

打印机接口说明

一 概述.....	2
二 使用.....	2
三 各个类的区别.....	3
① BLEPrinting、NETPrinting、IO 这三者的区别.....	3
② POSPrinting、LabelPrinting 这两者的区别.....	3
四 函数说明.....	4
POSPrinting.....	4
POS_PrintText.....	4
POS_PrintBarcode.....	6
POS_PrintQRCode.....	8
POS_PrintPicture.....	9
POS_FeedLine.....	10
POS_Reset.....	11
POS_SetRightSpacing.....	12
POS_SetLineHeight.....	13
POS_SetPrintSpeed.....	14
POS_CutPaper.....	15
POS_Beep.....	16
POS_KickDrawer.....	17
POS_QueryStatus.....	18
POS_RTQueryStatus.....	19
POS_TicketSucceed.....	21
LabelPrinting.....	22
PageBegin.....	22
PageEnd.....	23
PagePrint.....	24
DrawPlainText.....	25
DrawLine.....	26
DrawBox.....	27
DrawRectangel.....	28
DrawBarcode.....	29
DrawQRCode.....	31
DrawPDF417.....	32
DrawBitmap.....	33
DrawBitmap.....	34

一 概述

1 NETPrinting、BLEPrinting 处理底层读写。

他们分别有自己的打开，关闭函数，有自己的回调接口。

继承自 IO、重写了 IO 的 3 个重要函数。

- (bool) IsOpened;
- (int) Write:(Byte *) buffer offset:(int) offset count:(int) count;
- (int) Read:(Byte *)buffer offset:(int)offset count:(int)count timeout:(int)timeout;

2 POSPrinting 和 LabelPrinting 封装了打印指令并持有一个 IO 接口。

POSPrinting 封装了 ESC/POS 指令

LabelPrinting 封装了标签打印指令

他们按照指令集的格式将数据组织好，然后调用 IO 的 Write 函数进行写入数据，调用 Read 函数读取数据。

3 MemoryIO

该类可以将指令写到内存中（而不是发送给打印机），后续通过 GetBuffer 来获取打印数据。可用于调试，也可以用于其他用途（例如，将获取到的指令，通过别的接口发送出去）

二 使用

使用时，先实例化一个 IO 子类。以蓝牙打印为例：

① 先实例化一个 POSPrinting（为方便描述，设变量名为 pos），这个时候，直接调用 POS_XXX 系列函数，会发现并不能打印，因为这时候 POSPrinting 并没有持有一个可读写的 IO。需要进行②③步。

② 先实例化一个 BLEPrinting（为方便描述，设变量名为 ble），然后调用 Open 函数连接到蓝牙打印机。

③ 连接成功之后，调用 pos 的 SetIO，将让 pos 持有 ble，这样，后续的 POS_XXX 系列函数，就会通过 ble 的 Write 和 Read 与打印机通讯。

三 各个类的区别

① BLEPrinting、NETPrinting、IO 这三者的区别

- A) IO 是父类，提供统一的接口，供 POSPrinting 和 LabelPrinting 使用。
- B) BLEPrinting 是用于蓝牙通讯的，除了基本的 Open Close IsOpened Read Write 之外，还有
 - scan 扫描蓝牙打印机，扫描成功会调用回调接口 BLEPrintingDiscoverDelegate
 - stopScan 停止扫描
 - BLEPrintingOpenDelegate Open 成功之后会回调该接口
 - BLEPrintingDiscoverDelegate 扫描到打印机之后会回调该接口
 - BLEPrintingReceiveDelegate 收到数据之后会回调该接口
 - BLEPrintingDisconnectDelegate 蓝牙断开之后会回调该接口（多次调用 Close 不会调用）
- C) NETPrinting 是用于网络通讯的，除了基本的 Open Close IsOpened Read Write 之外，还有
 - NETPrintingOpenDelegate Open 成功之后会回调该接口
 - NETPrintingDisconnectDelegate 连接断开之后会回调该接口（多次调用 Close 不会调用）

② POSPrinting、LabelPrinting 这两者的区别

- A) POSPrinting 封装了便携指令集
- B) LabelPrinting 封装了标签指令集。

普通热敏打印机（只支持 ESC/POS 指令集），只能使用 POSPrinting 控制打印机打印。
标签打印机（支持 ESC/POS 指令集、标签指令集），可以使用 POSPrinting 控制打印机打印，也可以用 LabelPrinting 控制打印机打印。

