

Android Module Program Manual

CPCL

Mobile Printer

Rev. 1.094

CONTENTS

1. Instruction.
2. Method.

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 CPCLPrinter Class. Constant variable are defined in CPCLConst Interface.

2.1. CPCLPrinter

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

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

CPCLPrinter() , CPCLPrinter(String charset)

[Parameter]

* charset

- Character set name.

2.2. SetForm

This function is used for defining paper form.

void SetForm(String HorizonOffset,int XResol,int YResol,String LabelHeight,
int Quantity)

[Parameter]

* HorizonOffset

- Unicode which has a null-terminated string. It receives the horizontal offset of the total label as a factor. You can understand it as an absolute location value of left printing.

* XResol

- It receives the horizontal resolution as a factor.

* YResol

- It receives vertical resolution.

* LabelHeight

- It receives the value of label height for printing as a factor.

* Quantity

- It receives label q'ty to print as a factor.[Max. =< 1024]

2.3. PrintForm

This function is used for setting the paper at the top of the form after printing.

void PrintForm()

2.4. PrinterCheck

This function is used for printer status checking.

int PrinterCheck()

int PrinterCheck(int timeout)

[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_CPCL_PRINTEROFF : This value returns when a printer off status.

← Added in 1.087

CMP_STS_CPCL_TIMEOUT : This value returns when there is no response to the request.

← Added in 1.087

2.5. status

This function is used for getting the printer status.

int status()

[Return Values]

CMP_STS_CPCL_NORMAL: Printer Status is No Error.

CMP_STS_CPCL_BUSY : Printer Status is busy status.

CMP_STS_CPCL_PAPER_EMPTY : Printer Status is no paper.

CMP_STS_CPCL_COVER_OPEN : Printer Cover is open.

CMP_STS_CPCL_BATTERY_LOW : Printer battery capacity is low.

CMP_STS_CPCL_PRINTEROFF : Printer Status is OFF. ← Added in 1.087

2.6. SetMeasure

This function is used for setting measure of command.

void SetMeasure(int Measure)

[Parameter]

* Measure

- It receives measure set command as a factor.

Variable	Description
CMP_CPCL_INCH	Set up measure as inch.
CMP_CPCL_CENTI	Set up measure as centimeter
CMP_CPCL_MILLI	Set up measure as millimeter
CMP_CPCL_DOTS	Set up measure as dot(Default)

2.7. SetJustification

This function is used for setting justification of fields.

```
void setJustification (int Justify)
```

[Parameter]

* Justify

- It receives as a factor for setting justification of fields.

Variable	Description
CMP_CPCL_LEFT	Left justification
CMP_CPCL_CENTER	Center justification
CMP_CPCL_RIGHT	Right justification

2.8. PrintCPCLText

This function is used for printing text in a specified location on the form.

```
void PrintCPCLText(int Rotation,int FontType,int FontSize,int PrintX,int PrintY,  
String Data,int count)
```

[Parameter]

* Rotation

- It received the printing direction value of the text for printing as a factor.

Variable	Description
CMP_CPCL_NO_ROTATION	Print text with no rotation.
CMP_CPCL_90_ROTATION	Print text with 90 rotation.(counterclockwise)
CMP_CPCL_180_ROTATION	Print text with 180 rotation.(counterclockwise)
CMP_CPCL_270_ROTATION	Print text with 270 rotation.(counterclockwise)

* FontType

- It receives the font type of the text to print as a factor.

* FontSize

- It receives the font size value of the text to print as a factor.

* 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.

* Data

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

* count

- It receives the count function of the text to print as a factor.

2.9. SetConcat

This function is used for defining the start of the text concatenation and the printing position.

void SetConcat(int ConcatMode,int PrintX,int PrintY)

[Parameter]

* ConcatMode

- It receives the concatenation method of the text to concatenate as a factor.

Variable	Description
CMP_CPCL_CONCAT	It concatenates text as horizontal
CMP_CPCL_VCONCAT	It concatenates text as vertical

* 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.

2.10. ConcatText

This function is used for defining the font to concatenate.

void ConcatText(int FontType,int FontSize,int Offset,String Data)

[Parameter]

* FontType

- It receives the font type of the text to print as a factor.

