MZR-T08R 8通道电阻测量模块

使用说明书(V3.2)

1、概述

本模块采用高精度32位AD芯片+ARM32位工业级MCU，精度高，抗干扰好。可精确测量0.02Ω至10MΩ电阻，精度最高可达到千分之一，可广泛用于各种电阻测量场合。

具有以下特点：

- ✧ 具有宽电源供电9-30V。
- ✧ 32位高精度AD，高分辨率达1mΩ，精度最高达千分之一。
- ✧ 出厂默认2线制测量，如果被测电阻引线较长，可以采用3线制测量，精度更高；
- ✧ 采样周期具有1.3秒、0.6秒、0.35秒、0.2秒 四种速率可设置。
- ✧ 量程可自行设置，可以选0.02Ω至10MΩ自动量程，也可以选固定25Ω、1K、5K、20K、100K、1M、10M七种固定量程。
- ✧ 多个被测电阻有一端短路时，不影响测试效果。
- ✧ 采集输入、电源、通讯三种相互隔离，可靠性高；
- ✧ 可以用RS485或以太网做为通信接口，当采用以太网版本（插座会占用第7与第8通道位置，所以此版本只有6通道）时，可同时使用RS485接口，使模块同时拥有两个通讯口，可用于冗余高可靠场合；
- ✧ 可灵活自选Modbus-RTU或Modbus-TCP工业通信协议，与各种组态屏、工控软件以及模组进行可靠通信。

2、产品主型号

MZR-T08R-34M　　8通道电阻，RS485接口；
MZR-T06R-34Y　　6通道电阻，以太网接口+RS485接口；

3、性能指标

- ➢ 量程：通过修改0x0085寄存器，可以设定测量量程，见表：

表 1：量程设定对照明表

0x0085寄存器中的代码	量程	说明
0 （出厂默认设置）	0.02Ω~10MΩ	自动量程：检测电路自动切换合适档位，确保测量精度，测量慢
1	0.02Ω~25Ω	固定量程： 固定档位，不会自动变档； 相比自动量程检测速度快； 量程越大，误差越大。
2	0.02Ω~1KΩ	
3	0.02Ω~5KΩ	
4	0.02Ω~20KΩ	
5	0.02Ω~100KΩ	
6	0.02Ω~1MΩ	
7	0.02Ω~10MΩ	

- 精度误差：

表 2：精度与更新速率、量程关系表

量程方式	阻值范围或量程	第 1 与第 2 档更新速率精度	第 3 与第 4 档更新速率精度
自动量程	0.02Ω~1MΩ时	>±0.1%	>±1%
	1MΩ~10MΩ时	>±1%	>±10%
固定量程	0. 02Ω~25Ω	±0.006Ω	±0.06Ω
	0. 02Ω~1KΩ	±0.1Ω	±1Ω
	0. 02Ω~5KΩ	±0.5Ω	±5Ω
	0. 02Ω~20KΩ	±2Ω	±20Ω
	0. 02Ω~100KΩ	±10Ω	±100Ω
	0. 02Ω~1MΩ	±100Ω	±1000Ω
	0. 02Ω~10MΩ	±1000Ω	±10000Ω
此测量精度受限于出厂校正系统，实际使用时，精度会更高； 用户可以通过修改 0x02E0~0x02E7 来修正误差，这样精度可以达到更高； 以上数据为采用 2 线制时所测得的精度，如果被测电阻引线较长，可以采用 3 线制，可以避免因引线线阻引起的误差。			

- 数据更新时间

表 3：数据更新速率表(0x0081 寄存器可更改数据更新速度)

量程方式	量程	第 1 档（出厂默认）	第 2 档	第 3 档	第 4 档
固定量程	1MΩ以下量程时	1.3s	0.6s	0.35s	0.2s
	1MΩ或 10M 量程 时	1.4s	0.9s	0.6s	0.45s
自动量程	0.02Ω~10MΩ	自动量程测量时采用的是自动分段测量；测量时间会增加 1 至 2 倍； 自动量程适合做精确慢速测量； 固定量程适合做快速动态测量。			
注意：当采用 3 线制测量时，测量时间会加 1 倍。					

- 分 辨 率：最小可至 1mΩ，多个种类的寄存器存储不同分辨率数据，用户可以按需读取，列表如下：

表 4：寄存器与分辨率关系表

寄存器地址（16 进制）	分辨率	数据位数	能指示的阻值范围	说明
0x0000:0x0001~ 0x000E:0x000F	0.01Ω	32	0.00Ω~10MΩ	两个 16 位寄存器组成一个 32 位寄存器，高 16 位在前，低 16 位在后；显示 0xFFFFFFFF 时，表示超量程或断线
0x1000~0x1007	0.001Ω	16	0.000Ω~65.534Ω	显示 0xFFFF 时表示超显示范围
0x1040~0x1047	0.01Ω	16	0.00Ω~655.34Ω	
0x1080~0x1087	1Ω	16	0Ω~65534Ω	
0x10C0~0x10C7	0.1kΩ	16	0.0kΩ~6553.4kΩ	
0x1100~0x1107	1kΩ	16	0kΩ~10000kΩ	显示 0xFFFF 时表示超量程或断线

- 工作温度：-40℃～+70℃；
- 温度漂移：≤±5ppm/℃；

- 辅助电源：+9V~+30VDC；
- 额定功耗：<1.3W（无以太网口时）；
  <3W（有以太网口时）
- 安装方式：标准 35mm 导轨或螺丝安装，螺丝安装尺寸 105*70mm，Φ5mm；
- 产品重量：约 150g
- 输出接口：RS485 接口或以太网+RS485 接口；
- 通讯协议：Modbus-RTU 或 Modbus-TCP 通讯协议可配置；
- 通讯波特率：4800、9600、19200、38400、57600、115200bps；
- 数据格式： 8 个数据位，可自由配置无校验/奇校验/偶校验、1 位停止位/2 位停止位；
- 隔离耐压：>1500V DC（电源与通讯电路、电源与检测电路、通讯与检测电路相互隔离）；

RS485 口出厂参数: 地址为 1 号,波特率 9600,无校验,8 个数据位，1 个停止位;默认 Modbus-RTU 协议

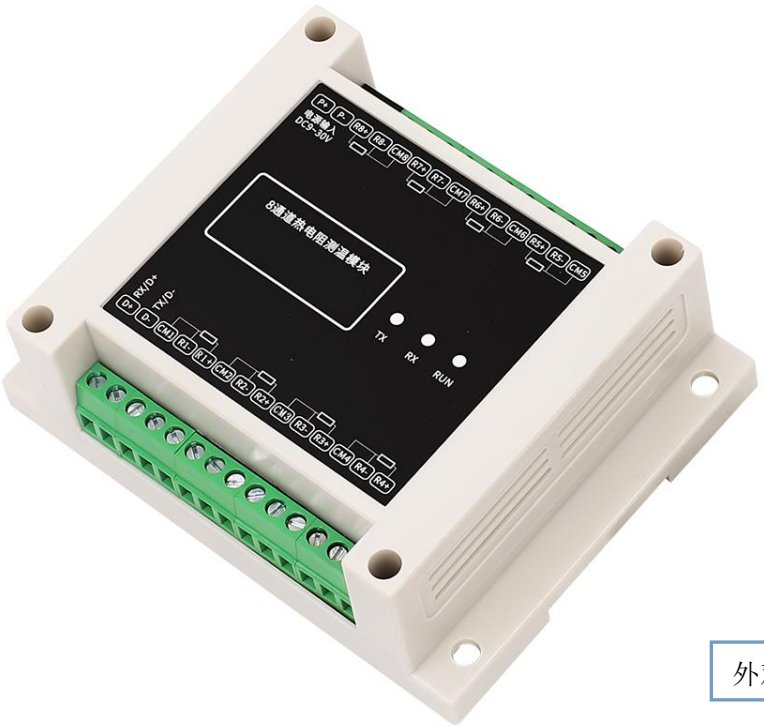
RJ45 网口出厂参数: TCP server 模式，IP:192.168.0.7,端口号:20108;默认 Modbus-TCP 协议
网页登录用户名:admin,登录密码:admin，可修改参数；
也可以用专用工具软件修改参数。