两套指令不能穿插使用，意思是：

使用 LabelPrinting 控制打印时，PageBegin 到 PagePrint 之间，不能穿插 POSPrinting 函数。

四 函数说明

POSPrinting

普通行式打印

POS_PrintText

描述：打印文本

Syntax

- (bool) POS_PrintText:(char *)pszString x:(int)x nWidthTimes:(int)nWidthTimes
nHeightTimes:(int)nHeightTimes nFontType:(int)nFontType nFontStyle:(int)nFontStyle

Parameters

pszString

要打印的内容。UTF8 编码字符串。

x

指定水平方向的起始点位置离打印区域左边界的点数。（横坐标）
支持左对齐，居中，右对齐

传入 x 见下表

x	含义
-1	左对齐
-2	居中对齐
-3	右对齐
大于等于 0	横坐标

nWidthScale

指定宽度放大倍数 [0,7]

nHeightScale

指定高度放大倍数 [0,7]

nFontType

字体类型

0 标准字体

1 压缩字体

nFontStyle

指定字体风格，可以为下表中的一个或者若干个（相加即可）

Value	Meaning
0x00	正常
0x08	加粗
0x80	1 点粗的下划线
0x100	2 点粗的下划线
0x200	倒置（只在行首有效）
0x400	反显（黑底白字）
0x1000	每个字符顺时针旋转 90 度

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

POS_PrintText 并不立刻打印，需要调用 POS_FeedXXX 系列函数，才会把行缓冲区中的内容打印出来。

同一行不支持多种对齐方式。

POS_PrintBarcode

打印条码

Syntax

- (bool) POS_PrintBarcode:(char *)pszString x:(int)x nType:(int)nType nUnitWidth:(int)nUnitWidth
nHeight:(int)nHeight nHriFontType:(int)nHriFontType nHriFontPosition:(int)nHriFontPosition

Parameters

pszString

条码内容

x

指定水平方向的起始点位置离打印区域左边界的点数。（横坐标）

支持左对齐，居中，右对齐

传入 x 见下表

x	含义
-1	左对齐
-2	居中对齐
-3	右对齐
大于等于 0	横坐标

nType

可以为以下列表中所列值之一。

Value	Meaning
0x41	UPC-A
0x42	UPC-C
0x43	JAN13(EAN13)
0x44	JAN8(EAN8)
0x45	CODE39
0x46	ITF
0x47	CODEBAR
0x48	CODE93
0x49	CODE 128

nUnitWidth

指定条码的基本元素宽度。

可以为以下列表中所列值（n）之一。

n	单基本模块 宽度 (连续型)	双基本模块宽度（离散型）	
		窄元素宽度	宽元素宽度
2	0. 25mm	0. 25mm	0. 625mm
3	0. 375mm	0. 375mm	1. 0mm
4	0. 5mm	0. 5mm	1. 25mm
5	0. 625mm	0. 625mm	1. 625mm
6	0. 75mm	0. 75mm	1.875mm

nHeight

条码高度

8 点即 1mm，填入 80 即可打印高度为 1CM 的条码。

nHriFontType

指定 HRI（Human Readable Interpretation）字符的字体类型。

可以为以下列表中所列值之一。

Value	Meaning
0x00	标准 ASCII
0x01	压缩 ASCII

nHriFontPosition

指定 HRI（Human Readable Interpretation）字符的位置。

可以为以下列表中所列值之一。

Value	Meaning
0x00	不打印
0x01	只在条码上方打印
0x02	只在条码下方打印
0x03	条码上、下方都打印

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

部分机型不支持指定起始位置，请使用左对齐，居中对齐，右对齐进行排版布局。

POS_PrintQRCode

打印二维码（QR 码）

Syntax

- (bool) POS_PrintQRcode:(char *)pszString x:(int)x nUnitWidth:(int)nUnitWidth
nVersion:(int)nVersion nECCLevel:(int)nECCLevel

