

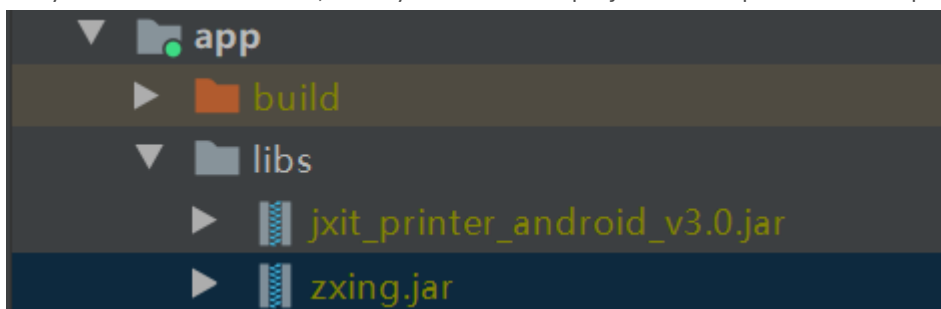
Android MY58 SDK Standard Developme...

1. SDK project integration configuration

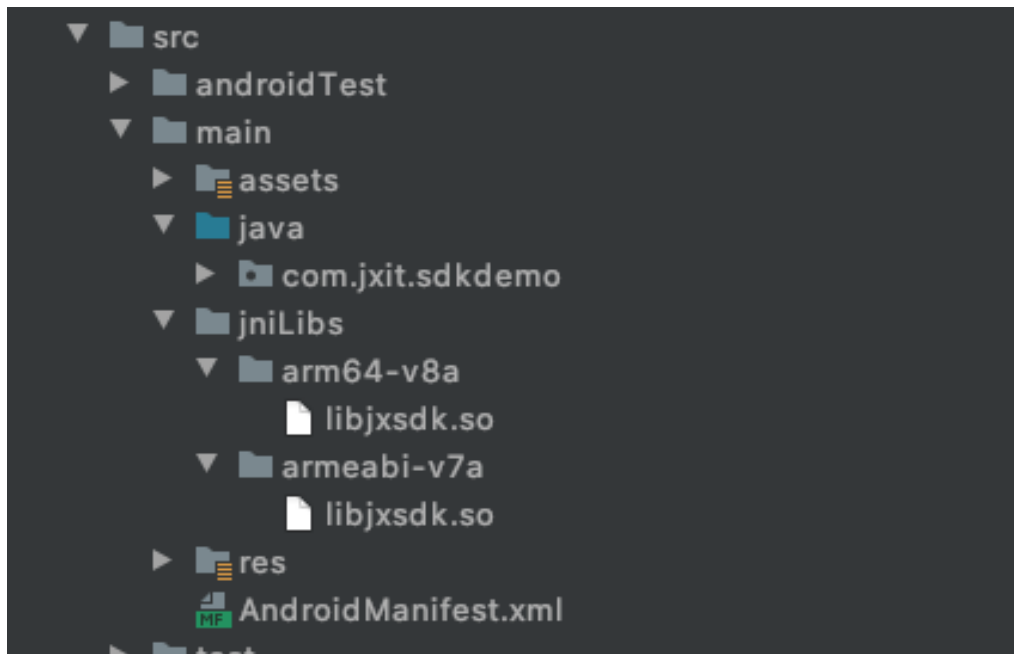
You can use the jixin Android SDK in your application in the following steps:

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 JNILIBS 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();
        //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 pair/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(getContext());
```

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.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);

```

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 value	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	<p>Points (8 dot/mm) 0–65535</p> <p>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</p>

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	<p>Points (8 dot/mm) 0–65535</p> <p>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</p>

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_setFontItalic(boolean isItalic) (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

