开发人员帮助文档

格瑞斯打印机使用 TSPL2 指令。具体指令请参考 TSCLIB.DLL 函数库使用说明和 TSPL2 指令集(中文版)。格瑞斯 GRIS-05GZ 腕带打印机、GRIS-03GR 瓶签打印机只用到 TSCLIB.DLL 函数库中的部分函数和少量的 TSPL2 指令。

下面是格瑞斯打印机打印腕带和标签所用到的 TSC 部分函数指令，及指令的详细设置说明。
重要提示：请在开发及测试之前，需务必先安装好条形码打印机及驱动程序。
注意：指令中的所有字符务必为英文字符

1、openport(a) 指令：

说明: 指定计算机端的输出端口，即指定打印条码的打印机。

方法：openport(“打印机名”)
实例：openport("格瑞斯　GRIS-05GZ")

2、setup(a,b,c,d,e,f,g) 指令：

说明: 设定腕带或标签的宽度、高度、打印速度、打印浓度、传感器类别、gap/black mark、垂直间距、gap/black mark 偏移距离)
参数：

A	字符串型	设定标签**宽度**，单位 mm
B	字符串型	设定标签**高度**，单位 mm
C	字符串型	设定**打印速度**，(打印速度随机型不同而有不同的选项)
		1：每秒 1.0 英寸打印速度
		1.5：每秒 1.5 英寸打印速度
		2：每秒 2.0 英寸打印速度
		3：每秒 3.0 英寸打印速度
		4：每秒 4.0 英寸打印速度
		5：每秒 5.0 英寸打印速度
		6：每秒 6.0 英寸打印速度
D	字符串型	设定**打印浓度**，0~15，数字越大打印结果越黑
E	字符串型	设定使用传感器类别
		0 表示使用垂直间距传感器(gap sensor)
		1 表示使用黑标传感器(black mark sensor)
F	字符串型	设定 gap/black mark 垂直间距高度，单位: mm
G	字符串型	设定 gap/black mark 偏移距离，单位: mm，此参数若使用一般标签时均设为 0

方法：**setup(A 宽度数字,B 高度数字,C 打印速度数字,D 打印浓度数字,E 感测器类型（腕带填写：1，标签填写：0），F 黑标或瓶签宽度数字（腕带填写：3，标签填写：3）,G 偏离距离一般不启用设为：0)**

以 280x30 常规尺寸腕带为例：
腕带实例：setup("30", "190", "4", "6", "1", "3", "0")　　// 成人和儿童腕带高度设为：190　　新生儿高度设为：140

"30"为**宽度**，对应表中 **A**
"190"为**高度**，对应表中 **B**
"4" 为**打印速度**，对应表中 **C**
"6"为**打印浓度**，对应表中 **D**
"1"为**感测器类型**，1 为腕带，0 为输液标签对应表中 **E**
"3" 材料为腕带时，代表腕带下方**黑标宽度**；瓶签时，代表**瓶签间距**，对应表中 **F**
"0"为**偏离距离**设置，一般情况下都不启用，设置为 0，对应表中 **G**

以 60x80 常规尺寸瓶签为例：
瓶签实例：setup("60", "80", "4", "10", "0", "3", "0")　　// 瓶签宽高设置为实际尺寸

特别注意：黑标打印设置的打印尺寸实际上是打印机的“感应区域”，而非“实际尺寸”，打印标签时才需设置“实际尺寸”。HIS 软件内的 “设置尺寸”应和打印机的“设置尺寸”保持一致，成人腕带、儿童腕带部分新生儿腕带的设置相同，无论“实际尺寸”为多少，均设置为 30mm x 190mm；“实际尺寸”长度≤（小于等于）190mm 的新生儿腕带的“设置尺寸”与前几者略有不同，“设置尺寸”应为 30mm x 140mm。

关于不同腕带材质下，**打印浓度**设置的一点建议：
常规型腕带打印浓度为 4 - 6
柔软型腕带打印浓度为 6 - 8
杜邦纸腕带打印浓度为 10 - 12
热敏型腕带（瓶签）打印浓度为 8 - 10
除了浓度设置和介质设置有差异，相同尺寸但不同材质的腕带其他设置并无差异