4、一键复位功能

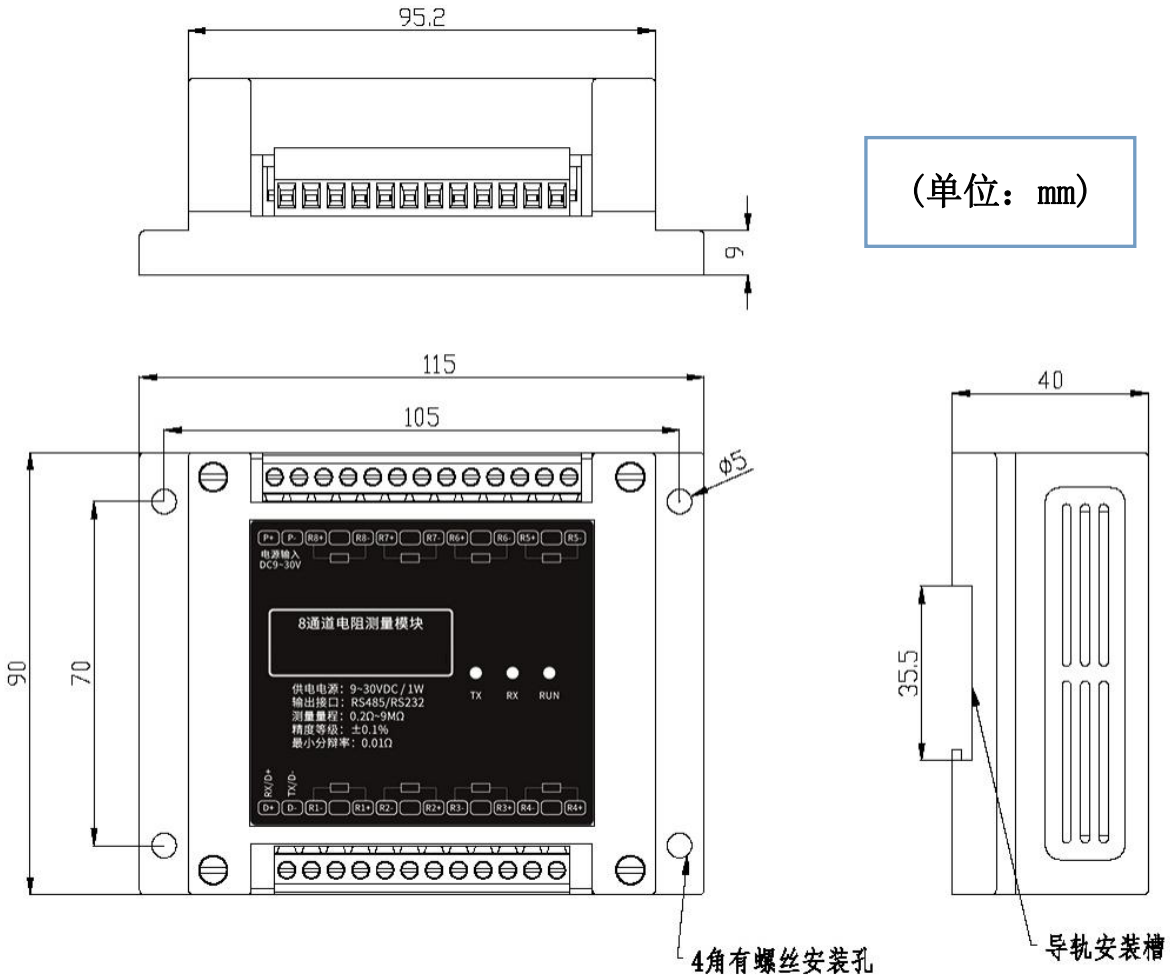
一键复位功能能在设置出错时，一键复位至出厂状态，步骤如下：

a. 打开外壳，按下 PCB 上的 SET 键不松开；
b. 重新上电或按一下复位键；
c. 此时保持 SET 键不松开，RUN 指示会先亮 1 秒，然后熄灭 2 秒；
d. 当出现 RUN 指示灯慢闪时，如果松开 SET 键，则复位通讯设置；
e. 如果想复位其它设置，则在出现 RUN 灯慢闪后，一直按住 SET 键不松开（约 30 秒），直到 RUN 熄灭，此时会复位所有设置，包括：通讯、采样速率、量程、电网设置等等，但不会复位校正参数。

