

# CITIZEN

モバイルプリンター

iOSモジュールプログラムマニュアル

**CPCL**

Ver.1.08

**シチズン・システムズ株式会社**

## 目次

1. はじめに
2. メソッド

バージョン	対象ライブラリ	説明
1.01	1.40	公開
1.02	1.42	SetEncodingの説明に日本語印刷例を追加
1.03	1.42	Statusの戻り値を修正、inverseLineメソッドの説明を修正
1.04	1.46	Bluetoothインターフェースを追加
1.05	1.49	Socket connection タイムアウトを変更(3秒) 送信データの先頭に2バイトのダミーデータを送ることでパワーセーブモードからの復帰時にデータ抜け等が起こらないようにした。 printCheckとなっていた関数名をprinterCheckに修正し、status関数のBluetoothに関する説明に補足をした。
1.06	1.60	Amd64 機能を追加 ライブラリバージョン 1.60 から各メソッドの戻り値が Int から long に変更されました。 以前のバージョンのライブラリからアップデートされる場合、Xcode が Warning メッセージを出力します。アプリケーション側のソースコードの修正をお願いします。 プリンターをエラー状態で Wi-Fi 接続を試みると、待ち状態に陥る現象を改善 openPortを2回実行後closePortする接続しなくなる不具合を修正
1.07	1.61	resetCPCLBarCodeメソッドを追加。可視コードを無効にできるようにした。 1.60にて行ったIntをlongにした変更が本マニュアルに反映していなかった点を修正
1.08	1.70b	SDKのバージョンがアップデートしたのみ

## 1. はじめに

この「iOSモジュールプログラムマニュアル」では、iOSモバイルアプリケーションの開発に必要なライブラリファイルおよびヘッダーファイルから得られるメソッドについて説明します。

説明はラベル紙をベースに行われております。

ブラックマーク紙を使う場合は、ラベルの長さ(高さ)＝ブラックマーク間の距離などと読み替えてください。

プリンターの仕様につき不明の点は、CMP-30の技術マニュアル、CPCLのコマンドマニュアルを参考にして頂きますよう、お願いいたします。

## 2. メソッド

CPCLPrinterクラスで定義されています。

### 2.1 openPort

プリンターを接続します。

- (long) openPort:(NSString\*)portName withPortParam:(int) port

[パラメーター]

- portName: IPアドレスまたは”bluetooth”を指定します。

- port: ポート番号です(デフォルト9100)。

[戻り値]

0	接続が確立されました。
-1 , -2	接続に失敗しました。
-3	無効なデバイスです。

### 2.2 closePort

プリンターの接続を切断します。

- (long) closePort

[戻り値]

0	成功しました。
-1	失敗しました。

## 2.3. setForm

この関数は、用紙フォームを定義するために使用されます。

- (long) setForm:(int) horizonOffset withResX:(int) resolX withResY:(int) resolY  
withLabelHeight:(int) labelHeight withQuantity:(int) quantity

[パラメーター]

\* horizonOffset

- nullで終了する文字列を持つUnicode。ラベル全体の水平方向のオフセットを係数として受け取ります。これは、左側の印刷の絶対的な場所の値として考えることができます。

\* resolX

- 水平方向の解像度を係数として受け取ります。

\* resolY

- 垂直方向の解像度を受け取ります。

\* labelHeight

- 印刷用ラベルの高さの値を係数として受け取ります。

\* quantity

- 印刷するラベルの数量を係数として受け取ります[最大数 =< 1024]。

[戻り値]

0                    成功しました。

-1                   失敗しました。

注：必ず、

**setForm**

**setMedia**

**printForm**

の組み合わせで使用すること。印字内容は**setMedia**と**printForm**の間に記述する

## 2.4. printForm

この関数は、印刷後、用紙をフォームの上部に設定するために使用されます。

- (long) printForm

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.5. printerCheck

この関数は、プリンターのステータスチェック用に使用されます。

ステータス値の取得のためにはこの機能の呼出し後、status()関数を使用すること。

Bluetooth通信の場合、通常の方法では戻り値が取得できません。iOSモジュールプログラムマニュアル (Bluetoothに関する補足)を参照してください。

- (long) printerCheck

[戻り値]

