
Android Module Program Manual

ESC/POS

Mobile, Thermal Printer

Rev. 1.111

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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_ALIGNMENT_RIGHT | Right alignment |

* attribute

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

| Variable | Description |
|------------------|-----------------------------------|
| LK_FNT_DEFAULT | FontA, Set up as a standard |
| LK_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 |

2.5. 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 1.093A

```
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(Value) | Description |
|------------------------|------------------|
| LK_ALIGNMENT_LEFT(0) | Left alignment |
| LK_ALIGNMENT_CENTER(1) | Center alignment |
| LK_ALIGNMENT_RIGHT(2) | Right alignment |

* Size

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

| Variable(Value) | Description |
|----------------------------|------------------------|
| LK_BITMAP_NORMAL(0) | Original(Normal) size |
| LK_BITMAP_DOUBLE_WIDTH(1) | Double width |
| LK_BITMAP_DOUBLE_HEIGHT(2) | Double height |
| LK_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 |
|----------------|-----------------------------------|
| LK_BCS_UPCA | Print UPC A BarCode |
| LK_BCS_UPCE | Print UPC E BarCode |
| LK_BCS_EAN8 | Print EAN-8 BarCode |
| LK_BCS_EAN13 | Print EAN-13 BarCode |
| LK_BCS_JAN8 | Print JAN-8 BarCode |
| LK_BCS_JAN13 | Print JAN-13 BarCode |
| LK_BCS_ITF | Print Interleaved 2 of 5 |
| LK_BCS_Codabar | Print Codabar BarCode |
| LK_BCS_Code39 | Print Code 3 of 9 BarCode |
| LK_BCS_Code93 | Print Code 93 BarCode |
| LK_BCS_Code128 | Print Code 128 BarCode |
| LK_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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_ALIGNMENT_RIGHT | Right alignment |

* TextPosition

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

| Variable | Description |
|-------------------|--------------------------------------|
| LK_HRI_TEXT_NONE | Do not print barcode data |
| LK_HRI_TEXT_ABOVE | Print barcode data above the barcode |
| LK_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 [Mobile Printer Only]

This function is used for printer status checking.

int printerCheck()

int printerCheck(int timeout) ← **Added in 1.077**

int printerCheck(int timeout, Boolean bCheckResults) ← **Added in 1.077**

[Parameter]

* Timeout : milliseconds. (Default : 5000 ms)

* bCheckResult

- set the result of printing.

| Variable | Description |
|----------|--|
| TRUE | Check the printing result (refer to printerResults function |
| FALSE | Don' t check the printing result |

[Return Values]

LK_SUCCESS : This value returns when a function succeeds.

LK_FAIL : This value returns when a function fails.

LK_STS_PRINTEROFF : This value returns when printer is off. ← **Added in 1.102**

LK_STS_TIMEOUT : This value returns when printer is no response or power off. ← **Added in 1.102**

(LK_STS_PRINTEROFF is returned when printerCheck function is used while printer is off.

LK_STS_TIMEOUT is returned when printerCheck function is used again)

2.9. status [Mobile Printer Only]

This function is used for getting the printer status.

```
int status()
```

[Return Values]

LK_STS_NORMAL: Printer Status is No Error and MSR is not Ready.

LK_STS_PAPER_EMPTY : Printer Status is no paper.

LK_STS_COVER_OPEN : Printer Cover is open.

LK_STS_BATTERY_LOW : Printer battery capacity is low.

LK_STS_MSR_READ : Currently MSR in read mode, printing is impossible.

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 |
|-------------------------|---------------|
| LK_BITMAP_NORMAL | Normal size |
| LK_BITMAP_DOUBLE_WIDTH | Double width |
| LK_BITMAP_DOUBLE_HEIGHT | Double height |
| LK_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 = [cellHeightRatio × 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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_ALIGNMENT_RIGHT | Right alignment |

[Return Values]

LK_SUCCESS : This value returns when a function succeeds.

others : This value returns when a function fails.

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 |
|----------------------|--------------------------------|
| LK_QRCODE_EC_LEVEL_L | Error correction Level L (7%) |
| LK_QRCODE_EC_LEVEL_M | Error correction Level M (15%) |
| LK_QRCODE_EC_LEVEL_Q | Error correction Level Q (25%) |
| LK_QRCODE_EC_LEVEL_H | Error correction Level H (30%) |

* align

- This value is alignment. It sets barcode alignment.

| Variable | Description |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_ALIGNMENT_RIGHT | Right alignment |

2.13 cutPaper [POSPrinter Only]

This method is used for supporting Paper cutting.

