0-9	14bits, 13numbers+1bits of check characters
2	GS1DataBar Truncated	Number 0-9	14bits, 13numbers+1bits of check characters
3	GS1 DataBar Stacked	Number 0-9	14bits, 13numbers+1bits of check characters
4	GS1 DataBar Stacked Omnidirectional	Number 0-9	14bits, 13numbers+1bits of check characters
5	GS1 DataBar Limited	Number 0-9	14bits, 13numbers+1bits of check characters
6	GS1 DataBar Expanded	0 ~ 9,A ~ Z, a ~ z ! " % & ' ( )* + , - ./ : ; < = > ? _ space FNC1	Max 74numbers or 41 letters
7	GS1 DataBar ExpandedStacked	0 ~ 9,A ~ Z, a ~ z ! " % & ' ( ) * + , - ./ : ; < = > ? _ space FNC1	Max 74numbers or 41 letters

[Notes]

When UPC-A, UPC-E, JAN13 (EAN13) or JAN8 (EAN8) is selected, if n is outside the specified range, this command is invalid

[Notes (standard mode)]

• If data is outside the specified range, the barcode can not be printed.

• If the horizontal size of the barcode exceeds printing area, the barcode can not be printed.

• This command feeds as much paper as is required to print the barcode, regardless of the line height specified by textSetLineHeight:.

• It is enabled only when no data exists in the print buffer. When data exists in the print buffer, the command is ignored.

• After printing barcode, this command sets the print position to the beginning of the line.

• This command is not affected by textSelectFont: (FontStyle as FontStyleReverse/FontStyleBold/FontStyleUnderlineOneDotThick/FontStyleUnderlineTwoDotThick, etc.), except for FontStyleUpsideDown.

[Notes in page mode]

• This command develops bar code data in the print buffer, but does not print it. After processing barcode data, this command moves the print position to the right side dot of the barcode.

• If d is out of the specified range, this command is ignored.

• If barcode width exceeds the printing area, this command is ignored.

When CODE128 (m = 73) is used:

• See also: Appendix A for the information of the CODE 128 barcode and the character set.

• When using the CODE 128 in this printer, take the following points into account for data transmission:

Character set must be selected before the barcode data (one of CODE A, CODE B or CODE C).

Special characters are defined by combining two characters "{" and one character. The ASCII character "{" is defined by transmitting "{" twice consecutively.

Specific character set	Transmit data
SHIFT	{S
CODE A	{A
CODE B	{B

CODE C	{C
FNC1	{1
FNC2	{2
FNC3	{3
FNC4	{4
"{"	{{

Example: print "123456" using CODE B, You can input: {B123456

• If the top of the barcode data is not the code set selection character, the printer stops command processing and processes the following data as normal data.

• If combination of "{" and the following character does not apply to any special character, the printer stops command processing and processes the following data as normal data.

• If the printer receives characters that cannot be used in the special code set, the printer stops command processing and processes the following data as normal data.

• The printer does not print HRI characters that correspond to the shift characters or code set selection characters.

• HRI characters for the function characters are not printed.

• HRI characters for the control characters (<00>H to <1F>H and <7F>H) are not printed.

• The left-side and right –side spacing which varies from one barcode type to another must be assured.

Appendix C. Code 128

1.   Description of the CODE128 Bar Code

In CODE128 bar code system, it is possible to represent 128 ASCII characters, the one hundred numbers from 00 to 99 and some special characters with three code sets: A, B and C. Each code set is used for representing the following characters:

• Code set A: ASCII characters 00H to 5FH

• Code set B: ASCII characters 20H to 7FH

• Code set C: 100 numerals from 00 to 99

The following special characters are also available in CODE128:

• SHIFT characters

In code set A, the character just after SHIFT is processed as a character for code set B. In code set B, the character just after SHIFT is processed as a character for code set A. SHIFT characters cannot be used in code set C.

• Code set selection character (CODE A, CODE B, CODE C).

This character switches the following code set to code set A, B, or C.

• Function character (FNC1, FNC2, FNC3, FNC4)

The usage of function characters depends on the application software. In code set C, only FNC1 is available.

2. Code Tables

Printable characters in code set A

Character	Transmit Data		Character	Transmit Data		Character	Transmit Data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
NULL	00	0	(	28	40	P	50	80
SOH	01	1	)	29	41	Q	51	81
STX	02	2	*	2A	42	R	52	82
ETX	03	3	+	2B	43	S	53	83
EOT	04	4	,	2C	44	T	54	84
ENQ	05	5	-	2D	45	U	55	85
ACK	06	6	.	2E	46	V	56	86
BEL	07	7	/	2F	47	W	57	87
BS	08	8	0	30	48	X	58	88
HT	09	9	1	31	49	Y	59	89
LF	0A	10	2	32	50	Z	5A	90
VT	0B	11	3	33	51	[	5B	91
FF	0C	12	4	34	52	\	5C	92
CR	0D	13	5	35	53	]	5D	93
SO	0E	14	6	36	54	^	5E	94
SI	0F	15	7	37	55	_	5F	95
DLE	10	16	8	38	56	FNC1	7B,31	123,49
	11	17	9	39	57	FNC2	7B,32	123,50
	12	18	:	3A	58	FNC3	7B,33	123,51
	13	19	;	3B	59	FNC4	7B,34	123,52
	14	20	<	3C	60	SHIFT	7B,53	123,83
	15	21	=	3D	61	CODEB	7B,42	123,66
	16	22	>	3E	62	CODEC	7B,43	123,67
	17	23	?	3F	63			
	18	24	@	40	64			
	19	25	A	41	65			
	1A	26	B	42	66			
	1B	27	C	43	67			
	1C	28	D	44	68			
	1D	29	E	45	69			
	1E	30	F	46	70			
	1F	31	G	47	71			
NAK	20	32	H	48	72			

