

# **Android-SDK Development Manual**

(V1.3)

**Document information**

File Name	Android-SDK Printer Developing Manual			
Author	Yang Xuezhi, Zhang Liang			
File name	Android-SDK Printer Developing Manual.doc			
Summary	This file is for Android-SDK 5.3 version. This version is modified on the basis of SDK5.2 version. It optimizes the way to query the status of printer and the completion process of printing, USB interface communication, images printing, Bitmap download to printers function and so on. And regulate the modulation logs for above funtions. Modify some bugs of 5.2 version.			
(REVISIONHISTORY)				
Version	Date	Modifier	Reviewers	Summary
1.0	2016-06-22	Fu Daohui	Qin Zhengfeng	Initially created this File
1.1	2016-12-09	Yang Xuezhi		1. Add TSPL instruction label printing method 2. Add the black mark positioning function 3. Add the color bitmap convert to the grayscale print function 4. Modify instruction of sendBytesData method in the document returned value to add a -3 5. Serial communication add need to call the specified .so instructions. 6. Add all methods to the hyperlink.

1.2	2017-07-26	Fu Daohui		<ol style="list-style-type: none"><li>1. Modify the PDF print in the SDK</li><li>2. Optimize the serial failure problem on non-root device</li><li>3. Add printed language instruction</li><li>4. Optimize the method of printing pictures</li><li>5. In the SDK, I added the print line, dotted line method</li><li>6. Optimize other known bugs</li><li>7. Added a quick reading guide on the document</li></ol>
-----	------------	-----------	--	--

1.3	2017-12-24	Yang Xuezhi		<p>1.Optimize the way to query the status of printer. Adjust the query logic order to ensure the stability. And specify the abnormal return to read abnormal and write abnormal.</p> <p>2.Optimizes the method of querying the completion process of printing. Remove the parameter which display the information of query results.</p> <p>3. Overload the method of printing images with ESC/POS instruction. This method can directly pass color Bitmap object, and can modify the threshold value of image's binarization.</p> <p>4.Modify the method of how to download bitmap to the printer with TSPL instruction, which enable normal bitmapdownload to printer cache.</p> <p>5.Modify the way of reading and writing data under USB communication, improve printer communication speed. Modify the size of single packet date, and delay between the single packet date. Also modify the returned value of read date in order to ensure the reads of returned values are the bytes which are sent successfully. 6. Modify the bugs which limit parameters of bitmap printing under TSPL command.</p> <p>7. Regulate the modulation logs for modifying the interfaces.</p>
-----	------------	----------------	--	---

## Content

<b>1.Introduction</b>	7
<b>2. Nouns Explanation</b>	7
2.1 Nouns Explanation	7
2.2 Relevant Knowledge	8
2.3 This document speed guide	8
<b>3. Related Class Explanation</b>	3
3.1 Printer Core Class (PrinterInstance)	3
3.1.1 Get Printer Instance Class	4
3.1.2 Printer Communication Related Method	5
3.1.3 Setting Printer Method	7
3.1.4 ESC/POS Instructions for printing related methods	8
3.1.5 CPCL Command Label Printing Related Method	11
3.1.6 TSPL Command Label Printing Related Method	17
3.1.7 Method to control printer hardware	29
3.1.8 Get the current state of the printer	30
3.1.9 The call examples of connecting printers	32
3.2 Barcode Printing Class	33
3.2.1 constructor function	34
3.2.2 Barcode Printing call example	35
3.3 Table Form Printing Class	35
3.3.1 Create Table examples	35
3.3.2 Add One Row Data	36
3.3.3 Setting the column data alignment in the Table	37

---

3.3.4 Form printing call example.....	37
3.4 Canvas Printing Class.....	37
3.4.1 Initialization canvas.....	45
3.4.2 Set Font Property.....	45
3.4.3 Canvas content related method.....	39
3.4.4 Call sample for canvas printing.....	40
3.5 BitmapConvertor class.....	41
3.5.1 Transfer colorful image to Black&White binary bitmap.....	41
3.6 Correlation Class of Printing PDF file.....	42
3.6.1 Calling example for PDF printing .....	42
3.7 CodePage Printer multilingual code page Print class.....	43
<b>4. Appendix.....</b>	<b>44</b>
4.1 Summarize of CODE128.....	44
4.2 Character Set.....	46

## 1.Introduction

Android-SDK5.3 is an Android function interface based on our printer development. This version is modified on the basis of SDK5.2 version. It optimizes the way to query the status of printer and the completion process of printing, print bitmap with ESC/POS command, download bitmap to the printer with TSPL instruction, USB communication and so on, also mudify other known bugs. Note that this version no longer supports SDK4.0 and earlier versions, compatible with 5.0 and above. If there are any software problems with SDK4.0 and earlier versions, please consult our company pre-sales.This SDK supports bluetooth, serial port, WIFI, and USB communication.The related class illustrations are below:

Class Name	Discription
PrinterInstance	Printer Core Class
Barcode	Barcode Printing Class
Table	Table Printing Class
CanvasPrint	Canvas Printing Class
BitmapConvertor	Colorful Bitmap Converting Class
PdfContext,CodecDocument,CodecPage	PDF File Printing Related Class
PrinterConstants	Related Constants Class

## 2. Nouns Explanation

### 2.1 Nouns Explanation

- Printing Width: The maximum horizontal printing range that printer supports, which is decided by printer itself. For example, as for 80mm width printer, the printing width is 72mm(576 dots); while as for 58mm printer, the printing width is 48mm(384 dots).
- Printing Area: Printing area can be set by commands, and it must be less than or equal to printing width;

- Row Height: The height of character row. Row Height = Character Height + Line Space;
- Black Mark Paper: Black mark is black patch pre-printed on the paper. Users can use it to locate printing position;
- Vertical or Horizontal Moving Unit: The default one moving unit is one printing dot. The horizontal distance is 1/8mm and the vertical distance is 1/8mm;

## 2.2 Relevant Knowledge

- Western Characters Printing: Western characters we mentioned including ASCII character and CodePage. The ASCII character range is 0x20~0x7F, the Codepage range is 0x80~0xFF. Western languages(such as German and Spanish) have their own single-byte codepages. Because there is overlapping section for Codepage codes and Chinese characters codes, pls print Codepage under Western Characters Printing mode. Common Western Characters Font are: Font A: 12x24(dots); Font B: 9x17(dots);
- Chinese Printing: The Chinese Characters we mentioned including simplified Chinese and traditional Chinese. The common simplified Chinese characters set we use are GB2312 and GB18030, and the common traditional Chinese characters set we use is BIG5. Common Chinese Characters Font: 24x24 (dots).
- Twice Height Printing: In this printing mode, the height of character is twice of normal height of character;
- Twice Width Printing: In this printing mode, the width of character is twice of normal width of character;

## 2.3 This document speed guide

- Because the printer model is not the same, the instruction is not the same, customer development personnel familiar with the printer have different degree, in order to let the customer developers can more quickly locate to call API method, the following text description will help you.
- No matter what you do, as long as you need under the android via bluetooth, USB, WIFI (or cable), a serial port connection to print, all can read 3.1.1 and 3.1.2 section query entity class for printers, open communication port, reading and writing data, close com API, such as additional 3.1.9 provides bluetooth for example, to initialize the printer, connect printers, print data, close the API calls such as application examples.
- If you want to print a small ticket on ordinary heat sensitive paper, you can read the contents of section 3.1.4. This section API provides printing text, printing one-dimensional code, qr code, printing images and other relevant methods.



- If you want to print the label, and your printer supports CPCL instructions, you can read 3.1.5. This section provides methods for printing lines, rectangles, text, and areas to print text, bar codes, etc.
- If you want to print the tag, and your printer supports TSPL instructions, you can read the 3.1.6 section. This section provides a similar approach to 3.1.5, except in a different way.
- If you have the need to query printer status, read the 3.1.8 section. Another 3.6 section provides a program for PDF file printing. In fact, you may want to print more rich format data, and the canvas printing method may be suitable for you, please read 3.4. Cut paper, control the buzzer, look for black label, read 3.1.7 section.

Note: the printer cannot print the data of the webpage directly, nor can it identify the rich and colorful format in the webpage. The average customer can convert the web data to bitmap, and print out the image by calling the `printImage` (`bitmap bitmap`, `PAlign alignType`, `int left`, `Boolean isCompressed`).

## 3. Related Class Explanation

### 3.1. Printer Core Class (PrinterInstance)

Class Explanation: It is the core class for printing, which provide getting printer instance method and printer communication related method, printing data related method, setting printer related method, printing label related method, getting printer status related method and control printer hardware related method.

<a href="#">Get Printer Instance Class</a>
<a href="#">Printer Communication Related Method</a>
<a href="#">Setting Printer Method</a>
<a href="#">Printing Content Related Method</a>
<a href="#">CPCL Command Label Printing Method</a>
<a href="#">TSPL Command Label Printing Method</a>
<a href="#">Control Printer Hardware Related Method</a>
<a href="#">Get Printer Status Related Method</a>
<a href="#">Connect Printing Invocation Example</a>

### 3.1.1 Get Printer Instance Class

Method name	Description
getPrinterInstance	Getting Bluetooth Printer Instance Class
getPrinterInstance	Getting USB Printer Instance Class
getPrinterInstance	Getting WiFi Printer Instance Class
getPrinterInstance	Getting Serial Printer Instance Class

- Getting Bluetooth Printer Instance Class object
- Parameter:
  - bthDevice BluetoothDevice Device Object
  - handler Handler object can be used to received message of connecting successfully or not
  - Returned Value: PrinterInstance Instance Class
- Getting USB Printer Instance Class
  - PrinterInstance static synchronized getPrinterInstance (Context context,UsbDevice usbDevice, Handler handler)
  - Parameter:
    - context context object
    - usbDevice UsbDevice USB device object
    - handler Handler object can be used to received message of connecting successfully or not
    - Returned Value: PrinterInstance Instance Class
- Getting WiFi Printer Instance Class
  - PrinterInstance static synchronized getPrinterInstance (String ipAddress,int portNumber, Handler handler)
  - PrinterInstance static synchronized getPrinterInstance(String ipAddress,int portNumber, Handler handler)
  - Parameter:
    - ipAddress ip address character string of dotted decimal printer, such as "192.168.0.100"
    - portNumber port number of printer, which is 9100 by default
    - handler Handler object can be used to received message of connecting successfully or not
    - Returned Value: PrinterInstance Instance Class
- Getting Serial Printer Instance Class

PrinterInstance static synchronized getInstance(File device, int baudrate, in flags, Handler handler)

Parameter

device Serial device object

Baudrate Baudrate of serial port, specific baudrate are subject to self test of printer.

flags The mark position of serial port. Default is 0

handler Handler object can be used to received message of connecting successfully or not

Returned Value:PrinterInstance Instance Class

Note: The handler parameter of all above construction methods are used to received connect states of printer, whose corresponding states are as below:

Note:

✎If use serial port, .so library file and PrintDemo/ libs/ armeabi /libserial\_port.so files must be added in demo.