0：関数が適切に処理されると、この値が戻されます。

-1：関数が適切に処理されなかった場合は、この値が戻されます。

## 2.6. status

この関数は、プリンターのステータスを取得するために使用されます。

Bluetooth通信の場合、通常の方法では戻り値が取得できません。戻り値の取得のためには、iOSモジュールプログラムマニュアル (Bluetoothに関する補足)を参照してください。

- (long) status

[戻り値]

STS\_CPCL\_NORMAL: プリンターステータスは「エラーなし」で、MSRは「準備ができていません」。

STS\_CPCL\_BUSY: プリンターのステータスはビジーステータスです。

STS\_CPCL\_PAPER\_EMPTY: プリンターステータスは「用紙がありません」です。

STS\_CPCL\_COVER\_OPEN: プリンターカバーが開いています。

STS\_CPCL\_BATTERY\_LOW: プリンターのバッテリー容量が少なくなっています。

注)それぞれのステータスの戻り値は、定義ファイル上

```
#define STS_CPCL_BATTERY_LOW 8
```

```
#define STS_CPCL_COVER_OPEN 4
```

```
#define STS_CPCL_PAPER_EMPTY 2
```

```
#define STS_CPCL_BUSY 1
```

```
#define STS_CPCL_NORMAL 0
```

となっていて、下位4ビットとそれぞれ対応していますが、上位4ビットが0001の値で固定になっているため、実際の戻り値は、16(10進)を足したものとなります。

## 2.7. setMeasure

この関数は、コマンドの単位を設定するために使用されます。

- (long) setMeasure:(int) measure

[パラメーター]

\* measure

- 単位セットコマンドを係数として受け取ります。

変数	説明
CPCL_INCH	単位をインチとして設定します。
CPCL_CENTI	単位をセンチメートルとして設定します。
CPCL_MILLI	単位をミリメートルとして設定します。
CPCL_DOTS	単位をドットとして設定します(デフォルト)。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.8. setJustification

この関数は、フィールドの位置揃えを設定するために使用されます。

- (long) setJustification:(int) justify

[パラメーター]

\* justify

- フィールドの位置揃えを設定するための係数を受け取ります。

変数	説明
CPCL_LEFT	左揃え
CPCL_CENTER	中央揃え
CPCL_RIGHT	右揃え

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.9. printCPCLText

この関数は、フォーム上の指定の場所にテキストを印刷するために使用されます。

- (long) printCPCLText:(int) rotation withFontType:(int) fontType withFontSize:(int) fontSize  
withPrintX:(int) printX withPrintY:(int) printY withData:(NSString \*) data withCount:(int) count

[パラメーター]

\* rotation

- 印刷するテキストの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	テキストを回転せずに印刷します。
CPCL_90_ROTATION	テキストを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	テキストを180度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	テキストを270度回転(反時計方向)して印刷します。

\* fontType

- 印刷するテキストのフォントの種類を係数として受け取ります。

\* fontSize

- 印刷するテキストのフォントのサイズの値を係数として受け取ります。

\* printX

- 印刷するテキストのx座標の値を係数として受け取ります。

\* printY

- 印刷するテキストのy座標の値を係数として受け取ります。

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。

\* count

- 印刷するテキストのカウント関数を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。



## 2.10. setConcat

この関数は、テキスト連結の開始と印刷位置を定義するために使用されます。

- (long) setConcat:(int) concatMode withPrintX:(int) printX withPrintY:(int) printY

[パラメーター]

\* concatMode

- 連結するテキストの連結方法を係数として受け取ります。

変数	説明
CPCL_CONCAT	テキストを横に連結します。
CPCL_VCONCAT	テキストを縦に連結します。

\* printX

- 印刷するテキストのx座標の値を係数として受け取ります。

\* printY

- 印刷するテキストのy座標の値を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.11. concatText

この関数は、連結するフォントを定義するために使用されます。

- (long) concatText:(int) fontType withFontSize:(int) fontSize withOffset:(int) offset  
withData:(NSString \*) data

[パラメーター]

