

[CPCL WindowsCE SDK]

[打印机 CPCL 指令集开发帮助文档 v1.0]

1. 手册信息	3
2. 运行平台	3
3. 备注	3
4. 方法	4
4.1 PrinterCreator	4
4.2 PrinterCreatorS	5
4.3 PrinterDestroy	6
4.4 PortOpen	7
4.5 PortClose	9
4.6 WriteData	10
4.7 ReadData	11
4.8 DirectIO	12
4.9 CPCL_AddLabel	14
4.10 CPCL_SetAlign	15
4.11 CPCL_AddText	16
4.12 CPCL_AddBarCode	18
4.13 CPCL_AddBarCodeText	21
4.14 CPCL_AddQRCode	23
4.15 CPCL_AddPDF417	25
4.16 CPCL_AddBox	27
4.17 CPCL_AddLine	29
4.18 CPCL_AddImage	31
4.19 CPCL_AddImageData	33
4.20 CPCL_SetFontSize	35
4.21 CPCL_SetDensity	36
4.22 CPCL_SetSpeed	37
4.23 CPCL_SetTextSpacing	38
4.24 CPCL_SetLeftMargin	39
4.25 CPCL_SetTextBold	40
4.26 CPCL_SetTextUnderline	41
4.27 CPCL_Abort	42
4.28 CPCL_Print	43
4.29 CPCL_NextLabelPos	44
4.30 CPCL_PreFeed	45
4.31 CPCL_PostFeed	46

1. 手册信息

本 SDK 手册提供了 WinCE 应用程序开发所需的*.dll 文件信息。

我们在不断地努力提高和升级我们所有产品的功能与质量。

之后，产品规格和用户手册的内容可能会更改，将不再另行通知。

2. 运行平台

- WinCE 4.0/5.0/6.0
- Windows Mobile

3. 备注

- 错误代码返回值大于0时，属于 Windows 系统内部错误，请查阅相关帮助文档。

4. 方法

4.1 PrinterCreator

此函数功能为创建指定机型的打印机对象（在进行任何打印机操作之前必须先创建打印机对象）。

```
int PrinterCreator(  
  
    void* handle,  
  
    const char* model  
  
);
```

参数：

```
void* handle  
    [in,out] 创建目标打印机对象。  
const char* model  
    [in] 指定目标打印机型号。
```

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_NO_ENOUGMEMORY	-2	计算机内存不足
E_INVALID_MODEL	-8	机型名称无效

4.2 PrinterCreatorS

此函数功能为创建指定机型的目标打印机，其功能与 `PrinterCreator` 相同（使用任何打印机操作之前必须先创建打印机对象）。

```
void* PrinterCreatorS(  
  
    const char* model  
  
);
```

参数：

const char model*
[in] 指定目标打印机型号。

返回值：

成功:返回打印机对象的句柄。

失败:返回 `NULL`，无效句柄。

4.3 PrinterDestroy

此函数功能为释放已创建指定机型打印机对象的资源（在操作结束后且不再进行打印机操作时必须释放创建的打印机对象）。

```
int PrinterDestroy(  
  
    void* handle  
  
);
```

参数：

void handle*
[in] 需要释放的目标打印机对象。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效

4.4 PortOpen

打开通讯端口，与打印建立连接。连接成功后才能正常使用其它功能。连接失败时，请查看函数返回的错误信息。

```
int PortOpen(  
    void* handle,  
    const TCHAR* ioSettings  
);
```

参数：

void handle*

[in,out] 创建的目标打印机对象。

const TCHAR ioSettings*

[in] 设置连接目标打印机的通讯端口参数。具体内容查看下表：

配置列表：

类别	配置	描述	示例
USB	USB [,Position/Model/PortNum]	USB：连接任一本公司 USB 打印机。 USB[,Position] ：当同时连接本公司多台打印机时，可以通过 USB 位置信息(Position 参数)来指定连接某一特定 USB 端口的打印机。	USB USB,Port_#0004.Hub_#0003 USB,LPG4 USB,USB001
NET	NET , IP Add (IPV4)[,Port]	指定网络打印机的 IP 地址和端口。如果不指定端口，默认端口是9100。	NET,192.168.0.36 NET,192.168.0.36,9100
COM	COMn ,BAUDRATE_rate	指定连接的串口端口号和波特率。	COM5,BAUDRATE_19200
BSP	BSPn ,BAUDRATE_rate	使用 WinCE 设备或者使用 Windows mobile，指定的端口是 BSPn，一般是 BSP2。	BSP2, BAUDRATE_115200
BT	BT ,Bluetooth address	指定蓝牙地址。	BT, 8CDE52992EEF

注：[]表示可选参数。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_OPEN_FAILED	-311	端口打开失败

