

Android Printer-SDK V1.0

一、简介

当前 SDK 基于蓝牙 封装了 TSPL、ESC 指令的打印方式。其中 TSPL 指令采取小图打印方式，在小图打印中，处理打印方向，图片旋转等坐标运算功能，速度可比全图打印快三倍以上。ESC 封装了字体大小、粗体、图片打印、多张图片打印等等功能。

打印图片更是封装了普通的二值化与基于 Floyd Steinberg 的错误抖动算法，让你打印出迷惑你的眼睛的黑白图片。

当前 SDK，还支持得到打印机的信息。连接蓝牙之后，可一键得到打印机的序列号、型号、纸张类型、纸张状态、指令类型。

二、打印接口说明

类名	功能
BluetoothManager	蓝牙扫描、配对、连接、读取、发送的管理类
PrintfESCManager	ESC 指令的打印管理类
PrintfInfoManager	打印机信息的管理类
PrintfTSPLManager	TSPL 指令的打印管理类
ParameterUtil	参数的设置类
BarcodeUtil	ESC 指令的条码类
ParameterUtil	一些参数的设置

注：带 Manager 结尾的类，均为单例，请用 getInstance()得到对象

BluetoothManager 类中的方法

1: isConnect(): 判断蓝牙是否连接。

return true: 当前蓝牙已经连接

return false: 当前蓝牙没有连接

2: addConnectResultCallBack(ConnectResultCallBack connectResultCallBack): 增加蓝牙连接结果的回调

ConnectResultCallBack 中的回调方法:

1) :void success(BluetoothDevice device); 当蓝牙连接成功时回调

2) :void close(BluetoothDevice device); 当蓝牙连接关闭时回调

3) :void fail(BluetoothDevice device); 当蓝牙连接失败时回调

注意：当页面销毁时，需要调用 removeConnectResultCallBack 方法移除页面的监听，防止内存

泄露

3: addScanBlueCallBack(ScanBlueCallBack scanBlueCallBack): 增加蓝牙扫描回调

ScanBlueCallBack 中的回调方法:

1) :void scanDevice(BluetoothModel bluetoothModel); 当扫描到蓝牙时回调。BluetoothModel 是一个封装了蓝牙名称、蓝牙 mac、蓝牙状态的类。详见 BluetoothModel 中的代码。

注意: 当页面销毁时, 需要调用 removeScanBlueCallBack 方法移除页面的监听, 防止内存泄露

4: connectLastBluetooth(): 连接上一次连接的蓝牙。如果没有上一次, 则不连接。

5: sendBytesToReadAsync(final byte[] bytes, final int time, final int number, final SendBytesToReadCallBack sendBytesToReadCallBack): 发送数据, 并且异步读取。

参数说明:

byte[] bytes: 需要发送的数据

time: 每一次读取前, 等到的时间 单位毫秒

number: 假如老是读取不到, 需要重复读取的次数

SendBytesToReadCallBack : 读取结果的回调

void callBytes(byte[] bytes); 当前方法, 是读取结果的回调

void callError(int error); 当前方法, 是读取失败的结果回调。

error == 1 当前发送的数据, 没有返回值

error == 2 当前发送的数据, 在处理的时候, 发送的异常

error == 3 蓝牙未连接

额外说明: time 与 number 参数, 总的等到时间不可以超过 20 秒。假如超过了 20 秒, 会自动把 time 参数置为 500, number 参数置为 40

sendBytesToRead(final byte[] bytes, final int time, final int number): 发送数据, 并且同步读取。参数说明与 sendBytesToReadAsync 类似。只是当前方法没有错误码回调。当发送错误时, return null。

6: beginSearch(): 开始搜索。搜索到的蓝牙, 会通过监听回调回调给需要的页面。

return: 1 开始扫描成功 2 开启扫描失败 蓝牙适配器未打开

stopSearch(): 停止搜索。

7: pairBluetooth(String mac): 配对蓝牙。参数传入 mac。当前 mac 只可从 BluetoothManager 中搜索而来, 不可以从其他路径中来。因为, 只有从 BluetoothManager 中来, 才可以从 mac 得到 BluetoothDevice 对象。

pairBluetooth(BluetoothDevice device): 配对蓝牙。参数传入 BluetoothDevice 对象。当前对象, 可以从任何路径中来。

注: BluetoothDevice 与 mac 方法, 配对过的与没有配对过的设置, 都是可以调用当前方法的。

8: write(byte[] data): 发送数据给当前连接的设备。

return:

-1:数据发送失败 蓝牙未连接

1:数据发送成功

-2:数据发送失败 抛出异常 失败

9: close(): 关闭当前连接

PrintfESCManager 类中的方法

1: printfBarcode(byte barcodeType, int param1, int param2, int param3, String content):
指令的方式, 打印一维码与二维码。

