

Android MY58 SDK 标准 开发指南-V5.0

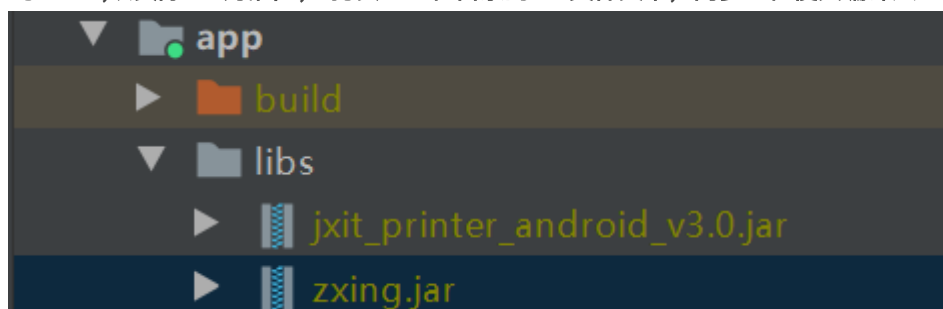
1. SDK项目集成配置

通过以下几步操作，即可在您的应用中使用捷昕android SDK：

1.1. 集成sdk

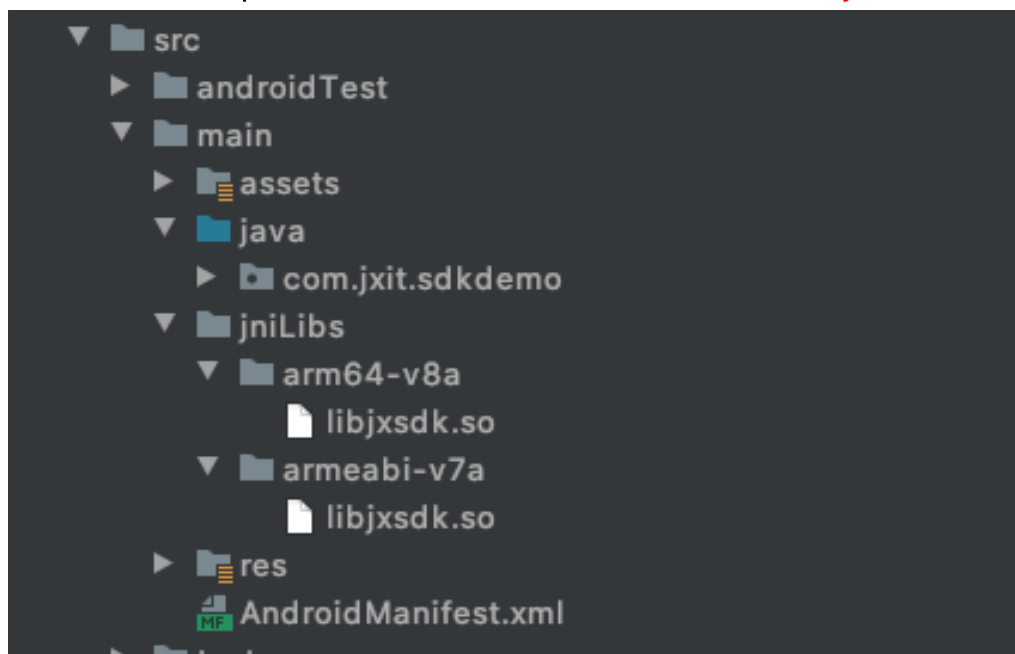
1.1.1. 添加sdk

将jxit_printer_android_v3.0.jar， **zxing.jar**（未使用global sdk理论上不需要集成zxing）（参考demo,以实际sdk为准）， 拷贝至工程目录的libs文件夹下，同步工程使其编译入工程。



1.1.2. 添加 Jni

将jniLibs文件夹拷贝到src/main/目录下，和java文件夹同级。jniLibs提供了armeabi-v7a和arm64-v8a两种主流cpu架构的支持so（未使用解锁、流控相关功能无需集成jni）



1.2. 配置AndroidManifest权限

1.2.1 集成usb打印机在<manifest>中添加如下

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

注：使用Usb通信需要使用OTG线，并使手机处于Usb Host模式。

MainActivity在AndroidManifest.xml需要配置如下：

```
<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>
```

<activity>标签中添加

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

<intent-filter>标签中添加

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

res文件夹下创建xml文件夹，并copy sdkdemo对应qwm_usb_xml.xml文件，详见sdk demo或参考Android Usb Host模式开发指南。

