

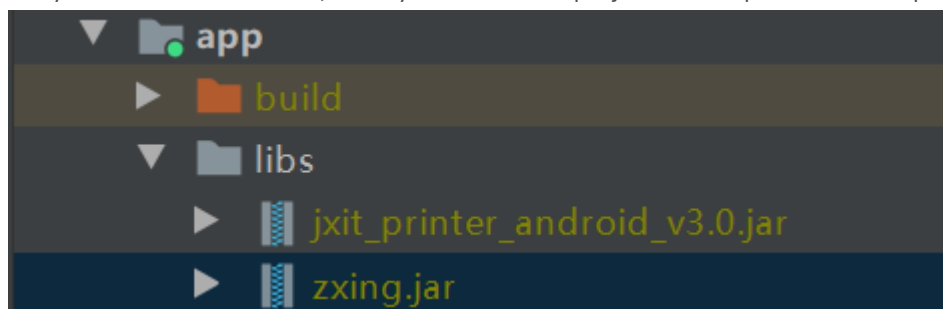
# Android C30 SDK Standard Development ...

## 1. SDK project integration configuration

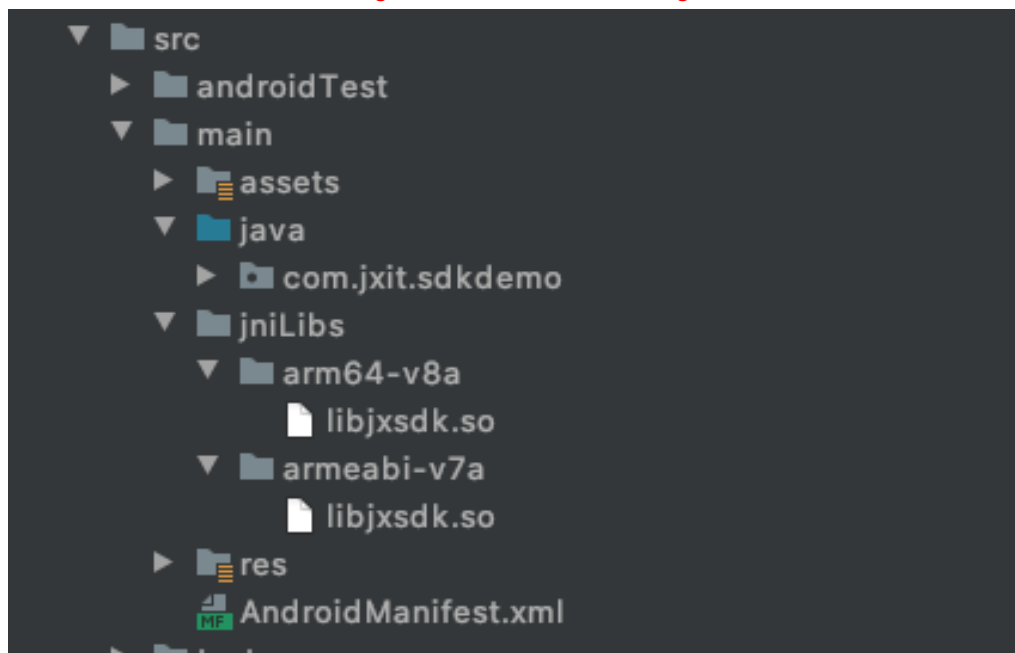
You can use the jixin Android SDK in your application in the following steps:

### 1.1. Integration SDK

1.1.1. jxit\_printer\_android\_v3.0.jar, zxing.jar (zxing.jar integration is theoretically not required without global SDK) (refer to demo for the actual SDK) , copy it to the project directory under the Libs Folder, and synchronize the project to compile it into the project.



1.1.2. Copy the JNLIBS folder to the SRC/main/directory, the same level as the Java folder. jinLibs provides support so for two mainstream CPU architectures: armeabi-v7a and arm64-v8a.(No need to integrate JNI without unlocking and flow control functions)



## 1.2. Configure the android manifest permission

### 1.2.1 Integrated USB printer added below in <manifest>

```
1 <!-- usb permissions -->
2 <uses-feature android:name="android.hardware.usb.host" />
3 <uses-permission android:name="android.hardware.usb.host" />
```

Note: Using USB communication requires using an OTG cable and keeping the phone in USB Host mode.

The main activity is in the ANDROIDMANIFESTO. The XML needs to be configured as follows:

```
<activity
    android:name=".MainActivity"
    android:screenOrientation="portrait">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
        <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
    </intent-filter>
    <meta-data
        android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED"
        android:resource="@xml/qwm_usb_xml" />
    </activity>
```

Add to the <activity> Tab

```
1 <meta-data
2     android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED"
3     android:resource="@xml/qwm_usb_xml" />
```

Add to the <intent-filter> Tab

```
1 <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
```

Create the XML folder under the RES folder and copy sdkdemo to QWM. XML files, see the SDK demo or refer to the Android USB Host mode development guide.

### 1.2.2 Integrated Bluetooth printers are added below in <manifest>

```
1 <!-- bluetooth permissions -->
2 <uses-permission android:name="android.permission.BLUETOOTH" />
3 <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
4 <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
5 <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
6 <!-- Android 12 bluetooth -->
7 <uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
8 <uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE" />
9 <uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
```

Note: the use of partial permissions since Android 6.0 requires developers to apply dynamically in the code. Android 12 BLUETOOTH\_SCAN, BLUETOOTH\_ADVERTISE, and BLUETOOTH\_CONNECT runtime permissions were introduced

### 1.2.3 Integrated WiFi printer is added below in <manifest>

```
1 <!-- Tcp permissions-->
2 <uses-permission android:name="android.permission.INTERNET" />
3 <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
4 <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

,

## 1.3. Initialization SDK

Create a new custom Application and complete the SDK initialization in its onCreate method. The example code is as follows:

```
public class DemoApplication extends Application {  
    @Override  
    public void onCreate() {  
        super.onCreate();  
        //printer init  
        JXPrinter.init(context: this, BuildConfig.DEBUG);  
        //global printer init  
        JXGlobalPrinter.init(context: this, BuildConfig.DEBUG);  
    }  
}
```

In the Androidmanifesto St. The Application is declared in the. XML file

```
<application  
    android:name=".DemoApplication"
```

## 1.4. Obfuscation SDK

If your application uses code obfuscation, add the following configuration to avoid the SDK being obfuscated and the SDK being unavailable

```
1 -keep class com.jx.it.printer.* {*;}
```

At this point, the SDK integration configuration is complete.

## 2. Bluetooth connected printers

### 2.1. General usage flow

(1) searching for a Bluetooth printer (2) a printer Bluetooth pairing (3) establishing a Bluetooth connection (4) sending a printer instruction

### 2.2. Get the JXBluetoothAPI Global Singleton

```
JXBluetoothAPI mApi = JXBluetoothAPI.getDefault(context)
```

### 2.3. Get Connection status

#### Overview

Gets the current Bluetooth connection status

#### Name

isConnected()

#### Response

True is returned when connected, false when not connected.

### 2.4. Enable/disable Bluetooth search, search for Bluetooth devices

#### Overview

Enable/disable Bluetooth search, search for Bluetooth devices

#### Name

startDiscovery()

cancelDiscovery()

#### Response

Searching for Bluetooth devices requires listening to system Bluetooth broadcasts, action for Bluetooth device. Action, see `sdkdemo # Bluetooth` fragment or refer to the Android Bluetooth development guide.

## 2.5. Get a list of paired Bluetooth devices

### Overview

Gets a list of paired Bluetooth devices

### Name

`getBondedDevices()`

### Response

`set<BluetoothDevice>`

## 2.6. Bluetooth pairing/unpairings

### Overview

Bluetooth pairing/unpairing

### Name

`createBTBond(BluetoothDevice device)`

`removeBTBond(BluetoothDevice device)`

### Response

The default matching password for the printer is 0000, Bluetooth device. Action, see `sdkdemo # Bluetooth` fragment or refer to the Android Bluetooth development guide.

## 2.7. Bluetooth connection/disconnect

### Overview

Bluetooth connected/disconnected

### Name

`openConnection(String macAddress)`

`closeConnection()`

### Response

Return true successfully, otherwise return false.

## 3. USB connected printer

### 3.1. General usage flow

(1) the mobile phone connects with the OTG line in host mode and the printer (2) establishes the USB connection (3) sends the printer instructions

### 3.2. Get the JXUSBAPI Global Singleton

```
JXUsbAPI mApi = JXUsbAPI.getDefault(getApplicationContext());
```

### 3.3. Get Connection status

#### Overview

Gets the current Bluetooth connection status

#### Name

`isConnected()`

#### Response

True is returned when connected, false when not connected.

### 3.4. Get Connected USB devices

#### Overview

Get the connected USB device

#### Name

getDevices()

#### Response

List<UsbDevice>

## 3.5. USB connectiondisconnect

### Overview

USB connection/disconnection

#### Name

openConnection(UsbDevice device)

closeConnection()

#### Response

Return true successfully, otherwise return false.

Note: Using USB communication requires using an OTG cable and keeping the phone in USB Host mode.

The main activity is in the ANDROIDMANIFESTO. The XML needs to be configured as follows:

```
<activity
    android:name=".MainActivity"
    android:screenOrientation="portrait">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
        <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
    </intent-filter>
    <meta-data
        android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED"
        android:resource="@xml/qwm_usb_xml" />
    </activity>
```

Add to the <activity> Tab

```
1 <meta-data
2     android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED"
3     android:resource="@xml/qwm_usb_xml" />
```

Add to the <intent-filter> Tab

```
1 <action android:name="android.hardware.usb.action.USB_DEVICE_ATTACHED" />
```

Create the XML folder under the RES folder and copy sdkdemo to QWM. XML files, see the SDK demo or refer to the Android USB Host mode development guide.

## 4. Get the printer instance

### 4.1. Get an instance of a Bluetooth printer

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 //Get the line mode printer instance
3 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode
4 //Get the page mode printer instance
5 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode
6 //Get other command printer instance
7 JXPrinter printer= new JXPrinter(mAPI);
```

## 4.2. Get a Usb printer instance

```
1 JXInterfaceAPI mApi = JXUsbAPI.getDefault(context);
2 //Get the line mode printer instance
3 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintModeLine);
4 //Get the page mode printer instance
5 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintModePage);
6 //Get other command printer instance
7 JXPrinter printer= new JXPrinter(mAPI);
```

## 5. Printer settings

### 5.1. Get print blackness

#### Overview

Gets the print blackness of the printer

#### Name

get\_print\_darkness()

#### Parameter

None

#### Response

Parameter	Name	Extra
value	Printer blackness	Printer Blackness 0~4

### 5.2. Set printer blackness

#### Overview

Sets the print blackness of the printer

#### Name

set\_print\_darkness(String value)

#### Parameter

Parameter	Name	Extra
value	Printer blackness <code>value</code>	Printer Blackness 0

#### Response

Successful true, Failure returns false.

