

Modbus RTU Relay

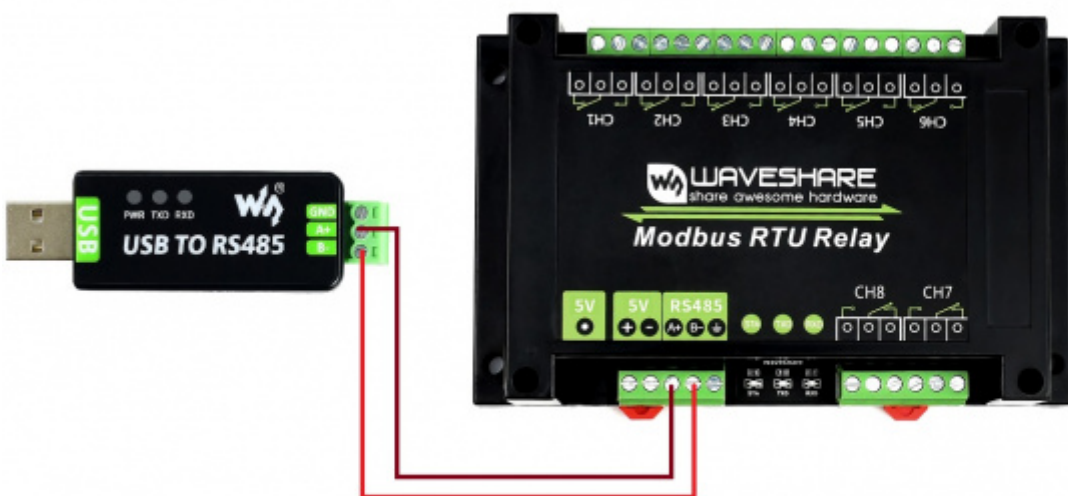
来自Waveshare Wiki
跳转至： [导航](#)、 [搜索](#)

