

# ZONERICH LABEL PRINTER

## Android SDK Help Doc V2.3

----- 2015.4.23

ZONERICH Label Printer Android SDK Help Doc V2.3 .....	1
1. Introduction.....	2
2. Functions.....	3
1.  BOOL prn_Connect(String strAddr) .....	3
2.  void prn_Disconnect().....	3
3.  void prn_PageSetup (int pageHeight, int pageWidth).....	4
4.  int prn_PagePrint(int rotate) .....	4
5.  void prn_PageClear() .....	4
6.  int prn_PrinterStop() .....	4
7.  void prn_DrawBox(int lineWidth, int x0, int y0, int x1, int y1) .....	5
8.  void prn_DrawLine(int lineWidth, int x0, int y0, int x1, int y1) .....	5
9.  void prn_DrawInverseLine(int x0, int y0, int x1, int y1) .....	5
10.  void prn_DrawText (int x, int y, String text, String fontName,int fontsize, int rotate, int bold, int underline, int reverse) .....	6
11.  void prn_DrawTextAlign (int x0, int y0, int x1, int y1, int nAlign, String text, String fontName,int fontsize, int rotate, int bold, int underline, int reverse).....	8
12.  int prn_DrawBarcode(int x, int y, String text, int barcodetype, int rotate, int linewidth, int height) ..	8
13.  void prn_BarcodeText (String fontName,int fontsize, int offset) .....	9
14.  void prn_PrintBitmap(int startx, int starty, Bitmap bmp) .....	10
15.  int prn_PrinterStatus().....	10
16.  int prn_AddLabelCommand(String strCmd) .....	10
3. Samples .....	11



**GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD**

Copyright © 2015 All Rights Reserved.

# 1. Introduction

## 1.1 Name: ZQLabelSDK

## 1.2 Develop IDE: Eclipse

## 1.3 Versions:

Version: V2.0

Date: 2015/2/9

Update: Distribution

Version: V2.1

Date: 2015/2/11

Update: add port function *prn\_PrinterStop()*

Version: V2.2

Date: 2015/4/3

Update: add port function *prn\_GetDeviceID()*

Version: V2.3

Date: 2015/4/23

Update: add functions *prn\_DrawInverseLine()*, *prn\_DrawBarcodeText()*, *prn\_DrawBox()*, *prn\_AddLabelCommand()*

## 1.4 Lib files:

zqlabelsdk.jar

armeabi\libserial\_port.so (Note: this file could be ignored if the RS232 and USB ports are not used.)

## 1.5 Lib files usage in Projects

1) Open a new Android project

2) Add file *zqlabel.jar* and directory *armeabi* in project *libs* directory (note: if there is no *libs* directory, make a new directory *libs*), then Refresh the project in Eclipse.



**GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD**

Copyright © 2015 All Rights Reserved.

3) Define a variable *private ZQLabelSDK = null*, and add statement *prn = new ZQLabelSDK(this)* in a proper position to invoke functions.

## 2. Funtions

### 2.1 BOOL prn\_Connect(String strAddr)

**Description:** to connect printer

**Parameter:**

String **strAdd**: the address of Bluetooth or COM port of the printer. The Bluetooth address is in format of MAC address, the COM port address is in format of "COM Port # : Baud Rate", for example: "ttyS1:115200"

**Returns:**

PrinterConst.ErrorCode.SUCCESS	0	Successfully opened
PrinterConst.ErrorCode.INVALIDPARAM	-5	Invalid printer parameter
PrinterConst.ErrorCode.PORTERROR	-1	Port error

**Note:**

- 1). The Bluetooth address and COM port baud rate could be found on printer self-testing receipt. To perform self-testing, press Feed button and power on the printer for several seconds till there is a receipt printed out.
- 2). Corresponding privileges of the Bluetooth function should be defined in *AndroidManifest.xml*, otherwise there will be failures.
- 3). The Bluetooth printer should be paired correctly with Android OS before any connection. The default password for pairing should be "000" or "1234".

### 2.2 void prn\_Disconnect()

**Description:** to disconnect the printer

**Parameter:** NULL

**Returns:** NULL

**Note:** n/a



GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD

Copyright © 2015 All Rights Reserved.