* FontSize

- It receives the font size value of the text to print as a factor.

* Offset

- It receives Unit-value at the starting point.

* Data

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

2.11. ResetConcat

This function is used for canceling text concatenation.

void ResetConcat()

2.12. SetMultiLine

This function is used for defining multi-line printing with same line length and same font.

void SetMultiLine(int LineHeight)

[Parameter]

* LineHeight

- It receives the line height value of the text to print as a factor.

2.13. MultiLineText

This function is used for printing text which is defined as SetMultiLine(). It received the font, the size and the printing location of the text to print as a factor.

void MultiLineText(int Rotation,int FontType,int FontSize,int PrintX,int PrintY)

[Parameter]

* Rotation

- It receives the printing direction value of the text to print as a factor.

Variable	Description
CMP_CPCL_NO_ROTATION	Print text with no rotation.
CMP_CPCL_90_ROTATION	Print text with 90 rotation.(counterclockwise)
CMP_CPCL_180_ROTATION	Print text with 180 rotation.(counterclockwise)
CMP_CPCL_270_ROTATION	Print text with 270 rotation.(counterclockwise)

* FontType

- It receives the font type of the text to print as a factor.

* FontSize

- It receives the font size value of the text to print as a factor.

* 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.

* Data

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

2.14. MultiLineData

This function is used for printing text to print the value which is defined in MultiLineText().

```
void MultiLineData(String Data)
```

[Parameter]

* Data

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

2.15. ResetMultiLine

This function is used for canceling the defined function in SetMultiLine().

```
void ResetMultiLine()
```

2.16. SetMagnify

This function is used for setting the magnification of text.

```
void SetMagnify(int Width,int Height)
```

[Parameter]

* Width

- It received the width ratio of text as a factor.

Variable (Set up width ratio)	Description
CMP_CPCL_TXT_1WIDTH	Set up width ratio as x1
CMP_CPCL_TXT_2WIDTH	Set up width ratio as x2
CMP_CPCL_TXT_3WIDTH	Set up width ratio as x3
CMP_CPCL_TXT_4WIDTH	Set up width ratio as x4
CMP_CPCL_TXT_5WIDTH	Set up width ratio as x5
CMP_CPCL_TXT_6WIDTH	Set up width ratio as x6
CMP_CPCL_TXT_7WIDTH	Set up width ratio as x7
CMP_CPCL_TXT_8WIDTH	Set up width ratio as x8
CMP_CPCL_TXT_9WIDTH	Set up width ratio as x9
CMP_CPCL_TXT_10WIDTH	Set up width ratio as x10
CMP_CPCL_TXT_11WIDTH	Set up width ratio as x11
CMP_CPCL_TXT_12WIDTH	Set up width ratio as x12
CMP_CPCL_TXT_13WIDTH	Set up width ratio as x13
CMP_CPCL_TXT_14WIDTH	Set up width ratio as x14
CMP_CPCL_TXT_15WIDTH	Set up width ratio as x15
CMP_CPCL_TXT_16WIDTH	Set up width ratio as x16

* Height

- It received the height ratio of text as a factor.

Variable (Set up height ratio)	Description
CMP_CPCL_TXT_1HEIGHT	Set up height ratio as x1

CMP_CPCL_TXT_2HEIGHT	Set up height ratio as x2
CMP_CPCL_TXT_3HEIGHT	Set up height ratio as x3
CMP_CPCL_TXT_4HEIGHT	Set up height ratio as x4
CMP_CPCL_TXT_5HEIGHT	Set up height ratio as x5
CMP_CPCL_TXT_6HEIGHT	Set up height ratio as x6
CMP_CPCL_TXT_7HEIGHT	Set up height ratio as x7
CMP_CPCL_TXT_8HEIGHT	Set up height ratio as x8
CMP_CPCL_TXT_9HEIGHT	Set up height ratio as x9
CMP_CPCL_TXT_10HEIGHT	Set up height ratio as x10
CMP_CPCL_TXT_11HEIGHT	Set up height ratio as x11
CMP_CPCL_TXT_12HEIGHT	Set up height ratio as x12
CMP_CPCL_TXT_13HEIGHT	Set up height ratio as x13
CMP_CPCL_TXT_14HEIGHT	Set up height ratio as x14
CMP_CPCL_TXT_15HEIGHT	Set up height ratio as x15
CMP_CPCL_TXT_16HEIGHT	Set up height ratio as x16

