

# Android CPCL 编程手册

**v3.1.1**

# 1. 介绍

这个手册介绍怎么通过 CPCL 指令实现标签打印。常量定义再 CPCLConst 类中。

## 2. CPCLPrinter

### 2.1. CPCLPrinter

构造函数，创建打印对象。

CPCLPrinter(IDeviceConnection connection)

#### 【参数】

➤ connection

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

### 2.2. initializePrinter

标签初始化

CPCLPrinter initializePrinter(int height)

CPCLPrinter initializePrinter(int height, int qty)

CPCLPrinter initializePrinter(int offset, int height, int qty)

#### 【参数】

➤ offset

标签的横向偏移量，默认为 0

➤ height

标签的最大高度

➤ qty

要打印的标签数量，默认为 1

#### 【返回值】

CPCLPrinter 对象

### 2.3. addText

文本打印

CPCLPrinter addText(int x, int y, String content)

CPCLPrinter addText(int x, int y, String font, String content)

CPCLPrinter addText(int x, int y, String rotation, String font, String content)

【参数】

➤ x

文本的起始 x 值

➤ y

文本的起始 y 值

➤ font

文本的字体类型，默认 FNT\_0。支持 FNT\_0、FNT\_1、FNT\_2、FNT\_4、FNT\_5、FNT\_6、FNT\_7、FNT\_24、FNT\_55。

➤ rotation

顺时针旋转角度，默认 ROTATION\_0

变量	描述
ROTATION_0	不旋转
ROTATION_90	顺时针旋转 90 度
ROTATION_180	顺时针旋转 180 度
ROTATION_270	顺时针旋转 270 度

➤ content

文本内容

【返回值】

CPCLPrinter 对象

## 2.4. setmag

将常驻字体放大指定倍数

CPCLPrinter setmag(int w, int h)

【参数】

➤ w

宽度放大倍数。有效放大倍数为 1 到 16。

➤ h

高度放大倍数。有效放大倍数为 1 到 16。

【返回值】

CPCLPrinter 对象

## 2.5. addPrint

整个命令集的结束命令，将会启动文件打印。

void addPrint()

【返回值】

无

2.6. addBarcode

绘制一维条码

CPCLPrinter addBarcode(int x, int y, String type, int height, String data)

CPCLPrinter addBarcode(int x, int y, String type, int width, int ratio, int height, String data)

横向一维条码

CPCLPrinter addBarcodeV(int x, int y, String type, int height, String data)

CPCLPrinter addBarcodeV(int x, int y, String type, int width, int ratio, int height, String data)

纵向一维条码

【参数】

➤ x

条码起始点横坐标，单位为点

➤ y

条码起始点纵坐标，单位为点

➤ type

条码类型

变量	描述
BCS_128	Code 128
BCS_UPCA	UPC-A
BCS_UPCE	UPC-E
BCS_EAN13	EAN/JAN-13
BCS_EAN8	EAN/JAN-8
BCS_39	Code 39
BCS_93	Code 93/Ext.93
BCS_CODABAR	Codabar

➤ width

窄条的单位宽度，默认为 1。

➤ ratio

宽条与窄条的比率，默认 BCS\_RATIO\_1

变量	宽条:窄条	变量	宽条:窄条
BCS_RATIO_0	1.5 :1	BCS_RATIO_23	2.3:1
BCS_RATIO_1	2.0 :1	BCS_RATIO_24	2.4:1
BCS_RATIO_2	2.5 :1	BCS_RATIO_25	2.5:1
BCS_RATIO_3	3.0 :1	BCS_RATIO_26	2.6:1
BCS_RATIO_4	3.5 :1	BCS_RATIO_27	2.7:1
BCS_RATIO_20	2.0:1	BCS_RATIO_28	2.8:1
BCS_RATIO_21	2.1:1	BCS_RATIO_29	2.9:1

BCS_RATIO_22	2.2:1	BCS_RATIO_30	3.0:1
--------------	-------	--------------	-------

- height  
条码的单位高度。
- data  
条码数据。

【返回值】  
CPCLPrinter 对象

## 2.7. addBarcodeText

添加条码注释  
CPCLPrinter addBarcodeText()

【返回值】  
CPCLPrinter 对象

## 2.8. addBarcodeTextOff

取消条码注释  
CPCLPrinter addBarcodeTextOff()

【返回值】  
CPCLPrinter 对象

## 2.9. addQRCode

绘制二维条码  
CPCLPrinter addQRCode(int x, int y, String data)  
CPCLPrinter addQRCode(int x, int y, int codeModel, int cellWidth, String data)

- 【参数】
- x  
二维码起始横坐标
  - y  
二维码起始纵坐标
  - codeModel  
QR Code 规范编号，默认为 QRCODE\_MODE\_ENHANCE。

变量	描述
----	----

QRCODE_MODE_ORG	原始规范
QRCODE_MODE_ENHANCE	增强后的规范

➤ **cellWidth**

单元格大小，范围【1，32】，默认为 6。

➤ **data**

二维码资料内容

**【返回值】**

CPCLPrinter 对象

## 2.10. addBox

绘制矩形

CPCLPrinter addBox(int x, int y, int width, int height, int thickness)

