

Android POS 编程手册

v3.5.8

（注：浏览时请使用 PDF 左侧导航栏）

1. 介绍

这个手册介绍怎么实现票据打印。常量定义在 POSConst 类中。

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。

注:使用对齐方式时，data 需以“\n”结尾，否则有可能会无效。

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，该方法不支持 76 针式打印机。

```
POSPrinter printBitmap(String bitmapPath, int alignment, int width)
POSPrinter printBitmap(String bitmapPath, int alignment, int width, int model)
POSPrinter printBitmap(String bitmapPath, int alignment, int width, int model, AlgorithmType
bmpType)
POSPrinter printBitmap(Bitmap bmp, int alignment, int width)
POSPrinter printBitmap(Bitmap bmp, int alignment, int width, int model)
POSPrinter printBitmap(Bitmap bmp, int alignment, int width, int model, AlgorithmType
bmpType)
压缩发送打印图片，支持部分机型
POSPrinter printBitmapCompress(Bitmap bmp, int alignment, int width, int model)
POSPrinter printBitmapCompress(Bitmap bmp, int alignment, int width)
```

【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

➤ bmpType

算法方式

变量	描述
Threshold , None	二值法(默认)
Dithering, Diffusion	扩散算法
Halftone	半色调算法

【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。不可与 listenPrinterMessage 同时使用

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);  
}
```

对应 type 返回的 data[0]状态参考以下表格。

type	位	功能	值	十进制值
STS_TYPE_PRINT	0	固定为 0	0	0
	1	固定为 1	1	2
	2	一个或两个钱箱打开	0	0
		两个钱箱都关闭	1	4
	3	联机	0	0
		脱机	1	8
	4	脱机	1	16
	5,6	固定为 0	0	0
STS_TYPE_OFFLINE	7	固定为 0	0	0
	0	固定为 0	0	0
	1	固定为 1	1	2
	2	上盖关	0	0
		上盖开	1	4
	3	未按走纸键	0	0
		按下走纸键	1	8
	4	固定为 1	1	16
	5	打印机不缺纸	0	0
		打印机缺纸	1	32
	6	无错误情况	0	0
		有错误情况	1	64

	7	固定为 0	0	0
STS_TYPE_ERR	0	固定为 0	0	0
	1	固定为 1	1	2
	2	未定义	0	0
	3	切刀无错误	0	0
		切刀有错误	1	8
	4	固定为 1	1	16
	5	无不可恢复错误	0	0
		有不可恢复错误	1	32
	6	打印头温度和电压正常	0	0
		打印头温度或电压超出范围	1	64
	7	固定为 0	0	0
STS_TYPE_PAPER	0	固定为 0	0	0
	1	固定为 1	1	2
	2,3	有纸	0	0
		纸将尽	1	12
	4	固定为 1	1	16
	5,6	有纸	0	0
		纸尽	1	96
	7	固定 0	0	0

2.14. printerStatusII

这个方法用于打印机常用状态查询，查询超时时间 3000ms,不可与 listenPrinterMessage 同时使用

void printerStatusII(IStatusCallback callback)

【参数】

➤ callback

读取的状态回调。

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

当 status 小于 0 时,CONNECT_ERROR: 连接已经断开 READ_TIMEOUT:等待超时。

当 status 等于 0 时,表示打印机状态正常。

当 status 大于 0 时,状态值如下表所示:

位	描述	对应的值
0	打印中	1
1	打印机开盖	2
2	打印机缺纸	4
3	纸将尽	8
4	钱箱打开	16

5	其他错误	32
6	切刀错误	64
7	打印头过热	128

2.15. getPrinterInfo

获取打印机的标识信息，仅支持部分机型。不可与 listenPrinterMessage 同时使用
void getPrinterInfo(IInfoCallback callback)

[Parameter]

➤ callback

Printer information callback.