Parameters

pszString

二维码文本

x

指定水平方向的起始点位置离打印区域左边界的点数。（横坐标）

支持左对齐，居中，右对齐

传入 x 见下表

x	含义
-1	左对齐
-2	居中对齐
-3	右对齐
大于等于 0	横坐标

nUnitWidth

QR 码单元宽度，范围[1,16]。

QR 码单元宽度越大，QR 码越大。

nVersion

QR 码版本。0 表示自动计算版本。

QR 码版本越大，能编码的字符就越多，QR 码也越大。

nECCLevel

QR 码纠错等级。[1,4]

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

部分机型不支持指定起始位置，请使用左对齐，居中对齐，右对齐进行排版布局。

POS_PrintPicture

打印图片

Syntax

- (bool) POS_PrintPicture:(UIImage *)mImage x:(int)x nWidth:(int)nWidth nHeight:(int)nHeight
nBinaryAlgorithm:(int)nBinaryAlgorithm nCompressMethod:(int)nCompressMethod

Parameters

mImage

要打印的图片

x

指定水平方向的起始点位置离打印区域左边界的点数。（横坐标）
支持左对齐，居中，右对齐

传入 x 见下表

x	含义
-1	左对齐
-2	居中对齐
-3	右对齐
大于等于 0	横坐标

nWidth

要打印的宽度

nHeight

要打印的高度

nBinaryAlgorithm

二值化算法

- 0 使用抖动算法，对彩色图片有较好的效果。
- 1 使用平均阈值算法，对文本类图片有较好的效果

nCompressMethod

压缩算法

- 0 不使用压缩算法
- 1 使用压缩算法

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

部分机型不支持指定起始位置，请使用左对齐，居中对齐，右对齐进行排版布局。

POS_FeedLine

Syntax

- (bool) POS_FeedLine

Parameters

Return value

返回 **true** 表示写入成功；返回 **false** 表示写入失败。

Remarks

打印机进纸一行

POS_Reset

复位打印机。会清空设置。

Syntax

- (bool) POS_Reset

Parameters

Return value

返回 **true** 表示写入成功；返回 **false** 表示写入失败。

Remarks

POS_SetRightSpacing

设置字符右边空白

Syntax

- (bool) POS_SetRightSpacing:(int)nDistance

Parameters

nDistance

字符右边空白

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

POS_SetLineHeight

设置行高

Syntax

- (bool) POS_SetLineHeight:(int)nHeight

Parameters

nHeight
行高

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

POS_SetPrintSpeed

设置打印速度 注：如果打印速度大于发送速度，打印会有卡顿感。

Syntax

- (bool) POS_SetPrintSpeed:(int)nSpeed

Parameters

nSpeed

打印速度（mm/s）

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

将打印速度设置为数据发送速度，可以是打印效果达到最好。

可以通过打印一张单据，测量单据的长度和所用时间，用长度/时间，即可。

POS_CutPaper

切纸

Syntax

- (bool) POS_CutPaper

Parameters

Return value

返回 **true** 表示写入成功；返回 **false** 表示写入失败。

Remarks

只对带切刀的机器有效

POS_Beep

蜂鸣器鸣叫

Syntax

- (bool) POS_Beep:(int)nBeepCount nBeepMillis:(int)nBeepMillis

Parameters

nBeepCount

鸣叫次数

nBeepMillis

每次鸣叫的时间 = 100 * nBeepMillis ms

Return value

返回 **true** 表示写入成功；返回 **false** 表示写入失败。

Remarks

POS_KickDrawer

打开钱箱

Syntax

- (bool) POS_KickDrawer:(int)nDrawerIndex nPulseTime:(int)nPulseTime

Parameters

nDrawerIndex

0 表示：脉冲发送到钱箱输出引脚 2

1 表示：脉冲发送到钱箱输出引脚 5

nPulseTime

脉冲时间

高电平时间： $nPulseTime * 2ms$

低电平时间： $nPulseTime * 2ms$

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

POS_QueryStatus

查询状态

打印机忙时，该命令会一直阻塞

返回的状态保存在 `status` 中

Syntax

```
- (bool) POS_QueryStatus:(int)type status:(Byte *)status timeout:(int)timeout  
MaxRetry:(int)MaxRetry
```

