

zyPrtLib 微打驱动库

微打系列产品

AN01010101

V1.00

Date: 2011/05/04

使用说明

类别	内容
关键词	zyPrtLib、驱动库、微打
摘 要	广州致远电子有限公司推出了多款热敏微打产品，该产品功能完善，支持三十多条常用 ESC/POS 指令，使用户快速完成产品开发，占领市场。我司为提高用户开发速度，提供 zyPrtLib 驱动库，封装各个 ESC/POS 命令，供用户开发应用程序。本文详细介绍库函数

修订历史

版本	日期	原因
V1.00	2011/05/04	创建文档

销售与服务网络（一）

广州周立功单片机发展有限公司

地址：广州市天河北路 689 号光大银行大厦 12 楼 F4
邮编：510630
电话：(020)38730916 38730917 38730972 38730976 38730977
传真：(020)38730925
网址：www.zlgmcu.com



广州专卖店

地址：广州市天河区新赛格电子城 203-204 室
电话：(020)87578634 87569917
传真：(020)87578842

南京周立功

地址：南京市珠江路 280 号珠江大厦 1501 室
电话：(025) 68123901 68123902
传真：(025) 68123900

北京周立功

地址：北京市海淀区知春路 113 号银网中心 A 座
1207-1208 室（中发电子市场斜对面）
电话：(010)62536178 62536179 82628073
传真：(010)82614433

重庆周立功

地址：重庆市石桥铺科园一路二号大西洋国际大厦
（赛格电子市场）1611 室
电话：(023)68796438 68796439
传真：(023)68796439

杭州周立功

地址：杭州市天目山路 217 号江南电子大厦 502 室
电话：(0571)89719480 89719481 89719482
89719483 89719484 89719485
传真：(0571)89719494

成都周立功

地址：成都市一环路南二段 1 号数码科技大厦 403 室
电话：(028)85439836 85437446
传真：(028)85437896

深圳周立功

地址：深圳市深南中路 2070 号电子科技大厦 C 座 4 楼 D 室
电话：(0755)83781788（5 线）
传真：(0755)83793285

武汉周立功

地址：武汉市洪山区广埠屯珞瑜路 158 号 12128 室
（华中电脑数码市场）
电话：(027)87168497 87168297 87168397
传真：(027)87163755

上海周立功

地址：上海市北京东路 668 号科技京城东座 7E 室
电话：(021)53083452 53083453 53083496
传真：(021)53083491

西安办事处

地址：西安市长安北路 54 号太平洋大厦 1201 室
电话：(029)87881296 83063000 87881295
传真：(029)87880865

销售与服务网络（二）

广州致远电子有限公司

地址：广州市天河区车陂路黄洲工业区3栋2楼

邮编：510660

传真：(020)38601859

网址：www.embedtools.com （嵌入式系统事业部）

www.embedcontrol.com （工控网络事业部）

www.ecardsys.com （楼宇自动化事业部）



技术支持：

CAN-bus：

电话：(020)22644381 22644382 22644253

邮箱：can.support@embedcontrol.com

iCAN 及数据采集：

电话：(020)28872344 22644373

邮箱：ican@embedcontrol.com

MiniARM：

电话：(020)28872684 28267813

邮箱：miniarm.support@embedtools.com

以太网：

电话：(020)22644380 22644385

邮箱：ethernet.support@embedcontrol.com

无线通讯：

电话：(020) 22644386

邮箱：wireless@embedcontrol.com

串行通讯：

电话：(020)28267800 22644385

邮箱：serial@embedcontrol.com

编程器：

电话：(020)22644371

邮箱：programmer@embedtools.com

分析仪器：

电话：(020)22644375 28872624 28872345

邮箱：tools@embedtools.com

ARM 嵌入式系统：

电话：(020) 22644383 22644384

邮箱：NXPARM@zlgmcu.com

楼宇自动化：

电话：(020)22644376 22644389 28267806

邮箱：mjs.support@ecardsys.com

mifare.support@zlgmcu.com

销售：

电话：(020)22644249 22644399 22644372 22644261 28872524

28872342 28872349 28872569 28872573 38601786

维修：

电话：(020)22644245

目 录

1. 驱动库函数.....	1
2. 函数详解.....	3
3. 免责声明.....	22

1. 驱动库函数

zyPrtLib微打驱动库函数列表如表 1.1所示。

表 1.1 驱动库函数

函数	功能	详细描述
uartSendStr	传送文字	见表 2.1
print	打印并进纸	见表 2.2
printAndFeedNDotLine	打印并进纸 n 点	见表 2.3
printAndBackFeedNDotLine	打印并退纸 n 点	见表 2.4
printAndFeedNFontLine	打印并进纸 n 行	见表 2.5
printAndBackFeedNFontLine	打印并退纸 n 行	见表 2.6
lineSpacingSet	设置行间距为 n 点	见表 2.7
lineSpacingDefault	设置行间距为默认值	见表 2.8
leftMarginSet	设置左边距	见表 2.9
rightMarginSet	设置右边距	见表 2.10
abscissaSet	设置打印位置	见表 2.11
fontTypeSet	设置字符打印方式	见表 2.12
horizontalAlign	设置打印对齐方式	见表 2.13
fontGrayscaleSet	设置打印灰度	见表 2.14
printSpeedSet	设置打印速度	见表 2.15
fontSizeSet	设置打印字体	见表 2.16
picturePrintV	填充图形垂直取模数据	见表 2.17
picturePrintH	打印图片水平取模数据	见表 2.18
horizontalTabSet	设置水平制表位置	见表 2.19
horizontalTab	水平制表	见表 2.20
formPrintV	打印 90° 表格	见表 2.21
barcodeHeightSet	设置一维条码高度	见表 2.22
barcodeWidthSet	设置一维条码宽度	见表 2.23
barcodeHriPosSet	设置一维条码可读字符（HRI）打印位置	见表 2.24
barcodeHriFontSet	设置一维条码字体	见表 2.25
barcodePrint	打印一维条码	见表 2.26
printerInit	初始化打印机	见表 2.27
printerClearBuffer	清打印缓存（实时）	见表 2.28
printerCutPaper	进纸并切纸	见表 2.29
printerStateRT	查询打印机状态（实时）	见表 2.30
printerStateAutoUpload	设置/取消打印机状态自动返回	见表 2.31
printerBaudRateSet	设置打印机波特率	见表 2.32
printerFlowCtrlSet	设置流控制方式	见表 2.33
printerEnterLowPowerMode	进入低功耗模式	见表 2.34
printerExitLowPowerMode	退出低功耗模式	见表 2.35

