
RD-K2C7-S

系列热敏打印机

开发手册



版权所有：北京荣达创新科技有限公司



目 录

一、概述	4
6B1.1 主要性能指标	4
7B1.2 电源连接器	4
8B1.3 操作（控制键为单按键作）	5
15B1.3.1 指示灯	5
16B1.3.2 操作键	5
17B1.3.3 操作	5
二、通讯接口	6
9B2.1 串行接口	6
18B2.1.1 接口定义	6
2.1.2 串口数据发送方法	6
三、指令系统	8
10B3.1 指令表	8
3.2 基本控制指令	10
19BUESC @	10
20BULF	10
21BU CR	10
22BUESC J	10
23BUESC d	11
24BUESC c	11
25BUHT	11
26BUESC !	11
27BUESC D n1 n2 ... nk NULL	12
28BUESC - n	13
29BUESC + n	13
30BUGS B n	14
31BUFS 2 n	14
32BUESC \$ nL nH	15
33BUESC l n	15
34BUESC Q n	15
35BUESC 1 n	16
36BUESC SP n	16
37BUESC a n	16
38BUFS r n	17
39BUESC U n	17
40BUESC V n	17
41BUESC X	18
42BUESC i	18
43BUESC m	18
44BUESC K nL nH d1 d2dk	19
45BUESC * m nL nH d1...dk	19

[HTTP://WWW.RD-CN.COM](http://www.rd-cn.com) 2 / 42



46BUGS v 0 m xL xH yL yH d1....dk	23
47BUGS h n	24
48BUGS w n.....	24
49BUGS H n.....	24
50BUGS Q n.....	25
51BUGS k.....	25
52BUGS W n.....	28
53BUEsc '	28
54BUEsc v	29
55BUFS &.....	30
56BUFS.....	30
57BUEsc 6	30
58BUEsc 7	31
ESC r d n.....	31
ESEsc R n.....	31
GS t n.....	32
FS V.....	33
GS k m v r d1.....dk	34
四、安装尺寸	36
五、打印机维护及故障排除	36
附 录.....	37
12BA 打印字符集	37
60BA.1 ASCII 码字符集	37
61BA.2 字符集 1	38
62BA.3 字符集 2	39
A.4 国际字符集	40
13BB 条码	41
63BB.1 条码编码规则	41
64BB.2 条码长度字符集表	41
14BD 安装尺寸	42

一、概述

RD-K2C7-S 系列打印机是一款配有切刀，支持全切，半切功能，打印纸宽为 57mm 的热敏式打印机，嵌入式壳体，更加方便客户的安装。

该系列打印机型号为 RD- K2C7-S-SN。数据接口为 5 芯 RS232 串行接口打印机。该系列打印机具有打印纸宽、速度快、体积小等特点，主要应用在银行排队机，超市的存包柜，游戏机，地铁优惠券等设备上。

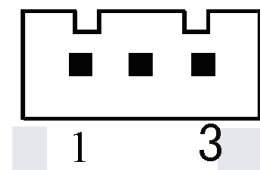
1.1 主要性能指标

打印方式	行式热敏打印	
打印纸宽	58mm	
纸卷直径	不限	
打印行宽	18汉字/行 36英文字符/行（标准24点阵高字体）	
每行点数	432点/54mm	
汉字编码方式	GBK	
内置字库	ASCII字库(16×8、24×12、8×6)、GBK(24×24)	
打印速度	170mm/s	
打印机芯寿命	100KM	
可打印内容	英文、数字、各种符号、汉字、图形、曲线、 条码(CODE39、EAN13、EAN8、CODABAR、CODE128、CODE93、ITF)	
分辨率	203 dpi ×203 dpi（8点/毫米）	
数据通讯接口	RD-K2C7-S-SN	RS232串口/TTL串口
电 源	24V/3A	
异常检测	缺纸检测	
切纸方式	全切、半切	
指 令 集	ESC/POS兼容指令集	
尺 寸	93.4mm*77mm*34mm（宽*高*厚）	
使用环境	温度 -10℃~50℃ 湿度 20%~85%	

1.2 电源连接器

电源接口-----2.54mm 间距垂直，3PIN

引脚	名称	说明
1	VCC	24V 供电
2	空	
3	GND	地



1.3 操作（控制键为单按键作）

1.3.1 指示灯

电源灯

当接通电源时，指示按键发绿色光并长亮。

状态灯

当打印机上无纸时，指示按键发绿光并闪亮。

1.3.2 操作键

进纸键

点按指示按键使打印机走纸一行；按住指示按键使打印机连续走纸。

1.3.3 操作

- 1、自检：打印机断电，按住指示按键，通电，约 2 秒钟，打印机进行自检。
- 2、手动上纸：
 - a. 拉开打印机的开仓扳手。
 - b. 将纸卷拽出约 5CM 长的纸头，纸卷的光滑面面向打印头放入打印机纸仓中，纸头露出。
 - c. 扣紧仓盖，使打印头胶辊压紧打印纸，完成上纸。
- 3、自动上纸：将打印纸通过导纸口插入打印机，打印机将自动吸纸，走纸约 20CM 后，切刀将多余走出的纸切掉，上纸完成。
- 4、走纸：在加电状态下按键，为走纸。

二、通讯接口

2.1 串行接口

2.1.1 接口定义

数据传送：串行

同步方式：异步

握手信号：CTS/RTS，DTR/DSR 或者XON/XOFF

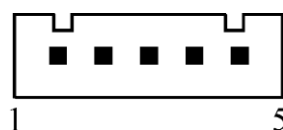
波特率： 9600 bps。

数据长度：8Bit

奇偶校验：None

停止位：1位；

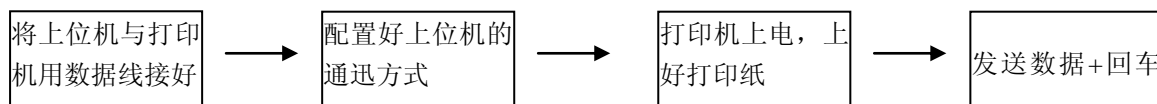
接口：板侧为针型 5 针，间距 2.54mm



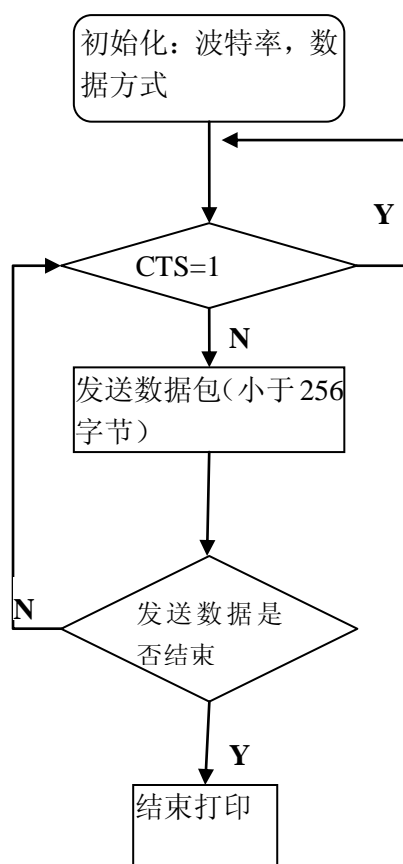
5 芯串口线	信 号	信号来源	方向	说 明
1	-	-	-	空脚
2	TXD	打印机	输出	控制板从主机接收数据。
3	RXD	主机	输入	控制板向主机发送数据。（当使用 X-ON/X-OFF 握手协议时，打印机向计算机发送控制码 X-ON/X-OFF。）
4	CTS	打印机	输出	该信号为“MARK”状态时，表示打印机正“忙”不能接受数据，而当该信号为“SPACE”状态时，表示打印机“准备好”，可以接受数据。
5	GND	——	——	信号地。

2.1.2 串口数据发送方法

由于打印机上配有16K字节的缓存,当一次发送数据少于16K时,可直接发送数,发送方法为:



如发送的数据量很大，则在发送数据时需判断一下 CTS 标志，当此标志为 1 时，不能发送数据，为 0 时，发送数据。数据可以以包的形式发送，也可以以字节形式发送。当以包的形式时，每个数据包不得超过 256 个字节，发送流程图如下图：



三、指令系统

3.1 指令表

- K2C7-S系列热敏打印机采用ESC/POS兼容指令。并增加了汉字打印、字符汉字旋转、字间距调整、条型码打印等功能。

下表中是打印指令的简表：

命令	功能
ESC @	初始化打印机
LF	打印并换行
CR	打印并回车
ESC J	打印并进纸
ESC d	打印并进纸 n 行
ESC c	允许/禁止反向打印
HT	水平制表
ESC !	选择字符打印模式
ESC D	设置水平制表位
ESC -	设置/取消下划线
ESC +	设置/取消上划线
GS B	设置/取消反白打印
FS 2	设置字符旋转打印
ESC \$	设置打印绝对位置
ESC I	设置打印位置
ESC Q	设置右侧不打印字符区域
ESC 1	设置行间距
ESC SP	设置字间距
ESC a	选择对齐方式
FS r	选择上下标
ESC U	水平放大字符
ESC V	垂直放大字符
ESC X	放大字符
ESC K	图形打印命令①
ESC *	图形打印命令②
GS v	打印光栅位图
GS h	设置条形码高度
GS w	设置条形码宽度
GS H	选择可识读字符
GS Q	设置条码水平打印位置
GS k	打印条码
ESC ‘	打印曲线
ESC i	切纸(仅限带切刀机型)