\* fontType

- 印刷するテキストのフォントの種類を係数として受け取ります。

\* fontSize

- 印刷するテキストのフォントのサイズの値を係数として受け取ります。

\* offset

- 開始点の単位値を受け取ります。

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.12. resetConcat

この関数は、テキスト連結を取り消すために使用されます。

- (long) resetConcat

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.13. setMultiLine

この関数は、同じ長さと同じフォントで複数行の印刷を定義するために使用されます。

- (long) setMultiLine:(int) lineHeight

[パラメーター]

\* lineHeight

- 印刷するテキストの行の高さの値を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.14. multiLineText

この関数は、SetMultiLine()として定義されるテキストを印刷するために使用されます。印刷するテキストのフォント、サイズ、および印刷位置を係数として受け取ります。

- (long) multiLineText:(int) rotation withFontType:(int) fontType withFontSize:(int) fontSize  
withPrintX:(int) printX withPrintY:(int) printY

[パラメーター]

\* rotation

- 印刷するテキストの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	テキストを回転せずに印刷します。
CPCL_90_ROTATION	テキストを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	テキストを180度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	テキストを270度回転(反時計方向)して印刷します。

\* fontType

- 印刷するテキストのフォントの種類を係数として受け取ります。

\* fontSize

- 印刷するテキストのフォントのサイズの値を係数として受け取ります。

\* printX

- 印刷するテキストのx座標の値を係数として受け取ります。

\* printY

- 印刷するテキストのy座標の値を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.15. multiLineData

この関数は、MultiLineText()で定義される値のテキストを印刷するために使用されます。

- (long) multiLineData:(NSString \*) data

[パラメーター]

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.16. resetMultiLine

この関数は、SetMultiLine()で定義された関数を取り消すために使用されます。

- (long) resetMultiLine

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.17. setMagnify

この関数は、テキストの表示倍率を設定するために使用されます。

- (long) setMagnify:(int) width withHeight:(int) height

[パラメーター]

\* width

- テキストの幅の比率を係数として受け取ります。

変数 (幅の比率を設定)	説明
CPCL_TXT_1WIDTH	幅の比率を1倍として設定
CPCL_TXT_2WIDTH	幅の比率を2倍として設定
CPCL_TXT_3WIDTH	幅の比率を3倍として設定
CPCL_TXT_4WIDTH	幅の比率を4倍として設定
CPCL_TXT_5WIDTH	幅の比率を5倍として設定
CPCL_TXT_6WIDTH	幅の比率を6倍として設定
CPCL_TXT_7WIDTH	幅の比率を7倍として設定
CPCL_TXT_8WIDTH	幅の比率を8倍として設定
CPCL_TXT_9WIDTH	幅の比率を9倍として設定
CPCL_TXT_10WIDTH	幅の比率を10倍として設定
CPCL_TXT_11WIDTH	幅の比率を11倍として設定
CPCL_TXT_12WIDTH	幅の比率を12倍として設定
CPCL_TXT_13WIDTH	幅の比率を13倍として設定

CPCL_TXT_14WIDTH	幅の比率を14倍として設定
CPCL_TXT_15WIDTH	幅の比率を15倍として設定
CPCL_TXT_16WIDTH	幅の比率を16倍として設定

\* height

- テキストの高さの比率を係数として受け取ります。

変数(高さの比率を設定)	説明
CPCL_TXT_1HEIGHT	高さの比率を1倍として設定
CPCL_TXT_2HEIGHT	高さの比率を2倍として設定
CPCL_TXT_3HEIGHT	高さの比率を3倍として設定
CPCL_TXT_4HEIGHT	高さの比率を4倍として設定
CPCL_TXT_5HEIGHT	高さの比率を5倍として設定
CPCL_TXT_6HEIGHT	高さの比率を6倍として設定
CPCL_TXT_7HEIGHT	高さの比率を7倍として設定
CPCL_TXT_8HEIGHT	高さの比率を8倍として設定
CPCL_TXT_9HEIGHT	高さの比率を9倍として設定
CPCL_TXT_10HEIGHT	高さの比率を10倍として設定
CPCL_TXT_11HEIGHT	高さの比率を11倍として設定
CPCL_TXT_12HEIGHT	高さの比率を12倍として設定
CPCL_TXT_13HEIGHT	高さの比率を13倍として設定
CPCL_TXT_14HEIGHT	高さの比率を14倍として設定
CPCL_TXT_15HEIGHT	高さの比率を15倍として設定
CPCL_TXT_16HEIGHT	高さの比率を16倍として設定