续上表

printerOverHeatHandle	打印机过热处理入口	见表 2.36
printerOverHeatResumeHandle	打印机过热恢复处理入口	见表 2.37
printerOfPaperHandle	打印机缺纸处理入口	见表 2.38
printerOfPaperResumeHandle	打印机缺纸恢复处理入口	见表 2.39
printerOverVoltageHandle	打印机过压处理入口	见表 2.40
printerOverVoltageResumeHandle	打印机过压恢复处理入口	见表 2.41
printerAxoOpenHandle	打印机压轴打开处理入口	见表 2.42
printerAxoCloseHandle	打印机压轴关闭处理入口	见表 2.43
printerCutterOnHandle	切刀移开处理入口	见表 2.44
printerCutterOffHandle	切刀关闭处理入口	见表 2.45

2. 函数详解

zyPrtLib微打驱动器函数详解见表 2.1~表 2.45。

表 2.1 uartSendStr

功能名称	传送字符串
原型	void uartSendStr (char *pucStr)
参数	pucStr: 字符串（以'\0'结束）
返回	无
支持型号	所有型号
说明	传送打印的文字信息 若所传的文字信息填满微打内部打印缓存，满行部分自动打印并进纸

表 2.2 print

功能	打印并进纸
原型	void print (void)
参数	无
返回	无
支持型号	所有型号
说明	将打印缓存里的内容打印，之后根据当前的行间距设置进纸一行，并调整打印位置至下一行的起始位置

表 2.3 printAndFeedNDotLine

功能	打印并进纸 n 点
原型	void printAndFeedNDotLine (unsigned char ucDotLineNum)
参数	ucDotLineNum: 点数（0~255）
返回	无
支持型号	所有型号
说明	将打印缓存里的内容打印并进纸 n 点 当打印缓存为空时，只进纸 n 点 本函数执行后，打印位置移动至下一行的起始位置

表 2.4 printAndBackFeedNDotLine

功能	打印并退纸 n 点
原型	void printAndBackFeedNDotLine (unsigned char ucDotLineNum)
参数	ucDotLineNum: 点数（0~255）
返回	无
支持型号	所有型号

续上表

说明	将打印缓存的内容打印并退纸 n 点 当打印缓存为空时，只退纸 n 点 本函数执行后，打印位置移动至下一行的起始位置
----	---

表 2.5 printAndFeedNFontLine

功能	打印并进纸 n 行
原型	void printAndFeedNFontLine (unsigned char ucFontLineNum)
参数	ucFontLineNum: 行数 (0~255)
返回	无
支持型号	所有型号
说明	将打印缓存的内容打印并进纸 n 行 当打印缓存为空时，只进纸 n 行，行间距由 lineSpacingSet 或 lineSpacingDefault 设置 本函数执行后，打印位置移动至下一行的起始位置

表 2.6 printAndBackFeedNFontLine

功能	打印并退纸 n 行
原型	void printAndBackFeedNFontLine (unsigned char ucFontLineNum)
参数	ucFontLineNum: 行数 (0~255)
返回	无
支持型号	所有型号
说明	将打印缓存的内容打印并退纸 n 行 当打印缓存为空时，只退纸 n 行，行间距由 lineSpacingSet 或 lineSpacingDefault 设置 本函数执行后，打印位置移动至下一行的起始位置

表 2.7 lineSpacingSet

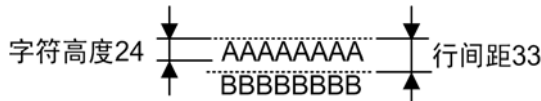
功能	设置行间距为 n 点
原型	void lineSpacingSet (unsigned char ucDotLineNum)
参数	ucDotLineNum: 点数 (0~255)
返回	无
支持型号	所有型号
说明	<p>设置行间距为 n 点，行间距示意如下：</p>  <p>若设定的行间距小于一行中的最大字符高度，那么该行行间距等于最大字符高度 若 lineSpacingDefault()、printerInit()、打印机复位、打印机断电，行间距恢复为默认值 33 点</p>

表 2.8 lineSpacingDefault

功能	设置行间距为默认值 33 点
原型	void lineSpacingDefault (void)
参数	无
返回	无
支持型号	所有型号
说明	设置行间距为默认的 33 点，行间距示意详看 lineSpacingSet() 若设定的行间距小于一行中的最大字符高度，那么该行行间距等于最大字符高度 可使用 lineSpacingSet()自定义行间距

