

版本	时间	内容
V1.0	2020.09.09	初始版本
V1.01	2020.10.16	增加打印图片接口说明
V1.02	2021.01.23	增加读写RFID接口说明

## 1 SDK加载及使用

### 2 SDK 连接方法

- 2.1 蓝牙连接接口
- 2.2 WIFI连接接口
- 2.3 USB连接接口
- 2.4 断开连接接口

### 3 SDK 打印接口

- 3.1 设置标签高度
- 3.2 打印
- 3.3 设置编码
- 3.4 标签定位
- 3.5 注释
- 3.6 终止指令
- 3.7 文本打印
- 3.8 计数
- 3.9 设置字符宽高放大倍数
- 3.10 对齐方式
- 3.11 打印条码
- 3.12 打印二维码
- 3.13 打印PFD417码
- 3.14 打印矩形框
- 3.15 打印直线
- 3.16 反白框
- 3.17 打印图片
- 3.18 打印浓度
- 3.19 打印速度
- 3.20 设置行间距
- 3.21 走纸
- 3.22 打印完走纸
- 3.23 蜂鸣器
- 3.24 下划线

- 3.25 延时打印
- 3.26 打印宽度
- 3.27 行模式下设置行间距
- 3.28 设置行模式字体大小与行高
- 3.29 发送数据接口
- 3.30 读取数据接口
- 3.31 字体加粗
- 3.32 获取打印机状态
- 3.33 文本打印自动换行
- 3.34 文字在文本框内居中显示
- 3.35 设置打印机纸张类型
- 3.36 自检页
- 3.37 旋转180度打印
- 3.38 打印完成的开关
- 3.39 获取打印完成时状态
- 3.40 打印机回退
- 3.41 水印
- 3.42 获取打印机SN
- 3.43 设置打印机Codepage
- 3.44 设置QRcode版本
- 3.45 读取QRcode版本
- 3.46 行打印模式打印文本
- 3.47 行打印模式字体加粗
- 3.48 设置行模式的X坐标
- 3.49 RFID 写入
- 3.50 RFID 读取

Tab 1-1

---

## 1 SDK加载及使用

1.1 在Android studio 中加载我们打印机的SDK jar包和SO库。

- 将jar包和so库都拷贝到app/libs文件夹下
- 在app中的build.gradle加入下面的代码

```
android{
    sourceSets {
        main {
            jniLibs.srcDirs = ['libs']
        }
    }
}
implementation files('libs/CPCL_SDK_V1.01.jar')
```

1.2 我们所用的接口都在PrinterHelper这个类中所有的接口都是静态方法，可以直接调用。

---

## 2 SDK 连接方法

### 2.1 蓝牙连接接口

- 描述

```
int portOpenBT(Context context, String portSetting)
```

- 参数

参数	描述
context	上下文对象
portSetting	蓝牙地址 (大写)

- 返回

值	描述
0	连接成功
-1	连接超时
-2	蓝牙地址格式错误
-3	打印机与SDK不匹配 (握手不通过)

---

### 2.2 WIFI连接接口

- 描述

```
int portOpenWIFI(Context context, String printIP)
```

- 参数

参数	描述
context	上下文对象
printIP	IP地址

- 返回

值	描述
0	连接成功
-1	连接超时
-2	IP地址格式错误
-3	打印机与SDK不匹配 (握手不通过)

---

## 2.3 USB连接接口

- 描述

```
int portOpenUSB(Context context, UsbDevice usbdevice)
```

- 参数

参数	描述
context	上下文对象
usbdevice	需要连接的USB设备对象

- 返回

值	描述
0	连接成功
-1	连接超时
-2	参数错误
-3	打印机与SDK不匹配 (握手不通过)

---

## 2.4 断开连接接口

- 描述

```
boolean portClose()
```

- 参数

无

- 返回

值	描述
true	断开成功
false	连接超时

## 3 SDK 打印接口

### 3.1 设置标签高度

- 描述

```
int printAreaSize(String offset, String Horizontal, String Vertical, String
height, String qty)
```

- 参数

参数	描述
offset	上下文对象
Horizontal	打印机水平方向dpi (根据实际打印机dpi设置)
Vertical	打印机垂直方向dpi (根据实际打印机dpi设置)
height	标签高度 (单位: dot) 200dpi 8 dot = 1mm, 300dpi 12 dot = 1mm
qty	打印次数

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form() //打印标签起定位作用（连续纸不可用）
PrinterHelper .Print()
```

## 3.2 打印

- 描述

```
int Print()
```

- 参数  
无
- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form() //打印标签起定位作用（连续纸不可用）
PrinterHelper .Print()
```