```
public interface IInfoCallback {  
    void receive(PrinterInfo info);  
}
```

Field	Description
printerUniqueId	打印机唯一编号
modelName	打印机型号名
firmwareVersion	固件版本号
markerName	标记名
connectionType	打印机的连接类型, BLE\NET\USB
modelId	打印机型号 id
serialNumber	打印机序列号

[Return]

void

2.16. cashBoxCheck

这个方法用于钱箱状态查询。不可与 listenPrinterMessage 同时使用
void cashBoxCheck(IStatusCallback callback)

2.17. printerStatus

这个方法用于打印机常用状态查询。打印机状态会通过 listenPrinterMessage 回传回来

void printerStatus()

【返回值】

void

【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

2.18. listenPrinterMessage

监听打印机回传消息，包含有开始任务([RevTaskStartMo](#))，结束任务([RevTaskEndMo](#))，打印机状态变化([RevStatusMo](#))，和打印机固件版本信息([RevFwmVersionMo](#))

void listenPrinterMessage(IReceiveCallback callback)

【参数】

➤ callback

相关对象的回调方法

```
public interface IReceiveCallback {  
    void callback(IReceive receive);  
}
```

【返回值】

void

2.19. stopListenPrinterMessage

停止监听打印机回传消息

void stopListenPrinterMessage()

【返回值】

void

2.20. taskStart

开始任务

POSPrinter taskStart(long taskID)

【参数】

➤ taskID

任务 id

【返回值】

POSPrinter 对象

2.21. taskEnd

结束任务

POSPrinter taskEnd(long taskID)

【参数】

➤ taskID

任务 id

【返回值】

POSPrinter 对象

2.22. checkFirmwareVersion

这个函数的功能用于查询打印机固件版本,打印机固件版本会通过 listenPrinterMessage 回传回来

void checkFirmwareVersion()

【返回值】

void

2.23. 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

2.24. 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

2.25. 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

2.26. 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

2.27. 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

2.28. 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

2.29. 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

2.30. 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

2.31. downloadNVImage

保存 NV 图片到 flash 中

POSPrinter downloadNVImage(String imagePaths, int imageWidth)

POSPrinter downloadNVImage(String imagePaths, int imageWidth, AlgorithmType bmpType)

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

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

【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

打印宽度,可用于缩放图片

➤ bmpType

算法方式

变量	描述
Threshold , None	二值法(默认)
Dithering, Diffusion	扩散算法
Halftone	半色调算法

【返回值】

POSPrinter 对象

2.32. 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

2.33. initializePrinter

初始化打印机

POSPrinter initializePrinter()

【返回值】

POSPrinter 对象

2.34. selectBitmapModel

选择位图模式

POSPrinter selectBitmapModel(int model, int width, Bitmap bmp)

【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(76 针式打印机不支持)
DOUBLE_DENSITY_24	24 点双密度(76 针式打印机不支持)

➤ width

打印的图片宽度

➤ bmp

需要打印的位图

【返回值】

POSPrinter 对象

2.35. 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

2.36. 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

2.37. setChineseModel

设置中文模式

void setChineseModel(boolean on)

【参数】

➤ on

true 表示设置中文模式 false 表示取消中文模式

注:

1. 在设置中日韩的编码之前，需要使用该接口来切换为中文模式，其他编码则需要取消中文模式。可以通过打印自检页来查看打印机是否处于中文模式。
2. 使用该接口后，打印机会进行重启。

2.38. selectCodePage

选择字符代码表

POSPrinter selectCodePage(int page)

【参数】

➤ page

代码表

值	描述	值	描述
0	PC437(Std.Europe)	56	PC861(Icelandic)
1	Katakana	57	PC863(Canadian)
2	PC850(Multilingual)	58	PC865(Nordic)
3	PC860(Portugal)	59	PC866(Russian)
4	PC863(Canadian)	60	PC855(Bulgarian)
5	PC865(Nordic)	61	PC857(Turkey)
6	West Europe	62	PC862(Hebrew)
7	Greek	63	PC864(Arabic)
8	Hebrew	64	PC737(Greek)
9	East Europe	65	PC851(Greek)
10	Iran	66	PC869(Greek)
16	WPC1252	67	PC928(Greek)
17	PC866(Cyrillic#2)	68	PC772(Lithuanian)

18	PC852(Latin2)	69	PC774(Lithuanian)
19	PC858	70	PC874(Thai)
20	IranII	71	WPC1252(Latin-1)
21	Latvian	72	WPC1250(Latin-2)
22	Arabic	73	WPC1251(Cyrillic)
23	PT1511251	74	PC3840(IBM-Russian)
24	PC747	75	PC3841(Gost)
25	WPC1257	76	PC3843(Polish)
27	Vietnam	77	PC3844(CS2)
28	PC864	78	PC3845(Hungarian)
29	PC1001	79	PC3846(Turkish)
30	Uigur	80	PC3847(Brazil-ABNT)
31	Hebrew	81	PC3848(Brazil)
32	WPC1255(Israel)	82	PC1001(Arabic)
255	Thai	83	PC2001(Lithuan)
33	WPC1256	84	PC3001(Estonian-1)
50	PC437(Std.Europe)	85	PC3002(Eston-2)
51	Katakana	86	PC3011(Latvian-1)
52	PC437(Std.Europe)	87	PC3012(Tatv-2)
53	PC858(Multilingual)	88	PC3021(Bulgarian)
54	PC852(Latin-2)	89	PC3041(Maltese)
55	PC860(Portuguese)	255	[Thai]
256	GB2312(Simplified Chinese)	257	BIG5(Traditional Chinese)
258	Korean	259	Japanese

【返回值】

POSPrinter 对象

注：

1. 在设置完成打印机编码后还需使用 [setCharSet](#) 接口来设置相应的上位机编码，保证下发的数据能够被打印机正确解析。
2. 该接口只能够临时修改打印机代码页，断电失效。

2.39. selectDefaultCodePage

设置默认代码页(断电仍有效)

POSPrinter selectDefaultCodePage(int page)

【参数】

➤ Page

同 [selectCodePage](#)

【返回值】

POSPrinter 对象

2.40. 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

2.41. 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

2.42. printPDF417

打印 PDF417 码

POSPrinter printPDF417(String pdfData)

POSPrinter printPDF417(String pdfData, int cellWidth, int cellHeightRatio, int numberOfColumns, int numberOfRows, int eclType, int eclValue, int alignment)

➤ pdfData

打印的数据

➤ cellWidth

单元宽度, 范围[2- 8]

- **cellHeightRatio**
设置行高。范围[2-8]。
 $\text{cell height} = \text{cellHeightRatio} \times \text{cellWidth}$
- **numberOfColumns**
数据区域的列数。范围[0-30]
- **numberOfRows**
设置行数，范围[0, 3-90]
- **eclType**
错误纠正等级类型
范围[0-1]。
0:误差校正级别由级别来设定
1:误差校正级别由比率设定。比率为 $\text{eclValue} \times 10\%$
- **eclValue**
错误纠正等级值
 $\text{eclType} = 0$:范围[0 – 8].
 $\text{eclType} = 1$:范围[1 – 40].
- **alignment**
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

2.43. sendData

该方法用于发送数据到打印机。

`POSPrinter sendData(byte[] data);`

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

`void sendData(byte[] datas, IStatusCallback callback);`

【参数】

- **data**
需发送的字节数组
- **datas**
需发送的字节数组集合
- **callback**
发送状态回调，1 表示发送成功，0 表示发送失败

【返回值】

POSPrinter 对象

2.44. setTransaction

设置事务状态。开启事务时，会把发送的数据存起来，直到调用 `transactionPrint` 函数，才会把所有缓存的数据发给打印机；关闭事务时，如果有事务缓存数据，会把缓存的数据发送给打印机。仅支持异步通信。

