

# [TSPL Linux SDK]

[打印机 TSPL 指令集开发帮助文档 v2.0.4]

## 目录

[TSPL Linux SDK] .....	1
1. 手册信息 .....	3
2. 操作系统 .....	3
3. 备注 .....	3
4. 方法 .....	3
4.1. InitPrinter .....	3
4.2. ReleasePrinter .....	3
4.3. OpenPort .....	4
4.4. ClosePort .....	5
4.5. WriteData .....	5
4.6. ReadData .....	6
4.7. TSPL_SelfTest .....	6
4.8. TSPL_Bar .....	7
4.9. TSPL_BarCode .....	7
4.10. TSPL_BitMap .....	9
4.11. TSPL_Image .....	10
4.12. TSPL_Setup .....	11
4.13. TSPL_ClearBuffer .....	12
4.14. TSPL_Box .....	12
4.15. TSPL_QrCode .....	13
4.16. TSPL_Text .....	14
4.17. TSPL_Print .....	15
4.18. TSPL_FormFeed .....	16
4.19. TSPL_SetTear .....	16
4.20. TSPL_SetRibbon .....	17
4.21. TSPL_Offset .....	17
4.22. TSPL_Direction .....	18
4.23. TSPL_Feed .....	18
4.24. TSPL_Home .....	19
4.25. TSPL_Learn .....	19
4.26. TSPL_GetSN .....	20
4.27. TSPL_GetPrinterStatus .....	20
4.28. TSPL_SetCodePage .....	21
4.29. TSPL_PDF417 .....	22
4.30. TSPL_Block .....	24
4.31. TSPL_Reverse .....	25
4.32. TSPL_GapDetect .....	26
4.33. TSPL_Dmatrix .....	26
4.34. TSPL_SetCutter .....	27

# 1. 手册信息

本 SDK 手册提供了 Linux 应用程序开发所需的 so 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. 操作系统

Linux debian 5.10.0 及以上版本

# 3. 备注

1. 错误代码返回值大于0时，属于 Linux 系统内部错误，请查阅相关帮助文档。
2. 打印机分辨率为200 dpi时，1 mm=8 dot(点);打印机分辨率为300 dpi时，1 mm=12 dot(点)。
3. SDK中引用了第三方库:libserialport、libusb-1.0。请提前在操作系统安装。
4. 串口连接需要root权限。

# 4. 方法

## 4.1.InitPrinter

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)。

```
void* InitPrinter (  
    const TCHAR* model  
);
```

参数：

*const TCHAR\* 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

## 4.2.ReleasePrinter

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 ReleasePrinter (
    void* hPrinter
);
```

参数：

*void\* hPrinter*

[in] 需要释放的目标打印机对象的句柄。

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败

### 4.3.OpenPort

此函数功能为打开通讯端口，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。目前支持 USB、网络通信、串口通信。

```
int OpenPort (
    void* hPrinter,
    const TCHAR* setting
);
```

参数：

*void\* hPrinter*

[in] 打印机对象句柄。

*const TCHAR\* setting*

[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,path	USB,USB路径	USB,/001/007
NET	NET, IP 地址 (IPV4)[,端口]	指定网络打印机的IP地址和端口。如果不指定端，默认端口是9100。	NET,192.168.1.10 NET,192.168.1.10,9100
COM	COM,path,rate	指定连接的串口路径和波特率。	COM,/dev/ttyACM0,19200

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_USB_DEVICE_NOT_FOUND	-17	找不到USB设备
ERROR_IO_OPEN_FAILED	-8	打开IO失败
ERROR_CM_INVALID_PARAMETER	-1	参数无效

## 4.4. ClosePort

此函数功能为关闭通讯。当不使用端口通讯时，请关闭端口。

```
int ClosePort (  
    void* hPrinter  
);
```

参数：

**void\* hPrinter**  
[in] 创建的目标打印机对象。

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-3	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-2	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败

## 4.5. WriteData

此函数功能为向打印机发送数据。

```
int WriteData(  
    void* handle,  
    unsigned char* buffer,  
    unsigned int size  
);
```

参数：

**void\* handle**  
[in] 打印机对象句柄。  
**unsigned char\* buffer**  
[in] 发给打印机的数据，数据是十六进制字符串。  
**unsigned int size**  
[in] 发送数据的长度。

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时

## 4.6.ReadData

此函数功能为读取打印机的数据。