## 3.3 设置编码

- 描述

```
int Encoding(String code)
```

- 参数

参数	描述
code	字符编码, gb2312: 中文 , ISO8859-1 : 西欧语 ISO8859-2 : 拉丁语 (2) ISO8859-3 : 拉丁语 (3) ISO8859-4 : 波罗的语 ISO8859-5 : 西里尔语 ISO8859-6 : 阿拉伯语 ISO8859-8 : 希伯来语 ISO8859-9 : 土耳其语 ISO8859-15 : 拉丁语 (9) iso8859-11 : 希腊语 (windows) iso8859-7: 希腊语 (ISO) windows-874: 泰语

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Encoding(gb2312)
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","中文")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.4 标签定位

- 描述  
在Print()之前调用, 只在标签模式下起作用

```
int Form()
```

- 参数  
无
- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form() //打印标签定位作用 (连续纸不可用)
PrinterHelper .Print()
```

## 3.5 注释

- 描述

```
int Note(String note)
```

- 参数

参数	描述
note	注释内容

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Note("注释: ")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.6 终止指令

- 描述

```
int Abort()
```

- 参数  
无
- 返回

值	描述
大于0	发送成功
-1	发送失败

---

## 3.7 文本打印

- 描述

文本打印有三个接口**PrintTextCPCL**和**Text**还有**PrintCodepageTextCPCL**

**PrintTextCPCL**用于中文固件。

**PrintCodepageTextCPCL**用于英文固件。

**Text**两种固件都能用。

```
int Text(String command, String font, String size, String x, String y, String data)
```

- 参数

参数	描述
command	文字的方向 PrinterHelper.TEXT: 水平。 PrinterHelper.TEXT90: 逆时针旋转90度。 PrinterHelper.TEXT180: 逆时针旋转180度。 PrinterHelper.TEXT270: 逆时针旋转270度。
font	字体点阵大小: (单位: dot) 注意: 英文固件只支持 (0和1) 。 0: 12x24。 1: 12x24 (中文模式下打印繁体) , 英文模式下字体变成 (9x17) 大小 2: 8x16。 3: 20x20。 4: 32x32或者16x32, 由ID3字体宽高各放大两倍。 7: 24x24或者12x24, 视中英文而定。 8: 24x24或者12x24, 视中英文而定。 20: 16x16或者8x16, 视中英文而定。 24: 24x24或者12x24, 视中英文而定。 55: 16x16或者8x16, 视中英文而定。 其它默认24x24或者12x24, 视中英文而定。
size	字体大小。 (该功能被屏蔽统一参数传0)
x	横坐标 (单位 dot)
y	纵坐标 (单位 dot)
data	文本数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```

PrinterHelper .printAreaSize("0","200","200","500","1")
PrinterHelper .SetBold("1")//对下面的字体进行加粗 (如不需要加粗不用添加)
PrinterHelper .SetMag("2","2")//对下面的字体进行放大 (如不需要不用添加)
PrinterHelper .Text(PrinterHelper.TEXT,"7","0","10","10","TEXT")
PrinterHelper .SetMag("1","1")//关闭放大
PrinterHelper .SetBold("0")//关闭加粗
PrinterHelper .Form()
PrinterHelper .Print()

```

```
int PrintTextCPCL(String command,int font ,String x,String y,String data,int n,boolean Iscenter,int width)
```

- 参数

参数	描述
command	文字的方向 PrinterHelper.TEXT: 水平。 PrinterHelper.TEXT270: 垂直。
font	字体点阵大小: (单位: dot) 1: 打印繁体字 (24x24或者12x24, 视中英文而定。) 16: 16x16或8x16, 视中英文而定。 24: 24x24或12x24, 视中英文而定。 32: 32x32或16x32, 由ID3字体宽高各放大2倍。
x	横坐标 (单位 dot)
y	纵坐标 (单位 dot)
data	文本数据
n	字体的特效: N&1==1: 加粗 N&2==2: 反白 N&4==4: 倍宽 N&8==8: 倍高
Iscenter	居中 true: 是 false: 否
Width	要居中的范围。 (Iscenter=true时才生效) 单位: dot

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```

PrinterHelper .printAreaSize("0","200","200","500","1")
//15 表示所有的特效都有
PrinterHelper
.PrintTextCPCL(PrinterHelper.TEXT,24,"10","10","TEXT",15,false,0)
PrinterHelper .Form()
PrinterHelper .Print()

```

- 

```

int PrintCodepageTextCPCL(String command,int font ,String x,String
y,String data,int n)

```

- 参数

参数	描述
command	文字的方向 PrinterHelper.TEXT: 水平。 PrinterHelper.TEXT270: 垂直。
font	字体点阵大小: (单位: dot) 0: 12x24。 1: 9x17。
x	横坐标 (单位 dot)
y	纵坐标 (单位 dot)
data	文本数据
n	字体的特效: N&1==1: 加粗 N&2==2: 反白 N&4==4: 倍宽 N&8==8: 倍高

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0","200","200","500","1")
PrinterHelper.Country("ISO8859-1");//设置打印机编码
PrinterHelper.LanguageEncode="iso8859-1";//设置SDK编码
//15表示所有的特效都有
PrinterHelper
.PrintCodepageTextCPCL(PrinterHelper.TEXT,0,"10","10","TEXT",15)
PrinterHelper.Form()
PrinterHelper.Print()
```

