

便携式打印机

# MPT-III 系列 技术手册



厦门汉印电子科技有限公司

地址：厦门湖里区火炬高新区产业园昂业楼 305#

电话：0592-5885991

传真：0592-5885992

网址：[www.hpri.com](http://www.hpri.com)

## 更改记录

版本	日期	更改内容	更改	审核	承认
1.0	2011.12.31	——	林丽婷	林扬	任小微
1.1	2013.07.17	A.更改 “3.MPT-III打印机工具软件” 说明； ( P11 ) B.删除wifi功能； C.公司更改成汉印、网址、logo等改成HPRT D. Bluetooth 1.1规范改成Bluetooth 2.0规范 ( P9 ) E.增加部分指令集(4.1.8、4.3.8、4.5.7、4.5.8、 4.5.9、4.5.10、4.6.2、4.8.2、4.8.3)	陈卫华	林扬	任小微
1.2	2013.08.14	A.变更 ESC ! n 指令 ( P20 )	陈卫华	林扬	任小微
1.4	2013.08.19	A.增加FS &指令	陈卫华	林扬	任小微
1.5	2013.11.01	A.增加ESC v指令	陈卫华	林扬	任小微
1.6	2013.12.17	A.删除条指令 ( CR/ESC V n/FS U nL nH/GS " ) B.增加条指令 ( GS E n(P17)/ESC K n(P17)/ESC R n(P22)/ESC t n(P23)/ GS v 0 m xL xH yL yH d1....dk(P39) )	陈卫华	林扬	任小微

## 目 录

### 目录

1.产品简介 .....	6
2.通讯接口 .....	7
2.1 红外: IR 端口.....	7
2.1.1 原始红外 (RAW-IR) .....	8
2.1.2 VIR*开发中.....	8
2.1.3 IRDA IrCOMM.....	8
2.2 蓝牙 (Bluetooth) 接口.....	9
2.2.1 配对.....	9
2.2.2 使用蓝牙接口打印.....	9
2.3 RS232 端口.....	10
3.MPT-III打印机工具软件 .....	11
4. 打印控制指令 .....	12
4.1 基本控制指令.....	15
4.1.1 ESC @ .....	15
4.1.2 FF.....	15
4.1.3 LF .....	15
4.1.4 ESC J n.....	15
4.1.5 ESC d n.....	16
4.1.6 HT.....	16
4.1.7 ESC v.....	17
4.1.8 GS E n .....	17
4.1.9 ESC K n.....	17
4.2 字符参数设置命令.....	18
4.2.1 ESC ! n .....	18
4.2.2 GS ! n.....	19
4.2.3 ESC M n .....	19
4.2.4 ESC - n.....	20
4.2.5 ESC E n .....	21
4.2.6 ESC G n.....	21
4.2.7 GS B n .....	22
4.2.8 ESC R n .....	22
4.2.9 ESC t n.....	23
4.3 打印排版参数设置命令.....	25
4.3.1 ESC \$ nL nH.....	25

4.3.2 ESC D n1 n2...nk NULL.....	25
4.3.3 ESC 2.....	26
4.3.4 ESC 3 n.....	26
4.3.5 ESC SP n.....	27
4.3.6 ESC a n.....	27
4.3.7 GS L nL nH.....	28
4.3.8 GS W.....	28
4.4 图形/图象打印命令.....	29
4.4.1 ESC * m nL nH d1...dk.....	29
4.4.2 GS * x y d1...dk.....	33
4.4.3 GS / n.....	35
4.4.4 FS p n m.....	36
4.4.5 FS q n [xL xH yL yH d1...dk]1...[xL xH yL yH d1...dk]n.....	36
4.4.6 GS v 0 m xL xH yL yH d1...dk.....	38
4.5 条码打印命令.....	40
4.5.1 GS h n.....	40
4.5.2 GS w n.....	40
4.5.3 GS H n.....	41
4.5.4 GS f n.....	41
4.5.5 GS k.....	41
4.5.6 GS k m v r d1...dk [NULL].....	44
4.5.7 GS ( k <Function 167>.....	45
4.5.8 GS ( k <Function 169>.....	45
4.5.9 GS ( k <Function 180>.....	45
4.5.10 GS ( k <Function 181>.....	46
4.6 曲线打印命令.....	48
4.6.1 GS '.....	48
4.7 自定义字符命令.....	51
4.7.1 ESC % n.....	51
4.7.2 ESC & y c1 c2 [x1 d1...d(y * x1)]...[xk d1...d(y * xk)].....	51
4.7.3 ESC ?.....	53
4.8 汉字命令.....	54
4.8.1 FS &.....	54
4.8.2 FS 2 c1 c2 d1...dk.....	54
4.8.3 FS.....	54
附录.....	56
A.打印字符集.....	56
B.条码.....	57
C. 预印刷黑标说明.....	57

## 1. 产品简介

MPT-III系列便携式热敏打印机是基于ARM平台, 性能优越的支持红外、蓝牙技术的便携式票据、标签打印机。可广泛使用在移动政务、警务、烟草、邮政、运输、物流等等行业。

MPT-III系列便携式微型热敏打印机包含两个型号, 分别为MPT-III/IR、MPT-III/BL、

MPT-III/IR: 红外便携式打印机

MPT-III/BL: 蓝牙便携式打印机

性能指标:

型号	MPT-III/IR	MPT-III/BL
打印方式	热敏行式打印	
打印纸宽	76~80mm	
打印有效距离	72mm	
重量	325g (不含纸卷)	
纸卷直径	≤40mm	
打印点阵	576点/行	
内置字库	ASCII字库(16×8、24×12)、GBK(16×16、24×24)	
打印速度	50~70mm/秒	
打印机芯寿命	50km(12.5%打印密度或以下)/100 million pulses	
可打印内容	英文、数字、各种符号、汉字、图形、曲线、预存储图标、条码(CODE39、EAN13、EAN8、CODABAR、CODE128、CODE93、ITF)	
分辨率	203DPI/8点/毫米	
数据通讯接口	IRDA/VIR/RS232	蓝牙V2.0 CLASS /RS232
电源	1500mAh 7.4V 可充电锂电池	
充电方式	带机充电/选配充电器	
异常检测	缺纸侦测/电量侦测/开关盖侦测	
黑标侦测	支持	
装纸方式	易装纸结构	
指令集	ESC/POS兼容指令集	
切纸方式	手动撕纸	
使用环境	温度 -10℃~50℃ 湿度 20%~85%	
充电环境	温度 5℃~40℃ 湿度 20%~85%	
储存环境	温度 -20℃~70℃ 湿度 5%~95%	
标配附件	长效热敏纸、电源充电器、电池	
可选附近	专用串口数据线/便携套	

## 2. 通讯接口

MPT-III系列便携式打印机可用四种物理接口与主机进行数据传输，分别为红外(IrDA)、蓝牙(Bluetooth)、RS232异步串行口。

红外(IrDA):国际红外数据通讯协会制定的IrCOMM协议;

国际红外数据通讯协会制定的物理层规范RAW-IR;

VIR为在RAW-IR基础上集成了VIR协议的数据通讯方式;

蓝牙 (Bluetooth): 支持Bluetooth 2.0协议

RS232:有线异步串口。

### 2.1 红外: IR端口

红外数据通讯技术是红外数据协会 (IrDA) 开发并发展起来的一项用红外光作为通讯数据载体的一种无线数据通讯技术。

红外 (IR) 端口作为无线数据传输接口，因其功耗低、技术成熟、使用方便等等诸多原因现在为大多数便携式设备作为数据通讯的主要手段。大部分的便携式设备都有红外 (IR) 端口，比如所有的WINCE掌上电脑、PALM掌上电脑、笔记本电脑，一部分手机、大部分便携式数据采集器，以及一小部分用于野外作业的测量仪器。

MPT-III 打印机在硬件上符合IrDA1.1物理层的规范。

由于红外 (IR) 端口是以红外光线作为数据载体，所以在数据发送设备和接收设备之间不能有障碍物，双方的红外端口要互相对准，并且距离不能太远。

所以MPT-III/IR打印机与其他符合标准IrDA物理层规范的主机设备通过红外 (IR) 端口连接时，要注意主机和MPT-III/IR打印机之间不能有障碍物，红外端口之间的夹角不能大于30°，距离不能超过0.5M。

MPT-III/IR可以使用原始红外 (RAW-IR) 与主机进行通讯，也可以选择采用Vir协议或IrCOMM协议 (IrDA标准)与主机进行通讯。

当采用原始红外与主机进行通讯时，红外收发器只是简单地按照一定地编码规则将串口数据信号转为红外光信号或将光信号转为串口数据信号。在这种用法下，红外端口被称为“原始红外 (RAW-IR)”，当您使用原始红外 (RAW-IR) 时，无法跟IrDA兼容，因为在收发数据的过程中，软件并没有使用IrDA协议栈。

VIR是一种更安全可靠的红红外数据通讯方式，即在原始红外的基础上增加了VIR协议。如果用户主机不支持IrCOMM协议，推荐使用VIR协议。

相较于VIR协议，IrCOMM虽然也是基于原始红外硬件基础上的协议，但它是一种更通用的有国际红外数据通讯协会制定的红外数据传输协议。IrCOMM是IrDA协议的一个子集，IrDA协议是在原始红外 (RAW-IR) 的硬件基础之上为了确保数据传输的稳定性、可靠性、易用性而由IrDA协会开发出的无线数据通讯协议，几乎所有支持IrDA协议栈的便携式设备(如WINCE、POCKET PC、PALM各种红外手机等等)都支持IrCOMM。在这种模式下，如果驱动打印机的主机设备使用的是支持IrDA协议栈的操作系统（比如WINCE、PALM OS），那么对于开发者来说，IrCOMM端口就是一个由软件虚拟的串行端口，如果主机设备的操作系统没有IrDA协议栈，如果要想实现IrCOMM模式数据传输，那么需要开发者自己编写IrDA协议栈。主机设备是否支持IrCOMM，请参阅主机设备的开发资料或向主机设备制造商咨询。

由于MPT-III/IR打印机中原始红外 (RAW-IR) 端口和VIR端口和IrCOMM端口使用同样的硬件资源，所以原始红外 (RAW-IR)、VIR和IrCOMM不能同时使用，MPT-III/IR默认的红外模式为IrComm，如要更改当前红外 (IR) 端口的使用模式，需要专用的工具MPTTools修改设置，详见【3、MPT-III打印机工具软件】。

如要了解详细的IrDA协议请参见IrDA协会官方网站 (<http://www.irda.org/>) 公布的技术资料

### 2.1.1 原始红外（RAW-IR）

原始红外(IR)端口由于是直接把红外（IR）收发器附加在RS232异步串口的输入输出端，所以红外收发器只是简单地将RS232异步串口数据信号按照一定的编码规则转为红外光信号或将光信号转为串口数据信号。因此对于原始红外端口来说，除了细微的差别，主机应用开发者只需要象操作RS232异步串口那样操作主机的原始红外（RAW-IR）就可以了。

在操作主机原始红外（RAW-IR）端口之前，必须先知道原始红外（RAW-IR）端口的端口号，原始红外（RAW-IR）端口的端口号可以从主机设备的开发资料或主机设备制造商处获知。对于部分可以使用原始红外（RAW-IR）端口的主机设备，打开原始红外（RAW-IR）端口的方法可能和打开串口的方法不一样，请仔细参阅主机设备的开发资料。

当然原始红外（RAW-IR）端口和标准的RS232异步串行口仍然有细微的差别，原始红外端口由于采用无线红外连接方式，所以只有串口的TXD信号和RXD信号有效，对于主机RS232其他引脚的操作不会对原始红外（RAW-IR）端口产生影响。

MPT-III/IR打印机在休眠状态下可以通过原始红外（RAW-IR）端口唤醒。

MPT-III/IR打印机的原始红外(RAW-IR)可以工作的波特率为: 9600bps, 19200bps, 38400bps, 57600bps, 115200bps。出厂时打印机的原始红外波特率(RAW-IR)被设置成9600bps，如果用户要更改波特率，需要专用的工具MPTTools修改设置，详见【3、MPT-III打印机工具软件】。

注意：并不是所有的主机的原始红外（RAW-IR）都能在这些波特率下工作，在修改波特率前，请先确认您所拥有主机设备的原始红外（RAW-IR）是否能在该波特率下工作。

在主机设备编程时，原始红外（RAW-IR）端口的设置请遵照如下设置：

数据位：8位； 停止位 1位； 奇偶校验：无； 流控制 无。

在使用原始红外（RAW-IR）端口时请注意，虽然对于一般的红外源，原始红外（RAW-IR）有足够的抗干扰能力，但是对于正在试图搜索其他红外设备的红外设备（比如笔记本电脑、打开红外功能的手机）所发出的红外光，打印机并不能有效的屏蔽这些干扰，所以使用时切记不能靠近这些红外信号源。

