

Flutter POS 编程手册

v1.2.0

1.介绍

通过阅读本手册,开发者可以快速掌握如何利用 Flutter 实现票据打印功能,并将其应用到实际的开发中去。包含 PrinterManager 类和 POSCommand 类的使用,以及 POSConst 类中各常量的含义及用法

2.PrinterManager

2.1.PrinterManager

构造函数,创建打印机管理对象。

PrinterManager()

2.2.connectBt

该方法用于蓝牙连接（android 使用经典蓝牙，iOS 使用 BLE）

Future<void> connectBt(String address, Function(int result) callback)

【参数】

➤address

蓝牙地址

➤callback

连接状态回调

2.3.connectUsb

该方法用于 USB 连接（仅支持 android）

Future<void> connectUsb(String path, Function(int result) callback)

【参数】

➤path

usb 地址

➤callback

连接状态回调

2.4.getUsbPaths

该方法用于获取当前连接的 USB 设备列表（仅支持 andirod）

Future<List> getUsbPaths()

【返回值】

List 对象：usb 地址列表集合

2.5.searchCallback

蓝牙搜索回调

searchCallback(Map args)

【返回值】

蓝牙信息的 Map 字典对象：key: address, value: name

2.6.sendESC

此方法用于发送票据指令

Future<void> sendESC(List<Map<String, dynamic>> data)

2.7.checkPermissions

此方法用于检查安卓蓝牙授权情况

Future<bool> checkPermissions()

【返回值】

bool 对象：是否授权成功

2.8.startScan

此方法用于 Bluetooth 搜索（android: 经典蓝牙搜索，iOS: BLE 搜索）

Future<void> startScan()

2.9.disconnect

断开连接

Future<void> disconnect()

2.10.posprinterStatus

这个方法用于票据打印机常用状态查询，查询超时时间 5000s

Future<int> posPrinterStatus()

对应的 status 值见下表

小于 0	-1 其他错误 -3:连接断开 -4:接收数据超时
STS_NORMAL	打印机正常
STS_COVEROPEN	上盖开
STS_PAPEREMPTY	打印机缺纸
STS_PRESS_FEED	按下走纸键
STS_PRINTER_ERR	打印机错误

2.11.checkIsConnect

查询连接状态

Future<int> checkIsConnect()

【返回值】

status	描述
STS_CONNECT	连接
STS_DISCONNECT	断开

2.12.getSerialNumber

获取打印机序列号

Future<String> getSerialNumber()

【返回值】

采用回调的方式获取查询到的 SN 码的字节数组。

2.13.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

2.14.cashBoxCheck

这个方法用于钱箱状态查询。

Future<int> cashBoxCheck()

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

3.POSCommand

3.1.POSCommand

构造函数，创建 POS 指令对象。

POSCommand()

3.2.printString

该方法用于普通文本的打印

POSCommand printString(String text)

【参数】

➤text

打印的文本字符串

【返回值】

POSCommand 对象

3.3.printText

这个方法用于特定格式的文本打印

POSCommand printText(String text, {int alignment = POSConst.ALIGNMENT_LEFT, int attribute = POSConst.FNT_DEFAULT, int textSize = POSConst.TXT_1WIDTH | POSConst.TXT_1HEIGHT})

【参数】

➤text

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，text 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】

POSCommand 对象

3.4.printBitmap

这个方法用于打印图片
POSCommand printBitmap(uint8List bitmap,int width, {int alignment = POSConst.ALIGNMENT_LEFT,int mode = POSConst.BMP_NORMAL})

【参数】

➤bitmap

需要打印的位图数据

➤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
- mode
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

【返回值】
POSCommand 对象

3.5.printBarcode

这个方法用于打印一维条码

POSCommand printBarcode(String data, int codeType, {int width = 1, int height = 1, int alignment = POSConst.ALIGNMENT_LEFT, int textPosition = POSConst.HRI_TEXT_BELOW})

- 【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】

POSCommand 对象

3.6.feedLine

这个方法用于打印并向前走纸 n 行

POSCommand feedLine({int lineCount = 1})