说明

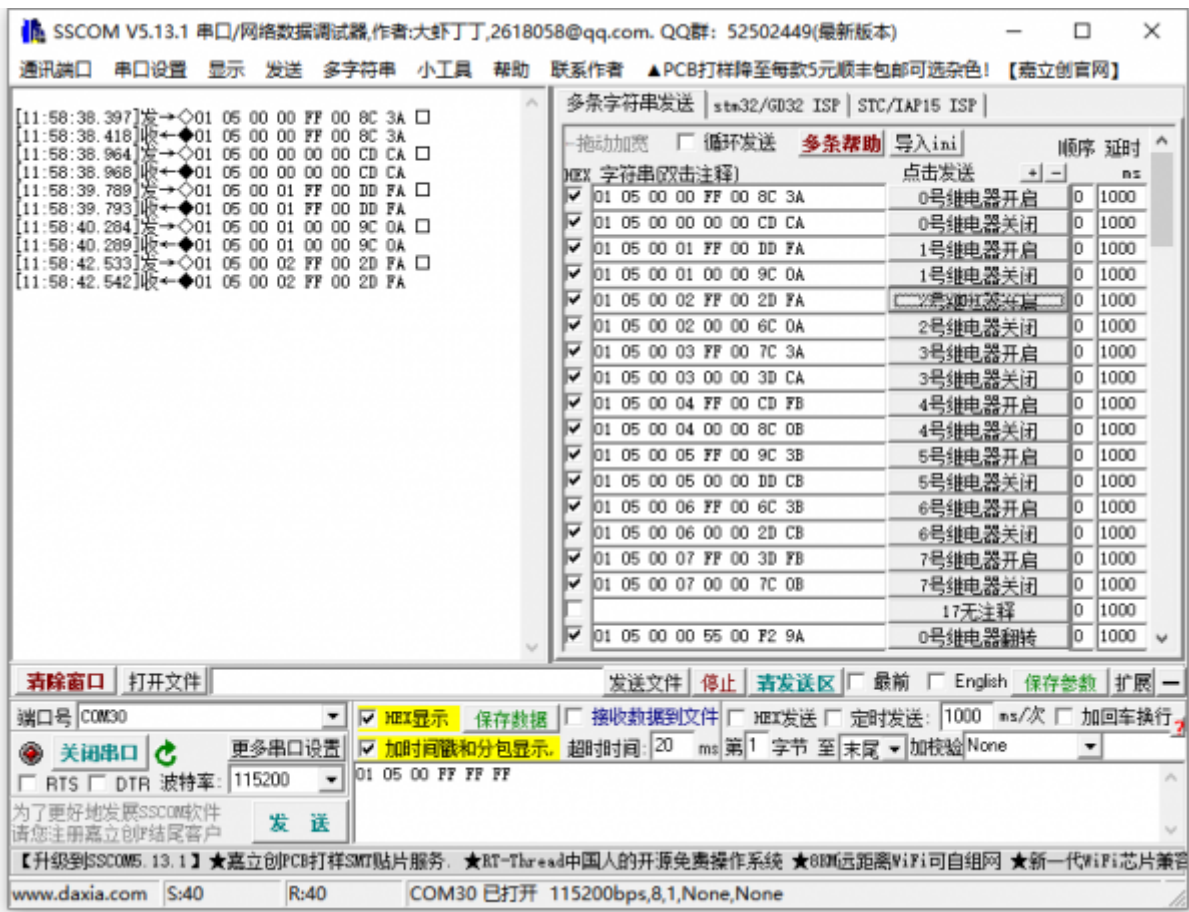
硬件测试

RS485测试

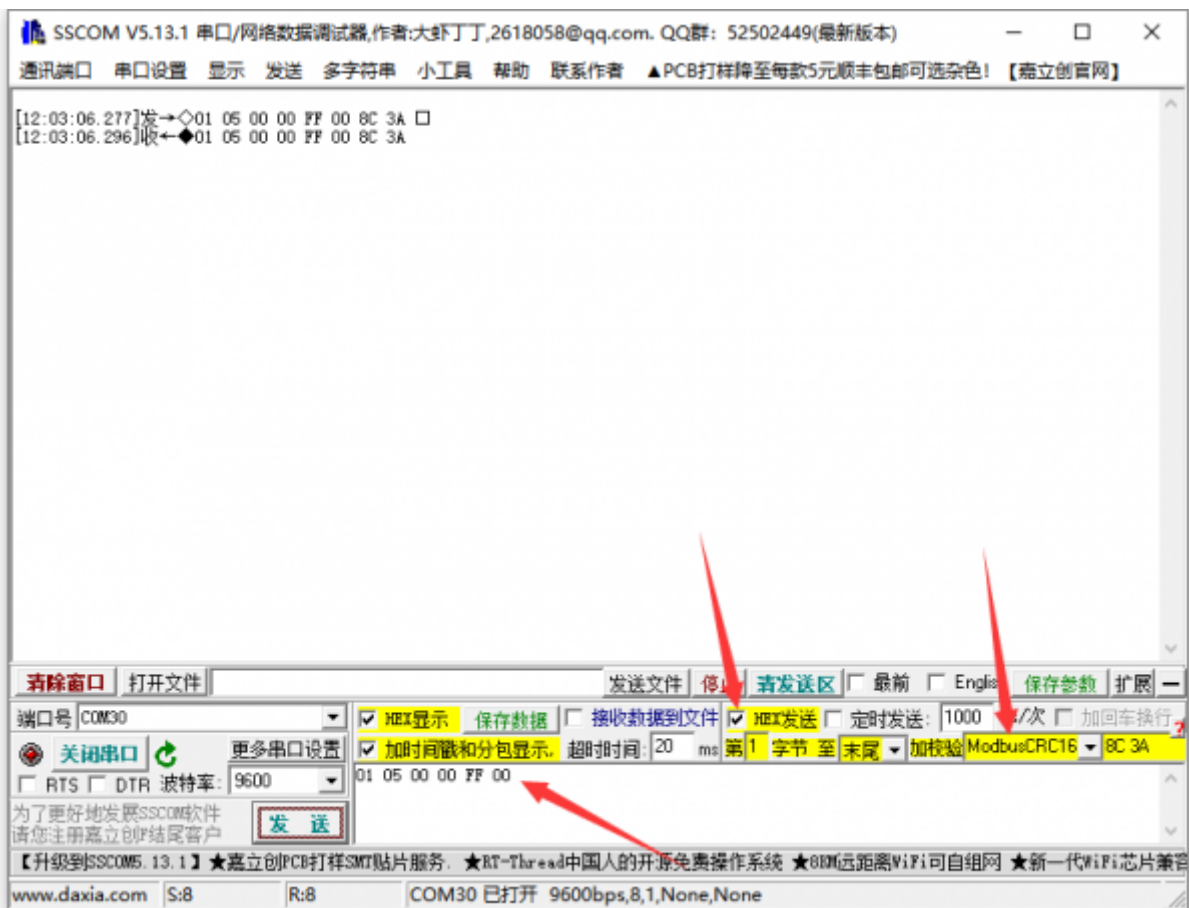
- 将USB TO 485和目标板通过杜邦线，将A-->A和B-->B相连，如下图所示：



