**Android-SDK Development Manual**

(V1.6)

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 5.6 version. This version is modified on the basis of SDK5.5 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 |  | 1. Modify the PDF print in the SDK  2. Optimize the serial failure problem on non-root device  3. Add printed language instruction  4. Optimize the method of printing pictures  5. In the SDK, I added the print line, dotted line method  6. Optimize other known bugs  7. Added a quick reading guide on the document |
| 1.3 | 2017-12-24 | | Yang Xuezhi |  | 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. 2.Optimizes the method of querying the completion process of printing. Remove the parameter which display the information  of query results.   1. 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.  4.Modify the method of how to download bitmap to the printer with TSPL instruction, which enable normal bitmapdownload to printer cache.  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.  7. Regulate the modulation logs for modifying the interfaces. |
| 1.4 | 2018-12-07 | | Sun Chao Yang Xuezhi |  | 1. Page Mode has been added. Refer to 3.8 2. Optimize the problem of losing data on the device with USB timeout. 3. Add print LOG to SD card, refer to 3.9 4. Capture printing function of Y33 has been added. Refer to 3.10 5. Modify the instantiation mode of the serial port and add a new parameter ctsrts to select the flow control mode. 6. Add a function to set whether the Y33 print service is configurable, and whether the Y33 print service can be configured. Refer to 3.1.2 |
| 1.5 | 2019-9-5 | | Sun Chao Yang Xuezhi |  | 1. Add soft flow control configuration. For details, please see Obtaining Serial Printer Entity Class Object |
| 1.6 | 2020-04-16 | | Sun Chao,  Yang Xuezhi |  | 1. The zxing package name conflicts, and multiple users feedback conflicts. Replaced with com.printer.sdk.  zxing package name to solve related problems.  2. The problem of converting the printed image to a monochrome image is missing or excessive.  3. Added an interface for querying the printer paper distance and an interface for resetting the printer paper distance. |

**Content**

[Content 8](#_Toc23360)

[1.Introduction 11](#_Toc15604)

[2. Nouns Explanation 11](#_Toc18678)

[2.1 Nouns Explanation 11](#_Toc18502)

[2.2 Relevant Knowledge 12](#_Toc31811)

[2.3 This document speed guide 12](#_Toc14004)

[3. Related Class Explanation 13](#_Toc32663)

[3.1. Printer Core Class (PrinterInstance) 13](#_Toc2550)

[3.1.1 Get Printer Instance Class 13](#_Toc4782)

[3.1.2 Printer Communication Related Method 15](#_Toc15415)

[3.1.3 Setting Printer Method 17](#_Toc763)

[3.1.4 ESC/POS instruction print related methods 19](#_Toc14476)

[3.1.5 CPCL Command Label Printing Related Method 23](#_Toc20083)

[3.1.6 TSPL Command Label Printing Related Method 28](#_Toc8973)

[3.1.7 Method to control printer hardware 41](#_Toc28161)

[3.1.8 Get the current state of the printer 42](#_Toc20428)

[3.1.9 The call examples of connecting printers 44](#_Toc19362)

[3.2 Barcode Printing Class 46](#_Toc30957)

[3.2.1 constructor function 46](#_Toc16726)

[3.2.2 Barcode Printing call example 47](#_Toc7983)

[3.3 Table Form Printing Class 47](#_Toc10392)

[3.3.1 Create Table examples 47](#_Toc2802)

[3.3.2 Add One Row Data 48](#_Toc29136)

[3.3.3 Setting the column data alignment in the Table 48](#_Toc13924)

[3.3.2 Add One Row Data 48](#_Toc21588)

[3.3.3 Setting the column data alignment in the Table 49](#_Toc29168)

[3.3.4 Form printing call example 49](#_Toc17287)

[3.4 Canvas Printing Class 49](#_Toc23921)

[3.4.1 Initialization canvas 50](#_Toc31189)

[3.4.2 Set Font Property 50](#_Toc12439)

[3.4.3 Canvas content related method 51](#_Toc4948)

[3.4.4 Call sample for canvas printing 52](#_Toc5167)

[3.5 BitmapConvertor class 53](#_Toc24554)

[3.5.1Transfer colorful image to Black&White binary bitmap 53](#_Toc2269)

[3.6 Correlation Class of Printing PDF file 54](#_Toc13243)

[3.6.1 Calling example for PDF printing 54](#_Toc1969)

[3.7 CodePage Printer multilingual code page Print class 55](#_Toc32017)

[3.8 Printing interfaces of CanvasDrawPrint under Page mode 57](#_Toc15449)

[3.8.1 Initialization page 57](#_Toc7056)

[3.8.2 Page content related method 58](#_Toc21738)

[3.8.3 Page mode interface related example implementation 63](#_Toc26582)

[3.9 Record Log saved to SD card 66](#_Toc17441)

[3.9.1 Examples 66](#_Toc8125)

