Cordova Plugin Manual

ESC/POS

Mobile Printer

Rev. 1.0.0

CONTENTS

1. Instruction

This Cordova Plugin Manual describes the method which is exposed from javascript(.js) file needed in developing Cordova application.

2. Method.

Defined in the CitizenPluginService.js file.

2.1. getPairedBT

This function used to get the paired Bluetooth list.

citizen_plugin.getPairedBT = function(successCallback, errorCallback);

[Parameter]

* none

2.2. connect

This function used to connect printer.

citizen_plugin.connect= function(successCallback, errorCallback,
address);

[Parameter]

* address
  - Bluetooth mac address.

2.3. disconnect

This function used to disconnect printer.

citizen_plugin.disconnect= function(successCallback, errorCallback);

[Parameter]

* none

2.4. getStatus

This function used to get the status of printer.

citizen_plugin.getStatus= function(successCallback, errorCallback);

[Parameter]

* none

[Return Value]

CMP_STS_NORMAL(0): Printer Status is No Error and MSR is not Ready.

CMP_BATTERY_LOW(8) : Printer battery capacity is low.

CMP_STS_MSR_READ(64) : Currently MSR in read mode, printing is impossible.

CMP_PAPER_EMPTY(32) : Printer Status is no paper.

CMP_COVER_OPEN(16) : Printer Cover is open.

CMP_FAIL(-1) : This value returns when a function fails.

CMP_STS_PRINTEROFF(128) : This value returns when printer is off.

CMP_STS_TIMEOUT(256) : This value returns when printer is no response or power off.

(CMP_STS_PRINTEROFF is returned when getStatus function is used while printer is off. CMP_STS_TIMEOUT is returned when getStatus function is used again)

2.5. printNormal

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

citizen_plugin.printNormal= function(successCallback, errorCallback,
data);

[Parameter]

* data

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

2.6. printText

This function used for supporting text printing

```
citizen_plugin.printText= function(successCallback, errorCallback,
            data, alignment, attribute, textSize);
```

[Parameter]

* data

- It sets text data to print.

* alignment

- This value is alignment. It sets text alignment.

Variable	Description
CMP_ALIGNMENT_LEFT(0)	Left alignment
CMP_ALIGNMENT_CENTER(1)	Center alignment
CMP_ALIGNMENT_RIGHT(2)	Right alignment

* attribute

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

Variable	Description
CMP_FNT_DEFAULT(0)	FontA, Set up as a standard
CMP_FNT_FONTB(1)	Set up as FontB
CMP_FNT_BOLD(8)	Set up as Bold attribute
CMP_FNT_UNDERLINE(128)	Set up as Underline attribute (1dot)
CMP_FNT_UNDERLINE2(256)	Set up as Underline attribute (2dot)
CMP_FNT_REVERSE(16)	Set up as reverse print attribute

* textSize

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

Variable (Set up width ratio)	Description
CMP_TXT_1WIDTH(0)	Set up width ratio as x1
CMP_TXT_2WIDTH(16)	Set up width ratio as x2
CMP_TXT_3WIDTH(32)	Set up width ratio as x3
CMP_TXT_4WIDTH(48)	Set up width ratio as x4
CMP_TXT_5WIDTH(64)	Set up width ratio as x5
CMP_TXT_6WIDTH(80)	Set up width ratio as x6
CMP_TXT_7WIDTH(96)	Set up width ratio as x7
CMP_TXT_8WIDTH(112)	Set up width ratio as x8

Variable (Set up height ratio)	Description
CMP_TXT_1HEIGHT(0)	Set up height ratio as x1
CMP_TXT_2HEIGHT(1)	Set up height ratio as x2
CMP_TXT_3HEIGHT(2)	Set up height ratio as x3
CMP_TXT_4HEIGHT(3)	Set up height ratio as x4
CMP_TXT_5HEIGHT(4)	Set up height ratio as x5
CMP_TXT_6HEIGHT(5)	Set up height ratio as x6
CMP_TXT_7HEIGHT(6)	Set up height ratio as x7
CMP_TXT_8HEIGHT(7)	Set up height ratio as x8

[Example]

```
citizen_plugin.printText( function(result) { }, function(err) { },
            data, 1, 0, 0 | 0 );
```

***You can also define the parameter values above.**

2.7. printBarCode

This function used for supporting 1D-barcode printing.

```
citizen_plugin.printBarCode= function(successCallback, errorCallback,
            data, symbology, height, width, alignment, hri);
```