2.17. ResetMagnify

This function is used for canceling the defined function in SetMagnify().

```
void ResetMagnify()
```

2.18. PrintCPCLBarCode

This function is used for printing barcode.

```
void PrintCPCLBarCode(int Rotation,int BarCodeType,int NB,int Ratio,int BarHeight,  
int PrintX,int PrintY,String Data,int count)
```

[Parameter]

* Rotation

- It receives the printing direction value of the barcode to print as a factor.

Variable	Description
CMP_CPCL_NO_ROTATION	Print barcode with no rotation.
CMP_CPCL_90_ROTATION	Print barcode with 90 rotation.(counterclockwise)
CMP_CPCL_180_ROTATION	Print barcode with 180 rotation. (counterclockwise)
CMP_CPCL_270_ROTATION	Print text with 270 rotation.(counterclockwise)

* BarCodeType

- It receives the barcode type to print as a factor.

Variable	Description
CMP_CPCL_BCS_39	Barcode 3 of 9 Standard.
CMP_CPCL_BCS_39C	Barcode 3 of 9 Standard with Check Digit.
CMP_CPCL_BCS_39F	Barcode 3 of 9 Full ASCII.
CMP_CPCL_BCS_39FC	Barcode 3 of 9 Full ASCII with Check Digit.

CMP_CPCL_BCS_93	Barcode Code 93.
CMP_CPCL_BCS_128	Barcode Code 128.
CMP_CPCL_BCS_EAN128	Barcode UCC-128.
CMP_CPCL_BCS_CODABAR	Barcode Codabar.
CMP_CPCL_BCS_CODABARC	Barcode Codabar with Check Digit.
CMP_CPCL_BCS_EAN8	Barcode EAN8.
CMP_CPCL_BCS_EAN82	Barcode EAN8 2-Digit Add-On.
CMP_CPCL_BCS_EAN85	Barcode EAN8 5-Digit Add-On.
CMP_CPCL_BCS_EAN13	Barcode EAN13.
CMP_CPCL_BCS_EAN132	Barcode EAN13 2-Digit Add-On.
CMP_CPCL_BCS_EAN135	Barcode EAN13 5-Digit Add-On.
CMP_CPCL_BCS_I2OF5	Barcode Interleaved 2 of 5.
CMP_CPCL_BCS_POSTNET	Barcode Postnet
CMP_CPCL_BCS_UPCA	Barcode UPCA.
CMP_CPCL_BCS_UPCA2	Barcode UPCA 2-Digit Add-On.
CMP_CPCL_BCS_UPCA5	Barcode UPCA 5-Digit Add-On.
CMP_CPCL_BCS_UPCE	Barcode UPCE.
CMP_CPCL_BCS_UPCE2	Barcode UPCE 2-Digit Add-On.
CMP_CPCL_BCS_UPCE5	Barcode UPCE 5-Digit Add-On.
CMP_CPCL_BCS_MSI	Barcode Plessey(MSI-1).
CMP_CPCL_BCS_MSI1C	Barcode Plessey(MSI-1) with Check Digit.
CMP_CPCL_BCS_MSI2C	Barcode Plessey(MSI-2) with Check Digit.
CMP_CPCL_BCS_MSI11C	Barcode Plessey(MSI-11) with Check Digit.
CMP_CPCL_BCS_PLUS2	Plus 2 Extension.
CMP_CPCL_BCS_PLUS5	Plus 5 Extension.

* NB

- It receives the width of the little bar of barcode as a factor.

* Ratio

- It receives the ratio value of the barcode as a factor.

