

Android POS 编程手册

v2.1.2

1. 介绍

这个手册介绍怎么实现票据打印。

2. POSPrinter

2.1. POSPrinter

构造函数，创建打印对象。

`POSPrinter(IDeviceConnection connection)`

【参数】

➤ `connection`

连接对象，可通过 `POSConnect.createDevice(deviceType)` 获取。

2.2. printString

这个方法用于普通文本的打印

`POSPrinter printString(String data)`

【参数】

➤ `data`

打印的文本字符串

【返回值】

`POSPrinter` 对象

2.3. printText

这个方法用于特定格式的文本打印

`POSPrinter printText(String data, int alignment, int attribute, int textSize)`

`POSPrinter printTextSize(String data, int textSize)`

`POSPrinter printTextAttribute(String data, int attribute)`

`POSPrinter printTextAlignment(String data, int alignment)`

【参数】

➤ `data`

打印的文本字符串

➤ `alignment`

文本的对齐方式，默认为 `ALIGNMENT_LEFT`

变量	描述
ALIGNMENT_LEFT	左对齐
ALIGNMENT_CENTER	居中对齐
ALIGNMENT_RIGHT	右对齐

➤ attribute

文本的属性，默认为 FNT_DEFAULT

变量	描述
FNT_DEFAULT	FontA, 标准字体
FNT_FONTB	FontB 字体
FNT_BOLD	粗字体
FNT_REVERSE	反打印属性
FNT_UNDERLINE	下划线属性
FNT_UNDERLINE2	粗下划线属性

➤ textSize

打印的文本字体大小，默认为 TXT_1WIDTH|TXT_1HEIGHT

变量(宽比例)	描述
TXT_1WIDTH	将宽度比设置为 x1
TXT_2WIDTH	将宽度比设置为 x2
TXT_3WIDTH	将宽度比设置为 x3
TXT_4WIDTH	将宽度比设置为 x4
TXT_5WIDTH	将宽度比设置为 x5
TXT_6WIDTH	将宽度比设置为 x6
TXT_7WIDTH	将宽度比设置为 x7
TXT_8WIDTH	将宽度比设置为 x8

变量(高比例)	描述
TXT_1HEIGHT	将高度比设置为 x1
TXT_2HEIGHT	将高度比设置为 x2
TXT_3HEIGHT	将高度比设置为 x3
TXT_4HEIGHT	将高度比设置为 x4
TXT_5HEIGHT	将高度比设置为 x5
TXT_6HEIGHT	将高度比设置为 x6
TXT_7HEIGHT	将高度比设置为 x7
TXT_8HEIGHT	将高度比设置为 x8

【返回值】

POSPrinter 对象

2.4. printBitmap

这个方法用于打印图片文件

POSPrinter printBitmap(String bitmapPath, int alignment, int width)
POSPrinter printBitmap(String bitmapPath, int alignment, int width, int model)
POSPrinter printBitmap(Bitmap bmp, int alignment, int width)
POSPrinter printBitmap(Bitmap bmp, int alignment, int width, int model)

【参数】

➤ bitmapPath

图片文件路径

➤ bmp

Bitmap 对象

➤ alignment

图片的对齐方式

变量	描述
ALIGNMENT_LEFT	左对齐
ALIGNMENT_CENTER	居中对齐
ALIGNMENT_RIGHT	右对齐

➤ width

打印宽度,可用于缩放图片

➤ model

打印模式

变量	描述
BMP_NORMAL	图片原始尺寸
BMP_WIDTH_DOUBLE	图片宽度双倍
BMP_HEIGHT_DOUBLE	图片高度双倍
BMP_WIDTH_HEIGHT_DOUBLE	图片宽高都双倍

【返回值】

POSPrinter 对象

2.5. printBarcode

这个方法用于打印一维条码

POSPrinter printBarcode(String data, int codeType)
POSPrinter printBarcode(String data, int codeType, int width, int height, int alignment)
POSPrinter printBarcode(String data, int codeType, int width, int height, int alignment, int
textPosition)

【参数】

➤ data

待打印的条码字符串

➤ codeType

条码类型

变量	描述
BCS_UPCA	UPC A 类型
BCS_UPCE	UPCE 类型
BCS_EAN8	EAN-8 类型
BCS_EAN13	EAN-13 类型
BCS_JAN8	JAN-8 类型
BCS_JAN13	JAN-13 类型
BCS_ITF	ITF 类型
BCS_Codabar	Codabar 类型
BCS_Code39	Code 39 类型
BCS_Code93	Code 93 类型
BCS_Code128	Code 128 类型，该类型时，data 必须加上{A, {B, {C 等