Parameters

`type`

当前固定为 1

`status`

`status` 该值目前无意义

`timeout`

单次查询状态的超时毫秒时间

`MaxRetry`

失败重试次数

Return value

返回 `true`，表明打印机状态 OK。否则，打印机未联机或打印机正忙。

Remarks

POS_RTQueryStatus

实时状态查询
无论打印机处于何种状态，只要打印机收到该命令就立刻回送状态
返回的状态保存在 status 中

Syntax
- (bool) POS_RTQueryStatus:(int)type status:(Byte *)status timeout:(int)timeout
MaxRetry:(int)MaxRetry

Parameters
type
type 可取值 [1,4]

1: 打印机状态

位	0/ 1	十六进制码	十进制码	功能
0	0	00	0	固定为 0
1	1	02	2	固定为 1
2	0	00	0	一个或两个钱箱打开 (没有钱箱的机器该位固定为零)
	1	04	4	两个钱箱都关闭
3	0	00	0	联机
	1	08	8	脱机
4	1	10	16	固定为 1
5, 6		--	--	未定义
7	0	00	00	纸已撕走
	1	80	96	纸未撕走

2: 传送脱机状态

位	0/ 1	十六进制码	十进制码	功能
0	0	00	0	固定为 0
1	1	02	2	固定为 1
2	0	00	0	上盖关
	1	04	4	上盖开
3	0	00	0	未按走纸键
	1	08	8	按下走纸键
4	1	10	16	固定为 1
5	0	00	0	打印机不缺纸
	1	20	32	打印机缺纸

6	0	00	00	没有出错情况
	1	40	64	有错误情况
7	0	00	0	固定为 0

3: 传送错误状态

位	0/ 1	十六进制码	十进制码	功能
0	0	00	0	固定为 0
1	1	02	2	固定为 1
2		--	--	未定义
3	0	00	0	切刀无错误
	1	08	8	切刀有错误
4	1	10	16	固定为 1
5	0	00	0	无不可恢复错误
	1	20	32	有不可恢复错误
6	0	00	00	打印头温度和电压正常
	1	40	64	打印头温度或电压超出范围
7	0	00	0	固定为 0

4: 传送纸传感器状态

位	0/ 1	十六进制码	十进制码	功能
0	0	00	0	固定为 0
1	1	02	2	固定为 1
2, 3	0	00	0	有纸
	1	0C	12	纸将近
4	1	10	16	固定为 1
5, 6	0	00	0	有纸
	1	60	96	纸尽
7	0	00	0	固定为 0

status

status 对应含义见上表

timeout

单次查询状态的超时毫秒时间

MaxRetry

失败重试次数

Return value

返回 true, 表明打印机通讯正常, 查询的状态保存在 status 中。

Remarks

POS_TicketSucceed

描述：询单据打印结果。

可以识别出因缺纸，过热，或其他错误导致单据打印不完整的情况。

Syntax

- (int) POS_TicketSucceed:(int)dwSendIndex timeout:(int)timeout;

Parameters

`dwSendIndex`

单据索引，填 0 或递增均可。

`timeout`

超时毫秒时间，根据单据情况设置即可。例如打印一张单据正常需要 3s 时间，那么此处可以最多等待 10s，来确定单据是否打印完成。参数可以填 10000

Return value

- * 返回 0，表示单据打印成功。
- * 返回-1，表示单据查询指令失败。失败原因：连接断开或已关闭。
- * 返回-2，表示单据查询指令失败。失败原因：发送失败。
- * 返回-3，表示单据查询指令失败。失败原因：打印机无响应。
- * 返回-4，表示单据打印失败。原因：打印机脱机。
- * 返回-5，表示单据打印不完整。原因：打印机因缺纸而中断打印。
- * 返回-6，表示单据打印失败。原因：其他原因。

Remarks

部分机型不支持单据查询指令

LabelPrinting

标签打印

PageBegin

描述：指示一个 Page 页面的开始，并设置 Page 页的大小，参考点坐标和页面旋转角度。

Syntax

```
- (bool) PageBegin:(int)startx starty:(int)starty width:(int)width  
height:(int)height rotate:(int)rotate
```

Parameters

`startx`

页面起始点 x 坐标

`starty`

页面起始点 y 坐标

`width`

页面页宽

`startx + width` 的范围为 $[1, 384]$ 。编写 SDK 的时候，该打印机一行的打印点数为 384 点。如果你不确定每行打印点数，请参考打印机规格书。一般来说有 384, 576, 832 这三种规格。