## 3.8 计数

- 描述

```
int Count(String ml)
```

- 参数

参数	描述
note	下次加减的数值

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0","200","200","100","2")
PrinterHelper.Text(PrinterHelper.TEXT,"4","0","0","0","10086")
PrinterHelper.Count("10")
PrinterHelper.Text(PrinterHelper.TEXT,"4","0","0","0","10000")
PrinterHelper.Count("-10")
PrinterHelper.Form()
PrinterHelper.Print()
```

## 3.9 设置字符宽高放大倍数

- 描述

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

- 参数

参数	描述
width	字体宽度的放大倍数
height	字体高度的放大倍数

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","500","1")
PrinterHelper .SetBold("1") //对下面的字体进行加粗 (如不需要加粗不用添加)
PrinterHelper .SetMag("2","2") //对下面的字体进行放大 (如不需要不用添加)
PrinterHelper .Text(PrinterHelper.TEXT,"7","0","10","10","TEXT")
PrinterHelper .SetMag("1","1") //关闭放大
PrinterHelper .SetBold("0") //关闭加粗
PrinterHelper .Form()
PrinterHelper .Print()
```

---

## 3.10 对齐方式

- 描述

```
int Align(String align)
```

- 参数

参数	描述
align	PrinterHelper.CENTER: 居中。 PrinterHelper.LEFT: 左对齐。 PrinterHelper.RIGHT: 右对齐。

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Align(PrinterHelper.CENTER)
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

---

### 3.11 打印条码

- 描述

```
int Barcode(String command,String type,String width, String ratio,String
height,String      x,String y,boolean undertext,String number,String
size,String offset, String data )
```

- 参数

参数	描述
command	PrinterHelper.BARCODE: 水平方向 PrinterHelper.VBARCODE: 垂直方向
type	条码类型： PrinterHelper.UPCA,PrinterHelper.UPCA2,PrinterHelper.UPCA5, PrinterHelper.UPCE, PrinterHelper.UPCE2,PrinterHelper.UPCE5 , PrinterHelper.EAN13,PrinterHelper.EAN132,PrinterHelper.EAN135, PrinterHelper.EAN8, PrinterHelper.EAN82,PrinterHelper.EAN85, PrinterHelper.code39, PrinterHelper.code39C,PrinterHelper.F39, PrinterHelper.F39C,PrinterHelper.code93,PrinterHelper.I2OF5, PrinterHelper.I2OF5C,PrinterHelper.I2OF5G,PrinterHelper.code128, PrinterHelper.UCCEAN128,PrinterHelper.CODABAR,PrinterHelper.CODABAR16, PrinterHelper.MSI,PrinterHelper.MSI10,PrinterHelper.MSI1010, PrinterHelper.MSI1110,PrinterHelper.POSTNET,PrinterHelper.FIM
width	窄条的单位宽度
ratio	宽条窄条的比例 0=1.5:1 , 1=2.0:1 , 2=2.5:1 , 3=3.0:1 , 4=3.5:1 , 20=2.0:1 , 21=2.1:1 , 22=2.2:1 , 23=2.3:1 , 24=2.4:1 , 25=2.5:1 26=2.6:1 , 27=2.7:1 , 28=2.8:1 , 29=2.9:1 , 30=3.0:1 ,
height	条码高度
x	条码的起始横坐标。 (单位: dot)
y	条码的起始纵坐标。 (单位: dot)
undertext	条码下方的数据是否可见。 ture: 可见, false: 不可见。
number	字体的类型 (undertext=true才生效)
size	字体的大小(undertext=true才生效)
offset	条码与文字间的距离(undertext=true才生效)
data	条码数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper
.Barcode(PrinterHelper.BARCODE,PrinterHelper.128,"1","1","50","0","0",
true,"7","0","5","123456789")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.12 打印二维码

- 描述

```
int PrintQR(String command, String x, String y, String M, String U,  
String data )
```

- 参数

参数	描述
command	PrinterHelper.BARCODE: 水平方向 PrinterHelper.VBARCODE: 垂直方向
x	二维码的起始横坐标。 (单位: dot)
y	二维码的起始纵坐标。 (单位: dot)
M	QR的类型: 1: 普通类型 2: 在类型1的基础上增加了个别的符号
U	单位宽度/模块的单元高度,范围是1到32默认为6
data	二维码的数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")  
PrinterHelper .PrintQR(PrinterHelper.BARCODE, "0", "0", "2" , "6",  
"123ABC" )  
PrinterHelper .Form()  
PrinterHelper .Print()
```

---

### 3.13 打印PDF417码

- 描述