参数说明:

barcodeType: 条码的类型。详见 BarcodeUtil.BarcodeType 类

param1: 条码的横向宽度 $2 \leq n \leq 6$ 默认为 2

param2: 条码的高度 $1 \leq n \leq 255$ 默认 162

param3: 条码的注释位置 0 不打印 1 上方 2 下方 3 上下都有

content: 当前条码的内容 S

2: setExtraLargeFontSize(): 设置打印机的字体为特大

setLargeFontSize(): 设置打印机的字体为大

setDefaultPrinterFontSize: 设置打印机的字体为默认

3: printfESCPrinterModelAsync(final List<ESCPrinterModel> escPrinterModels

,final MultiplePrintfResultCallBack multiplePrintfResultCallBack): 异步打印多组、多张

图片

参数说明:

List<ESCPrinterModel> escPrinterModels: 详细请看 ESCPrinterModel 类。

MultiplePrintfResultCallBack multiplePrintfResultCallBack: 请看文档结尾的

MultiplePrintfResultCallBack 接口说明。

4: printfBitmapAsync(Bitmap bitmap, int left): 打印图片

参数说明:

Bitmap bitmap: 当前需要打印的图片

int left: 打印图片的左边距

5: printfBitmapsAsync(final Bitmap bitmap, final int left,final int
number,MultiplePrintfResultCallBack multiplePrintfResultCallBack): 异步打印一张图片多次

参数说明:

Bitmap bitmap: 当前需要打印的图片

int left: 打印图片的左边距

int number: 打印图片的打印张数

MultiplePrintfResultCallBack multiplePrintfResultCallBack: 请看文档结尾的

MultiplePrintfResultCallBack 接口说明。

6: setPrinterCode(String code): 设置打印的编码。

注: 当前方法, 在 printfText(String text) 中生效

getPrinterCode()

return: 返回当前的编码

7: `setPrinterBold()`: 设置打印机为粗体模式
`setPrinterNoBold()`: 设置打印机为不粗体模式

8: `initPrinter()`: 初始化打印机

9: `printfText(String text)`: 打印文字

10: `printfTable(List<List<String>> datess,int effectiveWidth)`: 打印表格。

参数说明:

`List<List<String>> datess`: 一个二维码数据。对应表格的行列

`int effectiveWidth`: 有效打印宽度

注: 当前方法, 只兼容了 **GBK** 编码

PrintfInfoManager 类中的方法

1: `beginGetPrinterInfoAsync(final GetAllPrinterInfoCallBack getAllPrinterInfoCallBack)`: 开始得到所有打印机的序列号、打印机型号代码、当前打印机的纸张类型、当前打印机的指令类型、当前的纸张状态。

参数说明

`GetAllPrinterInfoCallBack getAllPrinterInfoCallBack`: 当得到打印机信息完成之后, 会回调当前接口。当回调了当前接口之后, 可以在 `PrintfInfoManager` 的 `PrinterInfo` 中得到打印机的信息。

2: `getPrinterSerialNumber(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 同步得到当前打印机的序列号

`getPrinterModel(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 同步得到当前打印机的型号代码

`getPrinterPaperType(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 同步得到当前打印机的纸张类型

`getPrinterCmdType(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 同步得到当前打印机的指令类型

`getPrinterPaperState(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 同步得到当前打印机的纸张状态

上述方法, 都是同步方法, 会阻塞当前线程。若不想阻塞当前线程, 有如下五个异步方法

`getPrinterSerialNumberAsync(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 异步得到当前打印机的序列号

`getPrinterModelAsync(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 异步得到当前打印机的型号代码

`getPrinterPaperTypeAsync(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 异步得到当前打印机的纸张类型

`getPrinterCmdTypeAsync(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 异步得到当前打印机的指令类型

`getPrinterPaperStateAsync(final GetPrinterCmdCallBack getPrinterCmdCallBack)`: 异步得到当前打印机的纸张状态

参数统一说明:

GetPrinterCmdCallBack getPrinterCmdCallBack: 当前接口是回调得到打印机信息的接口，有如下几个方法。

```
void getComplete(); //完成
void getError(int error); //失败 error 是失败码
void getSuccess(); //成功
```

失败码说明：

```
error == 1 当前发送的数据，没有返回值
error == 2 当前发送的数据，在处理的时候，发送的异常
error == 3 蓝牙未连接
error == 4 当前的数据不符合规则
```

3: changePaperToContinuityPaperAsync(final ChangePaperTypeCallBack changePaperTypeCallBack): 异步切换打印机的纸张类型为连续纸。

changePaperToGAPPaperAsync(final ChangePaperTypeCallBack changePaperTypeCallBack): 异步切换打印机的纸张类型为间隙纸。

