

Class

XCPCLCommand

```
@interface XCPCLCommand : NSObject
```

getCommand

Get the Printing Command

```
- (NSData*)getCommand;
```

Description

- This method retrieves the current CPCL object's printing data.

setCharEncoding:

Set Character Encoding

```
- (XCPCLCommand *)setCharEncoding:  
(NSStringEncoding)encoding;
```

Parameters

- `encoding`
Encoding format, default is GB_18030_2000.

Return Value

- XCPCLCommand object.
-

addData:

Add Custom Data

```
- (XCPCLCommand *)addData:(NSData *)customData;
```

Parameters

- `customData`
Custom data content.

Return Value

- XCPCLCommand object.
-

initializeLabelWithHeight:

Label Initialization

```
- (XCPCLCommand *)initializeLabelWithHeight:(int)height;
```

Parameters

- `height`
Maximum height of the label.

Return Value

- XCPCLCommand object.
-

initializeLabelWithHeight:count:

Label Initialization

```
- (XCPCLCommand *)initializeLabelWithHeight:(int)height  
count:(int)count;
```

Parameters

- `height`
Maximum height of the label.
- `count`
Number of labels to print, default is 1.

Return Value

- XCPCLCommand object.
-

initializeLabelWithHeight:count:offset x:

Label Initialization

```
- (XCPCLCommand *)initializeLabelWithHeight:(int)height  
count:(int)count offsetx:(int)offsetx;
```

Parameters

- `height`
Maximum height of the label.
- `count`
Number of labels to print, default is 1.
- `offsetx`
Horizontal offset of the label, default is 0.

Return Value

- XCPCLCommand object.

addTextAtX:y:content:

Text Printing

```
- (XCPCLCommand *)addTextAtX:(int)x  
                        y:(int)y  
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the text.
- `y`
Starting Y coordinate of the text.
- `content`
Text content.

Return Value

- XCPCLCommand object.

addTextAtX:y:font:content:

Text Printing

```
- (XCPCLCommand *)addTextAtX:(int)x
                        y:(int)y
                        font:(FontCPCL) font
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the text.
- `y`
Starting Y coordinate of the text.
- `font`
Text font type, default is CPCLFont0.

Variable	Description
CPCLFont0	Font 0
CPCLFont1	Font 1
CPCLFont2	Font 2
CPCLFont3	Font 3
CPCLFont4	Font 4
CPCLFont5	Font 5
CPCLFont6	Font 6
CPCLFont7	Font 7
CPCLFont24	Font 24
CPCLFont55	Font 55

- `content`
Text content.

Return Value

- XCPCLCommand object.

addTextAtX:y:rotation:font:content:

Text Printing

```
- (XCPCLCommand *)addTextAtX:(int)x
    y:(int)y
    rotation:(RotationCPCL)rotation
    font:(FontCPCL)font
    content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the text.
- `y`
Starting Y coordinate of the text.
- `rotation`
Clockwise rotation angle, default is CPCLRotation0.

Variable	Description
CPCLRotation0	No rotation
CPCLRotation90	Rotate 90 degrees
CPCLRotation180	Rotate 180 degrees
CPCLRotation270	Rotate 270 degrees

- `font`
Text font type, default is CPCLFont0.
- `content`
Text content.

Return Value

- XCPCLCommand object.

setmagWithw:h:

Enlarge the Resident Font

```
- (XCPCLCommand *)setmagWithw:(int)w h:(int)h;
```

Parameters

- `w`
Width magnification factor (1~16).
- `h`
Height magnification factor (1~16).

Return Value

- XCPCLCommand object.

addPrint

End Command for Printing

```
- (XCPCLCommand *)addPrint;
```

Description

- This is the end command for the entire command set, which starts file printing.

Return Value

- XCPCLCommand object.

addBarcodeAtX:y:type:height:content:

Horizontal 1D Barcode

```
- (XCPCLCommand *)addBarcodeAtX:(int)x
                        y:(int)y
                        type:(CPCLBarCodeType)type
                        height:(int)height
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the barcode, in dots.
- `y`
Starting Y coordinate of the barcode, in dots.
- `type`
Barcode type, refer to `CPCLBarCodeType` for details.

Variable	Description
CPCLBarCode128	Code 128
CPCLBarCodeUPCA	UPC-A
CPCLBarCodeUPCE	UPC-E
CPCLBarCodeEAN13	EAN-13
CPCLBarCodeEAN8	EAN-8
CPCLBarCode39	Code 39
CPCLBarCode93	Code 93
CPCLBarCodeCODABAR	CODABAR

- `height`
Barcode unit height.
- `content`
Barcode data content.

