[CPCL WindowsCE SDK]

[**Printer CPCL Command Development Manual** v1.0]

1. Information of the Manual

This SDK manual provides the dll file information for WinCE application development.

We continuously promote and update the function and quality of all our products. Any change to the product specification and the manual will be without any further notice.

2. Operation System

- WinCE 4.0/5.0/6.0
- Windows Mobile

3. Remark

- When error code Return Value is greater than 0, it is the internal error of Windows system, please refer to related help file.

4. Method

4.1 PrinterCreator

Set up the target printer of specified model (should create target printer before using any function).

int PrinterCreator(

    void* *handle*,

    const char* *model*

);

Parameter：

void handle*
    [in,out] The created target printer object.
const char model*
    [in] Specify the model of target printer.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_NO_ENOUGMEMORY	-2	No enough memory
E_INVALID_MODEL	-8	Invalid model name

4.2 PrinterCreatorS

Set up the target printer of specified model, the function is same to PrinterCreator (should create target printer before using any function).

void* PrinterCreatorS(

const char* *model*

);

Parameter：

const char* model
    [in] Specify the model of target printer

Return：

Success : return the handle of printer object.

Fail: return NULL, invalid handle.

4.3 PrinterDestroy

Release the resource of specified model printer that has set up (after operation completed and no more operation for printer, it should release the printer that has set up).

int PrinterDestroy(

    void* *handle*

);

Parameter：

void* handle
    [in] The handle of target printer object which needs to release.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle

4.4 PortOpen

Open the communication port and connect with the printer. After successfully connected, other functions can be used. If failed connecting, please check the error information.

int PortOpen(

    void* *handle,*

    const TCHAR* *ioSettings*

);

Parameter：

void* handle
    [in,out] The created target printer object.
const TCHAR* ioSettings
    [in]Set up the parameter of communication port that connected to the target
        printer.Please see as below:

Configuration List：

Type	Configuration	Description	Sample
USB	**USB**[,*Position/Model/PortNum]*	USB：connect any USB printer of our company USB[,*Position]*：When connecting to multi printers of our company, can specify connecting to one particular USB printer through USB position information (Position parameter)	USB USB,Port_#0004.Hub_#0003 USB,LPG4 USB,USB001
NET	**NET**, *IP Add (IPV4)[,Port]*	Specify the IP add and port of internet printer. If not specifying port, the default port is 9100.	NET,192.168.0.36 NET,192.168.0.36,9100
COM	**COM***n***,**BAUDRATE_*rate*	Specify the number and baud rate of connected serial port .	COM5,BAUDRATE_19200
BSP	**BSPn**,BAUDRATE_*rate*	If you use WinCE device or Windows mobile, the specified. port is BSPn, generally is BSP2.	BSP2,BAUDRATE_115200
BT	**BT**,Bluetooth address	Specify the Bluetooth address	BT, 8CDE52992EEF

Note：[ ] indicates selective parameter.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_OPEN_FAILED	-311	Port open failed

4.5 PortClose

This function is to close the communication port and disconnect with the printer.

int PortClose(

    void* *handle*

);

Parameter：

void* handle
    [in,out] The created target printer object.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle

4.6 WriteData

This function is to send data to the printer.

int WriteData(

    void* *handle,*

    unsigned char* *writeData,*

    unsigned int *writeNum*

);

Parameter：

void* handle
    [in,out] The created target printer object.
unsigned char* writeData
    [in] The data sent to the printer (hex string).
unsigned int writeNum
    [in] The length of the data sent.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout
E_IO_READ_FAILED	-331	Read failed
E_IO_READ_TIMEOUT	-332	Read timeout

4.7 ReadData

This function is to read the printer data.

int ReadData(

    void* *handle,*

    unsigned char* *readData,*

    unsigned int *readNum,*

    unsigned int* *preadedNum*

);

Parameter：

void* handle
    [in,out] The created target printer object.
unsigned char* readData
    [in] Printer data that needs to be read.
unsigned int readNum
    [in] The length of data that needs to be read.
unsigned int* preadedNum
    [in] The length of the data actually read.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_READ_FAILED	-331	Read failed
E_IO_READ_TIMEOUT	-332	Read timeout

