# 打印开发指南

**文档修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **版本号** | **\*变化状态** | **简要说明** | **日期** | **变更人** | **批准日期** | **批准人** |
| **V1.0** | **c** | **初始版本** | **2021/9/11** | **LinJinXing** |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

\*变化状态：C = 创立，A = 增加，M = 修改，D = 删除

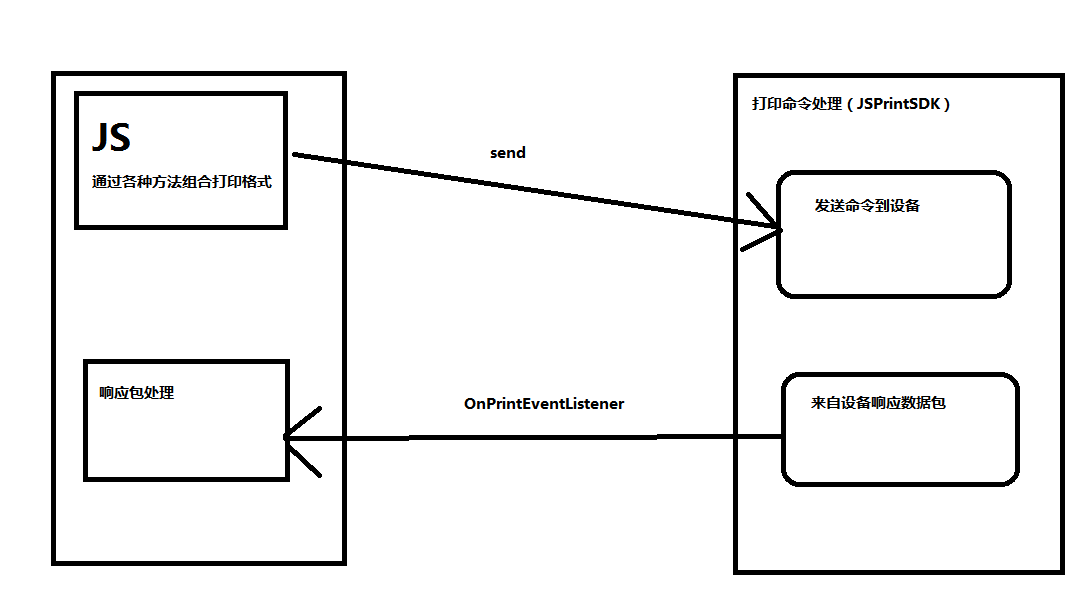
**文档审批记录**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **审批人** | **角色** | **审批日期** | **签字** | **备注** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# **1.简介**

为JS提供的打印sdk。

# **2.打印基本流程**



# 3.接口

## 3.1 获取打印机新旧模块

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getsupportprint () |
| 功能说明 | 获取打印机新旧模块 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：getsupportprint，  示例参考章节8.回调类示例 |

## 3.2 初始化打印机

|  |  |
| --- | --- |
| 函数接口 | **void** js\_initPrinter () |
| 功能说明 | 初始化打印机 |
| 参数说明 | 无 |
| 返回值 | 无 |

## 3.3 打印浓度

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printConcentration (String parm) |
| 功能说明 | 浓度 |
| 参数说明 | parm={  density：0-39  } |
| 返回值 | 无 |

## 3.4 文本加粗

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printTextBold (String parm) |
| 功能说明 | 强调模式，文本加粗 |
| 参数说明 | parm={  FONT\_BOLD：true打开，false关闭  } |
| 返回值 | 无 |

## 3.5 设置字体大小

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printFontSize (String parm) |
| 功能说明 | 设置字体大小 |
| 参数说明 | parm={fontSize：int }  参数参考章节7.常量类 |
| 返回值 | 无 |

## 3.6 获取字体大小

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getFontSize () |
| 功能说明 | 获取字体大小 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：getFontSize，  示例参考章节8.回调类示例 |

## 3.7 设置字体粗细

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setTextBold (String parm) |
| 功能说明 | 设置字体大小 |
| 参数说明 | parm={fontBold： true 加粗 false 不加粗} |
| 返回值 | 无 |

