Android Module Program Manual

ESC/POS

POS Printer

Rev. 1.102

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
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

* attribute

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

Variable	Description
AURES_FNT_DEFAULT	FontA, Set up as a standard
AURES_FNT_FONTB	Set up as FontB
AURES_FNT_BOLD	Set up as Bold attribute
AURES_FNT_UNDERLINE	Set up as Underline attribute
AURES_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
AURES_TXT_1WIDTH	Set up width ratio as x1
AURES_TXT_2WIDTH	Set up width ratio as x2
AURES_TXT_3WIDTH	Set up width ratio as x3
AURES_TXT_4WIDTH	Set up width ratio as x4
AURES_TXT_5WIDTH	Set up width ratio as x5
AURES_TXT_6WIDTH	Set up width ratio as x6
AURES_TXT_7WIDTH	Set up width ratio as x7
AURES_TXT_8WIDTH	Set up width ratio as x8

Variable (Set up height ratio)	Description
AURES_TXT_1HEIGHT	Set up height ratio as x1
AURES_TXT_2HEIGHT	Set up height ratio as x2
AURES_TXT_3HEIGHT	Set up height ratio as x3
AURES_TXT_4HEIGHT	Set up height ratio as x4
AURES_TXT_5HEIGHT	Set up height ratio as x5
AURES_TXT_6HEIGHT	Set up height ratio as x6
AURES_TXT_7HEIGHT	Set up height ratio as x7
AURES_TXT_8HEIGHT	Set up height ratio as x8

2.5. PrintBitmap

This function is used for printing image files.

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 1.077**

// Added in 1.093

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

void PrintBitmap(String bitmapName, int alignment, int size, int brightness, int reverseprint,
int compress)

void PrintBitmap(Bitmap bmp, int alignment, int size, int brightness, int reverseprint)

void PrintBitmap(Bitmap bmp, int alignment, int size, int brightness, int reverseprint,
int compress)

[Parameter]

* BitmapName

- Pointer to a null-terminated Unicode string. This value is the bitmap file name with full path of bitmap file. [BMP/JPEG/PNG/GIF]

* Bmp

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

* Alignment

- This value is alignment. It sets image alignment.

Variable	Description
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

* Size

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

Variable	Description
AURES_BITMAP_NORMAL(0)	Original(Normal) size
AURES_BITMAP_DOUBLE_WIDTH(1)	Double width
AURES_BITMAP_DOUBLE_HEIGHT(2)	Double height
AURES_BITMAP_QUADRUPLE(3)	Double size
Others	Printing size of image

* brightness

- This value is a brightness of image.

* reverseprint

- This value is a reverse printing. [0 = Normal, 1 = Reverse]

* compress

- This value is a compression of data. [0=Raw, 1=Compress]

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
AURES_BCS_UPCA	Print UPC A BarCode
AURES_BCS_UPCE	Print UPC E BarCode
AURES_BCS_EAN8	Print EAN-8 BarCode
AURES_BCS_EAN13	Print EAN-13 BarCode
AURES_BCS_JAN8	Print JAN-8 BarCode
AURES_BCS_JAN13	Print JAN-13 BarCode
AURES_BCS_ITF	Print Interleaved 2 of 5
AURES_BCS_Codabar	Print Codabar BarCode
AURES_BCS_Code39	Print Code 3 of 9 BarCode
AURES_BCS_Code93	Print Code 93 BarCode
AURES_BCS_Code128	Print Code 128 BarCode
AURES_BCS_3OF5	Print 3 out of 5(KorMail) BarCode

* Height

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

* Width

- This values barcode width in Dot Units. It sets total barcode width to print.

* Alignment

- This value is alignment. It sets barcode alignment.

Variable	Description
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

* TextPosition

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

Variable	Description

AURES_HRI_TEXT_NONE	Do not print barcode data
AURES_HRI_TEXT_ABOVE	Print barcode data above the barcode
AURES_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.

2.10. 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
AURES_BITMAP_NORMAL	Normal size
AURES_BITMAP_DOUBLE_WIDTH	Double width
AURES_BITMAP_DOUBLE_HEIGHT	Double height
AURES_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)

int printPDF417(String pdfData, int dataLength, int cellWidth, int cellHeightRatio,
 int numberOfColumns, int numberOfRows, int eclType, int eclValue,
 int alignment) ← **Added in 1.098**

[Parameter]

* pdfData

- Barcode data to print.

* dataLength

- Length of pdfData.

* numberOfColumns

- Set the number of columns.

Range : [0 – 30].

* numberOfRows

- Set the number of rows.

Range[0, 3 – 90].

* cellWidth

- Cell width.

Range[2 – 8].

* cellHeightRatio

- Cell height ratio. [Cell height = [cellHieghtRatio × cellWidth].

Range[2 – 8].

* eclType

- Set the error correction level.

Range[0 – 1].

0= The error correction level is set by "level

1= The error correction level is set by "ratio." The ratio is [eclValue × 10%].

* eclValue

- Set the error correction level.

eclType = 0 : Range[0 – 8].

eclType = 1 : Range[1 – 40].

* alignment

- This value is alignment. It sets barcode alignment.

