字符卡使用说明 V2

一、 节目内可以存在图文，表盘，时间，倒计时等分区，和一个字符分区。

二、 字符分区同时支持两种显示模式

1. Flash 显示模式：显示多条写入 Flash 的数据(可以通过命令切换上/下，以及某个索引的数据)。
2. 内存显示模式：在显示的时候可以通过命令直接写入一段字符串，字符卡直接显示当前命令发送的字符串。

三、 保存的字符串数量最高 250 个,每条字符串最多可以保存 500 个中文。

四、 支持 GB2312 以及 UNICODE 两种编码方式。

五、 中文字体大小支持 16*16,32*32,48*48，英文支持 16*8, 32*16,48*24。

六、 发送节目数据以及字库需本公司提供的字符卡定制上位机软件。

七、 使用网络卡设备，则协议数据封包后，在协议前面添加 0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x00,0x00,0x00 十个数据头。

➢字符卡通信命令说明

一帧数据由包头、协议版本、控制卡地址、命令、帧计数、数据长度、数据、检验字和包尾组成，具体说明见下表。超过 1byte 长度的数据类型，低字节在前,高字节在后。

发送	start	ver	addr	cmd	ident	frame	len	data	check	end
	0x78	0x34	0x0001	xx	0x00000000	0x00000000	xx	xx	xx	0xA5

返回	startR	ver	addr	cmd	ident	frame	len	data	state	check	end
	0x79	0x34	0x0001	xx	0x00000000	0x00000000	xx	xx	0x0000	xx	0xA5

上位机发送命令格式，表 1

序号	长度	名称	值	说明
1	1byte	start	0x78	包头一帧数据的起始
2	1byte	ver	0x34	协议版本
3	2byte	addr	0x0001	控制卡地址
4	**1byte**	**cmd**	**xx**	**通信命令，用于区分更新/删除/切换显示命令**
5	4byte	ident	0x00000000	识别标志
6	4byte	frame	0x00000000	帧计数
7	**2byte**	**len**	**xx**	**data 字段的数据长度**
8	**nbyte**	**data**	**xx**	**数据，控制命令中有些没有数据，n=0。**
9	**2byte**	**check**	**xx**	**CRC 校验，序号 1-8 数据的 CRC 效验。**
	1byte	end	0xA5	一帧结束标志

下位机返回命令格式，表 2

序号	长度	名称	值	说明
1	1byte	startR	0x79	帧起始
2	1byte	ver	0x34	协议版本
3	2byte	addr	0x0001	控制卡地址
4	**1byte**	**cmd**	**xx**	**通信命令**
5	4byte	ident	0x00000000	识别标志
6	4byte	frame	0x00000000	帧计数
7	**2byte**	**len**	**xx**	**data 字段的数据长度**
8	**nbyte**	**data**	**xx**	**数据，控制命令中有些没有数据，n=0。**
9	2byte	state	0x0000	状态值(无效）
10	**2byte**	**check**	**xx**	**CRC 校验，序号 1-9 数据的 CRC 效验。**
	1byte	end	0xA5	一帧结束标志