[3.10 Capture printing function of Y33 67](#_Toc27474)

[4. Appendix 72](#_Toc6148)

[4.1 Summarize of CODE128 72](#_Toc9103)

[4.2 Character Set 73](#_Toc19512)

# 1.Introduction

Android-SDK5.6 is a set of Android function interfaces developed based on our printer. Designed to help customers use our printers more conveniently. Note that this version no longer supports SDK4.0 and earlier, and is compatible with 5.0 or later. If there are software problems when using SDK4.0 and earlier versions, please consult our pre-sales company. This SDK supports Bluetooth, serial port, WIFI, and USB communication methods. The related class library is described as follows:

|  |  |
| --- | --- |
| **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 singel-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.

|  |
| --- |
| Get Printer Instance Class |
| Printer Communication Related Method |
| Setting Printer Method |
| Printing Content Related Method |
| CPCL Command Label Printing Method |
| TSPL Command Label Printing Method |
| Control Printer Hardware Related Method |
| Get Printer Status Related Method |
| Connect Printing Invocation Example |

### 3.1.1 **Get Printer Instance Class**

|  |  |
| --- | --- |
| **Method name** | **Description** |
| [getPrinterInstance](#获取蓝牙打印机实体类对象) | Getting Bluetooth Printer Instance Class |
| [getPrinterInstance](#获取USB打印机实体类对象) | Getting USB Printer Instance Class |
| [getPrinterInstance](#获取wifi打印机实体类对象) | Getting WiFi Printer Instance Class |
| [getPrinterInstance](#获取串口打印机实体类对象) | Getting Serial Printer Instance Class |
| [getPrinterInstance](#获取串口打印机实体类对象) | Getting AIDL 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 getPrinterInstance(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](/D//Youdao/Dict/6.3.69.5012/resultui/frame/javascript/void(0);) are used to received connect states of printer, whose corresponding states are as below:

Note:

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

* Getting AIDL Instance Class

PrinterInstance static synchronized getPrinterInstance()

Parameter: None

Returned Value:PrinterInstance Instance Class object

### 3.1.2 Printer Communication Related Method

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| [OpenConnection](#连接打印机) | Connecting To Printer |
| [closeConnection](#关闭与打印机的连接) | Close the Connection To Printer |
| [sendBytesData](#向打印机发送十六进制数据) | Sending [Hexadecimal](/D//Youdao/Dict/6.3.69.5012/resultui/frame/javascript/void(0);) Data To Printer |
| [read](#读取打印机返回的数据) | Read data returned from printer |
| [isSetAble](#设置远程打印服务是否可配置) | Set whether the remote print service is configurable (Only Y33 model) |
| [getSetable](#获取远程打印服务是否可配置的状态) | Get the status of the remote print service configurable(Only Y33 model) |

* C**onnecting 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:

> 0 Bytes numbers which are sent to printer successfully;

-1 means no initialize printing,

-2 means srcData is blank or srcData doesn’t have any data.

-3 Failed to send data to the printer

* 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:

> 0 Bytes numbers which are sent to printer successfully;

-1 means no initialize printing,

-2 means srcData is blank or srcData doesn’t have any data.

* Set whether the remote print service is configurable

Method Name: public void isSetAble(boolean isSetable)

Method Description: Set whether the remote print service is configurable (this method can only be used with the print service on the Y33 printer type. No meaning in other application scenarios.)

Parameters: true means that the remote print service can be configured, false means that the remote print service cannot be configured.

No returned value

* Get the status of the remote print service configurable

Method Name: public boolean getSetable()

Method Description: Set whether the remote print service is configurable (this method can only be used with the print service on the Y33 printer type. No meaning in other application scenarios.)

Parameters: Null

Return Value: true indicates that the remote print service is currently in a configurable state, false indicates that the remote print service is currently in an unconfigurable state.

### 3.1.3 Setting Printer Method

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

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| [initPrinter](#初始化打印机) | Initialize Printer |
| [setFont](#设置打印机字体) | Setting Printer Font |
| [setPrinter](#设置打印机打印) | Printer Printing Setting |
| [setLeftMargin](#设置距离左边距点数) | 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 inchs)

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

Parameter:inchs inchs numbers away from left margin

Returned Value: None

**E.g. 1:** setLeftMargin**8 inchs**

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** |
| [printText](#打印文字) | Printing Characters |
| [printImage](#打印图片) | Print Black and white binary picture |
| [printImage](#打印图片) | Print colorful image |
| [printColorImg2Gray](#printColorImg2Gray) | Print Grayscale color image |
| [printTable](#打印表格) | Print Table |
| [printBarCode](#打印条码) | Print Barcode |
| [printhorizontalLine](#打印水平直线，虚线) | 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\r\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, pleas 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 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 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, pleas 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 PAlign. 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** |
| [pageSetup](#设置要打印区域的页宽和页高) | Setting the page width and page height of printing area |
| [drawLine](#打印直线) | to print straight line |
| [drawBorder](#打印框体) | To print border |
| [drawText](#打印文本text) | To print text |
| [drawText](#区域中打印文字) | Print text in area |
| [drawBarCode](#打印一维条码) | Print 1D barcode |
| [drawBarCode](#区域中打印一维条码) | Print 1D barcode in area |
| [drawQrCode](#打印二维码) | Print 2D barcode |
| [drawGraphic](#打印位图) | Print bitmap |
| [drawGraphic](#区域内打印bitmap位图) | Print bitmap in area |
| [print](#打印) | 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 Yof the ending position