```
int ReadData(  
    void* handle,  
    unsigned char* buffer,  
    unsigned int size  
);
```

参数:

*void\* handle*

[in] 打印机对象句柄。

*unsigned char\* buffer*

[in] 需要读取的打印机数据。

*unsigned int size*

[in] 所需读取的数据长度。

返回值:

错误代码	值	描述
	>0	成功, 读取的数据长度
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_OPEN_FAILED	-8	打开IO失败

## 4.7.TSPL\_SelfTest

此函数功能为打印机打印自检页, 自检页中包含打印机的基本配置信息。

```
int TSPL_SelfTest(  
    void* hPrinter  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.8.TSPL\_Bar

此函数功能为绘制长条

```
int TSPL_Bar(  
    void* hPrinter,  
    int x,  
    int y,  
    int width,  
    int height  
);
```

**参数:**

*void\* hPrinter*

[in] 打印机对象句柄

*int x*

[in] 起始横坐标，单位为点

*int y*

[in] 起始纵坐标，单位为点

*int width*

[in] 宽度，单位为点

*int height*

[in] 高度，单位为点

**返回值:**

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.9.TSPL\_BarCode

此函数功能为打印一维条码。

```
int TSPL_BarCode(  
    void* hPrinter,  
    int x,  
    int y,  
    int type,  
    const char* content,  
    int height,  
    int showText = 0,  
    int rotation = 0,  
    int narrow = 2,  
    int wide = 2  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 起始横坐标，单位为点。

*int y*

[in] 起始纵坐标，单位为点。

*int type*

[in] 设置条码类型。

条码类型	值
Code 128	0
Code 128M	1
EAN 128	2
Interleaved 2 of 5	3
Interleaved 2 of 5 with check digits	4
Code 39	5
Code 39C	6
Code 93	7
EAN 13	8
EAN 13 with 2 digits add-on	9
EAN 13 with 5 digits add-on	10
EAN 8	11
EAN 8 with 2 digits add-on	12
EAN 8 with 5 digits add-on	13
Codabar	14
Postnet	15
UPC-A	16
UPC-A with 2 digits add-on	17
UPC-A with 5 digits add-on	18
UPC- E	19
UPC- E with 2 digits add-on	20
UPC- E with 5 digits add-on	21
China post code	22
MSI code	23
MSI with check digit	24
PLESSEY code	25
ITF 14 code	26
EAN 14 code	27
Code 11	28
TELEPEN	29
TELEPENN	30
PLANET	31
CODE_49	32
DPI	33
DPL	34

*const char\* content*

[in] 条码数据。

*int height*

[in] 高度，单位为点。

*int showText*

[in] 文本显示类型



显示类型	值
不显示文本	0
居左显示	1
居中显示	2
居右显示	3

*int rotation*

[in] 旋转方向。

旋转方向	值
不旋转	0
旋转90度	1
旋转180度	2
旋转270度	3

*int narrow*

[in] 窄 Bar 宽度, 以点(dot)表示。

*int wide*

[in] 宽 Bar 宽度, 以点(dot)表示。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.10. TSPL\_BitMap

此函数功能为绘制二值化后的位图。

```
int TSPL_BitMap(
    void* hPrinter,
    int x,
    int y,
    int width,
    int height,
    int mode,
    unsigned char * data
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 水平起始位置(范围: 0-32000, 单位: dot)。

*int y*

[in] 垂直起始位置(范围: 0-32000, 单位: dot)。

*int width*

[in] 图片宽度 (单位 字节)

*int height*

[in] 图片高度 (单位 点)

*int mode*

[in] 打印图片的模式。

模式	值
OVERWRITE	0
OR	1
XOR	2

*unsigned char \* data*

[in] 二值化后的位图数据

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.11. TSPL\_Image

此函数功能为打印指定的图片(仅支持单色bmp格式), 此函数适用于打印 LOGO。将指定路径的图片转化为位图数据发送给打印机并且打印。

```
int TSPL_Image(  
    void* hPrinter,  
    int x,  
    int y,  
    int mode,  
    const char* imgPath  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 起始横坐标, 单位为点。

*int y*

[in] 起始纵坐标, 单位为点。

*int mode*

[in] 打印图片的模式。

模式	值
OVERWRITE	0
OR	1
XOR	2

*const char\*imgPath*

[in] 位图文件完整路径。

返回值:

错误代码	值	描述
------	---	----

ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.12. TSPL\_Setup

此函数功能为打印机基本参数设置。

