CoiNel Bluetooth Printer

Android SDK User Manual

Version: 1.0.2

CoiNel Bluetooth Printer - Android SDK User Manual

Table of Contents

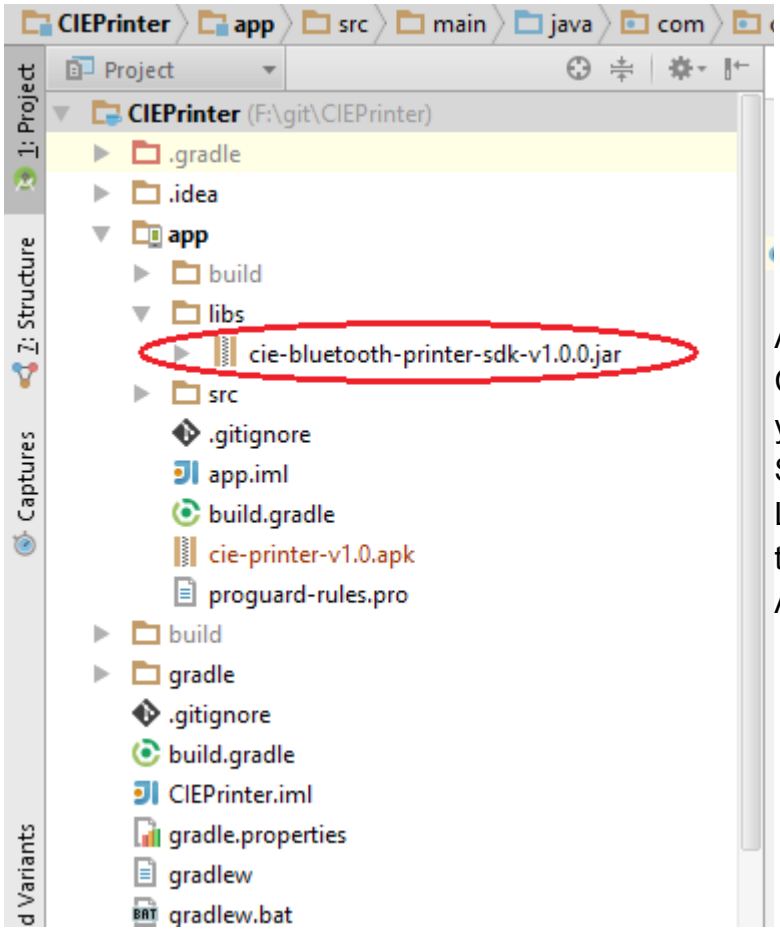
CoiNel Bluetooth Printer - Android SDK User Manual

Introduction

This document describes the android SDK for the COINEL Bluetooth printer. This SDK supports android version 8 (FROYO) and above. The COINEL Bluetooth printer SDK is a jar library that you have to add to your android project. The accompanying android demo project code shows how to use all the functionality of the COINEL Bluetooth printer.

Adding SDK to your android project in Android Studio

After creating your android project in android studio. Create a Libs folder, is not already present and copy the SDK Jar into that folder.



As shown in the image insert COINEL Bluetooth printer SDK in your android project. After inserting SDK right click and select “Add As Library”. Now your project is ready to use the COINEL Bluetooth printer Android SDK.

Using COINEL SDK in your android project

Using the Coinel Bluetooth SDK is very simple. The SDK exposes the Printer as a Singleton Class.

Here are steps described in brief for your easy understanding. please refer to the sample demo project in github for sample code implementation.

Your Projects Android Manifest file should contain permissions for **Bluetooth & Bluetooth Admin**.

In your base activity which will interact with the printer do the following

- override the following methods `onActivityResult, onResume, onPause & onDestroy.`
- Implement a message handler to receive messages from the printer.
- In the `onCreate` method, initialize the printer.

Once the printer is initialized, now you are ready to issue commands to the printer. You can call any of the printer methods that can send commands to the printer. The List of printer commands are given below

List of Printer Commands

Printer Selection

To select the printer pass the command according to the size of printer.

```
setPrinterWidth(PrinterWidth.PRINTER_WIDTH_48MM);

setPrinterWidth(PrinterWidth.PRINTER_WIDTH_72MM);

setPrinterWidth(PrinterWidth.PRINTER_WIDTH_104MM);
```

Print Line Feed

```
printLineFeed()
```

Makes the paper to move forward.

Print Reverse Line Feed

```
printReverseLineFeed()
```

Makes the paper to move backward.

To Print Text

```
printLineFeed(String txt)
```

```
Input: String text
```

Prints the given text, please note you can print a maximum of 42 characters on a single line. Please ensure that the string is not more than 100 bytes in length.