isSolidLinetrue 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

bold Bold or not 1 Bold 0 Not bold

reverse reverse or not 1 reverse 0 not reverse

underline 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:

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

fontSize refer to above method description to drawText:

bold Bold or not 1 Bold 0 Not bold

reverse reverse or not 1 reverse 0 not reverse

underline underline or not 1underline 0 not underline

deleteline deleteline or not 1deleteline 0 not deleteline

rotate 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** |
| [pageSetupTSPL](#设置页宽和页高TSPL) | Set page width and height |
| [drawLineTSPL](#画线TSPL) | Draw line |
| [drawBorderTSPL](#打印矩形框体TSPL) | To print border |
| [drawTextTSPL](#打印文字TSPL) | To print text |
| [drawTextTSPL](#区域中打印文字TSPL) | To print text in area |
| [drawBarCodeTSPL](#打印一维码TSPL) | To print 1D barcode |
| [draw2DBarCodeTSPL](#打印二维码TSPL) | To print 2D barcode(TL21type not support DMATRIX command) |
| [drawBitmapTSPL](#打印位图TSPL) | To print bitmap |
| [drawBitmapTSPL](#区域内打印图片TSPL) | To print bitmap in area |
| [printTSPL](#打印影像缓冲区内容TSPL) | Print images content in buffer |
| [getPrinterStatusTSPL](#查询打印机状态TSPL) | To get printer status |
| [setCharsetNameTSPL](#设置字符集编码格式TSPL) | Set the character set encoding format |
| [setPaperbackOrPaperFeedTSPL](#控制进纸或退纸TSPL) | Control the paper feeding or back |
| [reverseAreaTSPL](#指定区域反相打印TSPL) | reverse Print in area |
| [eraseAreaTSPL](#清除影像缓冲区数据TSPL) | Clear image buffer data |
| [setPrinterTSPL](#设置打印机相关功能TSPL) | Setting printer related function(TL21printer not support SPEED, BLINE, SHIFT, REPRINT command, TL51not support PEEL command) |
| [openCashBoxTSPL](#开钱箱TSPL) | Open cash drawer |
| [getPrinterNameTSPL](#查询打印机型号TSPL) | Checking the printer model |
| [setGAPTSPL](#设置标签纸间垂直间距TSPL) | setting the vertical spacing between labels |
| [selectCodePageTSPL](#选择字符代码页TSPL) | select codePage |
| [selectCountryTSPL](#选择国际字符集TSPL) | Select International character set(TL21model not support this method) |
| [beepTSPL](#控制蜂鸣器响一声TSPL) | Control buzzer beep one sound |
| [downloadBitmap2PrinterTSPL](#下载位图到打印机TSPL) | Download bitmap to printer(TL21model not support this method) |
| [putBitmapTSPL](#将已下载位图放置到打印缓冲区中TSPL) | Put downloaded bitmap into printer buffer(TL21does not support shi method) |
| [printSelfTestTSPL](#打印机打印自检页TSPL) | Print selftest paper(TL21does not support this method) |
| [setLabelReferenceTSPL](#设置标签内容的参考坐标原点TSPL) | Set reference origin of the label content |
| [sendStrToPrinterTSPL](#像打印机发送指令格式的字符串TSPL) | 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 rotate180 degree;

PRotate.Rotate\_90 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:

PPBarcodeType. 128 128 code

PPBarcodeType. EAN128 EAN128

PPBarcodeType. 39 39 code

PPBarcodeType. 93 93 code

PPBarcodeType. EAN13 EAN13 code

PPBarcodeType. 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<=wideWidth/(narrowWidth+1)<5

20<= wideWidth/(narrowWidth+1)<=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,ReadExcetion, 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 heigth)

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 let 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 heigth)

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≤value≤1000 means dots.