Variable	Description
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

[Return Values]

AURES_SUCCESS : This value returns when a function succeeds.

others : This value returns when a function fails.

2.12 QRCode

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
AURES_QRCODE_EC_LEVEL_L	Error correction Level L (7%)
AURES_QRCODE_EC_LEVEL_M	Error correction Level M (15%)
AURES_QRCODE_EC_LEVEL_Q	Error correction Level Q (25%)
AURES_QRCODE_EC_LEVEL_H	Error correction Level H (30%)

* align

- This value is alignment. It sets barcode alignment.

Variable	Description
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

2.13 CutPaper

This method is used for supporting Paper cutting.

void cutPaper();

2.14 PrinterSts

This method returns printer status. It can’t used in ASB Mode.

int printerSts();

int printerSts(Boolean bCheckResults);

[Parameter]

* bCheckResults

- Set the result of printing or not.

If bCheckResults is true,

[Return value]

AURES_STS_NORMAL	Normal Status
AURES_STS_COVEROPEN	Cover Open Status
AURES_STS_PAPERNEAREMPTY	Paper Near End Status
AURES_STS_PAPEREMPTY	Paper Empty Status

2.15 DrawerSts

This method returns cash drawer status. It can’t used in ASB Mode.

int drawerSts()

[Return value]

STS_CD_OPEN	Cashdrawer Open status
STS_CD_CLOSE	Cashdrawer Close status

2.16 OpenDrawer

Open a cash drawer.

void openDrawer(int pinNum, int onTime, int offTime)

[Parameter]

pinNum : Pin number to generate pulse.

Variable	Description
CD_PIN_TWO	PIN 2
CD_PIN_FIVE	PIN 5

onTime : Start time to generate pulse. (onTime * 2 ms)

offTime : Stop time to generate pulse. (offTime * 2 ms)

2.17 ASBOn

Enable ASB Mode. (ASB : Auto Status Back)

int asbOn()

[Return value]

AURES_SUCCESS , AURES_FAIL

2.18 ASBOff

Disable ASB Mode.

int asbOff()

[Return value]

AURES_SUCCESS , AURES_FAIL

2.19 isASBMode

Returns ASB Mode status.

boolean asbMode()

[Return value]

TRUE, FALSE ASB Mode Status

2.20 PrintPageModeData

Print and return to standard mode in page mode.

void printPageModeData()

2.21 ClearPageModeData

Cancel print data in page mode.

void clearPageModeData()

2.22 SetPageMode

Change to page mode or standard mode.

void setPageMode(boolean pagemode)

[Parameter]

* pagemode

- Enable or Disable page mode. (TRUE, FALSE)

2.23 SetPrintDirection

Select print direction in page mode.

void setPrintDirection(int direction)

[Parameter]

* direction

- Print direction

Variable	Description
DIRECTION_LEFT_RIGHT	Starting upper left
DIRECTION_BOTTOM_TOP	Starting lower left
DIRECTION_RIGHT_LEFT	Starting lower right
DIRECTION_TOP_BOTTOM	Starting upper right

2.24 SetPrintingArea

Set printing area in page mode.

void setPrintingArea(int pageWidth, int pageHeight)

[Parameter]

* pageWidth

  - Width of printing area.

* pageHeight

  - Height of printing area.

2.25 SetDPI

Set DPI in page mode.

void setDPI(int dpi)

[Parameter]

* dpi

  - Dot per Inch (Constant value, ex: 180 , 203)

2.26 SetAbsoluteHorizontal

Set absolute horizontal print position in page mode. (X axis)

void setAbsoluteHorizontal(int absolutePosition)

[Parameter]

* absolutePosition

  - Starting position.

2.27 SetRelativeHorizontal

Set relative horizontal print position in page mode. (X axis)

void setRelativeHorizontal(int relativePosition)

[Parameter]

* relativePosition

  - Starting position.

2.28 SetAbsoluteVertical

Set absolute vertical print position in page mode. (Y axis)

void setAbsoluteVertical(int absolutePosition)

[Parameter]

* absolutePosition

  - Starting position.

2.29 SetRelativeVertical

Set relative vertical print position in page mode. (Y axis)

void setRelativeVertical(int relativePosition)

[Parameter]

* relativePosition

- Starting position.

2.30 setCharSet

Set character Set.

void setCharSet(String charSet)

[Parameter]

* charSet

- Character set name.

2.31 ICRMode

Set the ICR communication.

int ICRMode(boolean bICR)

[Parameter]

* bICR

- Set the ICR communication mode.

Variable	Description
true	ICR enable
false	ICR disable

[Return value]

AURES_SUCCESS , AURES_FAIL

2.32 ICR_SendReceive

It used to send ICR Command and receive the response from ICR.

```
int ICR_SendReceive(    byte ucCommandLen,
                        byte [] prgucCommand,
                        byte pucReaderResponseLen,
                        byte [] prgucReaderResponse )
```

[Parameter]

* ucCommandLen

- Set the ICR Command length.

* prgucCommand

- Set the ICR Command.

* pucReaderResponseLen

- Set the ICR Response length.