4.5 PortClose

此函数功能为关闭通讯端口。

```
int PortClose(  
  
    void* handle  
  
);
```

参数:

void handle*
[in,out] 创建的目标打印机对象。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效

4.6 WriteData

此函数功能为用户自定义向打印机发送数据。

```
int WriteData(  
  
    void* handle,  
  
    unsigned char* writeData,  
  
    unsigned int writeNum  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- unsigned char* writeData*
[in] 发给打印机的数据，数据是十六进制字符串。
- unsigned int writeNum*
[in] 发送数据的长度。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常。
E_BAD_HANDLE	-6	句柄无效。
E_IO_PORT_NOT_OPEN	-309	通信端口未打开。
E_IO_WRITE_FAILED	-321	写入失败。
E_IO_WRITE_TIMEOUT	-322	写入超时。
E_IO_READ_FAILED	-331	读取失败。
E_IO_READ_TIMEOUT	-332	读取超时。

4.7 ReadData

此函数功能为读取打印机的数据。

```
int ReadData(  
  
    void* handle,  
  
    unsigned char* readData,  
  
    unsigned int readNum,  
  
    unsigned int* preadedNum  
  
);
```

参数：

- void* handle*
[in,out] 创建的目标打印机对象。
- unsigned char* readData*
[in] 需要读取的打印机数据。
- unsigned int readNum*
[in] 所需读取的数据长度。
- unsigned int* preadedNum*
[in] 实际读取到的数据长度。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常。
E_BAD_HANDLE	-6	句柄无效。
E_IO_PORT_NOT_OPEN	-309	通信端口未打开。
E_IO_READ_FAILED	-331	读取失败。
E_IO_READ_TIMEOUT	-332	读取超时。

4.8 DirectIO

此函数功能为用户自定义发送和读取打印机的数据。如果某些功能未提供函数接口时，用户可以通过此接口向打印机发送指令数据。

```
int DirectIO(  
  
    void* handle,  
  
    unsigned char* writedata,  
  
    unsigned int writeNum,  
  
    unsigned char* readdata,  
  
    unsigned int readNum,  
  
    unsigned int* preadedNum  
  
);
```

参数：

void handle*

[in,out] 创建的目标打印机对象。

unsigned char writedata*

[in] 写入打印机的数据。

unsigned int writeNum

[in] 写入打印机的数据长度。当 writeNum=0时，不执行写入数据操作。

unsigned char readdata*

[in,out] 获取打印机返回的数据。

unsigned int readNum

[in] 预设需要读取的数据长度。当 readNum=0时，不执行读取数据操作。

unsigned int preadedNum*

[in,out] 实际读取的数据长度。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时
E_IO_READ_FAILED	-331	读取数据失败
E_IO_READ_TIMEOUT	-332	读取数据超时

4.9 CPCL_AddLabel

此函数功能为设置标签大小和打印的数量。

```
int CPCL_AddLabel(  
  
    void* handle,  
  
    int offSet,  
  
    int height,  
  
    int qty  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int offSet*
[in] 标签的起始偏移量(单位: dot)。
- int height*
[in] 打印的标签高度(范围: 0-2400, 单位: dot)。
- int qty*
[in] 打印的标签数量。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.10 CPCL_SetAlign

此函数功能为设置文本对齐方式。

```
int CPCL_SetAlign(  
  
    void* handle,  
  
    int align  
  
);
```

参数:

void handle*
[in,out] 创建的目标打印机对象。
int align
[in]设置文本对齐方式。

位置	值
左边	0
中间	1
右边	2

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.11 CPCL_AddText

此函数功能为打印文本。

```
int CPCL_AddText(  
  
    void* handle,  
  
    int rotate,  
  
    int fontType,  
  
    int fontSize,  
  
    int xPos,  
  
    int yPos,  
  
    const TCHAR* data  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int rotate*
[in] 设置打印方向。

旋转角度	值
不旋转	0
旋转90度	1
旋转180度	2
旋转270度	3

int *fontType*
[in] 字体类型。

字体点阵	值
Font_9x17	1
Font_8x16	2
Font_20x20	3
Font_32x32 or Font_16x32	4
Font_24x24 or Font_12x24(视中英文而定)	7
Font_24x24 or Font_12x24(视中英文而定)	8
Font_16x16 or Font_8x16(视中英文而定)	20
Font_24x21 or Font_12x24(视中英文而定)	24
Font_16x16 or Font_8x16(视中英文而定)	55

int *fontSize*
[in] 字体大小(范围: 0-7)。

int *xPos*
[in] 水平起始位置。

int *yPos*
[in] 垂直起始位置。

*const TCHAR** *data*
[in] 文本数据。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.12 CPCL_AddBarCode

此函数功能为打印条码。