4.8 DirectIO

This function is for the user to customize the data sent and read by the printer. If some functions do not provide a function interface, the user can send command data to the printer through this interface.

int DirectIO(

    void* handle,

    unsigned char* writedata,

    unsigned int writeNum,

    unsigned char* readdata,

    unsigned int readNum,

    unsigned int* preadedNum

);

Parameter：

void* handle
    [in,out] The created target printer object.
unsigned char* writedata
    [in] The data written to the printer.
unsigned int writeNum
    [in] The length of the data written to the printer.
    When writeNum=0, the write data operation is not performed.
unsigned char* readdata
    [in,out] Get the data returned by the printer.
unsigned int readNum
    [in] Preset the length of data that needs to be read.
    When readNum=0, the read data operation is not performed.
unsigned int* preadedNum
    [in,out] The length of the data actually read.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout
E_IO_READ_FAILED	-331	Read failed
E_IO_READ_TIMEOUT	-332	Read timeout

4.9 CPCL_AddLabel

This function is to set the label size and the number of prints.

int CPCL_AddLabel(

    void* *handle,*

    int *offSet,*

    int *height,*

    int *qty*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int offSet
    [in] The starting offset of the tag (unit: dot).
int height
    [in] The height of the printed label (range: 0-2400, unit: dot).
int qty
    [in] The number of labels printed.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.10 CPCL_SetAlign

This function is to set the text alignment.

int CPCL_SetAlign(

    void* *handle,*

    int *align*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int align
    [in] Set the text alignment.

Position	Value
left	0
intermediate	1
right	2

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.11 CPCL_AddText

This function is to print text.

int CPCL_AddText(

    void* *handle,*

    int *rotate,*

    int *fontType,*

    int *fontSize,*

    int *xPos,*

    int *yPos,*

    const TCHAR* *data*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int rotate
    [in] Set the print orientation.

Rotation angle	Value
Not rotating	0
Rotate 90 degrees	1
Rotate 180 degrees	2
Rotate 270 degrees	3

int fontType

    [in] Font type.

Font	Value
Font_9x17	1
Font_8x16	2
Font_20x20	3
Font_32x32 or Font_16x32	4
Font_24x24 or Font_12x24(depending on Chinese and English)	7
Font_24x24 or Font_12x24(depending on Chinese and English)	8
Font_16x16 or Font_8x16(depending on Chinese and English)	20
Font_24x21 or Font_12x24(depending on Chinese and English)	24
Font_16x16 or Font_8x16(depending on Chinese and English)	55

int fontSize

    [in] Font size (range: 0-7).

int xPos

    [in] Horizontal starting position.

int yPos

    [in] Vertical starting position.

const TCHAR* data

    [in] Text data.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.12 CPCL_AddBarCode

This function is to print bar codes.

int CPCL_AddBarCode(

    void* *handle,*

    int *rotate,*

    int *type,*

    int *width,*

    int *ratio,*

    int *height,*

    int *xPos,*

    int *yPos,*

    const TCHAR* *data*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int rotate
    [in] Set the rotation mode.
     0: no rotation
     1: Rotate 90 degrees

int type
 [in] Set the barcode type.

Barcode type	Value
Code 128	0
Code 128A	1
Code 128B	2
Code 128C	3
Code 128 Extended	4
Code 39	5
Code 39 with Check Digit	6
Code 93	7
CodaBar	8
CodaBar with Checksum	9
EAN-13	10
EAN-13 Plus 2	11
EAN-13 Plus 5	12
EAN-8	13
EAN-8 Plus 2	14
EAN-8 Plus 5	15
Code 39 Full	16
Code 39 Full With Check Digit	17
Facing Identification Mark	18
Interleaved 2 of 5	19
I 2 of 5 with Checksum	20
German Post Code	21
MSI	24
MSI10	25
MSI1010	26
MSI1110	27

int width
 [in] Set the barcode width (unit: dot).
int ratio
 [in] Bar code black and white block width ratio.
int height
 [in] Set the bar code height (unit: dot).
int xPos
 [in] Horizontal starting position.
int yPos
 [in] Vertical starting position.
const TCHAR* data
 [in] Barcode data.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.13 CPCL_AddBarCodeText