3、 clearbuffer() 指令：

说明：清理打印机内部缓存

方法：clearbuffer()
实例：clearbuffer()　　//唯一书写方式，不能在括号里加任何内容

4、windowsfont(a,b,c,d,e,f,g,h) 指令：

说明：使用 Windows TTF 字型打印文字
参数：

A	整数型	文字 X （）方向起始点，以点(point)表示。
B	整数型	文字 Y 方向起始点，以点(point)表示。
C	整数型	字体大小，以点(point)表示。
D	整数型	旋转角度，逆时钟方向旋转
		0 -> 0 degree
		90-> 90 degree
		180-> 180 degree
		270-> 270 degree
		字体外形

E	整数型	**0-> 标准(Normal)**
		1-> 斜体(Italic)
		2-> 粗体(Bold)
		3-> 粗斜体(Bold and Italic)
F	整数型	底线
		0-> 无底线
		1-> 加底线
G	整数型	字符串型，字体名称。如: Arial, Times new Roman，细名体，标楷体
H	整数型	打印文字内容

方法：windowsfont (A 文字上下位置,B 文字左右位置,C 字体大小,D 文字旋转,E 字体属性（粗、细、斜）, F 文字是否加底线（0 不加、1 加）,G 字体，H 文字内容

以 280x30 常规尺寸腕带为例：

```
windowsfont(165, 135, 19, 270, 2, 0, "宋体", "姓名：张三")
```

165 为文字**上下位置**对应表 **A**，数字越大越靠上；
135 为文字**左右位置**对应表 B，数字越大越靠右；
19 为**字体大小**，数字越大文字越大对应表 **C**；
270 为**文字旋转**角度对应表 **D**；
2 为**字体属性**对应表 **E**；
0 为不启用**底线**，对应表 F；
"宋体"为**文本字体**，对应表 G；
"姓名：张三"为**文字内容**，可以替换为一个变量，对应表 H；

5、printlabel(a,b)指令：

说明: 打印标签内容
a: 字符串型别，设定打印标签个数(set)
b: 字符串型别，设定打印标签份数(copy)

方法：**printlabel(a 数字,b 数字)**
实例：printlabel("1", "1")

6、closeport() 指令：

说明: 关闭指定的计算机端输出端口
方法：closeport()
实例：closeport()　　//唯一书写方式，不能在括号里加任何内容

7、sendcommand(command)指令：

说明：以 TSCLIB.DLL 中的 sendcommand 为媒介，直接向打印机发送任意指令。该函数非常强大，可以直接绕过 TSCLIB.DLL 函数库中 sendcommand 以外的指令来实现上面的所有功能。用于弥补 DLL 中不存在的入口点。
参数：此命令支持直接向打印机发送 TSPL2 指令集中的任意函数指令。（主要用于调用 DLL 函数库中没有的函数，如二维码打印等）

方法：sendcommand(指令函数)

在 TSPL2 中的语法为 QRCODE x,y,ECC Level,cell width,mode,rotation,[model,mask,]"content"
我们可直接用 sendcommand(QRCODE x,y,ECC Level,cell width,mode,rotation,[model,mask,]"content") 配合 DLL 来调用

以 C# WPF 打印二维码为例可写为：

新增如下 Namespace 来调用 DLL：

```
using System.Runtime.InteropServices;
```

新建一个 public class，命名为： WD_DLL

```
public class WD_DLL
{
[DllImport("TSCLIB.dll", EntryPoint = "sendcommand")]
public static extern int sendcommand(string printercommand);
}
```

程序中调用：

```
WD_DLL.sendcommand("QRCODE 175,720,L,6,A,90,M2,S0, \"" + str_rqcode + "\"");    // str_rqcode 为一个变量
```