```
int PrintPDF417(String command, String x, String y, String XD, String YD,  
String C, String S, String data)
```

- 参数

参数	描述
command	PrinterHelper.BARCODE: 水平方向 PrinterHelper.VBARCODE: 垂直方向
x	PDF417的起始横坐标。 (单位: dot)
y	PDF417的起始纵坐标。 (单位: dot)
XD	最窄元素的单位宽度。范围是1到32, 默认为2
YD	最窄元素的单位高度。范围是1到32, 默认值是6
C	使用的列数,数据列不包括启动/停止字符和左/右指标,范围为1到30;默认值是3
S	安全级别表示要检测到的错误的最大金额和/或校正,范围为0到8;默认值是1
data	PDF417码的数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper
.PrintPDF417(PrinterHelper.BARCODE,"0","0","2","6","3","1","123ABC")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.14 打印矩形框

- 描述

```
int Box(String X0,String Y0,String X1,String Y1,String width)
```

- 参数

参数	描述
X0	左上角的X坐标。 (单位: dot)
Y0	左上角的Y坐标。 (单位: dot)
X1	右下角的X坐标。 (单位: dot)
Y1	右下角的Y坐标。 (单位: dot)
width	线条的单位宽度。 (默认: 1)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","200","1")
PrinterHelper .Box("0","0","150","150","1")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.15 打印直线

- 描述

```
int Line(String X0,String Y0,String X1,String Y1,String width )
```

- 参数

参数	描述
X0	起始的X坐标。 (单位: dot)
Y0	起始的Y坐标。 (单位: dot)
X1	结尾的X坐标。 (单位: dot)
Y1	结尾的Y坐标。 (单位: dot)
width	线条的单位宽度。 (默认: 1)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","200","1")
PrinterHelper .Line("10","10","150","10","1")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.16 反白框

- 描述

```
int InverseLine(String X0,String Y0,String X1,String Y1,String width )
```

- 参数

参数	描述
X0	起始的X坐标。 (单位: dot)
Y0	起始的Y坐标。 (单位: dot)
X1	结尾的X坐标。 (单位: dot)
Y1	结尾的Y坐标。 (单位: dot)
width	反白框的宽度。 (默认: 1)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","200","1")
PrinterHelper .InverseLine("10","10","150","10","1")
PrinterHelper .Form()
PrinterHelper .Print()
```

### 3.17 打印图片

- 描述

```
int printBitmapCPCL(Bitmap bitmap,int x,int y,int type,int
compressType,int light)
```

- 参数

参数	描述
bitmap	需打印图片的Bitmap的对象（需自己调节好图片尺寸，200dpi 8px=1mm）
x	图片起始的x坐标。（单位：dot）
y	图片起始的y坐标。（单位：dot）
type	图片算法。 0: 二值算法； 1: 半色调算法 2: 聚合算法
compressType	0: 不压缩， 1: 整体压缩，（适合较小的图片） 2: 分包压缩，（适合较大的图片）
light	亮度（范围 -100 到 100）

- 返回

值	描述
大于0	发送成功
-1	发送失败
-2	bitmap为空
-3	图片数据超过打印机缓冲区

- 例子

```
PrinterHelper.printAreaSize("0","200","200","500","1")
PrinterHelper.printBitmapCPCL(bitmap,0,0,0,0,0)
PrinterHelper.Form()
PrinterHelper.Print()
```

- 描述

该接口只可以由于行模式（里面已包含开头接口和Print接口）。

```
int printBitmap(int x,int y,int type,Bitmap bitmap,int compressType,boolean isform,int segments)
```

- 参数

参数	描述
bitmap	需打印图片的Bitmap的对象（需自己调节好图片尺寸，200dpi 8px=1mm）
x	图片起始的x坐标。（单位：dot）
y	图片起始的y坐标(与上面一行的间距)。（单位：dot）
type	图片算法。 0: 二值算法； 1: 半色调算法 2: 聚合算法
compressType	0: 不压缩， 1: 整体压缩，（适合较小的图片） 2: 分包压缩，（适合较大的图片）
isform	是否定位，（连续纸模式下只能使用false）
segments	分包次数（不小于1）默认1

- 返回

值	描述
大于0	发送成功
-1	发送失败
-2	bitmap为空
-3	图片数据超过打印机缓冲区

- 例子

```
PrinterHelper.printBitmap(0,0,0,bitmap,0,false,1)
```

### 3.18 打印浓度

- 描述

```
int Contrast(String contrast)
```

- 参数

参数	描述
contrast	浓度类型, 总的有四种: 默认 = 0 中 = 1 黑暗 = 2 非常深 = 3

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Contrast("1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