Horizantal Tab

```
setTab()
```

Moves the print position to the next horizontal tab position. The position of tab length is fixed.

To Batch Print Commands

```
setPrintMode(BtpConsts.PRINT_IN_BATCH)

batchPrint()
```

Print bulk of commands in a batch. Give *PRINT_IN_BATCH* constant to enable batch print and batchPrint() to print all commands which was batched.

Set Print Intensity

```
setLowIntensity()

setNormalIntensity()

setHighIntensity()
```

Intensity of font by using this commands we can change the Intensity of font as light, medium and bold.

Select Character Size

```
setFondSizeSmall()

setFondSizeMedium()

setFondSizeLarge()

setFondSizeXLarge()
```

By this commands wee can change the size of the fonts.

Select Justification

```
setAlignmentCenter()
```

`setAlignmentLeft()`

`setAlignmentRight()`

Aligns the data in a line centre, left and right by using this commands.

Select Font

`setStyleFixedsys()`

`setStyleCourier()`

`setStyleHindi()`

Selects the particular font type for text printing.

Test Print

`printerTest()`

Executes the test print command to print printer details

Reset Printer

`resetPrinter()`

It will reset all commands to default state

Paper Status

`getPaperStatus()`

Return the paper status from the printer while the paper is available or not.

Battery Status

`getBatteryStatus()`

Return the presentage of battery available in the printer

Platten Status

`getPlattenStatus()`

Return the Platten status if it is closed or not.

Print Image

```
boolean printDirect (String path, boolean invert, int threshold)
```

Inputs: image path, invert, threshold.

Returns: true when the image is printed successfully

Save Image

```
boolean saveImage (String path, boolean invert, int threshold, int id)
```

Inputs: image path, invert, threshold, and image id.

Returns: true when the image is saved successfully

Print Barcode

```
boolean printBarcode (String data, Barcode type, int width, int height)
```

Inputs: Barcode data, Barcode type, Width, Height

Returns: true if the bar code printed successfully

To Print Unicode Text

```
printUnicodeText(txt, Layout.Alignment.ALIGN_NORMAL, mDefaultTextPaint);
```

Input: String text Language, Alignment, fond size.

Print multiple languages using this commands

Sample Code to print a bill

```
// Batch all the commands
mPrinter.setPrintMode(AppConsts.PRINT_IN_BATCH);
mPrinter.setHighIntensity();

// Bill Header Start
mPrinter.setAlignmentCenter();
mPrinter.printLineFeed("MY COMPANY BILL\n");
mPrinter.printLineFeed("~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");
mPrinter.printLineFeed();
// Bill Header End

// Bill Details Start
mPrinter.setAlignmentLeft();
mPrinter.printTextLine("Customer Name     : John Doe  \n");
mPrinter.printTextLine("Customer Order ID : 12345     \n");
mPrinter.printTextLine("------------------------------\n");
mPrinter.printTextLine("  Item      Quantity     Price\n");
mPrinter.printTextLine("------------------------------\n");
mPrinter.printTextLine("  Item 1          1       1.00\n");
mPrinter.printTextLine("  Some big item   10   7890.00\n");
mPrinter.printTextLine("  Next Item       999 10000.00\n");
mPrinter.printLineFeed();
mPrinter.printTextLine("------------------------------\n");
mPrinter.printTextLine("  Total               17891.00\n");
mPrinter.printTextLine("------------------------------\n");
// Bill Details End

// Bill Footer Start
mPrinter.printLineFeed();
```

```
mPrinter.printTextLine("     Thank you ! Visit Again  \n");
mPrinter.printLineFeed();
mPrinter.printTextLine("*****************************\n");
mPrinter.printLineFeed();
// Bill Footer End

//Clearance for Paper tear
mPrinter.printLineFeed();
mPrinter.printLineFeed();
mPrinter.resetPrinter();


//send all batched commands to printer
mPrinter.batchPrint();
```

the printed bill will look like

Print Image

We can print directly any image of image type jpg, png, bmp. Here is the command to print image directly

```
Boolean Invert;
Invert = !bInvertBitmap;

boolean r= mPrinter.printDirect(fileUri.getPath(), Invert, threshold);
if (r) {
   Toast.makeText(getActivity(), "Image Printed",Toast.LENGTH_SHORT).show();
```

```
} else {
   Toast.makeText(getActivity(), mPrinter.getPrinterStatusMessage(),
Toast.LENGTH_SHORT).show();
}
```

The command takes 3 parameters,

1. **fileUri**.getPath() - image file path
2. Invert – set to true to print normal image, false to print inverted image.
3. **threshold** – value between 0 -255, to adjust contrast (default 127) 0 for full black 255 for full white.