✎All parameter of construct method are use to receive the connection method of printer, which are below:

PrinterConstants.Connect.SUCCESS; Connecting successfully

PrinterConstants.Connect.FAILED; Connecting unsuccessfully

PrinterConstants.Connect.CLOSED; Close connecting

### 3.1.2 Printer Communication Related Method

Method Name	Description
<a href="#">OpenConnection</a>	Connecting To Printer
<a href="#">closeConnection</a>	Close the Connection To Printer
<a href="#">sendBytesData</a>	Sending Hexadecimal Data To Printer
<a href="#">read</a>	Read data returned from printer

#### ➤ Connecting To Printer

Method Name:public boolean openConnection()

Method Description: Connecting printer, handler object can be used to received message of connecting successful or not

Parameter: None

Returned Value: true      Connecting successfully  
false      Connecting unsuccessfully

➤ Close the Connection To Printer

Method Name: public void closeConnection()

Method Description: Close the connection to printer, and handler object can be used to received message of connection closed

Parameter: None

Android-SDK Printer Developing Manual

Returned Value: None

➤ Sending Hexadecimal Data To Printer

Method Name: public int sendBytesData(byte[] srcData)

Method Description: Sending hexadecimal data to printer. If the SDK haven't provided function wanted, please refer to instruction manual to use this method to send instruction to printer directly.

For example: If the command for setting printing position in the center checked on the manual is: 0x1B,0x61,0x01, the code can be written as below to achieve setting printing position in the center.

```
byte[] command = new byte[3];
```

```
command[0] = 0x1B;
```

```
command[1] = 0x61;
```

```
command[2] = 0x01;
```

```
sendByteData (command);
```

Parameter: srcData byte array

Returned Value: Bytes numbers which are sent to printer successfully; otherwise the returned value is negative. -1 means no initialize printing, -2 means srcData is blank or srcData doesn't have any data.

➤ Read data returned from printer

Method Name: public int read(byte[] buffer)

Method Description: Read data returned from printer

Parameter: Used to receive the array which read bytes

Returned Value: Bytes numbers which are read successfully; otherwise the returned value is negative. -1 means no initialize printing, -2 means buffer is blank or the length of buffer array is 0.

### 3.1.3 Setting Printer Method

Note: the setting method is invalid in the CPCL commands program of label printer.

Method Name	Description
<a href="#">initPrinter</a>	Initialize Printer
<a href="#">setFont</a>	Setting Printer Font
<a href="#">setPrinter</a>	Printer Printing Setting
<a href="#">setLeftMargin</a>	Setting dots numbers which are away from left margin

➤ Initialize Printer

Android-SDK Printer Developing Manual

Method Name: public void initPrinter()

Method Description: Initialize Printer

Parameter: None

Returned Value: None

➤ Setting Printer Font

Method Name: public void setFont(int mCharacterType, int mWidth, int mHeight, int mBold, int mUnderline)

Method Description: Setting printer printing font, including setting boldface, twice height, twice width, underline, standard ASCII Font A(12x24), compressed ASCII Font B(9x17)

Parameter:

mCharacterType: 0 means 12x24 font size, 1 means 9x16 font size, The set font type is temporary effective. Printer do not save this setting, means invalid when printer reset. Need to send 1 do manual setting if printed 9\*16 font.

mWidth: twice width, the range is 0~7

mHeight: twice height, the range is 0~7

mBold: 0 means not bold, 1 means boldface

mUnderline: 0 means no underline, 1 means having underline

Returned Value: None

➤ Printer Printing Setting

Method Name: public void setPrinter(int command, int value)

Android-SDK Printer Developing Manual

Method Description: Setting printer, including printing and feeding paper for n dot line; printing and feeding paper for n character line; setting character align mode, align left, center, align right.

Parameter: command

PrinterConstants.Command.PRINT\_AND\_WAKE\_PAPER\_BY\_LNCH

Printing and feeding value dot line

PrinterConstants.Command.PRINT\_AND\_WAKE\_PAPER\_BY\_LINE

Printing and feeding value character line

PrinterConstants.Command.ALIGN Setting the position of printing content,

Value can be used to appoint specific position of setting, which can be:

PrinterConstants.Command.ALIGN\_LEFT;

PrinterConstants.Command.ALIGN\_CENTER;

PrinterConstants.Command.ALIGN\_RIGHT

### **E.g.1: Set the print text to center**

```
mPrinter.setPrinter(Command.ALIGN, Command.ALIGN_CENTER);
```

```
mPrinter.printText("hello world!\n");
```

### **E.g.2: Set the print 1D Bar code center**

```
mPrinter.setPrinter(Command.ALIGN, Command.ALIGN_CENTER);
```

```
Barcode barcode1 = new Barcode(BarcodeType.CODE128, 2, 150, 2,"123456");
```

```
mPrinter.printBarCode(barcode1);
```

### **E.g.3: Set the print 2D Bar code center**

```
PrinterInstance.mPrinter.setPrinter(Command.ALIGN, Command.ALIGN_CENTER);
```

```
Barcode barcode2 = new Barcode(BarcodeType.QRCODE, 2, 3, 6,"123456");
```

```
PrinterInstance.mPrinter.printBarCode(barcode2);
```

Note:Above for the print image printImage method is invalid

➤ Setting dots numbers which are away from left margin

Method Name:public void setLeftMargin(in inches)

Method Description: Setting inches number which are away from left margin

Parameter:inches inches numbers away from left margin

Returned Value: None

### **E.g. 1: setLeftMargin8 inches**

```
mPrinter.setLeftMargin(8);
```

```
mPrinter.printText("hello world!\n");
```

**This method is valid for printing 1D Bar code, 2D Bar code, and is not valid for print image printImage**

## **3.1.4 ESC/POS instruction print related methods**

Note: the setting method is unavailable in the CPCL commands program of label printer.

Method Name	Description
<a href="#">printText</a>	Printing Characters
<a href="#">printImage</a>	Print Black and white binary picture
<a href="#">printImage</a>	Print colorful image
<a href="#">printColorImg2Gray</a>	Print Grayscale color image
<a href="#">printTable</a>	Print Table
<a href="#">printBarCode</a>	Print Barcode
<a href="#">printhorizontalLine</a>	Print line , dotted line

➤ Print text

Method Name: `public void printText(String content)`

Method Description: Print text

Parameter: content Text content needs to be printed

Returned Value: None

Note: Printing data will buffer until data is enough for a line or add "\n" behind content, that is (`printText("hello world\n\n")`), then can print content data.

➤ Print Black and white binary picture

Method name: `public void printImage(Bitmap bitmap, PAlign alignType, int left, boolean isCompressed)`

Method Description: From the left position on the left margin begin to print picture, and please note the printer can only print binary image. If colored bitmap sent, printer will still print black and white binary bitmap, and will cause the serious picture distortion. Bitmap Convertor class provides the method to transfer the colored picture into binary picture, please refer to BitmapConvertor class introduction.

Parameter: bitmap monochrome bitmap will be printed, whose color depth is 1.

alignType: the position of printing picture, PAlign.START means the position is on the left, PAlign.CENTER means the position is on the center, PAlign.END means the position is on the right, PAlign.NONE means not appoint the printing position. Please note that when the parameter is PAlign. NONE, users can adjust the distance of pictures to left margin according to setting left parameter, other parameter will be invalid when sent into left. When setting the printing position on the center, on the right and on the left, please assign the printing width of printer before use above method, that is Printer Constants paper width=384(for 58mm paper, 576 for 80mm paper, 724 for 100mm paper).

Left: When the alignType is PAlign.NONE, sending parameter into Left can refined adjust the position of printing pictures away from the left margin. When the alignType is not PAlign.NONE, the parameter is invalid and left must be multiple of 8.

isCompressed: It means whether using compression algorithm or not.

Shows true means using compression algorithm, shows false means not Android-SDK Printer Developing Manual using compression algorithm. Using compression algorithm can improve the printing rate when printer print pictures. Please note using compress mode needs to use customized printer, since normal printer don't support this kind of function.

### **E.g.1: The following description sets the print picture center**

Print paper width: 58mm, horizontal effective print points 384 points

```
PrinterConstants.paperWidth = 384;
```

```
mPrinter.printImage(bmp, PAlign.Command.ALIGN_CENTER, 0, false);
```

### **E.g.2: The following description prints the picture from the left side 24 points**

Print paper width: 58mm, horizontal effective print points 384 points

```
PrinterConstants.paperWidth = 384;
```

```
mPrinter.printImage(bmp, PAlign.NONE, 24, false);
```

Returned value: Null

#### ➤ Print colorful binary picture

Method Name: public void printImage(Bitmap bitmap, PAlign alignType,int left, int level)

Method Description: From the left position on the left margin begin to print picture, and please note it's a way of overload the Black and white binary picture printing. It can directly transformate uploaded colourful bitmap to binary, and print. What's more, it can open the threshold vale of binary picture.

Parameter: bitmap which need to printed

alignType: the position of printing picture,PAlign.START means the position is on the left, PAlign.CENTER means the position is on the center,PAlign.END means the position is on the right, PAlign.NONE means not appoint the printing position. Please note that when the parameter is PAlign.NONE, users can adjust the distance of pictures to left margin according to setting left parameter, other parameter will be null when sent into left. When setting the printing position on the center, on the right or on the left, please assign the printing width of printer before use above method, that is Printer Constants paper width=384(for 58mm paper, 576 for 80mm paper, 724 for 100mm paper)



Left : When the align Type is PAlign.NONE, sending parameter into left can refined adjust the position of printing pictures away from the left margin. When the align type is not PAlign. NONE, the parameter is null and left must be multiple of 8.