表 2.9 leftMarginSet

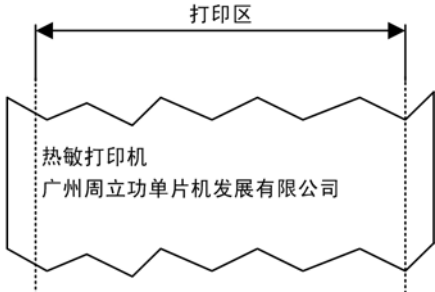
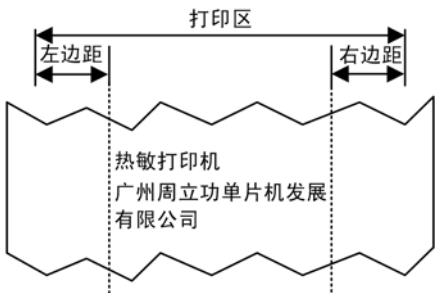
功能	设置左边距
原型	void leftMarginSet (unsigned char ucLeftMargin)
参数	ucLeftMargin: 左边距值，以 8 点为单位 ZYTP58、MTP58: $0 \leq \text{ucLeftMargin} \leq 47$ ，且 $0 \leq (\text{左边距} + \text{右边距}) \leq 47$ ZYTP80、MTP80: $0 \leq \text{ucLeftMargin} \leq 71$ ，且 $0 \leq (\text{左边距} + \text{右边距}) \leq 71$
返回	无
支持型号	所有型号
说明	<p>设置左边距，以 8 点为单位，使得内容打印不超过左边距位置 左边距位置表示打印内容的左侧边缘位置，左边距示意如下：</p> <div style="text-align: center;"> <p>情况一：左右边距为0</p>  <p>情况二：左右边距非0</p>  </div> <p>当 printerInit()、打印机复位、断电后，左边距设置失效（恢复为 0）</p>

表 2.10 rightMarginSet

功能	设置右边距
原型	void rightMarginSet (unsigned char ucRightMargin)

续上表

参数	ucRightMargin: 右边距值, 以 8 点为单位 ZYTP58、MTP58: $0 \leq \text{ucRightMargin} \leq 47$, 且 $0 \leq (\text{左边距} + \text{右边距}) \leq 47$ ZYTP80、MTP80: $0 \leq \text{ucRightMargin} \leq 71$, 且 $0 \leq (\text{左边距} + \text{右边距}) \leq 71$
返回	无
支持型号	所有型号
说明	设置右边距, 以 8 点为单位, 使得内容打印不超过右边距位置 右边距位置表示打印内容的右侧边缘位置, 右边距示意详看 leftMarginSet() 当 printerInit()、打印机复位、断电后, 右边距设置失效 (恢复为 0)

表 2.11 abscissaSet

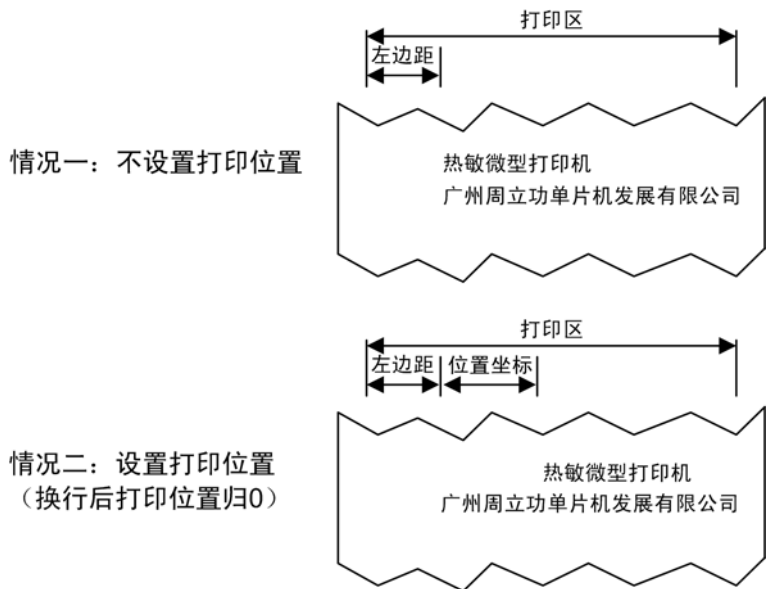
功能	设置打印位置
原型	void abscissaSet (unsigned short ucXNum)
参数	ucXNum: 打印位置, 以点为单位 (0~65535)
返回	无
支持型号	所有型号
说明	<p>调整打印位置到距离打印起始位置的 (ucXNum) 点处, 打印位置示意如图:</p> <div style="text-align: center;">  <p>情况一: 不设置打印位置</p> <p>情况二: 设置打印位置 (换行后打印位置归0)</p> </div> <p>若设置了左边距, 那么打印起始位置为左边距位置 此函数只对本行有效, 换行后打印位置复位为打印起始位置</p>

表 2.12 fontTypeSet

功能	设置字符打印方式
原型	void fontTypeSet (unsigned char ucFontType)

续上表

参数	<p>ucFontType: 字符打印方式, 可任意组合 (“或” 操作)</p> <p>FONT_TYPE_NORMAL 普通</p> <p>FONT_TYPE_ITALIC 斜体</p> <p>FONT_TYPE_BORDER 边框</p> <p>FONT_TYPE_EMPHASIZE 加粗</p> <p>FONT_TYPE_DOUBLE_HEIGHT 倍高</p> <p>FONT_TYPE_DOUBLE_WIDTH 倍宽</p> <p>FONT_TYPE_REVERSE_VEDIO 反白</p> <p>FONT_TYPE_UNDERLINE 下划线</p>
返回	无
支持型号	所有型号
说明	<p>设置字符打印方式（斜体、边框、加粗、倍宽、倍高、反白和下划线），可任意组合 此函数对中文字体及外文字体均有效 各种字符打印方式示意如下：</p> <div data-bbox="638 826 1069 1104" data-label="Image"> </div> <p>各种字符打印方式可自由组合 当 printerInit()、打印机复位、断电后，字符打印方式的设置失效</p>

表 2.13 horizontalAlign

功能	设置打印对齐方式
原型	void horizontalAlign(unsigned char align)
参数	<p>align: 打印对齐方式</p> <p>ALIGN_LEFT 居左</p> <p>ALIGN_CENTER 居中</p> <p>ALIGN_RIGHT 居右</p>
返回	<p>对一行中的所有数据进行对齐处理</p> <p>当 printerInit()、打印机复位、断电后，本函数的设置失效</p>
支持型号	所有型号
说明	无

表 2.14 fontGrayscaleSet

功能	设置打印灰度
原型	void fontGrayscaleSet (unsigned char ucGrayscale)
参数	ucGrayscale: 灰度等级 (1~8)