## 3.8 获取字体粗细

|  |  |
| --- | --- |
| 函数接口 | **void** js\_isFontBold () |
| 功能说明 | 获取字体粗细 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：isFontBold，  示例参考章节8.回调类示例 |

## 3.9 设置下划线

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setUnderLine (String parm) |
| 功能说明 | 设置下划线 |
| 参数说明 | parm={enable： true 加下划线 false 不加下划线} |
| 返回值 | 无 |

## 3.10 获取下划线状态

|  |  |
| --- | --- |
| 函数接口 | **void** js\_isUnderLine () |
| 功能说明 | 获取下划线状态 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：isUnderLine，  示例参考章节8.回调类示例 |

## 3.11 设置文本行间距

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setLineSpacing(String parm) |
| 功能说明 | 文本行间距 |
| 参数说明 | parm={  lineSpacing：1-n  } |
| 返回值 | 无 |

## 3.12 获取文本行间距

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getLineSpacing () |
| 功能说明 | 获取文本行间距 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：getLineSpacing，  示例参考章节8.回调类示例 |

## 3.13 打印空白行

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printLine(String parm) |
| 功能说明 | 打印空白行 |
| 参数说明 | parm={  line：int（行数）  } |
| 返回值 | 无 |

## 3.14 打印文本

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printText(String parm) |
| 功能说明 | 打印文本 |
| 参数说明 | parm={  text：String  } |
| 返回值 | 无 |

## 3.15 打印文本

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printText1(String parm) |
| 功能说明 | 打印文本 |
| 参数说明 | parm={  offset：int,  fontSize：int,  isBold：boolean,  isUnderLine：boolean,  text：String,  }  参数详情参考章节7常量类 |
| 返回值 | 无 |

## 3.16 打印条码

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printBarcode(String parm) |
| 功能说明 | 打印条码 |
| 参数说明 | parm={  text：String，  height：int，  barcodetype：int  }  参数参考章节7.常量类 |
| 返回值 | 无 |

## 3.17 打印条码

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printBarcode1(String parm) |
| 功能说明 | 打印条码 |
| 参数说明 | parm={  offset：int，  text：String，  height：int，  barcodetype：int ，  hriPosition：int  }  参数参考章节7.常量类 |
| 返回值 | 无 |

## 3.18 打印二维码

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printQR(String parm) |
| 功能说明 | 打印二维码 |
| 参数说明 | parm={  text：文本；  } |
| 返回值 | 无 |

## 3.19打印二维码

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printQR1(String parm) |
| 功能说明 | 打印二维码 |
| 参数说明 | parm={  text：String；  offset：int  height：int；（最大不超过384）  }  参数详情参考章节7常量类 |
| 返回值 | 无 |

## 3.20 开启黑标检测

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printEnableMark(String parm) |
| 功能说明 | 开启黑标检测 |
| 参数说明 | parm={  enable：true开启，false关闭  } |
| 返回值 | 无 |

## 3.21 检测黑标状态

|  |  |
| --- | --- |
| 函数接口 | **void** js\_isBlackLabel () |
| 功能说明 | 检测黑标的状态 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：isBlackLabel，  示例参考章节8.回调类示例 |

## 3.22开始走纸

|  |  |
| --- | --- |
| 函数接口 | **void** js\_start () |
| 功能说明 | 开始走纸 |
| 参数说明 | 无 |
| 返回值 | 无 |

## 3.23开始走纸

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printGoToNextMark() |
| 功能说明 | 走纸 |
| 参数说明 | 无 |
| 返回值 | 无 |

## 3.24开始走纸（设置走纸距离）

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printGoToNextMark1(String parm) |
| 功能说明 | 走纸（设置走纸距离） |
| 参数说明 | parm={  distance：int  } |
| 返回值 | 无 |

## 3.25 重置打印机

|  |  |
| --- | --- |
| 函数接口 | **void** js\_resetPrint() |
| 功能说明 | 重置打印机 |
| 参数说明 | 无 |
| 返回值 | 无 |

## 3.26 获取固件版本

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getVersion() |
| 功能说明 | 获取打印机固件版本 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：onVersion，  示例参考章节8.回调类示例 |