`height`

页面页高

`starty + height` 的范围 $[1, 936]$ 。编写 SDK 的时候，限制是 936，但是这个值并不确定，这和打印机的资源有关。即便如此，也不建议把页高设置过大。建议页宽和页高设置和标签纸匹配即可。

`rotate`

页面旋转。`rotate` 的取值范围为 $\{0, 1\}$ 。为 0，页面不旋转打印，为 1，页面旋转 90 度打印。

Return value

返回 `true` 表示写入成功；返回 `false` 表示写入失败。

Remarks

PageEnd

描述：指示一个 Page 页面的结束。

Syntax

- (bool) PageEnd

Parameters

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

PagePrint

描述：将 Page 页上的内容打印到标签纸上。

Syntax

- (bool) PagePrint:(int)num

Parameters

num

打印的次数，1-255。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawPlainText

描述：在 Page 页面上指定位置绘制文本。只能单行打印。

Syntax

- (bool) DrawPlainText:(int)startx starty:(int)starty font:(int)font style:(int)style str:(char *)str

Parameters

startx

定义文本起始位置 x 坐标，取值范围：[0, Page_Width-1]

starty

定义文本起始位置 y 坐标，取值范围：[0, Page_Height-1]

font

选择字体，有效值范围为{16, 24, 32, 48, 64, 80, 96}，当前打印机只可以使用 24。

style

字符风格。

数据位

定义

- | | |
|--------------|-----------------------|
| 0 加粗标志位： | 置 1 字体加粗，清零则字体不加粗。 |
| 1 下划线标志位： | 置 1 文本带下划线，清零则无下划线。 |
| 2 反白标志位： | 置 1 文本反白(黑底白字)，清零不反白。 |
| 3 删除线标志位： | 置 1 文本带删除线，清零则无删除线。 |
| [5,4] 旋转标志位： | 00 旋转 0° ； |
| | 01 旋转 90° ； |
| | 10 旋转 180° ； |
| | 11 旋转 270° ； |

[11,8] 字体宽度放大倍数；

[15,12] 字体高度放大倍数；

str

字符串数据流

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawLine

描述：在 Page 页指定两点间绘制一条直线段。

Syntax

- (bool) DrawLine:(int)startx starty:(int)starty endx:(int)endx endy:(int)endy width:(int)width
color:(int)color

Parameters

startx

直线段起始点 x 坐标值，取值范围：[0, Page_Width-1]。

starty

直线段起始点 y 坐标值，取值范围：[0, Page_Height-1]。

endx

直线段终止点 x 坐标值，取值范围：[0, Page_Width-1]。

endy

直线段终止点 y 坐标值，取值范围：[0, Page_Height-1]。

width

直线段线宽，取值范围：[1, Page_Height-1]。

color

直线段颜色，取值范围：{0, 1}。

当 Color 为 1 时，线段为黑色。

当 Color 为 0 时，线段为白色。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawBox

描述：在 Page 页指定位置绘制指定大小的矩形框。

Syntax

- (bool) DrawBox:(int)left top:(int)top right:(int)right bottom:(int)bottom
borderwidth:(int)borderwidth bordercolor:(int)bordercolor

Parameters

left

矩形框左上角 x 坐标值，取值范围：[0, Page_Width-1]。

top

矩形框左上角 y 坐标值。取值范围：[0, Page_Height-1]。

right

矩形框右下角 x 坐标值。取值范围：[0, Page_Width-1]。

bottom

矩形框右下角 y 坐标值。取值范围：[0, Page_Height-1]。

borderwidth

矩形框线宽。

bordercolor

矩形框线颜色，取值范围{0, 1}。当 Color = 1 时，绘制黑色矩形宽，Color = 0 时，绘制白色矩形框。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawRectangel

描述：在 Page 页指定位置绘制矩形块。

Syntax

- (bool) DrawRectangel:(int)left top:(int)top right:(int)right bottom:(int)bottom color:(int)color