---

### 3.19 打印速度

- 描述

```
int Speed(String speed )
```

- 参数

参数	描述
speed	速度类型, 总的有5种: 从0到5越来越快; 5是理想状态的最快速度。

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Speed("4")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.20 设置行间距

- 描述

```
int SetSp( String setsp)
```

- 参数

参数	描述
Setsp	间距 (单位: 行)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .SetSp(1)
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","50","TEXT")
PrinterHelper .SetSp(5)
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","90","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.21 走纸

- 描述

```
int Prefeed( String prefeed)
```

- 参数

参数	描述
prefeed	走纸的距离。 (单位: dot)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Prefeed("40")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.22 打印完走纸

- 描述

```
int Postfeed( String posfeed)
```

- 参数

参数	描述
posfeed	走纸的距离。 (单位: dot)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
//注意: 要在Form()之后
PrinterHelper .Postfeed("40")
PrinterHelper .Print()
```

---

### 3.23 蜂鸣器

- 描述

```
int Beep( String beep)
```

- 参数

参数	描述
beep	蜂鸣声的持续时间, (1/8) 秒为单位

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Beep("16")
PrinterHelper .Form()
PrinterHelper .Print()
```

## 3.24 下划线

- 描述

```
int Underline(boolean UL)
```

- 参数

参数	描述
UL	true: 添加下划线, false: 取消下划线。

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Underline(true)
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Underline(false)
PrinterHelper .Form()
PrinterHelper .Print()
```

---

## 3.25 延时打印

- 描述

```
int Wait( String wait)
```

- 参数

参数	描述
wait	延时。单位是：1/8秒

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .Wait("80")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

---

### 3.26 打印宽度

- 描述

```
int PageWidth(String pw)
```

- 参数

参数	描述
pw	指定页面宽度。 (单位: dot)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .PageWidth("100")
PrinterHelper .Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper .Form()
PrinterHelper .Print()
```

---

### 3.27 行模式下设置行间距

- 描述

```
int Setlf(String SF)
```

- 参数

参数	描述
SF	间距 (单位: 行)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.PrintData("text1 to print\r\n");
PrinterHelper.Setlf("5");
PrinterHelper.PrintData("text2 to print\r\n");
```

---

### 3.28 设置行模式字体大小与行高

- 描述

```
int Setlp(String font,String size,String spacing )
```

- 参数

参数	描述
font	字体点阵大小: (单位: dot) 注意: 英文固件只支持 (0和1) 。 0: 12x24。 1: 12x24 (中文模式下打印繁体) , 英文模式下字体变成 (9x17) 大小 2: 8x16。 3: 20x20。 4: 32x32或者16x32, 由ID3字体宽高各放大两倍。 7: 24x24或者12x24, 视中英文而定。 8: 24x24或者12x24, 视中英文而定。 20: 16x16或者8x16, 视中英文而定。 24: 24x24或者12x24, 视中英文而定。 55: 16x16或者8x16, 视中英文而定。 其它默认24x24或者12x24, 视中英文而定。
size	字体大小。 (该功能被屏蔽统一参数传0)
spacing	行高 (单位: dot)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.Setlp("5", "2", "32");
PrinterHelper.PrintData("text to print\r\n");
```

## 3.29 发送数据接口

- 描述

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

- 参数

参数	描述
bData	向打印机发送的数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.WriteData(new byte[] {0x0d, 0xa});
```

---

### 3.30 读取数据接口

- 描述

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

- 参数

参数	描述
second	超时时间 (秒)

- 返回

值	描述
数据长度0或者为空	读取失败
大于0	读取成功

- 例子

```
PrinterHelper.ReadData(2);
```

---

### 3.31 字体加粗

- 描述

```
int SetBold(String bold)
```

- 参数

参数	描述
bold	加粗系数 (范围: 1-5)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","500","1")
PrinterHelper .SetBold("1")//对下面的字体进行加粗 (如不需要加粗不用添加)
PrinterHelper .SetMag("2","2")//对下面的字体进行放大 (如不需要不用添加)
PrinterHelper .Text(PrinterHelper.TEXT,"7","0","10","10","TEXT")
PrinterHelper .SetMag("1","1")//关闭放大
PrinterHelper .SetBold("0")//关闭加粗
PrinterHelper .Form()
PrinterHelper .Print()
```

### 3.32 获取打印机状态

- 描述

```
int getstatus()
```

- 参数
- 无
- 返回

值	描述
0	打印机正常
-1	发送失败
2	缺纸
6	开盖

- 例子

```
PrinterHelper .getstatus()//该接口不是实时指令, 打印机正在打印时, 查询无效
```

### 3.33 文本打印自动换行

- 描述

该功能有两个接口分别是AutLine和AutLine2。前者不能使用泰语。后者打印机固件必须在A300 V1.01.40.01以上且文本字节数不得超过1024，超过部分自动忽略。

```
int AutLine(String x, String y, int width, int size, boolean isbole,  
           boolean isdouble, String str)
```