ESC v	向主机传送打印机状态
FS &	进入汉字模式
FS .	取消汉字模式
ESC 6	选择6X8字符集1
ESC 7	选择6X8字符集2
ESC r d n	打印深度调整指令

本章详细描述了控制打印机打印的指令，描述中的格式说明如下：

【COMMAND】 + 【*parameter*】

【COMMAND】 是命令部分，由转义字符和命令字符组成，有少量的单字节命令没有转义字符。

【*parameter*】 是参数部分，用斜体表示，参数并不是数字字符，而是字符的值。

本章所有例子都以 C 语言编写,其中 `SendDataToPrinter`函数为虚拟函数,需要开发者根据主机实际情况编写，该函数定义如下：

`SendDataToPrinter(unsigned char *buffer, unsigned int len)`

说明：向打印机发送数据。

Unsigned char *buf: 打印数据的指针；

Unsigned int len: 数据长度，单位：字节。

3.2 基本控制指令

ESC @

[名称] 初始化打印机

[格式] ASCII ESC @
 十进制 27 64
 十六进制 1B 40

[说明] 清除打印缓冲区中的数据，复位打印参数到当前打印机缺省参数。

[注意] • 打印机接收缓冲区的数据并不被清除。

[例子] unsigned char str[2];
 str[0] = 0x1B;
 str[1] = 0x40;
 SendDataToPrinter(str,2);

LF

[名称] 打印并换行

[格式] ASCII LF
 十进制 10
 十六进制 0A

[说明] 把打印缓冲区中的数据打印出来，并换行。

[注意] • 该命令把打印位置设置为行的开始位置。

[例子] unsigned char str[2];
 str[0] = 0x0A;//或str[0] = '\n'
 SendDataToPrinter(str,1);

CR

[名称] 打印并回车

[格式] ASCII CR
 十进制 13
 十六进制 0D

[说明] 把打印缓冲区中的数据打印出来，并回车。

[参考] LF

[例子] unsigned char str[2];
 str[0] = 0x0D;//或str[0] = '\r'
 SendDataToPrinter(str,1);

ESC J

[名称] 打印并进纸

[格式] ASCII ESC J *n*
 十进制 27 74 *n*
 十六进制 1B 4A *n*

[范围] $0 \leq n \leq 255$

[说明] 打印输出打印缓冲区中的数据，并进纸[$n \times 0.125\text{mm}(0.0049")$]。

[注意] • 打印结束后，将下一行的开始设定为打印起始位置。

[例子] unsigned char str[3];
 str[0] = 0x1B;
 str[1] = 0x4A;

```
str[2] = 0x4;  
SendDataToPrinter(str,3);//向前走纸0.5mm。
```

ESC d

- [名称] 打印并进纸 n 行
- [格式]
- | | | | |
|-------|-----|-----|-----|
| ASCII | ESC | d | n |
| 十进制 | 27 | 100 | n |
| 十六进制 | 1B | 64 | n |
- [范围] $0 \leq n \leq 255$
- [说明] 打印输出打印缓冲区中的数据，并进纸 n 行。
- [注意]
- 一行的距离为24个垂直点距(0.125mm)。
 - 打印结束后，该命令设置打印起始位置为行起点。
- [例子]
- ```
unsigned char str[3];
str[0] = 0x1B;
str[1] = 0x64;
str[2] = 0x4;
SendDataToPrinter(str,3);//向前走纸4行。
```

### ESC c

- [名称] 允许/禁止反向打印
- [格式]
- |       |     |    |     |
|-------|-----|----|-----|
| ASCII | ESC | c  | $n$ |
| 十进制   | 27  | 99 | $n$ |
| 十六进制  | 1B  | 63 | $n$ |
- [范围]  $0 \leq n \leq 1$
- [描述] 当 $n=1$  时，允许反向打印打印方向由左向右，当 $n=0$  时，禁止反向打印，打印方向由右向左。
- [注意]
- 通常在打印机垂直安装时，会使用反向打印方式。反向打印不但支持字符方式，也支持图形方式。在反向打印图形时，请注意图形单元的打印顺序，参考 ESC K。
- [例子]
- ```
unsigned char str[3];  
str[0] = 0x1B;  
str[1] = 0x63;  
str[2] = 0x1;  
SendDataToPrinter(str,3);//反向打印。
```

HT

- [名称] 水平制表
- [格式]
- | | |
|-------|----|
| ASCII | HT |
| 十进制 | 9 |
| 十六进制 | 09 |
- [说明] 移动打印位置到下一个水平制表位置。
- [注意]
- 通过ESC D命令设置水平制表位的位置。
 - 如果没有设置下一个水平制表位置，则该命令被忽略。
- [参考] ESC D。

ESC !

- [名称] 选择字符打印模式
- [格式]
- | | | | |
|-------|-----|---|-----|
| ASCII | ESC | ! | n |
|-------|-----|---|-----|

十进制	27	33	n
十六进制	1B	21	n

[范围] $0 \leq n \leq 255$

位	值	意义
0	0	西文字符（半宽）字体A (12x24)
	1	西文字符（半宽）字体B (8x16)
1	-	-
2	-	-
3	-	-
4	0	取消倍高模式
	1	设置倍高模式
5	0	取消倍宽模式
	1	设置倍宽模式
6	—	—
7	0	取消下划线模式
	1	设置下划线模式

[例子] unsigned char str[3];

str[0] = 0x1B;

str[1] = 0x21;

str[2] = 0x31;

SendDataToPrinter(str,3);//在倍宽倍高模式下打印8X16字符。

ESC D $n_1 n_2 \dots n_k$ NULL

[名称] 设置水平制表位

[格式] ASCII码 ESC D $n_1 \dots n_k$ NULL

十进制码 27 68 $n_1 \dots n_k$ 0

十六进制码 1B 44 $n_1 \dots n_k$ 00

[范围] $1 \leq n \leq 255$ $0 \leq k \leq 20$

[描述] 设置水平定位位置。

n 指定从一行开始的列号，用来设置水平定位位置。

k 表示将被设置水平定位点的总数。

[注意] • 水平制表位置作为一个值储存，这个值为 n 个西文字符宽度，是从行的开始测量的。字符宽度包括字符间距的缺省字符宽。

• 该命令不受字符放大命令(ESC X)的影响。

• 该命令删除了之前设定的水平定位位置。

• 字符打印位置超过定位位置将被处理为普通数据。

• 按升序传输 $[n]k$ ，并且在末尾放置一个NULL码0。

• 该命令中 $n_k > n(k-1)$ ，如果 n_k 小于或等于前面的值 $n(k-1)$ ，定位设定结束并且 $n(k-1)$ 后面的数据按普通数据处理。

• ESC D NULL 取消所有水平定位位置。

• 即使字符宽度变化，以前指定的水平定位位置也不变。

[缺省值] 缺省定位位置为字体A (12 × 24)。

[例子]

```
unsigned str[8];
unsigned char Order = 9;
str[0] = 0x1B;
str[1] = 0x44;
str[2] = 2;//距第1列1个字符间距
str[3] = 9;//距第1列7个字符间距
str[4] = 14;//距第1列15个字符间距
str[5] = 0; //结束
SendDataToPrinter (str,6)
SendDataToPrinter (&Order,1);
SendDataToPrinter ("HT1",3);
SendDataToPrinter (&Order,1);
SendDataToPrinter ("HT2",3);
SendDataToPrinter (&Order,1);
SendDataToPrinter ("HT3",3);
Order = 0x0D;
SendDataToPrinter (&Order,1);
SendDataToPrinter ("1234567890123456\r",17)
```

```
HT1    HT2  HT3
1234567890123456
```

ESC - n

[名称] 取消/设置下划线

[格式]

ASCII	ESC-	<i>n</i>
十进制	27 45	<i>n</i>
十六进制	1B 2D	<i>n</i>

[说明] *n* = 1, 允许下划线打印; *n* = 0, 取消下划线打印。

[注意]

- 反白及旋转字符此命令无效。
- 该命令仅对英文和汉字字符有效。

[缺省值] *n* = 0

[例子]

```
unsigned char str[3];
str[0] = 0x1B;
str[1] = 0x2D;
str[2] = 0x1;
SendDataToPrinter (str,3);//设置下划线
```

ESC + n

[名称] 取消/设置上划线

[格式]

ASCII	ESC+	<i>n</i>
十进制	27 43	<i>n</i>
十六进制	1B 2B	<i>n</i>

[说明] *n* = 1, 允许上划线打印; *n* = 0, 取消上划线打印。

[注意]

- 反白及旋转字符此命令无效。
- 该命令仅对英文和汉字字符有效。

[缺省值] $n = 0$

[例子] `unsigned char str[3];`
`str[0] = 0x1B;`
`str[1] = 0x2B;`
`str[2] = 0x1;`
`SendDataToPrinter (str,3);`//设置上划线