```
int TSPL_Setup(
    void* hPrinter,
    int printSpeed,
    int printDensity,
    int labelWidth,
    int labelHeight,
    int labelType,
    int gapHeight,
    int offset
);
```

**参数：**

*void\* hPrinter*

[in] 打印机对象句柄。

*int printSpeed*

[in] 打印速度。

*int printDensity*

[in] 打印浓度(范围: 0- 15)。

*int labelWidth*

[in] 标签宽度(单位:mm)。

*int labelHeight*

[in] 设标签长度(单位:mm)。

*int labelType*

[in] 标签纸类型(0：黑标/连续纸 1：缝标/连续纸)。

*int gapHeight*

[in] 两张标签之间的间距，如果设置为0，表示当前为连续纸(单位:mm)。

*int offset*

[in] 偏移位置(单位:mm)。

**返回值：**

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.13. TSPL\_ClearBuffer

此函数功能为清除打印机内存缓存。执行打印数据前,请先清除打印机缓存。

```
int TSPL_ClearBuffer(  
    void* hPrinter  
);
```

参数:

*void\* hPrinter*  
[in] 打印机对象句柄。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.14. TSPL\_Box

此函数功能为绘制方框。

```
int TSPL_Box(  
    void* hPrinter,  
    int x,  
    int y,  
    int x_end,  
    int y_end,  
    int thickness = 1,  
    int radius = 0  
);
```

参数:

*void\* hPrinter*  
[in] 打印机对象句柄

*int x*  
[in] 指定左上角的 X 坐标, 单位为点

*int y*  
[in] 指定左上角的 Y 坐标, 单位为点。

*int x\_end*  
[in] 指定右下角的 X 坐标, 单位为点。

*int y\_end*  
[in] 指定右下角的 Y 坐标, 单位为点。

*int thickness*  
[in] 线条粗细, 单位为点。

*int radius*

[in] 指定是否有圆角，单位为点，默认为0。

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.15. TSPL\_QrCode

此函数功能为打印二维码。

```
TSPL_QrCode(  
    void* hPrinter,  
    int x,  
    int y,  
    int width,  
    int eccLevel,  
    int mode,  
    int rotate,  
    int model,  
    int mask,  
    const char* data  
);
```

参数：

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 水平起始位置(单位:点)。

*int y*

[in] 垂直起始位置(单位:点)。

*int width*

[in] 二维码打印宽度(范围：1-10)。

*int eccLevel*

[in] 纠错级别。

容错级别	值
7%	0
15%	1
25%	2
30%	3

*int mode*

[in] 自动/手动编码(0:自动,1:手动)。

*int rotate*

[in] 二维码旋转角度。

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 model*

[in] 二维码版本(0:基本版,1:强化版)。

*int mask*

[in] 掩码方式(范围: 0-8)。

*const char\* data*

[in] 二维码数据。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.16. TSPL\_Text

此函数功能为打印机打印文本段落。

```
int TSPL_Text(
    void* hPrinter,
    int x,
    int y,
    const char* fontName,
    const char* content,
    int rotation = 0,
    int x_multiplication = 1,
    int y_multiplication = 1,
    int alignment = 0
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 起始横坐标, 单位为点。

*int y*

[in] 起始纵坐标, 单位为点。

*const char\* fontName*

[in] 字体名称。

字体点阵	值
normal	"0"
8x12	"1"
12x20	"2"
16x24	"3"

24x32	“4”
32x48	“5”
14x19	“6”
21x27	“7”
14x25	“8”
简体中文	“TSS24.BF2”

**const char\* content**

[in] 文本内容。

**int rotation**

[in] 设置旋转角度。

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 x\_multiplication**

[in] 水平放大比率，有效参数: 1~10。

**int y\_multiplication**

[in] 垂直放大比率，有效参数: 1~10。

**int alignment**

[in] 文本的对齐模式。

对齐方向	值
默认 (居左)	0
居左	1
居中	2
居右	3

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.17. TSPL\_Print

此函数功能为执行打印操作。

```
int TSPL_Print(
    void* hPrinter,
    int num,
    int copies
);
```

参数:

**void\* hPrinter**

[in] 打印机对象句柄。

*int num*

[in] 标签个数。

*int copies*

[in] 每个标签复印个数。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.18. TSPL\_FormFeed

此函数功能为将纸张推送到下张标签的起始位置。

```
int TSPL_FormFeed(  
    void* hPrinter  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.19. TSPL\_SetTear

此函数功能为设置打印机剥离器工作模式。

```
int TSPL_SetTear(  
    void* hPrinter,  
    int mode  
);
```

参数:

*void\* hPrinter*



[in] 打印机对象句柄。

*int mode*

[in] 剥离模式，0=关闭剥离功能；1=开启剥离功能

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.20. TSPL\_SetRibbon

此函数功能为设置碳带工作模式。