## 2.3 void prn\_PageSetup (int pageHeight, int pageWidth)

**Description:** to set the size of label page

**Parameter:**

int pageWidth: to define the width of label page

int pageHeight: to define the height of label page

**Returns:** NULL

**Note:** the maximum supported printing page is 576 x 2000dots. With 90 degrees rotated, it is 2000 x 576 dots.

## 2.4 int prn\_PagePrint(int rotate)

**Description:** to print label pages

**Parameter:**

int rotate: to set the rotate angle, values as below:

Value	Description
0	No rotate
1	90 degrees of clockwise rotating

**Returns:**

PrinterConst.ErrorCode.SUCCESS	0	Succeed
PrinterConst.ErrorCode.WRITEERROR	-2	Failed to send data to printer. The possibility of disconnection to printer

**Note:** n/a

## 2.5 void prn\_PageClear()

**Description:** to clear page, all configurations to be reset

**Parameter:** NULL

**Returns:** NULL

**Note:** n/a

## 2.6 int prn\_PrinterStop()

**Description:** to stop printing and clear buffer

**Parameter:** NULL

**Returns:**

PrinterConst.ErrorCode.SUCCESS	0	Succeed
--------------------------------	---	---------



GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD

Copyright © 2015 All Rights Reserved.

PrinterConst.ErrorCode.WRITEERROR	-2	Failed to send data to printer. The possibility of disconnection to printer
-----------------------------------	----	---

**Note:** n/a

## 2.7 void prn\_DrawBox(int lineWidth, int x0, int y0, int x1, int y1)

**Description:** to draw rectangles (boxes)

**Parameter:**

int lineWidth: to define the width of the drawing lines (unit: dots)

int x0: to define the x-coordinate of the start point

int y0: to define the y-coordinate of the start point

int x1: to define the x-coordinate of the end point

int y1: to define the y-coordinate of the end point

**Return:** NULL

**Note:** n/a

## 2.8 void prn\_DrawLine(int lineWidth, int x0, int y0, int x1, int y1)

**Description:** to draw lines

**Parameter:**

int lineWidth: to define the width of the drawing line (unit: dots)

int x0: to define the x-coordinate of the start point

int y0: to define the y-coordinate of the start point

int x1: to define the x-coordinate of the end point

int y1: to define the y-coordinate of the end point

**Returns:** NULL

**Note:** n/a

## 2.9 void prn\_DrawInverseLine(int x0, int y0, int x1, int y1)

**Description:** to draw blocks (generally for inverse printing)

**Parameter:**

int x0: to define the x-coordinate of the start point

int y0: to define the y-coordinate of the start point

int x1: to define the x-coordinate of the end point

int y1: to define the y-coordinate of the end point



GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD

Copyright © 2015 All Rights Reserved.

**Returns:** NULL

**Note:** the inverse printing should be like this: **SAVE**. To print like this, the first step should be to invoke function `prn_DrawInverseLine` to print out a whole block, then invoke `prn_DrawText` to print content (the inverse function should not be enabled in this invoking).

**2.10 void prn\_DrawText (int x, int y, String text, String fontName,int fontsize, int rotate, int bold, int underline, int reverse)**

**Description: to print text**

**Parameter:**

int x: to define the x-coordinate of the start point

int y: to define the y-coordinate of the start point

String text: the text content

String fontName: font names to refer below table.

int fontsize: font sizes to refer below table.

Character font sizes table reference:

fontName	fontsize	Width x Height	Description
0	0	8x16	fixed pitch dot font -- Dark
0	1	16x16	fixed pitch dot font -- Dark
0	2	8x32	fixed pitch dot font -- Dark
0	3	16x32	fixed pitch dot font -- Dark
0	4	24x32	fixed pitch dot font -- Dark
0	5	16x32	fixed pitch dot font -- Dark
0	6	32x48	fixed pitch dot font -- Dark
1	0	16x48	fixed pitch dot font -- <i>Italic</i>
2	0	16x16	fixed pitch dot font -- Light
4	0	24x48	fixed pitch dot font -- Light
4	1	24x72	fixed pitch dot font -- Light
4	2	48x48	fixed pitch dot font -- Light
4	3	24x72	fixed pitch dot font -- Light
4	4	24x72	fixed pitch dot font -- Light
4	5	24x72	fixed pitch dot font -- Light
4	6	24x72	fixed pitch dot font -- Light
4	7	24x72	fixed pitch dot font -- Light
5	0	12x24	fixed pitch dot font -- Dark
5	1	12x24	fixed pitch dot font -- Dark
5	2	24x48	fixed pitch dot font -- Dark



**GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD**

Copyright © 2015 All Rights Reserved.