## 3.27 打印图片

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printBitmap(String parm) |
| 功能说明 | 小图片打印（384x360） |
| 参数说明 | parm={  base64：String  } |
| 返回值 | 无 |

## 3.28 打印图片

|  |  |
| --- | --- |
| 函数接口 | **void** js\_printBitmap1(String parm) |
| 功能说明 | 小图片打印（384x360） |
| 参数说明 | parm={  offset：int，  base64：String  } |
| 返回值 | 无 |

## 3.29设置走纸距离

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setFeedPaperSpace (String parm) |
| 功能说明 | 设置走纸距离 |
| 参数说明 | parm={  space：int  } |
| 返回值 | 无 |

## 3.30 获取走纸距离

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getFeedPaperSpace () |
| 功能说明 | 获取走纸距离 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：getFeedPaperSpace，  示例参考章节8.回调类示例 |

## 3.31设置回纸距离

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setUnwindPaperLen (String parm) |
| 功能说明 | 设置回纸距离 |
| 参数说明 | parm={  length：int  } |
| 返回值 | 无 |

## 3.32 获取回纸距离

|  |  |
| --- | --- |
| 函数接口 | **void** js\_getUnwindPaperLen () |
| 功能说明 | 获取回纸距离 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：getUnwindPaperLen，  示例参考章节8.回调类示例 |

## 3.33 打印状态回调

|  |  |
| --- | --- |
| 函数接口 | void onPrintStatus(state) |
| 功能说明 | 打印状态回调 |
| 参数说明 | 无 |
| 返回值 | state：0 打印成功  参数详情参考章节7.常量类 |

## 3.34获取版本号

|  |  |
| --- | --- |
| 函数接口 | void onVersion(s) |
| 功能说明 | 版本号 |
| 参数说明 | 无 |
| 返回值 | s：版本 |

## 3.35设置黑白反转

|  |  |
| --- | --- |
| 函数接口 | **void** js\_setReverse (String parm) |
| 功能说明 | 设置黑白反转 |
| 参数说明 | parm={  reverse：boolean  } |
| 返回值 | 无 |

## 3.36 获取黑白反转

|  |  |
| --- | --- |
| 函数接口 | **void** js\_isReverse () |
| 功能说明 | 获取黑白反转 |
| 参数说明 | 无 |
| 返回值 | 回调方法获取，方法名：isReverse，  示例参考章节8.回调类示例 |

## 3.37截屏打印

|  |  |
| --- | --- |
| 函数接口 | **void** js\_screenshot () |
| 功能说明 | 截屏打印 |
| 参数说明 | 无 |
| 返回值 | 无 |

# 4 注意事项

1、打印机目前不支持水平布局，如左侧二维码，右侧文本，此种格式需要生成图片进行打印,图片的最大宽度为384。

2、进行连续标签打印时，直接用循环发送会导致数据错乱，循环会调的很快上一张的内容还没打完，下一张又发送了，请打印成功回调后在打印下一张。

# 5.接口使用示例

var barcodeparam = {

text: '1234567890',

height: 80,

barcodetype: 73,

};

var mbarcodeparam = JSON.stringify(barcodeparam);

Android\_LcPrinter.js\_printBarcode(mbarcodeparam);

# 6.快速上手

1、初始化

Android\_LcPrinter.js\_initPrinter();

2、打印

function printText() {

var strText = document.getElementById("printtext").value;

var param = {

text: strText

};

var strParam = JSON.stringify(param);

Android\_LcPrinter.js\_printText(strParam);

if (black) {

Android\_LcPrinter.js\_printGoToNextMark();

} else {

Android\_LcPrinter.js\_start();