Level: The threshold value of binary picture. Range: 0~255. Default is 128. Note: Although the theoretical threshold value can be 255, but when the threshold value reaches or closes to 255, the picture might transform to all black bitmap, and also will damage the printer. So the suggestion data is 128, 160, 192.

### **E.g.1: The following description sets the print picture center**

Print paper width: 58mm, horizontal effective print points 384 points

```
PrinterConstants.paperWidth = 384;
```

```
mPrinter.printImage(bmp, PAlign.Command.ALIGN_CENTER, 0, 128);
```

### **E.g.2: The following description prints the picture from the left side 24 points**

Print paper width: 58mm, horizontal effective print points 384 points

```
PrinterConstants.paperWidth = 384;
```

```
mPrinter.printImage((bmp, PAlign.NONE, 24, 128);
```

Returned value: Null

#### ➤ Print Grayscale color image

Method Name: public void printColorImg2Gray (Bitmap bitmap, PAlign alignType, int left, boolean isCompressed)

Method Description: From the left position on the left margin begin to print the grayscale color image

Parameter: bitmap color bitmap will be printed

alignType: the position of printing picture, PAlign.START means the position is on the left, PAlign.CENTER means the position is on the center, PAlign.END means the position is on the right, PAlign.NONE means not appoint the printing position. Please note that when the parameter is PAlign.NONE, users can adjust the distance of pictures to left margin according to setting left parameter, other parameter will be null when sent into left. When setting the printing position on the center, on the right and on the left, please assign the printing width of printer before use above method, that is Printer Constants paper width=384(for 58mm paper, 576 for 80mm paper, 724 for 100mm paper)

Left : When the align Type is PAign.NONE,sending parameter into left can refined adjust the position of printing pictures away from the left margin. When the align type is not PAign. NONE, the parameter is null and left must be multiple of 8.

IsCompressed: It means whether using compression algorithm or not.

True means using compression algorithm, False means not using. Using compression algorithm can improve the printing rate when printer print pictures.

Note: using compress mode needs to use customized printer, since normal printer don't support this kind function .

Returned Value: None

➤ Printing Table

Method Explanation: public void printTable(Table table)

Method Description :Printing table

Parameter: instantiation object of table .Please refer to Table Class 3.34

Returned Value :None

➤ Printing Barcode

Method Explanation:public void printBarCode(Barcode barcode)

Method Description :print 1D , 2D barcode

Parameter: instantiation object of barcode ,Please refer to Barcode Class 3..2.2 for specific instantiation example .

Returned Value: None

➤ Printing Straight line ,dotted line

Method Explanation:public int printhorizontalLine(int lineLength, int lineWith, boolean isSolidline, int interval)

➤ Method Description :Printing Straight line ,dotted line

Parameter:lineLength      Length of Straight line

                 lineWith      Width of Straight line

                 isSolidline      true,Straight line, false, dotted line

                 interval      Dotted line spacing, optional parameters are 4,8. The larger the value, the larger the spacing

Returned Value: None

### 3.1.5 CPCL Command Label Printing Related Method

Note: below methods are apply for label printer ,which support standard CPCL command set.

Method	Description
<a href="#">pageSetup</a>	Setting the page width and page height of printing area
<a href="#">drawLine</a>	to print straight line
<a href="#">drawBorder</a>	To print border
<a href="#">drawText</a>	To print text
<a href="#">drawText</a>	Print text in area
<a href="#">drawBarCode</a>	Print 1D barcode
<a href="#">drawBarCode</a>	Print 1D barcode in area
<a href="#">drawQrCode</a>	Print 2D barcode
<a href="#">drawGraphic</a>	Print bitmap
<a href="#">drawGraphic</a>	Print bitmap in area
<a href="#">print</a>	To print

- Setting the page width and page height of printing area

Method name: `public void pageSetup(LablePaperType paperWidth,int  
pageWidth, int pageHeight)`

Method description :Setting the page width and page height of printing area

Parameter:

`paperWidth` Set paper type, `LablePaperType.Size_80mm`, `80mmpaper`;

`LablePaperType.Size_58mm`, `58mmpaper`;

`LablePaperType.Size_100mm`, `100mmpaper`;

`pageWidth` Page Width

`pageHeight` Page Width

Returned Value: None

- Print straight line

Method name: `public void drawLine(int lineWidth, int startX, int startY, int  
endX,int endY,boolean isSolidLine)`

Method description :to print straight line

Parameter:

`lineWidth` width of the printing line

`startX` coordinate X of the starting position

`startY` coordinate Y of the starting position

endX                      coordinate X of the ending position

startY                    coordinate Y of the ending position

isSolidLine true solid line ;false dotted line

Returned value :Null

➤ To print border

Method description: public void drawBorder(int lineWidth, int top\_left\_x, int top\_left\_y, int bottom\_right\_x, int bottom\_right\_y)

Method description :to print border

Parameter :

lineWidth                Width of the printing line

top\_left\_x                Coordinate X of the left border position

top\_left\_y                Coordinate Y of the left border position

bottom\_right\_x          border bottom right position coordinate X

bottom\_right\_y          border bottom right position coordinate Y

Returned value:Null

➤ Print text

Method description: public void drawText(int text\_x, int text\_y, String text, LableFontSize fontSize, PRotate rotate, int bold, int reverse, int underline)

Method description: To print text at point(text\_x, text\_y)

Parameter description:

text\_x                      Coordinate X of printing started point

text\_y                      Coordinate Y of printing started point

text                        Text content to be printed

fontSize                    font size ,support font as follow:

LableFontSize.Size\_16              16dots

LableFontSize.Size\_24              24dots

LableFontSize.Size\_32              32dots

LableFontSize.Size\_48              48dots

LableFontSize.Size\_64              64dots

LableFontSize.Size\_72              72dots

LableFontSize.Size\_96              96dots

rotate                      Rotation angle

PRotate.Rotate\_0                      Rotation 0 degrees

PRotate.Rotate\_90                      Rotation 90 degrees

**PRotate.Rotate\_180**

Rotation 180 degrees

**PRotate.Rotate\_270**

Rotation 270degrees

<b>bold</b>	Bold or not	1 Bold	0 Not bold
<b>reverse</b>	reverse or not	1 reverse	0 not reverse
<b>underline</b>	underline or not	1underline	0 not underline

Returned value: Null

➤ **Print text in area**

Method instruction :`public void drawText(int area_start_x, int area_start_y, int area_end_x,int area_end_y, PAlign xAlign, PAlign yAlign, String text,LableFontSize fontSize, int bold, int reverse, int underline, int deleteline,PRotate rotate)`

Method description:to print text in point area

Parameter:

<b>area_start_x</b>	coordinate X of the rectangular region top left corner		
<b>area_start_y</b>	coordinate Y of the rectangular region top left corner		
<b>area_end_x</b>	coordinate X of the rectangular region right bottom corner		
<b>area_end_y</b>	coordinate Y of the rectangular region right bottom corner		
<b>xAlign</b>	Horizontal position,it can be left align PAlin.START; center PAlign.CENTER ;right align PAlign.END		
<b>yAlign</b>	Vertical position,it can be upper align PAlin.START; center PAlign.CENTER ;bottom align PAlign.END		
<b>fontSize</b>	refer to above method description to drawText:		
<b>bold</b>	Bold or not	1 Bold	0 Not bold
<b>reverse</b>	reverse or not	1 reverse	0 not reverse
<b>underline</b>	underline or not	1underline	0 not underline
<b>deleteline</b>	deleteline or not	1deleteline	0 not deleteline
<b>rotate</b>	rotation angle ,pls refer to above method.		

Returned value:Null

➤ **Print 1D barcode in area**

Method description: `drawBarCode(int area_start_x, int area_start_y, int area_end_x,int area_end_y, PAlign xAlign, PAlign yAlign, int start_x,int`

start\_y, String text, PBarcodeType type, int linewidth,int height, PRotate rotate)

Method description: to Print 1D barcode in area

Parameter:

area_start_x	coordinate X of the rectangular region top left corner
area_start_y	coordinate Y of the rectangular region top left corner
area_end_x	coordinate X of the rectangular region right bottom corner
area_end_y	coordinate Y of the rectangular region right bottom corner
xAlign	Horizontal position,it can be left align PAlin.START; center PAlign.CENTER ;right align PAlign.END
yAlign	Vertical position,it can be upper align PAlin.START; center PAlign.CENTER ;bottom align PAlign.END
start_x	X coordinate of the bar code starting position in the area
start_y	Ycoordinate of the bar code starting position in the area
text	The barcode data content to be printed
type	1D barcode type: supported type as follow: PBarcodeType.CODE128,PBarcodeType. JAN3_EAN13,PBarcodeType. JAN8_EAN8, PBarcodeType. CODE93,PBarcodeType. JAN3_EAN13,PBarcodeType. UPCAPBarcodeType. ITF, PBarcodeType. UPC_E
linewidth	Barcode line width
height	Height of barcode
rotate	Rotation angle of barcode ,i.e:PRotate.Rotate_0,PRotate.Rotate_90,PRotate.Rotate_180,PRotate.Rotate_270

Returned value:Null

#### ➤ Print 2D barcode

Method instruction: public void drawQrCode(int start\_x, int start\_y, String text, PRotate rotate, int ver, int lel)

Method description: to print 2D barcode

Parameter:

start_x	2D barcode starting position X
start_y	2D barcode starting position Y
text	Content data of 2D barcode
rotate	same rotation angel with 1D barcode
ver	magnification times of QrCode is(1-6),six by default

lel                      Error correction level ,refer to the commands manual for details  
 0,Error correction level L;1,Error correction level M; 2,Error correction level Q,3,  
 Error correction level H  
 Returned value:Null

➤ To print bitmap

Method instruction: `public void drawGraphic(int start_x, int start_y, int  
 bmp_size_x,  
 int bmp_size_y, Bitmap bmp)`

Method description: Print bitmap in the specified location

Parameter description:

start_x	coordinate X of the printed bitmap location
start_y	coordinate Y of the printed bitmap location
bmp_size_x	bitmap Width
bmp_size_y	bitmap Height
bmp	bitmap object

➤ To print bitmap in area

Method instruction: `public void drawGraphic(int area_start_x, int area_start_y,  
 int area_end_x,int area_end_y, PAlign xAlign, PAlign yAlign, int  
 bmp_size_x,int bmp_size_y, Bitmap bmp)`

Method description: Print bitmap in area

Parameter:

area_start_x	coordinate X of the rectangular region top left corner
area_start_y	coordinate Y of the rectangular region top left corner
area_end_x	coordinate X of the rectangular region right bottom corner
area_end_y	coordinate Y of the rectangular region right bottom corner
xAlign	Horizontal position,it can be left align PAlin.START; center PAlign.CENTER ;right align PAlign.END
yAlign	Vertical position,it can be upper align PAlin.START; center PAlign.CENTER ;bottom align PAlign.END
bmp_size_x	bitmap Width
bmp_size_y	bitmap Height
bmp	bitmap object

Returned value:Null

➤ To Print

Method instruction: public void print(PRotate rotate,int skip)

Method description: to print label, will not print immediately according to the previous order to print text or straight line,until this method is called .