- 参数

参数	描述
x	文字的起始的x坐标。 (单位: dot)
y	文字的起始的y坐标。 (单位: dot)
width	一行打印的宽度。 (单位: dot)
size	字体大小 3: 20x20或10x20, 视中英文而定。 4: 32x32或16x32, 由ID3字体宽高各放大2倍。 8: 24x24或12x24, 视中英文而定。 55: 16x16或8x16, 视中英文而定。
isbole	true: 加粗。 false: 不加粗。
isdouble	字体大小翻倍 true: 放大。 false: 不放大。
str	打印文本

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .AutLine("0","0",100,4,true,true"Text")
PrinterHelper .Form()
PrinterHelper .Print()
```

- 

```
int AutLine2(String x,String y,int width,int size,boolean isbole,boolean
isdouble,
String str)
```

- 参数

参数	描述
x	文字的起始的x坐标。 (单位: dot)
y	文字的起始的y坐标。 (单位: dot)
width	一行打印的宽度。 (单位: dot)
size	字体大小 0: 24x24或12x24, 视中英文而定。 (泰语: 24x48) 1: 7x19 (英文) , 24x24 (繁体) 。 3: 20x20或10x20, 视中英文而定。 4: 32x32或16x32, 由ID3字体宽高各放大2倍。 8: 24x24或12x24, 视中英文而定。 55: 16x16或8x16, 视中英文而定。
isbole	true: 加粗。 false: 不加粗。
isdouble	字体大小翻倍 true: 放大。 false: 不放大。
str	打印文本

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper .printAreaSize("0","200","200","100","1")
PrinterHelper .AutLine2("0","0",100,4,true,true"Text")
PrinterHelper .Form()
PrinterHelper .Print()
```

### 3.34 文字在文本框内居中显示

- 描述

```
int AutCenter(String command, String x, String y, int width, int size, String str)
```

- 参数

参数	描述
command	文字的方向, 总的有两种: PrinterHelper.TEXT: 水平。 PrinterHelper.TEXT270: 垂直。
x	文本框起始的x坐标。 (单位: dot)
y	文本框起始的y坐标。 (单位: dot)
width	文本框的宽度 (单位: dot) 。
size	字体大小。 3: 16x16或8x16, 视中英文而定。 4: 32x32或16x32, 由ID3字体宽高各放大2倍。 8: 24x24或12x24, 视中英文而定。 55: 16x16或8x16, 视中英文而定。
str	要打印的文本。

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0","200","200","100","1")
PrinterHelper.AutCenter(PrinterHelper.TEXT,"0","0",100,4,"Text")
PrinterHelper.Form()
PrinterHelper.Print()
```

### 3.35 设置打印机纸张类型

- 描述

```
void papertype_CPCL(int page)
```

- 参数

参数	描述
page	纸张类型 0: 连续纸 1: 标签纸 2: 后黑标 3: 前黑标 4: 三寸黑标 5: 2寸黑标

- 返回  
无
- 例子

```
//只支持 A300.
PrinterHelper.papertype_CPCL(0) //设置成连续纸
```

### 3.36 自检页

- 描述

```
void setSelf()
```

- 参数  
无
- 返回  
无

- 例子

```
PrinterHelper.setSelf() //调用后打印机会打印一些打印机参数。
```

### 3.37 旋转180度打印

- 描述

```
int PoPrint()
```

- 参数

无

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0","200","200","100","1")
PrinterHelper.Text(PrinterHelper.TEXT,"4","0","0","0","TEXT")
PrinterHelper.Form() //打印标签定位作用（连续纸不可用）
PrinterHelper.PoPrint()
```

### 3.38 打印完成的开关

- 描述

需要跟接口getEndStatus()配合使用，该功能只适用于A300 V1.27.01以上版本

```
void openEndStatic(boolean isopen)
```

- 参数

参数	描述
isopen	true: 开启 false: 关闭

- 返回  
无
- 例子

```
PrinterHelper.openEndStatic(true); //开启
PrinterHelper.PrintData(data); //打印数据
int endStatus = PrinterHelper.getEndStatus(16); //获取状态
PrinterHelper.openEndStatic(false); //关闭
```

### 3.39 获取打印完成时状态

- 描述

```
int getEndStatus(int time)
```

- 参数

参数	描述
time	设置超时时间 (单位: 秒)

- 返回

值	描述
0	发送成功
1	发送失败
2	打印失败 (开盖)
-1	超时 (在设置的时间内打印机没有回馈)