Return Value

- `XCPCCommand` object.

addBarcodeAtX:y:type:width:ratio:height:content:

Horizontal 1D Barcode with Width and Ratio

```
- (XCPCLCommand *)addBarcodeAtX:(int)x
                        y:(int)y
                        type:(CPCLBarCodeType)type
                        width:(int)width
                        ratio:(BarCodeRatioCPCL)ratio
                        height:(int)height
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the barcode, in dots.
- `y`
Starting Y coordinate of the barcode, in dots.
- `type`
Barcode type, refer to `CPCLBarCodeType` for details.
- `width`
Narrow bar unit width, default is `1`.
- `ratio`
Ratio of wide bar to narrow bar. Refer to the following for available options:

Variable	Description
CPCLBarCodeRatio0	Wide:Narrow = 1.5:1
CPCLBarCodeRatio1	Wide:Narrow = 2.0:1
CPCLBarCodeRatio2	Wide:Narrow = 2.5:1
CPCLBarCodeRatio3	Wide:Narrow = 3.0:1
CPCLBarCodeRatio4	Wide:Narrow = 3.5:1
CPCLBarCodeRatio20	Wide:Narrow = 2.0:1
CPCLBarCodeRatio21	Wide:Narrow = 2.1:1
CPCLBarCodeRatio22	Wide:Narrow = 2.2:1
CPCLBarCodeRatio23	Wide:Narrow = 2.3:1
CPCLBarCodeRatio24	Wide:Narrow = 2.4:1
CPCLBarCodeRatio25	Wide:Narrow = 2.5:1
CPCLBarCodeRatio26	Wide:Narrow = 2.6:1
CPCLBarCodeRatio27	Wide:Narrow = 2.7:1
CPCLBarCodeRatio28	Wide:Narrow = 2.8:1
CPCLBarCodeRatio29	Wide:Narrow = 2.9:1
CPCLBarCodeRatio30	Wide:Narrow = 3.0:1

- height
Barcode unit height.

- `content`

Barcode data content.

Return Value

- `XCPCLError` object.

addBarcodeVAtX:y:type:height:content:

Vertical 1D Barcode

```
- (XCPCLError *)addBarcodeVAtX:(int)x
                        y:(int)y
                        type:(CPCLErrorType)type
                        height:(int)height
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the barcode, in dots.
- `y`
Starting Y coordinate of the barcode, in dots.
- `type`
Barcode type, refer to `CPCLErrorType` for details.
- `height`
Barcode unit height.

- `content`

Barcode data content.

Return Value

- `XCPCLError` object.

addBarcodeVAtX:y:type:width:ratio:height:content:

Vertical 1D Barcode with Width and Ratio

```
- (XCPCLError *)addBarcodeVAtX:(int)x
                        y:(int)y
                        type:(CPCLErrorType)type
                        width:(int)width
                        ratio:(BarcodeRatioCPCLError)ratio
                        height:(int)height
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the barcode, in dots.
- `y`
Starting Y coordinate of the barcode, in dots.
- `type`
Barcode type, refer to `CPCLErrorType` for details.
- `width`
Narrow bar unit width, default is `1`.

- `ratio`

Ratio of wide bar to narrow bar, default is `CPCLBarCodeRatio1 (2.0:1)`.

- `height`

Barcode unit height.

- `content`

Barcode data content.

Return Value

- `XCPCLError` object.

barcodeText:

Add Barcode Text

```
- (XCPCLError *)barcodeText:(int)offsetx;
```

Parameters

- `offsetx`

Text offset from the barcode, in units.

Return Value

- `XCPCLError` object.

barcodeTextOff

Cancel Barcode Text

```
- (XCPCLCommand *)barcodeTextOff;
```

Return Value

- `XCPCLCommand` object.

addQRCodeAtX:y:content:

Draw QR Code (Simple Version)

```
- (XCPCLCommand *)addQRCodeAtX:(int)x  
                             y:(int)y  
                             content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the QR code.
- `y`
Starting Y coordinate of the QR code.
- `content`
QR code data content.