```
int TSPL_SetRibbon(  
    void* hPrinter,  
    int mode  
);
```

参数：

*void\* hPrinter*

[in] 打印机对象句柄。

*int mode*

[in] 碳带模式；0=关闭碳带功能；1=开启碳带功能；2=碳带为内卷式；3=碳带为外卷式。

返回值：

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.21. TSPL\_Offset

此函数功能为定义打印机每个标签走纸额外指定长度。

```
int TSPL_Offset(  
    void* hPrinter,  
    int distance  
)
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int distance*

[in] 指定走纸长度(单位:mm)。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.22. TSPL\_Direction

此函数功能为设置打印机打印方向。

```
int TSPL_Direction(  
    void* hPrinter,  
    int direction,  
    int mirror  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int direction*

[in] 定义打印机的打印方向(0-正向打印, 1-反向打印)。

*int mirror*

[in] 定义打印内容镜像, 0-正常; 1-镜像

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.23. TSPL\_Feed

此函数功能为打印机送出规定长度的纸张, 长度的数值单位为点。

```
int TSPL_Feed(
    void* hPrinter,
    int n
);
```

**参数:**

*void\* hPrinter*

[in] 打印机对象句柄。

*int n*

[in] 送纸长度 单位为点。有效参数:  $\pm 1 \sim 9999$ ，负数时候会向后退纸。

**返回值:**

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.24. TSPL\_Home

此函数功能为打印机送纸到起始位置。

```
int TSPL_Home(
    void* hPrinter
);
```

**参数:**

*void\* hPrinter*

[in] 打印机对象句柄。

**返回值:**

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.25. TSPL\_Learn

此函数功能为打印机学习标签。

```
int TSPL_Learn(
    void* hPrinter
```

);

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.26. TSPL\_GetSN

此函数功能为获取打印机 SN 序列号。

int TSPL\_GetSN(

void\* hPrinter,

char\* snBuffer,

int bufSize

);

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*char\* snBuffer*

[in/out]用于获取序列号的缓存地址。

*int bufSize*

[in]传入的缓存大小。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.27. TSPL\_GetPrinterStatus

此函数功能为获取打印机的当前状态。

```
int TSPL_GetPrinterStatus(
    void* hPrinter,
    unsigned int* printerStatus
)
```

参数:

*void\* hPrinter*

[in]打印机对象句柄。

*int \*printerStatus*

[in,out]打印机当前状态。

状态	值
正常	0
打印头被打开	1
卡纸	2
缺纸	4
缺碳带	8
打印暂停	16
打印中	32
上盖打开	64
其它错误	128

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_READ_FAILED	-11	读取失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.28. TSPL\_SetCodePage

此函数功能为打印机设置代码页。

```
int TSPL_SetCodePage(
    void* hPrinter,
    const char* codepage
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*const char\* codepage*

[in]国际字符集。

7-bit code page		8-bit code page		Linux code page		ISO code page	
US A	USA	437	United States	1250	Central Europe	8859- 1	Latin 1

BRI	British	737	Greek	1251	Cyrillic	8859-2	Latin 2
GER	German	850	Multilingual	1252	Latin I	8859-3	Latin 3
FRE	French	851	Greek1	1253	Greek	8859-4	Batltic
DAN	Danish	852	Slavic	1254	Turkish	8859-5	Cyrillic
ITA	Italian	855	Cyrillic	1255	Hebrew	8859-6	Arabic
SPA	Spansh	857	Turkish	1256	Arabic	8859-7	Greek
SWE	Sweidsh	860	Portuguese	1257	Batltic	8859-8	Hebrew
SWI	Swiss	861	Icelandic	1258	Vietnam	8859-9	Turkish
		862	Hebrew	932	Japanese shift-JIS	8859- 10	Latin 6
		863	Canadian/Fren ch	936	simplified	8859- 15	Latin 9
		864	Arabic	949	Korean		
		865	Nordic	950	Traditional Chinese BIG5		
		866	Russian	UTF-8	UTF 8		
		869	Greek2				