➢ 字符串卡命令

cmd	命令
0x29	字符串更新命令
0x2A	字符串删除命令
0x3F	切换显示字符串命令

➢ 字符串更新命令 cmd = 0x29

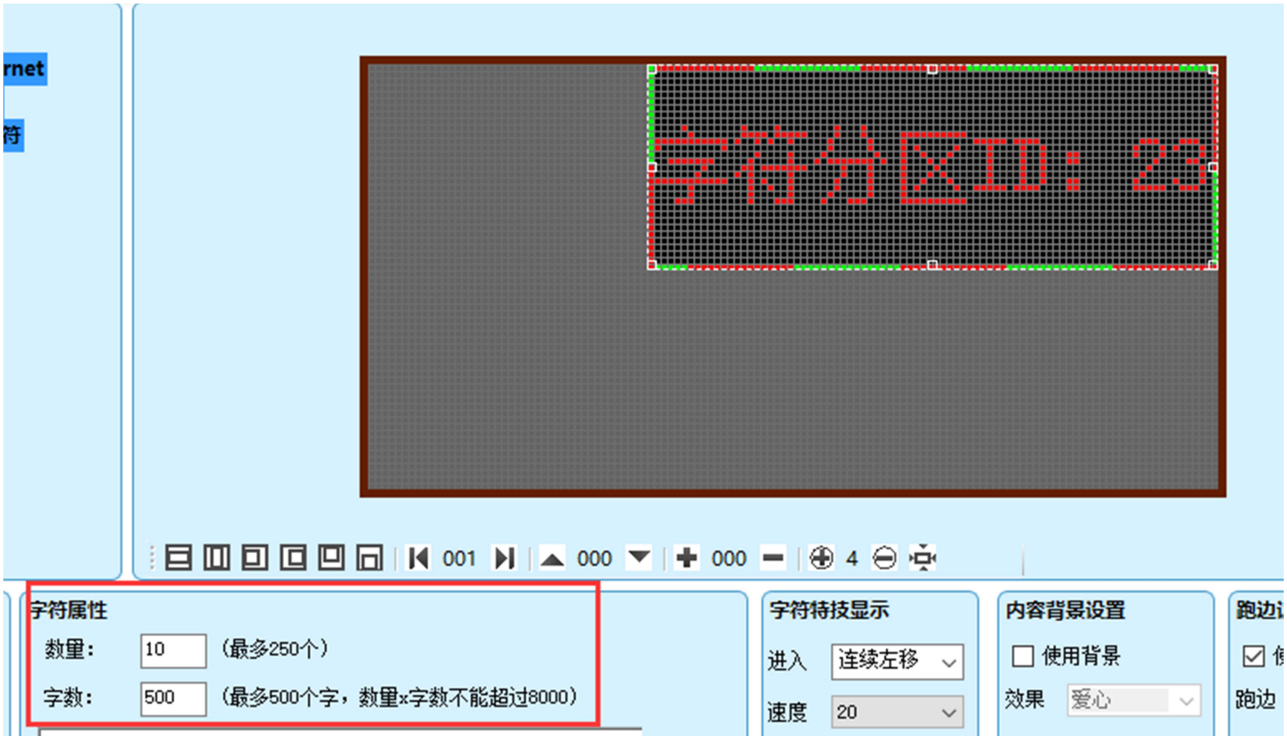
发送协议内 data 字段说明：

	编码方式	显示方式	字符串索引	颜色	长度	字符串
长度	1byte	1byte	1byte	1byte	2byte	Nbyte
备注	0=unicode 编码 1=gb2312 编码	0= 保存数据模式 2= 立即显示模式	不能超过最大数量（如下图）	1 = 红色 2 = 绿色 3 = 黄色 4 = 蓝色 5 = 紫色 6 = 青色 7 = 白色	不能超过最长字数（如下图）	字符串内容
例程一	1	2	0	0x02	5	我很好中航
例程二	1	0	5	0x01	6	123456

例程一：发送 gb2312 编码，立即显示模式，显示绿色“我很好中航”的字符串。
例程二：发送 gb2312 编码，索引为 5，保存数据模式，显示红色“012345”的字符串。

注意：

- 字符串内的数据，每个汉字或者英文（ASCII）都占据两个字节空间，英文的高位填 0。
  例如：”0123”的 ASCII 值 = “0x00, 0x30,0x00, 0x31,0x00, 0x32,0x00, 0x33,0x00, 0x30”;
  例如：”我很好”的 GB2312 值 = “0xCE,0xD2,0xBA,0xDC, 0xBA,0xC3”;
- 字符串索引只在保存数据模式下有效
- 使用上位机发送分区节目时存在两个两个参数（如下图）



设备返回协议内 len 长度字段说明：

	返回说明
长度	2byte
备注	Len = 0 (命令正确) Len = 1(字符串类型错误 或者超过最大数量 或者最大字数)

➢ 字符串删除命令 cmd = 0x2A

发送协议内 data 字段说明：

	删除类型	删除索引
长度	1byte	1byte
备注	0=删除 Flash 数据 2=删除立即显示数据	若索引等于 0xFF 则删除全部 Flash 数据,否则删除索引对应的字符串数据
例程一	2	0
例程二	0	6
例程三	0	255

例程一：删除立即显示字符串数据。
例程二：删除索引为 6 的字符串数据。
例程三：删除所有的保存数据模式下的字符串。

➢ 字符串切换显示命令 cmd = 0x3F

发送协议内 data 字段说明：

	切换类型	切换索引
长度	1byte	1byte
备注	0=显示指定字符串索引命令 1=自动显示下个索引字符串 2=自动显示上个索引字符串	字符串索引
例程一	0	7
例程二	1	0
例程三	2	0