1.2.2 集成蓝牙打印机在<manifest>中添加如下

```
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" />
```

注：自Android6.0起部分权限的使用需要开发者在代码中动态申请。Android 12起蓝牙权限系统引入了新的运行时权限 BLUETOOTH_SCAN、BLUETOOTH_ADVERTISE 和 BLUETOOTH_CONNECT权限

1.2.3 集成WiFi打印机在<manifest>中添加如下

```
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. 初始化sdk

新建一个自定义的Application，在其onCreate方法中完成SDK的初始化。示例代码如下：

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

在AndroidManifest.xml文件中声明该Application

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

1.4. 混淆sdk

如果您的应用使用了代码混淆，请添加如下配置，以避免SDK被错误混淆导致SDK不可用

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

至此，sdk集成配置已经完成。

2. 蓝牙连接打印机

2.1. 一般使用流程

(1) 搜索蓝牙打印机 (2) 打印机蓝牙配对 (3) 建立蓝牙连接 (4) 发送打印机指令

2.2. 获取JXBluetoothAPI全局单例

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

2.3. 获取连接状态

功能说明

获取当前蓝牙连接状态

方法名称

isConnected()

返回参数

已连接返回true,未连接则返回false。

2.4. 开启/关闭蓝牙搜索，搜索蓝牙设备

功能说明

开启/关闭蓝牙搜索，搜索蓝牙设备

方法名称

startDiscovery()

cancelDiscovery()

返回参数

搜索蓝牙设备，需要监听系统蓝牙广播，action为BluetoothDevice.ACTION_FOUND，详见 sdkdemo#BluetoothFragment或参考Android蓝牙开发指南。

2.5. 获取已配对蓝牙设备列表

功能说明

获取已配对蓝牙设备列表

方法名称

getBondedDevices()

返回参数

set<BluetoothDevice>

2.6. 蓝牙配对/取消配对

功能说明

蓝牙配对/取消配对

方法名称

createBTBond(BluetoothDevice device)

removeBTBond(BluetoothDevice device)

返回参数

打印机默认配对密码为 0000，BluetoothDevice.ACTION_BOND_STATE_CHANGED，详见 sdkdemo#BluetoothFragment或参考Android蓝牙开发指南。

2.7. 蓝牙连接/断开

功能说明

蓝牙连接/断开

方法名称

openConnection(String macAddress)

closeConnection()

返回参数

成功返回true,否则返回false。

3. Usb连接打印机

3.1. 一般使用流程

(1)手机用OTG线以主机模式连接和打印机 (2) 建立Usb连接 (3) 发送打印机指令

3.2. 获取JXUsbAPI全局单例

```
1 JXUsbAPI mApi = JXUsbAPI.getDefault(getContext());
```

3.3. 获取连接状态

功能说明

获取当前蓝牙连接状态

方法名称

isConnected()

返回参数

已连接返回true,未连接则返回false。

3.4. 获取已连接的Usb设备

功能说明

获取已连接的Usb设备

方法名称

getDevices()

返回参数

List<UsbDevice>

3.5. Usb连接/断开

功能说明

Usb连接/断开

方法名称

openConnection(UsbDevice device)

closeConnection()

返回参数

成功返回true,否则返回false。

注：使用Usb通信需要使用OTG线，并使手机处于Usb Host模式。

MainActivity在AndroidManifest.xml需要配置如下：

```
<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>
```

<activity>标签中添加

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

<intent-filter>标签中添加

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

res文件夹下创建xml文件夹，并copy sdkdemo对应qwm_usb_xml.xml文件，详见sdk demo或参考Android Usb Host模式开发指南。

4. 获取打印机实例

4.1. 获取蓝牙打印机实例

```
1 JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2 //获取行模式打印机实例
3 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
de.Line);
4 //获取页模式打印机实例
5 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
de.Page);
6 //获取其他指令打印机实例
7 JXPrinter printer= new JXPrinter(mAPI);
```

4.2. 获取Usb打印机实例

```
1 JXInterfaceAPI mApi = JXUsbAPI.getDefault(context);
2 //获取行模式打印机实例
3 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
de.Line);
4 //获取页模式打印机实例
```