### 2.1.2 VIR\*开发中

为使便携式微型打印机能可靠地与不带IrDA协议栈的诸多红外手持终端进行红外数据通讯，开发了VIR协议，本协议是基于符合IrDA物理层规范的硬件的打印数据通讯协议。

任何可以使用原始红外收发数据的设备都可以使用此协议控制MPT-III/IR打印。

打印机在休眠状态下可以通过VIR协议唤醒。

VIR协议对于没有集成IrDA协议栈的红外数据通讯设备来说，是一个很好的替代IrDA协议的数据通讯协议。VIR协议消耗资源少，并且可以很容易地实现，但同时可以有效地防止红外数据传输过程中出现误码。有关VIR协议的定义及如何实现VIR协议请见附录D，如需技术支持，请浏览本公司技术支持人员，信箱mk@prttech.com

### 2.1.3 IRDA IrCOMM

IrDA的协议有很多，IrCOMM是IrDA协会推荐使用在打印机上的红外数据通讯协议。

在使用IrCOMM协议时，打印机可以完全杜绝其他红外源的干扰。

MPT-III/IR支持IrCOMM协议。

在使用IrCOMM时，虽然把它当作一个虚拟串行口，但是不需要设置波特率，数据传输的真实波特率是在IrCOMM协议工作的时候自适应的。并且因为IrCOMM协议负责数据校验、数据缓冲等工作，所以对于IrCOMM串口的其他设置都是没有意义的。



对于开发者来讲，需要知道的是端口号以及如何将其打开。IrCOMM端口的端口号以及打开方法可以从主机设备的开发资料或主机设备制造商处获知。

打印机在休眠状态下可以通过IrCOMM端口唤醒。

如果开发者需要自己开发IrCOMM协议，请参见IrDA协会官方网站 (<http://www.irda.org/>) 公布的技术资料。

## 2.2 蓝牙 (Bluetooth) 接口

蓝牙是一种支持设备短距离通信（一般是10m之内）的无线电技术。能在包括移动电话、PDA、打印机、笔记本电脑、无线耳机等相关外设等众多设备之间进行无线信息交换。蓝牙的标准是IEEE802.15，工作在2.4GHz 频带，带宽为1Mb/s。如需蓝牙更详细的信息，请访问蓝牙官方网站：<http://www.bluetooth.org>。

MPT-III/BL 支持蓝牙无线数据传输接口，符合Bluetooth 2.0规范，功率级别为CLASS 2。

MPT-III/BL是一个蓝牙从设备，只能由蓝牙主设备（如PDA、手机、笔记本电脑）驱动打印机，其他如蓝牙耳机等蓝牙从设备无法通过蓝牙驱动打印机打印。

MPT-III/BL缺省的设备名为MPT-III/BL，开发者可根据自己的需要更改设备名。更改设备名的方法详见【3、MPT-III打印工具软件】。

MPT-III/BL缺省的蓝牙连接密码为PRT，开发者可根据自己的需要更改连接密码。更改设备名的方法详见【3、MPT-III打印工具软件】。

### 2.2.1 配对

MPT-III/BL便携式微型打印机工作前需与驱动MPT-III/BL便携式微型打印机的主设备配对，配对过程由主设备发起。

通常的配对方法如下：

- 1、打印机开机，
- 2、主设备搜寻外部蓝牙设备，
- 3、如果有多台外部蓝牙设备的话，选中**MPT-III/BL**打印机
- 4、输入密码“0000” 或通过打印机自检得到蓝牙密码

【自检方法】关机（如果打印机处于开机状态的话），然后按住走纸键【FEED】，再按电源【POWER】键开机，打印机即会打印出自检页，自检页中包含当前打印机的状态及打印机设置和打印的范例

- 5、完成配对。

具体的配对方法请参阅主设备蓝牙功能说明。

配对时，MPT-III/BL便携式微型打印机必须处于开机状态。

注意：配对时，请不要将多台打印机同时开机，否则可能无法判断配对成功的是哪一台打印机。

配对成功后，其他上位机仍然可以与该打印机配对，每台打印机最多可以与8台上位机配对，如果更多的上位机与打印机配对的话，那么最早与打印机的上位机会被打印机从配对列表中自动清除，此时如果这台上位机需要驱动打印机打印的话，需要重新配对。

### 2.2.2 使用蓝牙接口打印

对于有虚拟蓝牙串口的上位机(如使用SMARTPHONE的手机、POCKET PC、PALM、笔记本电脑等)来说，配对成功后，就可以通过虚拟蓝牙串口向MPT-III/BL便携式微型打印机发送打印数据进行打印了。如果上位机没有虚拟蓝牙串口，如要驱动MPT-III/BL便携式微型打印机打印，请咨询上位机供应商。



## 2.3 RS232端口

RS232接口是最常用的数据通讯接口。MPT-III打印机都带RS232异步串行数据接口。

MPT-III打印机RS232端口规格：

数据传送： 串行

同步方式： 异步

握手信号： 无

流控制： 硬件流控制/软件流控制/无 可选

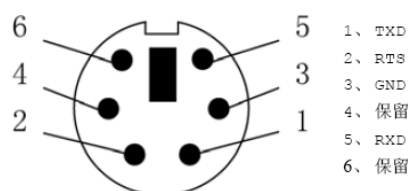
波特率： 9600bps、19200bps、38400bps、57600bps、115200bps可选。

数据字长度： 8位

校验方式： 无

停止位： 1位

插座管脚定义(打印机侧)： 6PIN MINI-DIN 下图为打印机RS232通讯插座的管脚定义：



TXD： 打印机数据发送

RTS： 打印机请求数据（仅当打印机设为硬件流控制方式时，该管脚生效，否则该管脚无用，设置打印机为硬件流控制方式的方法见【3、MPT-III打印机工具软件】）

GND： 地

RXD： 打印机数据接收

注意：保留的管脚本公司有其他用途，开发者如要自行制作连接线，请不要使用保留的管脚，否则可能导致打印机不能正常工作甚至损坏打印。

打印机的数据接收缓冲区大小为3K byte，当打印数据量小于3K byte时，无需使用流控制，接线方式如下：

打印机	主设备
RXD	TXD
GND	GND

当打印数据量大于3K byte时，需要使用流控制，当使用蓝牙、IrCOMM与打印机连接时，开发者无需考虑流控制，当使用VIR协议时，开发者需根据VIR的流控制方式来避免打印机的数据缓冲区溢出，原始红外无法使用流控制。

当打印机被设为硬件流控制时，应按如下方法连接主设备与打印机：

打印机	主设备
TXD	RXD
RTS	CTS
RXD	TXD
GND	GND

主设备在通讯过程中监测CTS的电平，当CTS电平为高时，主设备可以向打印机发送数据，当CTS电平为低时，表示打印机忙，需停止向打印机发送数据，直到再次监测到CTS电平为高时，继续发送打印数据。

软件流控制采用XON/XOFF方式，当采用软件流控制方式时，应按如下方式连接主设备与打印机：

打印机	主设备
TXD	RXD
RXD	TXD
GND	GND

当使用软件流控制时，主设备需检测自身RXD的数据以判断打印机数据缓冲区是否满。具体方法如下：开始打印时，主设备向打印机发送数据，同时监测串行口接收的数据，当接收到XOFF(0x13)时，停止向打印机发送数据，当接收到XON(0x11)时，重新开始发送数据，再次接收到XOFF时，再停止向打印机发送数据，等待再次接收XON(0x11)时，再重新开始发送数据，如此循环，直到发送完打印数据。

### 3.MPT-III打印机工具软件

MPT-III打印机有一个运行在PC上的工具软件MPTTools。

MPTTools为设置打印机用户参数的工具软件，可设置打印机串口波特率，红外通讯模式、原始红外和VIR的波特率，打印机的蓝牙设备名称及密码等等。

具体的使用方法请参见MPTTools帮助，该帮助随MPTTools一起发布。

## 4. 打印控制指令

MPT-III打印机控制采用ESC/POS兼容指令。通过打印机的通讯接口向打印机发送打印机控制指令和打印数据以控制打印机打印。

下表中是打印指令的简表，按照功能分类

章节	指令	简述
打印机控制指令		
4.1.1	<u>ESC @</u>	打印机初始化
4.1.2	<u>FF</u>	打印并走纸到下页首（仅用于黑标定位时）
4.1.3	<u>LF</u>	打印并换行
4.1.4	<u>ESC J n</u>	打印并进纸n个垂直运动单位
4.1.5	<u>ESC d n</u>	打印并进纸n行
4.1.6	<u>HT</u>	移动打印位置到下一个水平制表位置
4.1.7	<u>ESC v</u>	向主机传送打印机状态
4.1.8	<u>GS E n</u>	选择打印头控制模式
4.1.9	<u>ESC K n</u>	反向走纸
字符参数设置命令		
4.2.1	<u>ESC ! n</u>	设置字符打印模式
4.2.2	<u>GS ! n</u>	设置字符大小
4.2.3	<u>ESC M n</u>	设置打印字体
4.2.4	<u>ESC - n</u>	设置/取消下划线打印
4.2.5	<u>ESC E n</u>	设置/取消粗体打印
4.2.6	<u>ESC G n</u>	设置/取消重叠（粗体）打印、效果同ESC E
4.2.7	<u>GS B n</u>	设置/取消反白打印
4.2.8	<u>ESC R n</u>	选择国际字符集
4.2.9	<u>ESC t n</u>	选择字符代码表
打印排版参数设置命令		
4.3.1	<u>ESC \$ nL nH</u>	设置绝对打印位置
4.3.2	<u>ESC D n1 n2...nk NULL</u>	设置水平制表位
4.3.3	<u>ESC 2</u>	设置相对打印位置
4.3.4	<u>ESC 3 n</u>	设置行间距
4.3.5	<u>ESC SP n</u>	设置字间距
4.3.6	<u>ESC a n</u>	设置对齐方式
4.3.7	<u>GS L nL nH</u>	设置左边距
4.3.8	<u>GS W</u>	设置打印区域宽度
图形/图象打印命令		

4.4.1	<u>ESC * m nL nH d1...dk</u>	打印位图
4.4.2	<u>GS * x y d1...dk</u>	定义下传位图
4.4.3	<u>GS / n</u>	打印下传位图
4.4.4	<u>FS p n m</u>	打印下载到 NV 存储器中的位图
4.4.5	<u>FS q n [xL xH yL yH d1...dk]1...[xL xH yL yH d1...dk]n</u>	定义 NV 位图
4.4.6	<u>GS v 0 m xL xH yL yH d1....dk</u>	打印光栅位图
条形码打印命令		
4.5.1	<u>GS h n</u>	设置条码打印高度
4.5.2	<u>GS w n</u>	设置条码宽度
4.5.3	<u>GS H n</u>	设置条码阅读字符的位置
4.5.4	<u>GS f n</u>	设置条码阅读字符的字体
4.5.5	<u>GS k</u>	打印条码
4.5.6	<u>GS k m v r d1.....dk [NULL]</u>	打印QRCODE(二维条码)
4.5.7	<u>GS ( k &lt;Function 167&gt;</u>	QR码：设置模块大小
4.5.8	<u>GS ( k &lt;Function 169&gt;</u>	QR码：选择纠错水平
4.5.9	<u>GS ( k &lt;Function 180&gt;</u>	QR码：在符号存储区存储数据
4.5.10	<u>GS ( k &lt;Function 181&gt;</u>	QR码：在符号存储区打印符号数据
曲线打印命令		
4.6.1	<u>GS `</u>	打印曲线
用户自定义字符		
4.7.1	<u>ESC % n</u>	允许/禁止用户自定义字符
4.7.2	<u>ESC &amp; y c1 c2 [x1 d1...d(y * x1)]...[xk d1...d(y * xk)]</u>	定义用户自定义字符
4.7.3	<u>ESC ?</u>	取消用户自定义字符
汉字命令		
4.8.1	<u>FS &amp;</u>	设置汉字模式
4.8.2	<u>FS 2 c1 c2 d1...dk</u>	定义用户自定义汉字
4.8.3	<u>FS .</u>	解除汉字字符打印模式

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PrtSendData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：

```
PrtSendData(char *buf, int len)
```

描述：向打印机发送数据；

char \*buf：打印数据的指针；

int len：数据长度，单位：字节。

## 4.1 基本控制指令

### 4.1.1 ESC @

[名称] 初始化打印机

[格式] ASCII码 ESC @

十六进制码 1B 40

十进制码 27 64

[描述] 清除打印缓冲区中的数据，复位打印机打印参数到当打印机缺省参数。

[注意] • 不是完全恢复到出厂设置，用户参数设置（见3、MPT-III打印机工具软件）不会被更改。

```
[例子] char SendStr[3];
SendStr[0] = 0x1B
SendStr[1] = 0x40;
PrtSendData(SendStr, 2);
```