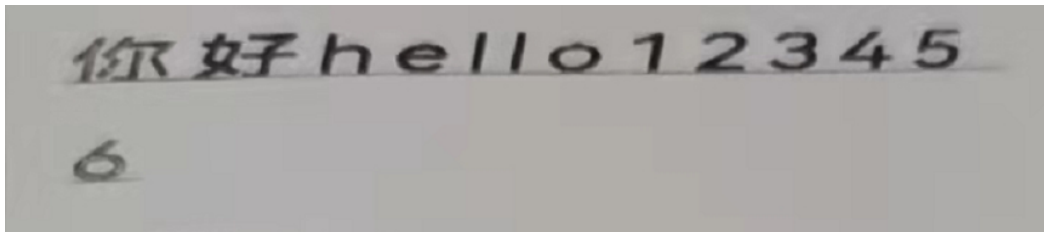
None

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

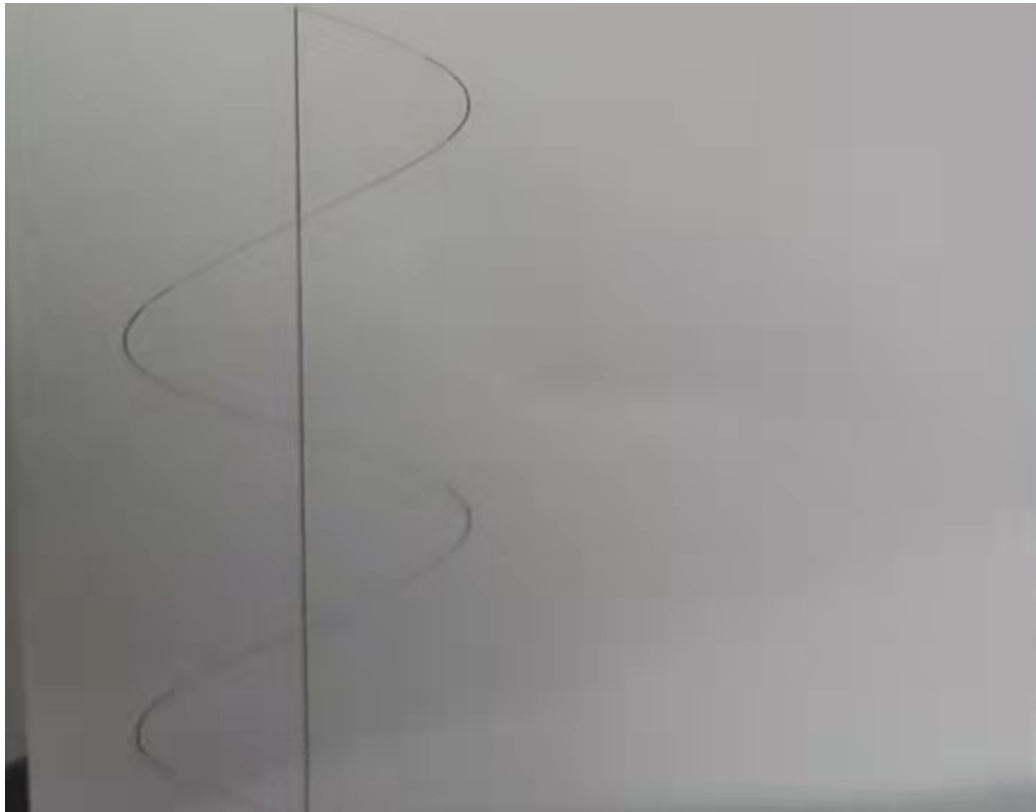
Successful true, Failure returns false.

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.

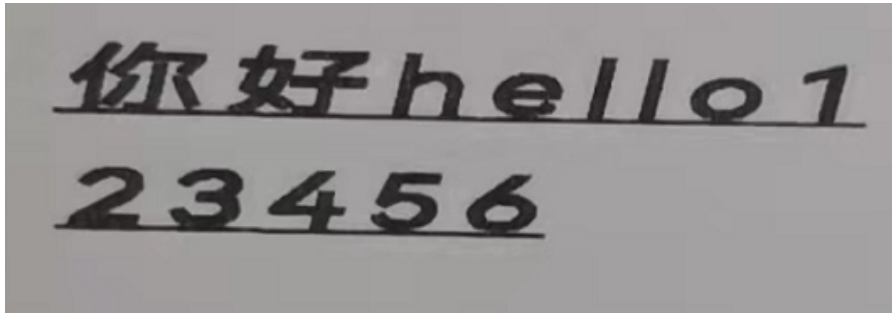
Examples

```
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

```

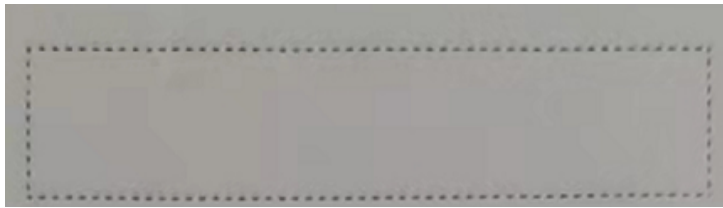
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode.Page);
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.page_setup(576, 800,0);

```

```

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.PrintMo
  de.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.PrintMo
  de.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.