**【参数】**

➤ **x**

矩形起始横坐标，单位为点

➤ **y**

矩形起始纵坐标，单位为点

➤ **width**

矩形宽度，单位为点

➤ **height**

矩形高度，单位为点

➤ **thickness**

线条宽度

**【返回值】**

CPCLPrinter 对象

## 2.11. addLine

绘制线条

CPCLPrinter addLine(int x, int y, int xend, int yend, int thickness)

**【参数】**

➤ **x**

线条起始横坐标，单位为点

➤ **y**

线条起始纵坐标，单位为点

➤ **xend**

线条结束横坐标，单位为点

➤ **yend**

线条结束纵坐标，单位为点

➤ **thickness**

线条宽度

【返回值】

CPCLPrinter 对象

## 2.12. addInverseLine

将指定区域的数据黑白反向显示

CPCLPrinter addInverseLine(int x, int y, int xend, int yend, int width)

【参数】

➤ **x**

反显区域起始横坐标，单位为点

➤ **y**

反显区域起始纵坐标，单位为点

➤ **xend**

反显区域结束横坐标，单位为点

➤ **yend**

反显区域结束纵坐标，单位为点

➤ **width**

反显区域宽度

【返回值】

CPCLPrinter 对象

## 2.13. addGraphics

图片打印，推荐使用 addCGraphics。

CPCLPrinter addCGraphics(int x, int y, int width, Bitmap bmp)

CPCLPrinter addCGraphics(int x, int y, int width, Bitmap bmp, AlgorithmType algorithmType)

使用 **byte** 类型传输图片数据

CPCLPrinter addEGraphics(int x, int y, int width, Bitmap bmp)

CPCLPrinter addEGraphics(int x, int y, int width, Bitmap bmp, AlgorithmType algorithmType)

使用 **String** 类型传输图片数据

【参数】

➤ **x**

图片起始横坐标，单位为点

➤ y

图片起始纵坐标，单位为点

➤ width

图片的打印宽度

➤ bmp

待打印的图片对象。

➤ algorithmType

算法类型。默认使用 `AlgorithmType.Threshold`。

Dithering 抖动算法

Threshold 二值法

**【返回值】**

CPCLPrinter 对象

## 2.14. addAlign

设置字段的对齐方式。默认情况下，打印机将左对齐所有字段。对齐命令将对所有后续字段保持有效，直至指定了其他对齐命令。

CPCLPrinter addAlign(int align)

CPCLPrinter addAlign(int align, int end)

**【参数】**

➤ align

对齐方式

变量	描述
ALIGNMENT_LEFT	左对齐所有后续字段
ALIGNMENT_CENTER	居中对齐所有后续字段
ALIGNMENT_RIGHT	右对齐所有后续字段

➤ end

对齐的结束点。

**【返回值】**

CPCLPrinter 对象

## 2.15. printerStatus

查询打印机状态

void printerStatus(IStatusCallback callback)



void printerStatus(int timeout, IStatusCallback callback)

【参数】

➤ callback

状态回调

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

status(HEX)	描述
00	正常
01	前盖开
02	卡纸
03	卡纸且前盖开
04	缺纸
05	缺纸且前盖开
08	无色带
09	无色带且前盖开
0A	无色带且卡纸
0B	无色带、卡纸且前盖开
0C	无色带、缺纸
0D	无色带、缺纸且前盖开
10	暂停
20	打印中
80	其他错误
-1	读取超时或数据异常

➤ timeout

超时时间，单位为毫秒。默认 5000ms。

【返回值】

CPCLPrinter 对象

## 2.16. addSpeed

设置打印速度

CPCLPrinter addSpeed(int level)

【参数】

➤ level

速度级别，范围【0，5】。

【返回值】

CPCLPrinter 对象

## 2.17. addPageWidth

设置打印宽度

CPCLPrinter addPageWidth(int width)

### 【参数】

➤ width

页面的单位宽度。

### 【返回值】

CPCLPrinter 对象

## 2.18. addBeep

此方法用于让打印机蜂鸣器发出给定时间长度的声音，未配备蜂鸣器的打印机将忽略此方法。

CPCLPrinter addBeep(int length)

### 【参数】

➤ length

蜂鸣持续时间，以 1/8 秒为单位，例如 16 表示发声时间为 2 秒。

### 【返回值】

CPCLPrinter 对象

## 2.19. getBtMac

获取蓝牙 mac 地址

void getBtMac(int timeout, IStrCallback callback)

### 【参数】

➤ timeout

超时时间，单位为 ms。

➤ callback

mac 地址回调

```
public interface IStrCallback {  
    void receive(String info);  
}
```

info 为返回的 mac 地址，为空字符串表示获取失败。

【返回值】

void

## 2.20. isConnect

查询连接状态

void isConnect(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

## 2.21. setCharSet

设置将打印内容传输给打印机所采用的字符编码，默认编码为“gbk”

void setCharSet(String charSet)

【参数】

➤ charSet

打印机所能识别的字符编码类型

## 2.22. sendData

该方法用于发送数据到打印机。

CPCLPrinter sendData(byte[] data);

CPCLPrinter 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】

CPCLPrinter 对象

## ImageUtils

### handleImageEffect

这个方法用于调整图片的对比度和亮度。

```
static Bitmap handleImageEffect(Bitmap bmp, float contrast, float brightness)
```

#### 【参数】

- **bmp**  
原图片
- **contrast**  
对比度，范围 0~2
- **brightness**  
亮度，范围-255~255

#### 【返回值】

调整后的图片对象