续上表

返回	无
支持型号	所有型号
说明	<p>设定打印内容的灰度，分 8 个等级 1~8，“1”为最浅，“8”为最深，用于解决不同热敏纸颜色深浅不一的问题</p> <p>对于 ZYTPxx-xx4xx、MTPxx-xx4xx：灰度值越小，打印速度越快，但灰度值太小可能会导致机芯中的步进电机失步，请根据实际情况调节</p> <p>对于 ZYTPxx-xx5xx、MTPxx-xx5xx：灰度值不影响打印速度</p> <p>当 printerInit()、打印机复位、断电后，本函数的设置失效</p>

表 2.15 printSpeedSet

功能	设置打印速度						
原型	void printSpeedSet (unsigned char ucPrintSpeed)						
参数	<p>ucPrintSpeed: 打印速度</p> <table> <tr> <td>PRINT_SPEED_LOW</td><td>低速</td></tr> <tr> <td>PRINT_SPEED_MIDIUM</td><td>中速</td></tr> <tr> <td>PRINT_SPEED_HIGHT</td><td>高速</td></tr> </table>	PRINT_SPEED_LOW	低速	PRINT_SPEED_MIDIUM	中速	PRINT_SPEED_HIGHT	高速
PRINT_SPEED_LOW	低速						
PRINT_SPEED_MIDIUM	中速						
PRINT_SPEED_HIGHT	高速						
返回	无						
支持型号	ZYTP80、MTP80、ZYTP58-xx5xx、MTP58-xx5xx						
说明	<p>设置打印速度</p> <p>ZYTP80/MTP80：在串口通信波特率为 9600bps 以下时，速度最大只可设置为中速 ($n \leq 1$)</p> <p>当 printerInit()、打印机复位、断电后，本函数的设置失效</p>						

表 2.16 fontSizeSet

功能	设置打印字体						
原型	void fontSizeSet (unsigned char ucFontSize)						
参数	<p>ucFontSize: 字体</p> <table> <tr> <td>FONT_SIZE_24</td><td>24×24</td></tr> <tr> <td>FONT_SIZE_16</td><td>16×16</td></tr> <tr> <td>FONT_SIZE_12</td><td>12×12</td></tr> </table>	FONT_SIZE_24	24×24	FONT_SIZE_16	16×16	FONT_SIZE_12	12×12
FONT_SIZE_24	24×24						
FONT_SIZE_16	16×16						
FONT_SIZE_12	12×12						
返回	无						
支持型号	ZYTP58-xx5xx、MTP58-xx5xx、ZYTP80-xx5xx、MTP80-xx5xx						
说明	<p>此函数对中文与外文均有效，并且仅支持多字体版本产品</p> <p>当 printerInit()、打印机复位、断电后，本函数的设置失效</p>						

表 2.17 picturePrintV

功能	图形垂直取模数据填充
原型	<p>void picturePrintV (unsigned char ucType,</p> <p> unsigned short usHSize,</p> <p> unsigned char * pucDataBuf)</p>

续上表

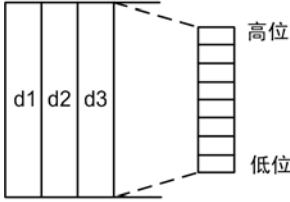
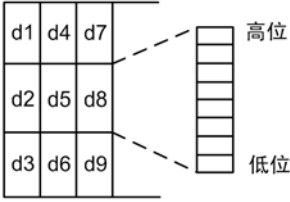
参数	<p>ucType: 点图方式, 可以为以下任意值</p> <p>DOT_8_SINGLE_DENSITY 8 点单密度</p> <p>DOT_8_DOUBLE_DENSITY 8 点双密度</p> <p>DOT_24_SINGLE_DENSITY 24 点单密度</p> <p>DOT_24_DOUBLE_DENSITY 24 点双密度</p> <p>usHSize: 水平方向点数</p> <p>ZYTP58、MTP58: $0 \leq \text{usHSize} \leq 384$</p> <p>ZYTP80、MTP80: $0 \leq \text{usHSize} \leq 576$</p> <p>pucDataBuf: 图片数据缓存首址</p>
返回	无
支持型号	所有型号
说明	<p>pucDataBuf 相应位为 1 则表示该点打印, 相应位为 0, 则表示该点不打印</p> <p>图像水平方向超出打印区域的部分将被忽略</p> <p>pucDataBuf 点图数据与打印效果的关系如下:</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;"> <p>8点方式</p>  <p>点图数据 (位图)</p> </div> <div style="text-align: center;"> <p>24点方式</p>  <p>点图数据 (位图)</p> </div> </div> <p>此函数只填充打印缓存, 图像的打印要在接收到打印函数后才开始, 图像打印完毕后打印缓存被清空</p> <p>若需要打印的图像高度较大, 可以先拆分为若干条高度为 8 (DOT_8_SINGLE_DENSITY 或 DOT_8_DOUBLE_DENSITY) 或 24 (DOT_24_SINGLE_DENSITY 或 DOT_24_DOUBLE_DENSITY) 点的图像分别打印</p> <p>填充图形数据后, 可以继续填充其它信息, 以使图形与其它信息一同被打印</p> <p>填充点图后, 一般使用 printAndFeedNdotLine(24)进行打印, 也可以使用 print()函数进行打印, 但是 print()函数会引发进纸操作 (按行间距进纸), 使得多行图像间断不连续</p>