CommandTSPL .REPRINT Run reprinting order when error; Close this funtion 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=<value<=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 unction 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≦m≤1(inch), 0≦m≤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 PrinterPorNullException,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** |
| [cutPaper](#切纸) | Cut paper |
| [ringBuzzer](#控制蜂鸣器响) | Control buzzer beep |
| [blackLableFind](#黑标定位功能) | Black mark location |
| [resetFeedDistance](#重置打印机走纸距离) | Reset printer paper distance |

* 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

* Reset the paper distance of the printer

Method description: public void resetFeedDistance()

Method description: reset the printer paper distance (supported by specific models)

Parameter description: none

Return value: 0 successfully reset

-1 Reset failed

### 3.1.8 Get the current state of the printer

|  |  |
| --- | --- |
| Method | Description |
| [getCurrentStatus](#获取打印机当前状态) | Get printer current status |
| [getPrintingStatus](#打印机打印是否完成) | Printing finished or not |
| [isPaperOut](#判断打印机是否缺纸) | Tell paper out or not |
| [isPaperWillOut](#判断打印机是否纸将尽) | Tell printer paper near end or not |
| [getFeedDistance](#获取打印机总计走纸距离) | Get the total paper distance of the printer |

* 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.

* Get the total paper distance of the printer

Method name: public int getFeedDistance()

Method description: query the total paper distance of the current printer (supported by specific printers)

Parameter description: None

Return value: the total paper feeding distance of the current printer (unit: mm), a return value <0 means the query is abnormal.

### 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

BluetoothDevicemDevice = 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 font9\*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, "printerstatusisnotnormal! ", 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

|  |
| --- |
| constructor function |
| Barcode Printing call example |

### 3.2.1 constructor function

|  |  |
| --- | --- |
| **Method** | **Description** |
| [Barcode](#创建Barcode实例) | 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<=n<=6, default is 2

param2: barcode height 1<=n<=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<=n<=30.

param2: indicates Error Correction Level, 0<=n<=8 0<=n<=8.

param3: indicates longitudinal magnification.

2. DATAMATRIX

param1: indicate graph height, 0<=n<=144(0:automatic selection).  
param2: indicate graph width, 8<=n<=144(param1 is 0, invalid).  
param3: indicates Longitudinal magnification.

3. QRCODE

param1: indicates graphic version NO., 1<=n<=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

|  |
| --- |
| Create Table examples |
| Add One Row Data |
| Setting the column data alignment in the Table |
| Form printing call example |

### 3.3.1 Create Table examples

|  |  |
| --- | --- |
| **Method** | **Description** |
| [Table](#Table方法) | 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** |
| [addRow](#addRow) | 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** |
| [setColumnAlignLeft](#setColumnAlignLeft) | 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** |
| [addRow](#addRow) | 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** |
| [setColumnAlignLeft](#setColumnAlignLeft) | 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 |

### 3.4.1 Initialization canvas

|  |  |
| --- | --- |
| **Method** | **Description** |
| [init](#init方法) | Initialization canvas |

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

### 3.4.2 Set Font Property

|  |  |
| --- | --- |
| **Method** | **Description** |
| [setFontProperty](#setFontProperty) | set Font Property |

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

Note: Calling this method requires to explicit instantiation FontProperty , then

call FontProperty Class

setFont( boolean bBold,boolean bItalic,boolean bUnderLine,boolean bStrikeout,int iSize,Typefaces Face)

setFont the order method parameters:

bBold true Bold; false normal font, Note: if Set Chines is Bold separately,it will not print.

bItalic true italic, false normal font

bUnderLine true underline,false no underline

bStrikeout true strikeout,false no strikeout

iSize font size (value is an integer)

bItalic true italic, false normal font

sFace font type (generally set to null, means using the system default font)

If not this method, it can be set separately as below:

setLineWidth(floatw); Painting Line Width

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 |
| [drawText](#画布画字符串) | Draw string |
| [drawLine](#画布绘制直线) | Draw Line |
| [drawRectangle](#画布绘制矩形) | Draw Rectangle |
| [drawEllips](#画布绘制椭圆) | Draw ellipse |
| [drawImage](#画布绘制图片) | Draw image |
| [getCanvasImage](#画布获得画布上绘制的图像) | Getting the image drew on canvas and sending to the printer to print. |
| [setTextAlignRight](#画布设置文本是否靠右) | Set whether the text is right align |
| [setTextExceedNewLine](#画布文本超出是否换行) | Set whether entering new line when text exceeds. |
| [setUseSplit](#画布文本超出后换行是否使用分隔字符串) | 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. |
| [setUseSplitAndString](#画布文本超出后换行是否使用分隔字符串) | 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:publi void drawLine(float startX, float startY, float stopX, float

stopY);

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

coodinates

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=**new**FontProperty();

// 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.

|  |
| --- |
| [Transfer colorful image to Black&White binary bitmap](#彩色图转换为黑白二值bitmap) |

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

|  |  |
| --- | --- |
| Method | Description |
| [convertBitmap](#convertBitmap方法) | 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.

|  |
| --- |
| Open PDF file |
| Getting one page object of PDF file |
| PDF is transferred to Bitmap |
| Calling example for PDF printing |

### 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 abovePDF 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 th reading.