例程一：切换显示索引=7 的字符串
例程二：切换下一个字符串。
例程三：切换上一个字符串。

设备返回协议内 len 长度字段说明：

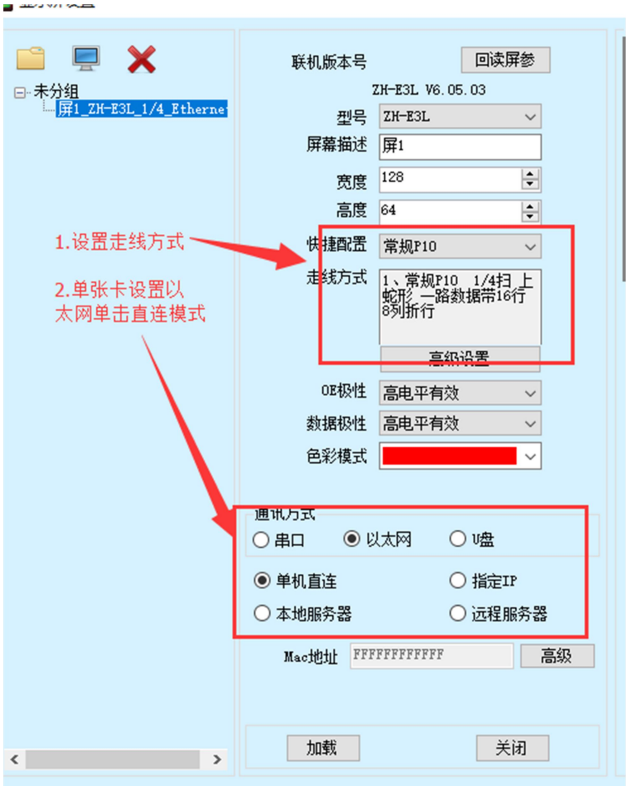
	返回说明
长度	2byte
备注	Len = 0(命令正确) Len = 1(索引不存在)

附一：字符卡使用步骤（三选一）

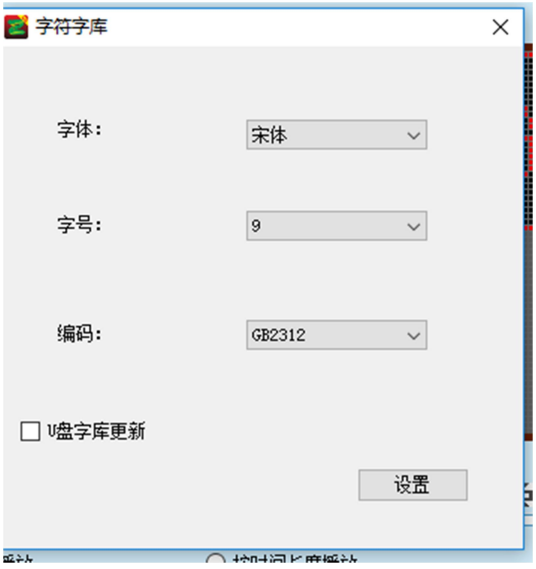
一、 字符卡单机直连使用步骤(单张卡)

1 字符卡插入网线并且与电脑直插

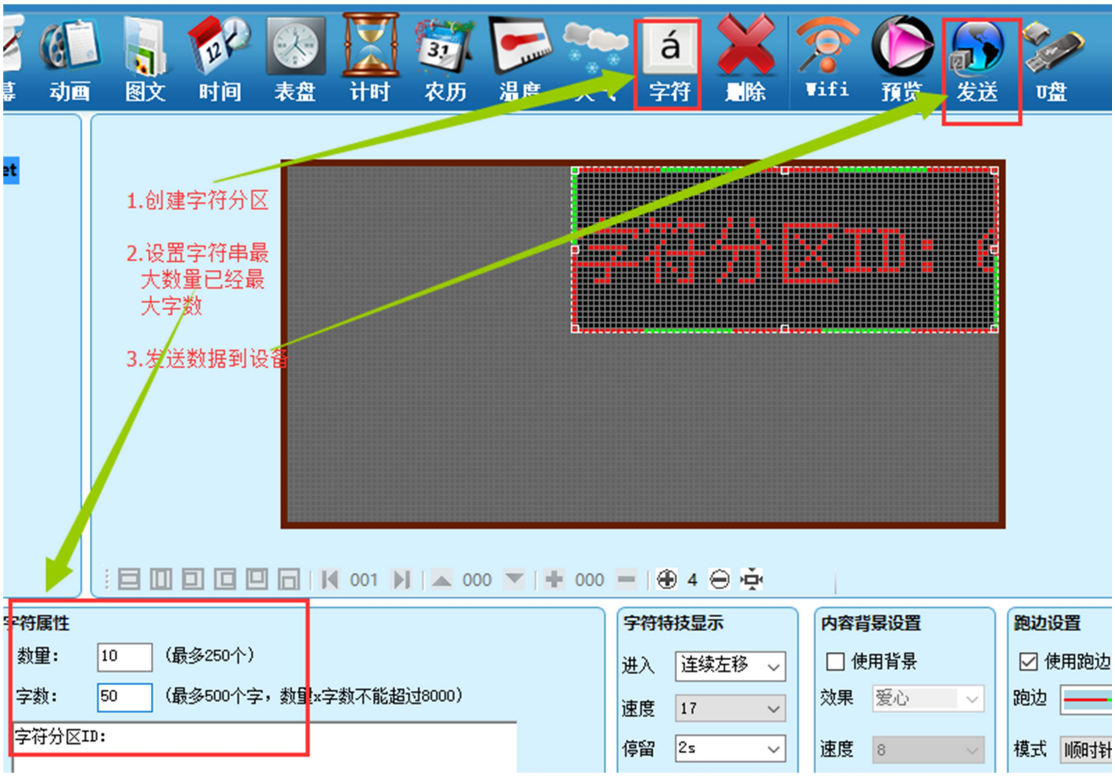
2 设置显示屏参数(菜单栏➔设置➔屏参设置➔密码 888)