```
5 JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMo
   de.Page);
6 //获取其他指令打印机实例
7 JXPrinter printer= new JXPrinter(mAPI);
```

5. 打印机设置

5.1. 获取打印黑度

功能说明

获取打印机的打印黑度

方法名称

get_print_darkness()

参数说明

无

返回参数

参数	名字	说明
value	打印机黑度值	打印黑度 0~4

5.2. 设置打印机黑度

功能说明

设置打印机的打印黑度

方法名称

set_print_darkness(String value)

参数说明

参数	名字	说明
value	打印机黑度值	打印黑度 0~4

返回参数

成功则返回true,失败则返回false。

5.3. 获取打印质量

功能说明

获取打印机的打印质量

方法名称

get_print_quality()

参数说明

无

返回参数

参数	名字	说明
value	打印机质量值	打印质量 0~2

5.4. 设置打印机质量

功能说明

设置打印机的打印黑度

方法名称

set_print_quality(String value)

参数说明

参数	名字	说明
value	打印机质量	打印质量 0~2

返回参数

成功则返回true,失败则返回false。

5.5. 获取打印机状态

功能说明

获取打印机当前状态

方法名称

get_printer_status()

参数说明

无

返回参数

参数	名字	说明
JXPrinterStatus	打印机状态	JXPrinterStatus.isReady//空闲 JXPrinterStatus.isPrinting//忙碌（打印中） JXPrinterStatus.isCoverOpened//开盖 JXPrinterStatus.isNoPaper//缺纸 JXPrinterStatus.isOverHeat//过热 JXPrinterStatus.isBatteryLow//电量低 JXPrinterStatus.isUnknownError//未定义错误

5.6. 打印自检页

功能说明

让打印机打印自检页

方法名称

print_check_page()

参数说明：无

返回参数

成功则返回true,失败则返回false。

5.7. 解锁打印机（定制功能）

功能说明

对于定制打印机在连接时进行打印机解锁，否则不能使用（使用此功能需要集成 jni）

方法名称

set_print_unlock()

参数说明：无

返回参数

成功则返回true,失败则返回false。

6. Global LINE指令

Global LINE指令支持多国语言文字打印。使用Global LINE指令需要额外依赖zxing.jar。指令方法均以line_开头。

- 1 一般用于连续纸的打印，步骤：
- 2 1、获取打印服务单例对象
- 3 2、调用设置打印机最大打印宽度方法设置打印宽度
- 4 3、调用打印文字或图片等方法设置打印内容，如果内容宽度超出一行的宽度就会打印出来，不足一行的会缓存起来，和后面要打印的内容拼接到一起，也可以调用强制打印缓存区的方法，把缓存内容强制打印出来。

6.1. 设置最大打印宽度

功能说明

根据打印机的实际打印宽度设置，默认384。创建JXGlobalPrinter()实例后需要优先设置，设置一次即可。(打印前必须设置)

方法名称

line_setMaxPrintWidth(int maxWidth)

参数说明

参数	名字	说明
maxWidth	打印机最大宽度点数	页面宽度点数（8点/mm）

返回参数

无。

6.2. 清除缓存区数据

功能说明

清除缓存区数据所有数据恢复默认值。

方法名称

line_reset()

返回参数

成功则返回true,失败则返回false。

6.3. 走纸

功能说明

向前走纸。

方法名称

line_feed(int n)

参数说明

参数	名字	说明
n	向前走纸点	点数（8点/mm）

返回参数

成功则返回true,失败则返回false。

6.4. 打印控制内容指令

功能说明

打印并清空缓存区内容。

方法名称

line_print()

返回参数

成功则返回true,失败则返回false。

6.5. 设置左边距

功能说明

设置打印的左边距，设置后除非主动取消，否则一直生效

方法名称

line_setLeftMargin(int margin)

参数说明

参数	名字	说明
margin	边距	点数（8点/mm） 0-65535 需要考虑左右边距，如果同时设置左右边距当可打印区域<100时设置失败

返回参数

成功则返回true,失败则返回false。

6.6. 设置右边距

功能说明

设置打印的右边距，设置后除非主动取消，否则一直生效

方法名称

line_setRightMargin(int margin)

参数说明

参数	名字	说明
margin	边距	点数（8点/mm） 0-65535 需要考虑左右边距，如果同时设置左右边距当可打印区域<100时设置失败

返回参数

成功则返回true,失败则返回false。

6.7. 设置相对打印位置

功能说明

设置相对打印位置，设置后除非主动取消，否则一直生效

方法名称

line_setAlign(int align)

参数说明

参数	名字	说明
align	位置	1居中，2居右，其它居左

返回参数

成功则返回true,失败则返回false。

6.8.设置打印文字字体属性

功能说明

对打印字体进行设置，设置后除非主动取消，否则一直生效，需要在打印文字前设置

方法名称

line_setFontTypeface(Typeface typeface)（设置打印字体）

line_setFontUnderline(boolean isUnderline)（设置文字是否有下划线）

line_setFontBold(boolean isBold)（设置文字是否加粗）

line_setFontItalic(boolean isItalic)（设置文字是否斜体）

line_setFontSize(int fontSize)（设置字体大小）

line_setFontScale(int widthScale, int heightScale)（设置字体放大倍数）

line_setLineSpacing(int lineSpacing)（设置行距）

line_setLetterSpacing(int letterSpacing)（设置字符间距）

参数说明

方法名称	参数	名字	说明
line_setFontTypeface()	typeface	字体	设置字体样式，若为null，则清除字体样式
line_setFontUnderline()	isUnderline	下划线模式	true：带下划线； false：不带下划线；
line_setFontBold()	isBold	粗体模式	false：取消粗体模式； true：设置为粗体模式；
line_setFontItalic()	isItalic	斜体模式	false：取消斜体模式； true：设置为斜体模式；
line_setFontSize()	fontSize	字体大小	打印字体的高度，1mm = 8 dot
line_setFontScale()	widthScale	宽度放大倍数	字体放大宽度倍数1-10
line_setFontScale()	heightScale	高度放大倍数	字体放大高度倍数1-10
line_setFontScale()	lineSpacing	行距大小	行距大小，1mm = 8 dot
line_setLetterSpacing()	letterSpacing	字符间距	字符间距，1mm = 8 dot

返回参数

无

6.9. 设置反色

功能说明

设置反色，设置后除非主动取消，否则一直生效

方法名称

line_setFontReverse(boolean isReverse)

参数说明

参数	名字	说明
isReverse	黑白反显模式	true：设置黑白反显模式; false：取消黑白反显模式;

返回参数

无

示例

