1 通讯

1.1 简介

mPLC2 系列小型 PLC 串口通讯可以选用 Modbus 通讯协议，支持 ASCII 和 RTU 两种通讯模式，可以设成主站或者从站。

1.2 链路特性

1. 物理层：RS232、RS485、以太网（RJ45）

2. 链路层：异步传输

(1)数据位：7 位（ASCII）或者 8 位（RTU）

(2)传输速率：300，600，1200，2400，4800，9600，19200，38400，57600，115200

(3)校验方式：偶校验、奇校验或无校验

(4)停止位： 1 位或者 2 位

3. 组网配置：最多 31 个设备，地址范围 1～247。支持广播。

1.3 RTU 传输模式

1. 十六进制数据。

2. 字符间间隔应该少于 1.5 个字符时间。

3. 没有帧头和帧尾，帧间间隔至少为 3.5 个字符时间。

4. 使用 CRC16 校验。

5. RTU 帧的最大帧长度是 256 个字节，帧结构如下表：

帧构成	地址	功能码	数据	CRC
字节数	1	1	0～252	2

6. 字符间隔时间计算：

通讯波特率为 19200，那么 1.5 个字符时间=1/19200×11×1.5×1000=0.86ms

3.5 个字符间隔=1/19200×11×3.5×1000=2ms。

1.4 ASCII 传输模式

1. 使用 ASCII 数据通讯。

2. 帧使用“：（3A）”作为头，CRLF（0D 0A）两个字符作为尾

3．允许的字符间的间隔时间是 1s。

4．使用 LRC 校验。

5．ASCII 的帧结构比 RTU 的帧要长，ASCII 模式传送一个字节（HEX）需要两个字符编码，ASCII 的数据域（2×252）的最大长度是 RTU 数据域（252）的两倍，ASCII 的最大帧长为 513 个字符，帧构成如下表：

帧构成	头	地址	功能码	数据	LRC	尾
字节数	1	2	2	0～2*252	2	2

1.5 支持的 Modbus 功能码

支持 Modbus 通讯协议中的功能码 01，02，03，04，05，06，15，16。

1.6 PLC 元件与 Modbus 通讯协议地址的对应关系

类型	元件符号	元件编号	功能码	协议地址（16 进制）	协议地址（10 进制）	备注
线圈／位元件	M 辅助继电器	M0～M7679	0x01 0x05 0x0F	0～1DFF	0～7679	
		M8000～8511	0x01 0x05 0x0F	1F40～213F	8000～8511	
	X 输入继电器	X0～X1777	0x01 0x02 0x05 0x0F	2710～2B0F	10000～11023	X 的编码为 8 进制
	Y 输出继电器	Y0～Y1777	0x01 0x05 0x0F	2EE0～32DF	12000～13023	Y 的编码为 8 进制
	S 状态继电器	S0～S4095	0x01 0x05 0x0F	36B0～46AF	14000～18095	
	T 定时器触点	T0～T511	0x01 0x05 0x0F	4A38～4C37	19000～19511	
	C 计数器触点	C0～C255	0x01 0x05 0x0F	4E20～4F1F	20000～20255	
	F 继电器	F0～F7999	0x01 0x05 0x0F	5208～7147	21000～28999	

	B 继电器	B0～B32767	0x01 0x05 0x0F	8000～BFFF	32768～49151	B0～B16383 映射到 Modbus 地址，B16384～B32767 为PLC 内部使用。
	L 继电器	L0～L32767	0x01 0x05 0x0F	C000～FFFF	49152～65535	L0～L16383 映射到 Modbus 地址，L16384～L32768 为PLC 内部使用。
寄存器/字元件	D 寄存器	D0～D7999	0x03 0x06 0x10	0～1F3F	0～7999	
		D8000～D8511	0x03 0x06 0x10	1F40～213F	8000～8511	
	R 寄存器	R0～R16383	0x03 0x06 0x10	2710～670F	10000～26383	
	VD 寄存器	VD0～VD16383	0x03 0x06 0x10	6978～A977	27000～43383	
	RD 寄存器	RD0～RD16383	0x03 0x06 0x10	ABE0～EBDF	44000～60383	
	T 定时器	T0～T511	0x03 0x06 0x10	EE48～F047	61000～61511	
	C(16 位)计数器	C0～C199	0x03 0x06 0x10	F230～F2F7	62000～62199	
	C(32 位)计数器	C200～C255	0x03 0x10	F2F8～F367	62200～62311	

