

# 动态连结函式库(DLL)的使用

我们提供了使用于Windows95/98/NT/ME/2000/XP等作业系统的DLL给使用者，档名为ez2000.dll。使用该DLL之前必须先安装由Scientific Software Tools, Inc.所撰写的DLPortIO，档名为port95nt.exe。若未先安装DLPortIO即使用ez2000.dll则会出现".dll not found"的错误讯息。

另外在Windows95/98作业系统的使用者，也可使用档名为ez98.dll的DLL，该DLL所提供的功能相同于ez2000.dll且执行效率也较高，并且不需要安装DLPortIO(port95nt.exe)。

## I. DLL 函式库表

函数	使用说明
1. setup	设定条码机的相关资料
2. setbaudrate	设定PC的Baud Rate
3. SetTimeOutTicks	设定传送与接收时的逾时时间
4. openport	指定条码机的输出埠
5. closeport	关闭所开启的输出埠
6. sendcommand	送出EZPL命令(EZPL命令使用说明请参考命令手册)
7. isready	检查条码机是否处于可列印的状态
8. intloadimage	载入图形档到条码机内部记忆体(PCX or BMP)
9. extloadimage	载入图形档到条码机外加记忆体(PCX or BMP)
10. ecTextOut	指定True Type字型做为输出文字的字型
11. ecTextOutR	指定True Type字型作为输出文字的字型，并提供字型旋转角度的设定
12.ecTextDownLoad	指定True Type字型和旋转角度下载至记忆体，可利用Y命令 (EZPL command)输出该字型
13.putimage	列印多色调图形(支援BMP和JPG)及设定图形的旋转角度，并以半色调技术处理后输出
14. downloadimage	下载多色调图形(支援BMP和JPG)及设定图形的旋转角度至记忆体中，可利用Y命令(EZPL command)输出该图形
15. SelectUsbPrinter	指定欲开启连结的USB Printer的USB识别字串
16. SelectUsbPortNumber	指定欲开启连结的USB Printer所使用的USB port
17. readusb	读取USB port回传的资料

## II. DLL 函式库使用说明

### 1. setup

功能说明	设定条码机的相关资料
命令格式	setup(a,b,c,d,e,f)
输入参数	a:(数值) 设定标签尺寸, 单位为mm b:(数值) 设定列印的明暗度 (0~19) c:(数值) 设定列印速度 (1~3) d:(数值) 设定所使用的标签种类 0 → 有间距的标签纸 1 → 连续纸 2 → 黑线标记的标签纸 e:(数值) 设定间距的宽度, 若不使用, 则将其设为 0 (单位: mm) f:(数值) 设定黑线的宽度, 若不使用, 则将其设为 0 (单位: mm)

### 2. setbaudrate

功能说明	设定PC的Baud Rate
命令格式	setbaudrate(n)
输入参数	n:(数值) 设定值可选择4800, 9600, 19200, 38400

### 3. SetTimeOutTicks

功能说明	设定传送与接收时的逾时时间
命令格式	SetTimeOutTicks(time)
输入参数	time:(数值) 设定逾时时间的数值, 单位为毫秒(ms)

### 4. openport

功能说明	指定条码机的输出埠
命令格式	openport(n)
输入参数	n:(字串) 指定输出埠的代表号 n = 0 ; LPT1 n = 1 ; COM1 n = 2 ; COM2 n = 3 ; COM3 n = 4 ; COM4 n = 5 ; LPT2 n = 6 ; USB

### 5. closeport

功能说明	关闭所开启的输出埠
命令格式	closeport()
输入参数	无

### 6. sendcommand

功能说明	送出印表机EZPL命令(EZPL命令使用说明请参考命令手册)
命令格式	sendcommand(command)
输入参数	(字串) EZPL命令内容

## 7. isready

功能说明	检查条码机是否处于可列印的状态
命令格式	isready()
输入参数	无
回传值	1 = (数值) 表示条码机可以开始接收资料 0 = (数值) 表示条码机无法接收资料