[Parameter]

* data

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

* symbology

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

Variable	Description
CMP_BCS_UPCA(101)	Print UPC A BarCode
CMP_BCS_UPCE(102)	Print UPC E BarCode
CMP_BCS_EAN8(103)	Print EAN-8 BarCode
CMP_BCS_EAN13(104)	Print EAN-13 BarCode
CMP_BCS_JAN8(105)	Print JAN-8 BarCode
CMP_BCS_JAN13(106)	Print JAN-13 BarCode
CMP_BCS_ITF(107)	Print Interleaved 2 of 5
CMP_BCS_Codabar(108)	Print Codabar BarCode
CMP_BCS_Code39(109)	Print Code 3 of 9 BarCode
CMP_BCS_Code93(110)	Print Code 93 BarCode
CMP_BCS_Code128(111)	Print Code 128 BarCode

* height

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

* width

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

* alignment

- This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT(0)	Left alignment
CMP_ALIGNMENT_CENTER(1)	Center alignment
CMP_ALIGNMENT_RIGHT(2)	Right alignment

* hri

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

Variable	Description
CMP_HRI_TEXT_NONE(0)	Do not print barcode data
CMP_HRI_TEXT_ABOVE(1)	Print barcode data above the barcode
CMP_HRI_TEXT_BELOW(2)	Print barcode data below the barcode

[Example]

```
citizen_plugin.printBarCode( function(result) { }, function(err) { },
            data, 109, 40, 2, 1, 2 );
```

***You can also define the parameter values above.**

2.8. printImage

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

```
citizen_plugin.printImage= function(successCallback, errorCallback,
            imagePath, alignment, size);
```

[Parameter]

* imagePath

- This value is the bitmap file name with full path of bitmap file.

* alignment

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

Variable	Description
CMP_ALIGNMENT_LEFT(0)	Left alignment
CMP_ALIGNMENT_CENTER(1)	Center alignment
CMP_ALIGNMENT_RIGHT(2)	Right alignment

* size

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

Variable	Description
CMP_BITMAP_NORMAL(0)	Normal size
CMP_BITMAP_DOUBLE_WIDTH(1)	Double width
CMP_BITMAP_DOUBLE_HEIGHT(2)	Double height
CMP_BITMAP_QUADRUPLE(3)	Double size
Others	Printing size of image

[Return Values]

CMP_SUCCESS(0) : This value returns when a function succeeds.

CMP_FAIL(-1) : This value returns when a function fails.

[Example]

```
citizen_plugin.printImage( function(result) { }, function(err) { },
            imagePath, 1, 0 );
```

***You can also define the parameter values above.**

2.9. lineFeed

This function used for sending feeding command to printer.

```
citizen_plugin.lineFeed= function(successCallback, errorCallback,
            iLine);
```

[Parameter]

* iLine

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

2.10. printQRCode

This function used for supporting QRCode barcode printing.

citizen_plugin.printQRCode= function(successCallback, errorCallback,
data, dataSize, cellSize, iECL, alignment);

[Parameter]

* data
  - Barcode data to print.
* dataSize
  - Length of barcode data.
* cellSize
  - Module size. (1 – 20)
* iECL
  - Error Correction Level.

Variable	Description
CMP_QRCODE_EC_LEVEL_L(0)	Error correction Level L (7%)
CMP_QRCODE_EC_LEVEL_M(1)	Error correction Level M (15%)
CMP_QRCODE_EC_LEVEL_Q(2)	Error correction Level Q (25%)
CMP_QRCODE_EC_LEVEL_H(3)	Error correction Level H (30%)

* alignment
  - This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT(0)	Left alignment
CMP_ALIGNMENT_CENTER(1)	Center alignment
CMP_ALIGNMENT_RIGHT(2)	Right alignment

[Example]

citizen_plugin.printQRCode(function(result) { }, function(err) { },
data, 0, 5, 1, 1);

***You can also define the parameter values above.**

2.11. printPDF417

This function used for supporting PDF417 barcode printing.

citizen_plugin.printPDF417= function(successCallback, errorCallback,
pdfData, dataSize, numberOfColumn, cellWidth, alignment);

[Parameter]

* pdfData

- Barcode data to print.

* dataSize

- Length of Barcode data.