### 5.3. Get print quality

#### Overview

Gets the print quality of the printer

#### Name

get\_print\_quality()

#### Parameter

None

#### Response

Parameter	Name	Extra
value	Printer quality value	Printer quality 0~2

## 5.4. Set printer quality

### Overview

Sets the print blackness of the printer

### Name

set\_print\_quality( String value)

### Parameter

Parameter	Name	Extra
value	Printer quality value	Printer quality 0~2

### Response

Successful true, Failure returns false.

## 5.5. Get printer status

### Overview

Gets the current state of the printer

### Name

get\_printer\_status()

### Parameter

None

### Response

Parameter	Name	Extra
JXPrinterStatus	Printer status	JXPrinterStatus.isReady JXPrinterStatus.isPrinting JXPrinterStatus.isCoverOpened JXPrinterStatus.isNoPaper JXPrinterStatus.isOverHeat JXPrinterStatus.isBatteryLow JXPrinterStatus.isUnknownError

## 5.6. Print self-check Page

### Overview

Print self-check page

### Name

print\_check\_page( )

**Parameter:** None

### Response

Successful true, Failure returns false. 。

## 5.7.Unlock the printer

### Overview

For custom printers, the printer is unlocked when connecting, otherwise it cannot be used  
(Using this feature requires integration with JNI)

#### Name

set\_print\_unlock( )

**Parameter:** None

#### Response

Successful true, Failure returns false.

## 6. Global Line Directive

The Global LINE directive supports multi-language text printing. Using the Global LINE directive requires an additional dependency on zxing.jar; if not, none. Instruction methods all begin with line\_.

```
1 Generally used for continuous paper printing. Steps:  
2 1. Obtain the print service singleton object  
3 2. Set the print width by calling the method of set the maximum print width of printer  
4 3. Call print text or image and other methods to set the print content, if the content width exceeds the width of a line will be printed, less than a line will be cached, concatenated with the content to be printed later, you can also call the method of forced print buffer, the cache content is forced to print out.
```

### 6.1. Set maximum print width

#### Overview

The default is 384, depending on the printer's actual print width setting. After creating an instance of JXGlobalPrinter ( ) , you need to set the priority setting once. **Must be set before printing**

#### Name

line\_setMaxPrintWidth(int maxWidth)

#### Parameter

Parameter	Name	Extra
maxWidth	maximum print width	Points of print width (8 point/mm)

#### Response

None.

### 6.2. Clear cache data

#### Overview

Clear cache data all data restore default values.

#### Name

line\_reset()

#### Response

Successful true, Failure returns false.

### 6.3. Feed



## Overview

Feed lines.

## Name

line\_feed(int n)

## Parameter

Parameter	Name	Extra
n	Line count	Line count (8 point/mm)

## Response

Successful true, Failure returns false.

## 6.4.Print cached content

### Overview

Print and clear the cache

### Name

line\_print()

### Response

Successful true, Failure returns false.

## 6.5.Set left margin

### Overview

Set the left margin for printing, which will remain in effect until you cancel it

### Name

line\_setLeftMargin(int margin)

### Parameter

Parameter	Name	Extra
margin	Margin points	Points (8 dot/mm) 0–65535  The left and right margins need to be considered. If both sides are set, the setting fails when the printable area is less than 100

### Response

Successful true, Failure returns false.

## 6.6. Set right margin

### Overview

Set the right margin of the print, after setting unless the active cancellation, otherwise will remain in effect

### Name

line\_setRightMargin(int margin)

### Parameter

Parameter	Name	Extra
margin	Margin points	Points (8 dot/mm) 0–65535  The left and right margins need to be considered. If both sides are set, the setting fails when the printable area is less than 100

#### Response

Successful true, Failure returns false.

## 6.7.Set relative print position

#### Overview

Set the relative print position. The setting will take effect until it is canceled

#### Name

line\_setAlign(int align)

#### Parameter

Parameter	Name	Extra
align	relative print position	0–AlignLeft, 1–AlignCenter, 2–AlignRight

#### Response

Successful true, Failure returns false.

## 6.8. Set print font properties

#### Overview

To set the print font, after setting, unless the active cancellation, otherwise always in effect, you need to set before the print text

#### Name

line\_setFontTypeface(Typeface typeface) (Set the print font)

line\_setFontUnderline(boolean isUnderline) (Sets whether text is underlined)

line\_setFontBold(boolean isBold) (Sets whether the text is bold)

line\_setFontItaly(boolean isItaly) (Sets whether the text is italic)

line\_setFontSize(int fontSize) (Set font size)

line\_setFontScale(int widthScale, int heightScale) (Set font magnification)

line\_setLineSpacing(int lineSpacing) (Set the spacing)

line\_setLetterSpacing(int letterSpacing) (Setting character Spacing)

#### Parameter

Method	Parameter	Name	Extra
line_setFontTypeface	typeface	typeface	Set font typeface, if null, clear font typeface, use default typeface of the phone
line_setFontUnderline	isUnderline	isUnderline	true: has an underscore false: has no underscore
line_setFontBold	isBold	isBold	false: cancel bold; true: set bold;
line_setFontItalic	isItalic	Is italic	false: cancel italic; true: set italic;
line_setFontSize	fontSize	Font size	Font size, 1mm = 8 dot
line_setFontScale	widthScale	Width amplification	Width amplification, 1–10
line_setFontScale	heightScale	Height amplification	Height amplification, 1–10
line_setLineSpacing	lineSpacing	Line spacing	Line spacing, 1mm = 8 dot
line_setLetterSpacing	letterSpacing	Letter spacing	Letter spacing, 1mm = 8 dot

#### Response

None

## 6.9. Invert black and white

#### Overview

Set the inverted color when draw text, after the setting unless the initiative to cancel, otherwise will take effect

#### Name

line\_setFontReverse(boolean isReverse)

#### Parameter

Parameter	Name	Extra
isReverse	Invert black and white	true: set invert B/W; false: cancel invert B/W ;

#### Response

None

#### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {

```

```

6         printer.line_setMaxPrintWidth(576);
7         printer.line_reset();
8         printer.line_setFontReverse(true);
9         printer.line_drawText("你好hello123456\n");
10    }
11    }).start();

```



## 6.10. Draw text

### Overview

Draw text, full line automatic printing, you can use N control line break, call line can force print buffer text

### Name

line\_drawText(String content)

### Parameter

Parameter	Name	Extra
content	Content to print	full line automatically print, you can use \n to control the newline, you can call line_print to force the print buffer text

### Response

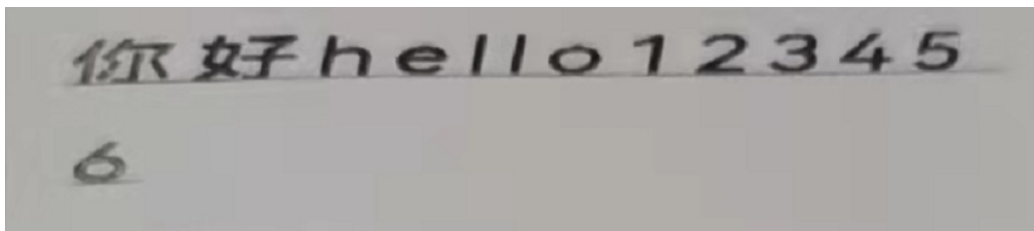
None

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.line_setMaxPrintWidth(576);
7          printer.line_reset();
8          printer.line_setFontBold(true);
9          printer.line_setFontUnderline(true);
10         printer.line_setFontItalic(true);
11         printer.line_setFontSize(32);
12         printer.line_setFontScale(2, 1);
13         printer.line_setLineSpacing(8);
14         printer.line_setLetterSpacing(8);
15         printer.line_drawText("你好hello123456");
16         printer.line_print();
17     }
18 }).start();

```



## 6.11.Draw Line

### Overview

draw a line

### Name

line\_drawLine(float lineLength, int lineWidth, boolean fullLine)

### Parameter

Parameter	Name	Extra
lineLength	Length of line	if the maximum print length is exceeded, only printable part can be printed
lineWidth	Width of line	Width of line
fullLine	Line type	true: solid line false: dot line

### Response

Successful true, Failure returns false.

### Examples

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.line_setMaxPrintWidth(576);
7          printer.line_reset();
8          printer.line_drawLine(80,2, true);
9          printer.line_print();
10     }
11 }).start();
```



## 6.12.Draw 1D barcode

### Overview

Draw 1D barcode . If you want to print content text, call line\_setBarcode1DText first

You can call line\_print to force the buffer to be printed

### Name

line\_drawBarcode1D(int width, int height, String content, int rotate, BarcodeFormat format)

Parameter

Parameter	Name	Extra
width	Width of total barcode	Points (8 dot/mm)
height	Height of total barcode	Points (8 dot/mm)
content	Content	content of barcode
rotate	Rotate clockwise	90: Rotate 90 degrees clockwise 180: Rotate 180 degrees clockwise 270: Rotate 270 degrees clockwise
format	Barcode format	BarcodeFormat.CODABAR, BarcodeFormat.CODE_39, BarcodeFormat.CODE_93, BarcodeFormat.CODE_128, BarcodeFormat.EAN_8, BarcodeFormat.EAN_13, BarcodeFormat.ITF, BarcodeFormat.UPC_A, BarcodeFormat.UPC_E;

Response

Successful true, Failure returns false.

Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.line_setMaxPrintWidth(576);
7         printer.line_reset();
8         printer.line_drawBarcode1D( 300, 100, "236563666", 0,
9             BarcodeFormat.CODE_128);
10        printer.line_print();
11    }
12 }).start();
```



## 6.13.Draw content of 1D barcode

### Overview

Print the content of the bar code, only once, can be used by calling line\_drawBarcode1D

### Name

line\_setBarcode1DText(boolean isPositionBelow, int fontSize, boolean isBold, int margin, boolean breakLine)

### Parameter

Parameter	Name	Extra
isPositionBelow	Whether the text is below the barcode	true: below false: above
fontSize	Font size	Points ,1mm = 8 dot
isBold	Bold font	false: cancel bold ; true: set bold;
margin	Margin between barcode and text	margin, 1mm = 8 dot
breakLine	Whether break the text if the text width is large than barcode width	false: don't break the text if the text width is large than barcode width; true: break the text if the text width is large than barcode width.

