

Android Module Program Manual

ESC/POS

Mobile Printer

Rev. 1.094

## CONTENTS

1. Instruction.
2. Method.
3. Command List supported by PrintNormal() function in OLE POS Command.

## **1. Instruction**

This Android Module Program Manual describes the method which is exposed from Jar package file needed in developing Android Mobile application.

## 2. Method.

Defined in the ESCPOSPrinter Class. Constant variable are defined in ESCPOSConst Interface.

### 2.1. ESCPOSPrinter

This is Constructor method. ESCPOSPrinter object select a character set using for parameter.

If do not use a parameter, default character set is ISO-8859-1.

ESCPOSPrinter() , ESCPOSPrinter(String charset)

ESCPOSPrinter(DeviceConnection connection)

ESCPOSPrinter(String charset, DeviceConnection connection)

[Parameter]

\* charset

- Character set name.

\* connection

- Device connection. (USBPortConnection, WiFiMultiConnection)

### 2.2. PrintNormal

This function is used for supporting text printing and OLE POS command.

void PrintNormal(String data)

[Parameter]

\* data

- Pointer to a null-terminated Unicode string. It is same as PrintNormal function in OLE POS Command.

### 2.3. PrintString

This function is used for supporting text printing with ESC command.

void PrintString(String data)

[Parameter]

\* data

- Pointer to a null-terminated Unicode string. It sets Unicode String to print.

## 2.4. PrintText

This function is used for supporting text printing

```
void PrintText(String data,int alignment,int attribute,int textSize)
```

[Parameter]

\* data

- Pointer to a null-terminated Unicode string. It sets Unicode text to print.

\* alignment

- This value is alignment. It sets text alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

\* attribute

- This value is text attributes. It sets text attributes to print.

Variable	Description
CMP_FNT_DEFAULT	FontA, Set up as a standard
CMP_FNT_FONTB	Set up as FontB
CMP_FNT_BOLD	Set up as Bold attribute
CMP_FNT_UNDERLINE	Set up as Underline attribute (1dot)
CMP_FNT_UNDERLINE2	Set up as Underline attribute (2dot)

\* textSize

- This value is text size. It sets text size to print.

Variable (Set up width ratio)	Description
CMP_TXT_1WIDTH	Set up width ratio as x1
CMP_TXT_2WIDTH	Set up width ratio as x2
CMP_TXT_3WIDTH	Set up width ratio as x3
CMP_TXT_4WIDTH	Set up width ratio as x4
CMP_TXT_5WIDTH	Set up width ratio as x5
CMP_TXT_6WIDTH	Set up width ratio as x6
CMP_TXT_7WIDTH	Set up width ratio as x7
CMP_TXT_8WIDTH	Set up width ratio as x8

Variable (Set up height ratio)	Description
CMP_TXT_1HEIGHT	Set up height ratio as x1
CMP_TXT_2HEIGHT	Set up height ratio as x2
CMP_TXT_3HEIGHT	Set up height ratio as x3

CMP_TXT_4HEIGHT	Set up height ratio as x4
CMP_TXT_5HEIGHT	Set up height ratio as x5
CMP_TXT_6HEIGHT	Set up height ratio as x6
CMP_TXT_7HEIGHT	Set up height ratio as x7
CMP_TXT_8HEIGHT	Set up height ratio as x8

## 2.5. PrintBitmap

This function is used for printing image files [BMP/JPEG/PNG/GIF].

void PrintBitmap(String bitmapName, int alignment)

void PrintBitmap(String bitmapName, int alignment, int size)

void PrintBitmap(Bitmap bmp, int alignment)

void printBitmap(Bitmap bmp, int alignment, int size) ← **Added in 1.073**

[Parameter]

\* BitmapName

- Pointer to a null-terminated Unicode string. This value is the bitmap file name with full path of bitmap file.

\* Bmp

- Android Bitmap Object. [android.graphics.Bitmap]

\* Alignment

- This value is alignment. It sets image alignment..

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

\* Size

- This value is image size. It sets image size to print.