3 发送字库数据（菜单栏➔编辑➔字符字库➔设置）



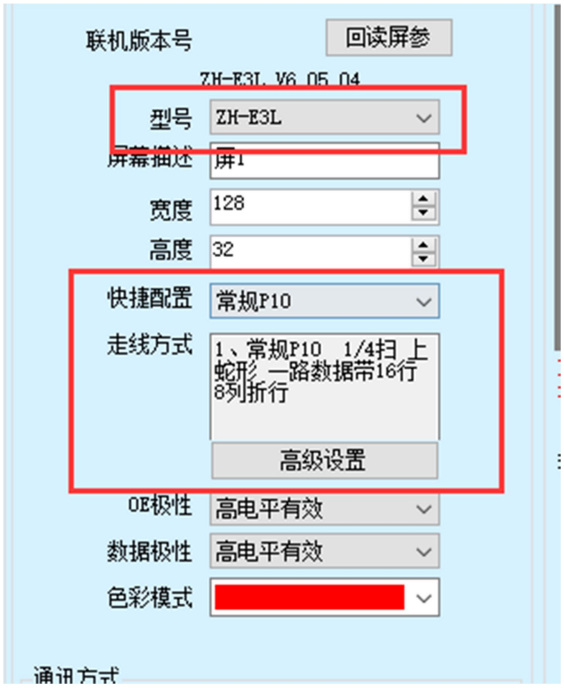
4 创建字符分区并发送数据到字符卡设备上



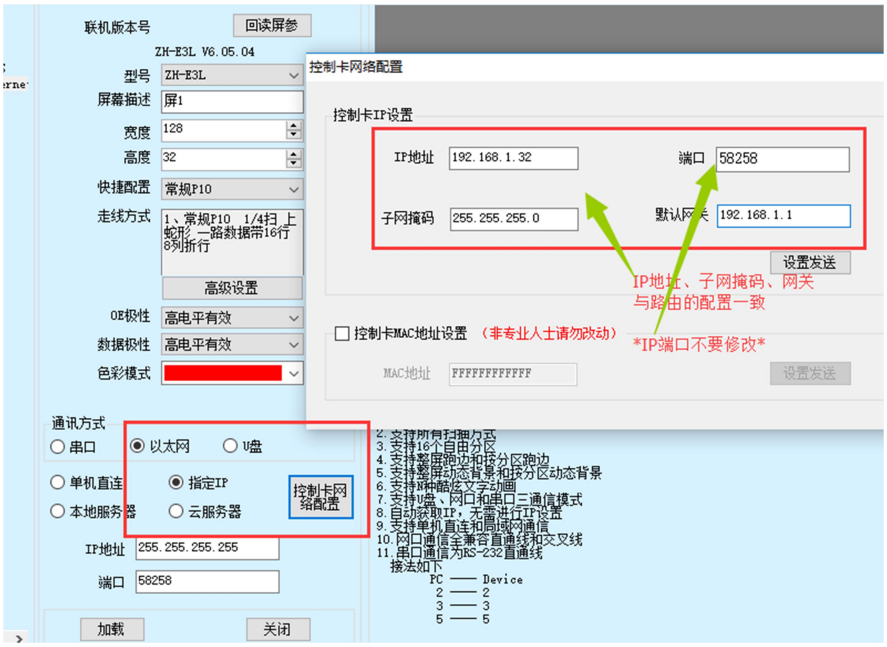
二、 字符卡指定 IP 通信使用步骤

1. 字符卡插入网线，网线的另一端插入电脑