5	3	24×48	fixed pitch dot font --深色
7	0	12×48	fixed pitch dot font --浅色
7	1	12×48	fixed pitch dot font --浅色
8	0	12×24	fixed pitch dot font --浅色
55	0	8×16	fixed pitch dot font --浅色

### Chinese character font sizes

fontName	fontsize	Width × Height	Description
55	The higher half of the byte represents for the times of width to zoom in The lower half of the byte represents for the times of height to zoom in	Chinese: 16×16 English: 8×8	fixed pitch dot font
24	The higher half of the byte represents for the times of width to zoom in The higher half of the byte represents for the times of height to zoom in	Chinese: 24×24 English: 12×12	fixed pitch dot font
32	The higher half of the byte represents for the times of width to zoom in The higher half of the byte represents for the times of height to zoom in	Chinese: 32×32 English: 16×32	

For example: fontsize=0x11, both width and height to be enlarged by 1 time. To double the width and height.

fontsize=0x10, to double the width. Height to be remained at original size.

fontsize=0x23, to enlarge the width to 3 times, to enlarge the height to 4 times.

int rotate: to set the rotating, values are as below:

Value	Description
0	No rotate
1	90 degrees at clockwise rotate
2	180 degrees at clockwise rotate
3	270 degrees at clockwise rotate

int bold: to set bold font, values from 0-5. 0 is for normal font.

int underline: to set underline, values are as below table,

Value	Description
0	No underline
1	Underlined by 1 dot width

int reverse: to set inverse printing, values are as below table,

Value	Description
0	No inverse (white background and black content)
1	Inverse printing (black background and white content)

Returns: NULL

Note: n/a



GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD

**2.11 void prn\_DrawTextAlign (int x0, int y0, int x1, int y1, int nAlign, String text, String fontName, int fontsize, int rotate, int bold, int underline, int reverse)**

**Description: to print text contents in a designated block space with certain alignment**

**Parameter:**

int x0: to define the x-coordinate of the start point

int y0: to define the y-coordinate of the start point

int x1: to define the x-coordinate of the end point

int y1: to define the y-coordinate of the end point

int nAlign: text alignment, values are as below:

Value	Description
0	Left alignment
1	Center alignment
2	Right alignment

String text: text content

String fontName: font name, refer to *prn\_DrawText*

int fontsize: font size, refer to *prn\_DrawText*

int rotate: to set rotate printing, refer to *prn\_DrawText*

int bold: to set bold font, values are from 0-5, 0 means normal font.

int underline: to set underline, refer to *prn\_DrawText*

int reverse: to set inverse printing, refer to *prn\_DrawText*

**Returns: NULL**

**Note:** the text alignment is only for horizontal directions. The vertical alignment could be fixed with y-coordinate pre-calculation.

**2.12 int prn\_DrawBarcode(int x, int y, String text, int barcodetype, int rotate, int linewidth, int height)**

**Description: to print barcode**

**Parameter:**

int x: to define the x-coordinate of the start point

int y: to define the y-coordinate of the start point

String text: the content of barcode

int barcodetype: to select barcode type, selections are as below table.

Value	Description
0	JAN13(EAN13)
1	JAN8(EAN8)
2	CODE39



**GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD**

Copyright © 2015 All Rights Reserved.



3	CODE93
4	CODE128
5	CODABAR
6	ITF
7	UPC-A
8	UPC-E
9	EAN13+2
10	EAN13+5
11	EAN8+2
12	EAN8+5
13	UPCA+2
14	UPCA+5
15	UPCE+2
16	UPCE+5
17	Postnet
18	MSI
19	QR Barcode
128	CODE128
39	CODE39
93	CODE93

int rotate: to set rotate printing. Refer to below table:

Value	Description
0	No rotate
1	90 degrees at clockwise rotate
2	180 degrees at clockwise rotate
3	270 degrees at clockwise rotate

int linewidth: Narrow barcode scale factor for 1D barcode (barcodetype!=19); QR code type for 2D barcode (barcodetype=19), values are 1 or 2.

int height: to define the barcode height (unit: dots) for 1D barcode (barcodetype!=19); to define the level of Width/Height for 2D barcode (barcodetype=19), values are from 1 to 32.

**Returns:** NULL

**Note:** n/a

## 2.13 void prn\_BarcodeText (String fontName,int fontsize, int offset)

**Description:** to start barcode text printing

**Parameter:**

String fontName: font name, refer to *prn\_DrawText*

int fontsize: font size, refer to *prn\_DrawText*

int offset: to define the space (offset) between barcode and text



**GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD**

Copyright © 2015 All Rights Reserved.

**Return:** NULL

**Note:** 1) The function should be invoked before *prn\_Barcode*

2) The barcode text is aligned centrally comparing to barcode by default

3) With invoking of this function, the text printing will be validated for all the current printing. To not print barcode text for a next barcode, *prn\_AddLabelCommand("BARCODE-TEXT OFF\r\n")* should be invoked before next printing.

## 2.14 void prn\_PrintBitmap(int startx, int starty, Bitmap bmp)

**Description:** to print Bitmap image

**Parameter:**

int startx: to define the x-coordinate of the start point

int starty: to define the y-coordinate of the start point

Bitmap bmp: image content of Bitmap

**Returns:** NULL

**Note:** n/a

## 2.15 int prn\_PrinterStatus()

**Description:** to fetch the status of printer

**Parameter:** NULL

**Returns:** -1: no response from printer

0: correct status from printer

>0: incorrect status from printer (Bit 0 represents for cover open; Bit 1 represents for no paper; Bit 3 represents for printer busy.)