GS B n

[名称] 设置/取消反白打印

[格式] ASCII GS B n
 十进制 29 66 n
 十六进制 1D 42 n

[描述] 设置或取消反白打印。

当 n 的最低有效位为0时，取消反白模式。

当 n 的最低有效位为1时，设置反白模式。

[注意] • 仅 n 的最低位有效。
 • 该命令对内置字符和用户自定义字符均有效。
 • 该命令仅对英文和汉字字符有效。

[缺省值] $n = 0$

[例子] `unsigned char str[3];`
`str[0] = 0x1D;`
`str[1] = 0x42;`
`str[2] = 1;`//设置反白模式
`SendDataToPrinter(str, 3);`

FS 2 n

[名称] 设置字符旋转打印

[格式] ASCII FS 2 n
 十进制 28 73 n
 十六进制 1C 49 n

[范围] $0 \leq n \leq 3$

[描述] 设置字符旋转模式

N (十进制)	意义
0	不旋转
1	设置90° 逆时针旋转
2	设置180° 逆时针旋转
3	设置270° 逆时针旋转

[注意] • 在90° 或270° 旋转模式下，字符放大命令的宽高放大方向与一般模式下的放大方向相反。

[缺省值] $n = 0$

[例子] `unsigned char str[3];`
`str[0] = 0x1C;`
`str[1] = 0x49;`
`str[2] = 1;`//设置90° 旋转
`SendDataToPrinter(str, 3);`

ESC \$ nL nH

- [名称] 设置打印绝对位置
- [格式] ASCII ESC\$ *nL nH*
 十进制 27 36 *nL nH*
 十六进制 1B 24 *nL nH*
- [范围] $0 \leq nL + (nH \times 256) < 384$
- [描述] 设定从一行的开始到将要打印字符的位置之间的距离。
 从一行的开始到打印位置的距离为N个水平点距。
 *nL nH*是双字节无符号整数N的低位和高位, $N = nL + nH \times 256$
- [注意] • 如果设定的打印位置超出了可打印区域 ($N > 372$) , 命令将被忽略。
- [例子]

```
unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x24;
str[2] = 32;//
SendDataToPrinter (str, 3); //绝对位置设为距左边界32水平点距。
```

ESC I n

- [名称] 设置左侧不打印区域
- [格式] ASCII ESC I *n*
 十进制 27 108 *n*
 十六进制 1B 6C *n*
- [范围] $0 \leq n \leq 32$
- [描述] 设定左侧不打印的字符数。
 从一行的开始到打印位置的距离为*n*个西文字符宽度。
- [注意] • 如果设定的打印位置超出了可打印区域, 命令将被忽略。
 • 字符宽度包括字符间距的缺省字符宽。
- [例子]

```
unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x6C;
str[2] = 3;//
SendDataToPrinter (str, 3); //左侧位置设为距左边界3个西文字符宽度。
```

ESC Q n

- [名称] 设置右侧不打印区域
- [格式] ASCII ESC Q *n*
 十进制 27 81 *n*
 十六进制 1B 51 *n*
- [范围] $0 \leq n \leq 32$
- [描述] 设定右侧不打印的字符数。
- [注意] • 如果设定的打印位置超出了可打印区域, 命令将被忽略。
 • 字符宽度包括字符间距的缺省字符宽。
- [例子]

```
unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x51;
str[2] = 3;//
SendDataToPrinter (str, 3); //右侧设置3个西文字符宽区域不打印。
```

ESC 1 n

[名称] 设置行间距

[格式] ASCII ESC1 *n*
 十进制 27 49 *n*
 十六进制 1B 31 *n*

[范围] $0 \leq n \leq 255$

[描述] 设置字符行间距为*n*个垂直点距。

[缺省值] *n* = 3

[例子] unsigned char str[4];
 str[0] = 0x1B;
 str[1] = 0x31;
 str[2] = 8;
 SendDataToPrinter(str,3);//设置行间距为8个垂直点距。

ESC SP n

[名称] 设置字间距

[格式] ASCII ESCSP *n*
 十进制 27 32 *n*
 十六进制 1B 20 *n*

[范围] $0 \leq n \leq 255$

[描述] 设置字符字间距为*n*个水平点距。

[缺省值] *n* = 0

[例子] unsigned char str[4];
 str[0] = 0x1B;
 str[1] = 0x20;
 str[2] = 8;
 SendDataToPrinter(str,3);//设置字间距为8个水平点距。

ESC a n

[名称] 选择对齐方式

[格式] ASCII ESCa *n*
 十进制 27 97 *n*
 十六进制 1B 61 *n*

[范围] $0 \leq n \leq 2$

[描述] 将一行数据按照*n*指定的位置对齐。

<i>N</i>	意义
0	左对齐
1	居中
2	右对齐

[注意] •仅在一行的开始处理时，该命令才有效。

[缺省值] *n* = 0

[例子] unsigned char str[4];
 str[0] = 0x1B;
 str[1] = 0x61;


```
str[2] = 1;  
SendDataToPrinter(str,3);//设置为居中方式打印
```

FS r n

- [名称] 选择上下标
- [格式]
- | | | | |
|-------|----|-----|---|
| ASCII | FS | r | n |
| 十进制 | 28 | 114 | n |
| 十六进制 | 1C | 72 | n |
- [范围] $0 \leq n \leq 1$
- [描述] $n=0$ 选择上标，一行字符图形顶部对齐。
 $n=1$ 选择下标，一行字符图形低部对齐
- [注意]
- 该命令对所有字符(英数字符和汉字) 有效。
 - 如果n 在定义范围之外，忽略该命令。
- [例子]
- ```
unsigned char str[3];
str[0] = 0x1C;
str[1] = 0x72;
str[2] = 0;
SendDataToPrinter(str,3);//.
```

### ESC U n

- [名称] 水平放大字符
- [格式]
- |       |      |      |
|-------|------|------|
| ASCII | ESCU | n    |
| 十进制   | 27   | 85 n |
| 十六进制  | 1B   | 55 n |
- [范围]  $1 \leq n \leq 8$
- [注意]
- 该命令对所有字符(英数字符和汉字) 有效。
  - 如果n 在定义范围之外，忽略该命令。
- [参考] ESC X
- [例子]
- ```
unsigned char str[4];  
str[0] = 0x1B;  
str[1] = 0x55;  
str[2] = 2;  
SendDataToPrinter(str,3);//设置水平放大2倍.
```

ESC V n

- [名称] 垂直放大字符
- [格式]
- | | | |
|-------|------|------|
| ASCII | ESCV | n |
| 十进制 | 27 | 86 n |
| 十六进制 | 1B | 56 n |
- [范围] $1 \leq n \leq 8$
- [注意]
- 该命令对所有字符(英数字符和汉字) 有效。
 - 如果n 在定义范围之外，忽略该命令。
- [参考] ESC X
- [例子]
- ```
unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x56;
```

```
str[2] = 2;
SendDataToPrinter(str,3);//设置垂直放大2倍.
```

## ESC X

- [名称] 放大字符
- [格式]
- |       |       |      |      |
|-------|-------|------|------|
| ASCII | ESC X | $n1$ | $n2$ |
| 十进制   | 27 88 | $n1$ | $n2$ |
| 十六进制  | 1B 58 | $n1$ | $n2$ |
- [范围]  $1 \leq n \leq 8$  ( $1 \leq n1$  水平倍数  $\leq 8$ ,  $1 \leq n2$  垂直倍数  $\leq 8$ )
- [注意]
- 该命令对除条码识读字符外的所有字符(英数字符和汉字) 有效。
  - 如果  $n$  在定义范围之外, 忽略该命令。
  - 垂直方向是指进纸方向, 水平方向与进纸方向垂直。然而, 当字符方向顺时针旋转  $90^\circ$  后, 垂直方向与水平方向之间的关系颠倒, 也就是说本命令优先级低于 **FS 2**, 当两个命令同时有效时, 字符显示是先旋转, 再放大。
- [例子]
- ```
unsigned char str[4];  
str[0] = 0x1B;  
str[1] = 0x58;  
str[2] = 2;  
str[3] = 2;  
SendDataToPrinter(str,4);//设置横向纵向放大2.
```

ESC i

- [名称] 全切纸
- [格式]
- | | |
|-------|--------|
| ASCII | ESC i |
| 十进制 | 27 105 |
| 十六进制 | 1B 69 |
- [描述]: 切刀全切纸。
- [注意]
- 该命令不产生换行。
 - 该命令只用于带切刀机型。
- [例子]
- ```
unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x69;
SendDataToPrinter(str,2);//发送全切纸命令.
```

## ESC m

- [名称] 半全切纸
- [格式]
- |       |        |
|-------|--------|
| ASCII | ESC i  |
| 十进制   | 27 109 |
| 十六进制  | 1B 6D  |
- [描述]: 切刀半全切纸。
- [注意]
- 该命令不产生换行。
  - 该命令只用于带切刀机型。
- [例子]
- ```
unsigned char str[4];  
str[0] = 0x1B;  
str[1] = 0x6D;  
SendDataToPrinter(str,2);//发送半切纸命令.
```