Return Value

- `XCPCLCommand` object.

addQRCodeAtX:y:codeModel:cellWidth:h:content:

Draw QR Code (Advanced Version)

```
- (XCPCLError *)addQRCodeAtX:(int)x
                        y:(int)y
                        codeModel:(QRCodeModesCPCL)codeModel
                        cellWidth:(int)cellWidth
                        content:(NSString *)content;
```

Parameters

- `x`
Starting X coordinate of the QR code.
- `y`
Starting Y coordinate of the QR code.
- `codeModel`
QR Code specification number. Possible values:
 - `CPCLQRCodeModeORG`: Original specification.
 - `CPCLQRCodeModeEnhance`: Enhanced specification.
- `cellWidth`
Cell size, range [1, 32]. Default is 6.
- `content`
QR code data content.

Return Value

- `XCPCLError` object.

addBoxAtX:y:width:height:thickness:

Draw Rectangle

```
- (XCPCLCommand *)addBoxAtX:(int)x
    y:(int)y
    width:(int)width
    height:(int)height
    thickness:(int)thickness;
```

Parameters

- `x`
Starting X coordinate of the rectangle, in dots.
- `y`
Starting Y coordinate of the rectangle, in dots.
- `width`
Rectangle width, in dots.
- `height`
Rectangle height, in dots.
- `thickness`
Line width.

Return Value

- `XCPCLCommand` object.

addLineAtX:y:xend:yend:width:

Draw Line

```
- (XCPCLCommand *)addLineAtX:(int)x
                        y:(int)y
                        xend:(int)xend
                        yend:(int)yend
                        width:(int)width;
```

Parameters

- `x`
Starting X coordinate of the line, in dots.
- `y`
Starting Y coordinate of the line, in dots.
- `xend`
Ending X coordinate of the line, in dots.
- `yend`
Ending Y coordinate of the line, in dots.
- `width`
Line width.

Return Value

- `XCPCLCommand` object.

addInverseLineAtX:y:xend:yend:width:

Reverse the Data in the Specified Area

```
- (XCPCLCommand *)addInverseLineAtX:(int)x
                        y:(int)y
                        xend:(int)xend
                        yend:(int)yend
                        width:(int)width;
```

Parameters

- `x`
Starting X coordinate of the inverse area, in dots.
- `y`
Starting Y coordinate of the inverse area, in dots.
- `xend`
Ending X coordinate of the inverse area, in dots.
- `yend`
Ending Y coordinate of the inverse area, in dots.
- `width`
Inverse area width.

Return Value

- `XCPCLCommand` object.

addGraphicsAtX:y:image

Draw Image

```
- (XCPCLCommand *)addGraphicsAtX:(int)x  
                        y:(int)y  
                        image:(UIImage *)image;
```

Parameters

- `x`
Starting X coordinate of the image, in dots.
- `y`
Starting Y coordinate of the image, in dots.
- `image`
Image object.

Return Value

- `XCPCLCommand` object.
-

addAlign:

Set Field Alignment (Simple Version)

```
- (XCPCLCommand *)addAlign:(AlignmentsCPCL)align;
```

Parameters

- `align`
Alignment mode. Possible values:
 - `CPCLAlignmentLeft`: Left-align all subsequent fields.
 - `CPCLAlignmentCenter`: Center-align all subsequent fields.

- `CPCLAlignmentRight`: Right-align all subsequent fields.

Return Value

- `XCPCCommand` object.

addAlign:end

Set Field Alignment (With Endpoint)

```
- (XCPCCommand *)addAlign:(AlignmentsCPCL)align  
    end:(int)end;
```

Parameters

- `align`
Alignment mode. Possible values:
 - `CPCLAlignmentLeft`: Left-align all subsequent fields.
 - `CPCLAlignmentCenter`: Center-align all subsequent fields.
 - `CPCLAlignmentRight`: Right-align all subsequent fields.
- `end`
Alignment end point.

Return Value

- `XCPCCommand` object.

addSpeed:

Set the Print Speed

```
- (XCPCLCommand *)addSpeed:(int)level;
```

Parameters

- `level`
Speed level, range [0, 5].

Return Value

- `XCPCLCommand` object.
-

addPageWidth:

Set the Page Width

```
- (XCPCLCommand *)addPageWidth:(int)width;
```

Parameters

- `width`
Page unit width.

Return Value

- `XCPCLCommand` object.
-

addBeep:

Set the Beep Duration

```
- (XCPCLCommand *)addBeep:(int)length;
```

Parameters

- `length`
Beep duration, in 1/8 second units (e.g., 16 means 2 seconds).

Return Value

- `XCPCLCommand` object.
-

form

Form Feed

```
- (XCPCLCommand *)form;
```

Return Value

- `XCPCLCommand` object.
-