### Response

None

### Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.line_setMaxPrintWidth(576);
7         printer.line_reset();
8         printer.line_setBarcode1DText(false, 24, false, 0, true);
9         printer.line_drawBarcode1D( 300, 100, "236563666", 0,
10             BarcodeFormat.CODE_128);
11         printer.line_print();
12     }
13 }).start();
```



## 6.14.Draw 2D barcode

### Overview

Draw 2D barcode .

You can call line\_print to force the buffer to be printed.

### Name

line\_drawBarcode2D(int width, int height, String content, int rotate, BarcodeFormat format)

### Parameter

Parameter	Name	Extra
width	Width of barcode	Points (8 dot/mm)
height	Height of barcode	Points (8 dot/mm)
content	Content	content of barcode
rotate	Rotate clockwise	90: Rotate 90 degrees clockwise 180: Rotate 180 degrees clockwise 270: Rotate 270 degrees clockwise
format	Barcode format	BarcodeFormat.AZTEC, BarcodeFormat.PDF_417, BarcodeFormat.QR_CODE, BarcodeFormat.DATA_MATRIX,

### Response

Successful true, Failure returns false..

### Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.line_setMaxPrintWidth(576);
7         printer.line_reset();
8         printer.line_drawBarcode2D( 80, 80, "1234567890", 0,
9             BarcodeFormat.QR_CODE);
10        printer.line_print();
11    }
12 }).start();
```





## 6.15.Draw Graphic

### Overview

Draw a bitmap

### Name

line\_drawGraphic(Bitmap bitmap)

### Parameter

Parameter	Name	Extra
bitmap	Bitmap	Bitmap

### Response

Successful true, Failure returns false.

### Examples

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Line);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.line_setMaxPrintWidth(576);
7          printer.line_reset();
8          printer.line_drawGraphic( bitmap);
9          printer.line_print();
10     }
11 }).start();
```

## 6.16.Draw Curve

### Overview

Print the line segments of a curve (a curve is made up of continuous line segments, and points can be understood as line segments of length 1)

### Name

line\_drawLineSegments(int... xPos)

### Parameter

Parameter	Name	Extra
xPos	Point array	<p>An array of x coordinates, each 2 numbers representing a point or line segment, with x values ranging from 0 to 243.</p> <p>If only one point X is printed, that is, Parameter passes X, X.</p> <p>If you print a line segment X, you pass in Xs, Xe.</p> <p>If you print multiple line segments X1 on the same horizontal line.. Xn, i.e. Xs1,Xe1,Xs2,Xe2... Xsn, Xen.</p>

### Response

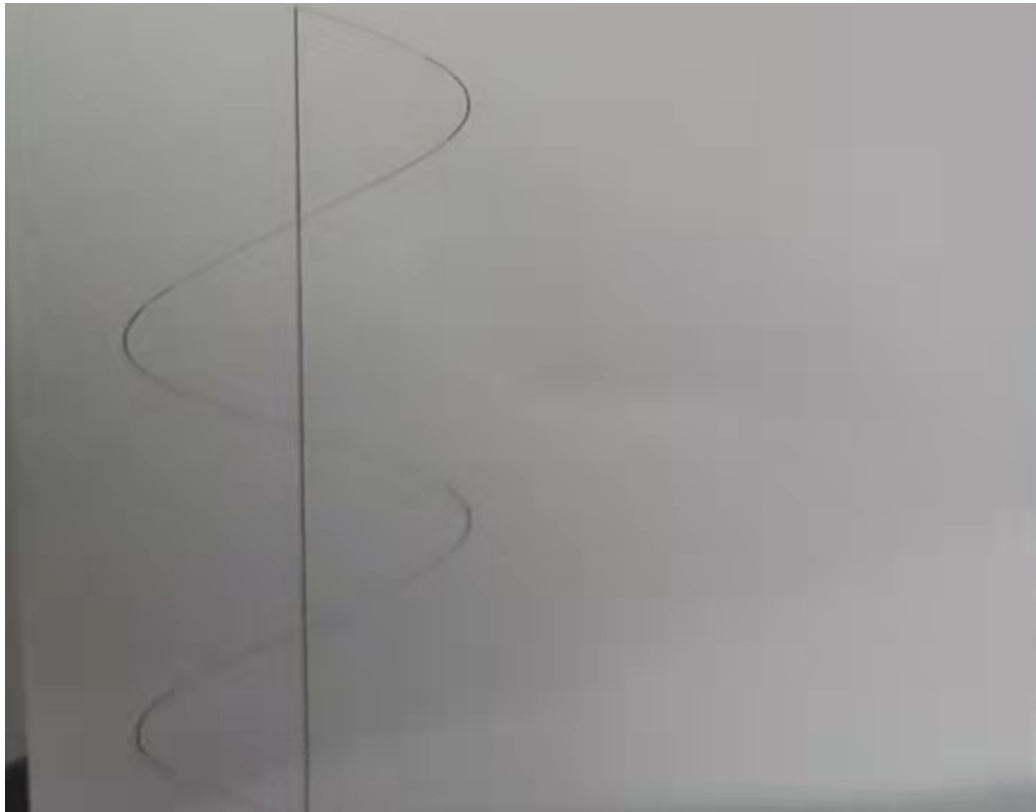
Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
   de.Line);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.line_setMaxPrintWidth(576);
7          printer.line_reset();
8          int x;
9          for (int y = 1; y <= 512; y++) {
10             x = (int) (sin(Math.PI * y/ 128) * 100 + 130);
11             printer.line_drawLineSegments(130, 130, x, x);
12         }
13     }
14 }).start();

```



## 7.Global Page Directive

The Global PAGE directive supports multi-language text printing. Using the Global PAGE directive requires an additional dependency on zxing.jar; if not, none. Instruction methods all begin with page\_.

```
1 Generally, it is used to print labels. Steps:
2 1. Obtain the print service singleton object
3 2. Set the print area by call the method of set the print paper size
4 3. Set the print content by call the method such as print text or image
5 4. Call the execute print method to print
```

### 7.1. Access Control Command

#### Overview

Let the printer into command control state, and set the print width, height, with page use (must be set before printing)

#### Name

page\_setup(int width, int height, int paperType)

#### Parameter

Parameter	Name	Extra
width	Page width points	Page width points(8 points / mm)
height	Page height points	Page height points(8 points / mm)
paperType	Paper Type	0 gap paper 1 continuous paper

#### Response

Successful true, Failure returns false.

## 7.2.Print Control Content Directive

### Overview

Causes the printer control state to end, and prints out the contents of the control section. This method must be called after you have sent the contents, **otherwise the sent instruction will not be printed**

### Name

page\_print(boolean isUpsideDown, int times)  
page\_print()

### Parameter

Parameter	Name	Extra
isUpsideDown	page order	true: Up and down 180 print upside down; false: Forward printing;
times	Number of printed copies	1–255

### Response

Successful true, Failure returns false.

## 7.3.Clear text Settings

### Overview

Clear the previous text Settings

### Name

page\_resetFontSettings()

### Response

None

## 7.4. Set print font properties

### Overview

To set the print font, after setting, unless the active cancellation, otherwise always in effect, you need to set before the print text

### Name

page\_setFontTypeface(Typeface typeface) (Set the print font)  
page\_setFontUnderline(boolean isUnderline) (Sets whether text is underlined)  
page\_setFontBold(boolean isBold) (Sets whether the text is bold)  
page\_setFontItalic(boolean isItalic) (Sets whether the text is italic)  
page\_setFontSize(int fontSize) (Set font size)  
page\_setFontScale(int widthScale, int heightScale) (Set font magnification)  
page\_setLineSpacing(int lineSpacing) (Set the spacing)  
page\_setLetterSpacing(int letterSpacing) (Setting character Spacing)

### Parameter

Parameter	Name	Extra
typeface	font	Set font style, if null, clear font style
isUnderline	Underline	true: Underline; false: Not Underline;
isBold	Bold	false: Not Bold; true: Bold;
isItaly	Italy	false: Not Italy; true: Italy;
fontSize	size	Print font height, 1mm = 8 dot
widthScale	Width amplification	Width amplification, 1–10
heightScale	Height amplification	Height amplification, 1–10
lineSpacing	Line spacing	Line spacing, 1mm = 8 dot
letterSpacing	Letter spacing	Letter spacing, 1mm = 8 dot

#### Response

None

## 7.5. Invert black and white

#### Overview

Set the inverted color when draw text, after the setting unless the initiative to cancel, otherwise will take effect

#### Name

page\_setFontReverse(boolean isReverse)

#### Parameter

Parameter	Overview	Extra
isReverse	Invert black and white	true: set invert B/W; false: cancel invert B/W ;

#### Response

None

#### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800, 0);
7          printer.page_resetFontSettings();
8          printer.page_setFontReverse(true);
9          printer.page_drawText(10, 210, 400, 700, 0, false,
10         "你好hello123456");
11         printer.page_print();

```