➤ height

条码高度，范围【1，255】。默认 162

➤ width

条码横向模块宽度，范围【2，6】，默认 3

➤ Alignment

条码的对齐方式，默认 ALIGNMENT_CENTER

变量	描述
ALIGNMENT_LEFT	左对齐
ALIGNMENT_CENTER	居中对齐
ALIGNMENT_RIGHT	右对齐

➤ textPosition

打印条码是，为 HRI 字符选择打印的位置。默认为 HRI_TEXT_BELOW。

变量	描述
HRI_TEXT_NONE	不打印
HRI_TEXT_ABOVE	条码上方
HRI_TEXT_BELOW	条码下方
HRI_TEXT_BOTH	条码上、下方都打印

【返回值】

POSPrinter 对象

2.6. lineFeed

这个方法用于打印并向前走纸 n 行

POSPrinter lineFeed(int lineCount)

POSPrinter lineFeed()

【参数】

➤ lineCount

走纸的行数。 默认为 1

【返回值】

POSPrinter 对象

2.7. printQRCode

这个方法用于打印二维码

POSPrinter printQRCode(String data)

POSPrinter printQRCode(String data, int alignment)

POSPrinter printQRCode(String data, int moduleSize, int alignment)

POSPrinter printQRCode(String data, int moduleSize, int ecLevel, int alignment)

【参数】

➤ data

二维码内容字符串

➤ moduleSize

单元大小，范围【1，16】，默认值 8

➤ ecLevel

错误纠正等级， 默认值为 QRCODE_EC_LEVEL_L

变量	描述
QRCODE_EC_LEVEL_L	错误纠正等级 L (7%)
QRCODE_EC_LEVEL_M	错误纠正等级 M (15%)
QRCODE_EC_LEVEL_Q	错误纠正等级 Q (25%)
QRCODE_EC_LEVEL_H	错误纠正等级 H (30%)

➤ alignment

二维码对齐方式， 默认值为 ALIGNMENT_CENTER

变量	描述
ALIGNMENT_LEFT	左对齐
ALIGNMENT_CENTER	居中对齐
ALIGNMENT_RIGHT	右对齐

【返回值】

POSPrinter 对象

2.8. cutPaper

这个方法用于切纸

POSPrinter cutPaper()

POSPrinter cutPaper(int model)
POSPrinter cutHalfAndFeed(int distance)
进纸 distance ×（纵向移动单位）英寸 并且半切纸

【参数】

➤ model

切纸模式，默认 CUT_HALF。

变量	描述
CUT_ALL	全切
CUT_HALF	半切

➤ distance

走纸距离

【返回值】

POSPrinter 对象

2.9. openCashBox

这个方法用于打开钱箱抽屉

POSPrinter openCashBox(int pinNum)
POSPrinter openCashBox(int pinNum, int onTime, int offTime)

【参数】

➤ pinNum

连接的引脚

变量	描述
PIN_TWO	引脚 2
PIN_FIVE	引脚 5

➤ onTime

钱箱产生脉冲开始时间 onTime*2ms，范围【0，255】。默认值为 30

➤ offTime

钱箱产生脉冲结束时间 offTime*2ms，范围【0，255】，默认值为 255
如果 onTime>offTime，结束时间为 onTime*2ms。

【返回值】

POSPrinter 对象

2.10. setCharSet

设置字符编码，默编码为“gbk”

void setCharSet(String charSet)

【参数】

➤ charSet
编码类型名

2.11. setTextStyle

这个方法用于设置字体信息

POSPrinter setTextStyle(int attribute, int textSize)

【参数】

➤ attribute
文本的属性

变量	描述
FNT_DEFAULT	FontA，标准字体
FNT_FONTB	FontB 字体
FNT_BOLD	粗字体
FNT_REVERSE	反打印属性
FNT_UNDERLINE	下划线属性
FNT_UNDERLINE2	粗下划线属性

➤ textSize

打印的文本字体大小

变量(宽比例)	描述
TXT_1WIDTH	将宽度比设置为 x1
TXT_2WIDTH	将宽度比设置为 x2
TXT_3WIDTH	将宽度比设置为 x3
TXT_4WIDTH	将宽度比设置为 x4
TXT_5WIDTH	将宽度比设置为 x5
TXT_6WIDTH	将宽度比设置为 x6
TXT_7WIDTH	将宽度比设置为 x7
TXT_8WIDTH	将宽度比设置为 x8

