

Print development guide

Document revision record

Version number	*Change state	Brief description	date	Changed by	Approval date	Approver
V1.0	c	Initial version	2021/9/11	LinJinXing		

*Change status: C = created, a = added, M = modified, d = deleted

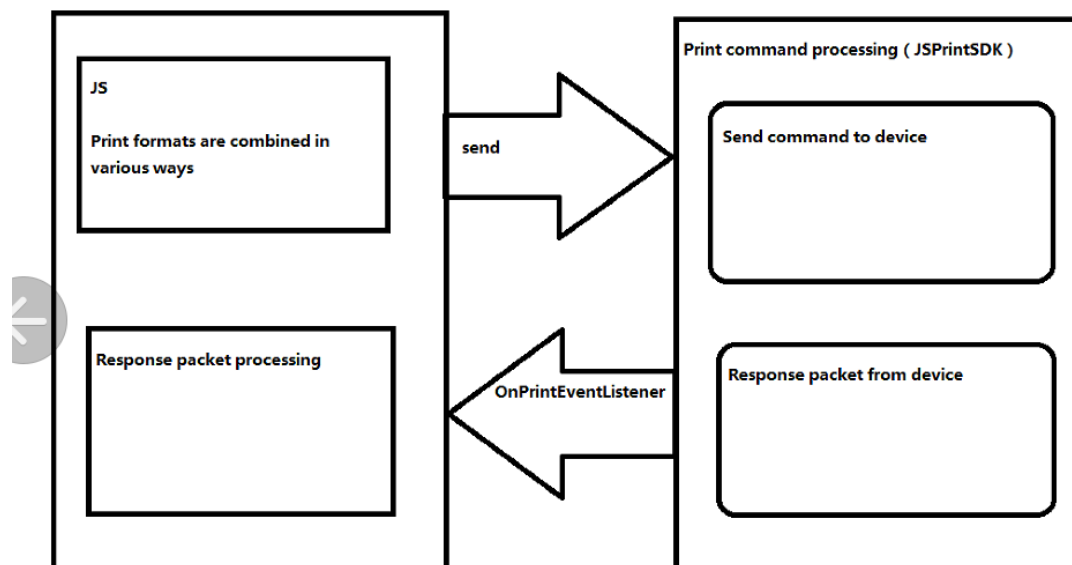
Document approval record

Serial number	Approver	role	Approval date	sign	remarks

1. brief introduction

Print SDK provided for JS.

2. Basic printing process



3. Interface

3.1 Get old and new printer modules

Function interface	void js_getsupportprint ()
Function description	Get old and new printer modules
Parameter description	nothing
Return value	Callback method, method name: getsupportprint, For example, refer to Chapter 8. Callback class example

3.2 Initialize printer

Function interface	void js_initPrinter ()
Function description	Initialize printer
Parameter description	nothing
Return value	nothing

3.3 Print density

Function interface	void js_printConcentration (String parm)
--------------------	---

Function description	concentration
Parameter description	parm={ density : 0-39 }
Return value	nothing

3.4 Text bold

Function interface	void js_printTextBold (String parm)
Function description	Emphasis mode, text bold
Parameter description	parm={ FONT_BOLD : true open , false close }
Return value	nothing

3.5 Set font size

Function interface	void js_printFontSize (String parm)
Function description	Set font size
Parameter description	parm={fontSize : int } Refer to Chapter 7. Constant class for parameters
Return value	nothing

3.6 Get font size

Function interface	void js_getFontSize ()
--------------------	-------------------------------

Function description	Get font size
Parameter description	nothing
Return value	Callback method, method name: getfontsize, For example, refer to Chapter 8. Callback class example

3.7 Set font weight

Function interface	void js_setTextBold (String parm)
Function description	Set font weight
Parameter description	parm={fontBold : true Bold /false No bold }
Return value	nothing

3.8 Get font weight

Function interface	void js_isFontBold ()
Function description	Get font weight
Parameter description	nothing
Return value	Callback method, method name: isfontbold, For example, refer to Chapter 8. Callback class example

3.9 Set underline

Function interface	void js_setUnderLine (String parm)
Function	Set underline

description	
Parameter description	parm={enable : true Underline /false Do not underline }
Return value	nothing

3.10 Get underline status

Function interface	void js_isUnderLine ()
Function description	Get underline status
Parameter description	nothing
Return value	Callback method, method name: isunderline, For example, refer to Chapter 8. Callback class example