```
12     }
13 }).start();
```



7.6.Draw Text

Overview

Print the text of parameters, support multiple languages. Use with page\_print

Name

page\_drawText(int start\_x, int start\_y, int end\_x, int end\_y, int rotate, boolean singleLine, String content)

Parameter

Parameter	Name	Extra
start_x	The starting position of the X-axis	The starting position of the X-axis
start_y	The starting position of the Y-axis	The starting position of the Y-axis
end_x	The end of the X-axis	Text Print areaThe end of the X-axis
end_y,	The end of the Y-axis	Text Print areaThe end of the Y-axis
content	content	Print the text content
rotate	Rotate the angle clockwise	90: Turn 90 degrees clockwise; 180: Rotate 180 degrees clockwise; 270: Rotate 270 degrees clockwise;
singleLine	Single Line mode	true: Print only one line, redundant truncation false: Auto linefeed

Response

Successful true, Failure returns false.

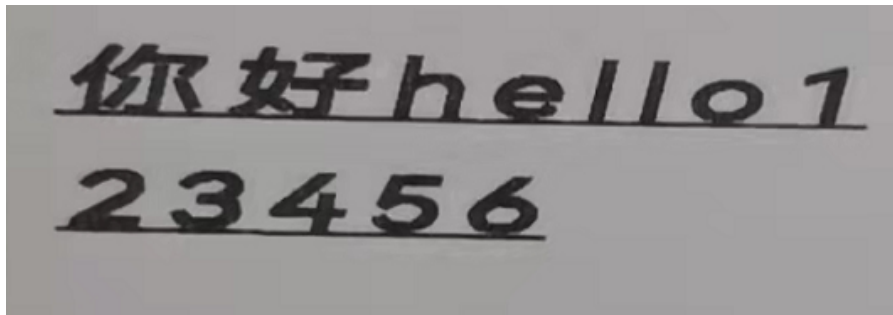
Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.page_setup(576, 800,0);
7         printer.page_resetFontSettings();
8         printer.page_setFontUnderline(true);
9         printer.page_setFontBold(true);
10        printer.page_setFontItaly(true);
11        printer.page_setFontSize(32);
```

```

12     printer.page_setFontScale(2, 1);
13     printer.page_setLineSpacing(16);
14     printer.page_setLetterSpacing(8);
15     printer.page_drawText(10, 210, 400, 700, 0, false,
16         "你好hello123456");
17     printer.page_print();
18 }
19 }).start();

```



## 7.7.Draw Box

### Overview

Add Rectangle Box Directive

### Name

page\_drawBox(int start\_x, int start\_y, int width, int height, int lineWidth, boolean fullLine)

### Parameter

Parameter	Name	Extra
start_x	Upper left corner X coordinates	Upper left corner X coordinates
start_y	Upper left corner Y coordinates	Upper left corner Y coordinates
width	width	Box width
height	height	Box height
lineWidth	Line width	Line width forming a rectangular shape
fullLine	Line Type	true: full line false: dotted line

### Response

Successful true, Failure returns false.

### Examples

```

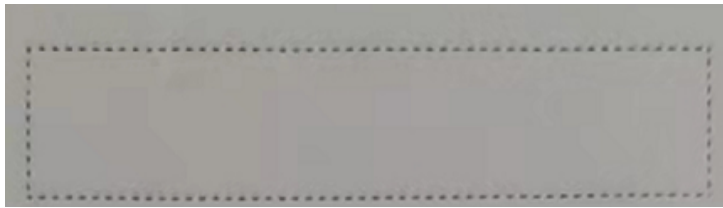
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800,0);

```

```

7         printer.page_drawBox( 0, 10, 510, 110, 2, false);
8         printer.page_print();
9     }
10 }).start();

```



## 7.8.Draw Line

### Overview

draw a line

### Name

page\_drawLine(int start\_x, int start\_y, int end\_x, int end\_y, int lineWidth, boolean fullLine)

### Parameter

Parameter	Name	Extra
start_x	Upper left corner X coordinates	Upper left corner X coordinates
start_y	Upper left corner Y coordinates	Upper left corner Y coordinates
width	width	Box width
height	height	Box height
lineWidth	Line width	Line width forming a rectangular shape
fullLine	Line Type	true: full line false: dotted line

### Response

Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800,0);
7          printer.page_drawLine(2, 0, 10, 510, 10, true);
8          printer.page_print();
9      }
10 }).start();

```





## 7.9.Draw 1D barcode

### Overview

Draw 1D barcode . If you want to print content text, call `page_setBarcode1DText` first  
You can call `line_print` to force the buffer to be printed

### Name

`page_drawBarcode1D(int start_x, int start_y, int width, int height, String content, int rotate, BarcodeFormat format)`

### Parameter

Parameter	Name	Extra
start_x	Upper left corner X coordinates	Upper left corner X coordinates
start_y	Upper left corner Y coordinates	Upper left corner Y coordinates
width	Width of total barcode	Points (8 dot/mm)
height	Height of total barcode	Points (8 dot/mm)
content	Content	content of barcode
rotate	Rotate clockwise	90: Rotate 90 degrees clockwise 180: Rotate 180 degrees clockwise 270: Rotate 270 degrees clockwise
format	Barcode format	BarcodeFormat.CODABAR, BarcodeFormat.CODE_39, BarcodeFormat.CODE_93, BarcodeFormat.CODE_128, BarcodeFormat.EAN_8, BarcodeFormat.EAN_13, BarcodeFormat.ITF, BarcodeFormat.UPC_A, BarcodeFormat.UPC_E;

### Response

Successful true, Failure returns false.

### Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
```

```

2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
de.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800,0);
7          printer.page_drawBarcode1D(10, 10, 300, 100,"236563666", 0,
8              BarcodeFormat.CODE_128);
9          printer.page_print();
10     }
11 }).start();

```



## 7.10.Draw content of 1D barcode

### Overview

Print the content of the bar code, only once, can be used by calling page\_drawBarcode1D

### Name

page\_setBarcode1DText(boolean isPositionBelow, int fontSize, boolean isBold, int margin, boolean breakLine)

### Parameter

Parameter	Name	Extra
isPositionBelow	Whether the text is below the barcode	true: below false: above
fontSize	Font size	Points ,1mm = 8 dot
isBold	Bold font	false: cancel bold ; true: set bold;
margin	Margin between barcode and text	margin, 1mm = 8 dot
breakLine	Whether break the text if the text width is large than barcode width	false: don't break the text if the text width is large than barcode width; true: break the text if the text width is large than barcode width.

### Response

None

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);

```

```

2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
de.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800,0);
7          printer.page_setBarcode1DText(false, 24, false, 0, true);
8          printer.page_drawBarcode1D(10, 10, 300, 100,"236563666", 0,
9          BarcodeFormat.CODE_128);
10         printer.page_print();
11     }
12 }).start();

```



## 7.11. Draw 2D barcode

### Overview

Draw 2D barcode

### Name

page\_drawBarcode2D(int start\_x, int start\_y, int width, int height, String content, int rotate, BarcodeFormat format)

### Parameter

Parameter	Name	Extra
start_x	Upper left corner X coordinates	Upper left corner X coordinates
start_y	Upper left corner Y coordinates	Upper left corner Y coordinates
width	Width of barcode	Points (8 dot/mm)
height	Height of barcode	Points (8 dot/mm)
content	Content	content of barcode
rotate	Rotate clockwise	90: Rotate 90 degrees clockwise 180: Rotate 180 degrees clockwise 270: Rotate 270 degrees clockwise
format	Barcode format	BarcodeFormat.AZTEC, BarcodeFormat.PDF_417, BarcodeFormat.QR_CODE, BarcodeFormat.DATA_MATRIX,

## Response

Successful true, Failure returns false.

## Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.page_setup(576, 800, 0);
7         printer.page_drawBarcode2D(8, 8, 80, 80, "1234567890", 0,
8             BarcodeFormat.QR_CODE);
9         printer.page_print();
10    }
11 }).start();
```



## 7.12.Draw Graphic

### Overview

Bitmap printing instruction that can print a black and white bitmap

### Name

page\_drawGraphic(int start\_x, int start\_y, Bitmap bmp)

### Parameter

Parameter	Name	Extra
start_x	Upper left corner X coordinates	Upper left corner X coordinates
start_y	Upper left corner Y coordinates	Upper left corner Y coordinates
bmp	Bitmap to be printed	Bitmap to be printed

## Response

Successful true, Failure returns false.

## Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3 new Thread(new Runnable() {
4     @Override
```

```

5      public void run() {
6          printer.page_setup(576, 800,0);
7          printer.page_drawGraphic(0, 0, bmp);
8          printer.page_print();
9      }
10  }).start();

```

## 7.13.Feed and position to the label

### Overview

Feed and find the label and sew away the paper. If not found, go up to 10cm

### Name

page\_feed()

### Response

Successful true, Failure returns false.

## 8.ESC Directive

### 8.1. Initialize printer settings

#### Overview

Returns all settings to their default values when the printer was powered on

#### Name

esc\_reset()

#### Response

Successful true, Failure returns false.

### 8.2.Set print text font properties

#### Overview

To set the printing font, it needs to be set before printing the text (used in conjunction with 8.12 Printing text)

#### Name

esc\_bold(boolean isBold) (set bold mode)

esc\_underline(int n) (select underline mode)

esc\_default\_line\_height() (set the default line height)

esc\_line\_height(int n) (set line height)

esc\_right\_spacing(int n) (set character spacing)

esc\_font(int n) (select print font)

esc\_rotate(int n) (select/unrotate clockwise)

esc\_absolute\_print\_position(int position) (set the absolute print position)

esc\_relative\_print\_position(int n) (move the print position from the current position to

n x (horizontal motion unit)

#### Parameter

Method	Parameter	Name	Extra
esc_bold	isBold	Whether to set to bold mode	true: set to bold mode; false: cancel bold mode;
esc_underline	n	Whether to set to underline mode	n=1 or n=49: set to underline mode and set the underline height to 1 point; n=2 or n=50: set to underline mode and set the underline height to 2 points; n=other: cancel underline mode;
esc_line_height	n	row height value	Set line height to [n × vertical or horizontal shift units] inches
esc_right_spacing	n	spacing	n<0: Set the right character spacing to 0; n≥0 or n≤255: Set the right character spacing to n* (horizontal or vertical movement unit); n>255: Set the right character spacing to 255* (horizontal or vertical movement unit);
esc_font	n	font type	Font B is selected when n=1 or 49; Font C is selected when n=2 or 50; When n=3 or 51, select font D; when n is other values, select font A;
esc_rotate	n	character set type	n=1 or n=49: set 90° clockwise rotation mode; n=2 or n=50: set 180° clockwise rotation mode; n=3 or n=51: set 270° clockwise rotation mode; Cancel the rotation mode when n takes other values;
esc_absolute_print_position	position	Indicates the low-order bits of the absolute print position	Set the current position to position × (horizontal or vertical movement unit) from the beginning of the line
esc_relative_print_position	n	x coordinate offset	Move the print position from the current position to n × (horizontal movement unit)

### Examples