- 例子

```
PrinterHelper.openEndStatic(true); //开启
PrinterHelper.PrintData(data); //打印数据
int endStatus = PrinterHelper.getEndStatus(16); //获取状态
PrinterHelper.openEndStatic(false); //关闭
```

### 3.40 打印机回退

- 描述

```
int ReverseFeed(int feed)
```

- 参数

参数	描述
feed	回退距离。 (单位: 行, 范围: 1-255) 。

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.ReverseFeed(50);
```

---

### 3.41 水印

- 描述

```
int PrintBackground(int x,int y,int size,int background,String data)
```

- 参数

参数	描述
feed	横坐标 (单位: dot)
y	纵坐标 (单位: dot)
size	字体大小 55: 16X16 (dot) 。 24: 24X24 (dot) 。 56: 32X32 (dot) 。 其他: 24X24 (dot) 。
background	背景黑度 (0-255) 。
data	数据

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0", "200", "200", "500", "1");
PrinterHelper.SetMag("8", "8"); //字体放大8倍
PrinterHelper.PrintBackground(0, 0, 56, 150, "A508");
PrinterHelper.SetMag("1", "1"); //还原字体大小
PrinterHelper.Print();
```

---

### 3.42 获取打印机SN

- 描述

```
String getPrintSN()
```

- 参数  
无
- 返回

值	描述
sn	打印机SN

- 例子

```
PrinterHelper.getPrintSN();
```

---

### 3.43 设置打印机Codepage

- 描述

```
int Country(String codepage)
```

- 参数

参数	描述
codepage	代码页 ISO8859-1 : 西欧语 ISO8859-2 : 拉丁语 (2) ISO8859-3 : 拉丁语 (3) ISO8859-4 : 波罗的语言 ISO8859-5 : 西里尔语 ISO8859-6 : 阿拉伯语 ISO8859-8 : 希伯来语 ISO8859-9 : 土耳其语 ISO8859-15 : 拉丁语 (9) WPC1253 : 希腊语 (windows) KU42: 希腊语 (ISO) TIS18: 泰语

- 返回

值	描述
大于0	打印机正常
-1	发送失败

- 例子

```
PrinterHelper.printAreaSize("0", "200", "200", "500", "1");
PrinterHelper.Country("ISO8859-1");
PrinterHelper.LanguageEncode="iso8859-1";
PrinterHelper
.PrintCodepageTextCPCL(PrinterHelper.TEXT, 0, "10", "10", "TEXT", 15)
PrinterHelper .Print()
```

## 3.44 设置QRcode版本

- 描述

该接口用于设置二维码的版本号，设置后不会因为二维码的内容而改变大小，只有部分机型和版本支持（可以询问客服）

```
int setQRcodeVersion(int version)
```

- 参数

参数	描述
version	版本号 (范围0-40) QR版本默认为00, QR版本为00时, QR码效果同旧版, 宽高会随数据量而改变。设置了版本号对二维码的数据量有范围要求, 超出范围二维码不打印。 具体查看结尾的表1-1.

- 返回

值	描述
大于0	发送成功
-1	发送失败
-2	超出参数范围

- 例子

```
PrinterHelper .setQRcodeVersion(20);
```

### 3.45 读取QRcode版本

- 描述

```
String getQRcodeVersion()
```

- 参数  
无
- 返回

值	描述
大于0	QRcode版本 (失败返回空)

- 例子

```
String version = PrinterHelper .getQRcodeVersion();
```

### 3.46 行打印模式打印文本

- 描述

```
int PrintData(String str)
```

- 参数

参数	描述
str	文本内容 (以\r\n结尾)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.PrintData("要打印的文本\r\n")
```

### 3.47 行打印模式字体加粗

- 描述

```
int RowSetBold(String bold)
```

- 参数

参数	描述
bold	加粗倍数, 1: 关闭加粗 其他: 加粗倍数

- 返回

值	描述
大于0	打印机正常
-1	发送失败

- 例子

```
PrinterHelper.Setlp("5","2","46")
PrinterHelper.RowSetBold("2")
PrinterHelper.PrintData("要打印的文本\r\n")
PrinterHelper.RowSetBold("1") //注意关闭加粗以免影响下面的打印内容
```

### 3.48 设置行模式的X坐标

- 描述

必须放在Setlp函数之前

```
int RowSetX(String x)
```

- 参数

参数	描述
bold	横坐标 (单位: dot)

- 返回

值	描述
大于0	发送成功
-1	发送失败

- 例子

```
PrinterHelper.RowSetX("200");
PrinterHelper.Setlp("5","2","32");
PrinterHelper.RowSetBold("2");
PrinterHelper.PrintData("text to print\r\n");
PrinterHelper.RowSetBold("1");
```

### 3.49 RFID 写入

- 描述