1.7 Modbus 从站

Modbus 从站不主动发送任何报文，只有接收到对本地寻址的报文后才根据具体情况看是否响应主站。从站仅支持 Modbus 功能码 01，02，03，05，06，08，15，16，其余的响应均为“非法功能码”（除广播帧）。

1.8 元件的读写

除了功能码 08，mPLC2 支持的其他功能码都是对元件读写操作的，原则上最多允许一帧读 2000 个位元件，写 1968 个位元件，读取 125 个字元件，写 120 个字元件。但由于实际的协议地址对不同类型元件是分开的，不连续，因此在对元件的读写操作时，一次读取的元件只能是一种类型的元件，而读取元件的最多数目也与实际定义的该类型元件个数有关系。

1.9 对双字元件的处理

C 元件的当前计数值为字元件或双字元件，C200～C255 为双字元件。对 C200～C255 的读写也是通过读写寄存器的功能码（0x03、0x10）来完成。每两个寄存器的地址对应一个 C 双字元件，读写时只能成对的读写寄存器。在读取双字元件时如果读取的开始地址不是偶数，会返回异常码非法地址，如果读取的寄存器个数不是偶数，会返回异常码非法数据。

1.10 对 LONG INT 的处理

对于一个 LONG INT 类型数据的存储，可能存在两个 D 元件内，例如：D3，D4 存一个 LONG INT 型的数， PLC 认为 D3 存储的是高 16 位，D4 存储的是低 16 位，当主站通过 Modbus 读取 LONG INT 数据时，读回数据后也应该按照 PLC 对 LONG INT 的存储原则重组 32 位的数据。FLOAT 的存储原则等同于 LONG INT 的存储原则。

1.11 诊断功能码

诊断功能码用来提供测试主站和从站之间的通讯，或者从站的各种内部差错状态。支持的诊断子功能码如下表所示：

功能码	子功能码	子功能码名称		功能码	子功能码	子功能码名称
08	00	返回询问数据		08	12	返回总线通讯错误计数
08	01	重新启动通讯选项		08	13	返回总线异常错误计数
08	04	强制只听模式		08	14	返回从站报文计数
08	10	清除计数器		08	15	返回从站无响应计数
08	11	返回总线报文计数		08	18	返回总线字符超限计数

1.12 异常码

当主站发送的命令，在正常的响应中，从站在数据域中返回数据或统计值。在异常的响应中，服务器在数据域中返回异常码，异常码如下表：

异常代码	异常代码意义
0x01	非法功能码
0x02	非法寄存器地址
0x03	非法的数据

另外，从站在以下几种情况接收到数据会不返回响应报文：

广播帧中有错误，例如数据错误，地址错误等。

字符超限不返回，例如 RTU 帧大于 256 个字节。

当 RTU 传输模式，字符间的间隔时间超时，相当于收到错误帧，不返回。

从站在只听模式不返回。

从站接收到错误的 ASCII 错误帧，包括帧尾错误，帧中的字符范围错误。

1.13 MODBUS 主站

1、指令格式

16bit 6 步		32bit 6 步		指令格式
MODBUS	\	\	\	MODBUS S D

操作数的数据类型如下表

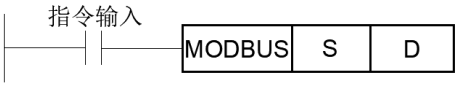
操作数	内容	类型
S	端口编号，K0-K15 表示端口编号，K300 表示 TCP	实数(2 进制)
D	行号，通信数据都以表格方式在主站设定界面设定	实数(2 进制)