* numberOfColumn

- Number of Data Cordwords. (1 – 20)

* cellWidth

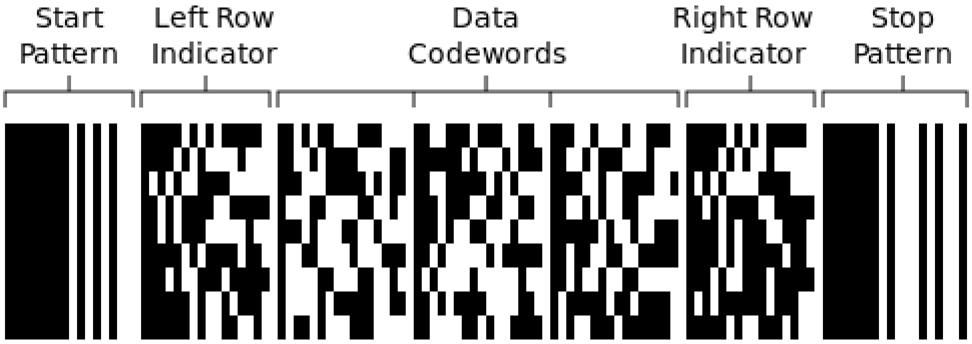
- Cell(Black or White bar) Width. (2 ~ 8)

* alignment

- This value is alignment. It sets barcode alignment.

Variable	Description
CMP_ALIGNMENT_LEFT(0)	Left alignment
CMP_ALIGNMENT_CENTER(1)	Center alignment
CMP_ALIGNMENT_RIGHT(2)	Right alignment

(Example) numberOfColumn = 3



numberOfColumn : number of Data Codewords

cellWidth : size of bar(black or White).

[Example]

```
citizen_plugin.printPDF417( function(result) { }, function(err) { },
            data, data_length, 10, 3, 1 );
```

***You can also define the parameter values above.**

2.12. setCharacterSet

This function is used to set the CharacterSet.

```
citizen_plugin.setCharacterSet= function(successCallback, errorCallback,
            iCharSet);
```

[Parameter]

* iCharSet

- Character set name(refer to ‘Charset for java.pdf’).

Codepage	Value
Codepage 437	437
Korean.	949
Japanese.	932
Chinese BIG5	950
Chinese GB2312	936
Codepage 850	850
Codepage 860	860
Codepage 863	863
Codepage 865	865
Codepage 866	866
Codepage 852	852
Codepage 858	858
Codepage 737	737
Codepage 857	857
Codepage 864	864
Codepage 1250	1250
Codepage 1251	1251
Codepage 1252	1252
Codepage 1253	1253
Codepage 1254	1254
Codepage 1256	1256
Codepage 1257	1257
Codepage 1258	1258
Codepage iso-8859-1	88591
Codepage iso-8859-2	88592
Codepage iso-8859-7	88597
Codepage iso-8859-9	88599

2.13. swipeMSR - [only Windows or Android]

This function used to read MSR track data.

```
citizen_plugin.swipeMSR= function(successCallback, errorCallback,
            selTrack);
```

[Parameter]

* selTrack

  - Select track for reading.

* SelectTrack

SelectTrack	MSR Track
0x31[0011 0001)	Track 1
0x32[0011 0010)	Track 2
0x33[0011 0011)	Track 1,2
0x34[0011 0100)	Track 3
0x36[0011 0110)	Track 2,3

2.14. cancelMSR - [only Windows or Android]

This function used to changing MSR Ready status to normal status

```
citizen_plugin.cancelMSR= function(successCallback, errorCallback);
```

[Parameter]

* none

2.15. searchBT - [only Android]

This function used to search a Bluetooth printers

```
citizen_plugin.searchBT= function(successCallback, errorCallback);
```

[Parameter]

* none

2.16. getCountBT - [only Android]

This function used to get the number of searched printers.

```
citizen_plugin.getCountBT= function(successCallback, errorCallback);
```

[Parameter]

* none

2.17. getListBTAddress - [only Android]

This function used to get the lists of searched printers.

****searchBT() must be called first before using this function**.

```
citizen_plugin.getListBTAddress= function(successCallback, errorCallback);
```

[Parameter]

* none

[Example]

```
citizen_plugin.printPDF417(
            function(result) {
                        var bluetooth= JSON.stringify(result);
                        var bluetoothInfo= JSON.parse(bluetooth);

//                      Use getCountBT () to retrieve the number of Bluetooth printers detected
                        for(var i=0; i<count; i++) {
//                                  Use bluetoothInfo[i] to get the Bluetooth address.
                        }
             },
            function(err) { } );
```

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.