【参数】

➤lineCount

走纸的行数。默认为 1

【返回值】

POSCommand 对象

3.7.printQRCode

这个方法用于打印二维码

POSCommand printQRCode(String content,{int moduleSize = 8, int ecLevel = POSConst.QRCODE_EC_LEVEL_L, int alignment = POSConst.ALIGNMENT_CENTER})

【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】

POSCommand 对象

3.8.cutPaper

这个方法用于切纸

POSCommand cutPaper({int model = POSConst.CUT_HALF})

POSCommand 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】

POSCommand 对象

3.9.openCashBox

这个方法用于打开钱箱抽屉

POSCommand openCashBox(int pinNum, {int onTime = 30, int offTime = 255})

【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】

POSCommand 对象

3.10.setCharSet

设置字符编码

POSCommand setCharSet(String charSet)

【参数】

➤charSet

编码类型名

【返回值】

POSCommand 对象

3.11.setAlignment

这个方法用于设置文本的对齐方式

POSCommand 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】

POSCommand 对象

3.12.setPrintArea

页模式下设置打印区域

POSCommand setPrintArea(int width, int height, {int x = 0, int y = 0})

【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】

POSCommand 对象

3.13.setPageModel

这个方法用于开启或者关闭页模式

POSCommand setPageModel(bool isOpen)

【参数】

➤isOpen

true 表示开启页模式，false 表示关闭页模式

【返回值】

POSCommand 对象

3.14.printPageModelData

打印页模式下的数据，并回到标准模式。

POSCommand printPageModelData()

【返回值】

POSCommand 对象

3.15.setPrintDirection

设置页模式下的打印区域方向

POSCommand 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】

POSCommand 对象

3.16.setAbsoluteHorizontal

设置横向绝对打印位置

POSCommand setAbsoluteHorizontal(int position)

【参数】

➤position
开始的位置

【返回值】

POSCommand 对象

3.17.setRelativeHorizontal

设置相对横向打印位置

POSCommand setRelativeHorizontal(int position)

【参数】

➤position
开始的位置

【返回值】

POSCommand 对象

3.18.setAbsoluteVertical

页模式下设置纵向绝对位置

POSCommand setAbsoluteVertical(int position)

【参数】

➤position
开始的位置

【返回值】

POSCommand 对象

3.19.setRelativeVertical

页模式下设置纵向相对位置

POSCCommand setRelativeVertical(int position)

【参数】

➤position
开始的位置

【返回值】

POSCCommand 对象

3.20.downloadNVImage

保存 NV 图片到 flash 中

POSCCommand downloadNVImages(List<UInt8List> bitmaps, int imageWidth)

【参数】

➤bitmaps
需要保存的 Bitmap 对象列表
➤imageWidth
图片宽度

【返回值】

POSCCommand 对象

3.21.printNVImage

打印存储在 Flash 中的图片

POSCCommand 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】

POSCCommand 对象

3.22.initializePrinter

初始化打印机

POSCommand initializePrinter()

【返回值】

POSCommand 对象

3.23.selectBitmapModel

选择位图模式

POSCommand selectBitmapModel(int model, int width, Uint8List 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

需要打印的位图数据

【返回值】

POSCommand 对象

3.24.feedDot

打印缓冲区数据并走纸 n 点。

POSCommand feedDot(int n)

【参数】

➤n

走纸的距离，单位为横向或纵向移动单位。默认为点。

【返回值】

POSCommand 对象

3.25.setLineSpacing

设置行高

POSCCommand 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】

POSCCommand 对象

3.26.setTurnUpsideDownMode

选择或取消倒置打印模式

POSCCommand setTurnUpsideDownMode(bool on)

【参数】

➤on

true 表示选择 false 表示取消

【返回值】

POSCCommand 对象

3.27.selectCodePage

选择字符代码表

POSCCommand 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]

【返回值】

POSCommand 对象

3.28.selectCharacterFont

选择字体

POSCommand 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】

POSCommand 对象

3.29.sendData

发送自定义数据到打印机

POSCommand sendData(UInt8List data)

【参数】

➤data

自定义数据

【返回值】

POSCommand 对象