```

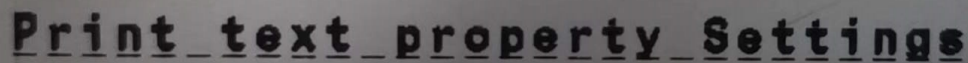
1 JXInterfaceAPI mAPI = PrinterManager.getInstance().getApi();
2 JXPrinter printer = new JXPrinter(mAPI);
3 new Thread(new Runnable() {

```

```

4      @Override
5      public void run() {
6          printer.esc_reset();
7          printer.esc_bold(true);
8          printer.esc_underline(1);
9          //printer.esc_default_line_height();
10         printer.esc_line_height(5);
11         printer.esc_right_spacing(5);
12         printer.esc_font(0);
13         printer.esc_rotate(0);
14         printer.esc_absolute_print_position(10);
15         //printer.esc_relative_print_position(int n);
16         printer.esc_print_text("Print text property Settings");
17
18     }
19 }).start();
20

```



### 8.3.Set the print left margin

#### Overview

set print left margin

#### Name

esc\_left\_margin(int n)

#### Parameter

Parameter	Name	Extra
n	left margin	set the left margin to n

#### Response

Successful true, Failure returns false.

#### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          try {
8              InputStream is = getResources().getAssets().open("logo.jpeg");
9              Bitmap bitmap = BitmapFactory.decodeStream(is);
10             printer.esc_print_formfeed();
11             printer.esc_left_margin(10); //打印左边距
12             printer.esc_bitmap_mode(1, bitmap);
13             printer.esc_reset();
14             printer.esc_print_formfeed();
15         } catch (IOException e) {

```

```

16         e.printStackTrace();
17     }
18 }
19 }).start();

```



## 8.4. Select the alignment mode

### Overview

Sets the alignment mode of the printed content.

### Name

esc\_align(int n)

### Parameter

Parameter	Name	Extra
n	x coordinate offset	n=1 or n=49: select center alignment n=2 or n=50: select right alignment n=other: select left alignment

### Response

Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          try {
8              InputStream is = getResources().getAssets().open("logo.jpeg");
9              Bitmap bitmap = BitmapFactory.decodeStream(is);
10             printer.esc_print_formfeed();
11             printer.esc_align(1); //打印内容的对齐模式
12             printer.esc_bitmap_mode(1, bitmap);
13             printer.esc_reset();
14             printer.esc_print_formfeed();
15         } catch (IOException e) {
16             e.printStackTrace();
17         }
18     }
19 }).start();

```





## 8.5. Horizontal Tabs

### Overview

Move the print position to the next horizontal tab position (used with method 8.6)

### Name

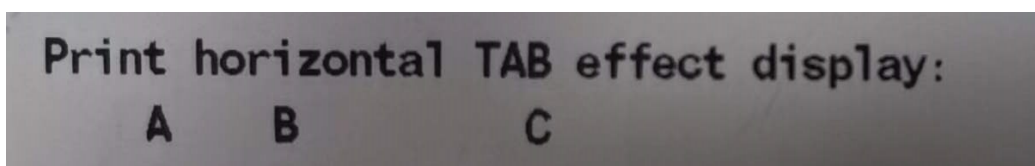
`esc_next_horizontal_tab()`

### Response

Successful true, Failure returns false.

### Examples

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          printer.esc_print_text("\n Print horizontal TAB effect display:\n");
8          printer.esc_horizontal_tab_position(new int[]{5,10,20});
9          printer.esc_next_horizontal_tab();
10         printer.esc_print_text("A");
11         printer.esc_next_horizontal_tab();
12         printer.esc_print_text("B");
13         printer.esc_next_horizontal_tab();
14         printer.esc_print_text("C");
15         printer.esc_print_formfeed();
16     }
17 }).start();
```



## 8.6. Set the horizontal tab position

### Overview

Set the horizontal tab position (only works on the same line, used in conjunction with 8.5 horizontal tabs)

#### Name

esc\_horizontal\_tab\_position(int[]n)

#### Parameter

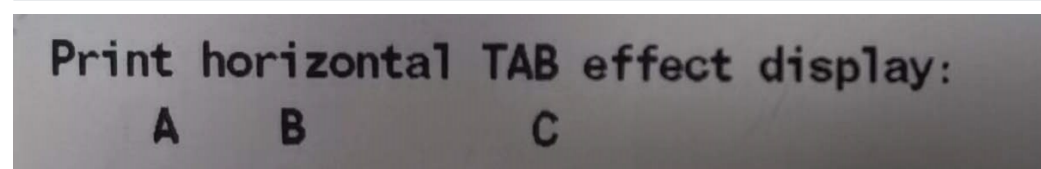
Parameter	Name	Extra
n	offset array	The length of n represents the number of horizontal tabs, and n[k] represents the value of the kth tab position. When the length of n is greater than 32, only the first 32 values are taken; when n[k] is greater than or equal to n[k-1 ], this command is ignored. This command is ignored when $n[k] \leq 0$ or $n[k] \geq 255$ .

#### Response

Successful true, Failure returns false.

#### Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXPrinter printer = new JXPrinter(mAPI);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
6         printer.esc_reset();
7         printer.esc_print_text("\n Print horizontal TAB effect display:\n");
8         printer.esc_horizontal_tab_position(new int[]{5,10,20});
9         printer.esc_next_horizontal_tab();
10        printer.esc_print_text("A");
11        printer.esc_next_horizontal_tab();
12        printer.esc_print_text("B");
13        printer.esc_next_horizontal_tab();
14        printer.esc_print_text("C");
15        printer.esc_print_formfeed();
16    }
17 }).start();
18
```



## 8.7. Select International Character Sets

#### Overview

Select international character set

#### Name

esc\_national\_character\_set(int n)

## Parameter

Parameter	Name	Extra
n	character set type	n≤0 or n>13: select the America character set; n=1: select the French character set; n=2: select German character set; n=3: select UK character set; n=4: select Denmark character set; n=5: select Swedish character set; n=6: select Italy character set; n=7: select Spain character set; n=8: select Japan character set; n=9: select the Norwegian character set; n=10: select Denmark character set; n=11: select SpainII character set; n=12: select Latin character set; n=13: select Korea character set;

## Response

Successful true, Failure returns false.

## 8.8. Select character code page

### Overview

Select character code page

### Name

esc\_character\_code\_page(int n)

### Parameter

Parameter	Name	Extra
n	code identification	n=1: select Page1Katakana; n=2: select Page2Multilingual(Latin-1) [CP850]; n=3: select Page 3Portuguese[CP860]; n=4: Select Page4Canadian-French[CP863]; n=5: select Page5Nordic[CP865]; n=6: select Page6Slavic(Latin-2)[CP852]; n=7: select Page7Turkish[CP857]; n=8: select Page8 Greek [CP737]; n=9: select Page9Russian(Cyrillic) [CP866]; n=10: select Page10 Hebrew [CP862]; n=11: select Page 11 Baltic [CP775]; n=12: select Page 12 Polish; n=13: Select Page 13 Latin-9 [ISO8859- 15]; n=14: Select Page 14 Latin1[Win1252]; n=15: Select Page 15 Multilingual Latin I + Euro[CP858]; n=16: select Page 16 Russian(Cyrillic) [CP855]; n=17: select Page 17 Russian(Cyrillic) [Win1251]; n=18: select Page 18 Central Europe[Win1250]; n=19: select Page 19 Greek[Win1253]; n=20: select Page20 Turkish[Win1254]; n=21: select Page21Hebrew[Win1255]; n=22: select Page22Vietnam[Win1258]; n=23: select Page23Baltic[Win1257]; n=24: select Page 24Azerbaijani; n=30: select Thai[CP874]; n=40: select Page25Arabic[CP720]; n=41: select Page 26 Arabic [Win 1256]; n=42: select Page 27 Arabic (Farsi); n=43: select Page 28 Arabic presentation formsB; n=50: select Page 29 Page25 Hindi_Devanagari; n=252: select Page 30Japanese[CP932]; n=253: select Page 31 Korean[CP949]; n=254: select Page 32 Traditional Chinese[CP950]; n=255: select Page33SimplifiedChinese[CP936]; When n takes other values, select Page0USA, StandardEurope[CP437];

## Response

Successful true, Failure returns false.

## 8.9. Set Chinese character mode

### Overview

Set Chinese character mode

### Name

esc\_chinese\_mode(boolean b)

### Parameter

Parameter	Name	Extra
b	Whether to set to Chinese character mode	true: set to Chinese character mode; false: Cancel Chinese character mode;

## Response

Successful true, Failure returns false.

## 8.10. Set character size

### Overview

Set the width and height of the characters

### Name

esc\_character\_size(int n)

### Parameter

Parameter	Name	Extra
n	character size	2 times higher when n=2; 3 times higher when n=3; 4 times higher when n=4; 2 times the width when n=20; 3 times the width when n=30; 4 times the width when n=40; 2 times the width and height when n=22; 3 times the width and height when n=33; 4 times the width and height when n=44; When n takes other values, it is 1 times the width and height;

## Response

Successful true, Failure returns false.

## 8.11. Set the inverse color

### Overview

set inverse color

### Name

esc\_black\_white\_reverse(boolean b)

### Parameter

Parameter	Name	Extra
b	Whether to select black and white reverse print mode	true: select the black and white reverse display printing mode; false: cancel the black and white reverse printing mode;

#### Response

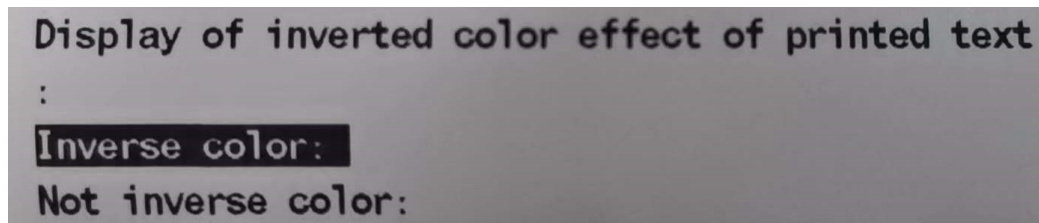
Successful true, Failure returns false.

#### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          printer.esc_print_text("Display of inverted color effect of printed
8          printer.esc_black_white_reverse(true);
9          printer.esc_print_text("Inverse color : \n");
10         printer.esc_black_white_reverse(false);
11         printer.esc_print_text("Not inverse color : \n");
12     }
13 }).start();

```



## 8.12. Print Text

#### Overview

Send text information to the printer device and perform printing operations

#### Name

esc\_print\_text(String text)

#### Parameter

Parameter	Name	Extra
text	text content	the text content to be printed

#### Response

Successful true, Failure returns false.