```
void setTransaction(boolean isOpen);
```

【参数】

➤ `isOpen`

是否开启事务

【返回值】

void

2.45. transactionPrint

发送事务缓存中的数据。不可与 `listenPrinterMessage` 同时使用

```
void transactionPrint(IStatusCallback callback);
```

【参数】

➤ `callback`

发送状态回调，1 表示发送成功，0 表示发送失败

【返回值】

void

2.46. clearTransaction

清除事务缓存中的数据。

```
void clearTransaction();
```

【返回值】

void

2.47. printTable

打印表格

POSPrinter printTable(PTable table)

【参数】

➤ table

需要打印的表对象，见 [PTable](#)

【返回值】

POSPrinter 对象

2.48. wifiConfig

设置打印机 Wi-Fi

void wifiConfig(byte[] ip, byte[] mask, byte[] gateway, String ssid, String password, byte encrypt)

【参数】

➤ ip

ip 地址,长度为 4 的字节数组。如:new byte[]{(byte)192,(byte)168,1,100}

➤ mask

子网掩码,长度为 4 的字节数组。

➤ gateway

默认网关,长度为 4 的字节数组

➤ ssid

Wi-Fi 名,不能为空。

➤ password

Wi-Fi 密码

➤ encrypt

加密类型

变量	描述
ENCRYPT_NULL	NULL
ENCRYPT_WEP64	WEP64
ENCRYPT_WEP128	WEP128
ENCRYPT_WPA_AES_PSK	WPA_AES_PSK
ENCRYPT_WPA_TKIP_PSK	WPA_TKIP_PSK
ENCRYPT_WPA_TKIP_AES_PSK	WPA_TKIP_AES_PSK
ENCRYPT_WPA2_AES_PSK	WPA2_AES_PSK
ENCRYPT_WPA2_TKIP	WPA2_TKIP
ENCRYPT_WPA2_TKIP_AES_PSK	WPA2_TKIP_AES_PSK
ENCRYPT_WPA_WPA2_MixedMode	WPA_WPA2_MixedMode

【返回值】

void

2.49. setIp

设置网络 ip 地址

void setIp(byte[] ip)

【参数】

➤ ip

ip 地址,长度为 4 的字节数组。如:new byte[]{(byte)192,(byte)168,1,100}

【返回值】

void

2.50. setMask

设置子网掩码

void setMask(byte[] mask)

【参数】

➤ mask

子网掩码,长度为 4 的字节数组。

【返回值】

void

2.51. setGateway

设置默认网关

void setGateway(byte[] gateway)

【参数】

➤ gateway

默认网关,长度为 4 的字节数组。

【返回值】

void

2.52. setNetAll

设置网络信息

```
void setNetAll(byte[] ip, byte[] mask, byte[] gateway, boolean dhcpIsOpen)
```

【参数】

➤ ip

ip 地址，长度为 4 的字节数组。如:new byte[] {(byte)192,(byte)168,1,100}

➤ mask

子网掩码，长度为 4 的字节数组。

➤ gateway

默认网关，长度为 4 的字节数组。

➤ dhcpIsOpen

是否打开 DHCP。 true 打开 false 关闭。

【返回值】

void

2.53. setBluetooth

设置蓝牙信息

```
void setBluetooth(String name, String pin)
```

【参数】

➤ name

蓝牙名称

➤ pin

蓝牙的 pin 码

【返回值】

Void

2.54. setDensity

设置浓度