以上为打印腕带和瓶签需要使用的所有机器指令。

需要打印整条腕带时，具体代码如下例：

新建 wpf 项，命名为 “Wpf_wdqrcode”

```
// 调用相关Namespace
using System.Windows;
using System.Runtime.InteropServices;

public class WD_DLL        // 创建类库，命名为 WD_DLL

{
```

```
    [DllImport("TSCLIB.dll", EntryPoint = "openport")]
    public static extern int openport(string printername);       // 引用DLL函数：打印机名称，入口点为 openport

    [DllImport("TSCLIB.dll", EntryPoint = "setup")]          // 引用DLL函数：打印机首选项设置，入口点为 setup
    public static extern int setup(string width, string height,
          string speed, string density,
          string sensor, string vertical,
          string offset);

    [DllImport("TSCLIB.dll", EntryPoint = "clearbuffer")]     // 引用DLL函数：清理打印机缓存，入口点为
clearbuffer
    public static extern int clearbuffer();

    [DllImport("TSCLIB.dll", EntryPoint = "windowsfont")]   // 引用DLL函数：调用系统字体，入口点为 windowsfont
    public static extern int windowsfont(int x, int y, int fontheight,
                int rotation, int fontstyle, int fontunderline,
                string szFaceName, string content);

    [DllImport("TSCLIB.dll", EntryPoint = "sendcommand")]     // 引用DLL函数：发送机器指令，入口点为 sendcommand
    public static extern int sendcommand(string printercommand);

    [DllImport("TSCLIB.dll", EntryPoint = "printlabel")]     // 引用DLL函数：设置打印分数，入口点为 printlabel
    public static extern int printlabel(string set, string copy);

    [DllImport("TSCLIB.dll", EntryPoint = "closeport")]    // 引用DLL函数：关闭驱动，入口点为 closeport
    public static extern int closeport();


}

namespace Wpf_wdqrcode                          // 创建一个namespace，命名为 Wpf_wdqrcode
{
    /// <summary>
    /// MainWindow.xaml 的交互逻辑
    /// </summary>
    public partial class MainWindow : Window
    {
        public MainWindow()
        {
            InitializeComponent();
        }

        private void Button_Click(object sender, RoutedEventArgs e)
        {

            string str_name, str_sex, str_age, str_bedno, str_ks, str_zyh;       // 创建打印文本的对应变量
```

```
str_name = "姓名：" + "调用姓名";
str_sex = "性别：" + "调用性别";
str_age = "年龄：" + "调用年龄";
str_bedno = "床号：" + "调用床号";
str_ks = "科室：" + "调用科室";
str_zyh = "住院号：" + "55219949";

if (str_ks == "新生儿科")       // 根据科室判断调用打印参数

{
    //158x30新生儿腕带建议参数：

    WD_DLL.openport("格瑞斯  GRIS-05GZ");
    WD_DLL.setup("30", "140", "4", "8", "1", "3", "0");
    WD_DLL.clearbuffer();
    WD_DLL.sendcommand("QRCODE 163,375,L,4,A,90,M2,S0,\"55219949\"");
    WD_DLL.windowsfont(165, 135, 19, 270, 2, 0, "宋体", str_name);
    WD_DLL.windowsfont(165, 275, 19, 270, 2, 0, "宋体", str_sex);
    WD_DLL.windowsfont(125, 135, 19, 270, 2, 0, "宋体", str_age);
    WD_DLL.windowsfont(125, 275, 19, 270, 2, 0, "宋体", str_bedno);
    WD_DLL.windowsfont(85, 135, 19, 270, 2, 0, "宋体", str_ks);
    WD_DLL.windowsfont(85, 275, 19, 270, 2, 0, "宋体", str_zyh);
    WD_DLL.printlabel("1", "1");
    WD_DLL.closeport();
}

else if (str_ks == "儿科")           // 根据科室判断调用打印参数

{
    //220x30儿童腕带建议参数：

    WD_DLL.openport("格瑞斯  GRIS-05GZ");
    WD_DLL.setup("30", "190", "4", "8", "1", "3", "0");
    WD_DLL.clearbuffer();
    WD_DLL.sendcommand("QRCODE 160,615,L,5,A,90,M2,S0,\"55219949\"");
    WD_DLL.windowsfont(180, 175, 25, 270, 2, 0, "宋体", str_name);
    WD_DLL.windowsfont(180, 395, 25, 270, 2, 0, "宋体", str_sex);
    WD_DLL.windowsfont(130, 175, 25, 270, 2, 0, "宋体", str_age);
    WD_DLL.windowsfont(130, 395, 25, 270, 2, 0, "宋体", str_bedno);
    WD_DLL.windowsfont(80, 175, 25, 270, 2, 0, "宋体", str_ks);
    WD_DLL.windowsfont(80, 395, 25, 270, 2, 0, "宋体", str_zyh);
    WD_DLL.printlabel("1", "1");
    WD_DLL.closeport();
}

else            // 根据科室判断调用打印参数
```

```
            {
                //280x30成人腕带建议参数：

                WD_DLL.openport("格瑞斯  GRIS-05GZ");
                WD_DLL.setup("30", "190", "4", "6", "1", "3", "0");
                WD_DLL.clearbuffer();
                WD_DLL.sendcommand("QRCODE 175,720,L,6,A,90,M2,S0,\"55219949\"");
                WD_DLL.windowsfont(180, 190, 30, 270, 2, 0, "宋体", str_name);
                WD_DLL.windowsfont(180, 470, 30, 270, 2, 0, "宋体", str_sex);
                WD_DLL.windowsfont(130, 190, 30, 270, 2, 0, "宋体", str_age);
                WD_DLL.windowsfont(130, 470, 30, 270, 2, 0, "宋体", str_bedno);
                WD_DLL.windowsfont(80, 190, 30, 270, 2, 0, "宋体", str_ks);
                WD_DLL.windowsfont(80, 470, 30, 270, 2, 0, "宋体", str_zyh);
                WD_DLL.printlabel("1", "1");
                WD_DLL.closeport();
            }

        }
    }
}
```