//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 |

### 

### 3.8、 Printing interfaces of CanvasDrawPrint under Page mode

### Brief description: The main definition page mode print interface is implemented by the host computer platform software. The way is to build a bitmap in the memory, the user completes the page layout through the interface, and finally calls the print interface when it is in memory. The bitmap data is constructed, and the data is sent to the printer by means of driving printing to complete the printing of the page. The failure of constructing the bitmap will be prompted in the interface return of the setCanvasDrawBack. For details, please see setCanvasDrawBack, and the drawing unit of the canvas is dot ( point).

Call printImage in 2.1.4 to finish printing the canvasdraw. The main methods are as follows:

|  |
| --- |
| Initialization page |
| Page content related method |
| Page mode interface related example implementation |

### 

### 3.8.1 Initialization page

|  |  |
| --- | --- |
| Method | Description |
| [SPRTSetPageMode](#SPRTSetPageMode) | initialize canvasdraw |

Method Name: public void SPRTSetPageMode(int width, int height)

Description: initialize the page

Parameters specification: width is the page width

Height is the page height

Returned data: Null

Exception: The establishment of the page mode is subject to the test device memory size limit. If the drawing is too large, you need to capture the exception yourself.

### 3.8.2 Page content related method

|  |  |
| --- | --- |
| Method | Description |
| [SPRTPrintLine](#页面上绘制直线) | Draw straight lines |
| [SPRTPrintBox](#页面上绘制矩形) | Draw rectangle |
| [SPRTPrintText](#页面上绘制文字) | Draw text |
| [SPRTPrintPDFCode](#页面上绘制专业版PDF417) | Draw professional edition PDF417 2D barcode |
| [SPRTPrintPDFCode](#页面上绘制普通版PDF417二维码) | Draw ordinary edition PDF417 2D barcode |
| [SPRTPrintQRCode](#页面上绘制专业版QRCode二维码) | Draw professional edition QRCode 2D barcode |
| [SPRTPrintQRCode](#页面上绘制普通版QRCode二维码) | Draw ordinary edition QRCode 2D barcode |
| [SPRTPrintCode](#页面上绘制一维码) | Draw 1D code |
| [SPRTPrintPicture](#页面上绘制Bitmap) | Draw Bitmap |
| [getCanvasImage](#获得页面上绘制的图像) | Get the image drawn on the page and send it to the printer to print |
| [setPaintStyle](#设置画笔空心还是实心) | Set the brush hollow or solid Default: solid |
| [setTextExceedNewLine](#文本超出是否换行) | Whether the text exceeds the line break |
| [setTextAlignRight](#设置页面上文本是否靠右) | Set whether the text is right |
| [setUseSplit](#页面文本超出后换行是否使用分隔字符串) | Whether the line breaks after the text exceeds the use of a separator string prevents a word from being split. Split by space (separator = space). |
| [setCanvasDrawBack](#监听绘制失败) | Check failure of drawing |
| [getLength](#返回当前页面绘制的剩余高度) | Returns the remaining height of the current page drawing, which is convenient for calculating the coordinates of the next element.  For details, please see the 3.83 page mode interface code example. |

* Draw straight line. Method name: SPRTPrintLine(int X\_Start, int Y\_Start, int X\_End, int Y\_End, int L\_Width)

Description:

1. X\_Start X the coordinate value of the starting point of the print line
2. Y\_Start Y the coordinate value of the starting point of the print line
3. X\_End X the coordinate value of the starting point of the print line
4. Y\_End Y the coordinate value of the starting point of the print line
5. L\_Width Line width

Returned value: Null

Abnormal: None

* Draw rectangle. Mehtod name: public void SPRTPrintBox(int X\_Start, int Y\_Start, int X\_End, int Y\_End, int L\_Width);

Description:

1. X\_Start the coordinate value of the starting point of the print line
2. Y\_Start the coordinate value of the starting point of the print line
3. X\_End the coordinate value of the starting point of the print line
4. Y\_End the coordinate value of the starting point of the print line
5. L\_Width Line width

Returned value: Null

Abnormal: None

* Draw text. Mehtod name: public void SPRTPrintText(int xPos, int yPos, String font, int font\_size, boolean bold, boolean underline, float italic, String content, Context mContext)

Description:

1. xPos The X coordinate vale of the string
2. yPos The Y coordinate vale of the string
3. font The font used, for example: "heiti.ttf" (the font file must be placed in the assets directory) Default: Pass null as the default font
4. font\_size size of font
5. bold bold or not boolean
6. underline underline or not boolean
7. italic Whether italic (positive number indicates tilt to the left, negative number indicates tilt to the right, normal between -1~1)
8. content Content of the string
9. mContext context

Returned value: Null

Abnormal: None

* Draw professional edition PDF417 2D barcode. Mehtod name: public void SPRTPrintPDFCode(int xPos, int yPos, int unit\_width, int unit\_height, int rows, int columns, int err\_level, String content)

Description:

1. xPos The starting x coordinate of the QR code
2. yPos The starting Y coordinate of the QR code
3. unit\_width width (the width of the 2D QR code, related to rows, columns, err\_level which can affect the value)
4. unit\_height Height(the width of the 2D QR code, related to rows, columns, err\_level which can affect the value)
5. rows barcode lines, range 1-90
6. columns column number of columns, range 1-30, if rows are specified, then columns must also be specified
7. err\_level Error correction level, range 1-8
8. content contect of barcode

Returned value: Null

Abnormal: None

* Draw ordinary edition PDF417 2D barcode. Mehtod name: public void SPRTPrintPDFCode(int xPos, int yPos, int width, int height, String content)

Description:

1. xPos The starting x coordinate of the QR code
2. yPos The starting Y coordinate of the QR code
3. width
4. height （Most of the PDF417 is rectangular, and is scaled to fit the aspect ratio.）
5. content contect of barcode

Returned value: Null

Abnormal: None

* Draw professional edition QRCode 2D barcode. Mehtod name: public void SPRTPrintQRCode(int xPos, int yPos, int module\_width, int module\_height, int white\_space, int version, int err\_level, String content)

Description:

1. xPos The starting x coordinate of the QR code
2. yPos The starting Y coordinate of the QR codemodule\_width
3. module\_width Width (the width of the QR code, related to the version, err\_level parameters can affect the value)
4. module\_height Height (the width of the QR code, related to the version, err\_level parameters can affect the value)
5. white\_space The width of the blank area around the QR code
6. version QR version, range 1-40
7. err\_level Error correction level, range 0-3
8. content contect of barcode

Returned value: Null

Abnormal: None

* Draw ordinary edition QRCode 2D barcode. Mehtod name: public void SPRTPrintQRCode(int xPos, int yPos, int width, int height, String content)

Description:

1. xPos The starting x coordinate of the QR code
2. yPos The starting x coordinate of the QR code
3. width
4. height
5. content contect of barcode

Returned value: Null

Abnormal: None

* Draw 1D code . Mehtod name: public void SPRTPrintCode(int codeType, int xPos, int yPos, int width, int height, boolean show\_text, String content)

Description:

1. codeType The type of one-dimensional code, currently supports the following bar code class（such as : BarcodeCreater.CODE39）
2. ITF
3. CODE39 Max data length 75
4. CODE128 Max data length 108
5. EAN13
6. EAN8
7. UPC-A
8. UPC-E
9. CODABAR
10. CODE\_93
11. xPos The starting x coordinate of the one-dimensional code
12. yPos The starting Y coordinate of the one-dimensional code
13. width
14. height
15. show\_text show text or not
16. content data of 1D code

Returned value: Null

Abnormal: None

* Draw Bitmap. Mehtod name: public void SPRTPrintPicture(int xPos, int yPos, Bitmap bitmap)

Description:

1. xPos Starting x coordinate
2. yPos Starting Y coordinate
3. Bitmap content

Returned value: Null

Abnormal: None

* Check failure of drawing. Mehtod name: public void setCanvasDrawBack(CanvasDrawBack mCanvasDrawBack)

CanvasDrawPrint returned data specification:

int LINE = 1; Line drawing failed

int BOX = 2; Rectangle drawing failed

int TEXT = 3; Text drawing failed

int PDF417\_PROFESSIONAL = 4; PDF417 professional edition drawing failed

int PDF417\_ORDINARY = 5; PDF417 ordinary edition drawing failed

int QRCODE\_PROFESSIONAL = 6; QRCODEprofessional edition drawing failed

int QRCODE\_ORDINARY = 7; QRCODE ordinary edition drawing failed

int ONE\_DIMENSION\_CODE = 8; 1D code drawing failed

int IMG = 9; bitmap drawing failed

For detailed error information, please check the related LOG.

Returned value: Null

* Set the brush hollow or solid Default: Solid

setPaintStyle(Paint.Style style)

Exception: If the page is not initialized, it will cause a null pointer exception (initialization can be ignored), you need to catch the exception yourself.

* Get the image drawn on the page and send it to the printer for printing.

getCanvasImage();

* Set whether the text on the page is right

setTextAlignRight(boolean alignRight);

* The text exceeds whether or not to wrap.

setTextExceedNewLine(boolean newLine);

Note: More than one line break or start drawing the next line based on the X axis

* If the page text exceeds the line break, use a separator string to prevent a word from being split. Split by space. (separated string = space)

mCanvasDrawPrint.setUseSplit(boolean useSplit)

* Returns the remaining height drawn by the current page.

getLength();

Note: This function returns the lower position of the highest bit currently drawn, which is convenient for level drawing.