```
void setDensity(byte level)
```

【参数】

➤ level

浓度等级， 范围【1， 8】

【返回值】

void

2.55. getSerialNumber

获取打印机序列号,不可与 listenPrinterMessage 同时使用

void getSerialNumber(IDataCallback callback)

【参数】

➤ callback

采用回调的方式获取查询到的 SN 码的字节数组。

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

【返回值】

void

2.56. awaitSendCompletionAndDisconnect

等待数据发送完成再断开接口。不可与 listenPrinterMessage 同时使用。

void awaitSendCompletionAndDisconnect()

【返回值】

void

3. PTable

表格打印类，表的列宽是以单字节字符为 1 个单位来统计的,例如字母‘a’的宽度为 1，中文的‘印’的宽度为 2。

如果需要换行可再文本中加入‘\n’。 如果文本宽度大于设定的列宽，会自动换行。

3.1. PTable

构造函数

PTable(String[] titles, Integer[] numberOfSingleBytesPerCol)

PTable(String[] titles, Integer[] numberOfSingleBytesPerCol, Integer[] align)

【参数】

➤ titles

标题数组集合

➤ numberOfSingleBytesPerCol

各列中单字节字符数量集合

➤ align

各列的对齐方式，0 左对齐 1 右对齐。 默认为 0。

【返回值】

PTable 对象

3.2. addRow

添加行

PTable addRow(String... row)

PTable addRow(String title, String[] row)

PTable addRow(String title, String[] row, String remark)

PTable addRow(String title, String[] row, String remark, TableBarcode barcode)

【参数】

➤ title

每一行标题，为空则无标题，默认为空。

➤ row

所在行的字符集合

➤ remark

每行的后面的备注，为空则无备注，默认为空。

Item	QTY	Price	Total
Apple Apple	1	7.99	7.99
100328	1	7.99	7.99
remarks:xxxxxxx	4	0.99	3.96
680015	4	0.99	3.96
102501102501	1	43.99	43.99
102501	1	4.99	4.99
021048	1	4.99	4.99

【返回值】

PTable 对象

4. TableBarcode

表格行属性中的条码辅助类

4.1. TableBarcode

构造函数

TableBarcode(String data, int codeType)

TableBarcode(String data, int codeType, int height)

TableBarcode(String data, int codeType, int width, int height, int alignment)

TableBarcode(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 80

➤ width

条码横向模块宽度，范围【2，6】，默认 2

➤ alignment

条码的对齐方式，默认 ALIGNMENT_LEFT

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】

TableBarcode 对象

5. PosUdpNet

POS 打印机 Udp 消息收发类，可通过该类实现搜索局域网内的打印机设备和修改网络信息。

5.1. searchNetDevice

搜寻局域网内的打印设备

`void searchNetDevice(UdpCallback callback)`

循环搜索与手机在同一网段的打印机

`void searchNetPrinterLoop(int interval, UdpCallback callback)`

【参数】

➤ interval

搜索间隔时间，单位为 ms。

➤ callback

通过回调的方式返回搜寻到的设备信息。