3.11 Set text line spacing

Function interface	void js_setLineSpacing(String parm)
Function description	Set text line spacing
Parameter description	parm={ lineSpacing : 1-n }
Return value	nothing

3.12 Get text line spacing

Function interface	void js_getLineSpacing ()
Function description	Get text line spacing

Parameter description	nothing
Return value	Callback method, method name: getlinespacing, For example, refer to Chapter 8. Callback class example

3.13 Print blank lines

Function interface	void js_printLine(String parm)
Function description	Print blank lines
Parameter description	parm={ line : int (Number of rows) }
Return value	nothing

3.14 Print text

Function interface	void js_printText(String parm)
Function description	Print text
Parameter description	parm={ text : String }
Return value	nothing

3.15 Print text

Function interface	void js_printText1(String parm)
Function description	Print text

Parameter description	<pre> parm={ offset : int, fontSize : int, isBold : boolean, isUnderLine : boolean, text : String, } </pre> <p>Refer to Chapter 7 constant class for parameter details</p>
Return value	nothing

3.16 Print barcode

Function interface	void js_printBarcode(String parm)
Function description	Print barcode
Parameter description	<pre> parm={ text : String , height : int , barcodetype : int } </pre> <p>Refer to Chapter 7 constant class for parameter details</p>
Return value	nothing

3.17 Print barcode

Function interface	void js_printBarcode1(String parm)
Function description	Print barcode
Parameter description	<pre> parm={ offset : int , text : String , height : int , </pre>

	<pre> barcodetype : int , hriPosition : int } Refer to Chapter 7 constant class for parameter details </pre>
Return value	nothing

3.18 Print QR code

Function interface	void js_printQR(String parm)
Function description	Print QR code
Parameter description	<pre> parm={ text : String } </pre>
Return value	nothing

3.19 Print QR code

Function interface	void js_printQR1(String parm)
Function description	Print QR code
Parameter description	<pre> parm={ text : String ; offset : int height : int ; (Up to 384) } Refer to Chapter 7 constant class for parameter details </pre>
Return value	nothing

3.20 Turn on black mark detection

Function interface	void js_printEnableMark(String parm)
Function description	Turn on black mark detection
Parameter description	parm={ enable : true open , false close }
Return value	nothing

3.21 Detect black mark status

Function interface	void js_isBlackLabel ()
Function description	Detect black mark status
Parameter description	nothing
Return value	Callback method, method name: isblacklabel, For example, refer to Chapter 8. Callback class example

3.22 Start moving paper

Function interface	void js_start ()
Function description	Start moving paper
Parameter description	nothing
Return value	nothing

3.23 Start moving paper

Function interface	void js_printGoToNextMark()
Function description	Start moving paper
Parameter description	nothing
Return value	nothing

3.24 Start paper feeding (set paper feeding distance)

Function interface	void js_printGoToNextMark1(String parm)
Function description	Start paper feeding (set paper feeding distance)
Parameter description	parm={ distance : int }
Return value	nothing

3.25 Reset printer

Function interface	void js_resetPrint()
Function description	Reset printer
Parameter description	nothing
Return value	nothing

3.26 Get firmware version

Function interface	void js_getVersion()
Function description	Get firmware version
Parameter description	nothing
Return value	Callback method, method name: onversion, For example, refer to Chapter 8. Callback class example

3.27 print pictures

Function interface	void js_printBitmap(String parm)
Function description	print pictures (384x360)
Parameter description	parm={ base64 : String }
Return value	nothing

3.28 print pictures

Function interface	void js_printBitmap1(String parm)
Function description	print pictures (384x360)
Parameter description	parm={ offset : int , base64 : String }
Return value	nothing

3.29 Set paper feed distance

Function interface	void js_setFeedPaperSpace (String parm)
Function description	Set paper feed distance
Parameter description	parm={ space : int }
Return value	nothing

3.30 Get paper distance

Function interface	void js_getFeedPaperSpace ()
Function description	Get paper distance
Parameter description	nothing
Return value	Callback method, method name: getfeedpaperspace, For example, refer to Chapter 8. Callback class example

3.31 Set paper return distance

Function interface	void js_setUnwindPaperLen (String parm)
Function description	Set paper return distance
Parameter description	parm={ length : int }
Return value	nothing

3.32 Get paper return distance

Function interface	void js_getUnwindPaperLen ()
Function description	Get paper return distance
Parameter description	nothing
Return value	Callback method, method name: getunwindpaperlen, For example, refer to Chapter 8. Callback class example