ESC K nL nH d1 d2dk

[名称] 图形打印命令①

[格式] ASCII ESC K nL nH d1...dk
十进制 27 75 nL nH d1...dk
十六进制 1B 4B nL nH d1...dk

[范围]: $0 \leq nL \leq 255$
 $0 \leq nH \leq 1$
 $0 \leq d \leq 255$

[描述]: 本命令只能打印高度为 8 点, 宽度不超过可打印区域的黑白位图。
nL nH 分别为无符号型双字节整数 N 的低位和高位字节, 表示水平方向上位图中的点数。

[参考] ESC *

[注意] •该图形命令受字符放大命令影响。
•当采用反向打印方式时, 要按图形从下到上的顺序依次打印每个图形单元。

[例子] unsigned char str[30];
unsigned char i=0;
str[i++] = 0x1B;
str[i++] = 0x4B;
str[i++] = 15; //打印15个点宽图形
str[i++] = 0x7C; str[i++] = 0x44; str[i++] = 0x44; str[i++] = 0xFF;
str[i++] = 0x44; str[i++] = 0x44; str[i++] = 0x7C; str[i++] = 0x00;
str[i++] = 0x41; str[i++] = 0x62; str[i++] = 0x54; str[i++] = 0xC8;
str[i++] = 0x54; str[i++] = 0x62; str[i++] = 0x41; str[i++] = 0x0D;
SendDataToPrinter(str,i);//发送图形打印命令。

ESC * m nL nH d1...dk

[名称] 图形打印命令②

[格式] ASCII ESC * m nL nH d1...dk
十进制 27 42 m nL nH d1...dk
十六进制 1B 2A m nL nH d1...dk

[范围]: $m = 0, 1, 32, 33$
 $0 \leq nL \leq 255$
 $0 \leq nH \leq 1$
 $0 \leq d \leq 255$

[描述]: 本命令只能打印高度为 8 点或 24 点, 宽度不超过可打印区域的黑白位图。
各参数含义如下:

用 m 选择位图的模式, 位图的水平方向点数由 nL 和 nH 指定, 如下所示:

m	垂直点数 (高度)	倍宽模式
0	8	两倍宽
1	8	单倍宽
32	24	两倍宽
33	24	单倍宽

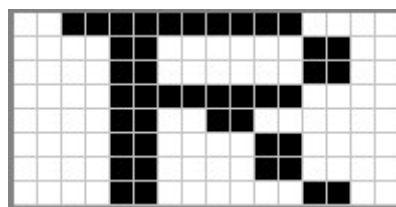
nL nH 分别为无符号型双字节整数 N 的低位和高位字节, 表示水平方向上位图中的点数。N 在单倍宽时最大值为 384, 在双倍宽时其值最大为 192。

d1.....dk 表示位图数据: 具体格式见下图:

[例子]

例 1: $m=0$ (8 点、两倍宽)d1 表示打印的第 1、2 列点的数据, dk 表示打印的第 2k-1 和 2k 列点的数据,bn 表示字节的第 n 位

d1	d2	d3	d4	d5	d6	d7	d8	
0	1	1	1	1	1	0	0	b7
0	0	1	0	0	0	1	0	b6
0	0	1	0	0	0	1	0	b5
0	0	1	1	1	1	0	0	b4
0	0	1	0	1	0	0	0	b3
0	0	1	0	0	1	0	0	b2
0	0	1	0	0	1	0	0	b1
0	0	1	0	0	0	1	0	b0

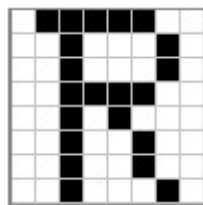


程序代码如下:

```
unsigned char str[100];
j=0;
str[j++] = 0x1B;
str[j++] = 0x2A;
str[j++] = 0; //m=0(高度 8 点、倍宽)
str[j++] = 8; //图象宽度为 8dots
str[j++] = 0;
//位图数据
str[j++] = 0x00;str[j++] = 0x80;str[j++] = 0xFF;str[j++] = 0x90;str[j++] = 0x98;
str[j++] = 0x96;str[j++] = 0x61;str[j++] = 0x00;str[j++] = 0x0D;//打印出图形
SendDataToPrinter(str,j);
```

例 2: $m=1$ (8 点、单倍宽)d1 表示打印的第 1 列点的数据, dk 表示打印的第 k 列点的数据, bn 表示字节的第 n 位

d1	d2	d3	d4	d5	d6	d7	d8	
0	1	1	1	1	1	0	0	b7
0	0	1	0	0	0	1	0	b6
0	0	1	0	0	0	1	0	b5
0	0	1	1	1	1	0	0	b4
0	0	1	0	1	0	0	0	b3
0	0	1	0	0	1	0	0	b2
0	0	1	0	0	1	0	0	b1
0	0	1	0	0	0	1	0	b0

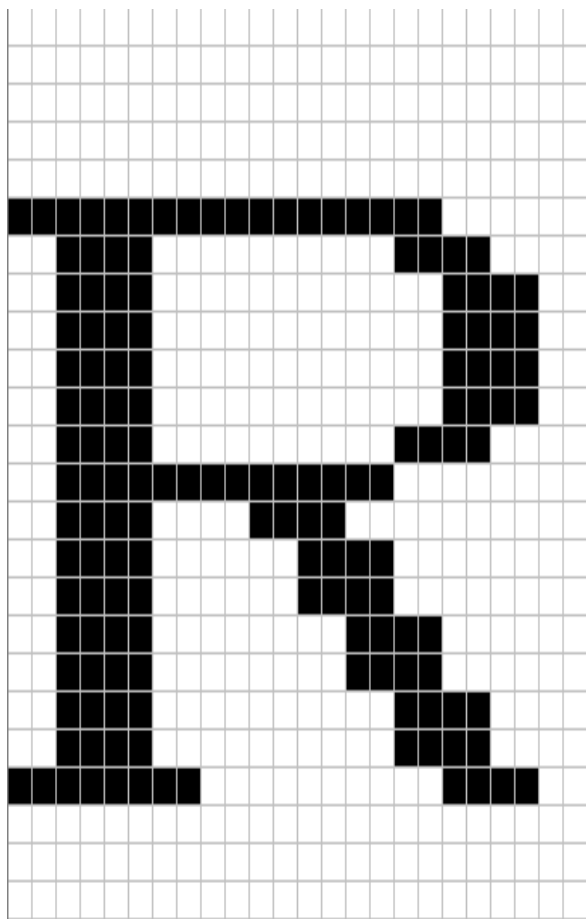


程序代码如下:

```
unsigned char str[100];
j=0;
str[j++] = 0x1B;
str[j++] = 0x2A;
str[j++] = 1; //m=1(高度 8 点、不放大)
str[j++] = 8; //图象宽度为 8dots
str[j++] = 0;
//位图数据
str[j++] = 0x00;str[j++] = 0x80;str[j++] = 0xFF;str[j++] = 0x90;str[j++] = 0x98;
str[j++] = 0x96;str[j++] = 0x61;str[j++] = 0x00;str[j++] = 0x0D; ;//打印出图形
SendDataToPrinter(str,j);
```

例 3: m=32(24 点、两倍宽)d1、d2、d3 表示打印的第 1、2、3 列点的数据, 依此类推; bn 表示字节的第 n 位

	d4	d7									D	d49
d1	0	0	0	0	0	0	0	0	0	0	0	b7
	0	0	0	0	0	0	0	0	0	0	0	b6
	0	0	0	0	0	0	0	0	0	0	0	b5
	1	1	1	1	1	1	1	1	0	0	0	b4
	0	1	1	0	0	0	0	0	1	1	0	b3
	0	1	1	0	0	0	0	0	0	1	1	b2
	0	1	1	0	0	0	0	0	0	1	1	b1
d2	0	1	1	0	0	0	0	0	0	1	1	b0
	0	1	1	0	0	0	0	0	0	1	1	b7
	0	1	1	0	0	0	0	0	1	1	0	b6
	0	1	1	1	1	1	1	0	0	0	0	b5
	0	1	1	0	0	1	1	0	0	0	0	b4
	0	1	1	0	0	0	1	1	0	0	0	b3
	0	1	1	0	0	0	1	1	0	0	0	b2
d3	0	1	1	0	0	0	0	1	1	0	0	b1
	0	1	1	0	0	0	0	1	1	0	0	b0
	0	1	1	0	0	0	0	0	1	1	0	b7
	0	1	1	0	0	0	0	0	1	1	0	b6
	1	1	1	1	0	0	0	0	0	1	1	b5
	0	0	0	0	0	0	0	0	0	0	0	b4
	0	0	0	0	0	0	0	0	0	0	0	b3
	0	0	0	0	0	0	0	0	0	0	0	b2
	0	0	0	0	0	0	0	0	0	0	0	b1
	0	0	0	0	0	0	0	0	0	0	0	b0