上面是C#的完整调用指令，如果换成其他语言，方法类似，机器语言完全一样，例如Delphi，打印腕带部分可写为：

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls;

type
  TForm1 = class(TForm)
    Button1: TButton;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
```

```
{$R *.DFM}

procedure openport(PrinterName:pchar);stdcall;far; external 'tsclib.dll';
procedure closeport; external 'tsclib.dll';
procedure sendcommand(Command:pchar);stdcall;far;external 'tsclib.dll';
procedure setup(LabelWidth, LabelHeight, Speed, Density, Sensor, Vertical, Offset:pchar);stdcall; far; external 'tsclib.dll';
procedure printerfont(X, Y, FontName, Rotation, Xmul, Ymul, Content:pchar);stdcall;far; external 'tsclib.dll';
procedure clearbuffer; external 'tsclib.dll';
procedure printlabel(NumberOfSet, NumberOfCopoy:pchar);stdcall; far;external 'tsclib.dll';
procedure windowsfont (X, Y, FontHeight, Rotation, FontStyle, FontUnderline : integer; FaceName,

procedure TForm1.Button1Click(Sender: TObject);
var
   i: Integer;

begin

openport('格瑞斯   GRIS-05GZ');                                          //调用打印机驱动
for i:=1 to 1 do
begin
setup('30', '190', '4', '6', '1', '3', '0');                          //设置宽度为“30mm”，高度为“190mm”（成人和儿童腕带设为190mm，新生儿腕带需设置为140mm），打印速度为“4寸/秒”，打印浓度为“6”，黑标感应开启“1”，黑标宽度为“3”mm
clearbuffer;                                                                  //清理打印机缓存
sendcommand('QRCODE 165,740,L,5,A,90,M2,S0,"55219949"');          //自动生成二维码。“55219949”可替换为变量 。备注：“165”为上下位置，数字越大越靠上；“740”为左右位置，数字越大越靠右；第5项“6”为二维码大小调整级别为1~10
windowsfont(180, 190, 30, 270, 2, 0, '宋体',   '姓名：张三');       //写入文本内容,“姓名：张三”可以替换为变量;(“180”为上下位置，数字越大越靠上；“190”为左右位置，数字越大越靠右；30为字体大小，数字越大文字越大）
windowsfont(180, 470, 30, 270, 2, 0, '宋体', '性别：男');           //写入文本内容，“性别：男”可以替换为变量
windowsfont(130, 190, 30, 270, 2, 0, '宋体', '年龄：34');           //写入文本内容，“年龄：34”可以替换为变量
windowsfont(130, 470, 30, 270, 2, 0, '宋体', '床号：12');           //写入文本内容，“床号：12”可以替换为变量