This function is to display the bar code content.

int CPCL_AddBarCodeText(

    void* *handle,*

    int *enable,*

    int *fontType,*

    int *fontSize,*

    int *offset*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int enable
    [in] Whether to display barcode content
     0 : not displayed
     1 : Display
int fontType
    [in] Font type (range: 0-8).
int fontSize
    [in] Font size.
int offset
    [in] Displacement distance.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.14 CPCL_AddQRCode

This function is to print QR codes.

int CPCL_AddQRCode(

    void* *handle,*

    int *rotate,*

    int *xPos,*

    int *yPos,*

    int *model,*

    int *unitWidth,*

    int *eccLevel,*

    const TCHAR* *data*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int rotate
    [in] Set the rotation mode.
     0: no rotation
     1: Rotate 90 degrees
int xPos
    [in] Horizontal starting position.
int yPos
    [in] Vertical starting position.
int model
    [in] Set the QR code version (1: Basic, 2: Enhanced).
int unitWidth
    [in] Set the QR code width.

int eccLevel
    [in] Error correction level.

Fault tolerance level	Value
7%	0
15%	1
25%	2
30%	3

const TCHAR data*
    [in] QR code data.

Return value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_INVALID_MODEL_TYPE	-3	This model does not support this feature.
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.15 CPCL_AddPDF417

This function is to print PDF417 code.

int CPCL_AddPDF417(

> **void*** *handle,*
>
> **int** *rotate,*
>
> **int** *xPos,*
>
> **int** *yPos,*
>
> **int** *xDots,*
>
> **int** *yDots,*
>
> **int** *columns,*
>
> **int** *rows,*
>
> **int** *eccLevel,*
>
> **const TCHAR*** *data*

);

Parameter：

void* handle
> [in,out]The created target printer object.

int rotate
> [in] Set the rotation mode.
> 0: no rotation
> 1: Rotate 90 degrees

int xPos
> [in] Horizontal starting position.

int yPos
> [in] Vertical starting position.

int xDots
    [in] Pixel width.
int yDots
    [in] Pixel height.
int columns
    [in] The number of barcode columns.
int rows
    [in] The number of barcode lines.
int eccLevel
    [in] Error correction level.

Degree of fault tolerance	Value
0	0
2	1
6	2
14	3
30	4
62	5
126	6
254	7
510	8

const TCHAR data*
    [in] PDF417 code data.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.16 CPCL_AddBox

This function is to draw a rectangular box.

int CPCL_AddBox(

>**void*** *handle,*

>**int** *xPos,*

>**int** *yPos,*

>**int** *endXPos,*

>**int** *endYPos,*

>**int** *thickness*

);

Parameter：

void handle*
>[in,out] The created target printer object.
int xPos
>[in] Horizontal starting position (unit: dot).
int yPos
>[in] Vertical starting position (unit: dot).
int endXPos
>[in] Horizontal end position (unit: dot).
int endYPos
>[in] Vertical end position (unit: dot).
int thickness
>[in] The width of the rectangle border.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.17 CPCL_AddLine

This function is to draw a line.

int CPCL_AddLine(

    void* *handle,*

    int *xPos,*

    int *yPos,*

    int *endXPos,*

    int *endYPos,*

    int *thickness*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int xPos
    [in] Horizontal starting position (unit: dot).
int yPos
    [in] Vertical starting position (unit: dot).
int endXPos
    [in] Horizontal end position (unit: dot).
int endYPos
    [in] Vertical end position (unit: dot).
int thickness
    [in] The width of the line.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.18 CPCL_AddImage

This function is to print pictures (format: bmp, jpg, gif, etc).

int CPCL_AddImage(

  void* *handle,*

  int *rotate,*

  int *xPos,*

  int *yPos,*

  const TCHAR* *filePath*

);