5、外形



外观参考图



（单位：mm）

外观尺寸：115*90*40mm

6、端子接线定义图

6.1 二线制电阻接法：

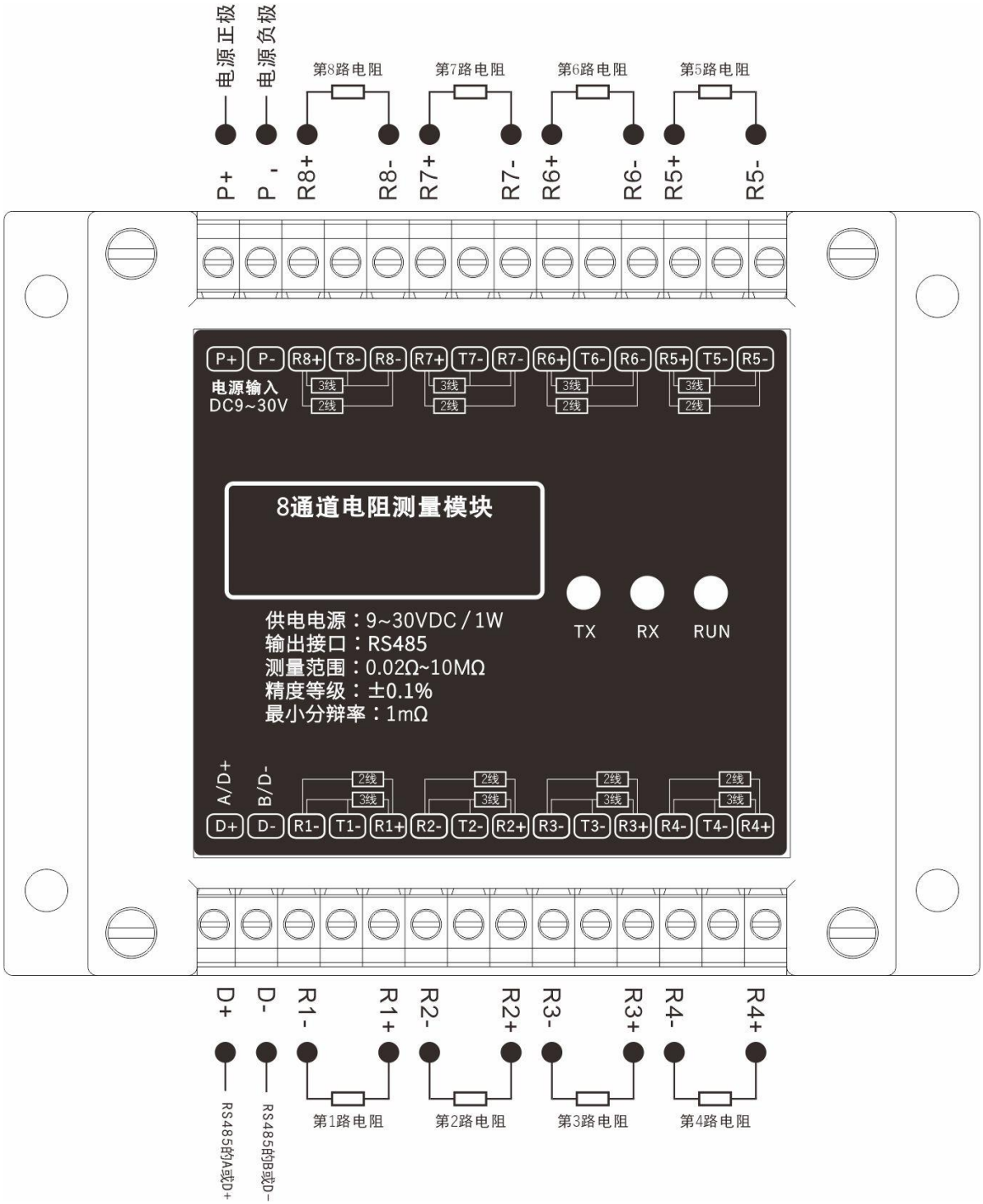


图 2：RS485 接口二线制版本接线图

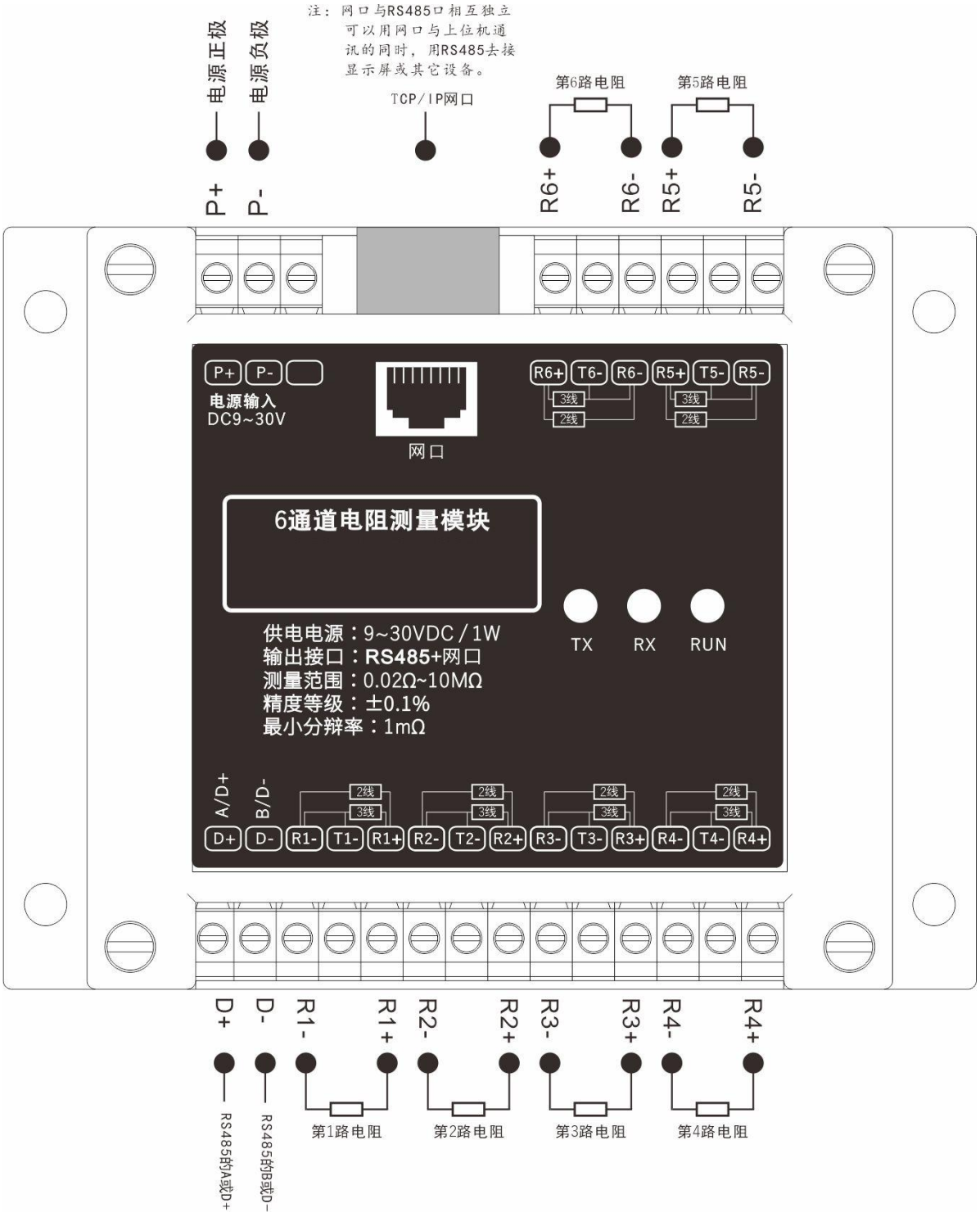


图 3：网口+RS485 接口二线制版本接线图

6.2 三线制电阻接法:

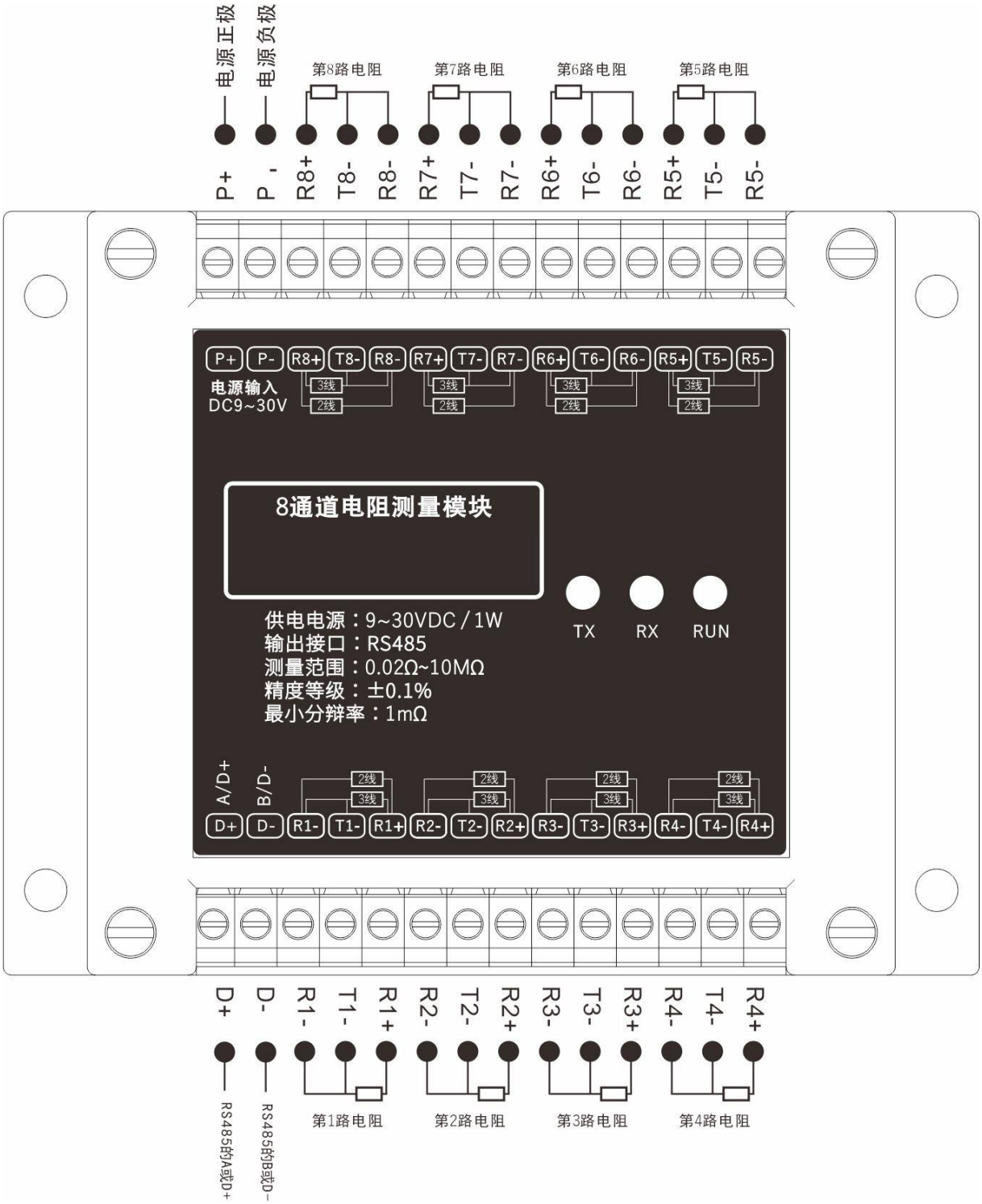


图 4：RS485 接口三线制版本接线图

6.3 端子定义说明表

表 5：端子定义说明表

端子名称	端子用途
P+	电源正极
P-	电源负极
D+	RS485 端口数据线 A
D-	RS485 端口数据线 B
R1+、R2+.....R8+	电阻正端输入
R1-、R2-.....R8-	电阻负端输入
T1-、T2-.....T8-	三线制电阻的第 3 条引线，二线制测电阻浮空不接
运行灯状态说明	RUN 灯：产品上电运行正常时会闪动；闪动频率与采样转换速率一样。 TX 灯：RS485 有数据发送时会闪烁 RX 灯：RS485 有数据接收时会闪烁

7、寄存器说明

7.1 电阻寄存器（支持用 03 和 04 功能码读，不能改写）

表 6：阻值寄存器表

寄存器内容		寄存器地址（十六进制）	对应 PLC 或组态软件配置地址（十进制）	寄存器数据说明
32 位电阻寄存器	第 1 通道	0x0000:0x0001	40001:40002	1. 此寄存器由两个 16 位寄存器组合成一个 32 位寄存器，高字节在前，低字节在后。 2. 电阻数据分辨率为 0.01Ω 3. 电阻值=寄存器的数据 ÷100，比如：寄存器数据为 0x00010100,则阻值为 657.92Ω。 4. 当阻值超过 10M 时，则数据为 0xFFFFFFFF。
	第 2 通道	0x0002:0x0003	40003:40004	
	第 3 通道	0x0004:0x0005	40005:40006	
	第 4 通道	0x0006:0x0007	40007:40008	
	第 5 通道	0x0008:0x0009	40009:40010	
	第 6 通道	0x000A:0x000B	40011:40012	
	第 7 通道	0x000C:0x000D	40013:40014	
	第 8 通道	0x000E:0x000F	40015:40016	
x0.001Ω 寄存器	第 1 通道	0x1000	44097	1. 此寄存器存 16 位 0.001Ω级电阻值； 2. 电阻数据分辨率为 0.001Ω 3. 电阻值=寄存器的数据 ÷ 1000，比如：寄存器数据为 0x0100,则阻值为 0.256Ω。
	第 2 通道	0x1001	44098	
	第 3 通道	0x1002	44099	
	第 4 通道	0x1003	44100	

	第 5 通道	0x1004	44101	4. 当数值为 0xFFFF 时(十进制 65535)，表示电阻超过了 65.534Ω。
	第 6 通道	0x1005	44102	
	第 7 通道	0x1006	44103	
	第 8 通道	0x1007	44104	
x0.01Ω 寄存器	第 1 通道	0x1040	44161	1. 此寄存器存 16 位 0.01Ω级电阻值； 2. 电阻数据分辨率为 0.01Ω 3. 电阻值=寄存器的数据 ÷ 100，比如：寄存器数据为 0x0100, 则阻值为 2. 56Ω。 4. 当数值为 0xFFFF 时(十进制 65535)，表示电阻超过了 655.34Ω。
	第 2 通道	0x1041	44162	
	第 3 通道	0x1042	44163	
	第 4 通道	0x1043	44164	
	第 5 通道	0x1044	44165	
	第 6 通道	0x1045	44166	
	第 7 通道	0x1046	44167	
	第 8 通道	0x1047	44168	
x1Ω 寄存器	第 1 通道	0x1080	44225	1. 此寄存器存 16 位 1Ω级电阻值； 2. 电阻数据分辨率为 1Ω 3. 电阻值=寄存器的数据 ，比如：寄存器数据为 0x0100, 则阻值为 256Ω。 4. 当数值为 0xFFFF 时(十进制 65535)，表示电阻超过了 65534Ω。
	第 2 通道	0x1081	44226	
	第 3 通道	0x1082	44227	
	第 4 通道	0x1083	44228	
	第 5 通道	0x1084	44229	
	第 6 通道	0x1085	44230	
	第 7 通道	0x1086	44231	
	第 8 通道	0x1087	44232	
x0.1kΩ 寄存器	第 1 通道	0x10C0	44289	1. 此寄存器存 16 位 0.1kΩ级电阻值； 2. 电阻数据分辨率为 0.1kΩ 3. 电阻值=寄存器的数据÷ 10 ，比如：寄存器数据为 0x0100, 则阻值为 25. 6kΩ。 4. 当数值为 0xFFFF 时(十进制 65535)，表示电阻超过了 6553.4kΩ。
	第 2 通道	0x10C1	44290	
	第 3 通道	0x10C2	44291	
	第 4 通道	0x10C3	44292	
	第 5 通道	0x10C4	44293	
	第 6 通道	0x10C5	44294	
	第 7 通道	0x10C6	44295	
	第 8 通道	0x10C7	44296	
x1kΩ 寄存器	第 1 通道	0x1100	44353	1. 此寄存器存 16 位 kΩ级电阻值； 2. 电阻数据分辨率为 1kΩ 3. 电阻值=寄存器的数据，比如：寄存器数据为 0x0100, 则阻值为 256kΩ。 4. 当数值为 0xFFFF 时(十进制 65535)，表示电阻超过了 65534kΩ。
	第 2 通道	0x1101	44354	
	第 3 通道	0x1102	44355	
	第 4 通道	0x1103	44356	
	第 5 通道	0x1104	44357	
	第 6 通道	0x1105	44358	
	第 7 通道	0x1106	44359	
	第 8 通道	0x1107	44360	