[戻り値]

0                    成功しました。  
-1                    失敗しました。

## 2.18. resetMagnify

この関数は、SetMagnify()で定義された関数を取り消すために使用されます。

- (long) resetMagnify

[戻り値]

0                    成功しました。

-1                    失敗しました。

## 2.19. printCPCLBarCode

この関数は、バーコードを印刷するために使用されます。

- (long) printCPCLBarcode:(int) rotation withBarcodeType:(NSString \*) barcodeType  
withNarrowBar:(int) NB withRatio:(int) ratio withBarHeight:(int) barHeight withPrintX:(int) printX  
withPrintY:(int) printY withData:(NSString \*) data withCount:(int) count

[パラメーター]

\* rotation

- 印刷するバーコードの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	バーコードを回転せずに印刷します。
CPCL_90_ROTATION	バーコードを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	バーコードを180度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	テキストを270度回転(反時計方向)して印刷します。

\* barCodeType

- 印刷するバーコードの種類を係数として受け取ります。

変数	説明
CPCL_BCS_39	バーコード3/9標準
CPCL_BCS_39C	バーコード3/9標準 チェックデジット付加
CPCL_BCS_39F	バーコード3/9標準 Full ASCII
CPCL_BCS_39FC	バーコード3/9 Full ASCII チェックデジット付加
CPCL_BCS_93	バーコード Code 93
CPCL_BCS_128	バーコード Code 128
CPCL_BCS_EAN128	バーコード UCC-128
CPCL_BCS_CODABAR	バーコード Codabar
CPCL_BCS_CODABARC	バーコード Codabar チェックデジット付加
CPCL_BCS_EAN8	バーコード EAN8
CPCL_BCS_EAN82	バーコード EAN8 2桁アドオン
CPCL_BCS_EAN85	バーコード EAN8 5桁アドオン
CPCL_BCS_EAN13	バーコード EAN13
CPCL_BCS_EAN132	バーコード EAN13 2桁アドオン
CPCL_BCS_EAN135	バーコード EAN13 5桁アドオン
CPCL_BCS_I2OF5	バーコード Interleaved 2 of 5

CPCL_BCS_POSTNET	バーコード PostNet
CPCL_BCS_UPCA	バーコード UPCA
CPCL_BCS_UPCA2	バーコード UPCA 2桁アドオン
CPCL_BCS_UPCA5	バーコード UPCA 5桁アドオン
CPCL_BCS_UPCE	バーコード UPCE
CPCL_BCS_UPCE2	バーコード UPCE 2桁アドオン
CPCL_BCS_UPCE5	バーコード UPCE 5桁アドオン
CPCL_BCS_MSI	バーコード Plessey(MSI-1)
CPCL_BCS_MSI1C	バーコード Plessey(MSI-1)チェックデジット付加
CPCL_BCS_MSI2C	バーコード Plessey(MSI-2)チェックデジット付加
CPCL_BCS_MSI11C	バーコード Plessey(MSI-11)チェックデジット付加
CPCL_BCS_PLUS2	Plus 2 Extension
CPCL_BCS_PLUS5	Plus 5 Extension