#### Examples

```

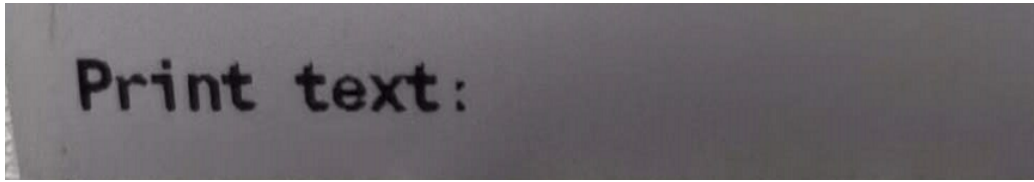
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {

```

```

6         printer.esc_reset();
7         printer.esc_print_text("Print text: \n");
8     }
9 }).start();
10

```



## 8.13. Set parameters and print barcode

### Overview

Set parameters and print barcode

### Name

esc\_barcode\_1d(int HRI\_position,int HRI\_font,int width,int height,int type,String content)

### Parameter

Parameter	Name	Extra
HRI_position	HRI character print position	HRI characters are displayed above the barcode when HRI_position=1 or HRI_position=49; HRI characters are displayed below the barcode when HRI_position=2 or HRI_position=50; HRI characters are not displayed when HRI_position takes other values;
HRI_font	HRI character font	Select font B when HRI_font=1 or HRI_font=49; Select font A when HRI_font takes other values;
width	barcode width	When width=2, set the barcode width to 2; When width=3, set the barcode width to 3; When width takes other values, set the barcode width to 1;
height	Bar code height	When $1 \leq \text{height} \leq 255$ , set the barcode height to height; When height takes other values, set the barcode height to 162;
type	Barcode Type	When type=0 or type=65, select the barcode type as UPC-A; When type=1 or type=66, select the barcode type as UPC-E; When type=2 or type=67, select the barcode type as EAN13; When type=3 or type=68, select the barcode type as EAN8;

		barcode type as EAN8; When type=4 or type=69, select the barcode type as CODE39; When type=5 or type=70, select the barcode type as ITF; When type=6 or type=71, select the barcode type as CODABAR; When type=7 or type=72, select the barcode type as CODE93; When type=8 or type=73, select the barcode type as CODE128;
content	barcode content	UPC-A (lengths 11, 12), UPC-E (lengths 7, 8, 11, 12), EAN13 (lengths 12, 13), EAN8 (lengths 7, 8), ITF (lengths greater than 2 even number) only supports numbers; CODE39 (length greater than 1 and less than 255, supports numbers, English, spaces, '\$', '%', '*', '+', '-', '.', ', /'); CODE93 (length greater than 1 and less than 255, supports numbers, English, spaces, '\$', '%', '+', '-', '.', ', /'); CODABAR (length greater than 2 and less than 255, supports numbers, English ABCDabcd, spaces, '\$', '+', '-', '.', ', /', ':'); CODE128 (length greater than 2 and less than 255, supports all English)

## Response

Successful true, Failure returns false.

## Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          printer.esc_print_formfeed();
8          printer.esc_barcode_1d(0, 1, 3, 80, 8, "123456789012");
9          printer.esc_print_formfeed();
10     }
11 }).start();

```



## 8.14. Print QR code



## Overview

print QR code

## Name

esc\_print\_barcode\_2d(int size,String content)

esc\_print\_barcode\_2d(int size, int qrVersion, int level, String content)

## Parameter

Parameter	Name	Extra
size	QR code magnification	Value 1–4
content	QR code content	QR code content
qrVersion	qr version	qr version, 1–10, the larger the value, the larger the width of the printed barcode, 3 is recommended
level	qr error correction level	qr error correction level, 0–3, the higher the value, the easier it is to identify, recommended 1

## Response

Successful true, Failure returns false.

## Examples

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          printer.esc_print_formfeed();
8          printer.esc_print_barcode_2d(4, "1234567890");
9          printer.esc_print_formfeed();
10         printer.esc_reset();
11         printer.esc_print_formfeed();
12         printer.esc_print_barcode_2d(4, 3,1,"1234567890");
13         printer.esc_print_formfeed();
14     }
15 }).start();
```



## 8.15. Print Raster Bitmaps

### Overview

print raster bitmap

### Name

esc\_raster\_image(Bitmap bitmap)

### Parameter

Parameter	Name	Extra
bitmap	picture	The information of the picture to be printed should not be too large due to the limited width of the printing paper.

### Response

Successful true, Failure returns false.

### Examples

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          try {
7              InputStream is = getResources().getAssets().open("logo.jpeg");
8              Bitmap bitmap = BitmapFactory.decodeStream(is);
9              printer.esc_print_formfeed();
10             printer.esc_align(1);
11             printer.esc_raster_image(bitmap);
12             printer.esc_reset();
13             printer.esc_print_formfeed();
14         } catch (IOException e) {
15             e.printStackTrace();
16         }
17     }
18 }
```

```

17     }
18     }).start();

```



## 8.16. Select Bitmap Mode to Print Pictures

### Overview

Set the mode of the bitmap, and print the image

### Name

esc\_bitmap\_mode(int m, Bitmap bitmap)

### Parameter

Parameter	Name	Extra
m	bitmap mode	m=1: Bitmap mode is 8-point double density; m=32: The bitmap mode is 24-point single density; m=33: Bitmap mode is 24-point double density; Except for m=1, 32, and 33, the bitmap mode is 8-point single density;
bitmap	picture	The information of the picture to be printed should not be too large due to the limited width of the printing paper.

### Response

Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer = new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.esc_reset();
7          try {
8              InputStream is = getResources().getAssets().open("logo.jpeg");

```

```

9         Bitmap bitmap = BitmapFactory.decodeStream(is);
10        printer.esc_print_formfeed();
11        printer.esc_align(1);
12        printer.esc_bitmap_mode(1, bitmap);
13        printer.esc_reset();
14        printer.esc_print_formfeed();
15    } catch (IOException e) {
16        e.printStackTrace();
17    }
18    }
19    }).start();

```



## 8.17. Print a line in parallel

### Overview

Based on the current line spacing, print the data in the buffer and feed one line

### Name

esc\_print\_formfeed()

### Response

Successful true, Failure returns false.

## 8.18. Print and advance paper n lines

### Overview

Based on the current line spacing, print the data in the buffer and feed n lines

### Name

esc\_print\_formfeed\_row(int n)

### Parameter

Parameter	Name	Extra
n	Lines of paper	n<0: feed 0 lines; n>255: feed 255 lines 0≤n≤255: feed n lines;

### Response

Successful true, Failure returns false.

## 8.19. Print and feed the paper to the right black mark

### Overview

Print all buffer data and feed the paper to the right black mark

### Name

`esc_print_to_right_black_label()`

### Response

Successful true, Failure returns false.

### Examples

## 8.20. Print and feed the paper to the left black mark

### Overview

Print out all the data in the print buffer and feed the paper to the left black mark

### Name

`esc_left_black_label()`

### Response

Successful true, Failure returns false.

## 8.21. Print and feeding paper to label

### Overview

Print all buffer data and feed paper to label

### Name

`esc_print_to_label()`

### Response

Successful true, Failure returns false.

## 8.22.Send a stream of data bytes

### Overview

Send a stream of bytes to the printer

### Name

`esc_write_bytes(byte[] bytes)`

### Parameter

Parameter	Name	Extra
bytes	byte stream	byte stream to send

### Response

Successful true, Failure returns false.

## 8.23.Read data byte stream

### Overview

Read the byte stream returned by the printer

### Name

`esc_read_bytes(byte[] bytes)`

### Parameter

Parameter	Name	Extra
bytes	byte stream	Read the returned byte stream

#### Response

Successful true, Failure returns false.

## 8.24.Clears the byte stream returned by the printer

#### Overview

Clears the byte stream returned by the printer

#### Name

esc\_clear()

#### Response

Successful true, Failure returns false.

## 9. CPCL Directive

### 9.1. Enter Control Directive

#### Overview

Let the printer enter the command control state, and set the print width and height, until the "print control content" method is called to end. Must be used one-to-one with cpcl\_print()

#### Name

cpcl\_pageSetup(int pageWidth, int pageHeight)

#### Parameter

Parameter	Name	Extra
pageWidth	page width in points	Page width dots (8 dots/mm)
pageHeight	page height in points	Maximum page height points (8 points/mm)

#### Response

Successful true, Failure returns false.

### 9.2. Print Control Content Directive

#### Overview

Ends the printer control state and prints out the contents of the control section. Must be used one-to-one with cpcl\_setup()

#### Name

cpcl\_print(int horizontal, int skip)

#### Parameter

Parameter	Name	Extra
horizontal	print order	0: Normal printing according to the control command sequence; Others: print in reverse order according to the control command;
skip	print start position	Whether to print after feeding the paper to the positioning point 0: First move the paper to the positioning point to print; Others: print directly without feeding paper;

#### Response

Successful true, Failure returns false.