Parameter description:

rotate 0: to print normally, no rotation 1: to print after the entire page is rotated clockwise 180degrees.

Skip: 0: after ending the printing ,directly stop without located.

1: located at label gap after ending the printing .stop after feeding paper at max.30mm if no gap .

Returned value:Null

### 3.1.6 TSPL Command Label Printing Related Method

Note: Below methods are apply for label printer ,which support standard TSPL command set.

Method	Description
<a href="#">pageSetupTSPL</a>	Set page width and height
<a href="#">drawLineTSPL</a>	Draw line
<a href="#">drawBorderTSPL</a>	To print border
<a href="#">drawTextTSPL</a>	To print text
<a href="#">drawTextTSPL</a>	To print text in area
<a href="#">drawBarCodeTSPL</a>	To print 1D barcode
<a href="#">draw2DBarCodeTSPL</a>	To print 2D barcode(TL21type not support DMATRIX command)
<a href="#">drawBitmapTSPL</a>	To print bitmap
<a href="#">drawBitmapTSPL</a>	To print bitmap in area
<a href="#">printTSPL</a>	Print images content in buffer
<a href="#">getPrinterStatusTSPL</a>	To get printer status
<a href="#">setCharsetNameTSPL</a>	Set the character set encoding format
<a href="#">setPaperbackOrPaperFeedTSPL</a>	Control the paper feeding or back
<a href="#">reverseAreaTSPL</a>	reverse Print in area
<a href="#">eraseAreaTSPL</a>	Clear image buffer data
<a href="#">setPrinterTSPL</a>	Setting printer related function(TL21printer not support SPEED, BLINE, SHIFT, REPRINT command, TL51not support PEEL command)



<a href="#">openCashBoxTSPL</a>	Open cash drawer
<a href="#">getPrinterNameTSPL</a>	Checking the printer model
<a href="#">setGAPTSPL</a>	setting the vertical spacing between labels
<a href="#">selectCodePageTSPL</a>	select codePage
<a href="#">selectCountryTSPL</a>	Select International character set(TL21model not support this method)
<a href="#">beepTSPL</a>	Control buzzer beep one sound
<a href="#">downloadBitmap2PrinterTSPL</a>	Download bitmap to printer(TL21model not support this method)
<a href="#">putBitmapTSPL</a>	Put downloaded bitmap into printer buffer(TL21does not support shi method)
<a href="#">printSelfTestTSPL</a>	Print selftest paper(TL21does not support this method)
<a href="#">setLabelReferenceTSPL</a>	Set reference origin of the label content
<a href="#">sendStrToPrinterTSPL</a>	Send string in command format to printer(TL21does not support SET COUNTER Commands)

All exports below are made by throwing the abnormal way. The main exceptions are as follows:

Write exception      WriteException

Read exception      ReadException

PrinterPort null exception.    PrinterPortNullException

Parameter error exception      ParameterErrorException

➤ Set page width and height

Method instruction: public void pageSetupTSPL(int paperSizeType, int pageWith, int pageHeight)

throws WriteException ,PrinterPorNullException, ParameterErrorException

Method description: Set page width and page height, users pass specified value to paperSizeType based on paper width 2 inch, 3 inch or 4 inch, The code maintains a static variable based on the paperSizeType size, when printing line, the border need to judge whether exceed paper width according to passed coordinate and paperSizeType.

Parameter description:

paperSizeType                      Paper size type

0 means choosing 2 inch printed paper (20mm~56mm)

1 means choosing 3 inch printed paper (20mm~80mm)

2 means choosing 4 inch printed paper (20mm~108mm)

pageWidth Page width, i.e. Printed content area width(dot), will print error if beyond area width

pageHeight Page height, i.e printed content area height(dot), will print error if beyond area height;

Returned value: Null

➤ Draw line

public void drawLineTSPL(int startX, int startY, int lineLength, int lineWidth)  
throws WriteException, PrinterPortNullException, PrinterPortNullException,  
Exception

public void drawLineTSPL(int startX, int startY, int lineLength, int lineWidth)  
throws WriteException, PrinterPortNullException, PrinterPortNullException, Exception

Method description: draw line from specified coordinate

Parameter description:

startX              Starting position on horizontal direction of left corner, in dots(no exceed label width)

startY              Starting position on vertical direction of left corner, in dots(no exceed label height)

lineHeight, in dots

lineWidth, in dots

Returned value: Null

➤ To print border

public void drawBorderTSPL(int lineWidth, int top\_left\_x, int top\_left\_y, int  
bottom\_right\_x, int bottom\_right\_y)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: print border

Parameter description:

lineWidth              line width of border, in dots

top\_left\_x              X starting position of left up corner in horizontal direction on  
left corner, in dots

top\_left\_y              Y starting position of left up corner in vertical direction on left  
corner, in dots

bottom\_right\_x      X starting position of right bottom corner in horizontal direction on left corner, in dots

bottom\_right\_y      X ending position of right bottom corner in vertical direction on right bottom corner, in dots

Returned value: Null

➤ Print text

public void drawTextTSPL(int start\_x, int start\_y, boolean isSimplifiedChinese, int xMultiplication, int yMultiplication, PRotate rotate, String content)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: Print text

Parameter description:

start\_x              starting position coordinate in X direction of text, in dots

start\_y              starting position coordinate in Y direction of text, in dots

isSimplifiedChinese      true Simplified Chinese 24×24Font(GB); false Traditional Chinese 24×24Font(BIG5), note English and numbers both 12\*24.

isSimplifiedChinese is false, it must be traditional Chinese for the content, and set character to BIG5, otherwise it will print out mess code.

xMultiplication      the multiplication times of text width, range 1~4

yMultiplication      the multiplication times of text height, range 1~4

Note: use xMultiplication, yMultiplication can set font size

rotate              clockwise direction rotate, PRotate.Rotate\_0 no rotate. PRotate.Rotate\_90 rotate 90degree; PRotate.Rotate\_180 rotate180 degree; PRotate.Rotate\_270 rotate 270 degree

content      Text content to be printed

➤ Print text in area

public void drawTextTSPL(int area\_start\_x, int area\_start\_y, int area\_end\_x, int area\_end\_y, PAlign xAlign, PAlign yAlign, boolean isSimplifiedChinese, int xMultiplication, int yMultiplication, PRotate rotate, String content)

throws WriteException, PrinterPortNullException,

ParameterErrorException

Method description: print text in area, it could be specified horizontal direction and vertical direction position. Horizontal direction: left align, center, right align; Vertical direction: above align, center, below align;

Parameter description:

lineWidth      Border line width(dot)  
 top\_left\_x      X coordinate on top left corner of area(dot)  
 top\_left\_y      X coordinate on top left corner of area(dot)  
 bottom\_right\_x    X coordinate on bottom corner of area(dot)  
 bottom\_right\_y    Y coordinate on bottom right corner of area(dot)  
 xAlign          Position setting in X direction, left align, PAlign.START;  
 Center, PAlign.CENTER; Right align, PAlign.END  
 yAlign          Position setting on Y direction, Top align, PAlign.START; Center,  
 PAlign.CENTER; Below align, PAlign.END

Note: PAlign is self-defined Enum type. Including START, CENTER, END, NONE.  
 Print according to real coordinate when it is NONE, no calculation to deal with. Same  
 for below.

isSimplifiedChinese true Simplified Chinese 24×24Font(GB);false traditional Chinese  
 24×24Font,when isSimplifiedChinese is false, it has to be traditional Chinese in  
 content, and set character to BIG5, otherwise it will printed mess code Chinese.

xMultiplication      Multiplication times for text width, range 1~4  
 yMultiplication      Multiplication times for text width, range 1~4

Note: Use xMultiplication, yMultiplication can set font size

rotate              clockwise rotate, PRotate.Rotate\_0 no rotate;  
 PRotate.Rotate\_90 rotate 90 degree; PRotate.Rotate\_180 rotate 180 degree;  
 PRotate.Rotate\_270 rotate 270 degree  
 content      text content to be printed

#### ➤ Print 1D barcode

public void drawBarCodeTSPL(int start\_x, int start\_y, PBarcodeType type,  
 int height, boolean isReadable, int narrowWidth ,  
 int wideWidth, PRotate rotate, String content)  
 throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: Print 1D barcode

Parameter description:

start\_x              Starting position coordinate on X direction (dot)  
 start\_y              Starting position coordinate on Y direction(dot)  
 type                  Barcode type, support below type:  
 PBarcodeType. 128              128 code  
 PBarcodeType. EAN128              EAN128  
 PBarcodeType. 39                  39 code  
 PBarcodeType. 93                  93 code  
 PBarcodeType. EAN13              EAN13 code  
 PBarcodeType. EAN8              EAN8 code

---

PPBarcodeType. CODA	CODA
PPBarcodeType. UPCA	UPCA code
PPBarcodeType. UPCE5	UPCE+5 code
height	Barcode height, in dots
isReadable	0, not readable by human eyes; 1, readable
narrowWidth	narrow strip width [0~5]
wideWidth	wide strip width $0 \leq \text{wideWidth} / (\text{narrowWidth} + 1) < 5$ $20 \leq \text{wideWidth} / (\text{narrowWidth} + 1) \leq 30$