windowsfont(80, 190, 30, 270, 2, 0, '宋体', '科室：骨科');          //写入文本内容，“科室：骨科”可以替换为变量
windowsfont(80, 470, 30, 270, 2, 0, '宋体', '住院号：55219949'); //写入文本内容，“住院号：55219949”可以替换为变量
printlabel('1', '1');                                                        //打印内容为 1份
            end;
               closeport;

end;

end.
```

以上为两个打印腕带的实例，标签打印机GRIS-03GR类似，只要设置好setup再用windowsfont调整打印位置即可。瓶签Setup的设置请详细阅读上面第3节。

我们提供了目前最常见十几种程序实例，打印腕带需要更改打印机相应参数以下是参数实例：

//280x30成人腕带建议参数：

```
openport("格瑞斯　GRIS-05GZ");
 //调用打印机驱动
setup("30", "190", "4", "6", "1", "3", "0");
//设置宽度为“30mm”，高度为“190mm”（成人和儿童腕带设为190mm，新生儿腕带需设置为140mm），打印速度为“4寸/秒”，打印浓度为“6”，黑标感应开启“1”，黑标宽度为“3”mm
clearbuffer();
//清理打印机缓存
sendcommand("QRCODE 175,720,L,6,A,90,M2,S0,\"55219949\"");
//自动生成二维码。“55219949”可替换为变量 。备注：“175”为上下位置，数字越大越靠上；“720”为左右位置，数字越大越靠右；第5项“6”为二维码大小调整级别为1~10
windowsfont(180, 190, 30, 270, 2, 0, "宋体", "姓名：昆山格瑞斯");
//写入文本内容，“姓名：昆山格瑞斯”可以替换为变量；（“180”为上下位置，数字越大越靠上；“210”为左右位置，数字越大越靠右；30为字体大小，数字越大文字越大）
windowsfont(180, 470, 30, 270, 2, 0, "宋体", "性别：男");
//写入文本内容，“性别：男”可以替换为变量
windowsfont(130, 190, 30, 270, 2, 0, "宋体", "年龄：77");
//写入文本内容，“年龄：77”可以替换为变量
windowsfont(130, 470, 30, 270, 2, 0, "宋体", "床号：24");
//写入文本内容，“床号：24”可以替换为变量
windowsfont(80, 190, 30, 270, 2, 0, "宋体", "科室：骨科一区");
//写入文本内容，“科室：骨科一区”可以替换为变量
windowsfont(80, 470, 30, 270, 2, 0, "宋体", "住院号：55219949");
//写入文本内容，“住院号：55219949”可以替换为变量
printlabel("1", "1");
//合并打印腕带内容
closeport();
//关闭驱动端口
```

//220x30儿童腕带建议参数：

```
openport("格瑞斯　GRIS-05GZ");
//调用打印机驱动
setup("30", "190", "4", "8", "1", "3", "0");
//设置宽度为“30mm”，高度为“190mm”（成人和儿童腕带设为190mm，新生儿腕带需设置为140mm），打印速度为“4寸/秒”，打印浓度为“6”，黑标感应开启“1”，黑标宽度为“3”mm
clearbuffer();
//清理打印机缓存
sendcommand("QRCODE 160,615,L,5,A,90,M2,S0,\"55219949\"");
//自动生成二维码。“55219949”可替换为变量 。备注：“160”为上下位置，数字越大越靠上；“615”为左右位置，数字越大越靠右；第5项“5”为二维码大小调整级别为1~10
windowsfont(180, 175, 25, 270, 2, 0, "宋体", "姓名：昆山格瑞斯");
```