程序代码如下:

```
unsigned char str[200];
j=0;
```

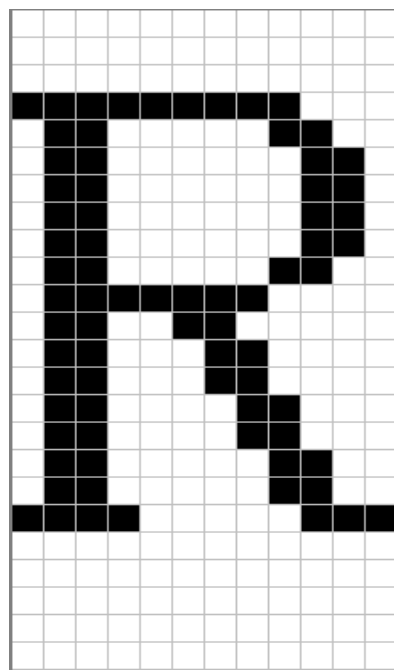
```

str[j++] = 0x1B;
str[j++] = 0x2A;
str[j++] = 32; //m=32(高度 24 点、倍宽)
str[j++] = 12; //图象宽度为 12dots
str[j++] = 0;
//位图数据
str[j++] = 0x10;str[j++] = 0x00;str[j++] = 0x20;str[j++] = 0x1F;str[j++] = 0xFF;str[j++] = 0xE0;
str[j++] = 0x1F;str[j++] = 0xFF;str[j++] = 0xE0;str[j++] = 0x10;str[j++] = 0x20;str[j++] = 0x20;
str[j++] = 0x10;str[j++] = 0x20;str[j++] = 0x00;str[j++] = 0x10;str[j++] = 0x30;str[j++] = 0x00;
str[j++] = 0x10;str[j++] = 0x3C;str[j++] = 0x00;str[j++] = 0x10;str[j++] = 0x2f;str[j++] = 0x00;
str[j++] = 0x18;str[j++] = 0x43;str[j++] = 0xC0;str[j++] = 0x0F;str[j++] = 0xC0;str[j++] = 0xE0;
str[j++] = 0x07;str[j++] = 0x80;str[j++] = 0x20;str[j++] = 0x00;str[j++] = 0x00;str[j++] = 0x20;
str[j++] = 0x0D; //打印出当前的图形
SendDataToPrinter(str,j);

```

例 4: m=33(24 点、无放大)d1、d2、d3 表示打印的第 1、2、3 列点的数据, 依此类推; bn 表示字节的第 n 位

	d4	d7									D	d49
d1	0	0	0	0	0	0	0	0	0	0	0	b7
	0	0	0	0	0	0	0	0	0	0	0	b6
	0	0	0	0	0	0	0	0	0	0	0	b5
	1	1	1	1	1	1	1	1	1	0	0	b4
	0	1	1	0	0	0	0	0	1	1	0	b3
	0	1	1	0	0	0	0	0	0	1	1	b2
	0	1	1	0	0	0	0	0	0	1	1	b1
	0	1	1	0	0	0	0	0	0	1	1	b0
d2	0	1	1	0	0	0	0	0	0	1	1	b7
	0	1	1	0	0	0	0	0	1	1	0	b6
	0	1	1	1	1	1	1	1	0	0	0	b5
	0	1	1	0	0	1	1	0	0	0	0	b4
	0	1	1	0	0	0	1	1	0	0	0	b3
	0	1	1	0	0	0	1	1	0	0	0	b2
	0	1	1	0	0	0	0	1	1	0	0	b1
	0	1	1	0	0	0	0	1	1	0	0	b0
d3	0	1	1	0	0	0	0	0	1	1	0	b7
	0	1	1	0	0	0	0	0	1	1	0	b6
	1	1	1	1	0	0	0	0	0	1	1	b5
	0	0	0	0	0	0	0	0	0	0	0	b4
	0	0	0	0	0	0	0	0	0	0	0	b3
	0	0	0	0	0	0	0	0	0	0	0	b2
	0	0	0	0	0	0	0	0	0	0	0	b1
	0	0	0	0	0	0	0	0	0	0	0	b0



程序代码如下:

```
unsigned char str[200];
j=0;
str[j++] = 0x1B;
str[j++] = 0x2A;
str[j++] = 32; //m=33(高度 24 点、无放大)
str[j++] = 12; //图象宽度为 12dots
str[j++] = 0;
//位图数据
str[j++] = 0x10;str[j++] = 0x00;str[j++] = 0x20;str[j++] = 0x1F;str[j++] = 0xFF;str[j++] = 0xE0;
str[j++] = 0x1F;str[j++] = 0xFF;str[j++] = 0xE0;str[j++] = 0x10;str[j++] = 0x20;str[j++] = 0x20;
str[j++] = 0x10;str[j++] = 0x20;str[j++] = 0x00;str[j++] = 0x10;str[j++] = 0x30;str[j++] = 0x00;
str[j++] = 0x10;str[j++] = 0x3C;str[j++] = 0x00;str[j++] = 0x10;str[j++] = 0x2f;str[j++] = 0x00;
str[j++] = 0x18;str[j++] = 0x43;str[j++] = 0xC0;str[j++] = 0x0F;str[j++] = 0xC0;str[j++] = 0xE0;
str[j++] = 0x07;str[j++] = 0x80;str[j++] = 0x20;str[j++] = 0x00;str[j++] = 0x00;str[j++] = 0x20;
str[j++] = 0x0D; //打印出当前的图形
SendDataToPrinter(str,j);
```

GS v 0 m xL xH yL yH d1....dk

[名称]打印光栅位图

[格式] ASCII GS v 0 m xL xH yL yH d1...dk

十进制 29 118 48 m xL xH yL yH d1...dk

十六进制 1D 76 30 m xL xH yL yH d1...dk

[范围] $0 \leq m \leq 3, 48 \leq m \leq 51$

$0 \leq xL \leq 255$

$0 \leq xH \leq 255$ where $1 \leq (xL + xH \times 256) \leq 128$

$0 \leq yL \leq 255$

$0 \leq yH \leq 8$ where $1 \leq (yL + yH \times 256) \leq 4095$

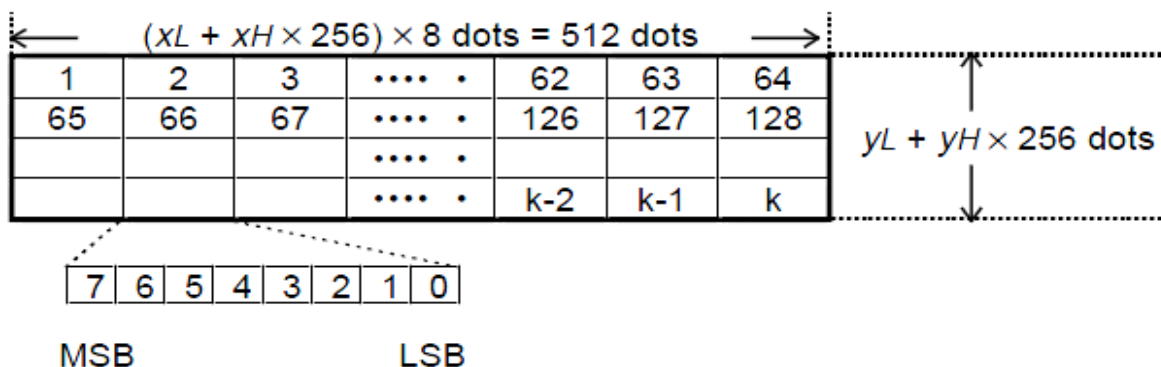
$0 \leq d \leq 255$

$k = (xL + xH \times 256) \times (yL + yH \times 256) (k \neq 0)$

m	模式	垂直点密度	水平点密度
0,48	正常	203.2dpi	203.2dpi
1,49	倍宽	203.2dpi	101.6dpi
2,50	倍高	101.6dpi	203.2dpi
3,51	倍宽倍高	101.6dpi	101.6dpi

□. xL, xH位图水平宽度所占字节数。

□. yL, yH,位图垂直点长度



GS h n

[名称] 设置条形码高度

[格式] ASCII GS h n
十进制 29 104 n
十六进制 1D 68 n

[范围]: $1 \leq n \leq 255$

[描述]: 设置选择条形码高度。
N为垂直方向的点数。

缺省值: $n = 48$

[例子] unsigned char str[4];
str[0] = 0x1D;
str[1] = 0x68;
str[2] = 30;
SendDataToPrinter(str,3);//设置条码高度为30个垂直点距高

GS w n

[名称] 设置条形码宽度

[格式] ASCII GS w n
十进制 29 119 n
十六进制 1D 77 n

[范围]: $1 \leq n \leq 4$

[描述]: 设置条形码水平尺寸。 n 取值定义如下:

N	多级条形码单位宽度 (mm)	二进制条码宽度	
		窄条线宽度	宽条线宽度
1	0.125	0.125	0.25
2	0.25	0.25	0.50
3	0.375	0.375	0.75
4	0.50	0.50	1.0

[例子] unsigned char str[4];
str[0] = 0x1D;
str[1] = 0x77;
str[2] = 3;
SendDataToPrinter(str,3);//设置条码宽度