Variable	Description
CMP_CPCL_BCS_0RATIO	Set up the barcode ratio as 1.5 : 1.
CMP_CPCL_BCS_1RATIO	Set up the barcode ratio as 2.0 : 1.
CMP_CPCL_BCS_2RATIO	Set up the barcode ratio as 2.5 : 1.
CMP_CPCL_BCS_3RATIO	Set up the barcode ratio as 3.0 : 1.
CMP_CPCL_BCS_4RATIO	Set up the barcode ratio as 3.5 : 1.
CMP_CPCL_BCS_20RATIO	Set up the barcode ratio as 2.0 : 1.
CMP_CPCL_BCS_21RATIO	Set up the barcode ratio as 2.1 : 1.
CMP_CPCL_BCS_22RATIO	Set up the barcode ratio as 2.2 : 1.
CMP_CPCL_BCS_23RATIO	Set up the barcode ratio as 2.3 : 1.
CMP_CPCL_BCS_24RATIO	Set up the barcode ratio as 2.4 : 1.
CMP_CPCL_BCS_25RATIO	Set up the barcode ratio as 2.5 : 1.
CMP_CPCL_BCS_26RATIO	Set up the barcode ratio as 2.6 : 1.
CMP_CPCL_BCS_27RATIO	Set up the barcode ratio as 2.7 : 1.

CMP_CPCL_BCS_28RATIO	Set up the barcode ratio as 2.8 : 1.
CMP_CPCL_BCS_29RATIO	Set up the barcode ratio as 2.9 : 1.
CMP_CPCL_BCS_30RATIO	Set up the barcode ratio as 3.0 : 1.

* BarHeight

- It receives the height value of the barcode to print as a factor.

* PrintX

- It receives the starting point of x-coordinates of the barcode to print as a factor.

* PrintY

- It receives the starting point of y-coordinates of the barcode to print as a factor.

* Data

- It receives the barcode data to print as a factor.

* Count

- It receives the count function of the barcode to print as a factor.

2.19. PrintBox

This function is used for printing box image.

```
void PrintBox(int xs,int ys,int xx,int yx,int Thickness)
```

[Parameter]

* xs

- It receives the starting point of x-coordinates for box printing as a factor.

* ys

- It receives the starting point of y-coordinates for box printing as a factor.

* xx

- It receives the ending point of x-coordinates for box printing as a factor.

* yx

- It receives the ending point of y-coordinates for box printing as a factor.

* Thickness

- It receives the thickness of the box line to print as a factor.

2.20. PrintLine

This function is used for printing line image.

```
void PrintLine(int xs,int ys,int xx,int yx,int Thickness)
```

[Parameter]

* xs

- It receives the starting point of x-coordinates for line printing as a factor.

* ys

- It receives the starting point of y-coordinates for line printing as a factor.

* xx

- It receives the ending point of x-coordinates for line printing as a factor.

* yx

- It receives the ending point of y-coordinates for line printing as a factor.

* Thickness

- It receives the thickness of the line to print as a factor.

2.21. InverseLine

This function is used for the inverse image in special area. It has Syntax like PrintLine() Method.

void InverseLine(int xs,int ys,int xx,int yx,int Thickness)

[Parameter]

* xs

- It receives the starting point of x-coordinates for inverse image printing as a factor.

* ys

- It receives the starting point of y-coordinates for inverse image printing as a factor.

* xx

- It receives the ending point of x-coordinates for inverse image printing as a factor.

* yx

- It receives the ending point of y-coordinates for inverse image printing as a factor.

* Thickness

- It receives the thickness of the line for inverse image printing as a factor.

2.22. SetPattern

This function is used for pattern printing.

void SetPattern(int PatternNum)

[Parameter]

* PatternNum

- It receives the value of the pattern to print as a factor.

Variable	Description
CMP_CPCL_DEFAULT_PATTERN	Filled(Black/default value)
CMP_CPCL_HORIZON_PATTERN	It prints the pattern as a horizontal line.
CMP_CPCL_VERTICAL_PATTERN	It prints the pattern as a vertical line.
CMP_CPCL_RDIAGON_PATTERN	It prints the diagonal pattern to the right.
CMP_CPCL_LDIAGON_PATTERN	It prints the diagonal pattern to the left.

CMP_CPCL_SQUARE_PATTERN	It prints the pattern as a square.
CMP_CPCL_CROSS_PATTERN	It prints the pattern as a diagonal line to right and left.

2.23. PrintBitmap

This function is used for Bit-mapped image printing.