- 下载SSCOM串口调试助手 并在电脑上打开，打开对应的端口号，设置波特率为9600，点击多字符串打开多条字符串发送窗口，点击对应的功能即可发送对应的命令，



- 如果需要发送其他命令则选择HEX发送，加检验选择ModbusCRC16校验，输入命令的前六个字节点击发送则会自动添加CRC校验码。



- 详细的控制指令请查看请查看开发协议。

2021年9月起，将采用V2版本开发协议，V2版本修改之前不兼容Modbus指令，完全兼容Modbus协议。

V1,V2版本硬件无区别，之前购买的客户需要采用V2开发协议可联系客服升级固件。

程序测试


注意：RS485不可以直接接到树莓派的串口否则可能烧毁设备，需要加485电平转换，树莓派建议搭配RS485 CAN HAT模块使用，NUCLEO-F103RB和Arduino建议搭配RS485 CAN Shield模块使用

树莓派

打开树莓派终端，输入以下指令进入配置界面

```
sudo raspi-config
```

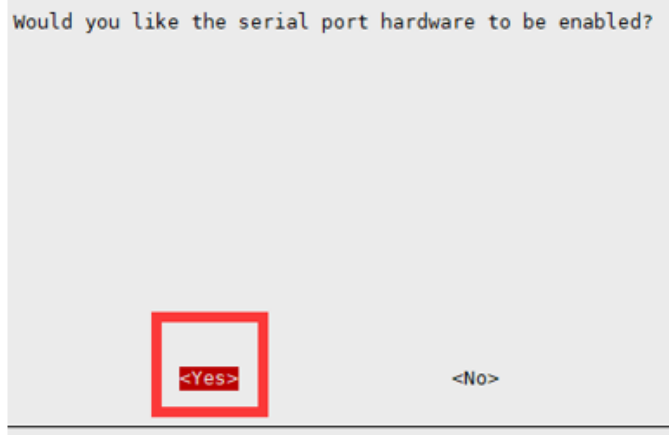
选择Interfacing Options -> Serial, 关闭shell访问, 打开硬件串口



```
Would you like a login shell to be accessible over serial?
```

<Yes>

<No>



```
Would you like the serial port hardware to be enabled?
```

<Yes>

<No>

然后重启树莓派：

```
sudo reboot
```

打开/boot/config.txt文件，找到如下配置语句使能串口，如果没有，可添加在文件最后面。

```
enable_uart=1
```

对于树莓派3B用户，串口用于蓝牙，需要注释掉：

```
#dtoverlay=pi3-miniuart-bt
```

然后重启树莓派：

```
sudo reboot
```

将RS485 CAN HAT模块插到树莓派，把ModBus RTU Relay模块和RS485 CAN HAT模块的A与B对应连接

如果使用的是其他的485设备，确定连线A-A,B-B即可

运行如下命令运行程序

```
sudo apt-get install unzip
wget https://www.waveshare.net/w/upload/f/f9/Modbus_RTU_Relay_Code.zip
unzip Modbus_RTU_Relay_Code.zip
cd Modbus_RTU_Relay_Code/Python3
sudo python3 main.py
```