GS H n

[名称] 选择可识读字符

[格式] ASCII GS h n
 十进制 29 72 n
 十六进制 1D 48 n

[范围]: $0 \leq n \leq 2$

[描述]: 打印条形码时, 选择可识读字符的打印。



n 选择打印, 如下所示:

n	识读字符位置
0	不打印
1	条形码上方
2	条形码下方

[缺省值]: $n = 0$

[例子] unsigned char str[4];

 str[0] = 0x1D;

 str[1] = 0x48;

 str[2] = 2;

 SendDataToPrinter(str,3);//设置可识读字符在条码的下方打印。

GS Q n

[名称] 设置条码水平打印位置

[格式] ASCII GS Q n
 十进制 29 81 n
 十六进制 1D 51 n

[范围]: $0 \leq n \leq 255$

[描述]: 设置条码从一行开始到打印位置的距离为N个水平点距。

[缺省值]: $n = 0$

[例子] unsigned char str[4];

 str[0] = 0x1D;

 str[1] = 0x51;

 str[2] = 32;

 SendDataToPrinter(str,3);//

GS k

[名称]: 打印条码。

[格式]: 该命令有两种格式:

 格式1: ($0 \leq m \leq 8$)

 ASCII码: GS k m d1...dk NUL

 十进制码: 29 107 m d1...dk 0

 十六进制码: 1D 6B m d1...dk 00

 格式2: ($65 \leq m \leq 73$)

 ASCII码: GS k m n d1...dn

 十进制码: 29 107 m n d1...dn

十六进制码: 1D 6B m n d1...dn

[范围]: $0 \leq m \leq 8$ (k 和 d 取决于使用的条码系统)

$65 \leq m \leq 73$ (n 和 d 取决于使用的条码系统)

n 为打印条码的数据长度

[描述]: 选定条码系统并打印条码。

M 定义所使用的条码系统, 如下表:

m		条码类型	长度	范围
格式 1	0	UPC-A	$11 \leq k \leq 12$	$48 \leq d \leq 57$
	1	UPC-E	$K \leq$	$48 \leq d \leq 57$
	2	JAN13 (EAN13)	$12 \leq k \leq 13$	$48 \leq d \leq 57$
	3	JAN 8 (EAN8)	$7 \leq k \leq 8$	$48 \leq d \leq 57$
	4	CODE39	$1 \leq k$	$48 \leq d \leq 57, 65 \leq d \leq 90, 32, 36, 37, 43, 45, 46, 47$
	5	ITF	$1 \leq k$ (even number)	$48 \leq d \leq 57$
	6	CODABAR	$1 \leq k$	$48 \leq d \leq 57, 65 \leq d \leq 68, 36, 43, 45, 46, 47, 58$
格式 2	65	UPC-A	$11 \leq n \leq 12$	$48 \leq d \leq 57$
	66	UPC-E	$n=8$	$48 \leq d \leq 57$
	67	JAN13 (EAN13)	$12 \leq n \leq 13$	$48 \leq d \leq 57$
	68	JAN 8 (EAN8)	$7 \leq n \leq 8$	$48 \leq d \leq 57$
	69	CODE39	$1 \leq n \leq 255$	$48 \leq d \leq 57, 65 \leq d \leq 90, 32, 36, 37, 43, 45, 46, 47$
	70	ITF	$1 \leq n \leq 255$ (even number)	$48 \leq d \leq 57$
	71	CODABAR	$1 \leq n \leq 255$	$48 \leq d \leq 57, 65 \leq d \leq 68, 36, 43, 45, 46, 47, 58$
	72	CODE93	$1 \leq n \leq 255$	$0 \leq d \leq 127$
	73	CODE128	$2 \leq n \leq 255$	$0 \leq d \leq 127$

*除UPC-E外, 其它条码的效验位都有打印机自动算出, 用户可不加效验位。

CODE39无需加

[注意]: • 当使用格式 1 的命令时, 如果条码类型中规定了条码的数据长度, 则 k (打印机接收到的条码数据长度) 应当等于规定的长度, 如果不等于规定的长度, 则该指令无效。有关条码的数据位长度见【附录 B】。

• 打印机接收到的条码数据字符应该包含在条码类型规定的字符集中, 如果条码数据字符中有字符超出了字符集, 该命令无效。有关条码的字符集见【附录 B】。

• 当使用格式 2 的命令时, n 的值要等于条码的规定数据长度 (如果该类型的条码规定了数据位长度的话), 如果 n 值不等于条码的规定数据位长度, 那么该命令无效, 有关条码的数据位长度见【附录 B】。

• INTERLEAVED 25(ITF)条码的数据长度为偶数, 如果使用格式 1 打印 ITF 条码, 那么 k (打印机

接收到的条码数据长度) 的值要为偶数, 如果为奇数, 最后一位数据将被忽略。如果使用格式 2 打印 ITF 条码, 那么 n 值要为偶数, 如果 n 为奇数, 那么最后一位数据将被忽略。

- 如果水平方向尺寸超出了打印区域, 超出的部分将被忽略。
- 该命令不受打印模式(粗体、重叠、下划线、字符大小、或反白打印)影响。
- 打印条码时要遵守条码的编码规范, 否则将会导致条码无法扫描。
- 打印机不计算校验码, 如果条码需要校验码, 需将校验码包含在条码数据当中, 打印机不负责核对校验码的正误, 用户计算校验码错误将导致条码无法扫描。
- CODE39 码不包括扩展 CODE39 码 (EXTERN CODE 39)。
- CODE93 码不包括扩展 CODE93 码 (EXTERN CODE 93)。
- CODE128 条码数据串的头部必须是编码集选择字符(CODE A, CODE B, 或 CODE C), 在一个条码内部也可切换编码集。用字符 '{' 和一个字符组合用以定义特殊功能。通过连续传送两次 '{' 定义 ASCII 字符'{'。如下图

ASCII	HEX	功能
{A	7B, 41	选择编码集A
{B	7B, 42	选择编码集B
{C	7B, 43	选择编码集C
{S	7B, 53	SHIFT
{1	7B, 31	FNC1
{2	7B, 32	FNC2
{3	7B, 33	FNC3
{4	7B, 34	FNC4

GS k m v r d1.....dk

[名称] 打印QR CODE二维条码

[格式] 该命令有两种格式:

格式1 m=32

ASCII码 GS k m v r d1...dk NUL

十进制码 29 107 m v r d1...dk 0

十六进制码 1D 6B m v r d1...dk 00

格式2 m=97

ASCII码 GS k m v r nL nH d1...dn

十进制码 29 107 m v r nL nH d1...dn

十六进制码 1D 6B m v r nL nH d1...dn

[范围] m=32或97

$1 \leq v \leq 20$ $1 \leq r \leq 4$

[描述] v为DQCODE版本号

r=1 纠错等级为L r=2 纠错等级为M r=3 纠错等级为Q r=4 纠错等级为H

nL, nH 为整数 N 的低位和高位, N 为打印条码的数据长度, 单位为字节

当使用第一种格式时, 指令以 00 结尾, $d1...dk$ 为条码数据。当使用第二种格式时, 打印机将把 nH 后的 N 个字符 ($d1...dn$) 作为条码数据。

[注意] • 因打印纸张宽度有限, QR CODE 的版本号最大为 20.

- 有关 QR CODE 的编码标准的详细信息, 请参见中国国家标准 GB/T 18284-2000 或 ISO 标准 ISO/IEC 18004:2000

[例子]

```
unsigned char str[16];

str[0] = 0x1D; str[1] = 0x6B; str[2] = 32;

str[3] = 1; // 版本为 1

str[4] = 2; // 纠错等级为 M

str[5] = '1'; str[6] = '2'; str[7] = '3'; str[8] = '4'; str[9] = '5'; str[10] = '6'; str[11] = '7';

str[12] = '8'; str[13] = '9'; str[14] = '0';

SendDataToPrinter ( str, 5);
```

GS W n

[名称] 设置二维条形倍数

[格式] ASCII GS w n
 十进制 29 87 n
 十六进制 1D 57 n

[范围]: $1 \leq n \leq 8$

[例子] unsigned char str[4];
 str[0] = 0x1D;
 str[1] = 0x57;
 str[2] = 3;
 SendDataToPrinter(str,3); // 设置二维条码放大 3 倍

ESC ‘

[名称] 打印一水平行上 n 个点

[格式] ASCII 码 GS ‘ nL nH $x1L$ $x1H$ $x21L$ $x21H$ xkL xkH CR
 十进制码 29 39 nL nH $x1L$ $x1H$ $x21L$ $x21H$ xkL xkH 13
 十六进制码 1B 27 nL nH $x1L$ $x1H$ $x21L$ $x21H$ xkL xkH 0D

[范围] : $0 \leq nL \leq 255$

$0 \leq nH \leq 1$

曲线点数 $N = nH \times 256 + nL$

曲线点在水平行上的位置 $X = xkH \times 256 + xkL$ 。

[描述] : 每条曲线都是由很多点组成。本指令为打印一水平行上 n 个点, 连续使用该指令可以打印出用户所需要的曲线。

[例子]: 曲线关系函数为下面 5 个函数

```

Y1=50+40*abs (-0.01*X) *sin (X/10)
Y2=50-40*abs (-0.01*X) *sin (X/10)
Y3=50
Y4=50+40*abs(-0.1*X)
Y5=50-40*abs(-0.01*X)