### 4.1.2 FF

[名称] 打印并走纸到下页首

[格式] ASCII码 FF

十六进制码 0C

十进制码 12

[描述] 将打印缓冲区中的数据全部打印出来并返回标准模式。

[注意] • 打印后，删除打印缓冲区中的数据。

- 该命令设置打印位置为行的起始点。
- 如果打印纸有预印刷黑标，则打印缓冲区中的数据后，走纸到黑标处，如果打印纸无黑标，则走纸0.5m后停止，预印刷黑标的规范请见附录C. 预印刷黑标说明。

```
[例子] char SendStr[2];
SendStr[0] = 0x0C;
PrtSendData( SendStr, 1);
```

### 4.1.3 LF

[名称] 打印并换行

[格式] ASCII码 LF

十六进制码 0A

十进制码 10

[描述] 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。

[参考] CR

```
[例子] char SendStr[2];
SendStr[0] = 0x0A; //C语言中'\n'即为换行
PrtSendData( SendStr, 1);
```

### 4.1.4 ESC J n

[名称] 打印并进纸

[格式] ASCII码 ESC J n

十六进制码 1B 4A  $n$

十进制码 27 74  $n$

[范围]  $0 \leq n \leq 255$

[描述] 打印输出打印缓冲区中的数据，并进纸 $n$ 个 垂直点距。

[注意] • 打印结束后，将下一行的开始设定为打印起始位置。

• 一个垂直点距为0.125mm, 以下同。

[参考] **ESC d**

[例子] `char SendStr[4];`

`SendStr[0] = 0x1B;`

`SendStr[1] = 0x4A;`

`SendStr[2] = 0x08;`

`PrtSendData( SendStr, 3); //走纸1mm`

#### 4.1.5 ESC d n

[名称] 打印并进纸 $n$  行

[格式] ASCII码 ESC d  $n$

十六进制码 1B 64  $n$

十进制码 27 100  $n$

[范围]  $0 \leq n \leq 255$

[描述] 打印打印缓冲区中的数据并进纸 $n$ 字符行。

[注意] • 该命令设置打印起始位置为行起点。

[参考] **ESC J**

[例子] `char SendStr[4];`

`SendStr[0] = 0x1B;`

`SendStr[1] = 0x64;`

`SendStr[2] = 0x02;`

`PrtSendData( SendStr, 3); //走纸2行`

#### 4.1.6 HT

[名称] 移动打印位置到下一个水平制表位置

[格式] ASCII码 HT

十六进制码 09

十进制码 9

[描述] 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。

• 水平定位位置的缺省设定为字符A(12 ? 24)的8个字符宽度(第9, 17, 25, ... 列)。

[参照] **ESC D**

[例子] `char NextPos = 0x09;`

`PrtSendData("商品名", 6);`

`PrtSendData(&NextPos, 1);`

`PrtSendData("单价", 4);`

`PrtSendData(&NextPos, 1);`



```
PrtSendData("数量",4);  
PrtSendData(&NextPos,1);  
PrtSendData("金额",4);
```

#### 4.1.7 ESC v

[名称] 向主机传送打印机状态

[格式] ASCII码       ESC v  
         十六进制码   1B 76  
         十进制码     27 118

[描述] 向主机传送打印机状态，仅串口型打印机有效。

当打印机接到该命令后，向主机传送一个字节的数据。

该字节各位定义如下：

返回1：打印机未就绪

返回0：打印机就绪

#### 4.1.8 GS E n

[名称] 选择打印头控制模式

[格式] ASCII码       GS     E       n  
         十六进制码   1D     45       n  
         十进制码    29     69       n

[范围]  $0 \leq n \leq 15$

[描述] 选择打印头控制模式

n	点行/ms	n	点行/ms
0	0.8	8	1.6
1	0.9	9	1.7
2	1.0	10	1.8
3	1.1	11	1.9
4	1.2	12	2.0
5	1.3	13	2.1
6	1.4	14	2.2
7	1.5	15	2.3

#### 4.1.9 ESC K n

[名称] 反向走纸

[格式] ASCII   ESC     K       n  
         Hex     1B     4B       n  
         Decimal 27     75       n

[描述] 反向走纸 [ $n \times 0.125$ ] 英寸。

[范围]  $0 \leq n \leq 255$

[注释] • 退纸可能引起以下问题：

- 走纸距离不准确；
- 退纸噪音比进纸噪音大；
- 退纸时，由于色带可能使纸着色。

## 4.2 字符参数设置命令

### 4.2.1 ESC ! n

[命令] 选择打印模式

[格式] ASCII码 ESC ! n

十六进制码 1B 21 n

十进制码 27 33 n

[范围]  $0 \leq n \leq 255$

[描述] 通过指定参数n 的值选择打印模式。参数n 的定义如下：

位	值		意义
0 ~1	0	0	西文字符（半宽）字体A (12×24)，汉字字符（全宽）字体A (24×24)
	1	0	
	0	1	西文字符（半宽）字体B (8×16)，汉字字符（全宽）字体B (16×16)
	1	1	西文字符（半宽）字体A (9×24)，汉字字符（全宽）字体A (18×24)
2	—		未定义
3	0		取消粗体模式
	1		设置粗体模式
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	—		未定义
7	0		取消下划线模式
	1		设置下划线模式

[注意]

- 当同时选择倍高及倍宽模式时，则打印出四倍大小字符。
- 打印机可以为所有字符加下划线，但不能为由HT命令产生的空白或顺时针旋转90的字符加下划线。？
- 当一行中有一些倍高或更高字符时，行中所有字符都沿基线对齐。
- ESC M也可设定字体。最后接收到的命令的设定有效。
- ESC E也可设定或取消粗体模式。最后接收到的命令的设定有效。
- ESC \_也可设定或取消下划线模式，最后接收到的命令的设定有效。
- GS !也可设定字符大小。最后接收到的命令的设定有效。
- 本命令对英数字符和汉字都有效。

[缺省值]  $n = 0$

[参照] ESC -, ESC E, GS !, ESC M

[例子] char SendStr[4];

```
SendStr[0] = 0x1B;
SendStr[1] = 0x21;
SendStr[2] = 0x28; // 00101000 倍宽粗体
PrtSendData( SendStr, 3);
```

### 4.2.2 GS ! n

[名称] 放大字符

[格式] ASCII码 GS ! n

十六进制码 1D 21 n

十进制码 29 33 n

[范围]  $0 \leq n \leq 255$  ( $1 \leq$  垂直倍数  $\leq 8$ ,  $1 \leq$  水平倍数  $\leq 8$ )

[描述] 用位0~3位选择字符高度, 用位4~7位选择字符宽度, 如下所示:

0	1	2	3	高度
0	0	0	0	1倍
1	0	0	0	2倍

4	5	6	7	宽度
0	0	0	0	1倍
1	0	0	0	2倍

[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于ESC V, 当两个命令同时有效时, 字符显示是先旋转, 再放大。
- 当字符以不同的尺寸在一行中放大时, 一行中所有的字符沿基线对齐。
- 用**ESC !** 命令也可以设置字符大小。以最后接收到的命令设置当前模式。

[缺省值]  $n = 0$

[参考] **ESC !**, **ESC @**

[例子] char SendStr[4];

```
SendStr[0] = 0x1D;
```

```
SendStr[1] = 0x21;
```

```
SendStr[2] = 0x01; // 00000001 倍高
```

```
PrtSendData( SendStr, 3);
```

### 4.2.3 ESC M n

[名称] 选择字符字体

[格式] ASCII码 ESC M n

十六进制码 1B 4D n

十进制码 27 77 n

[范围]  $n = 0, 1, 16, 17, 18, 19$

[描述] 选择字符字

N (十进制)	意义
0	西文字符 (半宽) 字体(12×24)
1	西文字符 (半宽) 字体(8×16)
16	简体汉字字符字体24×24

17	简体汉字字符字体16×16
18	BIG5汉字字符字体24×24
19	BIG5汉字字符字体16×16

[注意] • ESC !也可设定字体。最后接收到的命令的设定有效。

• 当使用该命令设定字体时。可以分别设定西文字体和汉字字体，而且互不影响。

[参考] **ESC !, ESC @**

[例子] char SendStr[8];

SendStr[0]=0x1B;

SendStr[1]=0x4d;

SendStr[2]=0x00; // 西文12×24

SendStr[0]=0x1B;

SendStr[1]=0x4D;

SendStr[2]=0x10; // 简体中文16×16

PrtSendData( SendStr, 6); //以后打印的中文字体为16×16，西文为12×24

#### 4.2.4 ESC - n

[名称] 设置 / 取消下划线

[格式] ASCII码 ESC - n

十六进制码 1B 2D n

十进制码 27 45 n

[范围]  $0 \leq n \leq 2$

[描述] 基于以下的n值，设定 / 解除下划线模式：

n (十进制)	意义
0	解除下划线模式
1	设定下划线模式(1 点粗)
2	设定下划线模式(2 点粗)

[注意] • 打印机不能给旋转字符以及反白字符打印下划线。

• 改变字符大小不影响当前下划线的粗细。

• 使用ESC !也可以设定或解除下划线模式。最后接收的命令设定有效。

• 该命令对英文和汉字字符都有效。

[缺省值]  $n = 0$

[参照] **ESC !**

[例子] char SendStr[16];

SendStr[0] = 0x1B;

SendStr[1] = 0x2D; //单行下划线

SendStr[2] = 0x01;

SendStr[3] = 't';

SendStr[4] = 'e';

SendStr[5] = 's';

SendStr[6] = 't';

SendStr[7] = 0x0A;

```
SendStr[8] = 0x1B;
SendStr[9] = 0x2D;
SendStr[10] = 0x00;
SendStr[11] = 't';
SendStr[12] = 'e';
SendStr[13] = 's';
SendStr[14] = 't';
SendStr[15] = 0x0A;
PrtSendData( SendStr, 16);
```

#### 4.2.5 ESC E n

[名称] 设定/解除粗体打印

[格式] ASCII码 ESC E *n*

十六进制码 1B 45 *n*

十进制码 27 69 *n*

[范围]  $0 \leq n \leq 255$

[描述] 设定或解除粗体打印模式。

当*n* 的最低位(LSB)为0时, 解除粗体打印模式。

当*n* 的最低位(LSB)为1时, 设定粗体打印模式。

[注意] • 仅*n* 的最低有效位允许使用。

• 使用ESC !也可以设置或取消粗体模式。最后接收的命令设定有效。

[缺省值] *n* = 0

[参照] ESC !, ESC G

[例子] char SendStr[3];

SendStr[0] = 0x1B;

SendStr[1] = 0x45;

SendStr[2] = 0x01;//粗体

PrtSendData(SendStr,3);

#### 4.2.6 ESC G n

[名称] 设定/解除重叠打印

[格式] ASCII码 ESC G *n*

十六进制码 1B 47 *n*

十进制码 27 71 *n*

[范围]  $0 \leq n \leq 255$

[描述] 设定或解除重叠打印模式。

当*n* 的最低有效位(LSB)为0时, 解除重叠打印模式。

当*n* 的最低有效位(LSB)为1时, 设定重叠打印模式。

[注意] • 仅*n* 的最低有效位允许使用。

• 在重叠模式和粗体模式中打印机输出是相同的。

[缺省值] *n* = 0

[参照] ESC E, ESC !