7.2 配置字寄存器

此类寄存器只能用 03 功能码读或 06 与 16 功能码写，见表如下：

表 7：设置寄存器表

寄存器地址(Hex)	保持寄存器内容	寄存器个数	寄存器状态	数据范围
0x0050	地址	1	读/写	地址(此值可填 1-253，254 与 255 为广播地址)(默认 01)
0x0051	波特率（RS485 口）	1	读/写	0000 设置波特率-115200bps 0001 设置波特率-9600bps(默认) 0002 设置波特率-19200bps 0003 设置波特率-38000bps 0004 设置波特率-2400bps 0005 设置波特率-4800bps 0006 设置波特率-9600bps 0007 设置波特率-19200bps 0008 设置波特率-38400bps 0009 设置波特率-57600bps 000A 设置波特率-115200bps
0x0052	寄偶校验（RS485 口)	1	读/写	0000 无校验，1 个停止位(默认) 0001 寄校验，1 个停止位 0002 偶校验，1 个停止位 0003 无校验，2 个停止位 0004 寄校验，2 个停止位 0005 偶校验，2 个停止位
0x0055	模块名称--高	1	读/写	默认:5A54H（ZT 的 ASCII 码）
0x0056	模块名称--中	1	读/写	默认:3038H（08 的 ASCII 码）
0x0057	模块名称--低	1	读/写	默认:5250H（R0 的 ASCII 码）
0x0058	软件版本	1	读	3033：03 的 ASCII 码
0x0059	软件子版本	1	读	3032：02 的 ASCII 码
0x005A	网口与 MCU 通讯速率	1	读/写	同 0x0051
0x005B	网口与 MCU 通讯寄偶校验	1	读/写	同 0x0052
0x0081	采样速率	1	读/写	0--第一档，最慢速率（出厂默认） 1--第二档 2--第三档 3--第四档 刷新时间越快，精度越低。
0x0082	电网配置	1	读/写	其值为： 50--适用频率为 50HZ 的电网（出厂默认） 60--适用频率为 60HZ 的电网
0x0083	校正标志	1	读/写	其值为 0xA5F0 时，表示出厂已校正
0x0084	电阻接线方式	1	读/写	2--二线制 （出厂默认） 3--三线制
0x0085	量程选择	1	读/写	量程选择：

				0-自动量程（出厂默认） 1--0.02Ω~25Ω 2--0.02Ω~1kΩ 3--0.02Ω~5kΩ 4--0.02Ω~20kΩ 5--0.02Ω~100kΩ 6-0.02Ω~1MΩ 7-0.02Ω~10MΩ
0x01FA	通讯协议定义	1	读/写	详见附件 《如何切换 Modbus-RTU 与 Modbus-TCP 协议》
0x02E0~ 0x02E7	线阻补偿	8	读/写	对应 1 至 8 通道电阻补偿，用户可填入正负数值修正电阻误差。此值单位为 0.001Ω.范围为 -32768~+32767（负数为补码），即可以修正最大误差为 -32.768Ω~+32.767Ω 实际电阻=测量电阻+此补偿值 如要改写此值，必须向 8000H 寄存器写 0x0A，改写完后，向 8000H 寄存器写 0x05 加锁保护。
0x7240	转换控制	1	读/写	向此寄存器写 0x5A，模块会停止转换电阻值，并把所有阻值寄存器复位； 向此寄存器写除 0x5A 以外的任何值，模块重新转换。
0x8000	线阻补偿与报警值写保护	1	写	向此寄存器写 0x0A 才能改写误差补偿寄存器；改写完后，写 0x05 锁存，并生效误差补偿

8、线阻补偿

当用户测量引线过长时，会引起测试误差，这时候用户可以通过 0x02E0~0x02E7 寄存器来修正此误差。0x02E0~0x02E7 寄存器的值为一个 16 位的有符号数，单位为 mΩ，最大可修正范围为 -32768mΩ~+32767mΩ。计算方法为：实际电阻=测量电阻+此寄存器补偿值。

用户可以先把测量的引线负载端短路，用模块测到电阻值，再把引电阻值改成负数填入寄存器。如：当第 1 通道的引线短路后测出的阻值为 18mΩ，则在 0x02E0 寄存器填入 0xFFEE，即-18。这样读电阻值寄存器时，就会自动减去这个误差了。

版本： **V2.0 2022.05.17 更新**
V2.1 2022.05.23 更新
V3.0 2022.07.27 更新
V3.1 2022.08.25 更新
V3.2 2022.8.29 更新的误差补偿

附件 1

Modbus-RTU 通讯协议

Modbus-RTU 通讯协议举例

如下所有命令都是以硬件地址为 01 来举例说明；

1. 读模块配置字寄存器命令（03 功能码）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	03	功能码	1
3	00 55	数据起始寄存器地址，高 8 位在前，低 8 位在后；参照“配置字寄存器表”	2
4	00 02	读取寄存器个数，高 8 位在前，低 8 位在后 （此列读取 2 个寄存器数据）	2
5	D4 1B	CRC 校验码（低位在前，高位在后）	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	03	功能码	1
3	04	返回的数据字节个数，2 个寄存器*2	1
4	35 39 30 39	读取的寄存器数据，每 2 个字节表示一个寄存器数据，高位在前，低位在后；第 1 个寄存器数据在前	可变
5	F1 E0	CRC 校验码（低位在前，高位在后）	2

2. 读 8 路电阻值(以 1Ω分辨率寄存器为列）命令（支持 04 与 03 功能码,字节读）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1

2	03	功能码	1
3	10 80	起始通道序号，高 8 位在前，低 8 位在后；参照”阻值寄存器表”	2
4	00 08	读取 8 个通道电阻值，高 8 位在前，低 8 位在后	2
5	41 24	CRC 校验码（低位在前，高位在后）	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	03	功能码	1
3	10	返回的数据字节个数，8 个寄存器*2	1
4	01 37 03 05 00 00 ...	读取的寄存器数据，每 2 个字节表示一个寄存器数据，高位在前，低位在后；第 1 个寄存器数据在前,数据还原参照 6.1	可变
5	CRC	CRC 校验码（低位在前，高位在后）	2

3. 配置寄存器修改命令：

3.1 单个寄存器修改命令（06 功能码，每次只能修改一个寄存器，举例修改通讯地址）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	06	功能码	1
3	00 50	寄存器地址，高 8 位在前，低 8 位在后，参照“配置字寄存器表”	2
4	00 02	寄存器数据，参照“配置字寄存器表”	2
5	08 1A	CRC 校验码（低位在前，高位在后）	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	06	功能码	1
3	00 50	寄存器地址，返回相同	2
4	00 02	寄存器数据，返回相同	2
5	08 1A	CRC 校验码，返回相同	2

3.2 连续修改多个寄存器命令（16 功能码，举例修改各通道补偿值）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
2	10	功能码	1
3	04 40	起始寄存器，高 8 位在前，低 8 位在后	2

		参照“配置字寄存器表”	
4	00 04	写入寄存器长度，高 8 位在前，低 8 位在后 （此列写入 4 个寄存器）	2
5	08	写入字节长度（写入寄存器长度 x2）	1
6	00 00 00 01 00 03 00 06	写入的数据,每 2 个字节表示一个寄存器数据，高位在前，低位在后；此列表示把 0440H 寄存器写入数据 0000H,0441H 寄存器写入数据 0001H，0442H 寄存器写入数据 0003H，0443H 寄存器写入数据 0006H	按序列 8 表示的字节数
7	F4 03	CRC 校验码（低位在前，高位在后）	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	01	从设备地址，与主设备发送报文保持一致	1
2	10	功能码	1
3	04 40	起始寄存器，高 8 位在前，低 8 位在后 与主设备发送的报文相同	2
4	00 04	写入寄存器长度，高 8 位在前，低 8 位在后 与主设备发送的报文相同	2
5	C1 2E	CRC 校验码（低位在前，高位在后）	2

附件 2：

Modbus-TCP 通讯协议

如下所有命令都是以硬件地址为 01 来举例说明；

1 读模块配置字寄存器命令（03 功能码）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，一般每次通信之后将被要求加 1 以区别不同的通信数据报文（此处以 3D 46 为列）	2
2	00 01	表示协议标识符（此处以 00 01 为列）	2
3	00 06	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	03	功能码	1
6	00 55	数据起始寄存器地址，高 8 位在前，低 8 位在后； 参照“配置字寄存器表”	2
7	00 02	读取寄存器个数，高 8 位在前，低 8 位在后 （此列读取 2 个寄存器数据）	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，应答报文要求与先前对应的主设备发送报文保持一致	2
2	00 01	表示协议标识符，与主设备发送报文保持一致	2
3	00 07	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 7 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	03	功能码	1
6	04	返回的数据字节个数，2 个寄存器*2	1
7	35 39 30 39	读取的寄存器数据，每 2 个字节表示一个寄存器数据，高位在前，低位在后；第 1 个寄存器数据在前	可变