Parameter：

void* handle
    [in,out]The created target printer object.
int rotate
    [in] Set the picture rotation mode.
     0: no rotation
     1: Rotate 90 degrees
int xPos
    [in] Horizontal starting position (unit: dot).
int yPos
    [in] Vertical starting position (unit: dot).
const TCHAR* filePath
    [in] The correct path to the picture.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.19 CPCL_AddImageData

This function is to print the picture (directly into the picture pixel data).

int CPCL_AddImageData(

    void* *handle,*

    int *rotate,*

    int *widthBytes,*

    int *height,*

    int *xPos,*

    int *yPos,*

    const char* *Data*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int rotate
    [in] Set the picture rotation mode.
     0: no rotation
     1: Rotate 90 degrees
int widthBytes
    [in] Image data width.
int height
    [in] Image height.
int xPos
    [in] Horizontal starting position (unit: dot).
int yPos
    [in] Vertical starting position (unit: dot).
const char Data*
    [in] Image data.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.20 CPCL_SetFontSize

This function is to set the font size.

int CPCL_SetFontSize(

    void* *handle,*

    int *width,*

    int *height*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int width
    [in] Set the font width (width magnification: 0-16).
int height
    [in] Set the font height (height magnification: 0-16).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.21 CPCL_SetDensity

This function is to set the print density.

int CPCL_SetDensity(

    void* *handle,*

    int *density*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int density
    [in] Printing density(range: 0-3).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_INVALID_MODEL_TYPE	-3	This model does not support this feature.
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.22 CPCL_SetSpeed

This function is to set the print speed.

int CPCL_SetSpeed(

    void* *handle,*

    int *speed*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int speed
    [in] Printing speed(range: 0-5).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.23 CPCL_SetTextSpacing

This function is to set the character spacing.

int CPCL_SetTextSpacing(

void* *handle,*

int *spacing*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int spacing
    [in] Character spacing(range: 0-255).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.24 CPCL_SetLeftMargin

This function is to set the value of the left margin when starting printing.

int CPCL_SetLeftMargin(

    void* *handle,*

    int *margin*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int margin
    [in] The value of the left margin (range: 0-999).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.25 CPCL_SetTextBold

This function is to set the font boldness.

int CPCL_SetTextBold(

    void* *handle,*

    int *bold*

);

Parameter：

void* handle
    [in,out] The created target printer object.
int bold
    [in] Font boldness (range: 0-5).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.26 CPCL_SetTextUnderline

This function is to set the text underline.

int CPCL_SetTextUnderline(

    void* *handle,*

    int *underline*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int underline
    [in] Underline.
      0: Turn off the underline
      1: Activate underline

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.27 CPCL_Abort

This function is to terminate the current control session without printing.

int CPCL_Abort(

    void* *handle*

);

parameter：

void handle*
    [in,out] The created target printer object.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.28 CPCL_Print

This function is to print the labels.

int CPCL_Print(

    void* *handle*

);

Parameter：

void* handle
    [in,out] The created target printer object.

Return Value：

Error Code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.29 CPCL_NextLabelPos

This function is to feed the paper to the next label.

int CPCL_NextLabelPos(

> **void*** *handle*

);

Parameter：

void* handle
> [in,out] The created target printer object.

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.30 CPCL_PreFeed

This function is to feed the paper to the specified distance before printing the label.

int CPCL_PreFeed(

    void* *handle,*

    int *distance*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int distance
    [in] The distance (range: -4000-4000, unit: dot).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout

4.31 CPCL_PostFeed

This function is to feed the paper to the specified distance after printing the label.

int CPCL_PostFeed(

    void* *handle,*

    int *distance*

);

Parameter：

void handle*
    [in,out] The created target printer object.
int distance
    [in] The distance (range: -4000-4000, unit: dot).

Return Value：

Error code	Value	Description
E_SUCCESS	0	Normal
E_INVALID_PARAMETER	-1	Invalid parameter
E_BAD_HANDLE	-6	Invalid handle
E_IO_PORT_NOT_OPEN	-309	Communication port not open
E_IO_WRITE_FAILED	-321	Write failed
E_IO_WRITE_TIMEOUT	-322	Write timeout