\* NB

- バーコードの細いバーの幅を係数として受け取ります。

\* ratio

- バーコードの比率の値を係数として受け取ります。

変数	説明
CPCL_BCS_0RATIO	バーコードの比率を 1.5 : 1 として設定します。
CPCL_BCS_1RATIO	バーコードの比率を2.0 : 1として設定します。
CPCL_BCS_2RATIO	バーコードの比率を2.5 : 1として設定します。
CPCL_BCS_3RATIO	バーコードの比率を3.0 : 1として設定します。
CPCL_BCS_4RATIO	バーコードの比率を3.5 : 1として設定します。
CPCL_BCS_20RATIO	バーコードの比率を2.0 : 1として設定します。
CPCL_BCS_21RATIO	バーコードの比率を2.1 : 1として設定します。
CPCL_BCS_22RATIO	バーコードの比率を2.2 : 1として設定します。
CPCL_BCS_23RATIO	バーコードの比率を2.3 : 1として設定します。
CPCL_BCS_24RATIO	バーコードの比率を2.4 : 1として設定します。
CPCL_BCS_25RATIO	バーコードの比率を2.5 : 1として設定します。
CPCL_BCS_26RATIO	バーコードの比率を2.6 : 1として設定します。
CPCL_BCS_27RATIO	バーコードの比率を2.7 : 1として設定します。
CPCL_BCS_28RATIO	バーコードの比率を2.8 : 1として設定します。
CPCL_BCS_29RATIO	バーコードの比率を2.9 : 1として設定します。
CPCL_BCS_30RATIO	バーコードの比率を 3.0 : 1 として設定します。

\* barHeight

- 印刷するバーコードの高さの値を係数として受け取ります。

\* printX

- 印刷するバーコードのx軸の開始点を係数として受け取ります。

\* printY

- 印刷するバーコードのy軸の開始点を係数として受け取ります。

\* data

- 印刷するバーコードデータを係数として受け取ります。

\* count

- 印刷するバーコードのカウント関数を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.20. printBox

この関数は、ボックス画像を印刷するために使用されます。

- (long) printBox:(int) xs withYs:(int) ys withXx:(int) xx withYX:(int) yx withThickness:(int) thickness

[パラメーター]

\* xs

- ボックス印刷のためのx軸の開始点を係数として受け取ります。

\* ys

- ボックス印刷のためのy軸の開始点を係数として受け取ります。

\* xx

- ボックス印刷のためのx軸の終了点を係数として受け取ります。

\* yx

- ボックス印刷のためのy軸の終了点を係数として受け取ります。

\* thickness

- 印刷するボックス線の太さを係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。



## 2.21. printLine

この関数は、線の画像を印刷するために使用されます。

- (long) printLine:(int) xs withYs:(int) ys withXx:(int) xx withYx:(int) yx withThickness:(int) thickness

[パラメーター]

\* xs

- 線の印刷のためのx軸の開始点を係数として受け取ります。

\* ys

- 線の印刷のためのy軸の開始点を係数として受け取ります。

\* xx

- 線の印刷のためのx軸の終了点を係数として受け取ります。

\* yx

- 線の印刷のためのy軸の終了点を係数として受け取ります。

\* thickness

- 印刷する線の太さを係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.22. inverseLine

この関数は、指定した領域内の画像を反転するために使用されます。PrintLine() メソッドと同じ構文です。

- (long) inverseLine:(int) xs withYs:(int) ys withXx:(int) xx withYx:(int) yx withThickness:(int) thickness

[パラメーター]

\* xs

- 原像の印刷のためのx軸の開始点を係数として受け取ります。

\* ys

- 原像の印刷のためのy軸の開始点を係数として受け取ります。

\* xx

- 原像の印刷のためのx軸の終了点を係数として受け取ります。

\* yx

- 原像の印刷のためのy軸の終了点を係数として受け取ります。

\* thickness

- 原像の印刷のための線の太さを係数として受け取ります。

## 2.23. setPattern

この関数は、パターン印刷のために使用されます。

- (long) setPattern:(int) patternNum

[パラメーター]

\* PatternNum

- 印刷するパターンの値を係数として受け取ります。

変数	説明
CPCL_DEFAULT_PATTERN	塗りつぶし(黒/デフォルト値)
CPCL_HORIZON_PATTERN	パターンを水平線として印刷します。
CPCL_VERTICAL_PATTERN	パターンを垂直線として印刷します。
CPCL_RDIAGON_PATTERN	右斜線パターンを印刷します。
CPCL_LDIAGON_PATTERN	左斜線パターンを印刷します。
CPCL_SQUARE_PATTERN	パターンを正方形として印刷します。
CPCL_CROSS_PATTERN	左斜線と右斜線の格子パターンを印刷します。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.24. printBitmap

この関数は、ビットマップ画像を印刷するために使用されます。

- (long) printBitmap:(NSString \*) filePath withPrintX:(int) printX withPrintY:(int) printY  
withBrightness:(int) bright