```
int CPCL_AddBarCode(  
    void* handle,  
    int rotate,  
    int type,  
    int width,  
    int ratio,  
    int height,  
    int xPos,  
    int yPos,  
    const TCHAR* data  
);
```

参数:

void* *handle*

[in,out] 创建的目标打印机对象。

int *rotate*

[in] 设置旋转模式。

0: 不旋转

1: 旋转90度

int type

[in] 设置条码类型。

条码类型	值
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] 设置条码宽度(单位: dot)。

int ratio

[in] 条码黑白块宽窄比率。

int height

[in] 设置条码高度(单位: dot)。

int xPos

[in] 水平起始位置。

int yPos

[in] 垂直起始位置。

const TCHAR data*

[in] 条码数据。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.13 CPCL_AddBarCodeText

此函数功能为显示条码内容。

```
int CPCL_AddBarCodeText(  
  
    void* handle,  
  
    int enable,  
  
    int fontType,  
  
    int fontSize,  
  
    int offset  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int enable

[in] 是否显示条码内容。

0 :不显示

1 :显示

int fontType

[in] 字体类型(范围: 0-8)。

int fontSize

[in] 字体大小。

int offset

[in] 位移距离。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.14 CPCL_AddQRCode

此函数功能为打印二维条码。

```
int CPCL_AddQRCode(  
  
    void* handle,  
  
    int rotate,  
  
    int xPos,  
  
    int yPos,  
  
    int model,  
  
    int unitWidth,  
  
    const TCHAR* data  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int rotate

[in] 设置旋转模式。

0: 不旋转

1: 旋转90度

int xPos

[in] 水平起始位置。

int yPos

[in] 垂直起始位置。

int model

[in] 设置二维码版本(1:基本版,2:强化版)。

int unitWidth

[in] 设置二维码宽度。

int eccLevel

[in] 纠错级别。

容错级别	值
7%	0
15%	1
25%	2
30%	3

const TCHAR data*

[in] 二维码数据。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.15 CPCL_AddPDF417

此函数功能为打印 PDF417二维条码。

```
int CPCL_AddPDF417(  
  
    void* handle,  
  
    int rotate,  
  
    int xPos,  
  
    int yPos,  
  
    int xDots,  
  
    int yDots,  
  
    int columns,  
  
    int rows,  
  
    int eccLevel,  
  
    const TCHAR* data  
  
);
```

参数：

void handle*

[in,out] 创建的目标打印机对象。

int rotate

[in] 设置旋转模式。

0: 不旋转

1: 旋转90度

int xPos

[in] 水平起始位置。

int yPos

[in] 垂直起始位置。

int xDots
[in] 像素点宽度。

int yDots
[in] 像素点高度。

int columns
[in] 条码列数。

int rows
[in] 条码行数。

int eccLevel
[in] 纠错级别。

容错程度	值
0	0
2	1
6	2
14	3
30	4
62	5
126	6
254	7
510	8

const TCHAR data*
[in] PDF417二维条码数据。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.16 CPCL_AddBox

此函数功能为打印机画出矩形框。

```
int CPCL_AddBox(  
  
    void* handle,  
  
    int xPos,  
  
    int yPos,  
  
    int endXPos,  
  
    int endYPos,  
  
    int thickness  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int xPos

[in] 水平起始位置(单位: dot)。

int yPos

[in] 垂直起始位置(单位: dot)。

int endXPos

[in] 水平结束位置(单位: dot)。

int endYPos

[in] 垂直结束位置(单位: dot)。

int thickness

[in] 矩形边框宽度。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.17 CPCL_AddLine

此函数功能为画线。

```
int CPCL_AddLine(  
  
    void* handle,  
  
    int xPos,  
  
    int yPos,  
  
    int endXPos,  
  
    int endYPos,  
  
    int thickness  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int xPos

[in] 水平起始位置(单位: dot)。

int yPos

[in] 垂直起始位置(单位: dot)。

int endXPos

[in] 水平结束位置(单位: dot)。

int endYPos

[in] 垂直结束位置(单位: dot)。

int thickness

[in] 直线宽度。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.18 CPCL_AddImage

此函数功能为打印图片(格式: bmp、jpg、gif 等)。

```
int CPCL_AddImage(  
  
    void* handle,  
  
    int rotate,  
  
    int xPos,  
  
    int yPos,  
  
    const TCHAR* filePath  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int rotate

[in] 设置图片旋转模式。

0: 不旋转

1: 旋转90度

int xPos

[in] 水平起始位置。

int yPos

[in] 垂直起始位置。

const TCHAR filePath*

[in] 图片的正确路径。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_NOT_ENOUGH_MEMORY	-9	计算机内存不足
E_IMAGE_BAD_SIZE	-25	图片大小错误
E_IO_PORT_NOT_OPEN	-309	通信端口未打开
E_IO_WRITE_FAILED	-321	写入失败
E_IO_WRITE_TIMEOUT	-322	写入超时