### 9.3. Add text Directive

#### Overview

Send byte stream to printer device

#### Name

cpcl\_drawText(int text\_x, int text\_y, String text, int fontSize, int rotate, int bold, boolean reverse, boolean underline)

#### Parameter

Parameter	Name	Extra
text_x	X-axis start position	X-axis start position
text_y	Y axis start position	Y axis start position
text	content	print text
fontSize	font number	<p>Versions before v3.8.26:</p> <p>1: Corresponding font number 55 (see the appendix at the end of the document for the font number introduction);</p> <p>2: Corresponding font number 1;</p> <p>3: Corresponding font number 4;</p> <p>4: corresponds to font number 1, and the font width and height are doubled;</p> <p>5: corresponds to font number 4, and the font width and height are doubled;</p> <p>6: Corresponding font number 1, and the font width and height are expanded by four times;</p> <p>7: Corresponds to font number 4, and the font width and height are tripled;</p> <p>The rest: corresponding to font number 1;</p> <p>Versions after v3.8.26:</p> <p>The font size is the number of dots in the height of the printed font, 1mm=8dot.</p> <p>If the print height is 24, set 24.</p>
rotate	rotation mode	<p>1: Rotate 90 degrees counterclockwise;</p> <p>2: Rotate 180 degrees counterclockwise;</p> <p>3: Rotate 270 degrees counterclockwise;</p>
bold	bold mode	<p>0: cancel bold mode;</p> <p>1: set to bold mode;</p>
reverse	Black and white reverse mode	<p>true: set black and white inverse display mode;</p> <p>false: cancel black and white inverse display mode;</p>
underline	underline mode	<p>true: underlined;</p> <p>false: no underscore;</p>

## Response

Successful true, Failure returns false.

## Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override

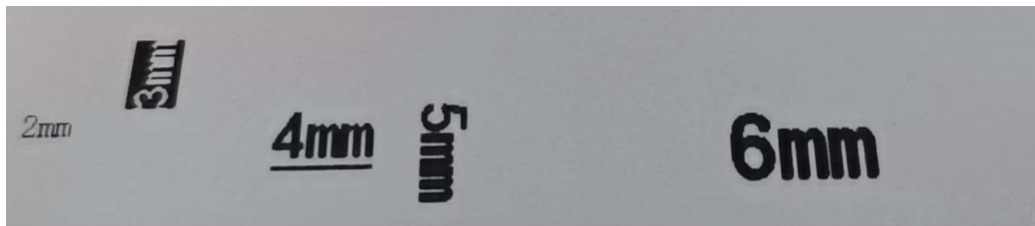
```



```

5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 100;
8          printer.cpcl_drawText(8, y, "2mm", 16,0,0,false,false );
9          printer.cpcl_drawText(56, y, "3mm", 24,1,0,true,false );
10         printer.cpcl_drawText(128, y, "4mm", 32,0,0,false,true );
11         printer.cpcl_drawText(224, y, "5mm", 40,3,0,false,false );
12         printer.cpcl_drawText(344, y, "6mm", 48,0,1,false,false );
13         printer.cpcl_print(0, 0);
14     }
15 }).start();

```



## 9.4. Add Line Segment Directive

### Overview

Add an instruction to draw a line segment

### Name

cpcl\_drawLine(int lineWidth, int start\_x, int start\_y, int end\_x, int end\_y, boolean fullline)

### Parameter

Parameter	Name	Extra
lineWidth	line width of the line segment	line width of the line segment
start_x	X coordinate of upper left corner	X coordinate of upper left corner
start_y	Y coordinate of upper left corner	Y coordinate of upper left corner
end_x	X coordinate of lower right corner	X coordinate of lower right corner
end_y	Y coordinate of lower right corner	Y coordinate of lower right corner

### Response

Successful true, Failure returns false.

### Examples

```

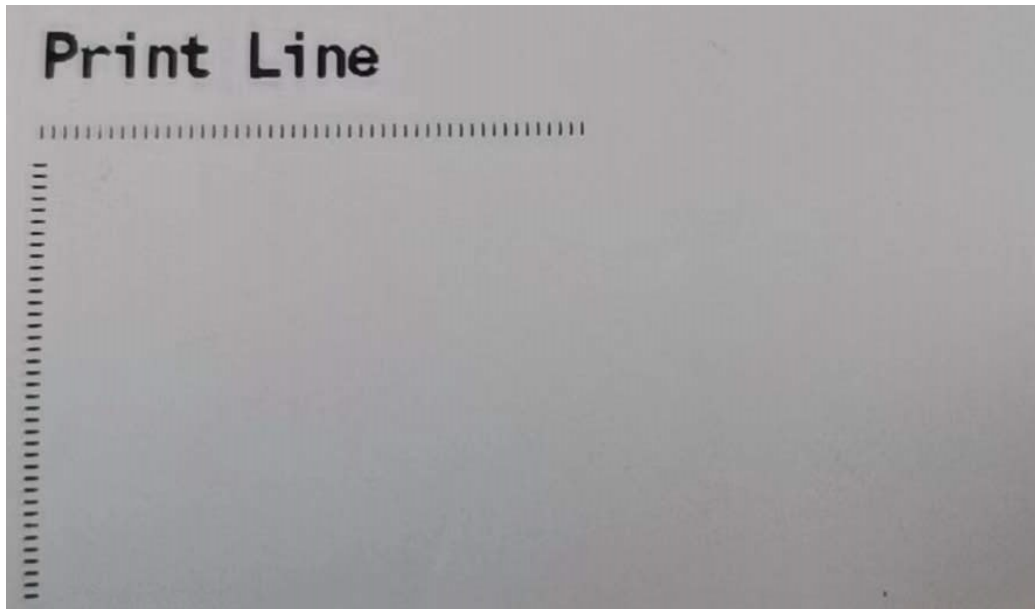
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {

```

```

6     printer.cpcl_pageSetup(576, 400, 0);
7     int y = 0;
8     printer.cpcl_drawText(8, y, "Print Line", 24,0,0,false,false );
9     printer.cpcl_drawLine(5,8, 40, 200, 40);
10    printer.cpcl_drawLine(5,8, 50, 8, 200);
11    printer.cpcl_print(0, 0);
12    }
13  }).start();

```



## 9.5. Rectangular Box Directive

### Overview

Add an instruction to draw a rectangle

### Name

cpcl\_drawBox(int lineWidth,int top\_left\_x,int top\_left\_y,int bottom\_right\_x,int bottom\_right\_y)

### Parameter

Parameter	Name	Extra
lineWidth	line width to form rectangle	line width to form rectangle
top_left_x	X coordinate of upper left corner	X coordinate of upper left corner
top_left_y	Y coordinate of upper left corner	Y coordinate of upper left corner
bottom_right_x	X coordinate of lower right corner	X coordinate of lower right corner
bottom_right_y	Y coordinate of lower right corner	Y coordinate of lower right corner

### Response

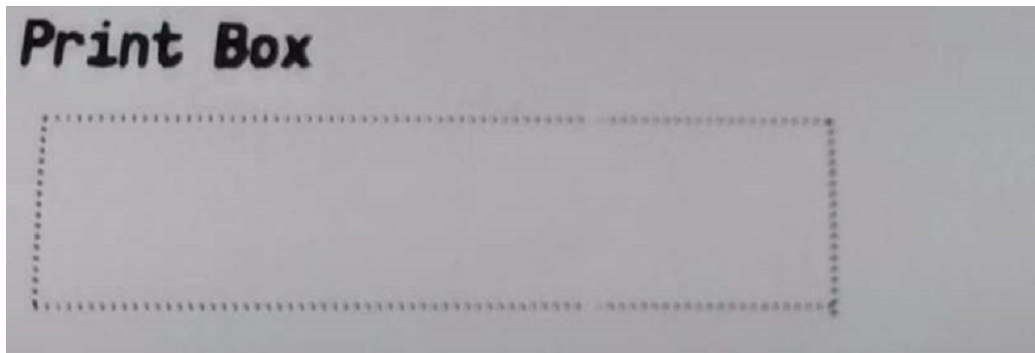
Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 0;
8          printer.cpcl_drawText(8, y, "Print Box", 24,0,0,false,false );
9          y+=30;
10         printer.cpcl_drawBox( 2,20, y, 310, 110 );
11         printer.cpcl_print(0, 0);
12     }
13 }).start();

```



## 9.6. 1D Barcode Directive

### Overview

Add instruction to print 1D barcode

### Name

cpcl\_drawBarCode(int start\_x, int start\_y, String text, int type, int rotate, int linewidth, int height)

### Parameter

Parameter	Name	Extra
start_x	The X-axis coordinate where the barcode starts	The X-axis coordinate where the barcode starts
start_y	The Y-axis coordinate where the barcode starts	The Y-axis coordinate where the barcode starts
text	barcode data	barcode data
type	Barcode Type	0: CODE_39 1: CODE_128 2: CODE_93 3: CODABAR 4: EAN_8 5: EAN_13 6: UPC_A 7: UPC_E (When selecting EAN_8, EAN_13, UPC_A, UPC_E type, do not input the last check digit content, otherwise it cannot be printed)
rotate	barcode orientation	0: Horizontal barcode 1: Vertical barcode
linewidth	Narrow barcode width in points	Narrow barcode width in points
height	Bar code height points (8 points/mm)	Bar code height points (8 points/mm)