[パラメーター]

\* filePath

- 印刷する画像のファイルパスを係数として受け取ります。

\* printX

- 印刷する画像のx軸の開始点を係数として受け取ります。

\* printY

- 印刷する画像のy軸の開始点を係数として受け取ります。

\* bright

- 画像の輝度を設定します。(1～10)

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.25. setContrast

この関数は、印刷するラベル全体のコントラストを調整するために使用されます。

- (long) setContrast:(int) darkness

[パラメーター]

\* darkness

- 印刷するラベル全体のコントラストの値を係数として受け取ります。[0 <= 暗さ <=3]

変数	説明
CPCL_CONT_DEFAULT	デフォルトとして設定
CPCL_CONT_MEDIUM	中間の輝度で印刷
CPCL_CONT_DARK	暗く印刷
CPCL_CONT_VERY_DARK	非常に暗く印刷

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.26. setPageWidth

この関数は、用紙フォームの幅を定義するために使用されます。

- (long) setPageWidth:(int) pageWidth

[パラメーター]

\* pageWidth

- 印刷するラベルのページ幅の値を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.27. printCPCLImage

この関数は、プリンターのフラッシュファイルシステムに保存されたpcx画像を印刷するために使用されます。  
『Desktop Application Manual』（デスクトップアプリケーションマニュアル）を参照してください。

- (long) printCPCLImage:(NSString \*) imageName withPrintX:(int) printX withPrintY:(int) printY

[パラメーター]

\* imageName

- nullで終了する文字列を持つUnicode。

フラッシュファイルシステムに保存されたpcxファイル名を係数として受け取ります。

\* printX

- 印刷するx軸の開始点を係数として受け取ります。

\* printY

- 印刷するy軸の開始点を係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.28. setSpeed

この関数は、印刷速度を定義するために使用されます。

- (long) setSpeed:(int) speed

[パラメーター]

\* speed

- 印刷速度を係数として受け取ります。[0 =< 速度 =<5]

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.29. setTone

この関数は、SetContrast()の代わりに使用されるメソッドであり、印刷するラベル全体の暗さを調整するために使用されます。

- (long) setTone:(int) tone

[パラメーター]

\* tone

- ラベル全体の暗さの色調を係数として受け取ります。[-99 =< 色調=< 200]

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.30. setCPCLBarcode

この関数は、HRI文字情報を設定するために使用されます。

- (long) setCPCLBarcode:(int) fontNum withFontSize:(int) fontSize withOffset:(int) offset

[パラメーター]

\* fontNum

- 印刷するテキストのフォントの種類を係数として受け取ります。

\* fontSize

- 印刷するテキストのフォントのサイズの値を係数として受け取ります。

\* offset

- 印刷するバーコードとテキストのフォントの間のオフセットを係数として受け取ります。

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.31. setMedia

この関数は、印刷するラベルの種類を設定するために使用されます。

- (long) setMedia:(int) mode

[パラメーター]

\* mode

- ラベルの種類を設定します。

変数	説明
CPCL_LABEL	間隔のあるラベル
CPCL_BLACKMARK	黒いマークのあるラベル
CPCL_CONTINUOUS	連続的なラベル

[戻り値]

0                    成功しました。

-1                   失敗しました。

## 2.32. setEncoding

このメソッドは、文字セットを設定するために使用されます。

詳細については、以下のURLを参照してください(StringEncodingセクション)。

[https://developer.apple.com/library/mac/documentation/Cocoa/Reference/Foundation/Classes/NSString\\_Class/Reference/NSString.html](https://developer.apple.com/library/mac/documentation/Cocoa/Reference/Foundation/Classes/NSString_Class/Reference/NSString.html)

- (void) setEncoding:(NSStringEncoding)

[パラメーター]