```
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); //设置反色
9         printer.line_drawText("你好hello123456\n");
10    }
11 }).start();
```



6.10. 绘制文字

功能说明

绘制文字，满一行自动打印，可以使用\n控制换行，可调用line_print可强制打印缓冲区文字

方法名称

line_drawText(String content)

参数说明

参数	名字	说明
content	内容	满一行自动打印 可以使用\n换行打印 调用{line_print()}可强制打印

返回参数

无

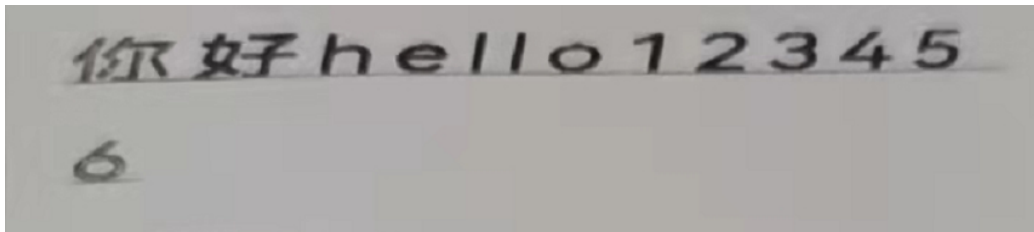
示例

```
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();//清除缓存
8          printer.line_setFontBold(true);//设置文字是否加粗
9          printer.line_setFontUnderLine(true);//设置下划线
10         printer.line_setFontItaly(true);//设置斜体
11         printer.line_setFontSize(32);//设置字体大小
12         printer.line_setFontScale(2, 1);//设置字体倍宽
13         printer.line_setLineSpacing(8);//设置行距
14         printer.line_setLetterSpacing(8);//设置字体间距
15         printer.line_drawText("你好hello123456");
16         printer.line_print();
17     }
18 }).start();

```



6.11. 线段指令

功能说明

添加指令画一条线段

方法名称

line_drawLine(float lineLength, int lineWidth, boolean fullLine)

参数说明

参数	名字	说明
lineLength	线段长度	线条长度，如果超出最大打印长度，只打印可打印部分，可调用{line_print()}可强制打印缓冲区
lineWidth	线段的线宽	线段的线宽
fullLine	线段类型	true: 实线 false: 虚线

返回参数

成功则返回true,失败则返回false。

示例