Note: narrow strip and wide strip width could be set by narrowWidth and wideWidth, to enlarge barcode width.

rotate                      clockwise rotate, PRotate.Rotate\_0 no rotate;  
 PRotate.Rotate\_90 rotate 90degree; PRotate.Rotate\_180 rotate 180degree;  
 PRotate.Rotate\_270 rotate 270 degree  
 content                      barcode content

Returned value: Null

➤ Print 2D barcode

public void draw2DBarCodeTSPL(int start\_x, int start\_y, TwoDarCodeType type , int param1, int param2, PRotate rotate, String content)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: Print 2D Barcode

Parameter description:

start\_x                      X direction starting point coordinate(dot)

start\_y                      Y direction starting point coordinate(dot)

type                      2D Barcode type (QRCode, DataMatrix)

TwoDarCodeType.QR                      QRcode

TwoDarCodeType. DataMatrix                      DataMatrix

param1                      when the bar code type is DataMatrix, this parameter is invalid;When barcode type is QRCode, this parameter represents the error correction level of the QR code, 0 represents L 7%, 1 represents M 15%, and 2 represents Q 25%, and 3 represents H 30%.

param2                      parameter represents 2D Barcode block width (1~6)

rotate                      The rotation angle of the bar code, clockwise. 0, not rotate;90,rotate 90 degree; 180, clockwise rotate 180 degree; 270, clockwise rotate 180 degree. Note: When barcode type is DataMatrix, this parameter is invalid, pass freely.

content                      Barcode content

Returned value: Null

➤ Print bitmap

public void drawBitmapTSPL(int start\_x, int start\_y, int mode, Bitmap bmp)  
 throws WriteException, PrinterPortNullException, ParameterErrorException  
 Method description: Print bitmap

Parameter description:

start_x	horizontal starting position in dots image(dot)
start_y	vertical starting position in dots image(dot)
mode	image drawing mode 0 OVERWRITE ;1 OR ;2 XOR
bmp	bitmap objective(single color bitmap)

Returned value: Null

➤ Print bitmap in area

public void drawBitmapTSPL(int area\_start\_x, int area\_start\_y, int area\_end\_x, int  
 area\_end\_y, PAlign xAlign, PAlign yAlign, int mode, Bitmap bmp)  
 throws WriteException, PrinterPortNullException, ParameterErrorException  
 Method description: print text in border area, could set center, right align, left align on  
 horizontal direction; Top align, center, below align on vertical direction.

Parameter description:

area_start_x	x coordinate of top left corner in area(dot)
area_start_y	y coordinate of top left corner in area(dot)
area_end_x	x coordinate of bottom right corner in area(dot)
area_end_y	y coordinate of bottom right corner in area(dot)
xAlign	position in horizontal direction
yAlign	position in vertical direction
mode	bitmap drawing mode 0 OVERWRITE;1 OR; 2 XOR
bmp	bitmap to be printed(single color bitmap)

Returned value: Null

➤ Print content in image buffer

public void printTSPL(int mSets, int mCopys)  
 throws WriteException, PrinterPortNullException,  
 ParameterErrorException  
 Method description: print image butter, call print line, print text, barcode etc, printer  
 will download the printed content into buffer, then print out by calling this method,  
 specify how much pieces to be printed via mSets parameter, and how much pieces  
 label to be printed in every piece of content via mCopus.

Parameter description:

mSets	Printed pieces quantity
mCopys	Printed label quantity in every piece

Returned value: Null

➤ Get printer status

`public int getPrinterStatusTSPL()`

throws `WriteException`, `PrinterPortNullException`, `ReadException`,  
`ParameterErrorException`

Method description: check printer status, including paper out, cover opened, and other error status which influence normal printing.

Parameter: Null

Returned value:

0 normal status

-1 paper out

-2 paper house cover opened

-3 other error

➤ Set character set code format

`public void setCharsetNameTSPL(String charsetName)`

throws `ParameterErrorException`

Method description: Set character set code format, note: this method is only applicable to ESC/POS, CPCL, TSPL command set.

Parameter:

charsetName Character set name; Simplified Chinese, "GBK"; Traditional Chinese, "BIG5"

Note: When using `drawTextTSPL` method to print traditional Chinese, call this method to set character set to "BIG5"

Returned value: Null

➤ To control feed paper or back

`public void setPaperbackOrPaperFeedTSPL(boolean isFeedBack, int mDot)`

throws `WriteException`, `PrinterPortNullException`, `ParameterErrorException`

Method description: control feed paper or back

Parameter:

isFeedBack back paper or not, true: back paper; false: feed one piece of paper

mDot dots of back paper; When choosing feeding paper, this parameter is invalid.

Returned value: Null

➤ Reverse printing in specified area

`public void reverseAreaTSPL(int start_x, int start_y, int width, int height)`

throws `WriteException`, `PrinterPortNullException`, `ParameterErrorException`

Method description: reverse printing in specified area

Parameter:

start\_x X coordinate of top left corner in area (dot)

start\_y Y coordinate of top left corner in area (dot)

width area width(dot)

height area height(dot)

Returned value: Null

➤ Clear data in image buffer

public void eraseAreaTSPL(int start\_x, int start\_y, int width, int height)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: clear data in this image buffer area

Parameter:

start\_x X coordinate of left corner in area (dot)

start\_y Y coordinate of left corner in area (dot)

width Area width (dot)

height Area height (dot)

Returned value: None

➤ To set related function of printer

public void setPrinterTSPL(CommandTSPL command,int value)

throws WriteException, PrinterPortNullException,

ParameterErrorException

Method description: set related function of printer

Parameter:

command

CommandTSPL .DIRECTION Set print direction; same as font when value is 0, opposite when value is 1.

CommandTSPL .FEED Control paper feeding set;  $1 \leq \text{value} \leq 1000$  means dots.

CommandTSPL .REPRINT Run reprinting order when error; Close this function when value is 1, open when value is 1.

CommandTSPL .SPEED Specify printing speed; value is the inch value the paper feed every second.

CommandTSPL .DENSITY Specify printing density;  $0 \leq \text{value} \leq 15$ , the bigger the number is, the bigger density is.

CommandTSPL .SHIFT Paper feed offset value; value >0, paper feed direction is same with printing direction; Opposite when it is <0;

CommandTSPL .FORMFEED Control the printer feed one piece of paper; Value transmit freely.

CommandTSPL .HOME Search starting position, make the gap align paper manual cutter; value transmit freely.

CommandTSPL .PRINTKEY Set printing by button order(forbidden, allowed, automatically set function of printing by button)

CommandTSPL .KEY1 Command of open function to preset KEY1 (this function is pause or feed paper)



CommandTSPL . KEY2      Command of open function to preset KEY2 (this function is pause or feed paper)

CommandTSPL . TEAR      Enable of close command of feeding paper to manual cutting position(gap align to manual cutting position)

CommandTSPL.PEEL      Enable peeler mode

Value      Detail meaning corresponds to command

Returned value: None

➤ Open cash drawer

public void openCashBoxTSPL()  
throws WriteException, PrinterPortNullException,  
ParameterErrorException  
Method description: open cash drawer  
Parameter: None  
Returned value: None

➤ Check printer model No.

public String getPrinterNameTSPL()  
throws WriteException, PrinterPortNullException,  
ParameterErrorException,ReadException  
Method description: check printer model NO.  
Parameter: None  
Returned value: Printer detail model No.

➤ Set vertical spacing between labels

public void setGAPTSPL(int value)  
throws WriteException, PrinterPortNullException,  
ParameterErrorException  
Method description: Set vertical spacing between labels  
Parameter:  
value      vertical spacing between labels(unit: mm),  
mm( $0 \leq m \leq 1$ (inch),  $0 \leq m \leq 25.4$ (mm))  
Returned value: None

➤ Select codepage

public void selectCodePageTSPL(int value)  
throws WriteException, PrinterPortNullException,  
ParameterErrorException  
Method description: Select codepage  
Parameter:  
Value      437: United States

850: Multilingual

852: Slavic

860: Portuguese

863: Canadian/French

865: Nordic

857: Turkish

1250: Central Europe

1252: Latin I

1253: Greek

1254: Turkish

Transmit other value: default by 437

Returned value: None

- Select international Character set

public void selectCountryTSPL(int value)

throws WriteException, PrinterPortNullException,

ParameterErrorException

- Method description: Select international Character set

Parameter:

Value

1: USA

2: Canadian-French

3: Spanish (Latin America)

33: French (France)

34: Spanish (Spain)

39: Italian

42: Slovak

44: United Kingdom

45: Danish

46: Swedish

47: Norwegian

49: German

61: English (International)

Pass other value: 1 by default

Returned value: None

- Control buzzer beep one sound

public void beepTSPL(int level,int interval)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: Set the sound level and interval of buzzer, need to send BEEP after set, then buzzer will sound

Parameter:

Level	Level of buzzer sound
Interval	Interval of buzzer sound

Returned value: None

➤ Download bitmap in printer

public void downloadBitmap2PrinterTSPL(boolean isMove2Flash, InputStream in, String fileName)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: download bitmap in printer

Parameter:

isMove2Flash false download bitmap to printer RAM(the data in RAM will lost if lose power); true download to Flash(enter in RAM data and move to FLASH by MOVE command)

in input stream of bitmap under the downloaded raw content.

filename download into RAM or printer FLASH, and specify filename, 8 characters longest.

Returned value: None

Note: Bitmap data dots as per BITMAP command rule.

➤ Put downloaded bitmap into printer buffer

public void putBitmapTSPL(String fileName,int start\_x, int start\_y)

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: Put downloaded bitmap data in printer flash RAM or FLASH into printing buffer area coordinate(start\_x,int start\_y)

Parameter:

filename the bitmap name which has downloaded into printer, have to be same with Bitmap name in printer RAM or FLASH, otherwise it possibly cannot be printed out, i.e, downloadBitmap2Printer method download bitmap into printer, put Bitmap in printing buffer area.

start\_x Bitmap data Horizontal X coordinate

start\_y Bitmap data Horizontal Y coordinate

Returned value: None

➤ Print selftest paper

public void printSelfTestTSPL()

throws WriteException, PrinterPortNullException, ParameterErrorException

Method description: printer send command to print selftest paper

Parameter: None

Returned value: None

- Set reference coordinate origin of label content

public void setLabelReferenceTSPL(int left,int top)

throws PrinterPortNullException,ParameterErrorException

Method description: Set reference coordinate origin of label, adjust the label content position on label.