Variable	Description
CMP_BITMAP_NORMAL	Normal size
CMP_BITMAP_DOUBLE_WIDTH	Double width
CMP_BITMAP_DOUBLE_HEIGHT	Double height
CMP_BITMAP_QUADRUPLE	Double size

## 2.6. PrintBarCode

This function is used for supporting barcode printing.

void PrintBarCode(String data,int symbology,int height,int width,int alignment,int textPosition)

[Parameter]

\* Data

- Pointer to a null-terminated Unicode string. It sets the barcode data to print.

\* Symbology

- This value is barcode symbol type. It sets barcode type to print.

Variable	Description
CMP_BCS_UPCA	Print UPC A BarCode
CMP_BCS_UPCE	Print UPC E BarCode
CMP_BCS_EAN8	Print EAN-8 BarCode
CMP_BCS_EAN13	Print EAN-13 BarCode
CMP_BCS_JAN8	Print JAN-8 BarCode
CMP_BCS_JAN13	Print JAN-13 BarCode
CMP_BCS_ITF	Print Interleaved 2 of 5
CMP_BCS_Codabar	Print Codabar BarCode
CMP_BCS_Code39	Print Code 3 of 9 BarCode
CMP_BCS_Code93	Print Code 93 BarCode
CMP_BCS_Code128	Print Code 128 BarCode

\* Height

- This value is barcode height in Dot Units. It sets barcode height to print.

\* Width

- This value is barcode width [2 <= value <= 6]. It sets total barcode width to print.

\* Alignment

- This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

\* TextPosition

- This value is printing position of barcode HRI letters(barcode data).

Variable	Description
CMP_HRI_TEXT_NONE	Do not print barcode data
CMP_HRI_TEXT_ABOVE	Print barcode data above the barcode
CMP_HRI_TEXT_BELOW	Print barcode data below the barcode

## 2.7. LineFeed

This function is used for sending feeding command to printer.

```
void LineFeed(int LFCount)
```

[Parameter]

\* LFCount

- This value is the number of lines for line feeding. It sets line feeding counter.

## 2.8. PrinterCheck

This function is used for printer status checking.

int PrinterCheck()

int PrinterCheck(int timeout)

int printerCheck(int timeout, Boolean bCheckResults) ← **Added in 1.073**

[Parameter]

\* Timeout : milliseconds. (Default : 5000 ms)

[Return Values]

CMP\_SUCCESS : This value returns when a function succeeds.

CMP\_FAIL : This value returns when a function fails.

CMP\_STS\_PRINTEROFF : This value returns when printer is off. ← **Added in 1.087**

CMP\_STS\_TIMEOUT : This value returns when there is no response to the request. ← **Added in 1.087**

## 2.9. status

This function is used for getting the printer status.

int status()

[Return Values]

CMP\_STS\_NORMAL: Printer Status is No Error and MSR is not Ready.

CMP\_PAPER\_EMPTY : Printer Status is no paper.

CMP\_COVER\_OPEN : Printer Cover is open.

CMP\_BATTERY\_LOW : Printer battery capacity is low.

CMP\_STS\_MSR\_READ : Currently MSR in read mode, printing is impossible.

CMP\_STS\_PRINTEROFF : Printer Status is OFF. ← **Added in 1.087**

## 2.10. PrintNVBitmap

This function is used to support the Bitmap Image printing stored in Flash Memory.

void PrintNVBitmap(int NVImageNumber)



```
void PrintNVBitmap(int NVImageNumber, int size)
```

[Parameter]

\* NVImageNumber

- It sets the Number image stored in Flash Memory to print.

\* Size

- This value is image size. It sets image size to print.

Variable	Description
CMP_BITMAP_NORMAL	Normal size
CMP_BITMAP_DOUBLE_WIDTH	Double width
CMP_BITMAP_DOUBLE_HEIGHT	Double height
CMP_BITMAP_QUADRUPLE	Double size

## 2.11 PrintPDF417

This method is used for supporting PDF417 barcode printing.

```
void printPDF417(String pdfData, int dataLength, int numberOfColumns, int cellWidth,  
int alignment)
```

[Parameter]

\* data

- Barcode data to print.

\* dataLength

- Length of Barcode data.

\* columns

- Columns. (1 – 20)

\* cWidth

- Cell Width.

\* align

- This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

## 2.12 PrintQRCode

This method is used for supporting QRCode barcode printing.

```
void printQRCode(String data, int dataLength, int moduleSize, int ecLevel, int alignment)
```

---

```
void printQRCode(String data, int dataLength, int version, int moduleSize, int ecLevel,
                 int alignment)
```

[Parameter]

\* data

- Barcode data to print.