Save Image

We can save any image to Printer , the image will be retained in memory even after printer switch off. Here is the command to print image

```
Boolean Invert;
Invert = !bInvertBitmap;

boolean r= mPrinter.saveImage(fileUri.getPath(), Invert, threshold, indexNumber);
if (r) {
   Toast.makeText(getActivity(), "Image saved on index "+indexNumber,
         Toast.LENGTH_SHORT).show();
} else {
   Toast.makeText(getActivity(),mPrinter.getPrinterStatusMessage(),
         Toast.LENGTH_SHORT).show();
}
```

The command takes 4 parameters,

1. **fileUri**.getPath() - image file path
2. Invert - set to true to print normal image, false to print inverted image.
3. **threshold** – value between 0 -255, to adjust contrast (default 127) 0 for full black 255 for full white.
4. **indexNumber –** image ID to store image (1 – 6) . printer can store upto 6 images.

Notes on Images

For a 2 Inch Printer the max image width is 384 pixels.

For a 3 Inch Printer the max image width is 576 pixels

For a 4 Inch Printer the max image with is 832 pixels.

The with of the image should always be a multiple of 8.

If the input image is bigger in size then it will be scaled proportionately based on the aspect ratio of the image.

If there are lines in the image, ensure that they are at least 2 pixels in width to ensure visibility.

If the image contains letters and characters, please endure they are bold and legible, this will make the letters to be printed clear and understandable.

Print Barcode

We can Print the barcode, pass the parameters barcode data, barcode format, width, height to print bar code as shown below.

mPrinter.encodeAsBitmap(txt, BarcodeFormat.***CODE_128***, ***BARCODE_WIDTH***, ***BARCODE_HEIGHT***);
the printed barcode will look like

The following barcode formats are supported AZTEC, CODABAR, CODE_39, CODE_93, CODE_128, DATA_MATRIX, EAN_8, EAN_13, ITF, MAXICODE, PDF_417, RSS_14, RSS_EXPANDED, UPC_A, UPC_E and UPC_EAN_EXTENSION

Print QR code

We can print QR code, pass the parameter QR code data to print QR code as shown below

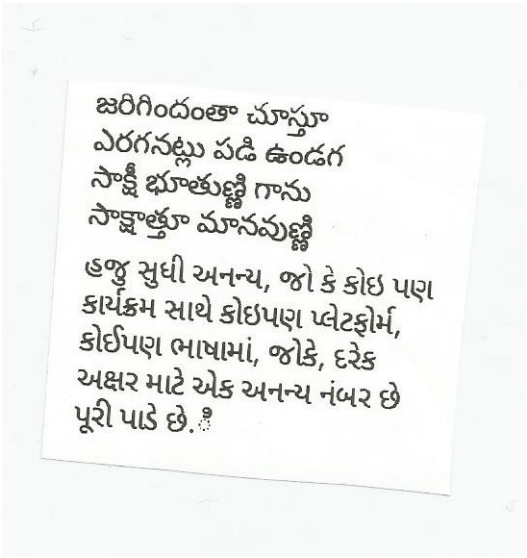
mPrinter.printQRcode(data);

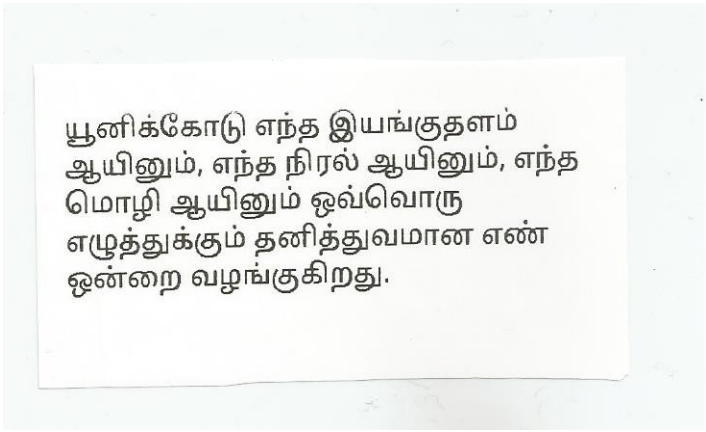
the printed barcode will look like

Print in choice of you language

You can print in any language of our choice on the printer. If the language is supported by the Android OS and the respective font files are available.

```
printUnicodeText(txt);
```





Showing List of available printers

The SDK will remember the last connected Printer and attempt to connect to it automatically, when you initialize the printer. However you can scan and connect to a COINEL Bluetooth printer by calling the method

```
mPrinter.showDeviceList(this);  //it will show list of available printer
```

Here is the screen shot for the dialog that shows up and when scanning for new printers.