Parameter: left left margin of label content to label paper(mm)

top top margin of label content to label paper(mm)

Returned value: None

- Send string in format of command to printer

Method description: public void sendStrToPrinterTSPL(String str)

throws PrinterPortNullException,WriteException,

ParameterErrorException

Method description: send TSPL command in SDK to printer

Parameter description: str conform to TSPL command format

Returned value: None

### 3.1.7 Method to control printer hardware

Method	Description
<a href="#">cutPaper</a>	Cut paper
<a href="#">ringBuzzer</a>	Control buzzer beep
<a href="#">blackLableFind</a>	Black mark location

- To cut paper

Method instruction: public void cutPaper(int cutterType, int n)

Method description: cut paper, full cut or partial cut

Parameter description: cutterType

48 full cut directly

49 partial cut directly

65 feed paper to (paper cutting position+[n ×(Vertical moving unit) ]) and full cut

66 feed paper to (paper cutting position+[n×( Vertical moving unit) ])and partial cut

Note: vertical moving unit is one unit

Android-SDK Printer Developing Manual

n: Longitudinal movement points, valid when cutterType is 65

Returned value: Null

➤ To control buzzer beep

Method instruction: public void ringBuzzer(int time)

Method description: to control the time buzzer will beep(seconds)

Parameter description: time: seconds which buzzer will beep

Returned value: Null

➤ Black mark location

Method description: public void blackLableFind()

Method description: printer feed paper to black mark position(printer need to support black mark location function, with black mark sensor). Meanwhile black mark location function is only valid under black mark mode, if could not find black mark, then printer will stop after feed paper 30cm. (How to open black mark location function, please contact our pre-sales technique to ask setting tool, set related function enable.)

Parameter: Null

Returned value: Null

### 3.1.8 Get the current state of the printer

Method	Description
<a href="#">getCurrentStatus</a>	Get printer current status
<a href="#">getPrintingStatus</a>	Printing finished or not
<a href="#">isPaperOut</a>	Tell paper out or not
<a href="#">isPaperWillOut</a>	Tell printer paper near end or not

➤ Get the current state of the printer

Method name: public int getCurrentStatus ()

Method description: to get the current status of the printer, including printer status normal, lack of paper, cover opening, paper near end, communication abnormal. All printers are with paper near end test function except EU seriez printer, which function need to be manual opened to test paper near end. (How to open paper near end function, please contact our pre-sale technique to get setting tool, set related function enable)

Parameter description: Null

Returned value:

0            Printer status normal

-1           Printer communication abnormal

- 2 Paper out
- 3 Paper near end
- 4 Paper house cover opened

➤ Printing finished or not

Method description: public int getPrintingStatus(String BufferMsg, int timeout)

Method description: Get whether the printer completes printing

Parameter Description: Msg, Receive information of whether printing is completed, specific value and the significance of the method's returned Android-SDK Printer Developing Manual value corresponds. that is, the "current printing has been completed," "unknown anomaly", "Printing is not completed, out of paper" and so on.

Timeout :estimate requiring time of printing one order

Returned value:

- 0 Printing completed
- 1 Unknown abnormal
- 2 The current printer is printing
- 3 Printing is not completed, out of paper
- 4 Printing is not completed, printer paper cover opens
- 5 Printing is not completed, fail to communicate with the printer
- 6 data transmission failure, abnormal communication
- 7 receiving data format is incorrect

Principle: printer single-chip microcomputer CPU execution data is sequential, send print data first, after sending the query whether print complete instructions, according to the principle of performing data after the "junior", execute commands to print complete judgment, must be performed to print data.

Example 1: query printing is completed

// 1. Judge the current status of the printer before sending the data to print

```
If (getCurrentStatus ()!= 0) {
```

```
Return;
```

```
}
```

// 2, send print data

```
MPrinter. PrinterText (" hello world!");
```

```
StringBuffer STR = new StringBuffer (" ");
```

```
Int timeout = 2000;// expected to print the time of the above text
```

// 3, determine whether the printing is finished

```
Int ret = mPrinter. GetPrintingStatus (STR, timeout);
```

```
If (ret == 0) {
```

```
Log. I (SPRT "", " text print finished!");
```

```
}
```

- Judge whether the printer is out of paper  
 Method Name: public boolean isPaperOut ()  
 Method Description: judge whether the printer is out of paper  
 Parameters: None  
 Return Value: true is out of paper, false is not out of paper
- Judge whether printer is out of paper  
 Method name: public boolean isPaperWillOut()  
 Method description: Judge printer is paper out or not, only some model have paper near end sensor, need to open paper near end function before getting paper near end status.  
 Parameter description: Null  
 Returned value: true paper near end, false paper enough.

### 3.1.9 The call examples of connecting printers

```
//1、 handler created by n is for receiving message of connection success or failure
privateHandlermHandler = newHandler() {
@Override
publicvoidhandleMessage(Messagemsg) {
switch (msg.what) {
caseConnect.SUCCESS:
    isConnected = true;//connect successfully
    break;
caseConnect.FAILED:
    isConnected = false;
    Toast.makeText(mContext, R.string.conn_failed,Toast.LENGTH_SHORT).show();
    Log.i(TAG, "connect fail!");
    break;
caseConnect.CLOSED:
    isConnected = false;
    Toast.makeText(mContext, R.string.conn_closed,Toast.LENGTH_SHORT).show();
    Log.i(TAG, "connection closed!");
    break;
caseConnect.NODEVICE:
    isConnected = false;
    Toast.makeText(mContext, R.string.conn_no, Toast.LENGTH_SHORT).show();
    break;
default:
```

```

        break;
    }
//2、 Instantiate printers, other communication mode instantiate this object, please refer to
demo
BluetoothDevice mDevice =
BluetoothAdapter.getDefaultAdapter().getRemoteDevice(devicesAddress);//devicesAddress is mac address of Bluetooth
PrinterInstance mPrinter = PrinterInstance.getPrinterInstance(mDevice, mHandler);
//3、 Open connection
mPrinter.openConnection();
//4、 Start to print data
//Judge whether connection is normal before printing, and printer is sufficient of paper, and
paper
cover is closed
if (mPrinter != null && mPrinter.getCurrentStatus() == 0){
    mPrinter.setFont(0,1,1,1,1);//Set the font, double height, double width, bold, underline
    mPrinter.setPrinter(Command.ALIGN, Command.ALIGN_CENTER);//Set font center
    mPrinter.printText("printTest!" + "\r\n");//Print text printTest!
    mPrinter.setPrinter(Command.PRINT_AND_WAKE_PAPER_BY_LINE, 2);
    mPrinter.setFont(1,1,1,1,1);//Set font 9*17, double height, double width, bold, underline
    mPrinter.printText("printTest!" + "\r\n");//Print text printTest!
}else {
    //Printer status is abnormal:connection error,out of paper, paper cover is opening, near-
end
paper
Toast.makeText(mContext, "printer status is not normal! ", 1).show();
}
//5、 close printer connection
mPrinter.closeConnection();//close connection
Below use Bluetooth communication method calls for example ,other connect call please
refer "PrintDemoV5.3".

```

## 3.2 Barcode Printing Class

<a href="#">constructor function</a>
<a href="#">Barcode Printing call example</a>



### 3.2.1 constructor function

Method	Description
<a href="#">Barcode</a>	Create Barcode example

➤ Create Barcode examples

Method name: Barcode(byte barcodeType, int param1, int param2, in param3, String content);

Method description: create Barcode examples

Parameter description:

barcodeType is barcode type, the Type constants begin with Printer Constants

1D Barcode: UPC\_A, UPC\_E, JAN13, JAN8, CODE39, ITF, CODABAR, CODE93, CODE128.

2D barcode: PDF417,DATAMATRIX,QRCODE.

param1,param2,param3 are specific barcode parameters:

When code type is 1D barcode, expressed by three parameters:

param1: barcode transverse width,  $2 \leq n \leq 6$ , default is 2

param2: barcode height  $1 \leq n \leq 255$ , default is 162

param3: barcode note location, 0--No print, 1--Above, 2--Below, 3--Above and below, both print . Barcode type is 2-D barcodes, the three parameters shows different meanings:

#### 1. PDF417

param1: indicates the number of characters per line,  $1 \leq n \leq 30$ .

param2: indicates Error Correction Level,  $0 \leq n \leq 8$   $0 \leq n \leq 8$ .

param3: indicates longitudinal magnification.

#### 2. DATAMATRIX

param1: indicate graph height,  $0 \leq n \leq 144$ (0:automatic selection).

param2: indicate graph width,  $8 \leq n \leq 144$ (param1 is 0, invalid).

param3: indicates Longitudinal magnification.

#### 3. QRCODE

param1: indicates graphic version NO.,  $1 \leq n \leq 30$ (0:automatic selection).

param2: indicates Error Correction Level,  $n = 76, 77, 81, 72$ (L: 7%, M:15%, Q:25%, H:30%).

param3: indicates Longitudinal magnification.

Content is barcode data.

### 3.2.2 Barcode Printing call example

- Print 1-D barcode call examples, for example code128

```
Barcodebarcode1 = newBarcode(BarcodeType.CODE128, 2, 150, 2,"123456");
mPrinter.printBarCode(barcode1);//mPrinter Printer Instance, and the printer is
connected
```

- Print 2-D barcode call examples, for example QRCode

```
Barcodebarcode2 = newBarcode(BarcodeType.QRCODE, 2, 3, 6,"123456");
mPrinter.printBarCode(barcode1);//mPrinter Printer Instance, and the printer is
connected
```

NOTE: For more barcode printing calls, pls See BarcoePrintActivity.java

## 3.3 Table Form Printing Class

<a href="#">Create Table examples</a>
<a href="#">Add One Row Data</a>
<a href="#">Setting the column data alignment in the Table</a>
<a href="#">Form printing call example</a>

### 3.3.1 Create Table examples

Method	Description
<a href="#">Table</a>	Create Table examples

```
public Table(String column, String regularExpression, int columnWidth[])
```

Method description: Create Table examples

Parameter: column parameter, column is the header which separated with regular.