4.19 CPCL_AddImageData

此函数功能为打印图片（直接传入图片像素数据）。

```
int CPCL_AddImageData(  
  
    void* handle,  
  
    int rotate,  
  
    int widthBytes,  
  
    int height,  
  
    int xPos,  
  
    int yPos,  
  
    const char* Data  
  
);
```

参数：

void* *handle*

[in,out] 创建的目标打印机对象。

int *rotate*

[in] 设置图片旋转模式。

0: 不旋转

1: 旋转90度

int *widthBytes*

[in] 图像数据宽度。

int *height*

[in] 图像高度。

int *xPos*

[in] 水平起始位置。

int *yPos*

[in] 垂直起始位置。

const char* *Data*

[in] 图像数据。

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_NOT_ENOUGH_MEMORY	-9	计算机内存不足
E_IMAGE_BAD_SIZE	-25	图片大小错误
E_IO_PORT_NOT_OPEN	-309	通信端口未打开
E_IO_WRITE_FAILED	-321	写入失败
E_IO_WRITE_TIMEOUT	-322	写入超时

4.20 CPCL_SetFontSize

此函数功能为设置字体大小。

```
int CPCL_SetFontSize(  
  
    void* handle,  
  
    int width,  
  
    int height  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int width*
[in] 设置字体宽度(宽度放大倍数: 0-16)。
- int height*
[in] 设置字体高度(高度放大倍数: 0-127)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.21 CPCL_SetDensity

此函数功能为设置打印浓度。

```
int CPCL_SetDensity(  
  
    void* handle,  
  
    int density  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int density

[in] 打印浓度(范围: 0-3)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.22 CPCL_SetSpeed

此函数功能为设置打印速度。

```
int CPCL_SetSpeed(  
  
    void* handle,  
  
    int speed  
  
);
```

参数:

void handle*

[in,out] 创建的目标打印机对象。

int speed

[in] 打印速度(范围: 0-5)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.23 CPCL_SetTextSpacing

此函数功能为设置字符间距。

```
int CPCL_SetTextSpacing(  
  
    void* handle,  
  
    int spacing  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int spacing*
[in] 字符间距(范围: 0-255)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.24 CPCL_SetLeftMargin

此函数功能为设置起始打印时左边距的值。

```
int CPCL_SetLeftMargin(  
  
    void* handle,  
  
    int margin  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int margin*
[in] 左边距的值(范围: 0-999)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.25 CPCL_SetTextBold

此函数功能为设置字体加粗程度。

```
int CPCL_SetTextBold(  
  
    void* handle,  
  
    int bold  
  
);
```

参数:

void handle*
[in,out] 创建的目标打印机对象。
int bold
[in] 字体加粗程度(范围: 0-5)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.26 CPCL_SetTextUnderline

此函数功能为设置文本下划线。

```
int CPCL_SetTextUnderline(  
  
    void* handle,  
  
    int underline  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int underline*
[in] 下划线。
0: 关闭下划线,
1: 激活下划线

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.27 CPCL_Abort

此函数功能为终止当前控制会话而不打印。

```
int CPCL_Abort(  
  
    void* handle  
  
);
```

参数:

void handle*
[in,out] 创建的目标打印机对象。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.28 CPCL_Print

此函数功能为打印标签。

```
int CPCL_Print(  
  
    void* handle  
  
);
```

参数：

```
void* handle  
    [in,out] 创建的目标打印机对象。
```

返回值：

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.29 CPCL_NextLabelPos

此函数功能为走纸到下一张标签。

```
int CPCL_NextLabelPos(  
  
    void* handle  
  
);
```

参数:

```
void* handle  
    [in,out] 创建的目标打印机对象。
```

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.30 CPCL_PreFeed

此函数功能为在打印标签前走纸指定的距离。

```
int CPCL_PreFeed(  
  
    void* handle,  
  
    int distance  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int distance*
[in] 走纸距离(范围: -4000-4000, 单位: dot)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时

4.31 CPCL_PostFeed

此函数功能为在打印标签后走纸指定的距离。

```
int CPCL_PostFeed(  
  
    void* handle,  
  
    int distance  
  
);
```

参数:

- void* handle*
[in,out] 创建的目标打印机对象。
- int distance*
[in] 走纸距离(范围: -4000-4000, 单位: dot)。

返回值:

错误代码	值	描述
E_SUCCESS	0	正常
E_INVALID_PARAMETER	-1	无效的参数
E_BAD_HANDLE	-6	句柄无效
E_IO_PORT_NOT_OPEN	-309	端口未打开
E_IO_WRITE_FAILED	-321	写入数据失败
E_IO_WRITE_TIMEOUT	-322	写入数据超时