Parameters

left

矩形块左上角 x 坐标值，取值范围：[0, Page_Width-1]。

top

矩形块左上角 y 坐标值。取值范围：[0, Page_Height-1]。

right

矩形块右下角 x 坐标值。取值范围：[0, Page_Width-1]。

bottom

矩形块右下角 y 坐标值。取值范围：[0, Page_Height-1]。

color

矩形块颜色，取值范围：{0, 1}。当 Color 为 1 时，矩形块为黑色。当 Color 为 0 时，矩形块为白色。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawBarcode

描述：在 Page 页指定位置绘制一维条码。

Syntax

- (bool) DrawBarcode:(int)startx starty:(int)starty type:(int)type height:(int)height
unitwidth:(int)unitwidth rotate:(int)rotate str:(char *)str

Parameters

startx

条码左上角 x 坐标值，取值范围：[0, Page_Width-1]。

starty

条码左上角 y 坐标值，取值范围：[0, Page_Height-1]。

type

标识条码类型，取值范围：[0, 29]。各值定义如下：

type	类型	长度	条码值范围（十进制）
0	UPC-A	11	48-57
1	UPC-E	6	48-57
2	EAN13	12	48-57
3	EAN8	7	48-57
4	CODE39	1-	48-57, 65-90, 32, 36, 37, 43, 45, 46, 47
5	I25	1-	偶数 48-57
6	CODABAR	1-	48-57, 65-68, 36, 43, 45, 46, 47, 58
7	CODE93	1-255	0-127
8	CODE128	2-255	0-127
9	CODE11		
10	MSI		
11	"128M",	//可以根据数据切换编码模式-> !096 - !105	
12	"EAN128",	// 自动切换编码模式	
13	"25C",	// 25C Check use mod 10-> 奇数先在前面补 0, 10 的倍数-[(奇数位的数字之和<从左至右)+(偶数位数字之和)*3]	
14	"39C",	//39 碼的檢查碼必須搭配「檢查碼相對值對照表」，如表所示，將查出的相對值累加後再除以 43，得到的餘數再查出相對的編碼字元，即為檢查碼字元。	
15	"39",	//Full ASCII 39 Code, 特殊字符用两个可表示的字来表示，39C 同样是包含 Full ASCII, 注意宽窄比处理	
16	"EAN13+2",	// 附加码与主码间隔 7-12 单位，起始为 1011 间隔为 01, (_0*10+_1) Mod 4-> 0--AA 1--AB 2--BA 3--BB	
17	"EAN13+5",	//附加码部分同上，模式((_0+_2+_4)*3+(_1+_3)*9) mod 10 ->"bbaaa", "babaa", "baaba", "baaab", "abbaa", "aabba", "aaabb", "ababa", "abaab", "aabab	
18	"EAN8+2",	// 同 EAN13+2	
19	"EAN8+5",	// 同 EAN13+5	

```

20      "POST", // 详见规格说明，是高低条码，不是宽窄条码
21      "UPCA+2", // 附加码见 EAN
22      "UPCA+5", // 附加码见 EAN
23      "UPCE+2", // 附加码见 EAN
24      "UPCE+5", //附加码见 EAN
25      "CPOST", // 测试不打印。。。
26      "MSIC", //将检查码作为数据再计算一次检查码
27      "PLESSEY", // 测试不打印。。。
28      "ITF14", // 25C 变种， 第一个数前补 0，检查码计算时需扣除最后一个数，
但仍填充为最尾端
29      "EAN14"

```

height

定义条码高度。

unitwidth

定义条码码宽。取值范围：[1, 4]。

各值定义如下：

Width 取值 多级条码单位宽度 (mm) 二进制条码窄线条宽度 二进制条码宽线条宽度

1 0.125 0.125 0.25

2 0.25 0.25 0.50

3 0.375 0.375 0.75

4 0.50 0.50 1.0

rotate

表示条码旋转角度。取值范围：[0, 3]。各值定义如下：

Rotate 取值 定义

0 条码不旋转绘制。

1 条码旋转 90° 绘制。

2 条码旋转 180° 绘制。

3 条码旋转 270° 绘制。

str

文本字符数据流

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawQRCode

描述：在 Page 页指定位置绘制 QRCode 码。

Syntax

- (bool) DrawQRCode:(int)startx starty:(int)starty version:(int)version ecc:(int)ecc
unitwidth:(int)unitwidth rotate:(int)rotate str:(char *)str