Example: "serial number, unit price, quantity, amount "

regularExpression is the string delimiter. For example the above ","

columnWidth is character width of each column in the table. The Calculation of default font size is two Chinese, one English, and then added together, such as the width of " Serical number" width is 4

Returned Value: Table examples

### 3.3.2 Add One Row Data

Method	Description
<a href="#">addRow</a>	Add One Row Data

public void addRow(String row)

Method Description: Add one row data

Parameters Description: A row

The data format is consistent with the header format. If the data in one table cell is beyond the limited character width, it will change line to print, if manually changing, it can add "\ n" on the position to change line.

Returned Value: No

### 3.3.3 Setting the column data alignment in the Table

Method	Description
<a href="#">setColumnAlignLeft</a>	Setting the column data alignment in the Table

Method description: setColumnAlignLeft(boolean left);

Method Description: Setting the column data alignment in the Table, default is right alignment.

Parameter Description: left Setting the column data align left in the Table.

Returned Value: Null

### 3.3.2 Add One Row Data

Method	Description
<a href="#">addRow</a>	Add One Row Data

public void addRow(String row)

Method Description: Add one row data

Parameters Description: A row

The data format is consistent with the header format. If the data in one table cell is beyond the limited character width, it will change line to print, if manually changing, it can add "\ n" on the position to change line.

Returned Value: No

### 3.3.3 Setting the column data alignment in the Table

Method	Description
<a href="#">setColumnAlignLeft</a>	Setting the column data alignment in the Table

Method description: `setColumnAlignLeft(boolean left);`

Method Description: Setting the column data alignment in the Table, default is right alignment.

Parameter Description: left Setting the column data align left in the Table.

Returned Value: Null

### 3.3.4 Form printing call example

```
String column ="Name, unit price, quantity, amount ";
Tabletable = newTable(column, ";", newint[] { 14, 6, 6, 6 });
table.addRow("Storage bags"+ ";10.00;1;10.00");
table.addRow("wire hook" + ";5.00;2;10.00");
table.addRow("Umbrella"+ ";5.00;3;15.00");
mPrinter.printTable(table);// mPrinteris is the real example object o f PrinterInstance, and
printer is connected.
```

NOTE: More details about pdf printing calls, pls refer to `print_note` click ticket printing example in `TextPrintActivity.java`

## 3.4 Canvas Printing Class

Brief Description: Canvas print is the layout of printing the uncommon language or custom by graphic type. Text can be drawn on the canvas; users can call a third-party font file to set the font, draw bar codes, graphics and so on, then Eventually convert the bitmap. please call `printImage` in 2.1.4 to complete canvas printing. The main methods are as follows:

Initialization canvas
Set Font Property
Canvas contextual method
Call sample for canvas printing

Method Name:public void init(PrinterType printerType)

Method Description: Initialization canvas

Parameters Description: Parameter is printer type .T9 designated canvas is 80mm paper width, PrinterType . TIII specified canvas 58mm paper width

Return Value: Null

Method Name:public void setFontProperty(FontProperty fp)

Method Description: set Font Property. Parameters is FontProperty type, FontProperty is a collection of font properties, including bold, italic, and so on.

Parameters Description: FontProperty examples

Returned Value: No

setTextSize(intsize);	Font Size
-----------------------	-----------

setItalic(booleanitalic);                      Italic or not  
 setStrikeThruText(booleanstrike);            deleting line or not  
 setUnderlineText(booleanunderline);        Underline or not  
 setFakeBoldText(booleanfakeBold);        Bold or not  
 Return Value: Null

### 3.4.3 Canvas content related method

Method	Description
<a href="#">drawText</a>	Draw string
<a href="#">drawLine</a>	Draw Line
<a href="#">drawRectangle</a>	Draw Rectangle
<a href="#">drawEllipse</a>	Draw ellipse
<a href="#">drawImage</a>	Draw image
<a href="#">getCanvasImage</a>	Getting the image drew on canvas and sending to the printer to print.
<a href="#">setTextAlignRight</a>	Set whether the text is right align
<a href="#">setTextExceedNewLine</a>	Set whether entering new line when text exceeds.
<a href="#">setUseSplit</a>	Set whether to use split string when changing the new line because of text exceeds, to avoid one word is split. The default is to split by Space.
<a href="#">setUseSplitAndString</a>	If the text is beyond the line, use a delimited string or not to avoid a single word split. Break up with the specified symbol.

#### ➤ Draw string

Method Name: public void drawText(String nStr);

public void drawText(float x, String nStr);

public void drawText(float x, float y, String nStr);

Parameters Description: x, y is the lower left corner coordinates of the string, nStr is the string to be drawn

Return Value: No

#### ➤ Draw Line

Method Name: public void drawLine(float startX, float startY, float stopX, float stopY);

Parameters Description: startX,startY is Start coordinates, stopX, stopY is end coordinates

Returned Value: Null

➤ Draw Rectangle

Method Name:drawRectangle(float left, float top, float right, float bottom);

Parameters Description:The parameter is the upper left corner, the bottom right corner

Returned Value: Null

➤ Drawing ellips,The parameter is the upper left corner of the ellipse, and the bottom right corner

drawEllips(float left, float top, float right, float bottom);

➤ Drawing image,The left and top parameters of the parameter are the top left coordinates of the image. Bitmap is the image file.

drawImage(Bitmapimage);

drawImage(float left, Bitmap image);

drawImage(float left, float top, Bitmap image);

➤ Getting the image drew on canvas and sending to the printer to print.

getCanvasImage();

➤ Set whether the text is on the right. For some special characters, such as Arabic language.

setTextAlignRight(boolean alignRight);

➤ Set whether to change one new line when text exceeds.

setTextExceedNewLine(boolean newLine);

➤ Set whether to use split string when changing the new line because of text exceeds, to avoid one word is split. The default is to split by Space.

setUseSplit(boolean useSplit);

setUseSplitAndString(boolean useSplit, String splitStr);

### 3.4.4 Call sample for canvas printing

```
CanvasPrintcp=newCanvasPrint();//Create canvas
```

```
BitmapbitmapCODE39 = createBitmapQR_CODE("123456789", 270, 270);//Create two-dimension bar code
```

```
cp.init(PrinterType.T9);
```

```
// Draw two-dimension bar code on the position of canvas with coordinate (0,0)
```

```
cp.drawImage(0, 0, bitmapCODE39);
```

```
cp.drawImage(0, 0, bitmapCODE39);
```

```
//Create Font
```

```

FontPropertyfp=newFontProperty();
// Set value of font property. Here the parameter number will be a little different
according to the different version of SDK.
fp.setFont(true, false, false, false, 40, null);
// Set Font
cp.setFontProperty(fp);
//Draw text on the appointed coordinate position of canva
cp.drawText(250,80,"scan upgrade");
cp.drawText(250, 120, "your intelligent car life");
cp.drawText(250,180,"service telephone number");
cp.drawText(250, 220, "4008317317");
//Draw text on the appointed coordinate position of canvas.
mPrinter.printImage(cp.getCanvasImage(), PAlign.NONE, 0);

```

Remark: About more detailed about canvas calling and printing, pls refer to btn\_canvas\_print in PicturePrintActivity.java

## 3.5 BitmapConvertor class

Class explanation: This class is mainly used to transfer colorful image to Black&White binary bitmap.

<a href="#">Transfer colorful image to Black&amp;White binary bitmap</a>
--

### 3.5.1 Transfer colorful image to Black&White binary bitmap

Method	Description
<a href="#">convertBitmap</a>	Transfer colorful image to Black&White binary bitmap

Name: public Bitmap convertBitmap(Bitmap inputBitmap)

Method description: Transfer colorful image to Black&White binary bitmap.

Support the common colorful image, including JPG, PNG, Bitmap, etc.

Parameter explanation:

inputBitmap : colorful bitmap, BitmapFactory.decodeResource can be used to transfer the common colorful image JPG, PNG, Bitmap, etc to inputBitmap , inputBitmap should be accord with printer printing width. Exceeding the printing width will cause incomplete printed image. If inputBitmap is too big, it will cause very slow transfer. So the compressing should be arranged firstly.



Returned value: null

Remark: Colorful image is 24, 32 bit depth. There will be different degree distortion when it is transferred Black&White binary bitmap with 1 bit depth. If need to print more clear picture, the user needs to make the clear Black&White binary bitmap. Calling `printImage(Bitmap bitmap)` in Printer Instance to Complete.

## 3.6 Correlation Class of Printing PDF file

Class explanation: The correlation classes are mainly

`PdfContext`, `CodecDocument`, `CodecPage`, these classes transfer PDF file to be bitmap, call `printImage` in `PrinterInstance` to let printer be able to print PDF file normally.

The PDF file should be black&white PDF file. If printing colorful PDF file, there will be different degree distortion.

<a href="#">Open PDF file</a>
<a href="#">Getting one page object of PDF file</a>
<a href="#">PDF is transferred to Bitmap</a>
<a href="#">Calling example for PDF printing</a>

### 3.6.1 Calling example for PDF printing

1、Add processing PDF files to code projects so bank, which is: `libmupdf.so`

2、Code processing flow

//1、Instantiation PDF files process core class

```
MuPDFCore core = new MuPDFCore(this, filePath);
```

//2、Get the page number of PDF files

```
int count = core.countPages();
```

//3、Get the width and height of PDF files

```
PointF pageSize = core.getPageSize(0); //Get the width and height of page 1 currently
```

```
float pageW = pageSize.x;
```

```
float pageH = pageSize.y;
```

//4、Transfer PDF files to bitmap

```
Bitmap bitmap = Bitmap.createBitmap((int)pageW, (int)pageH,
```

```
Bitmap.Config.ARGB_8888);
```

```
core.drawPage(0, bitmap, (int) pageW, (int) pageH, 0, 0, (int)pageW, (int)pageH);
```

//5、Zoom above PDF files

```
Bitmap zoomImage = Utils.zoomImage(bitmap, 500); //Scale by 500 pixels wide
```

//6、bitmaptransfer monochrome bitmap When bitmap is large and the operation is slow, please call processing in the threading.