```

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();//清除缓存
8          printer.line_drawLine(80, 2, true);
9          printer.line_print();
10     }
11 }).start();

```



6.12. 一维条码指令

功能说明

添加一维条码 默认不显示文字。如果需要打印文字，需要先调用ine_setBarcode1DText
可调用line_print可强制打印缓冲区

方法名称

line_drawBarcode1D(int width, int height, String content, int rotate, BarcodeFormat
format)

参数说明

参数	名字	说明
width	条码宽度	条码宽度点数 (8点/mm)
height	条码高度	条码高度点数 (8点/mm)
content	条码数据	条码数据
rotate	顺时针旋转角度	90: 顺时针旋转90度; 180: 顺时针旋转180度;
format	条码类型	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;

返回参数

成功则返回true,失败则返回false。

示例

```

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();//清除缓存
8          printer.line_drawBarcode1D( 300, 100, "236563666", 0,
9              BarcodeFormat.CODE_128);
10         printer.line_print();
11     }
12 }).start();

```



6.13. 打印条码文字

功能说明

打印条码的文字，仅生效一次,可调用line_drawBarcode1D使用

方法名称

line_setBarcode1DText(boolean isPositionBelow, int fontSize, boolean isBold, int margin, boolean breakLine)

参数说明

参数	名字	说明
isPositionBelow	文字在下方	true: 下方 false: 上方
fontSize	字体大小	打印字体的高度, 1mm = 8 dot
isBold	粗体模式	false: 取消粗体模式; true: 设置为粗体模式;
margin	文字上下间距	行距, 1mm = 8 dot
breakLine	设置条码最大宽度	false: 打印条码以文字和条码的最大宽度为宽度, 文字和条码居中展示; true: 以条码宽度为最大宽度, 当文字宽度小于条码宽度时, 不展示后续文字;

返回参数

无

示例

```

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();//清除缓存
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. QR 条码指令

功能说明

添加指令打印QR条码 ,可调用line_print可强制打印缓冲区

方法名称

line_drawBarcode2D(int width, int height, String content, int rotate, BarcodeFormat format)

参数说明

参数	名字	说明
width	二维码宽度	条码宽度点数 (8点/mm)
height	二维码高度	条码高度点数 (8点/mm)
content	条码数据	条码数据
rotate	顺时针旋转角度	90: 顺时针旋转90度; 180: 顺时针旋转180度;
format	条码类型	BarcodeFormat.AZTEC, BarcodeFormat.PDF_417, BarcodeFormat.QR_CODE, BarcodeFormat.DATA_MATRIX,

返回参数

成功则返回true,失败则返回false。

示例

```

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();//清除缓存
8          printer.line_drawBarcode2D( 80, 80, "1234567890", 0,
9              BarcodeFormat.QR_CODE);
10         printer.line_print();
11     }
12 }).start();

```



6.15. 制图指令

功能说明

位图打印指令，该指令可以打印黑白位图

方法名称

line_drawGraphic(Bitmap bitmap)

参数说明

参数	名字	说明
bmp	待打印位图	待打印位图

返回参数

成功则返回true,失败则返回false。

示例

```

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

```

6.16. 曲线

功能说明

打印曲线的线段（曲线是由连续的线段组成的，点可以理解为长度为1的线段）

方法名称

line_drawLineSegments(int... xPos)