```

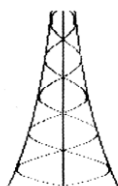
[打印例程]:

```

unsigned char str[50];
float X;
unsigned int  m_cur1,m_cur2,i;
for(X=0;X<150;X++)                                //打印150点行
{
    m_cur1= 40*exp(-0.01*X);
    YY= Y*sin(X/10);
    str[i++] = 0x1b;
    str[i++] = 0x27;
    str[i++] = 0x5; //打印5条曲线
    str[i++] = 0x0;
    str[i++] = 50+m_cur2;
    str[i++] = 0;
    str[i++] = 50-m_cur2;
    str[i++] = 0;
    str[i++] = 50;
    str[i++] = 0;
    str[i++] = 50+m_cur1;
    str[i++] = 0;
    str[i++] = 50-m_cur1;
    str[i++] = 0;
    str[i++] = 0x0D;
    SendDataToPrinter(str,i); //
}

```

[结果]:



ESC v

[名称] 向主机传送打印机状态

[格式] ASCII ESCv
 十进制 27 118
 十六进制 1B 76

[描述]: 向主机传送打印机状态。

[注意]: • 仅串口型打印机有效。

位	功能	值
---	----	---

		0	1
0	纸检测器	无纸	有纸
1	工作状态	空闲	打印中
2	接收缓冲区	未满	满
3	取纸传感器	无纸	有纸
4	未定义	---	---
5	未定义	---	---
6	未定义	---	---
7	未定义	---	---

[例子] unsigned char str[4];
str[0] = 0x1B;
str[1] = 0x76;
SendDataToPrinter(str,2);//向打印机发送状态查询命令.

FS &

[名称] 进入汉字模式

[格式] ASCII FS &
十进制 28 38
十六进制 1C 26

[描述]: 打印机进入汉字打印模式。

[注意]: • 上电后打印机默认为汉字打印模式。

[例子] unsigned char str[4];
str[0] = 0x1C;
str[1] = 0x26;
SendDataToPrinter(str,2);//进入汉字打印模式。

FS.

[名称] 取消汉字模式

[格式] ASCII FS .
十进制 28 46
十六进制 1C 2E

[描述]: 取消汉字字符模式，当取消汉字字符模式后，超过0x80的编码仍然当作12X24 ASCII字符处理，将不再打印汉字，除非再用FS &命令选择汉字模式。打印机进入汉字打印模式。

[例子] unsigned char str[4];
str[0] = 0x1C;
str[1] = 0x2E;
SendDataToPrinter(str,2);//进入ASCII字符打印模式。

ESC 6

[名称] 选择6X8字符集1

[格式] ASCII ESC 6
十进制 27 54
十六进制 1B 36

[描述]: 在该命令输入之后的所有字符均使用字符集1中的字符打印(见附录D)字符集1中有6x8点阵字符224个，包括ASCII字符及各种图形符号等。代码范围20H~FFH(32~255)。

[例子] unsigned char str[4];
str[0] = 0x1B;

```
str[1] = 0x36;  
SendDataToPrinter(str,2);//打印6X8字符集1中的字符。
```

ESC 7

- [名称] 选择6X8字符集2
- [格式] ASCII ESC 7
 十进制 27 55
 十六进制 1B 37
- [描述]: 在该命令输入之后的所有字符均使用字符集2中的字符打印（见附录D），字符集2中有6×8点阵字符224个，包括德、法、俄文、日语片假名等。代码范围20H~FFH(32~255)。
- [例子] unsigned char str[4];
 str[0] = 0x1B;
 str[1] = 0x37;
 SendDataToPrinter(str,2);// 打印6X8字符集2中的字符。

ESC r d n

- [名称] 打印深度调整指令
- [格式] ASCII ESC r d n
 十进制 27 114 d n
 十六进制 1B 72 d n
- [范围]: d=2B或2D ; $0 \leq n \leq 255$
- [描述]: d取2B时，表示在当前的打印深度的基础上调深。
 d取2D时，表示在当前打印深度的基础上加调浅。
 n表示为调深或调浅的程度值，值越大，调整的最大，最大值不超过255。
 当n为0时，无论是用1B 72 2B 00还是1B 72 2D 00 都将会复原回默认值。
- [缺省值]: $n = 0$
- [注意]: 当加深调整时，不要将n值调整到最大，加热太深的话不仅影响效果而且还会减少打印寿命。
 随着深度的增加，打印机的功耗将会增大，反之减小。
- [例子] unsigned char str[4];
 str[0] = 0x1B;
 str[1] = 0x72;
 str[2] = 0x2B;
 str[3] = 0x30;
 SendDataToPrinter(str,4);//加深0X30值

ESC R n

- [名称] 选择国际字符集
- [格式] ASCII码 ESC R n

十六进制码 1B 52 n

十进制码 27 82 n
- [范围] 0 . n . 13
- [描述] 按照下表选择n的值设置国际字符集

n	字符集
---	-----

0	U.S.A.
1	France
2	Germany
3	U.K.
4	Denmark I
5	Sweden
6	Italy
7	Spain I
8	Japan
9	Norway
10	Denmark II
11	Spain II
12	Latin
13	Korea
14	Slovenia/Croatia
15	China

[缺省值] n = 0

GS t n

[名称] 选择字符代码表

[格式] ASCII码 GS t n

十六进制码 1D 74 n

十进制码 29 116 n

[范围] 0 . n . 5, 16 . n . 19, n = 255

[描述] 从字符代码表中选择页n。

N	代码页	N	代码页
0	Normal*	32	Codepage 1252 (Windows Latin-1)
1	CodePage437 (USA, Std. Europe)	33	Codepage 1250 (Windows Latin-2)
2	Katakana	34	Codepage 1251 (Windows Cyrillic)
3	CodePage437 (USA, Std. Europe)	64	Codepage 3840 (IBM-Russian)
4	Codepage 858 (Multilingual)	65	Codepage 3841 (Gost)
5	Codepage 852 (Latin-2)	66	Codepage 3843 (Polish)
6	Codepage 860 (Portuguese)	67	Codepage 3844 (CS2)
7	Codepage 861 (Icelandic)	68	Codepage 3845 (Hungarian)
8	Codepage 863 (Canadian French)	69	Codepage 3846 (Turkish)
9	Codepage 865 (Nordic)	70	Codepage 3847 (Brazil-ABNT)
10	Codepage 866 (Cyrillic Russian)	71	Codepage 3848 (Brazil-ABICOMP)
11	Codepage 855 (Cyrillic Bulgarian)	72	Codepage 1001 (Arabic)
12	Codepage 857 (Turkey)	73	Codepage 2001 (Lithuanian-KBL)
13	Codepage 862 (Israel (Hebrew))	74	Codepage 3001 (Estonian-1)
14	Codepage 864 (Arabic)	75	Codepage 3002 (Estonian-2)

15	Codepage 737 (Greek)	76	Codepage 3011 (Latvian-1)
16	Codepage 851 (Greek)	77	Codepage 3012 (Latvian-2)
17	Codepage 869 (Greek)	78	Codepage 3021 (Bulgarian)
18	Codepage 928 (Greek)	79	Codepage 3041 (Maltese)
19	Codepage 772 (Lithuanian)		
20	Codepage 774 (Lithuanian)		
21	Codepage 874 (Thai)		

FS V

[名称] 垂直制表并打印

[格式] ASCII FS V

十进制 28 86 m LP1...LPm n IP1...IPn FT1 D11...D1k 0...FTn Dn1...Dnk 0

十六进制 1C 56 m LP1...LPm n IP1...IPn FT1 D11...D1k 0...FTn Dn1...Dnk 0

[描述] m 表格竖线数: $0 \leq m \leq 17$;
LP1...LPm 表格竖线坐标: $0 \leq LPm \leq 48$;
n 表格项数: $0 \leq n \leq 16$;
IP1...IPn 表项坐标: $0 \leq IPn \leq 45$;
FT1 第一个表项的字体类型:

位	功能	值	
		0	1
0	保留		
1	加粗	取消	设定
2	下划线	取消	设定
3	反白	取消	设定
4-7	保留		

[例子]

1C 56 06 00 09 12 1B 24 2D

0A 01 05 0A 0E 13 17 1C 20 25 29

02 20 B2 E2 C1 BF C8 D5 C6 DA 20 00

01 20 20 20 2D 20 20 2D 20 20 00

02 20 B2 E2 C1 BF CA B1 BC E4 00

01 20 20 20 3A 20 20 3A 20 20 00

02 20 D0 D4 20 20 20 20 B1 F0 00

01 20 20 20 20 20 D0 D4 00

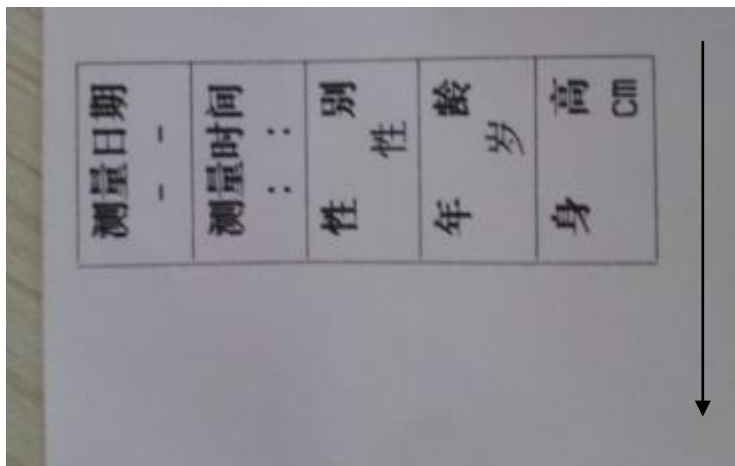
02 20 C4 EA 20 20 20 20 C1 E4 00

01 20 20 20 20 20 CB EA 00

02 20 C9 ED 20 20 20 20 B8 DF 00

01 20 20 20 20 20 20 20 63 6D 00

[结果]:



GS k m v r d1.....dk

[名称] 打印QRCODE二维条码

[格式] 该命令有两种格式:

格式1 m=32

ASCII码 GS k m v r d1...dk NUL

十进制码 29 107 m v r d1...dk 0

十六进制码 1D 6B m v r d1...dk 00

格式2 m=97

ASCII码 GS k m v r nL nH d1...dn

十进制码 29 107 m v r nL nH d1...dn

十六进制码 1D 6B m v r nL nH d1...dn

[范围] m=32或97

$1 \leq v \leq 20$ $1 \leq r \leq 4$

[描述] v为DQCODE版本号

r=1 纠错等级为L r=2 纠错等级为M r=3 纠错等级为Q r=4 纠错等级为H

nL,nH为整数N的低位和高位, N为打印条码的数据长度, 单位为字节

当使用第一种格式时, 指令以00结尾, d1...dk为条码数据。当使用第二种格式时, 打印机将把nH后的N个字符(d1...dn)作为条码数据。

[注意] • 因打印纸张宽度有限, QRCODE的版本号最大为20.

- 有关QRCODE的编码标准的详细信息, 请参见中国国家标准GB/T 18284-2000或ISO标准ISO/IEC 18004:2000

[例子]

unsigned char str[16];

[HTTP://WWW.RD-CN.COM](http://www.rd-cn.com)

```
str[0] = 0x1D; str[1] = 0x6B; str[2] = 32;  
str[3] = 1; //版本为1  
str[4] = 2; //纠错等级为M  
str[5] = '1'; str[6] = '2'; str[7] = '3'; str[8] = '4'; str[9] = '5'; str[10] = '6'; str[11] = '7';  
str[12] = '8'; str[13] = '9'; str[14] = '0';  
SendDataToPrinter ( str, 5);
```

四、安装尺寸

见附录 d

五、打印机维护及故障排除

为了确保打印机能正常工作,特别要注意不要随意拆卸打印机头,不要自行对打印机作改动。对于不使用打印机壳体的用户,更要注意保护机头。

1. 如果打印机长时间不使用,请不要将打印机接通电源。
2. 如发生打印机工作不正常时,请关掉打印机电源。
3. 使用电源必须符合要求,否则对打印头不利,甚至损坏打印头。
4. 更换纸卷时,请注意机头上是否有纸屑灰尘,如有请轻轻除去,热敏纸注意一下正反面,反面无涂层,无法打印出字迹。
5. 打印机在打印或送纸时,不能撕纸;更不能反向拽纸。
6. 保持打印机控制板干净无尘土。
7. 热敏打印机打印不清晰时,可用洁净的棉球沾少许酒精轻轻擦去打印头片加热元件表面脏物。
8. 打印机与主机连接时,应先连接好打印机的数据线,再接通打印机的电源。
9. 热敏打印机选择纸张时要选择质量好一点的纸张,这样不仅可以提高打印效果,同时也可减少对热敏片的磨损。

附录

A 打印字符集

A.1 ASCII码字符集

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	Ç	£	¥	℞	ƒ
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	¬	½	¼	¿	«	»	
B	⋮	⋮	⋮													
C	L	⊥	T		-	+										
D		⊥														
E	α	β	Γ	Π	Σ	σ	μ	γ	δ	θ	Ω	δ	ω	φ	€	Π
F	≡	±	≥	≤	∫	J	÷	≈	°	.	.	√	n	2		

A.2 字符集1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	Ø	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	↑	€
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8	0	-	=	≡	□	Ⅱ	⊥	∥	∩	†	π	⊗	⊙	⊕	⊖	¥
9	£	§	↓	→	^	±	÷	∞	≈	...	0	0	2	3	2	3
A	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	π	ρ
B	τ	φ	ψ	ω	Γ	Δ	Π	Σ	Ψ	Ω	Σ	Θ	∂	Φ	Ψ	∠
C	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
D	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
E	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
F	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□

A.3 字符集2

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	百	千	万	Ⅱ	℃	¥	°	4	4	½	¼	¼	°	×	√	⊥
3	#		U	∩	⊕	∩	∩	€	⊕	¥	∩	∩	∩	∩	∩	∩
4	∩	≡	≡	∩	≠	∩	∩	∩	∩	∩	∩	∩	∩	∩	∩	∩
5	※	※	()	《	》	『	』	『	』	∩	∩	∩	∩	∩	∩
6	♣	ア	イ	ウ	エ	オ	カ	キ	ク	ク	コ	サ	シ	ス	セ	ソ
7	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
8	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ヰ	ヱ	ヲ
9	ン	ア	ウ	エ	オ	ヤ	ユ	ヨ	ッ	°	°	°	°	°	°	°
A	И	Й	П	И	У	Ш	Ш	Ъ	Ы	Э	Ю	Я	Б	В	Г	Д
B	Ф	Г	Ü	É	Æ	Ä	À	À	À	À	À	À	À	À	À	À
C	À	É	Æ	Æ	Ô	Ö	Ô	Ô	Ô	ÿ	Ö	Ü	Æ	Æ	f	á
D	í	ó	ú	ñ	ñ	ä	ö	ü	ü	ü	ü	ü	ü	ü	ü	ü
E	ê	ë	è	ï	î	ì	Ä	Å	É	Æ	Æ	Ô	Ö	Ô	Ü	Ü
F	ÿ	ö	ü	Æ	Æ	f	ä	í	ó	ú	ñ	ñ	ä	ö	ü	ü

A.4 国际字符集

Country	ASCII Code(Hex)											
	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U.S.A.	#	\$	@	[\]	^	`	{		}	~
France	#	\$	à	°	ç	§	^	`	é	ù	è	¨
Germany	#	\$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
U.K.	£	\$	@	[\]	^	`	{		}	~
Denmark I	#	\$	@	Æ	Ø	Å	^	`	æ	ø	å	~
Sweden	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italy	#	\$	@	°	\	é	^	ù	à	ò	è	ì
Spain I	Pt	\$	@	¡	Ñ	¿	^	`	¨	ñ	}	~
Japan	#	\$	@	[¥]	^	`	{		}	~
Norway	#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Denmark II	#	\$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Spain II	#	\$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
Latin	#	\$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú
Korea	#	\$	@	[₩]	^	`	{		}	~
Slovenia/Croatia	#	\$	Ž	Š	Đ	Ć	Č	ž	š	đ	ć	č
China	#	¥	@	[\]	^	`	{		}	~

B 条码

B.1 条码编码规则

UPC-A: UPC-A 编码要符合 UCC 组织(<http://www.uccnet.org>)的规范。UPC-E: UPC-E 编码要符合 UCC 组织(<http://www.uccnet.org>)的规范。ENA8: ENA8 编码要符合 EAN 组织(<http://www.ean-int.org>)的规范。ENA13: ENA13 编码要符合 EAN(<http://www.ean-int.org>)组织的规范。

CODE39: 又称 39 码, CODE39 的起始位字符和终止位字符必须为 '*', 且起始位和终止位之间不能包含字符 '*', 本打印机*由打印机自动给出, 编程时不用给出, 数据中可包含校验码也可不包含校验码, 校验码有固定算法。

ITF: 又称 INTERLEAVED 25, 交叉 25 码, INTERLEAVED 2 of 5, 数据位长度只能为偶数, 数据中可包含校验码也可不包含校验码, 校验码有固定算法。

CODABAR: 又称库德巴码, 起始位和终止位必须为 A、B、C、D 四个字符中的一个, 起始位字符与终止位字符不必相同, 数据中可包含校验码也可不包含校验码, 校验码由编码人自定义。

CODE93: CODE93 的起始位字符和终止位字符必须为 '*', 且起始位和终止位之间不能包含字符 '*', 本打印机*由打印机自动给出, 编程时不用给出, CODE93 数据最后必须包含两个字符的校验码, 校验码有固定算法。

B.2 条码长度字符集表

条码类型	长度	字符集(ASCII)
UPC-A	12	0~9
UPC-E	8	0~9
ENA8	8	0~9
ENA13	13	0~9
CODE39	27	0~9 A~Z - . SP \$ / + % *
INTERLEAVED 25	偶数 52	0~9
CODABAR	32	0~9 - : / % . A~D
CODE93	无限制	0~9 A~Z - . SP \$ / + % *
CODE128	33	

D 安装尺寸