**3.8.3 Page mode interface related example implementation**

// Create a canvas

CanvasDrawPrint mCanvasDrawPrint = new CanvasDrawPrint();

// Define drawing failure listeners

mCanvasDrawPrint.setCanvasDrawBack(mCanvasDrawBack);

// Initialize the canvas width, height mCanvasDrawPrint.SPRTSetPageMode(500, 800);

mCanvasDrawPrint.setTextAlignRight(true);//Set alignment the way mCanvasDrawPrint.setUseSplit(true); Whether the line breaks after the text on the page exceeds the delimited string//

// Draw text

mCanvasDrawPrint.SPRTPrintText(0, 0, null, 60, true, true, 0.5f, "Test font multi-text test text wrap problem", mContext);

mCanvasDrawPrint.SPRTPrintText(0, mCanvasDrawPrint.getLength(), "kaiti.ttf", 60, false, false, 0, "Carcass text", mContext);

//Draw straight lines

mCanvasDrawPrint.SPRTPrintLine(0, mCanvasDrawPrint.getLength(), 450, mCanvasDrawPrint.getLength(), 10);

//Set brush style

mCanvasDrawPrint.setPaintStyle(Paint.Style.STROKE);

//Draw professional edition PDF417

mCanvasDrawPrint.SPRTPrintPDFCode(0, mCanvasDrawPrint.getLength(), 500, 500, 23, 8, 3, "123456789");

//Draw professional editionQRCode

mCanvasDrawPrint.SPRTPrintQRCode(0, mCanvasDrawPrint.getLength(), 200, 200, 10, 10, 4, "123456");

// Draw 1D code

mCanvasDrawPrint.SPRTPrintCode(BarcodeCreater.CODE\_128, 0, mCanvasDrawPrint.getLength(),300,200, true, "123456");

//Save the canvas as a picture and print it

mPrinter.printImage(mCanvasDrawPrint.getCanvasImage(), PAlign.*NONE*, 0);

CanvasDrawPrint.CanvasDrawBack mCanvasDrawBack = new CanvasDrawPrint.CanvasDrawBack() {

@Override

public void resultsBack(int i) {

String str = "";

if (i == CanvasDrawPrint.LINE) {

str = ("Straight line drawing erro refer to related log");

} else if (i == CanvasDrawPrint.BOX) {

str = ("rectangle drawing erro refer to related log");

} else if (i == CanvasDrawPrint.TEXT) {

str = ("Text drawing erro refer to related log");

} else if (i == CanvasDrawPrint.PDF417\_PROFESSIONAL) {

str = ("professional edition PDF417 drawing erro refer to related log");

} else if (i == CanvasDrawPrint.PDF417\_ORDINARY) {

str = ("Regular edition PDF417 drawing erro refer to related log");

} else if (i == CanvasDrawPrint.QRCODE\_PROFESSIONAL) {

str = ("professional edition 2D barcode drawing erro refer to related log");

} else if (i == CanvasDrawPrint.QRCODE\_ORDINARY) {

str = ("Regular edition 2D barcode drawing erro refer to related log");

} else if (i == CanvasDrawPrint.ONE\_DIMENSION\_CODE) {

str = ("1D code drawing erro refer to related log");

} else if (i == CanvasDrawPrint.IMG) {

str = ("bitmap drawing erro refer to related log");

}

Toast.makeText(mContext, str, 0).show();

}

};

Note: For more detailed page mode print calls, see the btn\_page\_mode\_printingt print page mode click event in PicturePrintActivity.java.

## 3.9、Record Log saved to SD card

Brief description: It is mainly convenient for debugging and finding bugs. Users can export the log files in the SD card to facilitate the analysis of our technicians. Remember that the bugs provided at the time can provide relevant logs to solve the bugs efficiently. Need to call ZLog.Init (); to initialize, the default is open after the initialization is completed, the method is a constant mark, each time you need to reset the switch, each log file saved is up to 2Mb, more than 2Mb file name is accumulated 1, A total of 9 files are generated, and when the 9 files are full, the previous files are replaced in sequence. If you encounter problems, you can send the set folder directly to us.

|  |  |
| --- | --- |
| Method | Description |
| [ZLog.Init()](#注册log工具。); | Register the log tool class and pass in the path |
| [ZLog.openSaveToFile()](#打开记录log。) | Open log record |
| [ZLog. closeSaveToFile ()](#关闭记录log。) | Close log record |

* Register the log tool. Mehtod name: public static synchronized void Init(String logDir)

Description:

logDir saves the log path

Returned value: Null

Abnormal: None

* Open log record. Mehtod name: public static synchronized void openSaveToFile()

Description: Null

Returned value: Null

Abnormal: None

* Close log record. Mehtod name: public static synchronized void closeSaveToFile()

Description:Null

Returned value: Null

Abnormal: None

**3.9.1 Examples**

//Initialize ZldLog should call the next method before use

ZLog.Init(Environment.getExternalStorageDirectory()

.getAbsolutePath() + "/SPRT\_LOGS/");

// open the record

ZLog.openSaveToFile()

// close the record

ZLog. closeSaveToFile ()

## 3.10 Capture printing function of Y33

The capture and print function is only available for Android applications based on the Y33 all-in-one.