[例子] char SendStr[3];

```
SendStr[0] = 0x1B;  
SendStr[1] = 0x47;  
SendStr[2] = 0x01;//设定重叠打印模式  
PrtSendData( SendStr, 3);
```

#### 4.2.7 GS B n

[名称] 设置/取消反白打印模式

[格式] ASCII    GS    B    n  
         Hex    1D   42    n  
         Decimal 29   66    n

[范围]  $0 \leq n \leq 255$

[描述] 设置/取消反白打印模式。

当n 的最低有效位(LSB)为0时，取消反白打印模式。

当n 的最低有效位(LSB)为1时，设定反白打印模式。

n只有最低位有效。

- 这条命令对所有字符（除了HRI字符）有效。
- 在反白打印模式下，在黑色的背景上打印出白色的字符。
- 这条指令不影响位图、自定义位图、条码、HRI字符以及由HT, ESC \$, and ESC \设置的空白。
- 黑白反显打印模式比下划线模式优先级高。在黑白反显打印模式选择时，下划线模式不起作用，取消黑白反显模式后，设置下划线模式才起作用。

[默认值]  $n = 0$

```
[例子] char SendStr[3];  
SendStr[0] = 0x1D;  
SendStr[1] = 0x42;  
SendStr[2] = 0x31;// 黑白反显打印  
PrtSendData( SendStr, 3);
```

#### 4.2.8 ESC R n

[名称]选择国际字符集

[格式] ASCII        ESC        R        n  
         Hex        1B        52        n  
         Decimal    27        82        n

[范围]  $0 \leq n \leq 15$

[描述] 从下表中选择一国际字符集 n

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

[默认值] n = 0

国际字符集的内容参考表

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	ˆ
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	\$	@	ı	Ñ	ı	^	`	ñ	ı	~	~
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	ı	Ñ	ı	é	`	í	ñ	ó	ú
Latin	#	\$	á	ı	Ñ	ı	é	ü	í	ñ	ó	ú
Korea	#	\$	@	[	₩	]	^	`	{		}	~
Slovenia/Croatia	#	\$	Ž	Š	Đ	Č	Č	ž	š	đ	ć	č
China	#	¥	@	[	\	]	^	`	{		}	~

## 4.2.9 ESC t n

[名称]选择字符代码表

[格式]ASCII      ESC      t      n

Hex      1B      74      n

Decimal    27    116      n

[范围]

$0 \leq n \leq 10$ ,  $15 \leq n \leq 50$

[描述]

从字符代码表中选择代码页n，n选择如下表所示：

N	Code Page	N	Code Page
0	CP437 [U.S.A., Standard Europe]	25	WCP1257 [Baltic]
1	Katakana	26	Thai
2	CP850 [Multilingual]	27	CP720[Arabic]
3	CP860 [Portuguese]	28	CP855
4	CP863 [Canadian-French]	29	CP857[Turkish]
5	CP865 [Nordic]	30	WCP1250[Central Eurpoe]
6	WCP1251 [Cyrillic]	31	CP775
7	CP866 Cyrillic #2	32	WCP1254[Turkish]
8	MIK[Cyrillic /Bulgarian]	33	WCP1255[Hebrew]
9	CP755 [East Europe, Latvian 2]	34	WCP1256[Arabic]



10	Iran	35	WCP1258[Vietnam]
11	reserve	36	ISO-8859-2[Latin 2]
12	reserve	37	ISO-8859-3[Latin 3]
13	reserve	38	ISO-8859-4[Baltic]
14	reserve	39	ISO-8859-5[Cyrillic]
15	CP862 [Hebrew]	40	ISO-8859-6[Arabic]
16	WCP1252 Latin I	41	ISO-8859-7[Greek]
17	WCP1253 [Greek]	42	ISO-8859-8[Hebrew]
18	CP852 [Latina 2]	43	ISO-8859-9[Turkish]
19	CP858 Multilingual Latin I +Euro)	44	ISO-8859-15 [Latin 3]
20	Iran II	45	Thai2
21	Latvian	46	CP856
22	CP864 [Arabic]	47	Cp874
23	ISO-8859-1 [West Europe]		
24	CP737 [Greek]		

## 4.3 打印排版参数设置命令

### 4.3.1 ESC \$ nL nH

[名称] 设置绝对打印位置

[格式] ASCII码 ESC \$ nL nH

十六进制码 1B 24 nL nH

十进制码 27 36 nL nH

[范围]  $0 \leq nL \leq 255$

$0 \leq nH \leq 255$

[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 ( $N > 384$ ), 则被设置为可打印区域的最大值 ( $N = 384$ )。

[参照] ESC \

[例子] char SendStr[4];

SendStr[0] = 0x1B;

SendStr[1] = 0x24;

SendStr[2] = 0x18; //  $3 \times 8 = 24$

SendStr[3] = 0x00;

PrtSendData( SendStr, 4); //绝对位置设为距左边界3毫米 (24水平点距)

PrtSendData( "从左侧3毫米处开始打印\n", 22);

### 4.3.2 ESC D n1 n2...nk NULL

[名称] 设置水平制表位

[格式] ASCII码 ESC D n1...nk NULL

十六进制码 1B 44 n1...nk 00

十进制码 27 68 n1...nk 0

[范围]  $1 \leq n \leq 255$   $0 \leq k \leq 8$

[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 ! GS !)的影响。
- 该命令删除了之前设定的水平定位位置。
- 当设置n = 8时, 通过发送HT, 打印位置被移到第九列。
- 可以设置8个定位位置(k = 8)。超过8定位位置的数据被处理为普通数据。
- 按升序传输[n]k, 并且在末尾放置一个NULL码0。
- 该命令中nk > n(k-1), 如果nk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)的8个字符间隔(列9, 17, 25, ... )。

[参照] **HT**

```
[例子] char SendStr[7];
char NextPos = 0x09;
SendStr[0] = 0x1B;
SendStr[1] = 0x44;
SendStr[2] = 0x0B; // 距第一列10个字符间距
SendStr[3] = 0x11; // 距第一列16个字符间距
SendStr[4] = 0x17; // 距第一列22个字符间距
SendStr[5] = 0x1D; // 距第一列28个字符间距
SendStr[6] = 0x00; // 结束
PrtSendData(SendStr,7)
PrtSendData("姓名",4);
PrtSendData(&NextPos,1);
PrtSendData("语文",4);
PrtSendData(&NextPos,1);
PrtSendData("数学",4);
PrtSendData(&NextPos,1);
PrtSendData("外语",4);
PrtSendData(&NextPos,1);
PrtSendData("总分",4);
```

### 4.3.3 ESC 2

[名称] 选择缺省行间距

[格式] ASCII码 ESC 2

十六进制码 1B 32

十进制码 27 50

[描述] 将当前字符行间距设置为缺省行间距：1mm（8个垂直点距）。

[注意] • 该命令将影响图片与字符之间的行间距。

[参照] **ESC 3**

```
[例子] char SendStr[4];
SendStr[0] = 0x1B;
SendStr[1] = 0x32;
PrtSendData(SendStr,2);
```

### 4.3.4 ESC 3 n

[名称] 设置行间距

[格式] ASCII码 ESC 3 n

十六进制码 1B 33 n

十进制码 27 51 n

[范围]  $0 \leq n \leq 255$

[描述] 设置字符行间距为n个垂直点距。

[注意] • 该命令将影响图片与字符之间的行间距。

[缺省值]  $n = 8$

[参照] **ESC 2**

[例子] `char SendStr[4];`

`SendStr[0] = 0x1B;`

`SendStr[1] = 0x33;`

`SendStr[2] = 0x10;`

`PrtSendData(SendStr, 3);` // 设置行间距为16个垂直点距 (2mm)

### 4.3.5 ESC SP $n$

[名称] 设置字符间距

[格式] ASCII码 `ESC SP  $n$`

十六进制码 `1B 20  $n$`

十进制码 `27 32  $n$`

[范围]  $0 \leq n \leq 255$

[描述] 设置字符右侧的间距为 $n$  个水平点距。

[注意] • 在倍宽模式下，字符右侧间距是正常值的两倍。当字符被放大时，字符右侧间距被放大同样的倍数。

• 该命令同时影响英文和汉字字符的设定。

[缺省值]  $n = 0$

[例子] `char SendStr[4];`

`SendStr[0] = 0x1B;`

`SendStr[1] = 0x20;`

`SendStr[2] = 8;`

### 4.3.6 ESC a $n$

[名称] 选择对齐方式

[格式] ASCII码 `ESC a  $n$`

十六进制码 `1B 61  $n$`

十进制码 `27 97  $n$`

[范围]  $0 \leq n \leq 2$

[描述] 将一行数据按照 $n$ 指定的位置对齐。

$n$ 的可选值及意义：

$n$	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。

• 该命令在打印区域执行对齐。

• 该命令根据**HT**, **ESC \$**或**ESC \**对齐空白区域。

[缺省值]  $n = 0$

```
[例子] char SendStr[4];
SendStr[0] = 0x1B;
SendStr[1] = 0x61;
SendStr[2] = 0x01;
PrtSendData(SendStr,3);//设置水平对齐方式为居中
```

### 4.3.7 GS L nL nH

[名称] 设置左边距

[格式] ASCII码 GS L nL nH

十六进制码 1D 4C nL nH

十进制码 29 76 nL nH

[范围]  $0 \leq nL \leq 255$   $0 \leq nH \leq 255$

[描述] 左边距设置为N个 水平点距。nL nH 分别为无符号双字节整数的低位字节和高位字节， $N = nL + nH \times 256$ ，左边距为可打印区域左边距离打印区域宽度。

[注意] •该命令仅在一行的起始位置处理时有效。

• 左边距最大可设为336，如果超过336，则被当作336。

[缺省值]  $nL = 0, nH = 0$

```
[例子] char SendStr[4];
```

```
SendStr[0] = 0x1D;
```

```
SendStr[1] = 0x4C;
```

```
SendStr[2] = 0x10;
```

```
SendStr[3] = 0x00;
```

```
PrtSendData(SendStr,4);//设置左边距为16水平点距(2mm)
```

### 4.3.8 GS W

[名称] 设置打印区域宽度

[格式] ASCII GS W nL nH

Hex 1D 57 nL nH

Decimal 29 87 nL nH

[范围]  $0 \leq (nL + nH \times 256) \leq 65535$  ( $0 \leq nL \leq 255, 0 \leq nH \leq 255$ )

[描述] 在标准模式下，将打印区域宽度设置为 $(nL + nH \times 256) \times$ (横向运动单元)。

[注意] • 在标准模式下，此命令仅在行首有效。

- 此命令不影响页模式下的打印。如果此命令在页模式下执行，当打印机返回到标准模式时，打印区域宽度的设置有效。
- 如果[左边距 + 打印区宽度]超出可打印区域，则打印区域宽度为可打印区域宽度减去左边距。
- 如果此命令和GS L设置打印区域的宽度小于一个字符的宽度，打印区域的宽度会扩大，以适应行的一个字符。
- 使用横向运动单元。
- 打印区域的宽度设置好后，如果横向运动单元改变了，打印区域的宽度设置不会随着改变。
- 在执行ESC @，打印机复位或者关闭电源前，打印区域的宽度设置是有效的。

## 4.4 图形/图象打印命令

### 4.4.1 ESC \* m nL nH d1...dk

[名称] 打印黑白位图

[格式] ASCII码 ESC \* m nL nH d1...dk

十六进制码 1B 2A m nL nH d1...dk

十进制码 27 42 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

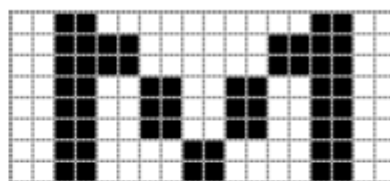
nL nH分别为无符号型双字节整数N的高位和低位字节，表示水平方向上位图中的点数。N在单倍宽时最大值为384，在双倍宽时其值最大为192。

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 d9

0	1	0	0	0	0	0	1	0	b7
0	1	1	0	0	0	1	1	0	b6
0	1	1	0	0	0	1	1	0	b5
0	1	0	1	0	1	0	1	0	b4
0	1	0	1	0	1	0	1	0	b3
0	1	0	1	0	1	0	1	0	b2
0	1	0	0	1	0	0	1	0	b1
0	1	0	0	1	0	0	1	0	b0



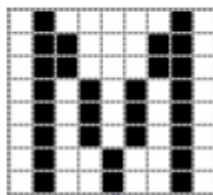
打印放大图

如要打印以上图象，程序代码如下

```
char SendStr[15];
SendStr[0] = 0x1B;
SendStr[1] = 0x2A;
SendStr[2] = 0x00; //m=0 (高度8点、倍宽)
SendStr[3] = 0x09; //图象宽度为9dots
SendStr[4] = 0x00;
SendStr[5] = 0x00; //图象点阵数据
SendStr[6] = 0xFF;
SendStr[7] = 0x60;
SendStr[8] = 0x1C;
SendStr[9] = 0x03;
SendStr[10] = 0x1C;
SendStr[11] = 0x60;
SendStr[12] = 0xFF;
SendStr[13] = 0x00;
SendStr[14] = 0x0a; // 走纸
PrtSendData(SendStr, 15); //打印图象
```

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	d9	
0	1	0	0	0	0	0	1	0	b7
0	1	1	0	0	0	1	1	0	b6
0	1	1	0	0	0	1	1	0	b5
0	1	0	1	0	1	0	1	0	b4
0	1	0	1	0	1	0	1	0	b3
0	1	0	1	0	1	0	1	0	b2
0	1	0	0	1	0	0	1	0	b1
0	1	0	0	1	0	0	1	0	b0