```
void cutPaper()
```

2.14 printerSts [POSPrinter Only]

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

```
int printerSts()
```

[Return value]

| | |
|--------------------|-----------------------|
| STS_NORMAL | Normal Status |
| STS_COVEROPEN | Cover Open Status |
| STS_PAPERNEAREMPTY | Paper Near End Status |
| STS_PAPEREMPTY | Paper Empty Status |

2.15 drawerSts [POSPrinter Only]

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 [POSPrinter Only]

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]

LK_SUCCESS , LK_FAIL

2.18 asbOff

Disable ASB Mode.

int asbOff()

[Return value]

LK_SUCCESS , LK_FAIL

2.19 isASBMode

Returns ASB Mode status.

boolean isASBMode()

[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]

LK_SUCCESS , LK_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)
void printAndroidFont(Typeface typeface, Boolean isBold, Boolean isItalic,
    boolean isUnderline, String textString, int widthDots, int textSize, int alignment,
    Boolean rotate180)

```

[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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_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.

* rotate180

- It receives the rotation(180degree) 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.

If timeout is zero, do not waiting the printing result ← **Added in 1.102.**

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 |
|------------------|-----------------------------------|
| LK_FNT_DEFAULT | FontA, Set up as a standard |
| LK_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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_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.

```

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 |
|---------------------|------------------|
| LK_ALIGNMENT_LEFT | Left alignment |
| LK_ALIGNMENT_CENTER | Center alignment |
| LK_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. |

2.42. setCompress ← Added in 1.097

This function is used for compression.

void setCompress(int iCompress)

[Parameter]

* iCompress

- This value is compression or not.

| Variable | Description |
|----------|----------------|
| 0 | No compression |
| Others | Compression |

2.43. downloadNVImage ← Added in 1.102

This function is used for save the NV images in flash.

void downloadNVImage(String bitmapPathList, int iSizeMode, int iDarkness, int iDither)

[Parameter]

* bitmapPathList

- It sets the absolute path of the image files. [BMP/JPEG/PNG/GIF]

'|' = separator

(ex) “//sdcard//temp//logo_s.jpg//sdcard//temp//logo_m.jpg”

* iSizeMode

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

| Variable(Value) | Description |
|----------------------------|-----------------------|
| LK_BITMAP_NORMAL(0) | Original(Normal) size |
| LK_BITMAP_DOUBLE_WIDTH(1) | Double width |
| LK_BITMAP_DOUBLE_HEIGHT(2) | Double height |
| LK_BITMAP_QUADRUPLE(3) | Double size |

| | |
|--------|------------------------|
| Others | Printing size of image |
|--------|------------------------|

* iDarkness

- This value is a darkness of image.

Range : [0 ~ 255]

* iDither

- This value is a dithering method.

| Variable(Value) | Description |
|------------------------------|--------------------------|
| LK_BITMAP_NO_DITHER(0) | Thresholding method |
| LK_BITMAP_ERROR_DIFFUSION(1) | Error diffusion method |
| LK_BITMAP_ORDERED_DITHER(2) | Ordered dithering method |

2.44. deleteNVImage ← Added in 1.102

This function is used to delete the NV images in flash.

void deleteNVImage()

[Parameter]

* none

2.45. printPDFFile

This function is used for printing pdf files.* **Available for Android 5.0 or later.**

void printPDFFile(String path, int alignment, int paper_size, int feed)

void printPDFFile(String path, int alignment, int paper_size, int feed, int compress)

void printPDFFile(File file, int alignment, int paper_size, int feed)

void printPDFFile(File file, int alignment, int paper_size, int feed, int compress)

void printPDFFile(ParcelFileDescriptor fd_PDF, int alignment, int paper_size, int feed,
int compress)

[Parameter]

* path

- This value is the full path of pdf file.

* file, fd_PDF

- File or ParcelFileDescriptor Object.

* Alignment

- This value is alignment. It sets image alignment.

| Variable(Value) | Description |
|------------------------|------------------|
| LK_ALIGNMENT_LEFT(0) | Left alignment |
| LK_ALIGNMENT_CENTER(1) | Center alignment |
| LK_ALIGNMENT_RIGHT(2) | Right alignment |

* paper_size

- This value is printing size. If the value is 0, the original size is printed.

* feed

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

* compress

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

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. |