

Android Module Program Manual

CPCL

Mobile Printer

Rev. 1.099

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) ,

CPCLPrinter(DeviceConnection connection) ,

CPCLPrinter(String charset, DeviceConnection connection)

[Parameter]

* charset

- Character set name.

* connection

- Device connection. (USBPortConnection, WiFiMultiConnection)

2.2. SetForm

This function is used for defining paper form.

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

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

String LabelWidth, int Quantity) ← **Added in 1.097**

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

String LabelWidth, int rotate, int Quantity) ← **Added in 1.097**

[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.

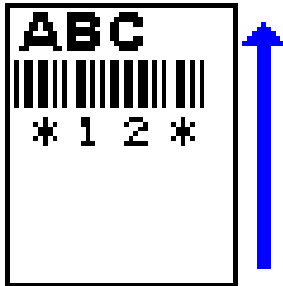
* LabelWidth

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

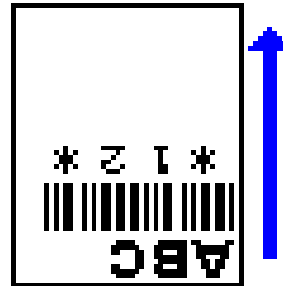
* rotate

- It receives label rotation to print as a factor.[0=Normal, 1=180-degree rotation]

Case rotate = 0



Case rotate = 1



* 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) ← **Added in 1.077**

[Parameter]

* Timeout : milliseconds. (Default : 5000 ms)

[Return Values]

AURES_SUCCESS : This value returns when a function succeeds.

AURES_FAIL : This value returns when a function fails.

2.5. status

This function is used for getting the printer status.

int status()

[Return Values]

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

AURES_STS_CPCL_BUSY : Printer Status is busy status.

AURES_STS_CPCL_PAPER_EMPTY : Printer Status is no paper.

AURES_STS_CPCL_COVER_OPEN : Printer Cover is open.

AURES_STS_CPCL_BATTERY_LOW : Printer battery capacity is low.

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 |
|------------------|--------------------------------|
| AURES_CPCL_INCH | Set up measure as inch. |
| AURES_CPCL_CENTI | Set up measure as centimeter |
| AURES_CPCL_MILLI | Set up measure as millimeter |
| AURES_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 |
|-------------------|----------------------|
| AURES_CPCL_LEFT | Left justification |
| AURES_CPCL_CENTER | Center justification |
| AURES_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 |
|-------------------------|---|
| AURES_CPCL_NO_ROTATION | Print text with no rotation. |
| AURES_CPCL_90_ROTATION | Print text with 90 rotation.(counterclockwise) |
| AURES_CPCL_180_ROTATION | Print text with 180 rotation.(counterclockwise) |
| AURES_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 |
|--------------------|------------------------------------|
| AURES_CPCL_CONCAT | It concatenates text as horizontal |
| AURES_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 |
|-------------------------|---|
| AURES_CPCL_NO_ROTATION | Print text with no rotation. |
| AURES_CPCL_90_ROTATION | Print text with 90 rotation.(counterclockwise) |
| AURES_CPCL_180_ROTATION | Print text with 180 rotation.(counterclockwise) |
| AURES_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 |
|-------------------------------|--------------------------|
| AURES_CPCL_TXT_1WIDTH | Set up width ratio as x1 |
| AURES_CPCL_TXT_2WIDTH | Set up width ratio as x2 |
| AURES_CPCL_TXT_3WIDTH | Set up width ratio as x3 |
| AURES_CPCL_TXT_4WIDTH | Set up width ratio as x4 |
| AURES_CPCL_TXT_5WIDTH | Set up width ratio as x5 |
| AURES_CPCL_TXT_6WIDTH | Set up width ratio as x6 |