变量(高比例)	描述
TXT_1HEIGHT	将高度比设置为 x1
TXT_2HEIGHT	将高度比设置为 x2
TXT_3HEIGHT	将高度比设置为 x3
TXT_4HEIGHT	将高度比设置为 x4
TXT_5HEIGHT	将高度比设置为 x5
TXT_6HEIGHT	将高度比设置为 x6
TXT_7HEIGHT	将高度比设置为 x7

TXT_8HEIGHT	将高度比设置为 x8
-------------	------------

【返回值】

POSPrinter 对象

2.12. setAlignment

这个方法用于设置文本的对齐方式

POSPrinter setAlignment(int alignment)

【参数】

➤ alignment

文本的对齐方式

变量	描述
ALIGNMENT_LEFT	左对齐
ALIGNMENT_CENTER	居中对齐
ALIGNMENT_RIGHT	右对齐

【返回值】

POSPrinter 对象

2.13. printerCheck

这个方法用于查询打印机所有状态。

void printerCheck(int type, int timeout, IDataCallback callback)

【参数】

➤ type

变量	描述
STS_TYPE_PRINT	打印状态
STS_TYPE_OFFLINE	脱机状态
STS_TYPE_ERR	错误状态
STS_TYPE_PAPER	传送纸状态

➤ timeout

接收超时时间，单位 ms

➤ callback

读取的数据回调，回调的内容为对应的打印机状态，如果 timeout 时间内，未收到数据，则返回空字节。

```
public interface IDataCallback {
    void receive(byte[] data);
}
```

```
}
```

2.14. printerStatus

这个方法用于打印机常用状态查询，查询超时时间 5000ms

```
void printerStatus(IStatusCallback callback)
```

【参数】

➤ callback

读取的状态回调。

```
public interface IStatusCallback {  
    void receive(int status);  
}
```

对应的 status 值见下表

STS_UNKNOWN	未知状态，读取数据超时，或者接收到的数据长度不为 1
STS_NORMAL	打印机正常
STS_COVEROPEN	上盖开
STS_PAPEREMPTY	打印机缺纸

2.15. cashBoxCheck

这个方法用于钱箱状态查询。

```
void cashBoxCheck(IStatusCallback callback)
```

【参数】

➤ callback

读取的状态回调。

```
public interface IStatusCallback {  
    void receive(int status);  
}
```

对应的 status 值见下表

STS_UNKNOWN	未知状态，读取数据超时，或者接收到的数据长度不为 1
STS_CASH_OPEN	钱箱开
STS_CASH_CLOSE	钱箱关

2.16. setPrintArea

页模式下设置打印区域

POSPrinter setPrintArea(int x, int y, int width, int height)

POSPrinter setPrintArea(int width, int height)

【参数】

➤ x

横向起始位置，默认为 0

➤ y

纵向起始位置，默认为 0

➤ width

区域宽度

➤ height

区域高度

【返回值】

POSPrinter 对象

2.17. setPageModel

这个方法用于开启或者关闭页模式

POSPrinter setPageModel(boolean isOpen)

【参数】

➤ isOpen

true 表示开启页模式，false 表示关闭页模式

【返回值】

POSPrinter 对象

2.18. printPageModelData

打印页模式下的数据，并回到标准模式。

POSPrinter printPageModelData()

【返回值】

POSPrinter 对象

2.19. setPrintDirection

设置页模式下的打印区域方向

POSPrinter setPrintDirection(int direction)

【参数】

➤ direction

打印方向

变量	描述
DIRECTION_LEFT_TOP	从左上角开始往右
DIRECTION_LEFT_BOTTOM	从左下角开始往上
DIRECTION_RIGHT_BOTTOM	从右下角开始往左
DIRECTION_RIGHT_TOP	从右上角开始往下

【返回值】

POSPrinter 对象

2.20. setAbsoluteHorizontal

设置横向绝对打印位置

POSPrinter setAbsoluteHorizontal(int position)

【参数】

➤ position

开始的位置

【返回值】

POSPrinter 对象

2.21. setRelativeHorizontal

设置相对横向打印位置

POSPrinter setRelativeHorizontal(int position)

【参数】

➤ position

开始的位置

【返回值】

POSPrinter 对象

2.22. setAbsoluteVertical

页模式下设置纵向绝对位置

POSPrinter setAbsoluteVertical(int position)

【参数】

➤ position

开始的位置

【返回值】

POSPrinter 对象

2.23. setRelativeVertical

页模式下设置纵向相对位置

POSPrinter setRelativeVertical(int position)

【参数】

➤ position

开始的位置

【返回值】

POSPrinter 对象

2.24. downloadNVImage

保存 NV 图片到 flash 中

POSPrinter downloadNVImage(String imagePaths, int imageWidth)