Character	Hex	Decimal	Character	Hex	Decimal	Character	Hex	Decimal
SYN	21	33	I	49	73			
ETB	22	34	J	4A	74			
CAN	23	35	K	4B	75			
EM	24	36	L	4C	76			
SUB	25	37	M	4D	77			
ESC	26	38	N	4E	78			
FS	27	39	O	4F	79			
GS								
RS								
US								
SP								
!								
"								
#								
$								
%								
&								
'								

Printable characters in code set B

Character	Transmit Data		Character	Transmit Data		Character	Transmit Data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
SP	20	32	H	48	72	p	70	112
!	21	33	I	49	73	q	71	113
"	22	34	J	4A	74	r	72	114
#	23	35	K	4B	75	s	73	115
$	24	36	L	4C	76	t	74	116
%	25	37	M	4D	77	u	75	117
&	26	38	N	4E	78	v	76	118
'	27	39	O	4F	79	w	77	119
(	28	40	P	50	80	x	78	120
)	29	41	Q	51	81	y	79	121
*	2A	42	R	52	82	z	7A	122
+	2B	43	S	53	83	{	7B,7B	123,12
,	2C	44	T	54	84	|}	7C	3
-	2D	45	U	55	85	—	7D	124
.	2E	46	V	56	86	DEL	7E	125

Character	Hex	Decimal	Character	Hex	Decimal	Character	Hex	Decimal
/	2F	47	W	57	87	FNC1	7F	126
0	30	48	X	58	88	FNC2	7B,31	127
1	31	49	Y	59	89	FNC3	7B,32	123,49
2	32	50	Z	5A	90	FNC4	7B,33	123,50
3	33	51	[	5B	91	SHIFT	7B,34	123,51
4	34	52	\	5C	92	CODEA	7B,53	123,52
5	35	53	]	5D	93	CODEC	7B,41	123,83
6	36	54	^	5E	94		7B,43	123,65
7	37	55	_	5F	95			123,67
8	38	56	`	60	96			
9	39	57	a	61	97			
:	3A	58	b	62	98			
;	3B	59	c	63	99			
<	3C	60	d	64	100			
=	3D	61	e	65	101			
>	3E	62	f	66	102			
?	3F	63	g	67	103			
@	40	64	H	68	104			
A	41	65	i	69	105			
B	42	66	j	6A	106			
C	43	67	k	6B	107			
D	44	68	l	6C	108			
E	45	69	m	6D	109			
F	46	70	n	6E	110			
G	47	71	o	6F	111			

Printable characters in code set C

Character	Transmit Data		Character	Transmit Data		Character	Transmit Data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
00	00	0	40	28	40	80	50	80
01	01	1	41	29	41	81	51	81
02	02	2	42	2A	42	82	52	82
03	03	3	43	2B	43	83	53	83
04	04	4	44	2C	44	84	54	84
05	05	5	45	2D	45	85	55	85
06	06	6	46	2E	46	86	56	86
07	07	7	47	2F	47	87	57	87

08	08	8	48	30	48	88	58	88
09	09	9	49	31	49	89	59	89
10	0A	10	50	32	50	90	5A	90
11	0B	11	51	33	51	91	5B	91
12	0C	12	52	34	52	92	5C	92
13	0D	13	53	35	53	93	5D	93
14	0E	14	54	36	54	94	5E	94
15	0F	15	55	37	55	95	5F	95
16	10	16	56	38	56	96	60	96
17	11	17	57	39	57	97	61	97
18	12	18	58	3A	58	98	62	98
19	13	19	59	3B	59	99	63	99
20	14	20	60	3C	60	FNC1	7B,31	123,49
21	15	21	61	3D	61	CODEA	7B,41	123,65
22	16	22	62	3E	62	CODEB	7B,42	123,66
23	17	23	63	3F	63			
24	18	24	64	40	64			
25	19	25	65	41	65			
26	1A	26	66	42	66			
27	1B	27	67	43	67			
28	1C	28	68	44	68			
29	1D	29	69	45	69			
30	1E	30	70	46	70			
31	1F	31	71	47	71			
32	20	32	72	48	72			
33	21	33	73	49	73			
34	22	34	74	4A	74			
35	23	35	75	4B	75			
36	24	36	76	4C	76			
37	25	37	77	4D	77			
38	26	38	78	4E	78			
39	27	39	79	4F	79			

Appendix D. Programming Flow