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.29. TSPL\_PDF417

此函数功能为打印 PDF417二维码。

```
int TSPL_PDF417(
    void* hPrinter,
    int x,
    int y,
    int width,
    int height,int rotate,
    const char* option,
    const char* data
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 水平起始位置(单位:点)。

*int y*

[in] 垂直起始位置(单位:点)。

*int width*

[in] 条码宽度(单位:点)。

*int height*

[in] 条码高度(单位:点)。

*int rotate*

[in] 二维码旋转角度。

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

*const char\* option*

[in] 可选项(示例: E3,W2,H8)。

P	数据压缩方式: 0:自动 1:二进制模式
E	错误校验等级(0~8)
M	条码中心打印模式 0:该模式将打印在左上角对齐区域 1:将打印在中间区域
Ux,y,z	代码可读 x:可读字符指定的 x 坐标 y:可读字符指定的 y 坐标 c:每一行最大可读字符数
W	模块宽度 (2~9:dot)
H	小条码的高(4~99:dot)
R	最大行数
C	最大列数
T	是否截断 0:否 1:是
Lm	表示长度(1~2048)

*const char\* data*

[in] 二维码数据。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败, 句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

### 4.30. TSPL\_Block

此函数功能为打印段内容到标签中。

```
int TSPL_Block(  
    void* hPrinter,  
    int x,  
    int y,  
    int width,  
    int height,  
    const char* fontName,  
    const char* content,  
    int rotation = 0,  
    int x_multiplication = 1,  
    int y_multiplication = 1,  
    int alignment = 0  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 起始横坐标，单位为点。

*int y*

[in] 起始纵坐标，单位为点。

*int width*

[in] 段落宽度 (单位 点)。

*int height*

[in] 段落高度 (单位 点)。

*const char\* fontName*

[in] 字体。

字体点阵	值
normal	"0"
8x12	"1"
12x20	"2"
16x24	"3"
24x32	"4"
32x48	"5"
14x19	"6"
21x27	"7"
14x25	"8"
简体中文	"TSS24.BF2"

*const char\* content*

[in] 文本内容。

*int rotation*

[in] 旋转方向。

旋转方向	值
不旋转	0
旋转90度	1
旋转180度	2
旋转270度	3

*int x\_multiplication*



[in]水平放大比率，有效参数: 1~10。

*int y\_multiplication*

[in]垂直放大比率，有效参数。

*int alginment*

[in] 文本的对齐模式。

对齐方式	值
默认	0
左对齐	1
中对齐	2
右对齐	3

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.31. TSPL\_Reverse

将指定区域内打印内容反白, 即空白部分为黑色,预打印部分为白色。

```
int TSPL_Reverse(  
    void* hPrinter,  
    int x,  
    int y,  
    int width,  
    int height  
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄。

*int x*

[in] 水平起始位置(单位:点)。

*int y*

[in] 垂直起始位置(单位:点)。

*int width*

[in] 宽度(单位:点)。

*int heigh*

[in] 高度(单位:点)。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败

ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.32. TSPL\_GapDetect

此函数功能侦测纸张大小及间隙大小。

```
int TSPL_GapDetect(
    void* hPrinter,
    int x = 0,
    int y = 0
);
```

参数:

*void\* hPrinter*

[in] 打印机对象句柄

*int x*

[in] 纸张高度，单位为点。

*int y*

[in] 间隙高度，单位为点。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.33. TSPL\_Dmatrix

此函数功能为打印 Data Matrix 二维条码。

```
int TSPL_Dmatrix(
    void* hPrinter,
    int x,
    int y,
    int width,
    int height,
    const char* content,
    int blockSize = 0,
    int row = 10,
    int col = 10
);
```

参数:

**void\* hPrinter**

[in] 打印机对象句柄。

**int x**

[in] 起始横坐标，单位为点。

**int y**

[in] 起始纵坐标，单位为点。

**int width**

[in] 二维条码区域宽度 (单位 点)。

**int height**

[in] 二维条码区域高度 (单位 点)。

**const char\* content**

[in] content matrix 二维条码数据。

**int blockSize**

[in] 可选，模块大小。

**int row**

[in] 可选，行码，有效参数: 10~144。

**int col**

[in] 可选，栏码，有效参数: 10~144。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败
ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码

## 4.34. TSPL\_SetCutter

此函数功能为 设置切刀工作模式

```
int TSPL_SetCutter(  
    void* hPrinter,  
    int pieces  
);
```

参数:

**void\* hPrinter**

[in] 打印机对象句柄。

**int copies**

[in] 参数，0=关闭切刀功能；-1=打印任务结束后切纸；1-65535=打印多少张标签后切纸。

返回值:

错误代码	值	描述
ERROR_CM_SUCCESS	0	成功
ERROR_CM_INVALID_HANDLE	-2	失败，句柄无效
ERROR_CM_INVALID_PARAMETER	-1	参数无效
ERROR_CM_INSUFFICIENT_MEMORY	-4	申请内存失败

ERROR_IO_WRITE_FAILED	-9	写入数据失败
ERROR_IO_WRITE_TIMEOUT	-10	写入数据超时
其他值	其他值	Linux系统返回的错误码