\* dataLength

- Length of barcode data.

\* version

- QRCode Version. (Auto = 0, Fixed = 1 ~ 40)

\* moduleSize

- Module size. (1 – 20)

\* ECLevel

- Error Correction Level.

Variable	Description
CMP_QRCODE_EC_LEVEL_L	Error correction Level L (7%)
CMP_QRCODE_EC_LEVEL_M	Error correction Level M (15%)
CMP_QRCODE_EC_LEVEL_Q	Error correction Level Q (25%)
CMP_QRCODE_EC_LEVEL_H	Error correction Level H (30%)

\* align

- This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

## 2.13. GetEmulation

This function is used for getting the printer emulation.

This function is not compatible with CMP-20 Printer.

```
int getEmulation()
```

[Return Values]

EMUL\_CPCL : CPCL Emulation.

EMUL\_ESCPOS : ESCPOS Emulation

## 2.14. printerResults ← Added in 1.071

This function is used to know the printing result.

```
int printerResults()
```

```
int printerResults(int timeout)
```

[Parameter]

\* timeout

- It receives the waiting time of the printing result.

[Return Values]

Variable	Description
CMP_STS_NORMAL	Printing is success.
CMP_STS_PAPER_EMPTY	Printer Status is no paper.
CMP_STS_COVER_OPEN	Printer Cover is open.
CMP_STS_PRINTEROFF	Printer Status is off.
CMP_STS_TIMEOUT	Printer not response.
CMP_FAIL	Function call failed.
Other values	Printer status is unknown error.

## 2.15. printAndroidFont ← Added in 1.071

This function is used for android embedded font printing.

```
void printAndroidFont(String textString, int widthDots, int textSize, int alignment)
```

```
void printAndroidFont(Typeface typeface, String textString, int widthDots, int textSize,  
    int alignment)
```

```
void printAndroidFont(Typeface typeface, Boolean isBold, String textString, int widthDots,  
    int textSize, int alignment)
```

```
void printAndroidFont(Typeface typeface, Boolean isBold, Boolean isItalic, String textString,  
    int widthDots, int textSize, int alignment)
```

```
void printAndroidFont(Typeface typeface, Boolean isBold, Boolean isItalic,  
    boolean isUnderline, String textString, int widthDots, int textSize, int alignment)
```

[Parameter]

\* textString

- Unicode which has a null-terminated string. It receives text to print as a factor.

\* widthDots

- It receives the printing width value of the text to print as a factor. [ Default : dot ]

\* textSize

- It receives the font size value of the text to print as a factor. [ Default : dot ]

\* Alignment

- This value is alignment. It sets image alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

\* typeface

- It receives the typeface of the android's font as a factor.

Variable	Description
SANS_SERIF	SANS_SERIF font
SERIF	SERIF font
MONOSPACE	MONOSPACE font

\* isBold

- It receives the bold of the android's font as a factor.

\* isItalic

- It receives the italic of the android's font as a factor.

\* isUnderline

- It receives the underline of the android's font as a factor.

## 2.16. setCharSet ← Added in 1.071

Set character Set.

```
void setCharSet(String charSet)
```

[Parameter]

\* charSet

- Character set name.

## 2.17. setCodepage ← Added in 1.071

Set character code table.

```
int setCodepage(int iCodepage)
```

[Parameter]

\* iCodepage

- Character code table.

## 2.18. setText ← Added in 1.071

This function is used for supporting text setting information

```
void setText(int attribute,int textSize)
```

[Parameter]

\* attribute

- This value is text attributes. It sets text attributes to print.

Variable	Description
CMP_FNT_DEFAULT	FontA, Set up as a standard
CMP_FNT_FONTB	Set up as FontB
CMP_FNT_BOLD	Set up as Bold attribute
CMP_FNT_UNDERLINE	Set up as Underline attribute
CMP_FNT_REVERSE	Set up as reverse print attribute

\* textSize

- This value is text size. It sets text size to print.