//The bitmap might have a certain degree distortion because of the zoom and rotation.

```
Bitmap monoChromeBitmap = convertor.convertBitmap(zoomImage);
```

//7、Print Bitmap

```
mPrinter.printImage(monoChromeBitmap, PAlign.START, 0, false);
```

Remark: About more details of calling PDF printing, pls refer to

PdfPrintActivity.java calling.

### 3.7、CodePage Printer multilingual code page Print class

Multilingual code page print examples are defined in this class, which support part of languages in the world. Users should know the codepage of the language in advance and print by invoking the specified codepage. Some codepage might not be supported by part of printer, if users need, please contact us for customization services.

Remark: Traditional Chinese, which can be print by printText directly, such as:

```
mPrinter.printText("Print text!\n");
```

Method	Description
CodePagePrinter(PrinterInstance mprinter)	Construction Method
printTextInCP437	Print Codepage CP437 font(American and European standard)
printTextInCP850	Print multi-language (Multilingual)
printTextInCP932	Print CodepageCP932 font (Katakana) font
printTextInCP860	Print CodepageCP860 Portuguese font
printTextInCP863	Print CodepageCP863 Canadian French font
printTextInCP865	Print CodepageCP863 Northern Europe language font
printTextInWCP1251	Print Codepage WCP1251 Slavic font
printTextInMIK	Print Codepage MIK Slavic/Bulgarian font
printTextInCP862	Print Codepage CP862 Hebrew font
printTextInWCP1252	Print Codepage WCP1252, Latin1 font
printTextInWCP1253	Print Codepage WCP1252
printTextInCP852	Print Codepage CP852 Latin2 font
printTextInCP858	Print Codepage CP858multi-language Latin1+Euro font
printTextInCP720	Print Codepage CP720 Arabic font

printTextInCP864	Print Codepage CP864 Arabic font
printTextInISO_8859_1	Print Codepage ISO-8859-1 Western Europe language font
printTextInCP737	Print Codepage CP737 Greek language font
printTextInWCP1257	Print Codepage CP737 Baltic language font
printTextInCP855	Print Codepage CP855Slavic font
printTextInCP857	Print Codepage CP857 Turkish font
printTextInWCP1250	Print Codepage WCP1250 Middle Europe language font
printTextInCP775	Print Codepage CP775 font
printTextInWCP1254	Print Codepage WCP1254 Turkish font
printTextInWCP1255	Print Codepage WCP1255 Hebrew font
printTextInWCP1256	Print Codepage WCP1256 Arabic font
printTextInWCP1258	Print Codepage WCP1258 Vietnamese font
printTextInISO_8859_2	Print Codepage ISO_8859_2 Latin2 font
printTextInISO_8859_3	Print Codepage ISO_8859_2Latin3 font
printTextInISO_8859_4	Print Codepage ISO_8859_4 Baltic language font
printTextInISO_8859_5	Print Codepage ISO_8859_5 Slavic font
printTextInISO_8859_6	Print Codepage ISO_8859_6 Latin font
printTextInISO_8859_7	Print Codepage ISO_8859_7 Turkish font
printTextInISO_8859_8	Print Codepage ISO_8859_8 Hebrew font
printTextInISO_8859_9	Print Codepage ISO_8859_9 Turkish font
printTextInISO_8859_15	Print Codepage ISO_8859_15 Latin9 font
printTextInCP874	Print Codepage CP874 font

## 4. Appendix

### 4.1 Summarize of CODE128

In CODE 128, through using Character Set A, Character Set B and Character Set C alternatively to encode 128pcs ASCII characters, 100pcs 00~99 numbers and some special characters. For each Character Set, encoded characters are as below,

- Character A: ASCII characters 00H to 5FH
- Character B: ASCII characters 20H to 7FH
- Character C: 100pcs numbers 00~99

Code128 can also encode the below special characters:

- SHIFT characters

"SHIFT" can transfer the first character after SHIFT character from Character Set A to Character Set B, or from Character Set B to Character Set A. From the second character, it will recover to the previous character set of SHIFT. "SHIFT" characters can only transfer between Character Set A and Character Set B, but can not make the current encoding character enter or quit the status of Character Set C.

- Characters to choose Character Set (CODE A、CODE B、CODE C)

These characters can transfer the next encoding characters to Character Set A, B or C.

- Function Character(FNC1、FNC2、FNC3、FNC4)

The function of these function characters depends on the application software.

In Character Set C, only FNC1 can be used.

## 4.2 Character Set

### Character Set A

Character	Send data		Character	Send data		Character	Send data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
NULL	00	0	&	26	38	L	4C	76
SOH	01	1	'	27	39	M	4D	77
STX	02	2	(	28	40	N	4E	78
ETX	03	3	)	29	41	O	4F	49
EOT	04	4	*	2A	42	P	50	80
ENQ	05	5	+	2B	43	Q	51	81
ACK	06	6	,	2C	44	R	52	82
BEL	07	7	-	2D	45	S	53	83
BS	08	8	.	2E	46	T	54	84
HT	09	9	/	2F	47	U	55	85
LF	0A	10	0	30	48	V	56	86
VT	0B	11	1	31	49	W	57	87
FF	0C	12	2	32	50	X	58	88
CR	0D	13	3	33	51	Y	59	89
SO	0E	14	4	34	52	Z	5A	90
SI	0F	15	5	35	53	[	5B	91
DLE	10	16	6	36	54	\	5C	92
DC1	11	17	7	37	55	]	5D	93
DC2	12	18	8	38	56	^	5E	94
DC3	13	19	9	39	57	_	5F	95
DC4	14	20	:	3A	58	FNC1	7B,31	123,49
NAK	15	21	;	3B	59	FNC2	7B,32	123,50
SYN	16	22	<	3C	60	FNC3	7B,33	123,51
ETB	17	23	=	3D	61	FNC4	7B,34	123,52
CAN	18	24	>	3E	62	SHIFT	7B,53	123,83
EM	19	25	?	3F	63	CODEB	7B,42	123,66
SUB	1A	26	@	40	64	CODEC	7B,43	123,67
ESC	1B	27	A	41	65			
FS	1C	28	B	42	66			
GS	1D	29	C	43	67			
RS	1E	30	D	44	68			
US	1F	31	E	45	69			
SP	20	32	F	46	70			
!	21	33	G	47	71			
"	22	34	H	48	72			
#	23	35	I	49	73			
\$	24	36	J	4A	74			
%	25	37	K	4B	75			

## Character Set B

character	Send data		character	Send data		character	Send data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
SP	20	32	F	46	70	l	6C	108
!	21	33	G	47	71	m	6D	109
"	22	34	H	48	72	n	6E	110
#	23	35	I	49	73	o	6F	111
\$	24	36	J	4A	74	p	70	112
%	25	37	K	4B	75	q	71	113
&	26	38	L	4C	76	r	72	114
'	27	39	M	4D	77	s	73	115
(	28	40	N	4E	78	t	74	116
)	29	41	O	4F	79	u	75	117
*	2A	42	P	50	80	v	76	118
+	2B	43	Q	51	81	w	77	119
,	2C	44	R	52	82	x	78	120
-	2D	45	S	53	83	y	79	121
.	2E	46	T	54	84	z	7A	122
/	2F	47	U	55	85	{	7B,7B	123,123
0	30	48	V	56	86		7C	124
1	31	49	W	57	87	}	7D	125
2	32	50	X	58	88	—	7E	126
3	33	51	Y	59	89	DEL	7F	127
4	34	52	Z	5A	90	FNC1	7B,31	123,49
5	35	53	[	5B	91	FNC2	7B,32	123,50
6	36	54	\	5C	92	FNC3	7B,33	123,51
7	37	55	]	5D	93	FNC4	7B,34	123,52
8	38	56	^	5E	94	SHIFT	7B,53	123,83
9	39	57	_	5F	95	CODEA	7B,41	123,65
:	3A	58	`	60	96	CODEC	7B,43	123,67
;	3B	59	a	61	97			
<	3C	60	b	62	98			
=	3D	61	c	63	99			
>	3E	62	d	64	100			
?	3F	63	e	65	101			
@	40	64	f	66	102			
A	41	65	g	67	103			
B	42	66	h	68	104			
C	43	67	i	69	105			
D	44	68	j	6A	106			
E	45	69	k	6B	107			

## CharacterC

Character	Send data		character	Send data		character	Send data	
	Hex	Decimal		Hex	Decimal		Hex	Decimal
0	00	0	38	26	38	76	4C	76
1	01	1	39	27	39	77	4D	77
2	02	2	40	28	40	78	4E	78
3	03	3	41	29	41	79	4F	79
4	04	4	42	2A	42	80	50	80
5	05	5	43	2B	43	81	51	81
6	06	6	44	2C	44	82	52	82
7	07	7	45	2D	45	83	53	83
8	08	8	46	2E	46	84	54	84
9	09	9	47	2F	47	85	55	85
10	0A	10	48	30	48	86	56	86
11	0B	11	49	31	49	87	57	87
12	0C	12	50	32	50	88	58	88
13	0D	13	51	33	51	89	59	89
14	0E	14	52	34	52	90	5A	90
15	0F	15	53	35	53	91	5B	91
16	10	16	54	36	54	92	5C	92
17	11	17	55	37	55	93	5D	93
18	12	18	56	38	56	94	5E	94
19	13	19	57	39	57	95	5F	95
20	14	20	58	3A	58	96	60	96
21	15	21	59	3B	59	97	61	97
22	16	22	60	3C	60	98	62	98
23	17	23	61	3D	61	99	63	99
24	18	24	62	3E	62	FNC1 CODEA CODEB	7B,31	123,49
25	19	25	63	3F	63		7B,41	123,65
26	1A	26	64	40	64		7B,42	123,66
27	1B	27	65	41	65			
28	1C	28	66	42	66			
29	1D	29	67	43	67			
30	1E	30	68	44	68			
31	1F	31	69	45	69			
32	20	32	70	46	70			
33	21	33	71	47	71			
34	22	34	72	48	72			
35	23	35	73	49	73			
36	24	36	74	4A	74			
37	25	37	75	4B	75			