STM32

STM32的例程是基于NUCLEO-F103RB和RS485 CAN Shield模块。

找到STM32程序文件目录并打开STM32工程。注意使用前先确保电脑已经安装好keil5软件，下载将程序下载到开发板即可。

正常正常运行继电器模块导通会依次打开然后依次关闭。串口会输出发送的命令。

Arduino

Arduino例程是基于NUO PLUS和RS485 CAN Shield模块。

使用Arduino IDE 打开程序，选择对应的开发板然后下载程序。

正常正常运行继电器模块导通会依次打开然后依次关闭。串口会输出发送的命令。

开发协议 V2

2021年9月起，将采用V2版本开发协议，V2版本修改之前不兼容Modbus指令，完全兼容Modbus协议。

V1,V2版本硬件无区别，之前购买的客户需要采用V2开发协议可联系客服升级固件。

功能码介绍

功能码	备注
01	读继电器状态
03	读取地址、版本
05	写单个继电器
06	设置波特率, 地址
0F	写全部继电器

控制单个继电器

发送码: 01 05 00 00 FF 00 8C 3A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 00	地址	要控制继电器的寄存器地址, 0x00 - 0x0008
FF 00	指令	0xFF00: 继电器开启; 0x0000: 继电器关闭; 0x5500: 继电器翻转;
8C 3A	CRC16	前6字节数据的CRC16校验和

返回码: 01 05 00 00 FF 00 8C 3A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 00	地址	要控制继电器寄存器地址, 0x0000-0x0008
FF 00	指令	0xFF00: 继电器开启; 0x0000: 继电器关闭; 0x5500: 继电器翻转;
8C 3A	CRC16	前6字节数据的CRC16校验和

示例: [1号地址设备]: 0号继电器开启 : 01 05 00 00 FF 00 8C 3A

0号继电器关闭 : 01 05 00 00 00 00 CD CA

1号继电器开启 : 01 05 00 01 FF 00 DD FA

1号继电器关闭 : 01 05 00 01 00 00 9C 0A

2号继电器开启 : 01 05 00 02 FF 00 2D FA

2号继电器关闭 : 01 05 00 02 00 00 6C 0A

3号继电器开启 : 01 05 00 03 FF 00 7C 3A

3号继电器关闭 : 01 05 00 03 00 00 3D CA

0号继电器翻转: 01 05 00 00 55 00 F2 9A

1号继电器翻转：01 05 00 01 55 00 A3 5A

2号继电器翻转：01 05 00 02 55 00 53 5A

3号继电器翻转：01 05 00 03 55 00 02 9A

控制全部继电器

发送码：01 05 00 FF FF 00 BC 0A

BC 0A||CRC16||前6字节数据的CRC16校验和

字段	含义	备注
01	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 FF	地址	固定0x00FF
FF 00	指令	0xFF00：继电器开启； 0x0000：继电器关闭； 0x5500：继电器翻转；

返回码：01 05 00 FF FF 00 BC 0A

字段	含义	备注
01	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 FF	地址	固定0x00FF
FF 00	指令	0xFF00：继电器开启； 0x0000：继电器关闭； 0x5500：继电器翻转；
BC 0A	CRC16	前6字节数据的CRC16校验和

示例：

[1号地址设备]：

全部继电器开启：01 05 00 FF FF 00 BC 0A

全部继电器关闭：01 05 00 FF 00 00 FD FA

全部继电器翻转：01 05 00 FF 55 00 C2 AA

读取继电器状态

发送码：01 01 00 00 00 08 3D CC

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
01	01指令	查询继电器状态指令
00 00	继电器起始地址	固定0x0000
00 08	继电器数量	固定0x0008
3D CC	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 01 01 00 51 88

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
01	01指令	查询继电器状态指令
01	字节数	返回状态信息的所有字节数。
00	查询的状态	返回的继电器状态 Bit0:第一个继电器状态; Bit1:第二个继电器状态; Bit2:第三个继电器状态; Bit7:第八个继电器状态;
8C 35	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

