

PRT Android TSPL SDK 说明文档

PRT Android TSPL SDK 说明文档.....	1
1 页标签开始指令:	2
2 页标签结束指令（行模式除外）:	2
3 打印打印机信息:.....	2
4 设置标签纸的间距.....	3
5 走纸:	3
6 清除打印缓冲区内容:	3
7 文本打印，包含一些变体:	3
8 打印方向:	5
9 定义标签的参考起点.....	5
10 定义国际字符集代码。	5
11 条码.....	5
12 打印二维码.....	6
13 标签进纸指定长度.....	7
14 将标签定位到下个标签起始处.....	7
15:打印线条:	7
16 将标签定位到内部传感器确定的起点.....	8
17 打印图片有两个函数:	8
18 打印浓度.....	9
19 打印速度:	9
20 切纸.....	9
21 打印矩形框.....	10
22 设置蜂鸣器鸣叫时间.....	10
23 发数据函数:	10
24 读数据函数:	11

1 页标签开始指令:

```
int printAreaSize(String width, String height)
```

参数:

width: 打印范围的宽。

height: 打印范围的高。

返回:

大于 0: 正常, 否则异常。

例子:

```
printAreaSize("80","200")
```

```
printText("TEXT")
```

```
Print("1","1")
```

2 页标签结束指令 (行模式除外):

```
int Print(String strnum, String strcopies)
```

参数:

strnum: 打印次数。

strcopies: 计数器。

返回:

大于 0: 正常, 否则异常。

3 打印打印机信息:

```
int SelfTest()
```

返回:

大于 0: 正常, 否则异常。

4 设置标签纸的间距

`int Gap(String distance, String offset)`

参数:

Distance: 两个标签的间距 (单位: mm)

Offset: 标签里的内容与标签底部的间距 (单位: mm)

返回:

大于 0: 正常, 否则异常。

5 走纸:

`int Offset(String distance)`

参数:

distance: 走纸距离。

返回:

大于 0: 正常, 否则异常。

例子:

6 清除打印缓冲区内容:

`int CLS()`

返回:

大于 0: 正常, 否则异常。

7 文本打印, 包含一些变体:

(1) `Int printText(String x_pos,String y_pos,String font,String rotation,String x_multiplication,String y_multiplication,String code_data)`

参数:

x_pos: 文字起始 x 坐标:

y_pos: 文字起始 y 坐标:

font:

0: Monotype CG Triumvirate Bold Condensed, font width and height is stretchable

1: 8 x 12 fixed pitch dot font

- 2: 12 x 20 fixed pitch dot font
- 3: 16 x 24 fixed pitch dot font
- 4: 24 x 32 fixed pitch dot font
- 5: 32 x 48 dot fixed pitch font
- 6: 14 x 19 dot fixed pitch font OCR-B
- 7: 21 x 27 dot fixed pitch font OCR-B
- 8: 14 x 25 dot fixed pitch font OCR-A

ROMAN.TTF: Monotype CG Triumvirate Bold Condensed, font width and height proportion is fixed

Rotation: 打印方向。

0 : No rotation

90 : degrees, in clockwise direction

180 : degrees, in clockwise direction

270 : degrees, in clockwise direction

x_multiplication: x 轴方向文字拉伸的倍数。

y_multiplication: y 轴方向文字拉伸的倍数。

code_data: 文本数据。

返回:

大于 0: 正常, 否则异常。

例子:

```
printAreaSize("100","200")
```

```
CLS();
```

```
printText("0","0","1","0","0","0","TEXT")
```

```
Print("1","1")
```

```
(2) int printText(String x_pos,String y_pos,String bold,String rotation,int size,String code_data)
```

参数:

x_pos: 文字起始 x 坐标。

y_pos: 文字起始 y 坐标。

Bold: 是否加粗:

0: 加粗。

1: 不加粗。

Rotation: 打印方向。

0 : No rotation

90 : degrees, in clockwise direction

180 : degrees, in clockwise direction

270 : degrees, in clockwise direction

Size: 字体大小 1~7。

code_data: 文本数据。

返回:

大于 0: 正常, 否则异常。

8 打印方向:

`int Direction(String direction)`

参数:

direction: 0 或 1。

返回:

大于 0: 正常, 否则异常。

例子:

9 定义标签的参考起点

`int Reference(String x_pos,String y_pos)`

参数:

x: 起点 x 轴坐标。

y: 起点 y 轴坐标。

返回:

大于 0: 正常, 否则异常。

10 定义国际字符集代码。

`int Codepage(String codepage)`

参数:

codepage: 代码编号

返回:

大于 0: 正常, 否则异常。

例子:

11 条码

`int printBarcode(String x_pos,String y_pos,String code_type,String height,String`

readable,String rotation,String narrow,String wide,String code_data)

参数:

rotation: 条码方向:

0 : No rotation

90 : degrees, in clockwise direction

180 : degrees, in clockwise direction

270 : degrees, in clockwise direction

Type: 条码类型:

128, 128M, EAN128 , 39 , 93, UPCA , MSI , ITF14

narrow : 窄条的单位宽度。

Wide: 宽条码的单位宽度。

Height: 条码高度。

Readable: 0: not readable

1: human readable

X: 条码的起始横坐标。

Y: 条码的起始纵坐标。

code_data: 条码数据。

返回:

大于 0: 正常, 否则异常。

例子:

12 打印二维码

printQRcode(String x_pos,String y_pos,String ecc_level,String width,String mode,String rotation,String code_data)

参数:

rotation: 条码方向:

0 : No rotation

90 : degrees, in clockwise direction

180 : degrees, in clockwise direction

270 : degrees, in clockwise direction

X: 二维码的起始横坐标。

Y: 二维码的起始纵坐标。

ecc_level: Error correction recovery level

L : 7%

M : 15%

Q : 25%

H : 30%

width: 宽度: 1~10

Mode: M1: (default), original version

M2: enhanced version

Data: 二维码的数据。

返回:

大于 0: 正常, 否则异常。

例子:

13 标签进纸指定长度

int Feed(String len)

参数:

len: 进纸长度 (单位 mm)

返回:

大于 0: 正常, 否则异常。

例子:

14 将标签定位到下个标签起始处

int Formfeed()

返回:

大于 0: 正常, 否则异常。

例子:

15:打印线条:

int Bar(String x_pos,String y_pos,String width, String height)

参数:

x_pos: 起始的 X 坐标。

y_pos: 起始的 Y 坐标。

width: 线条的宽度。

height: 线条的高度。

返回:

大于 0: 正常, 否则异常。

例子:

略

16 将标签定位到内部传感器确定的起点

`int Home()`

参数:

返回:

大于 0: 正常, 否则异常。

例子:

17 打印图片有两个函数:

1) `printImage(String x_pos,String y_pos,String url ,boolean isNegate)`

参数:

`x_pos`: 图片起始的 x 坐标。

`y_pos`: 图片起始的 y 坐标。

`Url`: 图片的路径。

`isNegate`: 图片反白。

`True`: 正常显示。

`False`: 取反显示。

返回:

大于 0 正常; 等于 -1 图片的宽度或者高度超出打印机的范围。

2) `int printImage(String x_pos,String y_pos,Bitmap bmp ,boolean isNegate)`

参数:

`X`: 图片起始的 x 坐标。

`Y`: 图片起始的 y 坐标。

`bmap`: 图片的 `Bitmap` 的对象。

返回:

大于 0 正常；等于-1 图片的宽度或者高度超出打印机的范围。

例子：

18 打印浓度

`int Density(String contrast)`

参数：

Contrast: 0~15

0: specifies the lightest level

15: specifies the darkest level

返回：

大于 0: 正常，否则异常。

例子：

19 打印速度：

`int Speed(String speed)`

参数：

speed

1 1.5 2 2.5 3 3.5 4 5 6 8 10 12

返回：

大于 0: 正常，否则异常。

例子：

20 切纸

`int Cut()`

参数：

返回：

大于 0: 正常，否则异常。

例子：

21 打印矩形框

```
int Box(String x_start,String y_start,String x_end, String y_end,String thickness)
```

参数：

x_start: 左上角 x 坐标。

y_start: 左上角 y 坐标。

x_end: 右下角 x 坐标。

y_end: 右下角 y 坐标。

Thickness: 线条宽度。

返回：

大于 0: 正常，否则异常。

例子：

22 设置蜂鸣器鸣叫时间

```
int Sound(String level, String interval)
```

参数：

level: 蜂鸣声的持续时间，(1/8) 秒为单位指定。

Interval: 间隔的时间

返回：

大于 0: 正常，否则异常。

例子：

23 发数据函数：

```
int WriteData(byte[] bData)
```

参数：

bData: 需要发给打印机的数据。

返回:

大于 0: 正常, 否则异常。

24 读数据函数:

```
int ReadData(byte[] bData)
```

参数:

bData: 从打印机获取的数据。

返回:

大于 0: 正常, 否则异常。