2　读 8 路电阻值(以 1Ω分辨率寄存器为列）命令（支持 04 与 03 功能码,字节读）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，一般每次通信之后将被要求加 1 以区别不同的通信数据报文（此处以 3D 46 为列）	2
2	00 01	表示协议标识符（此处以 00　01 为列）	2
3	00 06	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	03	功能码	1
6	10 80	起始通道序号，高 8 位在前，低 8 位在后；参照”阻值寄存器表”	2
7	00 08	读取 8 个通道电阻值，高 8 位在前，低 8 位在后	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，应答报文要求与先前对应的主设备发送报文保持一致	2
2	00 01	表示协议标识符，与主设备发送报文保持一致	2
3	00 13	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 19 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	03	功能码 03 或 04	1
6	10	返回的数据字节个数，8 个寄存器*2	1
7	01 37 03 05 00 00 ...	读取的寄存器数据，每 2 个字节表示一个寄存器数据，高位在前，低位在后；第 1 个寄存器数据在前,数据还原参照”阻值寄存器表”	可变

3 配置寄存器修改命令：

3.1 单个寄存器修改命令（06 功能码，每次只能修改一个寄存器，举例修改通讯地址）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，一般每次通信之后将被要求加 1 以区别不同的通信数据报文（此处以 3D 46 为列）	2
2	00 01	表示协议标识符（此处以 00 01 为列）	2
3	00 06	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	06	功能码	1
6	00 50	寄存器地址，高 8 位在前，低 8 位在后，参照“配置字寄存器表”	2
7	00 02	寄存器数据，参照“配置字寄存器表”	2

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，应答报文要求与先前对应的主设备发送报文保持一致	2
2	00 01	表示协议标识符，与主设备发送报文保持一致	2
3	00 06	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	06	功能码	1
6	00 50	寄存器地址，返回相同	2
7	00 02	寄存器数据，返回相同	2

3.2 连续修改多个寄存器命令（16 功能码，举例修改各通道补偿值）

主设备发送报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，一般每次通信之后将被要求加 1 以区别不同的通信数据报文（此处以 3D 46 为列）	2
2	00 01	表示协议标识符（此处以 00 01 为列）	2
3	00 0F	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，可变（1-253,254 与 255 为广播地址）（此列为 01 设备地址）	1
5	10	功能码	1
6	04 40	起始寄存器，高 8 位在前，低 8 位在后 参照“配置字寄存器表”	2
7	00 04	写入寄存器长度，高 8 位在前，低 8 位在后	2

		（此列写入 4 个寄存器）	
8	08	写入字节长度（写入寄存器长度 x2）	1
9	00 00 00 01 00 03 00 06	写入的数据,每 2 个字节表示一个寄存器数据，高位在前，低位在后；此列表示把 0440H 寄存器写入数据 0000H,0441H 寄存器写入数据 0001H，0442H 寄存器写入数据 0003H，0443H 寄存器写入数据 0006H	按序列 8 表示的字节数

从设备返回正确报文

序列	数据举列 (16 进制)	数据说明	字节数
1	3D 46	为此次通信事务处理标识符，应答报文要求与先前对应的主设备发送报文保持一致	2
2	00 01	表示协议标识符，与主设备发送报文保持一致	2
3	00 06	为数据长度，用来指示接下来数据的长度，高 8 位在前，低 8 位在后（此列表示后面跟随有 6 个字节的数据）	2
4	01	从设备地址，与主设备发送报文保持一致	1
5	10	功能码	1
6	04 40	起始寄存器，高 8 位在前，低 8 位在后 与主设备发送的报文相同	2
7	00 04	写入寄存器长度，高 8 位在前，低 8 位在后 与主设备发送的报文相同	2

附件 3：

如何切换 Modbus-RTU 与 Modbus-TCP 协议

（本说明适用 ZH-T16xx 与 ZH-T08xx 全系列产品）

如何在产品中切换 Modbus-TCP、Modbus-RTU、自定义协议以及用 Modbus-RTU 扩展下联模块？

A. 只需要用 06 功能码修改 0x1FA 寄存器就可改变串口的通信协议和工作方式。

B. 0x1FA 寄存器为 16 位寄存器，每 4 位对应一个通讯口设置，列表如下:

表（1）

0x1FA 寄存器位	**对应产品通讯接口序号**	**对应产品通信接口**	**数据含义代码（16 进制）**

Bit3:Bit0	第一通讯口	RS485 口	0x0--从机 Modbus-RTU 协议（RS485 出厂默认协议） 0x1--从机 Modbus-TCP 协议(以太网口出厂默认协议） 0x4--从机自定义协议 1 0x6--从机自定义协议 2
Bit7:Bit4	第二通讯口	以太网口	

C. 注意：因为所有通讯口的协议格式存储在同一寄存器（0x1FA)的不同位上(16 位 2 个字节)，而我们用 06 或 16 功能码修改时，是按字节修改的，所以在修改一个通讯口的协议时，要把其它通讯口的原协议代码保留填入，否则会同步修改。

D. 举例，当 RS485 口为 Modbus-RTU 协议时，通过 RS485 口更改通讯协议：

➢ RS485 口保持 Modbus-RTU 协议不变，以太网口协议修改为 Modbus-TCP，则需发送命令如下：

命令：01 06 01 FA 00 10 A9 CB(返回相同代码即修改成功)，解析如下表：

设备地址	功能码	改写的寄存器		改写的数据		CRC校验码	
		高8位	低8位	高8位(Bit15:Bit8)	低8位 (Bit7:Bit0)	高8位	低8位
01	06	01	FA	00 第4通讯口格式 第3通讯口格式	10 第2通讯口格式 第1通讯口格式	A9	CB

注：表中第 4 通讯口与第 3 通讯口未用到，填 0 就可以了。

➢ 当需要把 RS485 由当前通讯协议 Modbus-RTU 更改为 Modbus-TCP 协议，以太网口通讯协议改为 Modbus-RTU 时，，则需发送命令如下：

命令：01 06 01 FA 00 01 69 C7(返回相同指令即修改成功)；解析如下表：

设备地址	功能码	改写的寄存器		改写的数据		CRC校验码	
		高8位	低8位	高8位	低8位	高8位	低8位
01	06	01	FA	00 第4通讯口格式 第3通讯口格式	01 第2通讯口格式 第1通讯口格式	69	C7

E. 举例，由 Modbus-TCP 协议更改为 Modbus-RTU：

- RS485 口与以太网口当前通讯协议为 Modbus-TCP，如要全改成 Modbus-RTU 协议，则需要发命令：

命令：00 00 00 00 00 06 01 06 01 FA 00 00(返回相同代码即修改成功)；解析如下表：

事务标示符		协议标示符		数据长度		设备地址	功能码	改写的寄存器		改写的数据	
高8位	低8位	高8位	低8位	高8位	低8位			高8位	低8位	高8位	低8位
00	00	00	00	00	06	01	06	01	FA	00 第4通讯口格式 第3通讯口格式	00 第2通讯口格式 第1通讯口格式

附 4：

网络接口模块测试与设置方法

1、**网口功能特点**:

- 10/100Mbps 自适应以太网接口，支持 AUTO-MDIX 网线交叉直连自动切换;
- 工作模式可选择 TCP Serve、TCP Client、UDP Client、UDP Server、Httpd Client;
- 自定义心跳包机制，保证连接真实可靠，可用来检测死连接;
- 自定义注册包机制，可检测连接状态，识别模块，也可做自定义包头;
- TCP Server 模式下，连接 Client 的数量可在 1 到 16 个之间任意设置，默认 4 个，已连接 Client 的 IP 可在内置网页状态界面显示，按连接计算发送/接收数据;
- TCP Server 模式下，当连接数量达到最大值时，新连接是否踢掉旧连接可设置;
- 支持 TCP Client 短连接功能，短连接断开时间自定义;
- 支持超时重启（无数据重启）功能，重启时间自定义;
- TCP 连接建立前，数据缓存是否清理可设置;
- DHCP 功能，能够自动获取 IP;
- MAC 地址可修改，出厂烧写全球唯一 MAC，支持自定义 MAC 功能;
- DNS 功能，域名解析; DNS 服务器地址可自定义;
- 支持虚拟串口，可提供配套的虚拟串口软件;
- 可以跨越网关，交换机，路由器运行; 可以工作在局域网，也可访问外网;

网口出厂默认参数：工作模式：TCP Serve；IP：192.168.0.7；端口号：20108；用户名：admin；密码:admin
与主芯片通信波特率 115200pbs,数据位 8 位，1 位停止位，无奇偶校验。

2、模块工作方式设置（可网页登录设置或用专用的设置软件方式）:

2.1 自带内置的网页服务器，与常规的网页服务器相同，用户可以通过网页登录设置参数也可以通过网页查看模块的相关状态。网页服务器的端口号可设置，默认为 80。

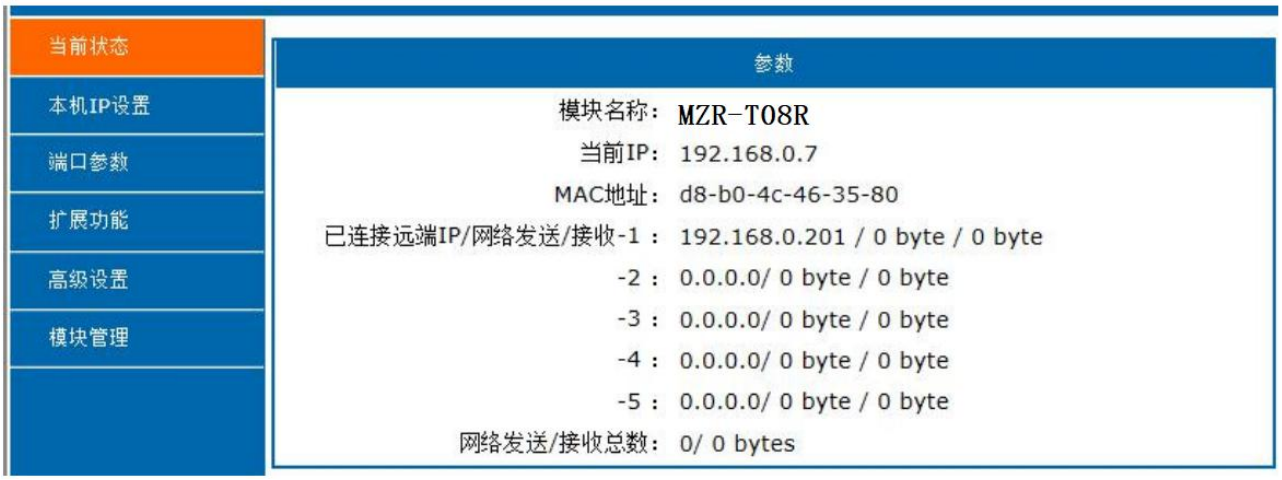
默认首页为当前状态界面，每隔 10s 刷新一次，显示模块工作状态：

网络发送总数：通过网络发送数据可以判断 模块发送多少数据到外网;

网络接收总数：通过接收计数可以判断有多少数据从网络发向模块;

已连接远端 IP/ 网络发送/ 接收：通过此项，可以看到 模块 与哪一个设备进行连接，该连接发送和接收的数据量有多少，目前只支持 5 个连接状态显示。

UDP Server 模式下，只显示发送/接收数据，不显示连接 IP。



图一、网页工作状态显示页面



图 2、模块参数网页设置页面

2.2 可至我司网站下载专用的设置工具软件，设置更直观快捷。

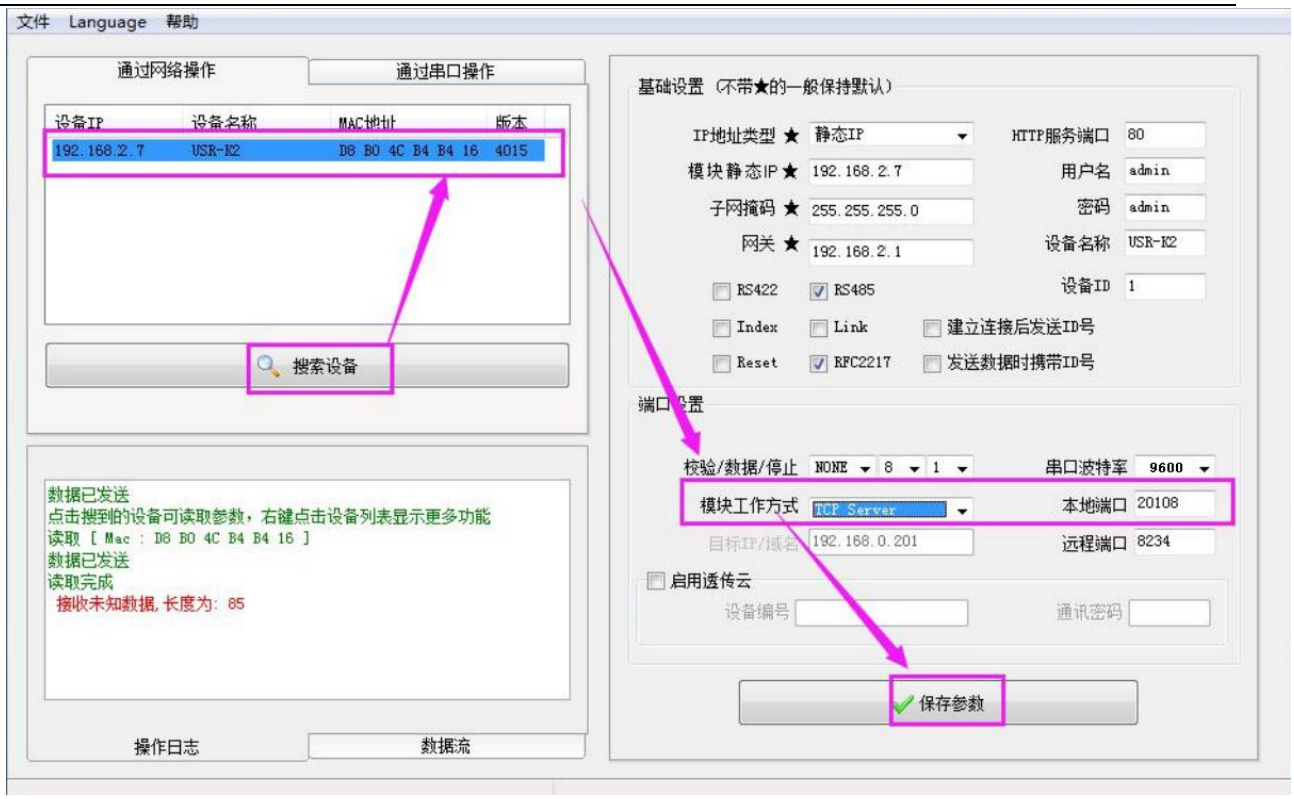


图 3、模块参数软件设置页面（可到本公司官网下载“网络设置软件”）

3、TCP Serve 模式通讯实例

模块设置为 TCP Serve 模式，IP 为 192.168.2.7，端口为 20108 的情况下，打开调试助手软件（本软件可以在本公司网站下载“串口调试助手”）按以下页面设置,本地 IP 需选择正错的本机电脑 IP;