发送: 01 01 00 00 00 08 3D CC

返回: 01 01 01 00 51 88 //全部继电器关闭

发送: 01 01 00 00 00 08 3D CC

返回: 01 01 01 01 90 48 //0号继电器开启, 其余继电器关闭

发送: 01 01 00 00 00 08 3D CC

返回: 01 01 01 41 91 B8 //0、6号继电器开启, 其余继电器关闭

写继电器状态

发送码: 01 0F 00 00 00 08 01 FF BE D5

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
0F	0F指令	写继电器状态指令
00 00	继电器起始地址	固定 0x0000
00 08	继电器数量	固定 0x0008
01	字节数量	固定 0x01
FF	继电器状态	Bit0:控制第一个继电器; Bit1:控制第二个继电器; Bit2:控制第三个继电器; Bit7:控制第八个继电器;
BE D5	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 0F 00 00 00 01 94 0B

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
0F	0F指令	全部寄存器控制指令
00 00	地址	固定0x0000
00 08	继电器数量	固定 0x0008
54 0D	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

继电器全部开启 : 01 0F 00 00 00 08 01 FF BE D5

继电器全部关闭 : 01 0F 00 00 00 08 01 00 FE 95

0-1开; 3-7关 : 01 0F 00 00 00 08 01 03 BE 94

继电器闪开闪闭指令

发送码: 01 05 02 00 00 07 8D B0

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	单个控制指令
02	指令	02 为是闪开指令, 04为闪闭指令
00	继电器地址	要控制的继电器地址, 0x00~0x08
00 07	间隔时间	延时时间为数据*100ms 数值:0x0007, 延时:7*100MS = 700MS
8D B0	CRC16	前 6 字节数据的 CRC16 校验和

返回码：01 05 02 00 00 07 8D B0

字段	含义	备注
01	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
05	05指令	单个控制指令
02	指令	02 为是闪开指令，04为闪闭指令
00	继电器地址	要控制继电器地址，0x00~0x08
00 07	间隔时间	延时时间为数据*100ms 数值:0x0007，延时:7*100MS = 700MS
8D B0	CRC16	前 6 字节数据的 CRC16 校验和

示例：

[1号地址设备]

0号继电器闪开：01 05 02 00 00 07 8D B0 //700MS = 7*100MS = 700MS

1号继电器闪开：01 05 02 01 00 08 9C 74 //800MS

0号继电器闪闭：01 05 04 00 00 05 0C F9 //500MS

1号继电器闪闭：01 05 04 01 00 06 1D 38 //600MS

设置波特率指令

发送码：00 06 20 00 00 05 43 D8

字段	含义	备注
00	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
06	06指令	设置波特率，设备地址
20 00	指令寄存器	0x2000为设置波特率，0x4000为设置设备地址
00	校验方式	0x00为无校验，0x01为偶检验，0x02为奇校验
05	波特率数值	波特率数值对应 0x00 : 4800 0x01 : 9600 0x02 : 19200 0x03 : 38400 0x04 : 57600 0x05 : 115200 0x06 : 128000 0x07 : 256000
43 D8	CRC16	前 6 字节数据的 CRC16 校验和

返回码：00 06 20 00 00 05 43 D8

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
20 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00	校验方式	0x00为无校验, 0x01为奇检验, 0x02为偶校验
05	波特率	波特率数值对应 0x00 : 4800 0x01 : 9600 0x02 : 19200 0x03 : 38400 0x04 : 57600 0x05 : 115200 0x06 : 128000 0x07 : 256000
43 D8	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

设置波特率 4800 : 00 06 20 00 00 00 83 DB

设置波特率 9600 : 00 06 20 00 00 01 42 1B

设置波特率115200 : 00 06 20 00 00 05 43 D8

设置设备地址指令

发送码: 00 06 40 00 00 01 5C 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
40 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00 01	设备地址	设置的设备地址,0x0001-0x00FF
5C 1B	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 00 06 40 00 00 01 5C 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
40 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00 01	设备地址	设置的设备地址,0x0001-0x00FF
5C 1B	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