Variable (Set up width ratio)	Description
CMP_TXT_1WIDTH	Set up width ratio as x1
CMP_TXT_2WIDTH	Set up width ratio as x2
CMP_TXT_3WIDTH	Set up width ratio as x3
CMP_TXT_4WIDTH	Set up width ratio as x4
CMP_TXT_5WIDTH	Set up width ratio as x5
CMP_TXT_6WIDTH	Set up width ratio as x6
CMP_TXT_7WIDTH	Set up width ratio as x7
CMP_TXT_8WIDTH	Set up width ratio as x8

Variable (Set up height ratio)	Description
CMP_TXT_1HEIGHT	Set up height ratio as x1
CMP_TXT_2HEIGHT	Set up height ratio as x2
CMP_TXT_3HEIGHT	Set up height ratio as x3
CMP_TXT_4HEIGHT	Set up height ratio as x4
CMP_TXT_5HEIGHT	Set up height ratio as x5
CMP_TXT_6HEIGHT	Set up height ratio as x6
CMP_TXT_7HEIGHT	Set up height ratio as x7
CMP_TXT_8HEIGHT	Set up height ratio as x8

## 2.19. setAlignment ← Added in 1.071

This function is used for supporting text alignment

```
void setAlignment(int alignment)
```

[Parameter]

\* alignment

- This value is alignment. It sets text alignment.

Variable	Description
CMP_ALIGNMENT_LEFT	Left alignment
CMP_ALIGNMENT_CENTER	Center alignment
CMP_ALIGNMENT_RIGHT	Right alignment

## 2.20. setDithering

This function is used to set the Bitmap Dithering.

```
void setDithering(int iDither)
```

[Parameter]

\* iDither

- This value is dithering method.

Variable	Description
CMP_BITMAP_NO_DITHER	Thresholding method
CMP_BITMAP_ERROR_DIFFUSION	Error diffusion method
CMP_BITMAP_ORDERED_DITHER	Ordered dithering method

### 3. Command List supported by PrintNormal() function in OLE POS Command.

**One Shots**      Perform indicated action.

Name	Data	Remarks
Feed and Paper cut	ESC  #fP	Cuts receipt paper, after feeding the paper by the RecLinesToPaperCut lines. The character ' #' is defined by the " Paper cut" escape sequence.
Print bitmap	ESC  #B	Prints the pre-stored bitmap. The character ' #' is replaced by the bitmap number.
Feed lines	ESC  #fF	Feed the paper forward by lines. The character ' #' is replaced by an ASCII decimal string telling the number of lines to be fed. If ' #' is omitted, then one line is fed.
Feed units	ESC  #uF	Feed the paper forward by mapping mode units. The character ' #' is replaced by an ASCII decimal string telling the number of units to be fed. If ' #' is omitted, then one unit is fed.

**Print Mode**      Characteristics that are remembered until explicitly changed.

Name	Data	Remarks
Font typeface selection	ESC  #fT	Selects a new typeface for the following data. Values for the character ' #' are: 0 = Default typeface. 1 = Select first typeface from the FontTypefaceList property. 2 = Select second typeface from the FontTypefaceList property. And so on.

**Print Line**      Characteristics that are reset at the end of each print method or by a "Normal"

sequence.

Name	Data	Remarks
Bold	ESC  [ ]bC	Prints in bold or double-strike.
Underline	ESC  [ ][#]uC	Prints with underline. The character ‘ #’ is replaced by an ASCII decimal string telling the width of the underline in printer dot units. If ‘ #’ is omitted, then a printer-specific default width is used.
Reverse video	ESC   [ ]rvC	Prints in a reverse video format. If ‘ !’ is specified then reverse video is disabled
Single high and wide	ESC 1C	Prints normal size.
Double wide	ESC 2C	Prints double-wide characters.
Double high	ESC 3C	Prints double-high characters.
Double high and wide	ESC 4C	Prints double-high/double-wide characters.
Scale horizontally	ESC  #hC	Prints with the width scaled ‘ #’ times the normal size, where ‘ #’ is replaced by an ASCII decimal string.
Scale vertically	ESC  #vC	Prints with the height scaled ‘ #’ times the normal size, where ‘ #’ is replaced by an ASCII decimal string.
Center	ESC cA	Aligns following text in the center.
Right justify	ESC rA	Aligns following text at the right.
Left justify	ESC lA	Aligns following text at the left.
Normal	ESC N	Restores printer characteristics to normal condition.