```
printRFIDCPCL(List<RFIDBeen> rfidBeens, Bitmap bitmap, int x, int y, int type,
                int compressType, int density)
```

- 参数

参数	描述
rfidBeens	RFID对象集合
memory	储存区, 0: 保留区 1:EPC区、 3: user区
address	起始地址, memory=0-->address(0-3)、 memory=1-->address(2-7)、 memory=3-->address(0-255)
data	需写入的数据, memory=0--> (address+data的字节长度<=4) 、 memory=1--> (address+data的字节长度<=26) 、 memory=3-->(address+data的字节长度<=256)
bitmap	图片对象
x	横坐标
y	纵坐标
type	图片算法, 0: 二值算法 、 1: 半色调算法
compressType	压缩类型, 0: 不压缩、 1: 整体压缩、 2: 分包压缩
density	浓度 (-1不设置浓度)

- 返回

值	描述
0	发送成功
-1	发送失败
-2	参数错误
-3	图片数据太大超出打印机压缩空间
-4	RFID写入失败 (打印机回传)

- 例子

```

List<RFIDBeen> rfidBeenList = new ArrayList<>();
RFIDBeen rfidBeen = new RFIDBeen();
rfidBeen.setMemory(1);
rfidBeen.setAddress(2);
rfidBeen.setData("123456");
rfidBeenList.add(rfidBeen);
Log.d("Print", "testRFID:
"+PrinterHelper.printRFIDCpcl(rfidBeenList,null,0,0,0,0,0));

```

## 3.50 RFID 读取

- 描述

```
readRFIDCPCL(List<RFIDBeen> rfidBeens)
```

- 参数

参数	描述
rfidBeens	RFID对象集合
memory	储存区, 0: 保留区、1:EPC区、2:TID区、3:user区
address	起始地址, memory=0-->address(0-3)、 memory=1-->address(2-7)、 memory=2-->address(2-7)、 memory=3-->address(0-255)
length	读取长度, memory=0-->length(1-8)、 memory=1-->length(1-24)、 memory=2-->length(1-24)、 memory=3-->length(1-256)

- 返回

值	描述
List	RFID对象集合 (空表示失败)
data	读取的数据

- 例子

```
List<RFIDBeen> rfidBeenList = new ArrayList<>();
RFIDBeen rfidBeen = new RFIDBeen();
rfidBeen.setMemory(1);
rfidBeen.setAddress(2);
rfidBeen.setLength(24);
rfidBeenList.add(rfidBeen);
List<RFIDBeen> rfidBeenListResult =
PrinterHelper.readRFIDCPCL(rfidBeenList);
if (rfidBeenListResult==null){
    Log.d("Print", "testRFIDRead: null");
    return;
```

```

    }
    for (int i = 0; i < rfidBeenListResult.size(); i++) {
        Log.d("Print", "testRFIDRead:
" + rfidBeenListResult.get(i).toString());

        ToastUtility.show(mContext, rfidBeenListResult.get(i).toString());
    }
}

```

**Tab 1-1**

<b>Version</b>	<b>Error Level</b>	<b>Number of data</b>
1	L M Q H	19 16 13 9
2	L M Q H	34 28 22 16
3	L M Q H	55 44 34 26
4	L M Q H	80 64 48 36
5	L M Q H	108 86 62 46
6	L M Q H	136 108 76 60
7	L M Q H	156 124 88 66
8	L M Q H	194 154 110 86
9	L M Q H	232 182 132 100
10	L M Q H	274 216 154 122
11	L M Q H	324 254 180 140
12	L M Q H	370 290 206 158
13	L M Q H	428 334 244 180
14	L M Q H	461 365 261 197
15	L M Q H	523 415 295 223
16	L M Q H	589 453 325 253
17	L M Q H	647 507 367 283
18	L M Q H	721 563 397 313
19	L M Q H	795 627 445 341

20	L M Q H	861 669 485 385
21	L M Q H	932 714 512 406
22	L M Q H	1006 782 568 442
23	L M Q H	1094 860 614 464
24	L M Q H	1174 914 664 514
25	L M Q H	1276 1000 718 538
26	L M Q H	1370 1062 754 596
27	L M Q H	1468 1128 808 628
28	L M Q H	1531 1193 871 661
29	L M Q H	1631 1267 911 701
30	L M Q H	1735 1373 985 745
31	L M Q H	1843 1455 1033 793
32	L M Q H	1955 1541 1115 845
33	L M Q H	2071 1631 1171 901
34	L M Q H	2191 1725 1231 961
35	L M Q H	2306 1812 1286 986
36	L M Q H	2434 1914 1354 1054
37	L M Q H	2566 1992 1426 1096
38	L M Q H	2702 2102 1502 1142
39	L M Q H	2812 2216 1582 1222
40	L M Q H	2956 2334 1666 1276