打印放大图

如要打印以上图象, 程序代码如下

```
char SendStr[15];
SendStr[0] = 0x1B;
SendStr[1] = 0x2A;
SendStr[2] = 0x01; //m=1 (高度8点、不放大)
SendStr[3] = 0x09; //图象宽度为9dots
SendStr[4] = 0x00;
SendStr[5] = 0x00; //图象点阵数据
```





```
SendStr[8] = 0x00; SendStr[9] = 0x00; SendStr[10] = 0x03;//2
SendStr[11] = 0x00; SendStr[12] = 0x00; SendStr[13] = 0xFE;//3
SendStr[14] = 0x00; SendStr[15] = 0x3F; SendStr[16] = 0xE0;//4
SendStr[17] = 0x03; SendStr[18] = 0xE0; SendStr[19] = 0x30;//5
SendStr[20] = 0x0E; SendStr[21] = 0x00; SendStr[22] = 0x18;//6
SendStr[23] = 0x11; SendStr[24] = 0x00; SendStr[25] = 0x08;//7
SendStr[26] = 0x20; SendStr[27] = 0xC0; SendStr[28] = 0x0C;//8
SendStr[29] = 0x40; SendStr[30] = 0xC0; SendStr[31] = 0x0C;//9
SendStr[32] = 0x80; SendStr[33] = 0xC0; SendStr[34] = 0x0C;//10
SendStr[35] = 0x80; SendStr[36] = 0x40; SendStr[37] = 0x1C;//11
SendStr[38] = 0x80; SendStr[39] = 0x60; SendStr[40] = 0x1C;//12
SendStr[41] = 0x80; SendStr[42] = 0xFF; SendStr[43] = 0xF8;//13
SendStr[44] = 0x43; SendStr[45] = 0x9F; SendStr[46] = 0xF0;//14
SendStr[47] = 0x7F; SendStr[48] = 0x07; SendStr[49] = 0xC0;//15
SendStr[50] = 0x3E; SendStr[51] = 0x00; SendStr[52] = 0x00;//16
SendStr[53] = 0x00; SendStr[54] = 0x00; SendStr[55] = 0x00;//17
SendStr[56] = 0x0a;
```

PrtSendData(SendStr,56); //打印图象

m=33(24点、单倍宽)d1、d2、d3表示打印的第1列点的数据，依此类推；bn表示字节的第n位，

	d4d7	d49
d1	0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0	b7
	0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0	b6
	0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 1 0	b5
	0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 0	b4
	0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0	b3
	0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0	b2
	0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 1 1 0	b1
	0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 1 1 0	b0
d2	0 0 0 0 1 0 0 1 1 1 0 0 0 1 1 0 0 0	b7
	0 0 0 0 1 0 0 1 1 1 1 1 1 1 0 0 0 0	b6
	0 0 0 1 1 1 0 0 0 0 0 0 0 1 1 0 0 0	b5
	0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0	b4
	0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0	b3
	0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0	b2
	0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0	b1
	0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0	b0
d3	0 0 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0	b7
	0 0 1 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0	b6
	0 0 1 1 1 0 0 0 0 0 0 0 1 1 1 0 0 0	b5
	0 0 1 0 1 1 0 0 0 0 0 1 1 1 1 0 0 0	b4
	0 0 1 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0	b3
	0 0 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0	b2
	0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	b1
	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	b0



打印放大图

如要打印以上图象，程序代码如下

```
char SendStr[64];
```

```
SendStr[0] = 0x1B;
SendStr[1] = 0x2A;
SendStr[2] = 0x21; //m=33 (高度24点、不放大)
SendStr[3] = 0x11; //图象宽度为17dots
SendStr[4] = 0x00;
//图象数据
```

```
SendStr[5] = 0x00; SendStr[6] = 0x00; SendStr[7] = 0x00; //1
SendStr[8] = 0x00; SendStr[9] = 0x00; SendStr[10] = 0x03; //2
SendStr[11] = 0x00; SendStr[12] = 0x00; SendStr[13] = 0xFE; //3
SendStr[14] = 0x00; SendStr[15] = 0x3F; SendStr[16] = 0xE0; //4
SendStr[17] = 0x03; SendStr[18] = 0xE0; SendStr[19] = 0x30; //5
SendStr[20] = 0x0E; SendStr[21] = 0x00; SendStr[22] = 0x18; //6
SendStr[23] = 0x11; SendStr[24] = 0x00; SendStr[25] = 0x08; //7
SendStr[26] = 0x20; SendStr[27] = 0xC0; SendStr[28] = 0x0C; //8
SendStr[29] = 0x40; SendStr[30] = 0xC0; SendStr[31] = 0x0C; //9
SendStr[32] = 0x80; SendStr[33] = 0xC0; SendStr[34] = 0x0C; //10
SendStr[35] = 0x80; SendStr[36] = 0x40; SendStr[37] = 0x1C; //11
SendStr[38] = 0x80; SendStr[39] = 0x60; SendStr[40] = 0x1C; //12
SendStr[41] = 0x80; SendStr[42] = 0xFF; SendStr[43] = 0xF8; //13
SendStr[44] = 0x43; SendStr[45] = 0x9F; SendStr[46] = 0xF0; //14
SendStr[47] = 0x7F; SendStr[48] = 0x07; SendStr[49] = 0xC0; //15
SendStr[50] = 0x3E; SendStr[51] = 0x00; SendStr[52] = 0x00; //16
SendStr[53] = 0x00; SendStr[54] = 0x00; SendStr[55] = 0x00; //17
PrtSendData(SendStr,56); //打印图象
```

[注意] • 如果m的值超出了指定的范围，可能出现不可预料的结果。

- 如果位图数据输入超出了一行上能被打印的点数，那么超出的数据被忽略。
- 在打印一个位图之后，打印机返回常规数据处理模式。
- 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。

#### 4.4.2 GS \* x y d1...dk

[名称] 定义下传位图

[格式] ASCII码 GS \* x y d1...dk

十六进制码 1D 2A x y d1...dk

十进制码 29 42 x y d1...dk

[范围]  $1 \leq x \leq 255$

$1 \leq y \leq 8$

$x ? y \leq 1536$

$0 \leq d \leq 255$

$k = x * y * 8$

[描述] 用x 和y 指定的点数定义下传位图。

- $x * 8$  为水平方向点数。

•  $y*8$  为垂直方向点数。

[注意]• 如果  $x * y$  超出了指定范围，则禁止该命令。

•  $d$  表示位图数据。数据( $d$ ) 指定打印位为1，不打印位为0。

• 用该命令定义的下传位图，通过  $GS / n$ 命令打印

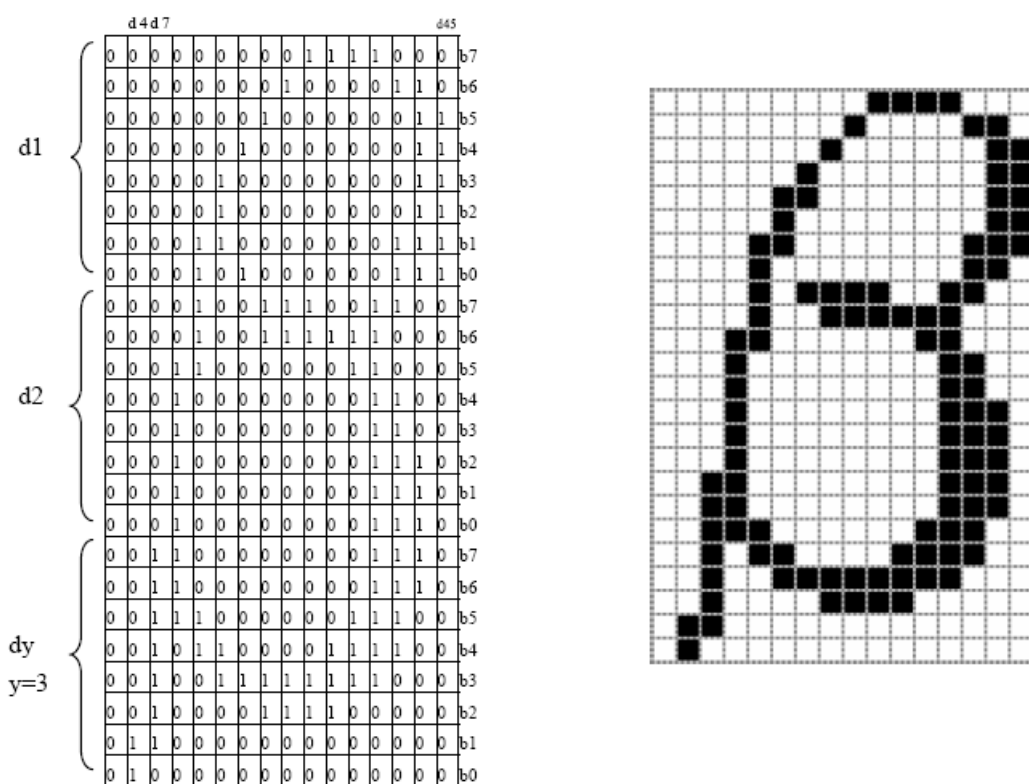
• 在下列情况下，清除下传位图定义：

1. 执行 **ESC @** 。

2. 打印机复位或关闭电源。

[例子] 下传位图与打印数据之间关系如下图例子所示。

如果  $x=2, y=3, d1\dots dk$  的值如下图，则位图如右图所示。



如要定义以上图象，程序代码如下

```
char SendStr[64];
SendStr[0] = 0x1D;
SendStr[1] = 0x02;//x=2(宽度16点)
SendStr[3] = 0x03;//y=3(高度24点)
//图象数据
SendStr[4] = 0x00; SendStr[5] = 0x00; SendStr[6] = 0x00;//1
SendStr[7] = 0x00; SendStr[8] = 0x00; SendStr[9] = 0x03;//2
SendStr[10] = 0x00; SendStr[11] = 0x00; SendStr[12] = 0xFE;//3
SendStr[13] = 0x00; SendStr[14] = 0x3F; SendStr[15] = 0xE0;//4
```

```
SendStr[16] = 0x03; SendStr[17] = 0xE0; SendStr[18] = 0x30;//5
SendStr[19] = 0x0E; SendStr[20] = 0x00; SendStr[21] = 0x18;//6
SendStr[22] = 0x11; SendStr[23] = 0x00; SendStr[24] = 0x08;//7
SendStr[25] = 0x20; SendStr[26] = 0xC0; SendStr[27] = 0x0C;//8
SendStr[28] = 0x40; SendStr[29] = 0xC0; SendStr[30] = 0x0C;//9
SendStr[31] = 0x80; SendStr[32] = 0xC0; SendStr[33] = 0x0C;//10
SendStr[34] = 0x80; SendStr[35] = 0x40; SendStr[36] = 0x1C;//11
SendStr[37] = 0x80; SendStr[38] = 0x60; SendStr[39] = 0x1C;//12
SendStr[40] = 0x80; SendStr[41] = 0xFF; SendStr[42] = 0xF8;//13
SendStr[43] = 0x43; SendStr[44] = 0x9F; SendStr[45] = 0xF0;//14
SendStr[46] = 0x7F; SendStr[47] = 0x07; SendStr[48] = 0xC0;//15
SendStr[49] = 0x3E; SendStr[50] = 0x00; SendStr[51] = 0x00;//16
```

PrtSendData(SendStr,52);//定义图象

SendStr[0] = 0x1D;

SendStr[1] = 0x2F;

SendStr[2] = 0x00;

PrtSendData(SendStr,3);//打印图象

#### 4.4.3 GS / n

[名称] 打印下传位图

[格式] ASCII码 GS / n

十六进制码 1D 2F n

十进制码 29 47 n

[范围]  $0 \leq n \leq 3$

[描述] 用n 指定的模式打印由GS \*命令定义的下传位图。

n从下表选择模式:

n	放大模式
0	普通
1	倍宽
2	倍高
3	倍宽倍高

[注意] • 如果位图数据未定义, 则忽略该命令。

- 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。
- 如果将要打印的下传位图超出可打印区域, 则超出的数据不打印。
- 如果由**GS L**设定的打印区域宽度小于图象宽度, 减少左边缘空白量以打印图形。
- 如果位图高度超出64点, 则忽略该命令, 如果用倍高模式打印, 则位图高度不能超过32点。

[参照] **GS \***

[例子] 参见 4.4.2 GS \* 例子

#### 4.4.4FS p n m

[名称] 打印下载到NV存储器中的位图

[格式]

ASCII	FS	p	n	m
Hex	1C	70	n	m
Decimal	28	112	n	m

[范围]  $1 \leq n \leq 2550 \leq m \leq 3, 48 \leq m \leq 51$

[描述] 以m指定的模式打印下载到NV存储器中的位图。

m	模式	纵向分辨率 (DPI)	横向分辨率 (DPI)
0.48	正常模式	203	203
1.49	倍宽模式	203	101
2.50	倍高模式	101	203
3.51	倍高倍宽模式	101	101