| | |
|------------------------|---------------------------|
| AURES_CPCL_TXT_7WIDTH | Set up width ratio as x7 |
| AURES_CPCL_TXT_8WIDTH | Set up width ratio as x8 |
| AURES_CPCL_TXT_9WIDTH | Set up width ratio as x9 |
| AURES_CPCL_TXT_10WIDTH | Set up width ratio as x10 |
| AURES_CPCL_TXT_11WIDTH | Set up width ratio as x11 |
| AURES_CPCL_TXT_12WIDTH | Set up width ratio as x12 |
| AURES_CPCL_TXT_13WIDTH | Set up width ratio as x13 |
| AURES_CPCL_TXT_14WIDTH | Set up width ratio as x14 |
| AURES_CPCL_TXT_15WIDTH | Set up width ratio as x15 |
| AURES_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 |
|--------------------------------|----------------------------|
| AURES_CPCL_TXT_1HEIGHT | Set up height ratio as x1 |
| AURES_CPCL_TXT_2HEIGHT | Set up height ratio as x2 |
| AURES_CPCL_TXT_3HEIGHT | Set up height ratio as x3 |
| AURES_CPCL_TXT_4HEIGHT | Set up height ratio as x4 |
| AURES_CPCL_TXT_5HEIGHT | Set up height ratio as x5 |
| AURES_CPCL_TXT_6HEIGHT | Set up height ratio as x6 |
| AURES_CPCL_TXT_7HEIGHT | Set up height ratio as x7 |
| AURES_CPCL_TXT_8HEIGHT | Set up height ratio as x8 |
| AURES_CPCL_TXT_9HEIGHT | Set up height ratio as x9 |
| AURES_CPCL_TXT_10HEIGHT | Set up height ratio as x10 |
| AURES_CPCL_TXT_11HEIGHT | Set up height ratio as x11 |
| AURES_CPCL_TXT_12HEIGHT | Set up height ratio as x12 |
| AURES_CPCL_TXT_13HEIGHT | Set up height ratio as x13 |
| AURES_CPCL_TXT_14HEIGHT | Set up height ratio as x14 |
| AURES_CPCL_TXT_15HEIGHT | Set up height ratio as x15 |
| AURES_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 |
|-------------------------|---|
| AURES_CPCL_NO_ROTATION | Print barcode with no rotation. |
| AURES_CPCL_90_ROTATION | Print barcode with 90 rotation.(counterclockwise) |
| AURES_CPCL_180_ROTATION | Print barcode with 180 rotation. (counterclockwise) |
| AURES_CPCL_270_ROTATION | Print text with 270 rotation.(counterclockwise) |