3.33 Print status callback

Function interface	void onPrintStatus(state)
Function description	Print status callback
Parameter description	nothing
Return value	State: 0 printing succeeded Refer to Chapter 7. Constant class for parameter details

3.34 Get version number

Function interface	void onVersion(s)
Function description	Get version number
Parameter description	nothing
Return value	s : edition

3.35 Set black and white reversal

Function interface	void js_setReverse (String parm)
Function description	Set black and white reversal
Parameter description	parm={ reverse : boolean }
Return value	nothing

3.36 Get black and white inversion

Function interface	void js_isReverse ()
Function description	Get black and white inversion
Parameter description	nothing
Return value	Callback method, method name: isreverse, For example, refer to Chapter 8. Callback class example

3.37 Screen capture printing

Function interface	void js_screenshot ()
Function description	Screen capture printing
Parameter description	nothing
Return value	nothing

4 matters needing attention

- 1、 At present, the printer does not support horizontal layout, such as QR code on the left and text on the right. This format needs to generate pictures for printing. The maximum width of the pictures is 384.
- 2、 During continuous label printing, direct circular transmission will lead to data disorder. The cycle will adjust quickly. The content of the previous one has not been typed, and the next one is sent again. Please print the next one after successful printing and callback.

5. Interface usage example

```
var barcodeparam = {  
    text: '1234567890',  
    height: 80,  
    barcodetype: 73,  
};  
var mbarcodeparam = JSON.stringify(barcodeparam);  
Android_LcPrinter.js_printBarcode(mbarcodeparam);
```


6. Get started quickly

1、 initialization

```
Android_LcPrinter.js_initPrinter();
```

2、 Print

```
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. Constant class

Constant class : android.bld.print.configuration.PrintConfig

//IErrorCode : Error code

```

public class IErrorCode {
    public static final int ERROR_CMD = 19;// Instruction error
    public static final int ERROR_DATA_INPUT = 165;// Input parameter error
    public static final int ERROR_DATA_INVALID = 18;// data is invalid
    public static final int ERROR_DEV_BMARK = 7;// Abnormal black mark
detection
    public static final int ERROR_DEV_FEED = 5;// Moving paper
    public static final int ERROR_DEV_IS_BUSY = 1;// Device busy
    public static final int ERROR_DEV_NOT_OPEN = 16;// The device is not
turned on
    public static final int ERROR_DEV_NO_BATTERY = 4;// Low electricity
    public static final int ERROR_DEV_PRINT = 6;// print now
    public static final int ERROR_GRAY_INVALID = 20;// Illegal concentration
    public static final int ERROR_NO_DATA = 17;// Print data cannot be empty
    public static final int ERROR_NO_ERROR = 0;// Printing succeeded
    public static final int ERROR_PRINT_BARCODE = 162;// Error printing
barcode
    public static final int ERROR_PRINT_BITMAP = 161;// Error printing bitmap
    public static final int ERROR_PRINT_BITMAP_WIDTH_OVERFLOW = 164;//
Print bitmap width overflow
    public static final int ERROR_PRINT_DATA_MAC = 167;// Mac check error
    public static final int ERROR_PRINT_HOT = 2;// high temperature
    public static final int ERROR_PRINT_ILLEGAL_ARGUMENT = 166;//
Parameter error
    public static final int ERROR_PRINT_NOPAPER = 3;// Paper shortage
    public static final int ERROR_PRINT_QRCODE = 163;// Error printing QR code
    public static final int ERROR_PRINT_TEXT = 160;// Print text error
    public static final int ERROR_PRINT_UNKNOWN = 255;// unknown error
    public static final int ERROR_RESULT_EXIST = 168;// Result already exists
    public static final int ERROR_TIME_OUT = 169;// overtime
}

//StateType : Status type

```

```

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 : One dimensional code text position
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 : Barcode Type
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 : Alignment

public static class Align {
    public static final int ALIGN_CENTER = 2;// Center
    public static final int ALIGN_LEFT = 1;// be at the left side
    public static final int ALIGN_RIGHT = 3;// be at the right
}

//FontSize : Character font

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. Callback class example

```
function onPrintStatus(msg) {
    //0 printing succeeded 1 paper shortage 2 printing failed 3 printing paper to
    the last one
    if (msg === '0') {
        alert("Printing succeeded ");
    }
}
```

```
    console.log(msg)
}
```

```
function onVersion(msg) {
    alert("Version number : " + msg);
    console.log(msg)
}
```