}

}

# 7.常量类

常量类：android.bld.print.configuration.PrintConfig

//IErrorCode：错误码

public class IErrorCode {

public static final int ERROR\_CMD = 19;//指令错误

public static final int ERROR\_DATA\_INPUT = 165;//输入参数错误

public static final int ERROR\_DATA\_INVALID = 18;//数据非法

public static final int ERROR\_DEV\_BMARK = 7;//黑标检测异常

public static final int ERROR\_DEV\_FEED = 5;//正在走纸

public static final int ERROR\_DEV\_IS\_BUSY = 1;//设备忙

public static final int ERROR\_DEV\_NOT\_OPEN = 16;//设备未打开

public static final int ERROR\_DEV\_NO\_BATTERY = 4;//低电

public static final int ERROR\_DEV\_PRINT = 6;//正在打印

public static final int ERROR\_GRAY\_INVALID = 20;//浓度非法

public static final int ERROR\_NO\_DATA = 17;//打印数据不能为空

public static final int ERROR\_NO\_ERROR = 0;//打印成功

public static final int ERROR\_PRINT\_BARCODE = 162;//打印条码错误

public static final int ERROR\_PRINT\_BITMAP = 161;//打印位图错误

public static final int ERROR\_PRINT\_BITMAP\_WIDTH\_OVERFLOW = 164;//打印位图宽度溢出

public static final int ERROR\_PRINT\_DATA\_MAC = 167;//Mac校验错误

public static final int ERROR\_PRINT\_HOT = 2;//高温

public static final int ERROR\_PRINT\_ILLEGAL\_ARGUMENT = 166;//参数错误

public static final int ERROR\_PRINT\_NOPAPER = 3;//缺纸

public static final int ERROR\_PRINT\_QRCODE = 163;//打印二维码错误

public static final int ERROR\_PRINT\_TEXT = 160;//打印文本错误

public static final int ERROR\_PRINT\_UNKNOWN = 255;//未知错误

public static final int ERROR\_RESULT\_EXIST = 168;//结果已存在

public static final int ERROR\_TIME\_OUT = 169;//超时

}

//StateType：状态类型

public class StateType {

public static final int CHECK\_ALL = 1;

public static final int CHECK\_BMASK = 7;

public static final int CHECK\_BUSY = 2;

public static final int CHECK\_FEED = 5;

public static final int CHECK\_PAPER = 4;

public static final int CHECK\_PRINT = 6;

public static final int CHECK\_TEMP = 3;

}

//HRIPosition：一维码文本位置

public static class HRIPosition {

public static final int POSITION\_ABOVE = 2;

public static final int POSITION\_BELOW = 3;

public static final int POSITION\_BOTH = 4;

public static final int POSITION\_NONE = 1;

}

//BarCodeType：条码类型

public static class BarCodeType {

public static final int TOP\_TYPE\_CODABAR = 71;

public static final int TOP\_TYPE\_CODE128 = 73;

public static final int TOP\_TYPE\_CODE39 = 69;

public static final int TOP\_TYPE\_CODE93 = 72;

public static final int TOP\_TYPE\_EAN13 = 67;

public static final int TOP\_TYPE\_EAN8 = 68;

public static final int TOP\_TYPE\_ITF = 70;

public static final int TOP\_TYPE\_UPCA = 65;

public static final int TOP\_TYPE\_UPCE = 66;

}

//ALIGN：对齐方式

public static class Align {

public static final int ALIGN\_CENTER = 2;//居中

public static final int ALIGN\_LEFT = 1;//居左

public static final int ALIGN\_RIGHT = 3;//居右

}

//FontSize：字符字体

public static class FontSize {

public static final int TOP\_FONT\_SIZE\_LARGE = 5;

public static final int TOP\_FONT\_SIZE\_MIDDLE = 3;

public static final int TOP\_FONT\_SIZE\_SMALL = 1;

public static final int TOP\_FONT\_SIZE\_SUPER = 7;

public static final int TOP\_FONT\_SIZE\_XLARGE = 6;

public static final int TOP\_FONT\_SIZE\_XMIDDLE = 4;

public static final int TOP\_FONT\_SIZE\_XSMALL = 2;

public static final int TOP\_FONT\_SIZE\_XSUPER = 8;

}

# 8.回调类示例

function onPrintStatus(msg) {

//0 打印成功 1 缺纸 2打印失败 3打印走纸到最后一张

if (msg === '0') {

alert("打印成功");

}

console.log(msg)

}

function onVersion(msg) {

alert("版本号：" + msg);

console.log(msg)

}