*** BarCodeType**

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

| Variable | Description |
|-------------------------|---|
| AURES_CPCL_BCS_39 | Barcode 3 of 9 Standard. |
| AURES_CPCL_BCS_39C | Barcode 3 of 9 Standard with Check Digit. |
| AURES_CPCL_BCS_39F | Barcode 3 of 9 Full ASCII. |
| AURES_CPCL_BCS_39FC | Barcode 3 of 9 Full ASCII with Check Digit. |
| AURES_CPCL_BCS_93 | Barcode Code 93. |
| AURES_CPCL_BCS_128 | Barcode Code 128. |
| AURES_CPCL_BCS_EAN128 | Barcode UCC-128. |
| AURES_CPCL_BCS_CODABAR | Barcode Codabar. |
| AURES_CPCL_BCS_CODABARC | Barcode Codabar with Check Digit. |
| AURES_CPCL_BCS_EAN8 | Barcode EAN8. |
| AURES_CPCL_BCS_EAN82 | Barcode EAN8 2-Digit Add-On. |
| AURES_CPCL_BCS_EAN85 | Barcode EAN8 5-Digit Add-On. |
| AURES_CPCL_BCS_EAN13 | Barcode EAN13. |
| AURES_CPCL_BCS_EAN132 | Barcode EAN13 2-Digit Add-On. |
| AURES_CPCL_BCS_EAN135 | Barcode EAN13 5-Digit Add-On. |
| AURES_CPCL_BCS_I2OF5 | Barcode Interleaved 2 of 5. |
| AURES_CPCL_BCS_POSTNET | Barcode Postnet |
| AURES_CPCL_BCS_UPCA | Barcode UPCA. |
| AURES_CPCL_BCS_UPCA2 | Barcode UPCA 2-Digit Add-On. |
| AURES_CPCL_BCS_UPCA5 | Barcode UPCA 5-Digit Add-On. |
| AURES_CPCL_BCS_UPCE | Barcode UPCE. |
| AURES_CPCL_BCS_UPCE2 | Barcode UPCE 2-Digit Add-On. |
| AURES_CPCL_BCS_UPCE5 | Barcode UPCE 5-Digit Add-On. |
| AURES_CPCL_BCS_MSI | Barcode Plessey(MSI-1). |
| AURES_CPCL_BCS_MSI1C | Barcode Plessey(MSI-1) with Check Digit. |
| AURES_CPCL_BCS_MSI2C | Barcode Plessey(MSI-2) with Check Digit. |
| AURES_CPCL_BCS_MSI11C | Barcode Plessey(MSI-11) with Check Digit. |
| AURES_CPCL_BCS_PLUS2 | Plus 2 Extension. |
| AURES_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 |
|------------------------|--------------------------------------|
| AURES_CPCL_BCS_0RATIO | Set up the barcode ratio as 1.5 : 1. |
| AURES_CPCL_BCS_1RATIO | Set up the barcode ratio as 2.0 : 1. |
| AURES_CPCL_BCS_2RATIO | Set up the barcode ratio as 2.5 : 1. |
| AURES_CPCL_BCS_3RATIO | Set up the barcode ratio as 3.0 : 1. |
| AURES_CPCL_BCS_4RATIO | Set up the barcode ratio as 3.5 : 1. |
| AURES_CPCL_BCS_20RATIO | Set up the barcode ratio as 2.0 : 1. |
| AURES_CPCL_BCS_21RATIO | Set up the barcode ratio as 2.1 : 1. |
| AURES_CPCL_BCS_22RATIO | Set up the barcode ratio as 2.2 : 1. |
| AURES_CPCL_BCS_23RATIO | Set up the barcode ratio as 2.3 : 1. |
| AURES_CPCL_BCS_24RATIO | Set up the barcode ratio as 2.4 : 1. |
| AURES_CPCL_BCS_25RATIO | Set up the barcode ratio as 2.5 : 1. |
| AURES_CPCL_BCS_26RATIO | Set up the barcode ratio as 2.6 : 1. |
| AURES_CPCL_BCS_27RATIO | Set up the barcode ratio as 2.7 : 1. |
| AURES_CPCL_BCS_28RATIO | Set up the barcode ratio as 2.8 : 1. |
| AURES_CPCL_BCS_29RATIO | Set up the barcode ratio as 2.9 : 1. |
| AURES_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 |
|-----------------------------|---|
| AURES_CPCL_DEFAULT_PATTERN | Filled(Black/default value) |
| AURES_CPCL_HORIZON_PATTERN | It prints the pattern as a horizontal line. |
| AURES_CPCL_VERTICAL_PATTERN | It prints the pattern as a vertical line. |
| AURES_CPCL_RDIAGON_PATTERN | It prints the diagonal pattern to the right. |
| AURES_CPCL_LDIAGON_PATTERN | It prints the diagonal pattern to the left. |
| AURES_CPCL_SQUARE_PATTERN | It prints the pattern as a square. |
| AURES_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.

```
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/JPEG/PNG/GIF]

* 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 |
|----------|-------------|
|----------|-------------|

| | |
|---------------------------|----------------------------|
| AURES_CPCL_CONT_DEFAULT | Set up as default |
| AURES_CPCL_CONT_MEDIUM | Print as middle brightness |
| AURES_CPCL_CONT_DARK | Print as dark |
| AURES_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 |
|-------------------------|---|
| AURES_CPCL_NO_ROTATION | Print text with no rotation. |
| AURES_CPCL_90_ROTATION | Print text with 90 rotation.(counterclockwise) |
| AURES_CPCL_180_ROTATION | Print text with 180 rotation.(counterclockwise) |
| AURES_CPCL_270_ROTATION | Print text with 270 rotation.(counterclockwise) |

* BarCodeType

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

| Variable | Description |
|---------------------------|---------------------|
| AURES_CPCL_BCS_PDF417 | PDF417 Barcode. |
| AURES_CPCL_BCS_QRCODE | QRCode Barcode. |
| AURES_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 |
|---------------------------|---|
| AURES_CPCL_BCS_PDF417 | Cell Width. Range is 1 to 32. Default is 2. |
| AURES_CPCL_BCS_QRCODE | Cell Width. Range is 1 to 24. Auto = 4. |
| AURES_CPCL_BCS_DATAMATRIX | Cell Width. Range is 1 to 24. Auto = 4. |

* UnitHeight

- According to BarCodeType.

| BarCodeType | Description |
|---------------------------|--|
| AURES_CPCL_BCS_PDF417 | Cell Height. Range is 1 to 32. Default is 6. |
| AURES_CPCL_BCS_QRCODE | QRCode Version. Range is 1 to 40. Auto = 0. |
| AURES_CPCL_BCS_DATAMATRIX | Default(Zero). |

* Column

- According to BarCodeType.

| BarCodeType | Description |
|---------------------------|--|
| AURES_CPCL_BCS_PDF417 | Number of columns to use. Range is 1 to 30. Default is 3. |
| AURES_CPCL_BCS_QRCODE | Error Correction Level. Range is 0 to 3. Default is 1. |
| AURES_CPCL_BCS_DATAMATRIX | Error Correction Level. Range is 0, 50, 80, 100, 140, 200. |

* SecurityLevel

- According to BarCodeType.

| BarCodeType | Description |
|---------------------------|---|
| AURES_CPCL_BCS_PDF417 | Security level indicates maximum amount of errors to be detected and/or corrected. Range is 0 to 8. Default is 1. |
| AURES_CPCL_BCS_QRCODE | Mask Pattern. Range is 0 to 7. Auto = 8. |
| AURES_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 |
|-----------------------|------------------------|
| AURES_CPCL_LABEL | Label with Gap. |
| AURES_CPCL_BLACKMARK | Label with Black Mark. |
| AURES_CPCL_CONTINUOUS | Continuous Label |

2.32. PrinterResults ← Modified 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.

[Return Values]

| Variable | Description |
|-------------------------|----------------------------------|
| AURES_STS_CPCL_NORMAL | Printing is success. |
| LK_STS_CPCL_PAPER_EMPTY | Printer Status is no paper. |
| LK_STS_CPCL_COVER_OPEN | Printer Cover is open. |
| LK_STS_CPCL_BATTERY_LOW | Printer battery capacity is low. |
| Other values | Printer status is unknown error. |

2.33. PrintAndroidFont ← Added in 1.075

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 |
|------------------------|------------------|
| 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.

2.34 printAndroidFont ← Added in 1.085

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.