参数说明

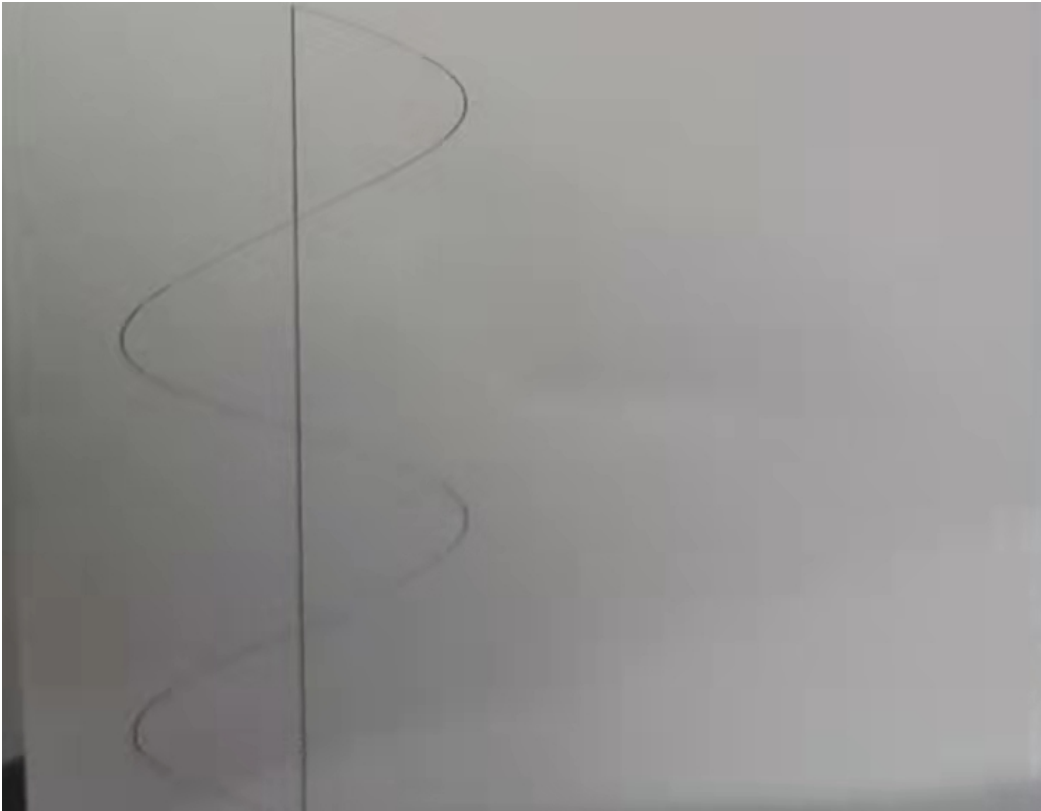
参数	名字	说明
xPos	点	x坐标数组，每2个数表示一个点或线段，x取值0-243。 如果仅打印一个点X，即参数传 X, X。 如果打印一个线段X，即传入Xs, Xe。 如果在同一水平线上打印多个线段X1..Xn，即传入Xs1,Xe1,Xs2,Xe2...Xsn,Xen。

返回参数

成功则返回true,失败则返回false。

示例

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode
3  new Thread(new Runnable() {
4      @Override
5      public void run() {
6          printer.line_setMaxPrintWidth(576);
7          printer.line_reset();//清除缓存
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指令

Global PAGE指令支持多国语言文字打印，使用Global PAGE指令需要额外依赖zxing.jar。
PAGE 指令方法均以page_开头。

- 1 一般用于标签纸的打印，步骤：
- 2 1、获取打印服务单例对象
- 3 2、调用设置打印纸张大小方法设置打印区域
- 4 3、调用打印文字或图片等方法设置打印内容
- 5 4、调用执行打印方法进行打印

7.1. 进入控制指令

功能说明

让打印机进入命令控制状态，并设置打印宽度，高度，配合page_print使用(打印前必须设置)

方法名称

page_setup(int width, int height, int paperType)

参数说明

参数	名字	说明
width	页面宽度点数	页面宽度点数（8点/mm）
height	页面高度点数	页面最大高度点数（8点/mm）
paperType	纸张类型	0 间隙纸 1 连续纸 2黑标纸

返回参数

成功则返回true,失败则返回false。

7.2. 打印控制内容指令

功能说明

使打印机控制状态结束，并打印出控制部分的内容。（发送完内容后必须调用此方法，否则不会打印发送的指令）

方法名称

page_print(boolean isUpsideDown, int times)

page_print()

参数说明

参数	名字	说明
isUpsideDown	打印顺序	true：上下180颠倒打印; false: 正向打印; 默认 false
times	打印份数	1-255 默认 1

返回参数

成功则返回true,失败则返回false。

7.3. 清除文字设置

功能说明

清除之前的文字设置

方法名称

page_resetFontSettings()

返回参数

无

7.4. 设置打印文字字体属性

功能说明

对打印字体进行设置，设置后除非主动取消，否则一直生效，需要在打印文字前设置

方法名称

page_setFontTypeface(Typeface typeface)（设置打印字体）

page_setFontUnderline(boolean isUnderline)（设置文字是否有下划线）

page_setFontBold(boolean isBold)（设置文字是否加粗）

page_setFontItalic(boolean isItalic)（设置文字是否斜体）

page_setFontSize(int fontSize)（设置字体大小）

page_setFontScale(int widthScale, int heightScale)（设置字体放大倍数）

page_setLineSpacing(int lineSpacing)（设置行距）

page_setLetterSpacing(int letterSpacing)（设置字符间距）

参数说明

参数	名字	说明
typeface	字体	设置字体样式，若为null，则清除字体样式
isUnderline	下划线模式	true：带下划线； false：不带下划线；
isBold	粗体模式	false：取消粗体模式； true：设置为粗体模式；
isItalic	斜体模式	false：取消斜体模式； true：设置为斜体模式；
fontSize	字体大小	打印字体的高度，1mm = 8 dot
widthScale	宽度放大倍数	字体放大宽度倍数1-10
heightScale	高度放大倍数	字体放大高度倍数1-10
lineSpacing	行距大小	行距大小，1mm = 8 dot
letterSpacing	字符间距	字符间距，1mm = 8 dot