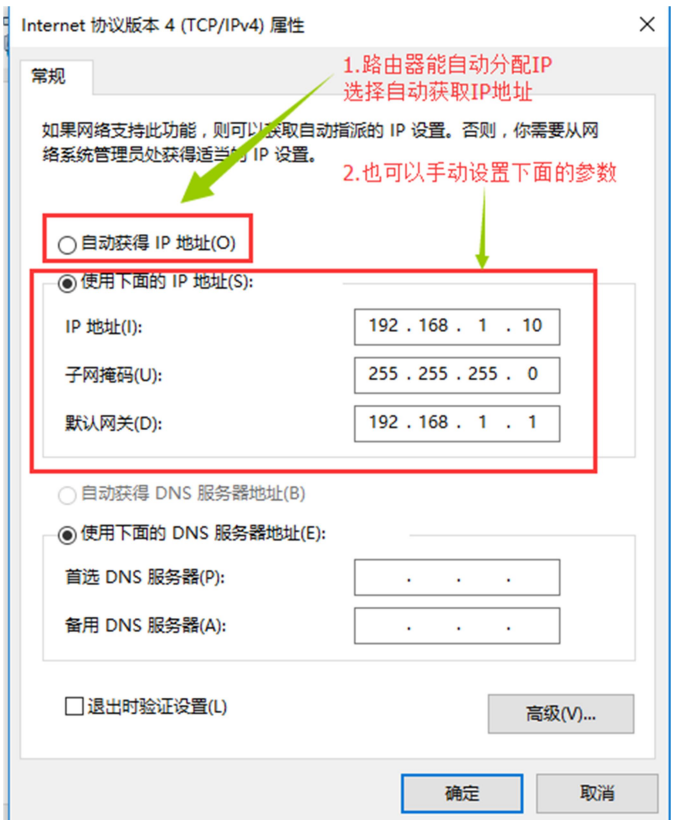
2. 设置显示屏参数（菜单栏➔设置➔屏参设置➔密码 888）



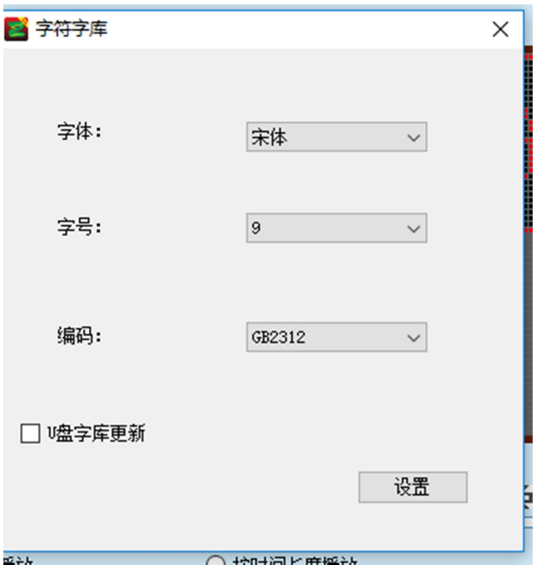
3. 设置字符卡设备 IP 参数（菜单栏➔设置➔屏参设置➔选择指定 IP➔控制卡网络参数配置➔密码 888）