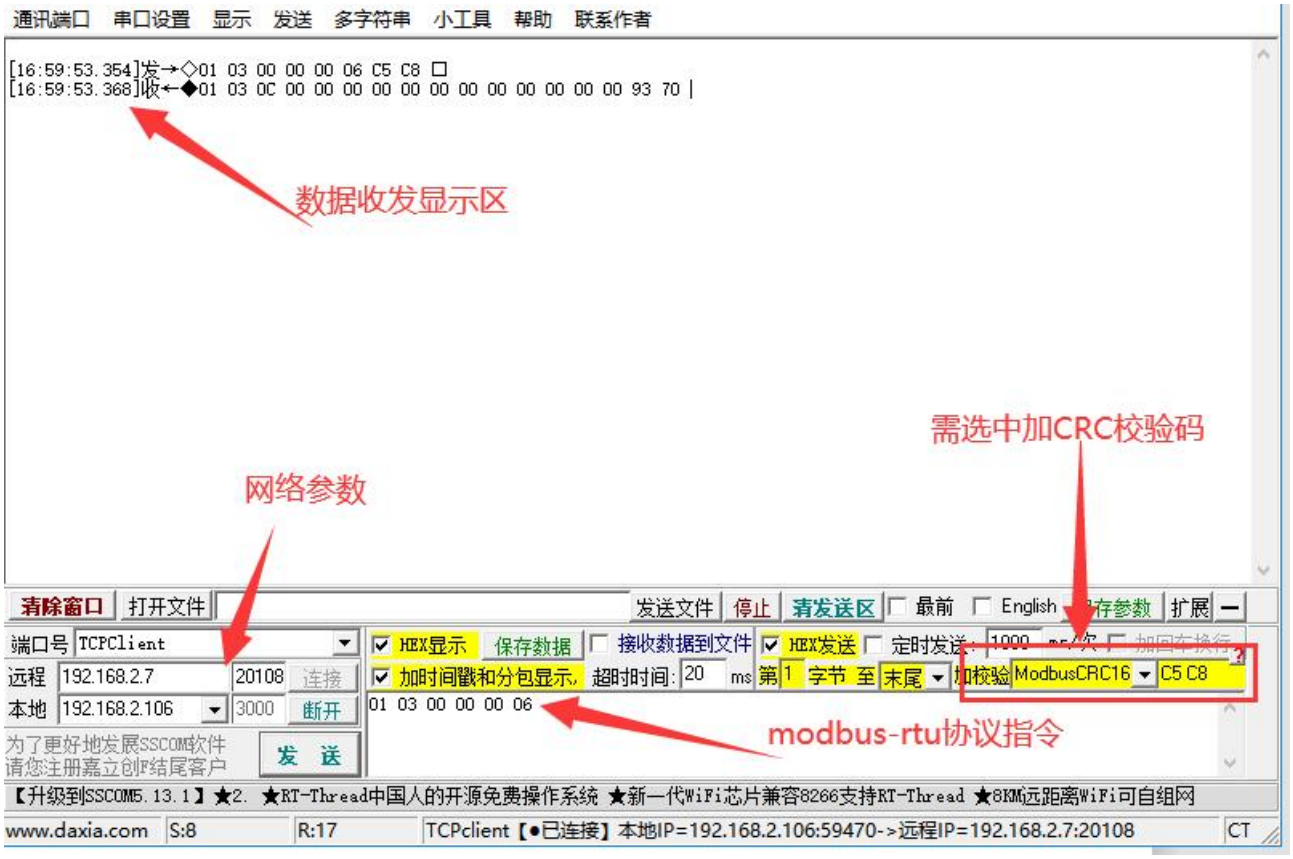


图 4、modbus-rtu 协议指令测试页面

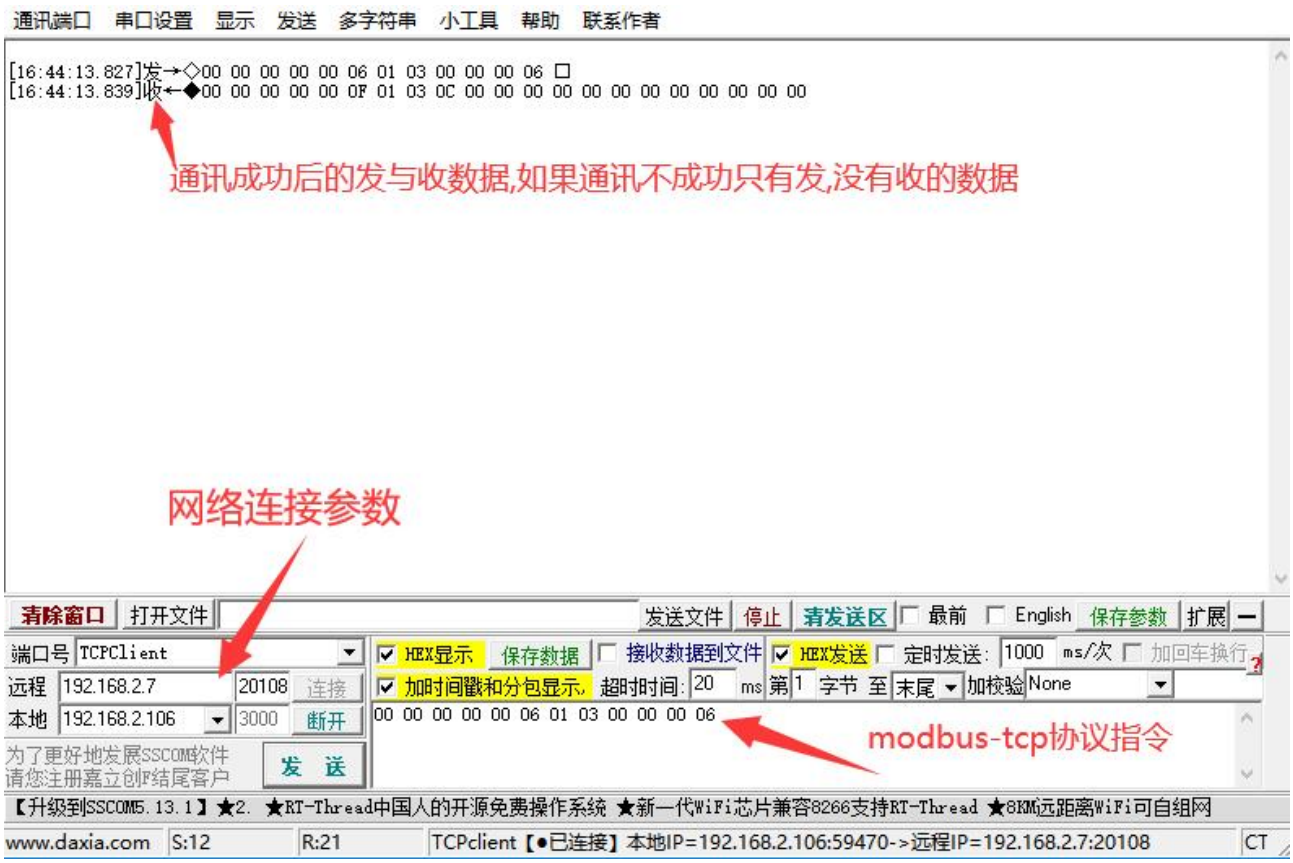


图 5、modbus-tcp 协议指令测试页面