Capture print schematic diagram:

Third party application

Custom application

Printing SDK

Ethernet

Printing service

BT

usb1

RS232

usb2

POS

Summary of the collection and printing functions: The Y33 all-in-one provides a print service for parsing and processing data sent from the POS (Usb or Serial) or third-party applications (network or Bluetooth). By parsing the data, the print data is sent to the printer and the data to be collected is uploaded to the Android custom application. The Android custom application developed based on the Y33 all-in-one can detect the printing action (before printing, after printing) and decide whether to append data or perform recording operations. Android custom applications can also collect print data uploaded by the Y33 all-in-one.

Print Service Configuration Description: In the configuration interface of the print service, you can configure the collection mode, capture port, whether to print, whether to notify the Android custom application before printing, and whether to notify the Android custom application and other configuration options after printing. As follows:

Acquisition method: You can choose to collect drive data in drive mode or choose to collect common data in sdk mode.

Acquisition port: You can choose to collect data transmitted through the network port or data transmitted through the USB port or data transmitted via Bluetooth. (The three can only collect one at a time, not concurrently.)

Whether to print: You can choose to collect only data or collect data and print.

Whether to pre-print notification: You can choose whether to notify the Android custom application before the printer prints a single data to decide whether to append the print header or do some recording operations.

Whether to notify after printing: You can choose whether to notify the Android custom application after the printer prints a single data to decide whether to append the print data tail or do some recording operations.

Turn off the service and start the service: By default, the print service is powered on. When an exception occurs, the data can be reacquired by shutting down and restarting the service.



Android application development process based on Y33 one machine:

1. Depend on printersdkv5.4.jar
2. In the configuration file AndroidManifest.xml, the application continues com.printer.sdk.application.MyApplication
3. Registered broadcast receiver in the configuration file AndroidManifest.xml com.receiver.sprt.yxz.printservice.client.receiver.PrintServiceBroadcastReceiver. And add below action:

<!-- PrintService sent broadcast usbep1 -->

<action android:name="com.sprt.yxz.printservice.service.PrinterService.usbep1\_start\_print"/>

<action android:name="com.sprt.yxz.printservice.service.PrinterService.usbep1\_end\_print" />

<action android:name="com.sprt.yxz.printservice.service.PrinterService.usbep1\_parser\_data" />

<!-- PrintService sent broadcast net -->

<action android:name="com.sprt.yxz.printservice.service.PrinterService.net\_start\_print" />

<action android:name="com.sprt.yxz.printservice.service.PrinterService.net\_end\_print" />

<action android:name="com.sprt.yxz.printservice.service.PrinterService.net\_parser\_data" />

<!-- PrintServicesent broadcast bluetooth -->

<action android:name="com.sprt.yxz.printservice.service.PrinterService.bt\_start\_print" />

<action android:name="com.sprt.yxz.printservice.service.PrinterService.bt\_end\_print" />

<action android:name="com.sprt.yxz.printservice.service.PrinterService.bt\_parser\_data" />

1. Development demonstration

4.1 Instantiate the PrinterInstance object

PrinterInstance myPrinter = PrinterInstance.getPrinterInstance();

4.2 Open the port for appending data

boolean isConnected = myPrinter.openConnection();

4.3 Monitor print action

myPrinter.setOnPrintListener(new OnPrintListener() {

@Override

public void doBeforePrint() {

XLog.i(TAG, "Client Appends the printhead before printing begins.");

try {

4.4 Listen to print start, append print header

if(myPrinter！=null)

int writeLen = myPrinter.sendBytesData("\*\*\*\*\*\*\*\*\*\*\*\*\*beginning data of one order \*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n".getBytes("gbk"));

} catch (UnsupportedEncodingException e) {

e.printStackTrace();

}

}

@Override

public void doAfterPrint() {

4.5 Listen to end of printing, append print data tail

XLog.i(TAG, "Client appends the end of the data after printing");

try {

if(myPrinter！=null)

int writeLen = myPrinter.sendBytesData("\*\*\*\*\*\*\*\*\*\*\*\*\*End data of one order\*\*\*\*\*\*\*\*\*\*\*\*\*\*\r\n".getBytes("gbk"));

} catch (UnsupportedEncodingException e) {

e.printStackTrace();

}

}

@Override

public void onReceiveParserData(int type, byte[] data) {

if (data == null || data.length == 0) {

XLog.i(TAG, "The uploaded data uploaded by the client is empty.");

return;

}

if (type == 0) //The client receives the collected data uploaded by the USB port.

if (type == 1) //The client receives the collected data uploaded by the NTE port.

if (type == 2) //The client receives the collected data uploaded by the BT port.

}

});

4.6 Close the port for appending data (Note: The closeConnection() method is paired with the openConnection() method. Please close the port after sending the append data).

myPrinter.closeConnection();

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

Character Set A

Character Set B

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

CharacterC

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