- n表示位图（由指令FS q定义）的图号。

- m指定打印位图的模式。

[细节] • NV位图是由指令FS q定义存储在NV存储器中并用指令FS p打印的位图。

- 当NV位图没有被定义时，该指令无效。

- 在标准模式下，该指令只有在打印缓冲区里没有数据时有效。

- 该指令除了受倒置打印模式的影响外，不受其他打印模式的影响（如：倍高、倍宽、下划线、字符放大、反显打印、顺时针旋转90度等）。

- 如果所要打印的下载位图超出当前打印区域，则不打印超出的部分。

- 如果由GS L 和 GS W设置的打印区域小于指令FS p所需要的打印宽度，以下动作会立即被执行（但不能超出最大的打印宽度）

- ①打印区域向右边扩展以适应打印位图的数据量；

- ②如果步骤①不能提供足够的扩展宽度，左边距也会相应减小以适应位图宽度；

- 在普通和倍宽模式下，该命令进纸n点，n为NV位图高度，在倍高和四倍大小模式下，该命令进纸2n点，n为NV位图高度，与ESC 2 或ESC 3 设置的行间距无关。

- 打印完位图后，打印机换行。

[参考] ESC \*, FS q, GS /, GS v 0

#### 4.4.5FS q n [xL xH yL yH d1...dk]1...[xL xH yL yH d1...dk]n

[名称] 定义NV位图

[格式]

FS	q	n [ xL xH yL yH d1...dk]...[ xL xH yL yH
Hex	1C	71 n [xL xH yL yH d1...dk]...[ xL xH yL yH
Decimal	28	113 n [xL xH yL yH d1...dk]...[ xL xH yL yH

[范围]  $1 < n < 255$  d1...dk]

 $0 \leq xL \leq$ 
 $1 \leq (xL + xH \times 256) \leq 1023$ 
 $1 \leq (yL + yH \times 256) \leq 800$ 
 $0 \leq d \leq 255$ 
 $k = (xL + xH \times 256) \times (yL + yH \times 256) \times 8$ 

NV存储器下载容量最大为64K字节



[描述] 定义NV位图:

- **n** 指定所要定义的NV位图的数目。
- **xL**、**xH**指定NV位图的横向点数为 $(xL + xH \times 256) \times 8$ 。
- **yL**、**yH**指定NV位图的纵向点数为 $(yL + yH \times 256) \times 8$ 。

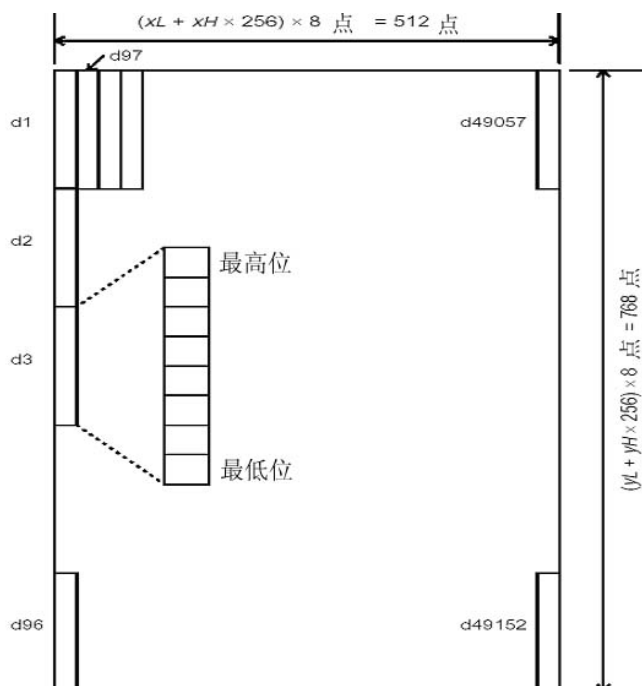
[注释]

- 频繁执行该指令能损坏NV存储器, 推荐每天最多写NV存储器10次。
- 该指令删除所有的以前由该指令定义的NV位图, 打印机不能对上一次定义的多幅位图中的一个进行重新定义, 在这种情况下, 必须重新发送所有的数据。
- 由于在处理该指令的过程中, 打印机处于忙状态, 它向NV存储器中写数据并且停止接收其他指令, 因此, 在该指令的执行过程中, 禁止向打印机发送其他命令, 包括实时指令。
- NV位图是由指令**FS q**定义存储在NV存储器中并用指令**FS p**打印的位图。
- 在标准模式下, 该指令仅在行首时有效。
- 从 **FS** 到 **yH** 的七个字节数据作为命令数据处理, 不是图形数据的一部分。
- 当位图数据字节数超出由它左边的**xL**、**xH**、**yL**、**yH**定义的范围, 则打印机只处理**xL**、**xH**、**yL**、**yH**定义的范围的数据。
- 在第一组NV位图里, 当**xL**、**xH**、**yL**、**yH**中任何一个参数超出定义的范围时, 该指令无效。
- 在下载多幅位图时, 如果打印机处理**xL**、**xH**、**yL**、**yH**超出定义的范围, 则打印机停止执行这条指令。命令中在此之后的位图无效, 在此之前的位图有效。
- **d**是定义的位图数据, 在数据中相应位为1表示打印该点, 为0表示不打印。
- 这条指令定义了 **n** 幅NV位图。每个位图的序列号从 1 依次增加, 因此, 第一个数据组 **[xL xH yL yH d1...dk]** 是NV位图 1 的数据, 最后一个数据组 **[xL xH yL yH d1...dk]** 是NV位图 **n** 的数据。在用**FS p** 指令打印位图时, 也是如此。
- 定义一幅NV位图的数据由 **[xL xH yL yH d1...dk]** 组成。因此, 当只有一幅位图 时, **n = 1**。打印机占用NV存储器字节数如下:  
[位图数据字节数:  $(xL + xH \times 256) \times (yL + yH \times 256) \times 8$ ] + [头信息: 4]
- 在打印机里NV存储器的下载空间最大为64K bytes, 该指令可以定义多幅NV位图, 但不能定义一幅大小超过40K bytes的位图不同的打印机下载空间也不相同, 请参照打印机配置信息)。
- 在写NV存储器之前, 打印机立即处于忙状态。
- 在处理该指令的过程中, 打印机不传送状态也不执行状态查询。
- 在宏定义的过程中接收到该指令, 打印机将结束宏定义, 开始执行该指令。
- 如果一幅NV位图被定义, 执行**ESC @**命令、复位和关闭电源不能将其擦除。
- 该指令只是定义NV位图, 不执行打印, 打印NV位图由**FS p**指令执行。
- 每幅NV位图在NV存储器内占用空间大小等于NV位图数据大小加上4字节。



[参考] **FS p**

[实例] 当  $xL = 64, xH = 0, yL = 96, yH = 0$



#### 4.4.6 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  
           Hex      1D      76      30      m      xL      xH      yL      yH      d1...      dk  
           Decimal      29      118      48      m      xL      Xh      yL      yH      d1.      ..dk

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

$0 \leq xL \leq 255$

$0 \leq xH \leq 255$

$0 \leq yL \leq 255$

$0 \leq d \leq 255$

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

[描述]打印光栅位图，由m值选择光栅位图模式

M	模式	纵向分辨率 (DPI)	横向分辨率 (DPI)
0, 48	正常	203DPI	203 DPI
1, 49	倍宽	203 DPI	101 DPI
2, 50	倍高	101 DPI	203 DPI
3, 51	倍宽、倍高	101 DPI	101 DPI

•  $xL, xH$ 表示水平方向位图字节数 ( $xL + xH \times 256$ )

•  $yL, yH$ 表示垂直方向位图点数 ( $yL + yH \times 256$ )

[注释] • 在标准模式下，只有打印机缓冲区无数据时该指令才有效。

- 字符放大、加粗、双重打印、倒置打印、下划线、黑白反显等打印模式对该指令无效。

- 位图超出打印区域的部分不打印。
- ESC a（选择对齐模式）对光栅位图有效。
- 宏定义的过程中，该命令将停止宏定义而执行该命令。该命令不作为宏定义的一部分。
- d 代表位图数据。每个字节的相应位为1表示打印该点，为0不打印该点。

[实例] 当  $xL + xH \times 256 = 64$

## 4.5 条码打印命令

### 4.5.1 GS h n

[名称] 设置条形码高度

[格式] ASCII码 GS h n

十六进制码 1D 68 n

十进制码 29 104 n

[范围]  $1 \leq n \leq 40$

[描述] 选择条形码高度。

n 设定垂直方向的点数。

[注意] • 如果  $n > 40$ , 条码高度将被设为40。

[缺省值]  $n = 36$

[参照] GS k

[例子] 参见 4.5.5 GS k 例子

### 4.5.2 GS w n

[名称] 设置条形码宽度

[格式] ASCII码 GS w n

十六进制码 1D 77 n

十进制码 29 119 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

• 以下是多级条形码:

UPC-A, UPC-E, EAN13, EAN8, CODE93

• 以下是二进制条形码:

CODE39, CODABAR

[缺省值]  $n = 2$

[参照] GS k

[例子] 参见 4.5.5 GS k 例子

### 4.5.3 GS H n

[名称] 选择条码可识读字符的打印位置

[格式] ASCII码 GS H n

十六进制码 1D 48 n

十进制码 29 72 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位置。



n 选择打印位置，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

[注意] • 用**GS f** 所指定的字体打印可识读字符。

• 条码可识读字符不受打印模式(粗体、重叠、下划线、字符大小、或反白打印)影响。

[缺省值]  $n = 0$

[参照] **GS f**, **GS k**

[例子] 参见 4.5.5 GS k 例子

### 4.5.4 GS f n

[名称] 选择条码识读字符字体

[格式] ASCII码 GS f n

十六进制码 1D 66 n

十进制码 29 102 n

[范围]  $n = 0, 1$

[描述] 打印条形码时，为识读字符选择一种字体。

n 的可选值及意义

n	识读字符字体
0	字体A (12 * 24)
1	字体B (8 * 16)

[注意] • 在由**GS H** 指定的位置打印条码识读字符。

• 条码识读字符不受打印模式(粗体、重叠、下划线、字符大小、或反白打印)影响。

[缺省值]  $n = 0$

[参照] **GS H**, **GS k**

[例子] 参见 4.5.5 GS k 例子

### 4.5.5 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

十六进制码 1D 6B m d1...dk 00

十进制码 29 107 m d1...dk 0

格式②( $65 \leq m \leq 73$ )

ASCII码 GS k m n d1...dn

十六进制码 1D 6B m n d1...dn

十进制码 29 107 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	字符个数	d
① 0	UPC-A	$11 \leq k \leq 12$	$48 \leq d \leq 57$
1	UPC-E	$7 \leq k \leq 8$	$48 \leq d \leq 57$
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	$1 \leq k \leq 255$	$45 \leq d \leq 57, 65 \leq d \leq 90, 32, 36, 37, 43$
5	ITF	$1 \leq k \leq 255$	$48 \leq d \leq 57$
6	CODABAR	$1 \leq k \leq 255$	$48 \leq d \leq 57, 65 \leq d \leq 68, 36, 43, 45, 46, 47, 58$
7	CODE93	$1 \leq n \leq 255$	$0 \leq d \leq 127$
8	CODE128	$2 \leq n \leq 255$	$0 \leq d \leq 127$
② 65	UPC-A	$11 \leq n \leq 12$	$48 \leq d \leq 57$
66	UPC-E	$11 \leq n \leq 12$	$48 \leq d \leq 57$
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	$1 \leq n \leq 255$	$45 \leq d \leq 57, 65 \leq d \leq 90, 32, 36, 37, 43$
			$d1 = dk = 42$
70	ITF	$1 \leq n \leq 255$	$48 \leq d \leq 57$
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	$2 \leq n \leq 255$	$0 \leq d \leq 127$

[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）应当等于规定的的数据长度，如果不等于规定的的数据长度，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A。