设置设备地址为0x01 : 00 06 40 00 00 01 5C 1b

设置设备地址为0x02 : 00 06 40 00 00 02 1C 1A

设置设备地址为0x03 : 00 06 40 00 00 03 DD DA

读取设备地址指令

发送码: 00 03 40 00 00 01 90 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取设备地址指令
40 00	指令寄存器	0x0200为读取软件版本, 0x0040为读取设备地址
00 01	字节数	固定0x0001
90 1B	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 03 02 00 01 79 84

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
02	字节数	返回字节数
00 01	设备地址	设置的设备地址,0x0001-0x00FF
79 84	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 01 03 02 00 01 79 84 //地址0x01

[2号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 02 03 02 00 02 7D 85 //地址0x02

[3号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 03 03 02 00 03 81 85 //地址0x03

读取软件版本指令

发送码: 00 03 80 00 00 01 AC 1B

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
80 00	指令寄存器	0x4000为读取设备地址, 0x8000为读取软件版本
00 01	字节数	固定0x0001
8F CA	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 03 02 00 C8 F0 B8

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
02	字节数	返回字节数
00 C8	软件版本	转为十进制然后小数点左移两位即表示软件版本 0x00C8 = 200 = V2.00
F0 B8	CRC16	前 6 字节数据的 CRC16 校验和

示例:

发送: 00 03 80 00 00 01 AC 1B

返回: 03 03 02 00 C8 F1 00 //0x00C8 = 200 =V2.00

开发协议 V1

2021年9月起, 将采用V2版本开发协议, V2版本修改之前不兼容Modbus指令, 完全兼容Modbus协议。

V1,V2版本硬件无区别, 之前购买的客户需要采用V2开发协议可联系客服升级固件。

发送指令格式

发送指令总共包括8个字节, 分别是

字节1: 地址

字节2: 功能码

字节3 4: 寄存器地址 (大端格式)

字节5 6: 寄存器数据 (大端格式)

字节7 8: CRC校验 (小端格式)

功能码介绍

功能码	备注
01	读继电器状态
03	读取地址、版本
05	写单个继电器
06	设置波特率, 地址
0F	写全部继电器

控制单个继电器

发送码: 01 05 00 00 FF 00 8C 3A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 00	地址	要控制继电器的寄存器地址, 0x00 - 0x0008
FF 00	指令	0xFF00: 继电器开启; 0x0000: 继电器关闭; 0x5500: 继电器翻转;
8C 3A	CRC16	前6字节数据的CRC16校验和

返回码: 01 05 00 00 FF 00 8C 3A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 00	地址	要控制继电器寄存器地址, 0x0000-0x0008
FF 00	指令	0xFF00: 继电器开启; 0x0000: 继电器关闭; 0x5500: 继电器翻转;
8C 3A	CRC16	前6字节数据的CRC16校验和

示例: [1号地址设备]: 0号继电器开启 : 01 05 00 00 FF 00 8C 3A

0号继电器关闭 : 01 05 00 00 00 00 CD CA

1号继电器开启 : 01 05 00 01 FF 00 DD FA

1号继电器关闭 : 01 05 00 01 00 00 9C 0A

2号继电器开启 : 01 05 00 02 FF 00 2D FA

2号继电器关闭 : 01 05 00 02 00 00 6C 0A

3号继电器开启 : 01 05 00 03 FF 00 7C 3A

3号继电器关闭 : 01 05 00 03 00 00 3D CA

0号继电器翻转: 01 05 00 00 55 00 F2 9A

1号继电器翻转: 01 05 00 01 55 00 A3 5A

2号继电器翻转: 01 05 00 02 55 00 53 5A

3号继电器翻转: 01 05 00 03 55 00 02 9A

控制全部继电器

发送码: 01 05 00 FF FF FC 4A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 FF	地址	固定0x00FF
FF FF	指令	0xFFFF: 继电器开启;
FC 4A	CRC16	前6字节数据的CRC16校验和