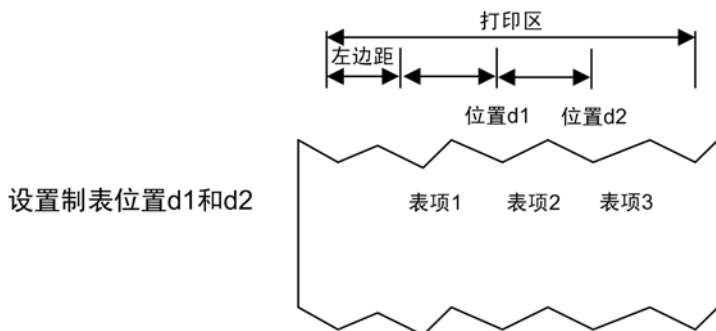
表 2.18 picturePrintH

功能	图片水平取模数据打印
原型	<pre>void picturePrintH (unsigned char ucType, unsigned short usHSize, unsigned short usVSize, unsigned char * pucDataBuf)</pre>
参数	<p>ucType: 点图方式, 可以为以下任意值</p> <p>PICTURE_TYPE_NORMAL 普通</p> <p>PICTURE_TYPE_DOUBLE_WIDTH 倍宽</p> <p>PICTURE_TYPE_DOUBLE_HEIGHT 倍高</p> <p>PICTURE_TYPE_QUADRUPLE 倍高倍宽</p> <p>usHSize: 水平方向字节数</p>

续上表

	<p>ZYTP58、MTP58: $0 \leq \text{usHSize} \leq 48$</p> <p>ZYTP80、MTP80: $0 \leq \text{usHSize} \leq 72$</p> <p>usVSize: 垂直方向点数（0~65535）</p> <p>pucDataBuf: 图片数据缓存首址</p>																				
返回	无																				
支持型号	所有型号																				
说明	<p>pucDataBuf 相应位为 1 则表示该点打印，相应位为 0，则表示该点不打印</p> <p>若图像水平字节数超出打印区域，超出部分将被忽略</p> <p>此函数执行时按图像大小进纸，不受 lineSpaceSet()、lineSpaceSetDefault()的行间距设置影响</p> <p>此函数执行后，打印坐标复位到左边距位置处，图像内容被清空</p> <p>位图数据与打印效果的关系如下：</p> <table><tr><td>d1</td><td>d2</td><td>.....</td><td>dx</td></tr><tr><td>d(x+1)</td><td>d(x+2)</td><td>.....</td><td>d(x+2)</td></tr><tr><td> </td><td> </td><td>.....</td><td> </td></tr><tr><td>.....</td><td>d(k-2)</td><td>d(k-1)</td><td>dk</td></tr><tr><td>MSB</td><td>LSB</td><td>MSB</td><td>LSB</td></tr></table> <p>此函数带有打印功能，边传数据边打印，不需要再使用打印函数</p>	d1	d2	dx	d(x+1)	d(x+2)	d(x+2)			d(k-2)	d(k-1)	dk	MSB	LSB	MSB	LSB
d1	d2	dx																		
d(x+1)	d(x+2)	d(x+2)																		
																				
.....	d(k-2)	d(k-1)	dk																		
MSB	LSB	MSB	LSB																		

表 2.19 horizontalTabSet

功能	设置水平制表位置
原型	void horizontalTabSet (unsigned char * pucNBuf, unsigned char ucDataNum)
参数	<p>pucNBuf: 水平制表位置数据, 以 8 点单位</p> <p>ZYTP58、MTP58: $0 \leq \text{pucNBuf}[k] \leq 48$</p> <p>ZYTP80、MTP80: $0 \leq \text{pucNBuf}[k] \leq 72$</p> <p>$\text{pucNBuf}[k] \geq \text{pucNBuf}[k-1]$</p> <p>ucDataNum: 数据个数 (0~16)</p>
返回	无
支持型号	所有型号
说明	<p>制表位置示意如下:</p> <div data-bbox="496 1503 1214 1832" data-label="Diagram">  </div> <p>最多支持 16 个制表位置的设定</p> <p>使用此函数将取消以往制表位置的设置</p> <p>若 pucNBuf[k]小于或等于 pucNBuf[k-1], 视为结束, 剩余数据视为普通数据处理</p> <p>制表位置可由 horizontalTab()切换</p>

续上表

	当左边距改变后，制表位置同时改变 当 printerInit()、打印机复位、断电后，本函数的设置失效
--	--

表 2.20 horizontalTab

功能	水平制表
原型	void horizontalTab (void)
参数	无
返回	无
支持型号	所有型号
说明	制表位置由 horizontalTabSet() 设定 若制表位置未设置（默认无水平制表位置），此函数将视为 print() 函数 若制表位置超出打印区域，坐标将移至下一行的起始位置（视本行数据已满，打印并换行）

表 2.21 formPrintV

功能	打印 90° 表格
原型	void formPrintV (unsigned char ucVLineNum, unsigned char *pucVLinePos, unsigned char ucItemNum, formItem *psItem)
参数	ucVLineNum: 表格竖线数 (0~17) pucVLinePos: 表格竖线坐标数组，以 8 点为单位 ZYTP58、MTP58: $0 \leq \text{pucVLinePos}[k] \leq 48$ ZYTP80、MTP80: $0 \leq \text{pucVLinePos}[k] \leq 72$ $\text{pucVLinePos}[k] \geq \text{pucVLinePos}[k-1]$ ucItemNum: 表格项数 (0~16) psItem: 表项内容，包括坐标、文字属性和文字内容等，结构体定义如下： typedef struct __formItem { unsigned char ucPos; /* 表项坐标 */ /* 范围与 pucVLinePos 相同 */ unsigned char ucFontType; /* 表项字体属性 */ unsigned char ucDataLen; /* 表项数据长度 (0~20) */ unsigned char *pucDataBuf; /* 表项数据 */ }formItem; 表项字体属性可选项： VF_FONT_TYPE_NORMAL 普通 VF_FONT_TYPE_EMPHASIZE 加粗 VF_FONT_TYPE_UNDERLINE 下划线 VF_FONT_TYPE_REVERSE_VEDIO 反白
返回	无
支持型号	ZYTPxx-xxxCx、MTPxx-xxxCx