* prgucReaderResponse

- Set the ICR Response buffer.

[Return value]

Variable	Description
NO_ERR	Communication success
Others	Communication error

2.33 getResponseSize

It used to send ICR Command and receive the response from ICR.

int getResponseSize()

[Parameter]

* none

[Return value]

Returns the size of reponse length in ICR_SendReceive function.

2.34 cardInserted

Get the status of card.

boolean cardInserted()

[Parameter]

* none

[Return value]

Variable	Description
true	Card is inserted
false	Card is not inserted

2.34. PrintAndroidFont ← Added in 1.075

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
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_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.

2.35. PrinterResults ← Added in 1.077

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.

2.36. SetText ← Added in 1.077

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
AURES_FNT_DEFAULT	FontA, Set up as a standard
AURES_FNT_FONTB	Set up as FontB
LK _FNT_BOLD	Set up as Bold attribute
LK _FNT_UNDERLINE	Set up as Underline attribute
LK _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
LK _TXT_1WIDTH	Set up width ratio as x1
LK _TXT_2WIDTH	Set up width ratio as x2
LK _TXT_3WIDTH	Set up width ratio as x3
LK _TXT_4WIDTH	Set up width ratio as x4
LK _TXT_5WIDTH	Set up width ratio as x5
LK _TXT_6WIDTH	Set up width ratio as x6
LK _TXT_7WIDTH	Set up width ratio as x7
LK _TXT_8WIDTH	Set up width ratio as x8

Variable (Set up height ratio)	Description
LK _TXT_1HEIGHT	Set up height ratio as x1
LK _TXT_2HEIGHT	Set up height ratio as x2
LK _TXT_3HEIGHT	Set up height ratio as x3
LK _TXT_4HEIGHT	Set up height ratio as x4
LK _TXT_5HEIGHT	Set up height ratio as x5
LK _TXT_6HEIGHT	Set up height ratio as x6
LK _TXT_7HEIGHT	Set up height ratio as x7
LK _TXT_8HEIGHT	Set up height ratio as x8

2.37. SetAlignment ← Added in 1.077

This function is used for supporting text alignment

void SetAlignment(int alignment)

[Parameter]

* alignment

- This value is alignment. It sets text alignment.

Variable	Description
LK _ALIGNMENT_LEFT	Left alignment

LK _ALIGNMENT_CENTER	Center alignment
LK _ALIGNMENT_RIGHT	Right alignment

2.38. printPdfFile ← Added in 1.094

This function is used for supporting print pdf file.

void printPdfFile(String pdfName, int pageNum, int alignment, int widthSize, int brightness,
int reverseprint, int compress)

[Parameter]

* pdfName

- Path of pdf file to be printed.

* pageNum

- page number to be printed

* alignment

- This value is alignment. It sets image alignment.

Variable	Description
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_ALIGNMENT_RIGHT	Right alignment

* widthSize

- This value is size of printing. It sets width size to print.

* brightness

- This value is a brightness of image.

* reverseprint

- This value is a reverse printing. [0 = Normal, 1 = Reverse]

* compress

- This value is a compression of data. [0=Raw, 1=Compress]

2.39. setLabel ← Added in 1.095

This function is used for android embedded font printing.

It sets the canvas size for android embedded font.

int setLabel(int printWidth, int printHeight)

int setLabel(int printWidth, int printHeight, int iCompress) ← **Added in 1.097.**

[Parameter]

* printWidth

- It sets the printing width. [Default : dot]

* printHeight

- It sets the printing height. [Default : dot]

* iCompress

- This value is compression or not.

Variable	Description
0	No compression
Others	Compression

[Return Values]

Variable	Description
CMP_SUCCESS	Setting is success.
Other values	No response. Printer error.

2.40. printLabel ← Added in 1.095

This function is used for android embedded font printing.

It sets the start of printing.

int printLabel()

2.41. PrintAndroidFont ← Added in 1.095

This function is used for android embedded font printing.

Reference to 2.39 and 2.40.

void PrintAndroidFont(int printX, int printY, String textString, int widthDots, int textSize, int alignment)

void PrintAndroidFont(int printX, int printY, Typeface typeface, String textString, int widthDots, int textSize,
int alignment)

void PrintAndroidFont(int printX, int printY, Typeface typeface, Boolean isBold, String textString, int widthDots,
int textSize, int alignment)

void PrintAndroidFont(int printX, int printY, Typeface typeface, Boolean isBold, Boolean isItalic, String textString,
int widthDots, int textSize, int alignment)

void PrintAndroidFont(int printX, int printY, Typeface typeface, Boolean isBold, Boolean isItalic,
boolean isUnderline, String textString, int widthDots, int textSize, int alignment)

[Parameter]

* printX

- It receives the x-coordinates value of the text to print as a factor.

* printY

- It receives the y-coordinates value of the text to print as a factor.

* 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
AURES_ALIGNMENT_LEFT	Left alignment
AURES_ALIGNMENT_CENTER	Center alignment
AURES_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.

[Return Values]

Variable	Description
CMP_SUCCESS	Printing is success.
Other values	No response. Printer error.

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 |#lF	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.