

Android 开发文档

(3.0)

本开示例代码是基于 android studio 3.0 开发

目录

- 目录..... 2
- 一、 简介..... 3
- 二. PrinterInstance 打印机类 3
 - 1. 构造函数 PrinterInstance: 3
 - 2. 打开与关闭连接: 4
 - 3. 常用方法: 4
 - 4. 设置方法:..... 5
- 三. Barcode 类..... 9
 - 1. 构造函数 9
 - 1.1 参数 barcodeType 为条码类型: 9
 - 1.2 参数 param1, param2, param3 为条码参数: 10
 - 1.2.1 PDF417..... 10
 - 1.2.2 DATA MATRIX 11
 - 1.2.3. QR CODE 11
 - 1.3. 参数 content 为条码数据。 11

一、简介

此开发包包括蓝牙，**USB** 和 **WiFi**。

1. 软件包名称: `com.android.print.sdk`
2. 类型库名称:

Barcode	条码打印类
PrinterConstants	相关常量
PrinterInstance	打印机类

二、PrinterInstance 打印机类

1. 构造函数 PrinterInstance:

- a) //蓝牙接口打印机构造函数

```
PrinterInstance(Context context, BluetoothDevice bluetoothDevice,  
Handler handler)
```

- b) //USB 接口打印机构造函数

```
PrinterInstance(Context context, UsbDevice usbDevice, Handler handler)
```

- c) //WiFi 接口打印机构造函数

```
PrinterInstance(String ipAddress, int portNumber, Handler handler)
```

`handler` 用于接收打印机的连接状态，对应的状态为：

PrinterConstants.Connect.SUCCESS;	//连接成功
PrinterConstants.Connect.FAILED;	//连接失败
PrinterConstants.Connect.CLOSED;	//连接关闭

2. 打开与关闭连接:

- a) `openConnection()`
- b) `closeConnection()`

3. 常用方法:

- a) 初始化打印机

`init();`

- b) 打印普通文本

`printText(String content);`

- c) 发送 Byte 数据

`sendByteData(byte[] content);`

若 SDK 中未提供开发手册中的指令, 开发者可以使用此方法直接给打印机发送标准 **ESC/POS** 指令。如发送设置打印位置居中的指令为:

```
byte[] command = new byte[3];
```

```
command[0] = 0x1B;
```

```
command[1] = 0x31;
```

```
command[2] = 49;
```

```
sendByteData (command);
```

d) 打印图片

bitmap 为图片； *width* 为图片左边距； *mode* 为放大倍数

0:正常, 1:倍宽, 2:倍高, 3:倍宽倍高

```
printImage(Bitmap bitmap);
```

```
printImage(Bitmap bitmap, int width, int mode);
```

e) 打印条码

```
PrintBarCode(Barcode barcode);
```

4. 设置方法:

a) 设置打印文本的字符编码格式， 默认为“gbk”

```
setEncoding(String encoding);
```

b) 设置字符放大倍数，参数 x 为宽，y 为高。x，y 取值为[0,7]，0 为默认宽高

```
setCharacterMultiple(int x, int y);
```

c) 设置打印区域左边距，值为 $(nL + nH * 256) * \text{横向单位}$ 。一般只传

nL, nH 传 0。

`setLeftMargin(int nL, int nH);`

d) 设置打印模式

`setPrintModel(boolean isBold, boolean isDoubleHeight, boolean isDoubleWidth, boolean isUnderLine);`

isBold: 是否粗体

isDoubleHeight: 是否倍高

isDoubleWidth: 是否倍宽

isUnderLine: 是否下划线

e) 设置打印机(命令常量以 `PrinterConstants.Command`.开头)

`setPrinter(int command);`

command 为打印机命令:

NIT_PRINTER: 初始化打印机(已提出单独方法 `init()`) *WAKE_PRINTER*: 唤醒打印机

PRINT_AND_RETURN_STANDARD: 页模式下打印并返回标准模式

PRINT_AND_NEWLINE: 打印并换行

PRINT_AND_ENTER: 打印并回车

MOVE_NEXT_TAB_POSITION: 移动打印位置到下一个水平制表位置

DEF_LINE_SPACING: 恢复默认行高

`setPrinter(int command, int value);`

value 为命令对应的值:

PRINT_AND_WAKE_PAPER_BY_LNCH: 打印并进纸 value 高度（英寸）

PRINT_AND_WAKE_PAPER_BY_LINE : 打印并走纸 value 行

CLOCKWISE_ROTATE_90: 顺时针旋转 90 度 0-false, 1-true

LINE_HEIGHT: 设置行高

CHARACTER_RIGHT_MARGIN: 字符右间距

ALIGN: 对齐模式。 3 种对齐方式变量如下:

ALIGN_LEFT 左对齐;

ALIGN_CENTER 居中;

ALIGN_RIGHT 右对齐

5. 读取数据

```
read();
```

读取打印机返回的数据。（wifi机器需要定时调用 `read()`; 读取打印机数据用于检测打印机是否断开连接，如果断开了连接会触发Handler消息）。

6. 打印机升级

`updatePrint(fileByte);`

fileByte 为打开机升级程序字节数组。

7. 注意事项

1. `PrinterInstance` 相关 API 在 WIFI 打印机实例中涉及网络通讯，开发者请在异步线程中调用。否则可能抛出：

[`android.os.NetworkOnMainThreadException`](#)

2. 图片打印，开发者应该先处理 `Bitmap`，宽度要求是 24 或 24 的倍数，如 58mm 打印机最大宽度不超过 384，否则打印会乱码。
3. 打印机升级为耗时操作，建议开发者在异步线程中执行。

三、Barcode 类

1. 构造函数

`Barcode(byte barcodeType);`

`Barcode(byte barcodeType, int param1, int param2, int param3);`

`Barcode(byte barcodeType, int param1, int param2, int param3, String content);`

1.1 参数 barcodeType 为条码类型：

类型常量以 *PrinterConstants.BarcodeType* 开头：

一维条码：

UPC_A,

UPC_E,

JAN13,

JAN8,

CODE39,

ITF,

CODABAR,

CODE93,
CODE128。

二维条码:

PDF417,
DATAMATRIX,
QRCODE。

1.2 参数 param1, param2, param3 为条码参数:

条码类型 **type** 为一维条码时, 三个参数表示:

param1: 条码横向宽度 $2 \leq n \leq 6$, 默认为 2。

param2: 条码高度 $1 \leq n \leq 255$, 默认 162

param3: 条码注释位置, 0 不打印, 1 上方, 2 下方, 3 上下方均有 条

码类型 **type** 为二维条码时, 三个参数表示不同的意思:

1.2.1 PDF417

param1: 表示每行字符数, $1 \leq n \leq 30$ 。

param2: 表示纠错等级, $0 \leq n \leq 8$ 。

param3: 表示纵向放大倍数。

1.2.2 DATA MATRIX

param1: 表示图形高, $0 \leq n \leq 144$ (0: 自动选择)。

param2: 表示图形宽, $8 \leq n \leq 144$ (*param1* 为 0 时, 无效)。

param3: 表示纵向放大倍数。

1.2.3. QR CODE

param1: 表示图形版本号, $1 \leq n \leq 30$ (0: 自动选择)。

param2: 表示纠错等级, $n = 76, 77, 81, 72$ (L: 7%, M: 15%, Q: 25%, H: 30%)。

param3: 表示纵向放大倍数。

1.3. 参数 content 为条码数据。