续上表

说明	<p>垂直表格相关参数示意图如下：</p> <p>坐标以打印纸进纸方向右边为 0 参考 每个表项最多 10 个汉字或 20 个外文字符 不需要打印表格边框线时 m 为 0</p>
----	---

表 2.22 barcodeHeightSet

功能	设置一维条码高度
原型	void barcodeHeightSet(unsigned char ucHeight)
参数	ucHeight: 条码高度, 以点为单位 (0~255)
返回	无
支持型号	ZYTPxx-xxxBx、ZYTPxx-xxxEx、MTPxx-xxxBx、MTPxx-xxxEx
说明	<p>设置条码的高度为 ucHeight 点, 示意如下:</p> <p>当 printerInit()、打印机复位、断电后, 本函数的设置失效</p>

表 2.23 barcodeWidthSet

功能	设置一维条码宽度
原型	void barcodeWidthSet(unsigned char ucWidth)
参数	ucWidth: 条码黑白条宽度, 以点为单位, 范围 (1~6)
返回	无
支持型号	ZYTPxx-xxxBx、ZYTPxx-xxxEx、MTPxx-xxxBx、MTPxx-xxxEx

续上表

说明	<p>设置条码单元为 ucWidth 点，示意如下：</p> <div style="text-align: center;">  宽度为 3  宽度为 4 </div> <p>当 printerInit()、打印机复位、断电后，本函数的设置失效</p>
----	--

表 2.24 barcodeHriPosSet

功能	设置条码可读字符（HRI）打印位置								
原型	void barcodeHriPosSet (unsigned char ucHriPos)								
参数	<p>ucHriPos: 条码 HRI 显示位置</p> <table border="0"> <tr> <td>BARCODE_HRI_POS_NONE</td><td>无</td></tr> <tr> <td>BARCODE_HRI_POS_TOP</td><td>上面</td></tr> <tr> <td>BARCODE_HRI_POS_BOTTOM</td><td>下面</td></tr> <tr> <td>BARCODE_HRI_POS_TOP_BOTTOM</td><td>上面+下面</td></tr> </table>	BARCODE_HRI_POS_NONE	无	BARCODE_HRI_POS_TOP	上面	BARCODE_HRI_POS_BOTTOM	下面	BARCODE_HRI_POS_TOP_BOTTOM	上面+下面
BARCODE_HRI_POS_NONE	无								
BARCODE_HRI_POS_TOP	上面								
BARCODE_HRI_POS_BOTTOM	下面								
BARCODE_HRI_POS_TOP_BOTTOM	上面+下面								
返回	无								
支持型号	ZYTPxx-xxxBx、ZYTPxx-xxxEx、MTPxx-xxxBx、MTPxx-xxxEx								
说明	多字体版本模块（xxTPxx-xx5Bx）的条码可读字符（HRI）的字体可由 barcodeHriFontSet 设定；当 printerInit()、打印机复位、断电后，本函数的设置失效								

表 2.25 barcodeHriFontSet

功能	设置一维条码字体						
原型	void barcodeHriFontSet(unsigned char ucHriFont)						
参数	<p>ucHriFont: 条码文字字体</p> <table border="0"> <tr> <td>BARCODE_HRI_FONT_24</td><td>12×24</td></tr> <tr> <td>BARCODE_HRI_FONT_16</td><td>8×16</td></tr> <tr> <td>BARCODE_HRI_FONT_12</td><td>6×12</td></tr> </table>	BARCODE_HRI_FONT_24	12×24	BARCODE_HRI_FONT_16	8×16	BARCODE_HRI_FONT_12	6×12
BARCODE_HRI_FONT_24	12×24						
BARCODE_HRI_FONT_16	8×16						
BARCODE_HRI_FONT_12	6×12						
返回	无						
支持型号	ZYTPxx-xxxBx、ZYTPxx-xxxEx、MTPxx-xxxBx、MTPxx-xxxEx						
说明	当 printerInit()、打印机复位、断电后，本函数的设置失效						

表 2.26 barcodePrint

功能	打印条码
原型	void barcodePrint(unsigned char ucBarcodeSys, unsigned char *pucCodeBuf, unsigned char ucCodeLen);

续上表

参数	ucBarcodeSys: 条码系统					
	条码系统			代号		
	BARCODE_SYS_UPCA			0		
	BARCODE_SYS_UPCE			1		
	BARCODE_SYS_EAN13			2		
	BARCODE_SYS_EAN8			3		
	BARCODE_SYS_CODE39			4		
	BARCODE_SYS_ITF25			5		
	BARCODE_SYS_CODABAR			6		
	BARCODE_SYS_CODE93			7		
	BARCODE_SYS_CODE128			8		
BARCODE_SYS_EAN128			9			
pucCodeBuf: 条码数据首地址						
ucCodeLen: 条码数据长度						
返回	无					
支持型号	ZYTPxx-xxxBx、ZYTPxx-xxxEx、MTPxx-xxxBx、MTPxx-xxxEx					
说明	编码系统、条码数据、条码数据长度的关系:					
	代号	条码系统	数据长度		条码数据（SP 表示空格）	
					字符集	数据（d）
	0	UPC-A	固定	n = 11, 12	0~9	48≤d≤57
	1	UPC-E	固定	6≤n≤8, n = 11, 12	0~9	48≤d≤57 [当 k = 7, 8, 11, 12 时 d1 = 48]
	2	JAN13(EAN13)	固定	n = 12, 13	0~9	48≤d≤57
	3	JAN8(EAN8)	固定	n = 7, 8	0~9	48≤d≤57
	4	CODE39	可变	1≤n≤255	0~9, A~Z SP, \$, %, *, +, -, ., /	48≤d≤57, 65≤d≤90 d = 32, 36, 37, 42, 43, 45, 46, 47
	5	ITF (Interleaved 2 of 5)	可变	2≤n≤255 (偶数)	0~9	48≤d≤57
	6	CODABAR (NW-7)	可变	1≤n≤255	0~9, A~D, a~d \$, +, -, ., /, :	48≤d≤57, 65≤d≤68, 97≤d≤100, d = 36, 43, 45, 46, 47, 58 (65≤d1≤68, 65≤dk≤68, 97≤d1≤100, 97≤dk≤100)
	7	CODE93	可变	1≤n≤255	00H~7FH	0≤d≤127