• 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。
- 如果可识读字符被设置成打印, **CODE93**和**CODE128**码的中的不可见字符不能打印出来, 此时用‘□’取代。
- **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

```
[例子] char SendStr[16];
SendStr[0] = 0x1D;
SendStr[1] = 0x68;
SendStr[2] = 0x28;
PrtSendData(SendStr,3);//设置条码高度为40垂直点距(5mm)
SendStr[0] = 0x1D;
SendStr[1] = 0x77;
SendStr[2] = 0x02;
PrtSendData(SendStr,3);//设置条码宽度为2
SendStr[0] = 0x1D;
SendStr[1] = 0x48;
SendStr[2] = 0x02;
PrtSendData(SendStr,3);//设置条码可识读字符位置为条码下方
SendStr[0] = 0x1D;
SendStr[1] = 0x66;
SendStr[2] = 0x00;
PrtSendData(SendStr,3);//设置条码可识读字符为12*24点阵字符
SendStr[0] = 0x1D;
```

```
SendStr[1] = 0x6B;
SendStr[2] = 0x04;
SendStr[3] = 'T'; SendStr[4] = 'E' ;
SendStr[5] = 'S'; SendStr[6] = 'T'; SendStr[7] = '8' ;
SendStr[8] = '0'; SendStr[9] = '5'; SendStr[10] = '2' ;
SendStr[11] = 0;
PrtSendData(SendStr,12); //用格式1打印CODE39码“TEST8052” 用户不必自行添加‘*’
```

#### 4.5.6 GS k m v r d1.....dk [NULL]

[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
十六进制码 1D 6B  m v r d1...dk 00
十进制码   29 107 m v r d1...dk 0
```

格式2 m=97

```
ASCII码    GS k    m v r nL nH d1...dn
十六进制码 1D 6B  m v r nL nH d1...dn
十进制码   29 107 m v r nL nH d1...dn
```

[范围] m=32或97

$1 \leq v \leq 17$      $1 \leq r \leq 4$

[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] char SendStr[16];

```
SendStr[0] = 0x1D; SendStr[1] = 'k'; SendStr[2] = 32;
SendStr[3] = 1; //版本为1
SendStr[4] = 2; //纠错等级为M
PrtSendData( SendStr, 5);
strcpy(SendStr, "123456789");
PrtSendData( SendStr, 10);
```

### 4.5.7GS ( k <Function 167>

[名称] QR码：设置模块大小

[格式] ASCII GS ( k pL pH cn fn n  
Hex 1D 28 6B 03 00 31 43 n  
Decimal 29 40 107 3 0 49 67 n

[范围]  $(pL + pH \times 256) = 3$  ( $pL = 3, pH = 0$ )  
cn = 49  
fn = 67

[描述] 设置QR码为n点的模块大小。

[注意] •此功能的设置会影响到功能181和182。  
•设置单位取决于打印机的型号。  
•在执行**ESC @**，打印机复位或者关闭电源前，此功能的设置是有效的。  
•n = 模块宽度 = 模块高度(因为QR码模块是正方形的)。

### 4.5.8GS ( k <Function 169>

[名称] QR码：选择纠错水平

[格式] ASCII GS ( k pL pH cn fn n  
Hex 1D 28 6B 03 00 31 43 n  
Decimal 29 40 107 3 0 49 69 n

[范围]  $(pL + pH \times 256) = 3$  ( $pL = 3, pH = 0$ )  
cn = 49  
fn = 69  
 $48 \leq n \leq 51$

[默认值] n = 48

[描述] 选择QR码的纠错水平。

n	功能	恢复能力% (大约)
48	选择纠错水平L	7
49	选择纠错水平M	15
50	选择纠错水平Q	25
51	选择纠错水平H	30

[注意] •此功能的设置会影响到功能181和182。  
•QR码运用RS错误校正生成一系列的纠错编码词。  
•在执行**ESC @**，打印机复位或者关闭电源前，此功能的设置是有效的。

### 4.5.9GS ( k <Function 180>

[名称] QR码：在符号储存区存储数据

[格式] ASCII GS ( k pL pH cn fn m d1...dk  
Hex 1D 28 6B pL pH 31 50 30 d1...dk  
Decimal 29 40 107 pL pH 49 80 48 d1...dk

[范围]  $4 \leq (pL + pH \times 256) \leq 7092$  ( $0 \leq pL \leq 255, 0 \leq pH \leq 27$ )  
cn = 49  
fn = 80



$m = 48$

$0 \leq d \leq 255$

$k = (pL + pH \times 256) - 3$

[描述] 在符号存储区存储QR码符号数据（d1... dk）。

[注意] ★储存区的数据储存是通过功能181和182处理的。在执行功能181和182后，符号储存区的数据才被保留。

★d1... dk的k字节符号数据处理。

★以下是一个QR码编码，除了以下d1... dk数据外， 不包括任何数据。

数据类别	字符
数值模式	“0” ~ “9”
字母数字模式	“0” ~ “9”，“A” ~ “Z”，SP，\$，%， *，+，-，.，/，:
汉字模式	JIS 值转移（从 JISX0208 移值）
8位字节模式	00H ~ FFH

★此功能的设置是有效的，直到执行以下处理：

- 执行功能080或180或280
- 执行ESC @
- 打印机复位或者关闭电源

#### 4.5.10GS ( k <Function 181>

[名称] QR码：在符号储存区打印符号数据

[格式]

ASCII	GS	(	k	pL	pH	cn	fn	m
Hex	1D	28	6B	03	00	31	51	m
Decimal	29	40	107	3	0	49	81	m

[范围]  $(pL + pH \times 256) = 3$  ( $pL = 3$ ,  $pH = 0$ )

$cn = 49$

$fn = 81$

$m = 48$

[描述] 在符号储存区编码和打印QR码符号数据使用<功能180>的过程。

[注意] ★在标准模式中，当打印机“在行首”或是“在打印缓冲区无数据”时使用此功能。

★符号的大小超过打印区域就不能被打印出来。

★如果在符号储存区的数据有任何的错误描述就不能被打印出来。

- 无数据（不处理功能180）。
- 如果符号存储区的数据超过指定的模型和数据压缩模式所允许的数据。（这种情况下是一个不寻常的数据。）
- 下面列出的四种数据压缩模式（以压实率）。自动选择最佳的符号存储区的数据压缩模式。
  - 数值模式
  - 字母数字模式
  - 汉字模式
  - 8位字节模式

★以下数据自动添加编码处理。

- 位置检测模式
  - 分离器位置检测模式
  - 时序模式
  - 格式信息
  - 版本信息
  - 错误校正码（采用RS错误检测和校正算法）
  - 垫码
  - 字符计数指示器中的数位
  - 模式指示灯
  - 终止码
  - 对齐模式（模式2时选定）
  - 扩展模式（模式1时选定）
- ★该命令除了受字符的大小和倒置打印模式的影响外，不受其他打印模式的影响（粗体，双重突出打印，下划线，白色/黑色反向印刷，或顺时针旋转90°）。
- ★在标准模式下，无论纸设置命令设置的进纸量是多少，此命令执行印刷符号进纸量所需的数额。在打印符号后，打印位置返回到可打印区域的左侧，打印机处于“开始线”或“打印缓冲区无数据”的状态。
- ★在页面模式下，打印机将符号数据储存在打印缓冲区中而不执行实际印刷。打印机将打印位置移动到最后一个符号数据的下一个点上。
- ★安静的区域不包括在打印数据中。使用此功能时，一定要包括安静的区域。

## 4.6 曲线打印命令

### 4.6.1 GS ‘

[名称] 打印一水平上n个线段

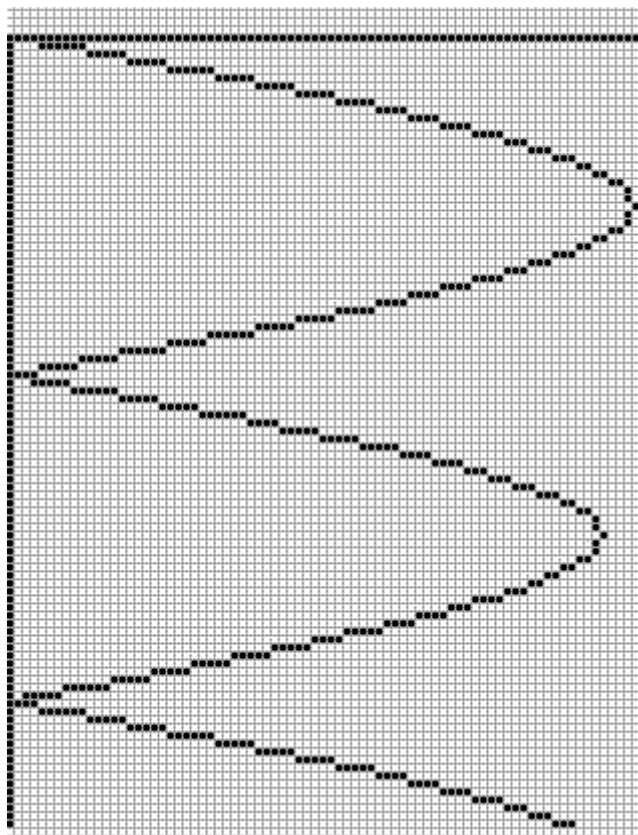
[格式] ASCII码 GS ‘ *n xlsL xlsH xleL xleH ... xnsL xnsH xneL xneH*

十六进制码 1D 27 *n xlsL xlsH xleL xleH ... xnsL xnsH xneL xneH*

十进制码 29 39 *n xlsL xlsH xleL xleH ... xnsL xnsH xneL xneH*

[范围]  $0 \leq n \leq 8$

[描述] 如下打印放大图所示：每条曲线都是由很多水平线段(点可视为长度为1的线段)组成。本指令为打印一水平上n个线段，连续使用该指令可以打印出用户所需要的线段。



xksL 第k条线段起始点横向坐标的低位；

xksH 第k条线段起始点横向坐标的高位；

xkeL 第k条线段结束点横向坐标的低位；

xkeH 第k条线段结束点横向坐标的高位；

坐标从打印区域最左侧开始计算，最小值为0，最大值为383，也就是说 $xkeL + xkeH * 256$ 最大值为383。

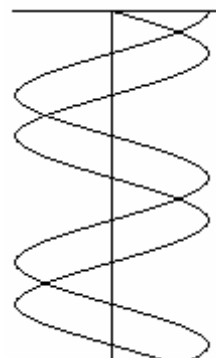
线段的数据不必按照顺序排列；

[注意] • 当打印一个点时， $xkeL = xksL$ ， $xkeH = xksH$ 。

[参照] GS “

[例子] 打印sin,cos函数的连续曲线，打印结果如右图：

```
char SendStr[8];
char SendStr2[16];
float i;
short y1,y2,y1s,y2s;
//打印y轴轴线(一条线)
SendStr[0] = 0x1D;
SendStr[1] = 0x27;
SendStr[2] = 1; //一条线段
SendStr[3] = 30;
SendStr[4] = 0; //起始点为
SendStr[5] = 104;
SendStr[6] = 1; //结束点
PrtSendData(SendStr,7);
```



打印图例

```
//打印曲线
SendStr[0] = 0x1D;
SendStr[1] = 0x27;
SendStr[2] = 3; //三条线段，分别为x坐标轴，sin函数和cos函数曲线
SendStr[3] = 180; SendStr[4] = 0; //x坐标轴位置为
SendStr[5] = 180; SendStr[6] = 0;
for(i=1;i<1200;i++){
    y1 = sin(i/180*3.1416)*(380-30)/2+180; //计算sin函数坐标点
    y2 = cos(i/180*3.1416)*(380-30)/2+180; //计算cos函数坐标点
    if(i==1) { y1s = y1; y2s = y2; }
    PrtSendData( SendStr, 7 );

    if(y1s < y1)
    {
        PrtSendData( &y1s, 2 ); //sin函数曲线位于该行的线段的起始点
        PrtSendData( &y1, 2 ); //sin函数曲线位于该行的线段的结束点
    }
    else
    {
        PrtSendData( &y1, 2 ); //sin函数曲线位于该行的线段的起始点
        PrtSendData( &y1s, 2 ); //sin函数曲线位于该行的线段的结束点
    }
    if(y2s < y2)
    {
        PrtSendData( &y2s, 2 ); //cos函数曲线位于该行的线段的起始点
        PrtSendData( &y2, 2 ); //cos函数曲线位于该行的线段的结束点
    }
}
```