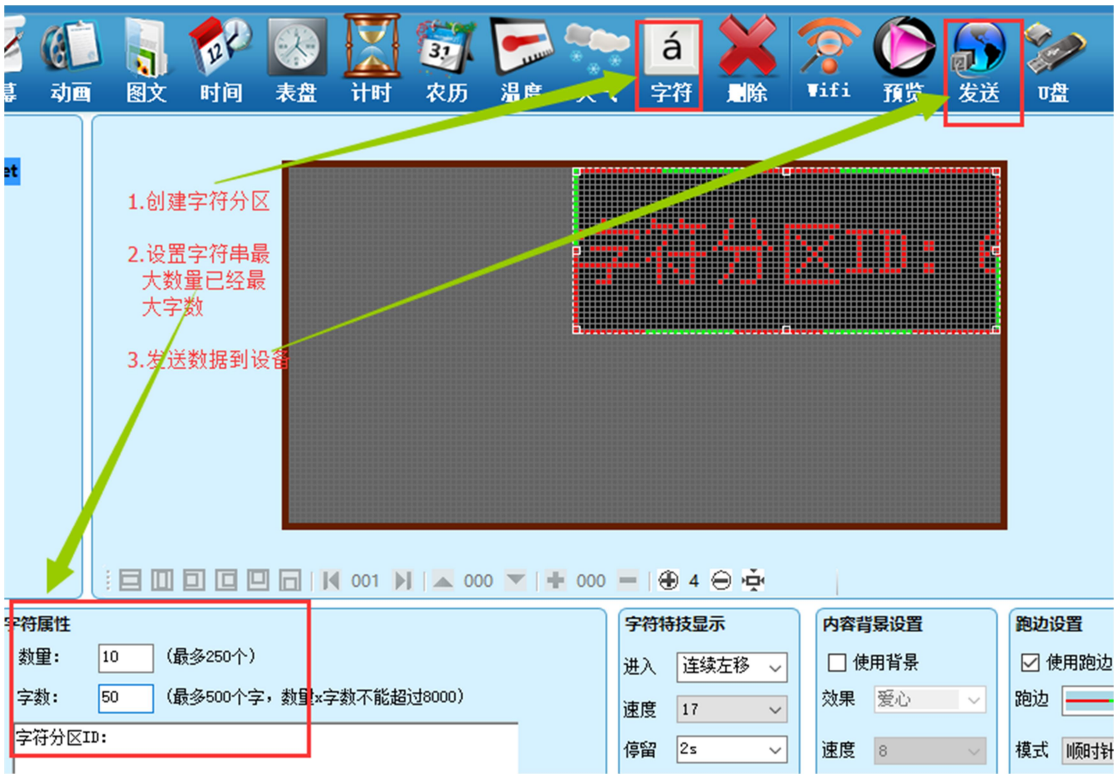
4. 将控制卡插入路由器下，并且将电脑连接同个路由器，并配置电脑 IP 参数



5 发送字库数据（菜单栏➔编辑➔字符字库➔设置）



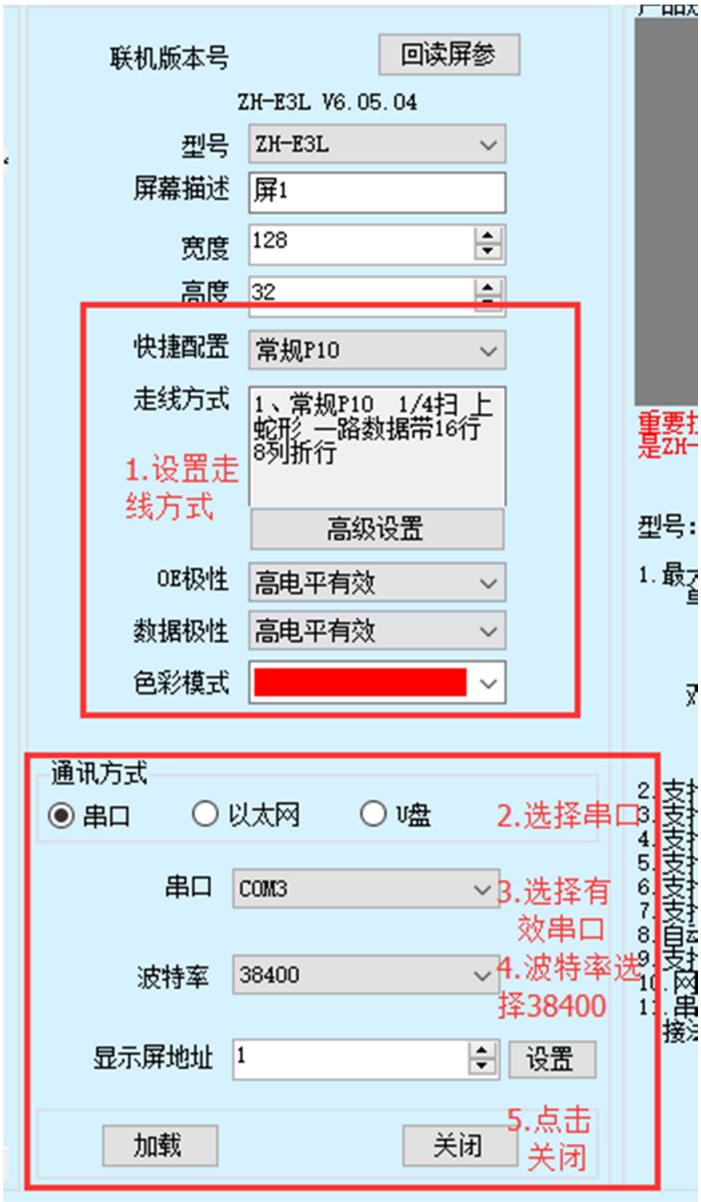
6 创建字符分区并发送数据到字符卡设备上



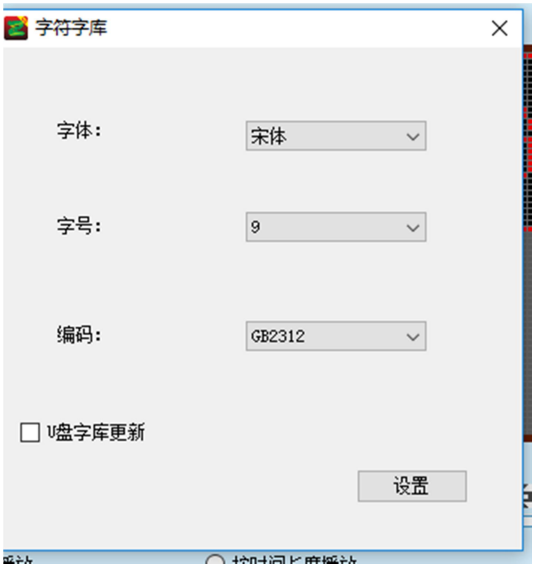
三、 字符卡串口通信使用步骤

1. 将 232 串口线一端接字符卡，一端接电脑

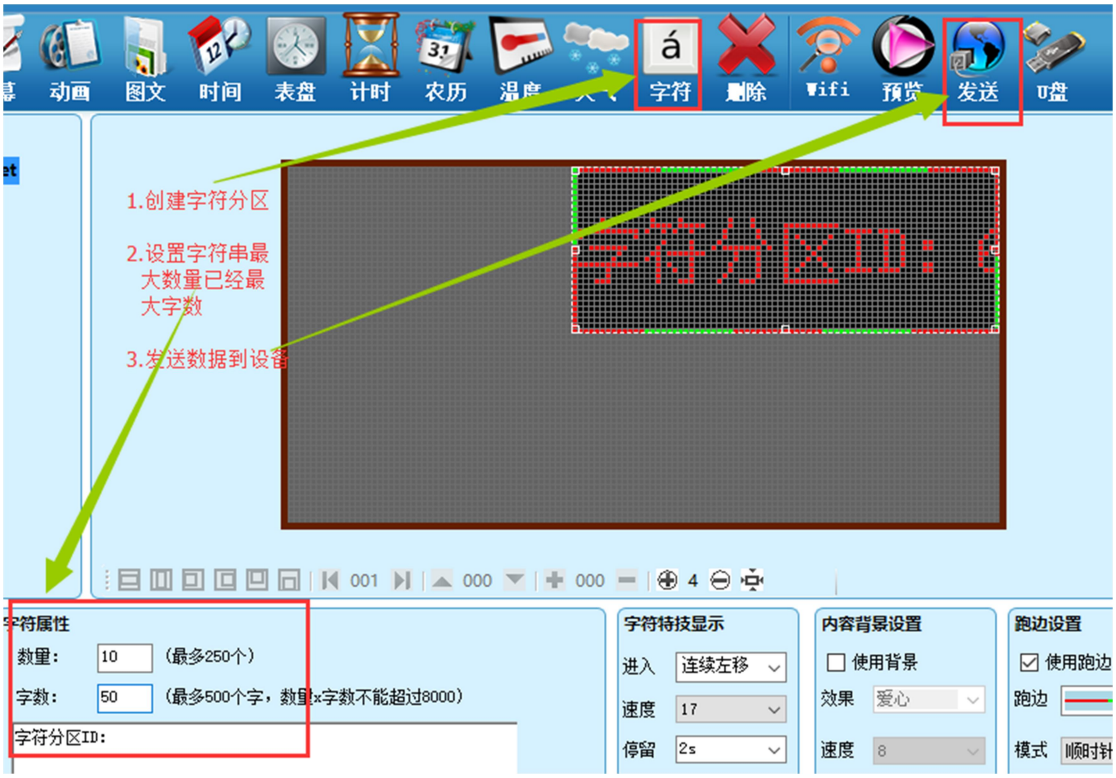
2. 设置显示屏参数（菜单栏➔设置➔屏参设置➔密码 888）



3. 发送字库数据（菜单栏➔编辑➔字符字库➔设置）



4. 创建字符分区并发送数据到字符卡设备上



提示：

1. 若设备上没有字库或者字库被破坏，字符分区内显示"Font Err"。
2. 若字库编码格式与字符串编码格式不一致，字符分区内显示"Code Err"。
3. 若设备内没有保存的字符串数据，字符串分区内显示"字符分区 ID："。
4. 本公司字符卡上位机软件需要定制版本，通用版本上位机软件无字符分区。
5. 本公司字符卡设备是定制版本（ V6.50.03），通用版本无字符分区功能。
6. 字符卡不能使用本地服务器模式以及远程服务器模式。

附二：发送命令控制字符卡例程

1. 以下数据为十六进制（可通过串口或者 TCP 调试软件直接发送相关命令）
2. 网络发送的数据比串口发送的数据多了十个字节（FF FF FF FF FF FF 00 00 00 00）
3. 红色是 cmd 字段
4. 紫色是 Len 字段
5. 蓝色是 data 字段
6. 黄色是 check 字段