返回码: 01 05 00 FF FF FC 4A

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	控制继电器指令
00 FF	地址	固定0x00FF
FF FF	指令	0xFFFF: 继电器开启; 0x0000: 继电器关闭; 0x5A00: 继电器翻转;
FC 4A	CRC16	前6字节数据的CRC16校验和

示例:

[1号地址设备]:

全部继电器开启: 01 05 00 FF FF FC 4A

全部继电器关闭: 01 05 00 FF 00 00 FD FA

全部继电器翻转: 01 05 00 FF 5A 00 C7 5A

读取继电器状态

发送码: 01 01 00 FF 00 01 CD FA

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
01	01指令	查询继电器状态指令
00 FF	地址	固定0x00FF
00 01	指令	固定0x0001
CD FA	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 01 01 00 51 88

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
01	01指令	查询继电器状态指令
01	字节数	返回状态信息的所有字节数。
00	查询的状态	返回的继电器状态 Bit0:第一个继电器状态; Bit1:第二个继电器状态; Bit2:第三个继电器状态; Bit7:第八个继电器状态;
8C 35	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

发送: 01 01 00 FF 00 01 CD FA

返回: 01 01 01 00 51 88 //全部继电器关闭

发送: 01 01 00 FF 00 01 CD FA

返回: 01 01 01 01 90 48 //0号继电器开启, 其余继电器关闭

发送: 01 01 00 FF 00 01 CD FA

返回: 01 01 01 41 91 B8 //0、6号继电器开启, 其余继电器关闭

写继电器状态

发送码: 01 0F 00 00 00 01 94 0B

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
0F	0F指令	写继电器状态指令
00 00	地址	固定0x0000
00 01	指令	Bit0:控制第一个继电器; Bit1:控制第二个继电器; Bit2:控制第三个继电器; Bit7:控制第八个继电器;
94 0B	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 0F 00 00 00 01 94 0B

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
0F	0F指令	全部寄存器控制指令
00 00	地址	固定0x0000。
00 01	指令	0x0001, 写继电器数据 Bit0:控制第一个继电器; Bit1:控制第二个继电器状态; Bit2:控制第三个继电器状态; Bit7控制:第八个继电器状态;
94 0B	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

继电器全部开启 : 01 0F 00 00 00 FF 15 8B

继电器全部关闭 : 01 0F 00 00 00 00 55 CB

0-1开; 3-7关 : 01 0F 00 00 00 03 15 CA

继电器闪开闪闭指令

发送码: 01 05 02 00 00 07 8D B0

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
05	05指令	单个控制指令
02	指令	02 为是闪开指令, 04为闪闭指令
00	继电器地址	要控制的继电器地址
00 07	间隔时间	延时时间为数据*100ms 数值:0x0007, 延时:7*100MS = 700MS
8D B0	CRC16	前 6 字节数据的 CRC16 校验和

返回码：01 05 02 00 00 07 8D B0

字段	含义	备注
01	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
05	05指令	单个控制指令
02	指令	02 为是闪开指令，04为闪闭指令
00	继电器地址	要控制继电器地址
00 07	间隔时间	延时时间为数据*100ms 数值:0x0007, 延时:7*100MS = 700MS
8D B0	CRC16	前 6 字节数据的 CRC16 校验和

示例：

[1号地址设备]

0号继电器闪开：01 05 02 00 00 07 8D B0 //700MS = 7*100MS = 700MS

1号继电器闪开：01 05 02 01 00 08 9C 74 //800MS

0号继电器闪闭：01 05 04 00 00 05 0C F9 //500MS

1号继电器闪闭：01 05 04 01 00 06 1D 38 //600MS

设置波特率指令

发送码：00 06 20 00 00 05 43 D8

字段	含义	备注
00	设备地址	0x00表示广播地址；0x01-0xFF表示设备地址
06	06指令	设置波特率，设备地址
20 00	指令寄存器	0x2000为设置波特率，0x4000为设置设备地址
00 05	波特率数值	波特率数值对应 0x0000 : 4800 0x0001 : 9600 0x0002 : 19200 0x0003 : 38400 0x0004 : 57600 0x0005 : 115200 0x0006 : 128000 0x0007 : 256000
43 D8	CRC16	前 6 字节数据的 CRC16 校验和