续上表

续上表

8	CODE128	可变	$1 \leq n \leq 255$	00H~7FH C1H~C4H(FNC)	$0 \leq d \leq 127$ $d = 193, 194, 195, 196$
9	UCC/EAN128	可变	$1 \leq n \leq 255$	00H~7FH C1H~C4H(FNC)	$0 \leq d \leq 127$ $d = 193, 194, 195, 196$

打印 UPCA（代号为 0）时，需要注意：

不论输入数据长度是 11 还是 12，校验位自动插入或纠错

起始符、中间分隔符、结束符自动插入

打印 UPCE（代号为 1）时，需要注意：

当数据长度为 6 时，系统字符（NSC）0 自动插入

当数据长度为 7、8、11 和 12 时，第一位系统字符（NSC）d1 必须为 0

不论输入数据长度是 6、7、8、11 还是 12，校验位自动插入或纠错

不论输入数据长度是 6、7、8、11 还是 12，条码可读字符（HRI）只显示 6 为数据，不包含系统字符（NSC）和校验码；

传输数据与打印数据转换关系如下：

传输的数据											打印的数据					
d2	d3	d4	d5	d6	d7	d8	d9	d10	d11		d1	d2	d3	d4	d5	d6
0~9	0~9	0	0	0	-	-	0~9	0~9	0~9		d2	d3	d9	d10	d11	0
0~9	0~9	1	0	0	-	-	0~9	0~9	0~9		d2	d3	d9	d10	d11	1
0~9	0~9	2	0	0	-	-	0~9	0~9	0~9		d2	d3	d9	d10	d11	2
0~9	0~9	3~9	0	0	-	-	-	0~9	0~9		d2	d3	d4	d10	d11	3
0~9	0~9	0~9	1~9	0	-	-	-	-	0~9		d2	d3	d4	d5	d11	4
0~9	0~9	0~9	0~9	1~9	-	-	-	-	5~9		d2	d3	d4	d5	d6	d11

当 d6 为 1~9 时，应保证 d11 为 5~9

起始符、结束符自动插入

打印 EAN13（代号为 2）时，需要注意：

不论输入数据长度是 12 还是 13，校验位自动插入或纠错

起始符、中间分隔符、结束符自动插入

打印 EAN8（代号为 3）时，需要注意：

不论输入数据长度是 7 还是 8，校验位自动插入或纠错

起始符、中间分隔符、结束符自动插入

打印 CODE39（代号为 4）时，需要注意：

当 d1 或 dn 不为起始符/结束符“*”时，编码器自动插入“*”

当数据中间遇到“*”时，编码器视其为结束符，其余数据视为普通数据处理；

校验位不会自动计算和添加

打印 ITF25（代号为 5）时，需要注意：

起始符和结束符自动插入

校验位不会自动计算和添加

打印 CODABAR（NW-7）（代号为 6）时，需要注意：

起始符和结束符不会自动插入，需要用户手动添加，范围为“A”~“D”或“a”

续上表

~ “d”

校验位不会自动计算和添加

打印 CODE93（代号为 7）时，需要注意：

起始符和结束符自动插入

两个校验码自动计算并插入

当设置条码可读字符（HRI）打印时，不设任何表示起始/结束的 HRI 字符

当设置条码可读字符（HRI）打印时，控制字符将用空格代替

打印 CODE128（代号为 8）时，需要注意：

编码系统智能识别数据并实现最小长度编码，无需用户设置字符集（包括起始字符集）或切换字符集

功能字符 FNC1~FNC4 使用 C1H~C4H 输入

校验位自动计算和添加

当设置条码可读字符（HRI）打印时，控制字符和 FNC1~FNC4 将用空格代替

打印 EAN128（代号为 9）时，需要注意

基本结构如下：

起始 字符集	FNC1	AI	数据 部分	校验位 A	校验位 B	结束符
自动插入		(d1 ... dk)			自动插入	

连接结构如下：

起始 字符集	FNC1	AI	数 据 部 分	校 验 位 A	FNC1	AI	数 据 部 分	校 验 位 A	校验位 B	结束符
自动插入		(d1 ... dk)						自动插入		

编码系统智能识别数据并实现最小长度编码，无需用户设置字符集（包括起始字符集）或切换字符集

功能字符 FNC1~FNC4 使用 C1H~C4H 输入

用户输入数据中 AI 不需要用 “（” “）” 指示，编码系统自动插入，否则侧会出错，如：
GS k 74 18 "019501234567890*"，01 是 AI，以下是错误的：GS k 74 18 "(01)9501234567890*"

当使用连接结构时，中间需要插入 FNC1（C1H “Decimal = 193”）输入例子如下：
GS k 74 18 "019501234567890*" 193 "029501234567890*"

当设置条码可读字符（HRI）打印时，控制字符将用空格代替，而 FNC1~FNC4 将去掉

表 2.27 printerInit

功能	初始化打印机
原型	void printerInit (void)
参数	无
返回	无
支持型号	所有型号

续上表

说明	初始化打印机下列内容： <ul style="list-style-type: none"> ● 清除打印缓存 ● 各参数恢复默认值
----	--

表 2.28 printerClearBuffer

功能	立即清空打印缓存（实时）
原型	void printerClearBuffer (void)
参数	无
返回	无
支持型号	所有型号
说明	当接收到该函数，打印机无条件立即执行清空缓存区的操作

表 2.29 printerCutPaper

功能	走纸并切纸
原型	void printerCutPaper(unsigned char ucDotLineNum, unsigned char type)
参数	ucDotLineNum: 切纸前走纸的点行数 type: 切纸方式 CUT_TYPE_HALF 半切 CUT_TYPE_ALL 全切
返回	无
支持型号	ZYTPxx-xxxxC、MTPxx-xxxxC
说明	先进纸再切纸