## 8. intloadimage

功能说明	载入图形档到条码机内部记忆体
命令格式	intloadimage(filename, image_name, image_type)
输入参数	filename : (字串) 图形档的档名 image_name : (字串) 载入图形的名称, 可以使用与图形档相同的名称或另外指定 image_type : (字串) 载入图形的格式 pcx → 以PCX为副档名的图形档 bmp → 以BMP为副档名的图形档

## 9. extloadimage

功能说明	载入图形档到条码机外加记忆体
命令格式	extloadimage(filename, image_name, image_type)
输入参数	filename : (字串) 图形档的档名 image_name : (字串) 载入图形的名称, 可使用与图形档相同的名称或另外指定 image_type : (字串) 载入图形的格式 pcx → 以PCX为副档名的图形档 bmp → 以BMP为副档名的图形档

## 10. ecTextOut

功能说明	指定Windows中所提供的字型做为输出文字的字型
命令格式	ecTextOut(x,y,b,c,d)
输入参数	x : (数值) 设定文字所在位置的X轴座标(dots) y : (数值) 设定文字所在位置的Y轴座标(dots) b : (数值) 设定文字的高度 c : (字串) 指定所使用的字型名称 d : (字串) 所要列印的文字资料
回传值	1 = (数值) 成功 0 = (数值) 失败

## 11. ecTextOutR

功能说明	指定True Type字型作为输出文字的字型，并提供字型旋转角度的设定
命令格式	ecTextOutR(x,y,b,c,d,e,f,g)
输入参数	x:(数值) 设定文字所在位置的X轴座标(dots) y:(数值) 设定文字所在位置的Y轴座标(dots) b:(数值) 设定文字的高度 c:(字串) 指定所使用的字型名称 d:(字串) 所要列印的文字资料 e:(数值) 设定文字宽度(预设值=0) f:(数值) 设定文字线条粗细(预设值=40, 容许调整值为 0~100) g:(数值) 输出文字的旋转角度 0→旋转零度; 90→旋转90度; 180→旋转180度; 270→旋转270度
回传值	1=(数值) 成功 0=(数值) 失败

## 12. ecTextDownload

功能说明	指定True Type字型和旋转角度下载至记忆体，可利用EZPL的Y命令输出该字型
命令格式	ecTextDownload(b,c,d,e,f,g,name)
输入参数	b:(数值) 设定文字的高度 c:(字串) 指定所使用的字型名称 d:(字串) 所要下载的文字资料 e:(数值) 设定文字宽度 f:(数值) 设定文字线条粗细(预设值=40, 容许调整值为0~100) g:(数值) 输出文字的旋转角度 0→旋转零度; 90→旋转90度; 180→旋转180度; 270→旋转270度 name:(字串) 指定字型在记忆体中的名称，以便使用EZPL的Y命令输出该字型
回传值	1=(数值) 成功 0=(数值) 失败

## 13. putimage

功能说明	列印多色调图形(支援BMP和JPG)及设定图形的旋转角度，并以半色调技术处理后输出
命令格式	putimage(x,y,filename,degree)
输入参数	x:(数值) 设定图形所在位置的X轴座标(dots) y:(数值) 设定图形所在位置的Y轴座标(dots) filename:(字串) 图档名称(支援BMP和JPG图档格式) degree:(数值) 旋转角度, 0→不旋转; 90→旋转90度
回传值	1=(数值) 成功 0=(数值) 失败

#### 14. downloadimage

功能说明	可下载多色调图形(支援BMP和JPG)及设定图形的旋转角度至记忆体中, 可利用EZPL的Y命令输出该图形
命令格式	downloadimage(filename,degree,name)
输入参数	filename : (字串) 图档名称(支援BMP和JPG图档格式) degree : (数值) 旋转角度, 0→不旋转; 90→旋转90度 name : (字串) 指定图形在记忆体中的名称, 以便使用EZPL的Y命令输出图形
回传值	1 = (数值) 成功 0 = (数值) 失败