### Response

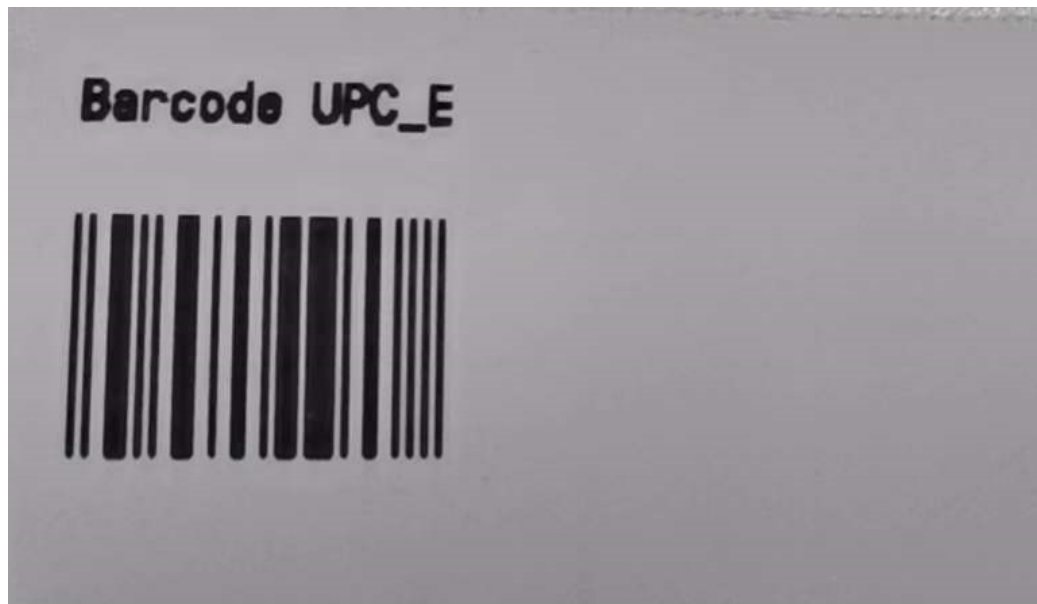
Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 0;
8          printer.cpcl_drawText(8, y, "Barcode UPC_E", 24,0,0,false,false );
9          y += 60;
10         printer.cpcl_drawBarCode(8, y, "0402931",7, 0, 3,100);
11         printer.cpcl_print(0, 0);
12     }
13 }).start();

```



## 9.7. 1D Barcode Directive

### Overview

Add instruction to print 1D barcode

### Name

`cpcl_drawBarCode(int start_x, int start_y, int end_x, int end_y, String text, int type, int rotate, int linewidth, int height, int direction)`

### Parameter

Parameter	Name	Extra
start_x	The X-axis coordinate where the barcode starts	The X-axis coordinate where the barcode starts
start_y	The Y-axis coordinate where the barcode starts	The Y-axis coordinate where the barcode starts
end_x	The X-axis coordinate of the end of the barcode	The X-axis coordinate of the end of the barcode Used to calculate, so that the barcode is displayed in the center of the confirmation area
end_y	The Y-axis coordinate of the end of the barcode	The Y-axis coordinate of the end of the barcode Used to calculate, so that the barcode is displayed in the center of the confirmation area
text	barcode data	barcode data
type	Barcode Type	0: CODE_39 1: CODE_128 2: CODE_93 3: CODABAR 4: EAN_8 5: EAN_13 6: UPC_A 7: UPC_E (When selecting EAN_8, EAN_13, UPC_A, UPC_E type, do not input the last check digit content, otherwise it cannot be printed)
rotate	barcode orientation	0: Horizontal barcode 1: Vertical barcode
linewidth	Narrow barcode width in points	Narrow barcode width in points
height	Bar code height points (8 points/mm)	Bar code height points (8 points/mm)
direction	Alignment	0: Left-aligned 1: Center-aligned 2: Right-aligned

## Response

Successful true, Failure returns false.

## Examples

```

1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXPrinter printer= new JXPrinter(mAPI);

```

```

3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 0;
8          printer.cpcl_drawText(8, y, "Barcode", 24,0,0,false,false );
9          y += 60;
10         printer.cpcl_drawBarCode(8, y, 308,y+200,"0123456789",1, 0, 3,80,2)
11         printer.cpcl_print(0, 0);
12     }
13 }).start();

```



## 9.8. 1D Barcode Directive (with text)

### Overview

Add instruction to print 1D barcode

### Name

cpcl\_drawBarCode(int start\_x, int start\_y, String text, int type, int rotate, int lineWidth, int height, int fontSize, int topMargin)

### Parameter

Parameter	Name	Extra
start_x	The X-axis coordinate where the barcode starts	The X-axis coordinate where the barcode starts
start_y	The Y-axis coordinate where the barcode starts	The Y-axis coordinate where the barcode starts
end_x	The X-axis coordinate of the end of the barcode	The X-axis coordinate of the end of the barcode Used to calculate, so that the barcode is displayed in the center of the confirmation area
end_y	The Y-axis coordinate of the end of the barcode	The Y-axis coordinate of the end of the barcode Used to calculate, so that the barcode is displayed in the center of the confirmation area
text	barcode data	barcode data
type	Barcode Type	0: CODE_39 1: CODE_128 2: CODE_93 3: CODABAR 4: EAN_8 5: EAN_13 6: UPC_A 7: UPC_E (When selecting EAN_8, EAN_13, UPC_A, UPC_E type, do not input the last check digit content, otherwise it cannot be printed)
rotate	barcode orientation	0: Horizontal barcode 1: Vertical barcode
linewidth	Narrow barcode width in points	Narrow barcode width in points
fontSize	font size: point	Print barcode content below barcode Font size Font size: point
topMargin	The number of points from the barcode content to the barcode	The number of points from the barcode content to the barcode

### Response

Successful true, Failure returns false.

### Examples



```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 0;
8          printer.cpcl_drawText(8, y, "Barcode with text", 24,0,0,false,false
9          y += 60;
10         printer.cpcl_drawBarCode(8, y, "0123456789",1, 0, 3,80,24,0);
11         printer.cpcl_print(0, 0);
12     }
13 }).start();

```



## 9.9. QR Code Directive

### Overview

Add instruction to print QR barcode

### Name

cpcl\_drawQrCode(int start\_x, int start\_y, String text, int rotate, int ver, int lel)

### Parameter

Parameter	Name	Extra
start_x	The X-axis coordinate where the barcode starts	The X-axis coordinate where the barcode starts
start_y	The Y-axis coordinate where the barcode starts	The Y-axis coordinate where the barcode starts
text	barcode data	barcode data
ver		
lcl	height/width module points	The number of module points for height/width, the value range is 1–32, the default is 6

### Response

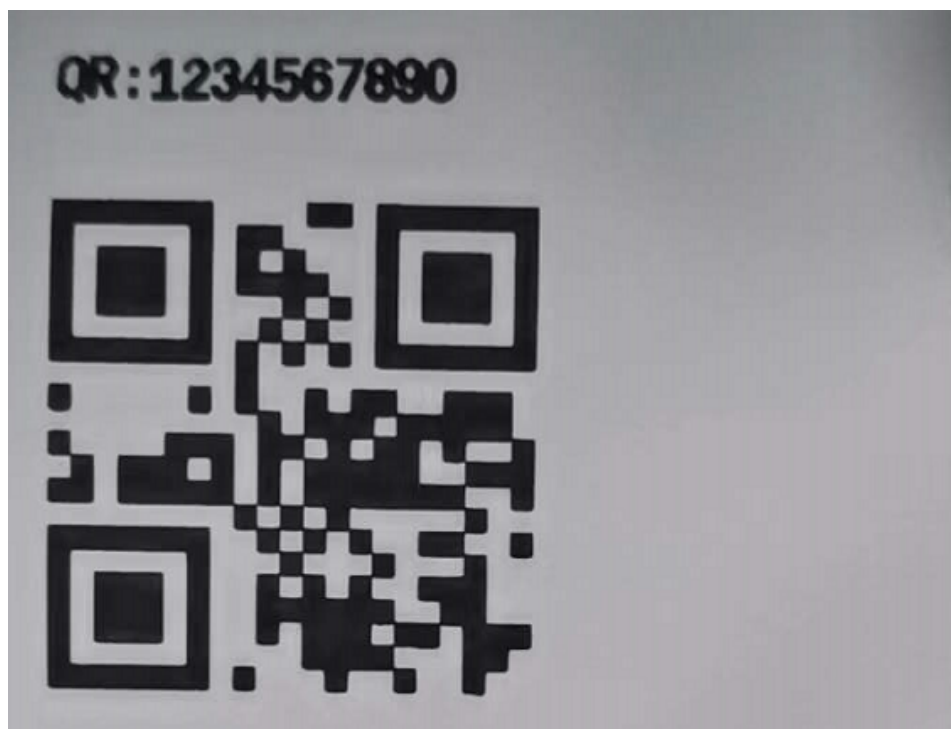
Successful true, Failure returns false.

### Examples

```

1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXPrinter printer= new JXPrinter(mAPI);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.cpcl_pageSetup(576, 400, 0);
7          int y = 0;
8          printer.cpcl_drawText(8, y, "QR_CODE;content : 1234567890\n", 24,0,0,
9          y += 60;
10         printer.cpcl_drawQrCode(8, y, "1234567890", 1, 9);
11         printer.cpcl_print(0, 0);
12     }
13 }).start();

```



## 9.10. Cartographic Directive

### Overview

Bitmap print command, which can print black and white bitmaps

### Name

`cpcl_drawGraphic(int start_x, int start_y, int bmp_size_x, int bmp_size_y, Bitmap bmp)`

### Parameter

Parameter	Name	Extra
start_x	The x-coordinate of the start of the bitmap	The x-coordinate of the start of the bitmap
start_y	Y-coordinate of the start of the bitmap	Y-coordinate of the start of the bitmap
bmp_size_x	The width of the bitmap to be printed	The width of the bitmap to be printed
bmp_size_y	The height of the bitmap to be printed	The height of the bitmap to be printed
bmp	Bitmap to be printed	Bitmap to be printed

### Response

Successful true, Failure returns false

### Examples

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 JXPrinter printer= new JXPrinter(mAPI);
3 new Thread(new Runnable() {
4     @Override
5     public void run() {
```

```

6      try {
7          printer.cpcl_pageSetup(576, 400, 0);
8          InputStream is = getResources().getAssets().open("logo.jpeg");
9          Bitmap bmp = BitmapFactory.decodeStream(is);
10         printer.cpcl_drawGraphic(8, 30, bmp.getWidth(), bmp.getHeight(), b
11         printer.cpcl_print(0, 0);
12     } catch (Exception e) {
13         e.printStackTrace();
14     }
15 }
16 }).start();

```



## 9.11. Navigate to the label

### Overview

定Navigate to the label

### Name

cpcl\_feed()

### Response

Successful true, Failure returns false.

## 9.12. Send Directive

### Overview

Send Directive

### Name

cpcl\_writeCmd(String cmd)

### Response

Successful true, Failure returns false.