发送一条 gb2312 编码 索引为 6 保存数据模式 显示红色 “我爱中航软件” 的字符串 **(回复错误：字符串索引超过最大数量)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 29 BC FD 00 00 00 00 00 00 12 00 01 00 06 01 0C 00 CE D2 B0 AE D6 D0 BA BD C8 ED BC FE F9 6A A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 29 BC FD 00 00 00 00 00 00 01 00 01 00 00 6A 39 A5
	串口发送	78 34 01 00 29 BC FD 00 00 00 00 00 00 12 00 01 00 06 01 0C 00 CE D2 B0 AE D6 D0 BA BD C8 ED BC FE F9 6A A5
	串口回复	79 34 01 00 29 BC FD 00 00 00 00 00 00 01 00 01 00 00 6A 39 A5
发送一条 gb2312 编码，索引为 0，保存数据模式，显示绿色 “1234567” 的字符串 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 29 12 F2 00 00 00 00 00 00 14 00 01 00 00 02 0E 00 00 31 00 32 00 33 00 34 00 35 00 36 00 37 6D 84 A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 29 12 F2 00 00 00 00 00 00 00 00 00 00 06 04 A5
	串口发送	78 34 01 00 29 12 F2 00 00 00 00 00 00 14 00 01 00 00 02 0E 00 00 31 00 32 00 33 00 34 00 35 00 36 00 37 6D 84 A5
	串口回复	79 34 01 00 29 12 F2 00 00 00 00 00 00 00 00 00 00 06 04 A5
发送一条 gb2312 编码 索引为 0 立即显示模式 显示黄色 “Led 显示 5.3” 的字符串 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 29 12 F2 00 00 00 00 00 00 16 00 01 02 00 02 10 00 00 4C 00 65 00 64 CF D4 CA BE 00 35 00 2E 00 33 1E D7 A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 29 12 F2 00 00 00 00 00 00 00 00 00 00 06 04 A5
	串口发送	78 34 01 00 29 12 F2 00 00 00 00 00 00 16 00 01 02 00 02 10 00 00 4C 00 65 00 64 CF D4 CA BE 00 35 00 2E 00 33 1E D7 A5
	串口回复	79 34 01 00 29 12 F2 00 00 00 00 00 00 00 00 00 00 06 04 A5

删除立即显示模式的字符串 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 02 00 52 8F A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
	串口发送	78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 02 00 52 8F A5
	串口回复	79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
删除索引等于 6 的字符串数据 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 00 06 D3 ED A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
	串口发送	78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 00 06 D3 ED A5
	串口回复	79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
删除所有的保存数据模式下的数据 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 00 FF 13 AF A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
	串口发送	78 34 01 00 2A 12 F2 00 00 00 00 00 00 02 00 00 FF 13 AF A5
	串口回复	79 34 01 00 2A 12 F2 00 00 00 00 00 00 00 00 00 00 03 C7 A5
切换显示索引为 7 的字符串数据 **(回复错误：长度字段等于 1,不存在索引为 7 的字符串数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 00 07 20 B8 A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 3F 12 F2 00 00 00 00 00 00 01 00 00 00 00 AE 14 A5
	串口发送	78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 00 07 20 B8 A5
	串口回复	79 34 01 00 3F 12 F2 00 00 00 00 00 00 01 00 00 00 00 AE 14 A5

切换显示索引为 9 的字符串数据 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 00 09 A1 7C A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
	串口发送	78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 00 09 A1 7C A5
	串口回复	79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
切换显示上个字符串数据 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 02 00 60 1A A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
	串口发送	78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 02 00 60 1A A5
	串口回复	79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
	若当前存在的字符串索引（0,2,6,8,9） 1. 当前显示索引 6 则收到命令后播放索引 2 2. 当前显示索引 9 则收到命令后播放索引 8 3. 当前显示索引 0 则收到命令后播放索引 9	
切换显示下个字符串数据 **(回复正确：长度字段等于 0,无 data 字段数据)**	网络发送	FF FF FF FF FF FF 00 00 00 00 78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 01 00 60 EA A5
	网络回复	5A 48 05 51 67 1F 00 00 00 00 79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
	串口发送	78 34 01 00 3F 12 F2 00 00 00 00 00 00 02 00 01 00 60 EA A5
	串口回复	79 34 01 00 3F 12 F2 00 00 00 00 00 00 00 00 00 00 31 52 A5
	若当前存在的字符串索引（0,2,6,8,9） 1. 当前显示索引 6 则收到命令后播放索引 8 2. 当前显示索引 9 则收到命令后播放索引 0 3. 当前显示索引 0 则收到命令后播放索引 2	

附三：CRC 校验函数

Little-Endian Mode

```
/*
message：data pointer
len      ：data length
return ：32bits CRC Value
*/
uint32 GetCRC(void* message, uint16    len)
{
    uint32 CRCFull = 0xFFFF;
    uint8 CRCLSB;
    int i = 0, j = 0;
    uint8 *mess = message;
    for (i = 0; i < len; i++)
    {
        CRCFull = (uint16)(CRCFull ^ mess[i]);
        for (j = 0; j < 8; j++)
        {
            CRCLSB = (uint8)(CRCFull & 0x0001);
            CRCFull = (uint16)((CRCFull >> 1) & 0x7FFF);
            if (CRCLSB == 1)
                CRCFull = (uint16)(CRCFull ^ 0xA001);
        }
    }
    return CRCFull;
}
```