```
else
{
    PrtSendData( &y2, 2 ); //cos函数曲线位于该行的线段的起始点
    PrtSendData( &y2s, 2 ); //cos函数曲线位于该行的线段的结束点
}

y1s = y1; //设置下一行打印时sin函数曲线位于该行的起始点坐标
y2s = y2; //设置下一行打印时cos函数曲线位于该行的起始点坐标
}
```

## 4.7 自定义字符命令

### 4.7.1 ESC % $n$

[名称] 选择/取消用户自定义字符集

[格式] ASCII码 ESC %  $n$

十六进制码 1B 25  $n$

十进制码 27 37  $n$

[范围]  $0 \leq n \leq 255$

[描述] 选择或取消用户自定义字符集。

当 $n$ 的最低位(LSB)为0时, 取消用户自定义字符集。

当 $n$ 的最低位(LSB)为1时, 选择用户自定义字符集。

[注意] • 当取消用户自定义字符集时, 自动选择内部字符集。

•  $n$  仅最低有效位可用。

[缺省值]  $n = 0$

[参照] ESC &, ESC ?

[例子] char SendStr[4];

SendStr[0] = 0x1B;

SendStr[1] = 0x25;

SendStr[2] = 0x01;

PrtSendData(SendStr,3);//选择用户自定义字符集

### 4.7.2 ESC & y c1 c2 [x1 d1...d(y \* x1)]...[xk d1...d(y \* xk)]

[名称] 定义用户自定义字符

[格式] ASCII码 ESC & y c1 c2 [x1 d1...d(y \* x1)]...[xk d1...d(y \* xk)]

十六进制码 1B 26 y c1 c2 [x1 d1...d(y \* x1)]...[xk d1...d(y \* xk)]

十进制码 27 38 y c1 c2 [x1 d1...d(y \* x1)]...[xk d1...d(y \* xk)]

[范围]  $y = 3$  字体A ( $12 * 24$ )

$y = 2$  字体B ( $8 * 16$ )

$32 \leq c1 \leq c2 \leq 126$

$x = 12$  字体A ( $12 * 24$ )

$x = 8$  字体B ( $8 * 16$ )

$0 \leq d1 \dots d(y * xk) \leq 255$

[描述] 定义用户自定义字符。

□  $y$  指定垂直方向字节数,

当定义字体A的字符时 $y=3$ ,

当定义字体B的字符时 $y=2$ ,

•  $c1$  指定起始字符编码,  $c2$  指定结束字符编码。

•  $x$  指定水平方向点数,

当定义字体A的字符时 $x=12$ ,

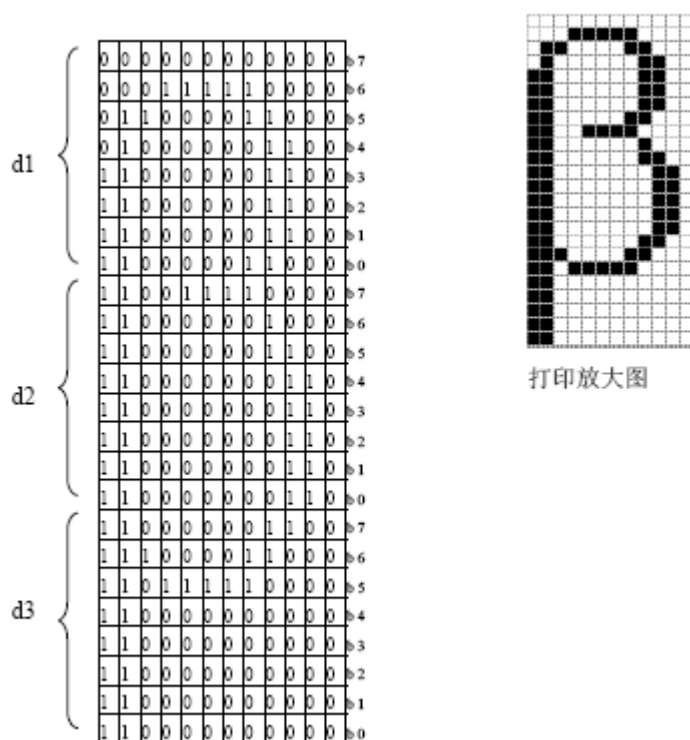
当定义字体B的字符时 $y=8$ ,

•  $k=c2-c1+1$

[注意] • 可定义字符编码的范围: 从0x20到0x7E的ASCII 码(95 字符)。

- 可定义多个字符的连续字符编码。当仅需要一个字符时，令  $c1 = c2$ 。
- $d$  是字符的点阵数据。
- 定义用户自定义字符的数据是  $(y * x)$  字节。
- 设定打印点的相应位为1或不打印点的相应位为0。
- 该命令可对每一种字体定义不同的用户自定义字符模式。用ESC ! 设定字体。
- 用户自定义字符和下传位图总数据量不超过32k。
- 在下列情况下，用户自定义字符被清除：
  1. 执行**ESC @**。
  2. 执行**ESC ?**。
  3. 打印机复位或关闭电源。

[例子]  $y=3, x=12$  将字符'B' (编码为0x42) 定义为如下12\*24点阵字符



```
char SendStr[42];
SendStr[0] = 0x1B;
SendStr[1] = 0x26;
SendStr[2] = 0x03; //y=3 字符高度为24
SendStr[3] = 0x42; // 字符'B' (编码0x42) 被重新定义
SendStr[4] = 0x42;
SendStr[5] = 0x0C; //x=12 字符宽度为12
SendStr[6] = 0x0F; SendStr[7] = 0xFF; SendStr[8] = 0xFF;//1
SendStr[9] = 0x3F; SendStr[10] = 0xFF; SendStr[11] = 0xFF;//2
SendStr[12] = 0x20; SendStr[13] = 0x00; SendStr[14] = 0x40;//3
SendStr[15] = 0x40; SendStr[16] = 0x00; SendStr[17] = 0x20;//4
SendStr[18] = 0x40; SendStr[19] = 0x80; SendStr[20] = 0x20;//5
```

```
SendStr[21] = 0x40; SendStr[22] = 0x80; SendStr[23] = 0x20;//6
SendStr[24] = 0x40; SendStr[25] = 0x80; SendStr[26] = 0x20;//7
SendStr[27] = 0x61; SendStr[28] = 0x80; SendStr[29] = 0x60;//8
SendStr[30] = 0x3F; SendStr[31] = 0x60; SendStr[32] = 0xC0;//9
SendStr[33] = 0x1E; SendStr[34] = 0x3F; SendStr[35] = 0x80;//10
SendStr[36] = 0x00; SendStr[37] = 0x1F; SendStr[38] = 0x00;//11
SendStr[39] = 0x00; SendStr[40] = 0x00; SendStr[41] = 0x00;//12
```

PrtSendData(SendStr, 42); //选择用户自定义字符集

[缺省值] 内部字符集

[参照] **ESC %**, **ESC ?**, **FS 2**

### 4.7.3 ESC ?

[命令] 清除用户自定义字符

[格式] ASCII码 **ESC ? n**

十六进制码 **1B 3F n**

十进制码 **27 63 n**

[范围]  $32 \leq n \leq 126$

[描述] 清除用户自定义字符。

[注意] • 该命令清除指定的用户自定义字符，字符编码由n 指定。在用户自定义字符被取消后，以内部字符相应模式打印。

• 如果没有为指定字符代码定义一个用户自定义字符，则打印机忽略该命令。

[参照] **ESC &**, **ESC %**

[例子] char SendStr[4];

SendStr[0] = 0x1B;

SendStr[1] = 0x3F;

SendStr[2] = 0x42;

PrtSendData(SendStr, 3); //清除为0x42定义的字符



## 4.8 汉字命令

### 4.8.1 FS &

[名称] 设定汉字模式

[格式] ASCII码 FS &

十六进制码 1C 26

十进制码 28 38

[描述] 选择汉字字符模式。

[注意] • 打开电源时，打印机已选择汉字模式。

[参照] FS ., FS C

[例子] char SendStr[4];

SendStr[0] = 0x1C;

SendStr[1] = 0x26;

PrtSendData(SendStr,2);//设定汉字模式

### 4.8.2 FS 2 c1 c2 d1...dk

[名称] 定义用户自定义汉字字符

[格式] ASCII码 FS 2 c1 c2 d1...dk

十六进制码 1C 32 c1 c2 d1...dk

十进制码 28 50 c1 c2 d1...dk

[范围] c1 和c2 表示所定义的字符的代码。

$c1 = FEH A1H \leq c2 \leq FEH$

$0 \leq d \leq 255$

$k = 72$

[描述] 定义用户自定义汉字字符，由c1 和c2 指定字符代码。

[注意] • c1 和c2 表示所定义的字符的代码。c1 为第一个字节， c2 为第二字节。

• d 表示该字符的点阵数据。相应位置为1，打出该点，相应位为0，不打该点。

定义数据的格式与ESC &指令相同

• 当用户选择字体A时，定义的汉字应为24点阵汉字，选择字体B时，定义的汉字应为16点阵汉字

[缺省值] 全空白。

[参照] ESC &

### 4.8.3 FS .

[名称] 取消汉字字符模式

[格式] ASCII码 FS .

十六进制码 1C 2E

十进制码 28 46

[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。

[注意] • 打开电源时，打印机选择汉字模式。

[参照] FS &, FS C

[例子] char SendStr[4];

---

```
SendStr[0] = 0x1C;  
SendStr[1] = 0x2E;  
PrtSendData(SendStr,2);//取消汉字模式
```

## 附录

### A. 打印字符集

本打印字符集0x80及之后的编码为取消汉字打印模式下打印出的字符。有关汉字字符，请参见国标GB-2312和微软代码页CP936。

HE X		HE X		HE X		HE X		HE X		HE X		HE X		HE X	
20	(空格)	21	!	22	“	23	#	24	\$	25	%	26	&	27	‘
28	(	29	)	2A	*	2B	+	2C	,	2D	-	2E	.	2F	/
30	0	31	1	32	2	33	3	34	4	35	5	36	6	37	7
38	8	39	9	3A	:	3B	;	3C	<	3D	=	3E	>	3F	?
40	@	41	A	42	B	43	C	44	D	45	E	46	F	47	G
48	H	49	I	4A	J	4B	K	4C	L	4D	M	4E	N	4F	O
50	P	51	Q	52	R	53	S	54	T	55	U	56	V	57	W
58	X	59	Y	5A	Z	5B	[	5C	\	5D	]	5E	^	5F	_
60	`	61	a	62	b	63	c	64	c	65	e	66	f	67	g
68	h	69	i	6A	j	6B	k	6C	l	6D	m	6E	n	6F	o
70	p	71	q	72	r	73	s	74	t	75	u	76	v	77	w
78	x	79	y	7A	z	7B	{	7C		7D	}	7E	~	7F	
80	Ç	81	ü	82	é	83	â	84	ä	85	à	86	å	87	ç
88	ê	89	ë	8A	è	8B	ï	8C	î	8D	ì	8E	Ä	8F	Å
90	É	91	æ	92	Æ	93	ô	94	ö	95	ò	96	û	97	ù
98	ÿ	99	Ö	9A	Ü	9B	ø	9C	£	9D	¥	9E	₣	9F	ƒ
A0	á	A1	í	A2	ó	A3	ú	A4	ñ	A5	Ñ	A6	ª	A7	º
A8	¿	A9	¬	AA	¬	AB	½	AC	¼	AD	¡	AE	«	AF	»
B0	⌘	B1	⌘	B2	⌘	B3		B4	⌘	B5	⌘	B6	⌘	B7	⌘
B8	⌘	B9	⌘	BA	⌘	BB	⌘	BC	⌘	BD	⌘	BE	⌘	BF	⌘
C0	⌘	C1	⌘	C2	⌘	C3	⌘	C4	⌘	C5	⌘	C6	⌘	C7	⌘
C8	⌘	C9	⌘	CA	⌘	CB	⌘	CC	⌘	CD	=	CE	⌘	CF	⌘
D0	⌘	D1	⌘	D2	⌘	D3	⌘	D4	⌘	D5	⌘	D6	⌘	D7	⌘
D8	⌘	D9	⌘	DA	⌘	DB	■	DC	■	DD	■	DE	■	DF	■
E0	α	E1	β	E2	γ	E3	Π	E4	Σ	E5	σ	E6	μ	E7	γ
E8	Φ	E9	θ	EA	Ω	EB	δ	EC	∞	ED	φ	EE	€	EF	∩
F0	≡	F1	±	F2	≥	F3	≤	F4	∫	F5	∫	F6	÷	F7	≈
F8	°	F9	•	FA	•	FB	√	FC	ⁿ	FD	²	FE	▪	FF	

## B. 条码

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'\*', 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'\*', 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。

## C. 预印刷黑标说明

如果用户使用预印刷黑标进行票据定位, 印刷黑标时务必遵守以下黑标预印刷规范, 否则可能导致打印机无法识别黑标。

印刷位置: 如左图所示, 黑标应印刷于文字面的右侧边缘。

宽度范围: 宽度 $\geq 7\text{mm}$

高度范围:  $4\text{mm} \leq \text{高度} \leq 6\text{mm}$

对红外光的反射率:  $<10\%$  (纸张黑标宽度其他部分对于红外光的反射率 $>65\%$ )

Hps: Hps为打印机黑标下边缘距打印起始上边缘的距离。

$0\text{mm} \leq \text{Hps} \leq 1\text{mm}$