Supported graphic format is BMP/JPEG/PNG/GIF.

void PrintBitmap(String FilePath,int PrintX,int PrintY)

void PrintBitmap(Bitmap Bmp,int PrintX,int PrintY)

[Parameter]

* FilePath

- It receives the filepath of the image to print as a factor.

* Bmp

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

* PrintX

- It receives the starting point of x-coordinates on the image to print as a factor.

* PrintY

- It receives the starting point of y-coordinates on the image to print as a factor.

2.24. SetContrast

This function is used for adjusting contrast of the whole label to print.

void SetContrast(int Darkness)

[Parameter]

* Darkness

- It receives the contrast value of the whole label to print as a factor. [0 <= Darkness <=3]

Variable	Description
CMP_CPCL_CONT_DEFAULT	Set up as default
CMP_CPCL_CONT_MEDIUM	Print as middle brightness
CMP_CPCL_CONT_DARK	Print as dark
CMP_CPCL_CONT_VERY_DARK	Print as very dark.

2.25. SetPageWidth

This function is used for defining the width of the paper form.

void SetPageWidth(int PageWidth)

[Parameter]

* PageWidth

- It receives the page width value of the label to print as a factor.

2.26. PrintCPCLImage

This function is used for printing pcx image stored in flash file system of the printer. Reference to Desktop Application Manual.

```
void PrintCPCLImage(String ImageName,int PrintX,int PrintY)
```

[Parameter]

* ImageName

- Unicode which has a null-terminated string.

It receives the pcx file name stored in flash file system as a factor.

* PrintX

- It receives the starting point of x-coordinates to print as a factor.

* PrintY

- It receives the starting point of y-coordinates to print as a factor.

2.27. SetSpeed

This function is used for defining printing speed.

```
void SetSpeed(int Speed)
```

[Parameter]

* Speed

- It receives the printing speed as a factor. [0 =< Speed =<5]

2.28. SetTone

This function is the Method which is used instead of SetContrast() and it is used for adjusting the darkness of the whole label to print

```
void SetTone(int Tone)
```

[Parameter]

* Tone

- It receives the tone of darkness of the whole label as a factor. [-99 =< Tone=< 200]

2.29. SetCPCLBarCode

This function is used for setting HRI character information.

```
void SetCPCLBarCode(int FontNum,int FontSize,String Offset)
```

[Parameter]

* FontNum

- It receives the font type of the text to print as a factor.

* FontSize

- It receives the font size value of the text to print as a factor.

* Offset

- It receive the offset between barcode and font of the text to print as as factor.

2.30. PrintCPCL2DBarCode

This function is used for printing two-dimension barcode.

```
void PrintCPCL2DBarCode(int Rotation,int BarCodeType,String PrintX,String PrintY,
int UnitWidth,int UnitHeight,int Column,int SecurityLevel,String Data)
```

[Parameter]

* Rotation

- It receives the printing direction value of the barcode to print as a factor.

Variable	Description
CMP_CPCL_NO_ROTATION	Print text with no rotation.
CMP_CPCL_90_ROTATION	Print text with 90 rotation.(counterclockwise)
CMP_CPCL_180_ROTATION	Print text with 180 rotation.(counterclockwise)
CMP_CPCL_270_ROTATION	Print text with 270 rotation.(counterclockwise)

* BarCodeType

- It receives the barcode type to print as a factor.

Variable	Description
CMP_CPCL_BCS_PDF417	PDF417 Barcode.
CMP_CPCL_BCS_QRCODE	QRCode Barcode.
CMP_CPCL_BCS_DATAMATRIX	DataMatrix Barcode.

* PrintX

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

* PrintY

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

* UnitWidth

- According to BarCodeType.

BarCodeType	Description
CMP_CPCL_BCS_PDF417	Cell Width. Range is 1 to 32. Default is 2.
CMP_CPCL_BCS_QRCODE	Cell Width. Range is 1 to 24. Auto = 4.
CMP_CPCL_BCS_DATAMATRIX	Cell Width. Range is 1 to 24. Auto = 4.

* UnitHeight

- According to BarCodeType.