表 2.30 printerStateRT

功能	获取打印机状态（实时的）
原型	unsigned char printerStateRT(unsigned char *pucState, unsigned short usTimeOut)
参数	pucState: 打印机返回状态值，各个数据位独立 RINTER_STATE_NORMAL 一切正常 PRINTER_STATE_OVERVOLATGE 过压 PRINTER_STATE_AXOOPEN 压轴打开 PRINTER_STATE_OFPAPER 缺纸 PRINTER_STATE_OVERHEAT 过热 PRINTER_STATE_CUTTERON 切刀移开 usTimeOut: 应答超时（1ms 为单位）
返回	FAIL: 失败, SUCCESS: 成功
支持型号	所有型号
说明	注意，zyPrtLib 库将会自动对返回的状态值进行处理，可能会调用过热、缺纸等入口函数

表 2.31 printerStateAutoUpload

功能	设置/取消打印机状态自动返回
原型	void printerStateAutoUpload(unsigned char key)
参数	key: 选项开关, 多个选项可以相或 PRINTER_STATE_OVERVOLATGE 过压 PRINTER_STATE_AXOOPEN 压轴打开 PRINTER_STATE_OFPAPER 缺纸 PRINTER_STATE_OVERHEAT 过热 PRINTER_STATE_CUTTERON 切刀移开
返回	无
支持型号	所有型号
说明	默认为全部关闭, 建议模块上电或低功耗恢复后首次打印前设置, 其他情况下设置可能会出现不可预计的效果 当 printerInit()、打印机复位、断电后, 本函数的设置失效

表 2.32 printerBaudRateSet

功能	打印机波特率设置
原型	void printerBaudRateSet (unsigned long ulBaudRate)
参数	ulBaudRate: 波特率值, 如 9600, 支持范围 110~115200
返回	无
支持型号	所有型号
说明	建议模块上电或低功耗恢复后首次打印前设置, 其他情况下设置可能会出现不可预计的效果。 当打印机复位、断电后, 本函数的设置失效

表 2.33 printerFlowCtrlSet

功能	打印机流控制设置
原型	void printerFlowCtrlSet(unsigned char Mode)
参数	Mode: 流控制模式 FLOW_CTRL_NONE 0 /* 不使用流控制 */ FLOW_CTRL_SOFTWARE 49 /* 使用软件流控制 */ FLOW_CTRL_HARDWARE 48 /* 使用硬件流控制 */
返回	无
支持型号	所有型号
说明	默认无流控制, 建议模块上电或低功耗恢复后首次打印前设置, 其他情况下设置可能会出现不可预计的效果。 当 printerInit()、打印机复位、断电后, 本函数的设置失效

表 2.34 printerEnterLowPowerMode

功能	打印机进入低功耗模式
原型	unsigned char printerEnterLowPowerMode (unsigned short usTimeOut)
参数	usTimeOut: 应答超时（1ms 为单位）
返回	FAIL: 失败, SUCCESS: 成功
支持型号	除 ZYTPxx-xxxxC、MTPxx-xxxxC 以外其他型号均支持
说明	无

表 2.35 printerExitLowPowerMode

功能	打印机退出低功耗模式
原型	unsigned char printerExitLowPowerMode (unsigned short usTimeOut)
参数	usTimeOut: 应答超时（1ms 为单位）
返回	FAIL: 失败, SUCCESS: 成功
支持型号	除 ZYTPxx-xxxxC、MTPxx-xxxxC 以外其他型号均支持
说明	进入低功耗模式后，应该插入适当延时再退出低功耗模式。

表 2.36 printerOverHeatHandle

功能	打印机过热处理入口
原型	void printerOverHeatHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.37 printerOverHeatResumeHandle

功能	打印机过热恢复处理入口
原型	void printerOverHeatResumeHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.38 printerOfPaperHandle

功能	打印机缺纸处理入口
原型	void printerOfPaperHandle (void)
参数	无
返回	无

续上表

说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间
----	---

表 2.39 printerOfPaperResumeHandle

功能	打印机缺纸恢复处理入口
原型	void printerOfPaperResumeHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.40 printerOverVoltageHandle

功能	打印机过压处理入口
原型	void printerOverVoltageHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.41 printerOverVoltageResumeHandle

功能	打印机过压恢复处理入口
原型	void printerOverVoltageResumeHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.42 printerAxoOpenHandle

功能	打印机压轴打开处理入口
原型	void printerAxoOpenHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.43 printerAxoCloseHandle

功能	打印机压轴关闭处理入口
原型	void printerAxoCloseHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.44 printerCutterOnHandle

功能	打印机切刀移开处理入口
原型	void printerCutterOnHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

表 2.45 printerCutterOffHandle

功能	打印机切刀关闭处理入口
原型	void printerCutterOffHandle (void)
参数	无
返回	无
说明	在串口接收中断函数中调用，请勿写带阻塞的代码，且建议代码处理时间应小于串口字节接收时间

3. 免责声明

广州致远电子有限公司随附提供的软件或文档资料旨在提供给您(本公司的客户)使用,仅限于且只能在本公司制造或销售的产品上使用。

该软件或文档资料为本公司和/或其供应商所有,并受适用的版权法保护。版权所有。如有违反,将面临相关适用法律的刑事制裁,并承担违背此许可的条款和条件的民事责任。本公司保留在不通知读者的情况下,修改文档或软件相关内容的权利,对于使用中所出现的任何效果,本公司不承担任何责任。

该软件或文档资料“按现状”提供。不提供保证,无论是明示的、暗示的还是法定的保证。这些保证包括(但不限于)对出于某一特定目的应用此文档的适销性和适用性默示的保证。在任何情况下,公司不会对任何原因造成的特别的、偶然的或间接的损害负责。