#### 15. SelectUsbPrinter

功能说明	指定欲开启连结的USB Printer的USB识别字串
命令格式	SelectUsbPrinter(ID)
输入参数	ID : (字串) USB Printer的USB识别字串, 限定为8位元

#### 16. SelectUsbPortNumber

功能说明	指定欲开启连结的USB Printer所使用的USB port
命令格式	SelectUsbPortNumber(portnumber)
输入参数	portnumber : (数值) 设定USB port数值, 范围为0 ~ 127

#### 17. readusb

功能说明	读取USB port回传的资料
命令格式	readusb(buff,length)
输入参数	buff : (字串) 设定欲接收资料的储存位址 length : (数值) 设定欲接收资料的容量大小, 单位为byte

### III. 使用范例

#### A. 开启USB Printer连结范例

```
SelectUsbPrinter("01234587");  
SelectUsbPortNumber(2);
```

```
SetTimeOutTicks(1000);  
openport(6);
```

```
sendcommand("~B");  
Sleep(200);  
readusb(buff,&length);  
buff[length] = '\0';
```

```
closeport();
```

## B. FoxPro 5.0 for Windows 95

```
path="c:\userdata\qlabel31\ez2000.dll"
declare mb[10]
value=0

i=0
j=0
for i=1 to 10
mb[i]=0
next

SET LIBRARY TO foxtools.fll ADDITIVE
mb [1]= RegFn32("openport","C","",path)
mb [2] = RegFn32("setup","IIIII","",path)
mb [3] = RegFn32("closeport","", "",path)
mb [4] = RegFn32("sendcommand","C","",path)
mb [5] = RegFn32("intloadimage","CCC","",path)
mb [6] = RegFn32("extloadimage","CCC","",path)
mb [7] = RegFn32("isready","", "I",path)
mb [8] = RegFn32("ecTextOut","IIICC","",path)

for j=1 to 7
  if mb[j]= -1
    RELEASE LIBRARY FoxTools.FLL
    WAIT WINDOW "FAILURE!!" NOWAIT
    RETURN
  endif
next

value = CallFn(mb[7])
if value=1
  mem1="变数测试--使用TT的中文字"
  = CallFn(mb[1],"0")
  = CallFn(mb[2],25,5,2,1,0,0)
  = CallFn(mb[4],"^P1")
  = CallFn(mb[4],"^L")
  = CallFn(mb[4],"BA,15,100,2,4,25,0,1,12345")
  = CallFn(mb[4],"AZ,10,10,1,1,1,0,&mem1")
  = CallFn(mb[8],10,45,50,"细明体","Window 中文字型")
  = CallFn(mb[4],"E")
else
  wait windows "PRINTER ERROR" nowait
endif

RELEASE LIBRARY FoxTools.FL
```

## C. Visual Basic 5.0 for Windows 98

```
Private Declare Sub setup Lib "Ez98.dll" (ByVal a%, ByVal b%, ByVal c%, ByVal d%,  
ByVal e%, ByVal f%)  
Private Declare Sub openport Lib "Ez98.dll" (ByVal command$)  
Private Declare Sub sendcommand Lib "Ez98.dll" (ByVal command$)  
Private Declare Sub intloadimage Lib "Ez98.dll" (ByVal filename$, ByVal  
image_name$, ByVal image_type$)  
Private Declare Sub extloadimage Lib "Ez98.dll" (ByVal filename$, ByVal  
image_name$, ByVal image_type$)  
Private Declare Sub ecTextOut Lib "Ez98.dll" (ByVal x%, ByVal y%, ByVal b%, ByVal  
c$, ByVal d$)  
Private Declare Sub closeport Lib "Ez98.dll" ()  
Private Sub command1_Click()  
Call openport("0")  
Call setup(30, 7, 2, 1, 0, 0)  
Call sendcommand("^W70")  
Call sendcommand("^P1")  
Call sendcommand("^L")  
Call sendcommand("AC,20,60,1,1,1,0,TEST")  
Call ecTextOut(20, 10, 34, "标楷体", "中文测试")  
Call sendcommand("E") Call closeport  
End Sub
```