BarCodeType	Description
CMP_CPCL_BCS_PDF417	Cell Height. Range is 1 to 32. Default is 6.
CMP_CPCL_BCS_QRCODE	QRCode Version. Range is 1 to 40. Auto = 0.
CMP_CPCL_BCS_DATAMATRIX	Default(Zero).

* Column

- According to BarCodeType.

BarCodeType	Description
CMP_CPCL_BCS_PDF417	Number of columns to use. Range is 1 to 30. Default is 3.
CMP_CPCL_BCS_QRCODE	Error Correction Level. Range is 0 to 3. Default is 1.
CMP_CPCL_BCS_DATAMATRIX	Error Correction Level. Range is 0, 50, 80, 100, 140, 200.

* SecurityLevel

- According to BarCodeType.

BarCodeType	Description
CMP_CPCL_BCS_PDF417	Security level indicates maximum amount of errors to be detected and/or corrected. Range is 0 to 8. Default is 1.
CMP_CPCL_BCS_QRCODE	Mask Pattern. Range is 0 to 7. Auto = 8.
CMP_CPCL_BCS_DATAMATRIX	Default(Zero).

* Data

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

2.31. SetMedia

This function is used to set Label Type to print.

```
void SetMedia(int mode)
```

[Parameter]

* Mode

- It sets the Label Type.

Variable	Description
CMP_CPCL_LABEL	Label with Gap.
CMP_CPCL_BLACKMARK	Label with Black Mark.
CMP_CPCL_CONTINUOUS	Continuous Label

2.32. SetCountry

This function is used to set the appropriate character set for the specified country.

```
void SetCountry(String country)
```

[Parameter]

* country

- It sets the Country.

Variable	Description
CMP_CPCL_COUNTRY_USA	USA
CMP_CPCL_COUNTRY_GERMANY	GERMANY
CMP_CPCL_COUNTRY_FRANCE	FRANCE
CMP_CPCL_COUNTRY_SWEDEN	SWEDEN
CMP_CPCL_COUNTRY_SPAIN	SPAIN
CMP_CPCL_COUNTRY_NORWAY	NORWAY
CMP_CPCL_COUNTRY_ITALY	ITALY
CMP_CPCL_COUNTRY_UK	UK
CMP_CPCL_COUNTRY_CP850	CP850
CMP_CPCL_COUNTRY_LATIN9	LATIN9

2.33. ResetCountry

This function is used for resetting the country setting.

```
void ResetCountry()
```

2.34. Temperature

This function is used for getting the printing head temperature.

```
double temperature()
```

[Return Values]

Temperature value.

2.35. Voltage

This function is used for getting the battery voltage.

```
double voltage()
```

[Return Values]

Voltage value.

2.36. 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

2.37. PrinterResults ← Added in 1.073

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_CPCL_NORMAL	Printing is success.
CMP_STS_CPCL_PAPER_EMPTY	Printer Status is no paper.
CMP_STS_CPCL_COVER_OPEN	Printer Cover is open.
CMP_STS_CPCL_BATTERY_LOW	Printer battery capacity is low.
CMP_STS_CPCL_PRINTEROFF	Printer Status is off.
CMP_STS_CPCL_TIMEOUT	Printer not response.
CMP_FAIL	Function call failed.
Other values	Printer status is unknown error.

2.38. PrintAndroidFont ← Added in 1.071

This function is used for android embedded font printing.

```
void PrintAndroidFont(String textString,int widthDots, int textSize, int PrintY, int alignment)
```

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

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

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

```
void PrintAndroidFont(Typeface typeface, Boolean isBold, Boolean isItalic, boolean isUnderline,
```

String textString, int widthDots, int textSize, int PrintY, 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]

* PrintY

- It receives the starting point of y-coordinates on the image to print as a factor.

* 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.

2.39 printAndroidFont ← Added in 1.080

This function is used for android embedded font printing with x, y coordination.

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

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

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

void PrintAndroidFont(int printX, int printY, Typeface typeface, Boolean isBold, Boolean isItalic,

String textString, int widthDots, int textSize)

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

[Parameter]

* printX, printY

- Set the start x, y coordination of font position. (Unit is dot)

* 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. (Unit is dot)

* textSize

- It receives the font size value of the text to print as a factor. (Unit is point)

* 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.