返回参数

无

7.5. 设置反色

功能说明

设置反色，设置后除非主动取消，否则一直生效

方法名称

page_setFontReverse(boolean isReverse)

参数说明

参数	名字	说明
isReverse	黑白反显模式	true: 设置黑白反显模式; false: 取消黑白反显模式;

返回参数

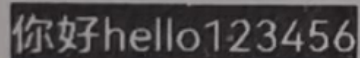
无

示例

```

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();//清除之前设置的文字信息
8          printer.page_setFontReverse(true);//设置反色
9          printer.page_drawText(10, 210, 400, 700, 0, false,
10             "你好hello123456");
11          printer.page_print();
12      }
13  }).start();

```



7.6. 文字指令

功能说明

打印传参文字，支持多国语言。配合page_print使用

方法名称

page_drawText(int start_x, int start_y, int end_x, int end_y, int rotate, boolean singleLine, String content)

参数说明

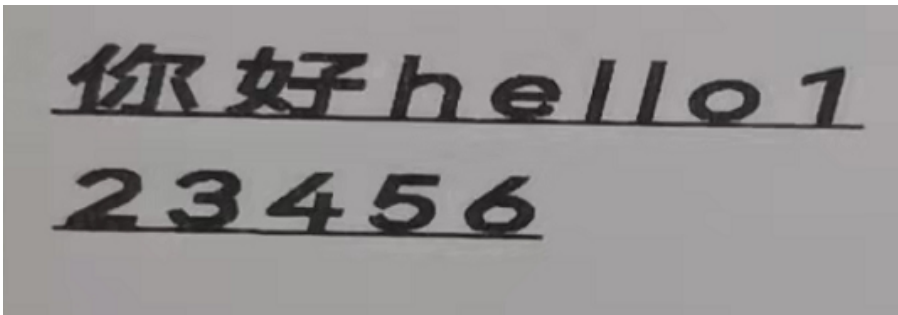
参数	名字	说明
start_x	X轴起始位置	X轴起始位置
start_y	Y轴起始位置	Y轴起始位置
end_x	X轴结束位置	文本打印区域X轴结束位置
end_y,	Y轴结束位置	文本打印区域Y轴结束位置
content	内容	打印文字内容
rotate	顺时针旋转角度	90: 顺时针旋转90度; 180: 顺时针旋转180度; 270: 顺时针旋转170度;
singleLine	单行模式	true: 只打印一行, 多余截断 false: 自动换行

返回参数

成功则返回true,失败则返回false。

示例

```
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();//清除之前设置的文字信息
8         printer.page_setFontUnderline(true);// ( 设置文字是否有下划线 )
9         printer.page_setFontBold(true);// ( 设置文字是否加粗 )
10        printer.page_setFontItaly(true);// ( 设置文字是否斜体 )
11        printer.page_setFontSize(32); // ( 设置字体大小, 默认24 )
12        printer.page_setFontScale(2, 1);// ( 设置字体放大 )
13        printer.page_setLineSpacing(16);// ( 设置行距 )
14        printer.page_setLetterSpacing(8);// ( 设置字符间距 )
15        printer.page_drawText(10, 210, 400, 700, 0, false,
16        "你好hello123456");
17        printer.page_print();
18    }
19 }).start();
```



7.7. 矩形框指令

功能说明

添加指令画一个矩形框

方法名称

page_drawBox(int start_x, int start_y, int width, int height, int lineWidth, boolean fullLine)

参数说明

参数	名字	说明
start_x	左上角X坐标	左上角X坐标
start_y	左上角Y坐标	左上角Y坐标
width	矩形宽度	点宽
height	矩形高度	点高
lineWidth	形成矩形的线宽	形成矩形的线宽
fullLine	线段类型	true: 实线 false: 虚线