注：只有在标签小票打印机，当前方法才有效

参数说明：

ChangePaperTypeCallBack changePaperTypeCallBack: 切换结果，通过当前接口回调。当 **result(int result)** 回调 1 时，表示切换成功。回调 2 时，表示切换失败。

4: changeCMDToESCAsync(final ChangeCMDTypeCallBack changeCMDTypeCallBack): 异步切换成 ESC 指令

changeCMDToCPCLAsync(final ChangeCMDTypeCallBack changeCMDTypeCallBack): 异步切换成 CPCL 指令

changeCMDToTSPL(final ChangeCMDTypeCallBack changeCMDTypeCallBack): 异步切换成 TSPL 指令

注：只有在支持对应的指令的打印机上，切换方法才有效

参数说明：

ChangeCMDTypeCallBack changeCMDTypeCallBack: 切换结果，通过当前接口回调。当 **result(int result)** 回调 1 时，表示切换成功。回调 2 时，表示切换失败。

PrintfTSPLManager 类中的方法

1: printfLabelsAsync(List<TSPLPrinterModel> TSPLPrinterModels, final MultiplePrintfResultCallBack multiplePrintfResultCallBack): 打印多组标签，每一组对应一个 TSPLPrinterModel，每一个 TSPLPrinterModel 可以打印多张。

参数说明：

List<TSPLPrinterModel> TSPLPrinterModels:

TSPLPrinterModel 说明：

参数：

labelW: 标签的宽度 单位 mm

labelH: 标签的高度 单位 mm

printfDirection: 当前标签的打印方向。详见 TSPLPrinterModel.DirectionAngle

List<PrintfModel> printfModels: 当前的图片内容。

PrintfModel 说明：

坐标起点，在左上角

x: 当前图片的 x 坐标 单位 mm

y: 当前图片的 y 坐标 单位 mm

bitmapW: 图片的宽 单位 mm

bitmapH: 图片的高 单位 mm

rotate: 详见 PrintfModel.RotateAngle

rotatePoint: 当前旋转时，设计要坐标的旋转，当前参数，是坐标旋转的围绕的点。可传入 null。当 rotatePoint 是 null。默认为图片的中点。

threshold: 图片二值化的阈值。默认是 128

Int MM_TO_PX: mm 与 px 的转化比率

int printfNumber: 当前标签的打印数量

注：当前方法，核心思想是在需要打印的地方，传入相应的小图片即可。这样可以大大减少数据量。但是这样有时候，会出现半毫米的左右的误差。倘若要介绍误差，则使用全图打印。全图打印，也是调用当前的方法。只是 List<PrintfModel> printfModels 中只有一个 PrintfModel，这个 PrintfModel 铺满了整个标签即可。

此方法已经处理了打印方向、控件旋转等等坐标转化。所以，当调用当前方法时，开发者不需要关心打印方向之后的控件坐标转化，控件自身旋转的坐标转化。

2: printfLabelAsync(final TSPLPrinterModel tsplPrinterModel1

, final PrintfResultCallBack printfResultCallBack): 打印一组标签

参数说明：

TSPLPrinterModel tsplPrinterModel1: 如第一个方法

PrintfResultCallBack printfResultCallBack: 请看文档结尾的接口说明

ParameterUtil 参数类说明

1: setBitmapTowValuedType(int type, Context context) 设置图片的二值化算法

参数说明：

type: 当前的二值化算法。详细见 ParameterUtil.TowValuedType 类

2: setFloydSteinberyThreshold(int threshold, Context context)

参数说明：

threshold: FLOYD_STEINBERY 算法的阈值设置

MultiplePrintfResultCallBack 接口说明

当前接口，是 ESC 打印多组多张图片，TSPL 打印多组多张标签时的回调接口。

回调方法：

void printfIndexResult(int result, int group, int index);

第几组的第几张的打印结果

group: 第几组

index: 第几组的第几张

result: 打印结果

void printfCompleteResult(int result);

全部打印完成

result: 打印接口
void printfGroupCompleteResult(int group,int result);
第几组打印完成
group: 第几组
result: 打印结果
常量: (result 的值)
int MULTIPLE_PRINTF_SUCCESS = 1; 打印成功
int MULTIPLE_PRINTF_ERROR = 2; 打印错误
int MULTIPLE_PRINTF_INTERRUPT = 3; 打印中断

PrintfResultCallBack 接口说明

当前接口，是打印的接口回调。一般用于 TSPL 打印单张标签时的回调
回调方法

void callBack(int result); 打印结果
常量: (result 的值)
int PRINTF_RESULT_SUCCESS = 1; 打印成功
int PRINTF_RESULT_CMD_ERROR = 2; 打印错误
int PRINTF_RESULT_BLUETOOTH = 3; 蓝牙未连接
int PRINTF_RESULT_PARAMETER_ERROR = 4; 传入参数非法

2019/5/7
刘裕育编写