**Note:** n/a

## 2.16 int prn\_AddLabelCommand(String strCmd)

**Description:** to send label printing commands

**Parameter:**

String strCmd: Label printing commands, to refer to corresponding command list manual.

**Returns:** NULL

**Note:** to accomplish special printing effects.



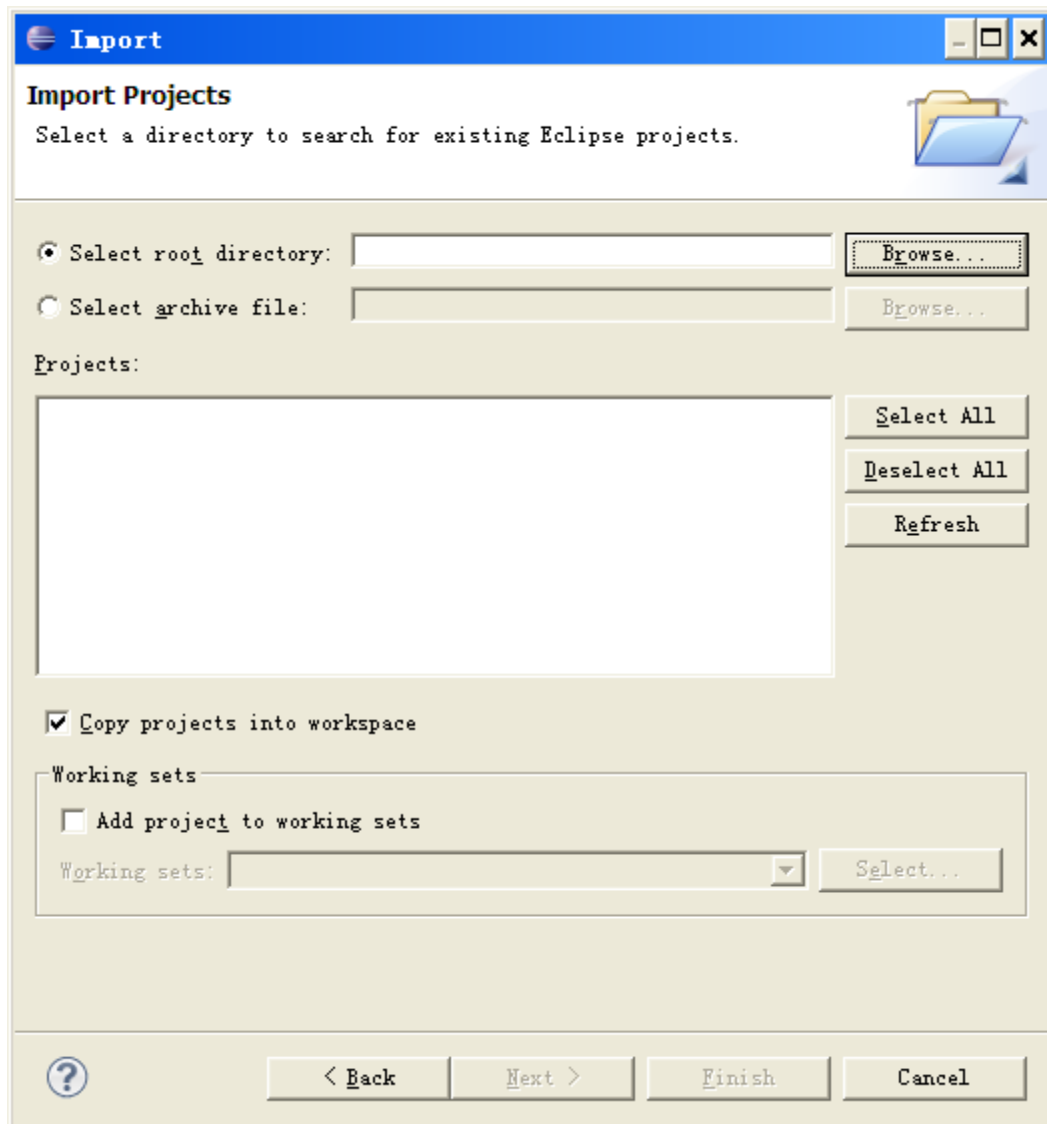
GUANGZHOU ZONERICH BUSINESS MACHINE CO LTD

Copyright © 2015 All Rights Reserved.

### 3. Samples

ZQLabelSample is for Eclipse environment.

In Eclipse, click File->Import->General->Existing Projects into Workspace



Click Browse and select ZQLabelSample folder, then click Finish button.