Parameters

startx

QRCode 码左上角 x 坐标值，取值范围：[0, Page_Width-1]。

starty

QRCode 码左上角 y 坐标值，取值范围：[0, Page_Height-1]。

version

指定字符版本。取值范围：[0,20]。当 version 为 0 时，打印机根据字符串长度自动计算版本号。

ecc

指定纠错等级。取值范围：[1, 4]。各值定义如下：

ECC 纠错等级

1 L: 7%，低纠错，数据多。

2 M: 15%，中纠错

3 Q: 优化纠错

4 H: 30%，最高纠错，数据少。

unitwidth

QRCode 码码块，取值范围：[1, 4]。各值定义与一维条码指令输入参数 UniWidth 相同。

rotate

QRCode 码旋转角度，取值范围：[0, 3]。各值定义与一维条码指令输入参数 Rotate 相同。

str

QRCode 文本字符数据流

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawPDF417

描述：在 Page 页指定位置绘制 PDF417 条码。

Syntax

- (bool) DrawPDF417:(int)startx starty:(int)starty colnum:(int)colnum lwratio:(int)lwratio
ecc:(int)ecc unitwidth:(int)unitwidth rotate:(int)rotate str:(char *)str

Parameters

startx

PDF417 码左上角 x 坐标值，取值范围：[0, Page_Width-1]。

starty

PDF417 码左上角 y 坐标值，取值范围：[0, Page_Height-1]。

colnum

ColNum 为列数，表述每行容纳多少码字。一个码字为 17*UnitWidth 个点。行数由打印机自动产生，行数范围限定为 3~90。ColNum 的取值范围：[1,30]。

lwratio

宽高比。取值范围：[3,5]。

ecc

纠错等级，取值范围：[0..8]。

ecc 取值 纠错码数 可存资料量（字节）

0 2 1108

1 4 1106

2 8 1101

3 16 1092

4 32 1072

5 64 1024

6 128 957

7 256 804

8 512 496

unitwidth

PDF417 码码块，取值范围：[1, 3]。各值定义与一维条码指令输入参数 UniWidth 相同。

rotate

PDF417 码旋转角度，取值范围：[0, 3]。各值定义与一维条码指令输入参数 Rotate 相同。

str

PDF417 文本字符数据流。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawBitmap

描述：在 Page 页指定位置绘制位图。

Syntax

- (bool) DrawBitmap:(int)startx starty:(int)starty width:(int)width height:(int)height style:(int)style
pdata:(Byte *)pdata

Parameters

startx

位图左上角 x 坐标值，取值范围：[0, Page_Width]。

starty

位图左上角 y 坐标值，取值范围：[0, Page_Height]。

width

位图的像素宽度。

height

位图的像素高度。

style

位图打印特效，各位定义如下：

位 定义

0 反白标志位，置 1 位图反白打印，清零正常打印。

[2:1] 旋转标志位：

00 旋转 0° ；

01 旋转 90° ；

10 旋转 180° ；

11 旋转 270°

[7:3] 保留。

[11:8] 位图宽度放大倍数。

[12:15] 位图高度放大倍数。

pdata

位图的点阵数据。

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks

DrawBitmap

描述：在 Page 页指定位置绘制位图。

Syntax

- (bool) DrawBitmap:(int)startx starty:(int)starty width:(int)width height:(int)height style:(int)style
img:(UIImage *)img nBinaryAlgorithm:(int)nBinaryAlgorithm

Parameters

startx

位图左上角 x 坐标值，取值范围：[0, Page_Width]。

starty

位图左上角 y 坐标值，取值范围：[0, Page_Height]。

width

要打印的宽度。

height

要打印的素高度。

style

位图打印特效，各位定义如下：

位 定义

0 反白标志位，置 1 位图反白打印，清零正常打印。

[2:1] 旋转标志位：

00 旋转 0° ；

01 旋转 90° ；

10 旋转 180° ；

11 旋转 270°

[7:3] 保留。

[11:8] 位图宽度放大倍数。

[12:15] 位图高度放大倍数。

img

要打印的图片。

nBinaryAlgorithm

二值化算法

0 使用抖动算法，对彩色图片有较好的效果。

1 使用平均阈值算法，对文本类图片有较好的效果

Return value

返回 true 表示写入成功；返回 false 表示写入失败。

Remarks