POSPrinter downloadNVImage(List<Bitmap> bitmaps, int imageWidth)

【参数】

➤ imagePaths

图片绝对路径，不同的图片用“,” 隔开。

例如: "/storage/emulated/0/tmp/logo1.bmp,/storage/emulated/0/tmp/logo2.bmp"

➤ bitmaps

需要保存的图片 Bitmap 对象列表

➤ imageWidth

图片宽度

【返回值】

POSPrinter 对象

2.25. printNVImage

打印存储在 Flash 中的图片

POSPrinter printNVImage(int index, int model)

【参数】

➤ index

图片下标，范围【1，255】

➤ model

打印位图的模式

变量	描述
BMP_NORMAL	图片原始尺寸
BMP_WIDTH_DOUBLE	图片宽度双倍
BMP_HEIGHT_DOUBLE	图片高度双倍
BMP_WIDTH_HEIGHT_DOUBLE	图片宽高都双倍

【返回值】

POSPrinter 对象

2.26. initializePrinter

初始化打印机

POSPrinter initializePrinter()

【返回值】

POSPrinter 对象

2.27. selectBitmapModel

选择位图模式

POSPrinter selectBitmapModel(int model, byte[] bytes)

【参数】

➤ model

位图模式

变量	描述
----	----

SINGLE_DENSITY_8	8 点单密度
DOUBLE_DENSITY_8	8 点双密度
SINGLE_DENSITY_24	24 点单密度
DOUBLE_DENSITY_24	24 点双密度

➤ bytes

需要打印的位图字节数组

【返回值】

POSPrinter 对象

2.28. printAndFeed

打印缓冲区数据并走纸 n 点。

POSPrinter printAndFeed(int n)

【参数】

➤ n

走纸的距离，单位为横向或纵向移动单位。默认为点。

【返回值】

POSPrinter 对象

2.29. setLineSpacing

设置行高

POSPrinter setLineSpacing(int space)

【参数】

➤ space

行空间高度，如果想恢复到默认，则传入 SPACE_DEFAULT。

【返回值】

POSPrinter 对象

2.30. isConnected

查询连接状态

void isConnected(IStatusCallback callback)

【参数】

➤ callback

状态回调

```
public interface IStatusCallback {  
    void receive(int status);  
}
```

status	描述
STS_CONNECT	连接
STS_DISCONNECT	断开

2.31. setTurnUpsideDownMode

选择或取消倒置打印模式

POSPrinter setTurnUpsideDownMode(boolean on)

【参数】

➤ on

true 表示选择 false 表示取消

【返回值】

POSPrinter 对象

2.32. selectCodePage

选择字符代码表

POSPrinter selectCodePage(int page)

【参数】

➤ page

代码表

变量	描述
CODE_PAGE_PC437	PC437 [美国, 欧洲标准]
CODE_PAGE_KATAKANA	片假名
CODE_PAGE_PC850	PC850 [多语言]
CODE_PAGE_PC860	PC860 [葡萄牙语]
CODE_PAGE_PC863	PC863 [加拿大-法语]
CODE_PAGE_PC865	PC865 [北欧]
CODE_PAGE_WEST_EUROPE	West Europe
CODE_PAGE_GREEK	Greek
CODE_PAGE_HEBREW	Hebrew
CODE_PAGE_EAST_EUROPE	PC755:East Europe

CODE_PAGE_IRAN	Iran
CODE_PAGE_WPC1252	WPC1252
CODE_PAGE_PC866	PC866:Cyrillice*2
CODE_PAGE_PC852	PC852:Latin2
CODE_PAGE_PC858	PC858

【返回值】

POSPrinter 对象

2.33. selectCharacterFont

选择字体

POSPrinter selectCharacterFont(int font)

【参数】

➤ font

字体类型

变量	描述
FONT_STANDARD	标准 ASCII 码字体 (12 × 24)
FONT_COMPRESS	压缩 ASCII 码字体 (9 × 17)

【返回值】

POSPrinter 对象

2.34. setCharRightSpace

设置字符的右间距

POSPrinter setCharRightSpace(byte space)

【参数】

➤ space

右间距距离为[n×横向移动单位或纵向移动单位]英寸。

【返回值】

POSPrinter 对象

2.35. sendData

该方法用于发送数据到打印机。

```
POSPrinter sendData(byte[] data);
```

```
POSPrinter sendData(List<byte[]> datas);
```

【参数】

➤ data

需发送的字节数组

➤ datas

需发送的字节数组集合

【返回值】

POSPrinter 对象