```
public interface UdpCallback {
    void receive(UdpDevice device);
}
```

【返回值】

void

5.2. stopSearchLoop

停止循环搜索网络打印机

`void stopSearchLoop()`

【返回值】

void

5.3. udpNetConfig

通过广播的形式，修改对应设备的网口信息。
static void udpNetConfig(byte[] macAddress, byte[] ipAddress, byte[] mask, byte[] gateway, boolean dhcp)

【参数】

- macAddress
设备的 MAC 地址
- ipAddress
ip 地址
- mask
子网掩码
- gateway
默认网关
- dhcp
是否开启 dhcp

【返回值】

void

6. PosFirmwareUpdate

PosFirmwareUpdate 用于票据打印机固件升级
注:请不要再打印机更新重启完成前断电。

6.1. PosFirmwareUpdate

PosFirmwareUpdate(IDeviceConnection connect, String filePath, IStatusCallback progress)

【参数】

- connect
打印机连接对象
- filePath
固件文件路径
- progress
更新进度回调，具体解释如下

Progress	描述
UPDATE_FAIL	更新失败
UPDATE_SUCCESS	更新成功

UPDATE_FMW_ERROR	固件与打印机不匹配
>0	更新固件真实进度

【返回值】

void

6.2. getGd427PrinterVersion

获取打印机固件版本号, 注:仅支持 GD427 主控芯片

void getGd427PrinterVersion(IDeviceConnection connect, IStrCallback callback)

【参数】

➤ callback

回调固件版本号。如果发生错误，则返回空字符串

【返回值】

void

6.3. startUpdate

开始更新固件

void startUpdate()

【返回值】

Void

7. UdpDevice

搜索到的网络打印机信息对象

7.1. isIpV6

设备是否是 ipv6 地址打印机。否则为 ipv4 地址打印机

7.2. getMacAddress

打印机的 mac 地址

byte[] getMacAddress()

String getMacStr()

【返回值】

Mac 地址

7.3. getIpv6UserStr

ipv6 的用户地址。请在 isIPv6 为 true 情况下使用。

String getIpv6UserStr() throws UnknownHostException

【返回值】

ipv6 的用户地址

7.4. getIpv6LocalStr

ipv6 的本地地址。请在 isIPv6 为 true 情况下使用。

String getIpv6LocalStr() throws UnknownHostException

【返回值】

ipv6 的本地地址。

7.5. getIpV4Str

ipv4 地址。请在 isIPv6 为 false 情况下使用。

String getIpV4Str()

【返回值】

ipv4 地址。

7.6. getMaskStr

ipv4 设备的子网掩码。请在 isIPv6 为 false 情况下使用。

String getMaskStr()

【返回值】

ipv4 设备的子网掩码。

7.7. getGatewayStr

ipv4 设备的默认网关。请在 isIpV6 为 false 情况下使用。

String getGatewayStr()

【返回值】

ipv4 设备的默认网关。

7.8. isDhcp

是否为 dhcp

boolean isDhcp()

【返回值】

dhcp

7.9. getIpStr

ip 地址。如果是 ipv6 则返回 Ipv6Local 地址，否则返回 ipv4 地址

String getIpStr() throws UnknownHostException

【返回值】

Ip 地址

7.10. getModelName

获取打印机型号，仅支持部分机型。不支持的设备返回为空字符串。

String getModelName()

【返回值】

打印机型号

8. RevFwmVersionMo

打印机固件版本号对象，调用 `checkFirmwareVersion` 函数时，通过 `listenPrinterMessage` 返回。

【属性】

- `String connectInfo`
连接信息
- `String version`
当前的打印机固件版本

9. RevStatusMo

打印机状态对象，当打印机状态发生变化，或者调用 `printerStatus` 函数时，通过 `listenPrinterMessage` 返回。

【属性】

- `String connectInfo`
连接信息
- `boolean isNormal`
打印机状态是否正常
- `boolean isCashBoxOpen`
打印机钱箱是否开启
- `boolean isCoverOpen`
打印头是否打开
- `boolean isCutterError`
是否有切刀错误
- `boolean isPaperNearEmpty`
是否纸即将用尽，部分机型适用
- `boolean isOutOfPaper`
是否缺纸
- `boolean isOtherError`
是否有其它错误
- `boolean isHeadOverheat`
是否打印头过热

10. RevTaskEndMo

当打印机结束当前任务时通过 `listenPrinterMessage` 返回，配合 `taskEnd` 函数使用

【属性】

- `long taskId`

当前任务 id

➤ String connectInfo

连接信息

11. RevTaskStartMo

当打印机开始当前任务时通过 listenPrinterMessage 返回，配合 taskStart 函数使用

【属性】

➤ long taskId

当前任务 id

➤ String connectInfo

连接信息

12. SendFailMo

当发送消息出现异常时通过 listenPrinterMessage 返回

【属性】

➤ String errMsg

异常消息

➤ String connectInfo

连接信息

13. ConnectInterruptMo

当连接中断时通过 listenPrinterMessage 返回

【属性】

➤ String errMsg

异常消息

➤ String connectInfo

连接信息