```
//写入文本内容，“姓名：昆山格瑞斯”可以替换为变量；（“180”为上下位置，数字越大越靠上；“175”为左右位置，数字越大越靠右；30为字体大小，数字越大文字越大）
windowsfont(180, 395, 25, 270, 2, 0, "宋体", "性别：男");
//写入文本内容，“性别：男”可以替换为变量
windowsfont(130, 175, 25, 270, 2, 0, "宋体", "年龄：7");
 //写入文本内容，“年龄：7”可以替换为变量
windowsfont(130, 395, 25, 270, 2, 0, "宋体", "床号：21");
//写入文本内容，“床号：21”可以替换为变量
windowsfont(80, 175, 25, 270, 2, 0, "宋体", "科室：儿科一区");
//写入文本内容，“科室：儿科一区”可以替换为变量
windowsfont(80, 395, 25, 270, 2, 0, "宋体", "住院号：55219949");
//写入文本内容，“住院号：55219949”可以替换为变量
printlabel("1", "1");
//合并打印腕带内容
closeport();
//关闭驱动端口



//158x30新生儿腕带建议参数：

openport("格瑞斯   GRIS-05GZ");
//调用打印机驱动
setup("30", "140", "4", "8", "1", "3", "0");
//设置宽度为“30mm”，高度为“140mm”（成人和儿童腕带设为190mm，新生儿腕带需设置为140mm），打印速度为“4寸/秒”，打印浓度为“6”，黑标感应开启“1”，黑标宽度为“3”mm
clearbuffer();
//清理打印机缓存
sendcommand("QRCODE 163,375,L,4,A,90,M2,S0,\"55219949\"");
//自动生成二维码。“55219949”可替换为变量 。备注：“163”为上下位置，数字越大越靠上；“375”为左右位置，数字越大越靠右；第5项“4”为二维码大小调整级别为1~10
windowsfont(165, 135, 19, 270, 2, 0, "宋体", "姓名：格瑞斯");
//写入文本内容，“姓名：格瑞斯”可以替换为变量；（“165”为上下位置，数字越大越靠上；“135”为左右位置，数字越大越靠右；“19”为字体大小，数字越大字体越大）
windowsfont(165, 275, 19, 270, 2, 0, "宋体", "性别：女");
//写入文本内容，“性别：女”可以替换为一个变量
windowsfont(125, 135, 19, 270, 2, 0, "宋体", "年龄：12天");
//写入文本内容，“年龄：12天”可以替换为一个变量
windowsfont(125, 275, 19, 270, 2, 0, "宋体", "床号：A3");
//写入文本内容，“床号：A3”可以替换为一个变量
windowsfont(85, 135, 19, 270, 2, 0, "宋体", "科室：新生儿科");
//写入文本内容，“科室：新生儿科”可以替换为一个变量
windowsfont(85, 275, 19, 270, 2, 0, "宋体", "住院号：55219949");
//写入文本内容，“住院号：55219949”可以替换为一个变量
printlabel("1", "1");
//合并打印腕带内容
closeport();
```

//关闭驱动端口

备注：
windowsfont项可根据实际情况调整打印内容和打印位置。

打印一维条型码调用参数如下：

先在类库中添加如下参数：

```
[DllImport("TSCLIB.dll", EntryPoint = "barcode")]
    public static extern int barcode(string x, string y, string type,
                string height, string readable, string rotation,
                string narrow, string wide, string code);
```

类库指令调用：

```
barcode("175", "750", "128", "100", "1", "90", "2", "2", "051255219949");
 // 自动生成一维码。"051255219949" 可替换为变量 。备注：“175”为上下位置，数字越大越靠上；“750”为左右位置，数字越大越靠右；第4项"100"为条码高度，可设置小一些；第5项“1”是否显示一维码数字，1为显示0为隐藏。
```