返回码：00 06 20 00 00 05 43 D8

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
20 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00 05	波特率	波特率数值对应 0x0000 : 4800 0x0001 : 9600 0x0002 : 19200 0x0003 : 38400 0x0004 : 57600 0x0005 : 115200 0x0006 : 128000 0x0007 : 256000
43 D8	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

设置波特率 4800 : 00 06 20 00 00 00 83 DB

设置波特率 9600 : 00 06 20 00 00 01 42 1B

设置波特率115200 : 00 06 20 00 00 05 43 D8

设置设备地址指令

发送码: 00 06 40 00 00 01 5C 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
40 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00 01	设备地址	设置的设备地址,0x0001-0x00FF
5C 1B	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 00 06 40 00 00 01 5C 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
06	06指令	设置波特率, 设备地址
40 00	指令寄存器	0x2000为设置波特率, 0x4000为设置设备地址
00 01	设备地址	设置的设备地址,0x0001-0x00FF
5C 1B	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

设置设备地址为0x01 : 00 06 40 00 00 01 5C 1b

设置设备地址为0x02 : 00 06 40 00 00 02 1C 1A

设置设备地址为0x03 : 00 06 40 00 00 03 DD DA

读取设备地址指令

发送码: 00 03 40 00 00 01 90 1B

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取设备地址指令
40 00	指令寄存器	0x0200为读取软件版本, 0x0040为读取设备地址
00 01	设备地址	设备地址
90 1B	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 03 01 01 31 88

字段	含义	备注
00	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
01	字节数	返回字节数
01	设备地址	设备地址
31 88	CRC16	前 6 字节数据的 CRC16 校验和

示例:

[1号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 01 03 01 01 31 88 //地址0x01

[2号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 02 03 01 02 71 CD //地址0x02

[3号地址设备]

发送 : 00 03 40 00 00 01 90 1B

返回 : 03 03 01 03 B1 F1 //地址0x03

读取软件版本指令

发送码: 01 03 20 00 00 01 8F CA

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
20 00	指令寄存器	0x0200为读取软件版本, 0x0040为读取设备地址
00 01	设备地址	设备地址
8F CA	CRC16	前 6 字节数据的 CRC16 校验和

返回码: 01 03 01 64 F1 A3

字段	含义	备注
01	设备地址	0x00表示广播地址; 0x01-0xFF表示设备地址
03	03指令	读取软件版本, 读取设备地址指令
01	字节数	返回字节数
64	软件版本	转为十进制然后小数点左移两位即表示软件版本 0x64 = 100 = V1.00
F1 A3	CRC16	前 6 字节数据的 CRC16 校验和

示例:

发送: 01 03 20 00 00 01 8F CA

返回: 01 03 01 64 F1 A3 // 64 = 100 = V1.00

资料

程序

- 示例程序

软件

- Sscom5.13.1_for_Modbus_RTU_Relay_V2
- Sscom5.13.1_for_Modbus_RTU_Relay

FAQ

问题: 在非windows系统上使用不了, 如何解决?

- 如果遇到windows以外的系统无法正常使用时，可以进入下方官网链接，安装对应系统驱动后使用：<https://www.ftdichip.cn/Drivers/D2XX.htm>

问题：支持Linux系统么？

- 多系统兼容，支持Win7/8/8.1/10、Mac、Linux、Android、WinCE等系统。

问题：RS485近距离通信异常怎么办？

- 去掉RS485的120欧姆终端匹配电阻。

问题：支持USB同时转RS232,RS485和TTL吗？

- 不支持，只能USB转三个中的一个，USB转RS232,RS485或者TTL。

问题：支持RS232,RS485和TTL互转吗？

- 不支持，RS232,RS485和TTL不能互转，支持USB转RS232,RS485或者TTL。