返回参数

成功则返回true,失败则返回false。

示例

```
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. 线段指令

功能说明

添加指令画一条线段

方法名称

page_drawLine(int start_x, int start_y, int end_x, int end_y, int lineWidth, boolean fullLine)

参数说明

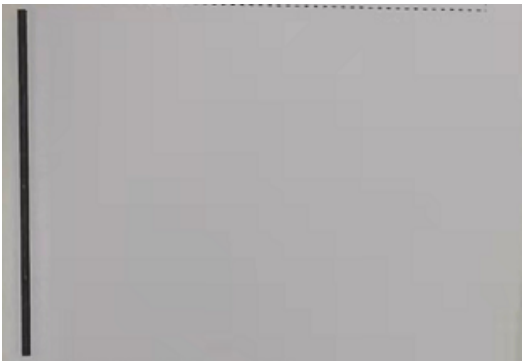
参数	名字	说明
lineWidth	线段的线宽	线段的线宽
start_x	左上角X坐标	左上角X坐标
start_y	左上角Y坐标	左上角Y坐标
end_x	右下角X坐标	右下角X坐标
end_y	右下角Y坐标	右下角Y坐标
fullLine	线段类型	true: 实线 false: 虚线

返回参数

成功则返回true,失败则返回false。

示例

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode
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. 一维条码指令

功能说明

添加一维条码 默认不显示文字。如果需要打印文字，需要先调用page_setBarcode1DText

方法名称

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

参数说明

参数	名字	说明
start_x	条码开始的X轴坐标	条码开始的X轴坐标
start_y	条码开始的Y轴坐标	条码开始的Y轴坐标
width	条码宽度	条码宽度点数 (8点/mm)
height	条码高度	条码高度点数 (8点/mm)
content	条码数据	条码数据
rotate	顺时针旋转角度	90: 顺时针旋转90度; 180: 顺时针旋转180度;
format	条码类型	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;

返回参数

成功则返回true,失败则返回false。

示例

```

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_drawBarcode1D(10, 10, 300, 100, "236563666", 0,
8              BarcodeFormat.CODE_128);
9          printer.page_print();
10     }
11 }).start();

```



7.10. 打印条码文字

功能说明

打印条码的文字，仅生效一次，配合page_drawBarcode1D使用

方法名称

page_setBarcode1DText(boolean isPositionBelow, int fontSize, boolean isBold, int margin, boolean breakLine)

参数说明

参数	名字	说明
isPositionBelow	文字在下方	true: 下方 false: 上方
fontSize	字体大小	打印字体的高度，1mm = 8 dot
isBold	粗体模式	false: 取消粗体模式; true: 设置为粗体模式;
margin	文字上下间距	行距，1mm = 8 dot
breakLine	设置条码最大宽度	false: 打印条码以文字和条码的最大宽度为宽度，文字和条码居中展示; true: 以条码宽度为最大宽度，当文字宽度小于条码宽度时，不展示后续文字;

返回参数

无

示例

```
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_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. QR 条码指令

功能说明

添加指令打印QR条码

方法名称

page_drawBarcode2D(int start_x, int start_y, int width, int height, String content, int rotate, BarcodeFormat format)

参数说明

参数	名字	说明
start_x	条码开始的X轴坐标	条码开始的X轴坐标
start_y	条码开始的Y轴坐标	条码开始的Y轴坐标
width	二维码宽度	条码宽度点数（8点/mm）
height	二维码高度	条码高度点数（8点/mm）
content	条码数据	条码数据
rotate	顺时针旋转角度	90：顺时针旋转90度； 180：顺时针旋转180度；
format	条码类型	BarcodeFormat.AZTEC, BarcodeFormat.PDF_417, BarcodeFormat.QR_CODE, BarcodeFormat.DATA_MATRIX,

返回参数

成功则返回true,失败则返回false。

示例

```
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. 制图指令

功能说明

位图打印指令，该指令可以打印黑白位图

方法名称

page_drawGraphic(int start_x, int start_y, Bitmap bmp)

参数说明

参数	名字	说明
start_x	位图开始的X轴坐标	位图开始的X轴坐标
start_y	位图开始的Y轴坐标	位图开始的Y轴坐标
bmp	待打印位图	待打印位图

返回参数

成功则返回true,失败则返回false。

示例

```
1  JXInterfaceAPI mApi = JXBluetoothAPI.getDefault(context);
2  JXGlobalPrinter printer = new JXGlobalPrinter(api, JXGlobalPrinter.PrintMode
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. 定位到标签

功能说明

定位到标签，找标签缝走纸，找不到最多走10cm

方法名称

page_feed()

返回参数

成功则返回true,失败则返回false。