- (NSStringEncoding) :変換する文字セットです。

### 日本語印刷例

```
cpcl = [[CPCLPrinter alloc] init];
```

```
[cpcl openPort:<IPアドレス> withPortParam:<ポート番号>];
```

```
[cpcl setForm:0 withResX:200 withResY:200 withLabelHeight:203 withQuantity:1]
```

```
[cpcl setMedia:CPCL_CONTINUOUS];
```

```
[cpcl setEncoding:NSShiftJISStringEncoding];
```

```
[cpcl printCPCLText:0 withFontType:0 withFontSize:0 withPrintX:0
```

```
withPrintY:0 withData:@"漢字" withCount:0]
```

```
[cpcl printForm]
```

```
[cpcl closePort];
```

```
[cpcl release];
```

## 2.33. printPDF417

この関数は、印刷するPDF417バーコードを定義するために使用されます。

- (void) printPDF417:(int) rotation withPrintX:(int) printX withPrintY:(int) printY withUnitWidth:(int) unitWidth withUnitHeight:(int) unitHeight withNumOfColumns:(int) column withSecurityLevel:(int) securityLevel withData:(NSString \*) data

[パラメーター]

\* rotation

- 印刷するバーコードの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	バーコードを回転せずに印刷します。
CPCL_90_ROTATION	バーコードを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	バーコードを 180 度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	テキストを270度回転(反時計方向)して印刷します。

\* printX

- 印刷するバーコードのx軸の開始点を係数として受け取ります。

\* printY

- 印刷するバーコードのy軸の開始点を係数として受け取ります。

\* unitWidth

- 最も細いエレメント(ナローバー)の単位幅。(1~32、デフォルト2)

\* unitHeight

- 最も細いエレメント(ナローバー)の単位高さ。(1~32、デフォルト6)

\* column

- 使用する列数。(1~30、デフォルト3)

\* securityLevel

- セキュリティレベルは、検出および/または修正されるエラーの最大数を示します。(0~8、デフォルト1)

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。

## 2.34. printDATAMATRIX

この関数は、印刷するDATAMATRIXバーコードを定義するために使用されます。

- (void) printDATAMATRIX:(int) rotation withPrintX:(int) printX withPrintY:(int) printY  
withECCLevel:(int) eccLevel withCellSize:(int) cellSize withData:(NSString \*) data

[パラメーター]

\* rotation

- 印刷するバーコードの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	バーコードを回転せずに印刷します。
CPCL_90_ROTATION	バーコードを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	バーコードを 180 度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	テキストを270度回転(反時計方向)して印刷します。

\* printX

- 印刷するバーコードのx軸の開始点を係数として受け取ります。

\* printY

- 印刷するバーコードのy軸の開始点を係数として受け取ります。

\* eccLevel

- DATAMATRIXのECCレベルです。(0、50、80、100、140、200、デフォルト0)

\* cellSize

- セルのサイズです。(1～24、デフォルト4)

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。



## 2.35. printQRCODE

この関数は、印刷するQRCODEバーコードを定義するために使用されます。

- (void) printQRCODE:(int) rotation withPrintX:(int) printX withPrintY:(int) printY withVersion:(int) version withECLevel:(int) ecLevel withModuleSize:(int) moduleSize withMaskNo:(int) maskNo withData:(NSString \*) data

[パラメーター]

\* rotation

- 印刷するバーコードの印刷方向の値を係数として受け取ります。

変数	説明
CPCL_0_ROTATION	バーコードを回転せずに印刷します。
CPCL_90_ROTATION	バーコードを90度回転(反時計方向)して印刷します。
CPCL_180_ROTATION	バーコードを180度回転(反時計方向)して印刷します。
CPCL_270_ROTATION	バーコードを270度回転(反時計方向)して印刷します。

\* printX

- 印刷するバーコードのx軸の開始点を係数として受け取ります。

\* printY

- 印刷するバーコードのy軸の開始点を係数として受け取ります。

\* version

- QRCODEのバージョンです。(1 ~ 40)

\* ecLevel

- エラー修正レベルです。(0~3、デフォルト1)

\* moduleSize

- モジュールのサイズです。(1~24、デフォルト4)

- マスク番号です。(0~7。値が8を超える場合は自動選択されます)

\* data

- nullで終了する文字列を持つUnicode。印刷するテキストを係数として受け取ります。

## 2.36. resetCPCLBarCode

この機能は、可視コードをキャンセルするために使用されます。

- (long) resetCPCLBarcode

[戻り値]

0 成功しました。

-1 失敗しました。