操作数的对象软元件如下表

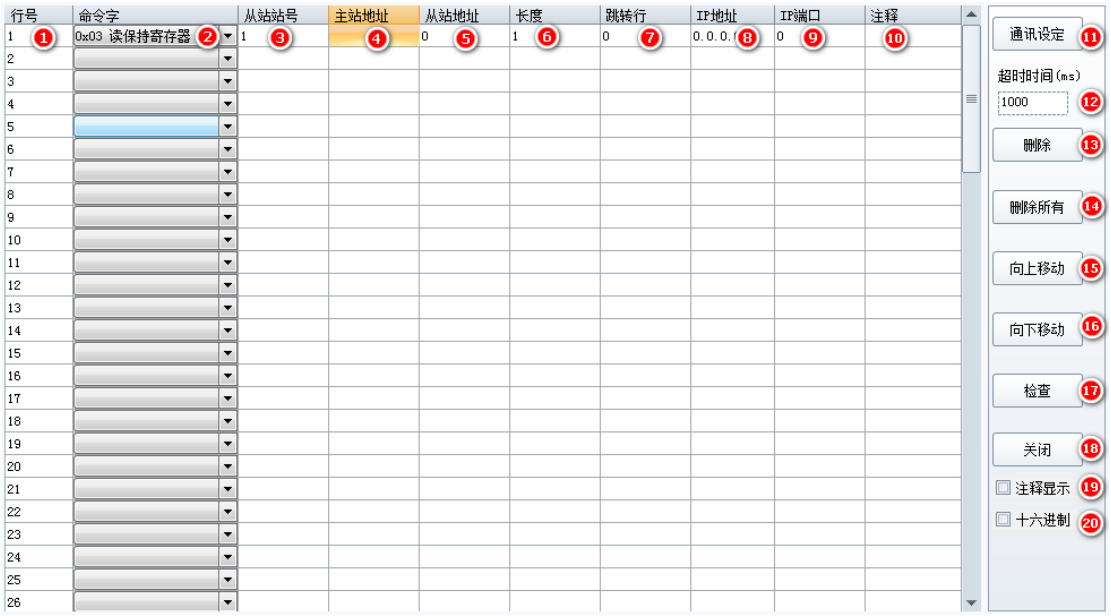
操作数种类	位软元件										常数		实数	字符串
	X	Y	M	T	C	S	B	L	F	Dx.y	K	H	E	“”
S											●	●		
D											●	●		
操作数种类	字软元件										变址			指针
	KnX	KnY	KnM	KnS	T	C	D	R	VD	RD	V	Z	修饰	P
S							●	●					●	
D							●	●					●	

2、功能和动作说明

MODBUS 主站指令



主站表格参数设置



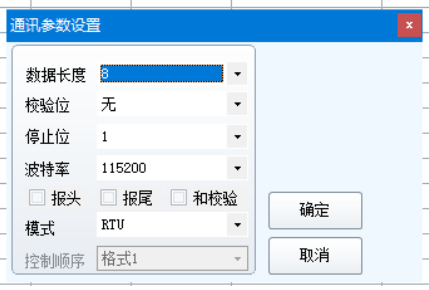
1	行号（1~99），作为 MODBUS 指令的参数索引
2	命令字 0x01 读线圈 0x02 读离散量输入 0x03 读保持寄存器 0x04 读输入寄存器 0x05 写单个线圈 0x06 写单个寄存器 0x0F 写多个线圈 0x10 写多个寄存器
3	从站的站号（从 1 开始）
4	用于存放从从站接收的数据，或者存放要发送给从站的数据。 对于字命令可使用 D 或 R；对于位命令，可使用 M 位元件。
5	从站地址，可参照从站的寄存器定义设置
6	读写寄存器的长度
7	设定范围为 0~100，0 表示不跳转；如果设定为 0 以外的值，例如设成 3，表示执行 MODBUS 指令指定的行号后，自动跳转到执行行号为 3 的命令，执行当前行后完成位 M8260 置位。
8	服务器 IP 地址，Modbus/TCP 客户端用
9	服务器 IP 端口，Modbus/TCP 客户端用
10	注释说明
11	设置串口波特率校验位等参数
12	从站回复超时时间
13	删除当前行配置

14	删除所有配置
15	当前行配置向上移动
16	当前行配置向下移动
17	检查所有配置
18	关闭当前设置页面
19	勾选后，显示[4]主站地址注释
20	勾选后，[5]从站地址用 16 进制显示

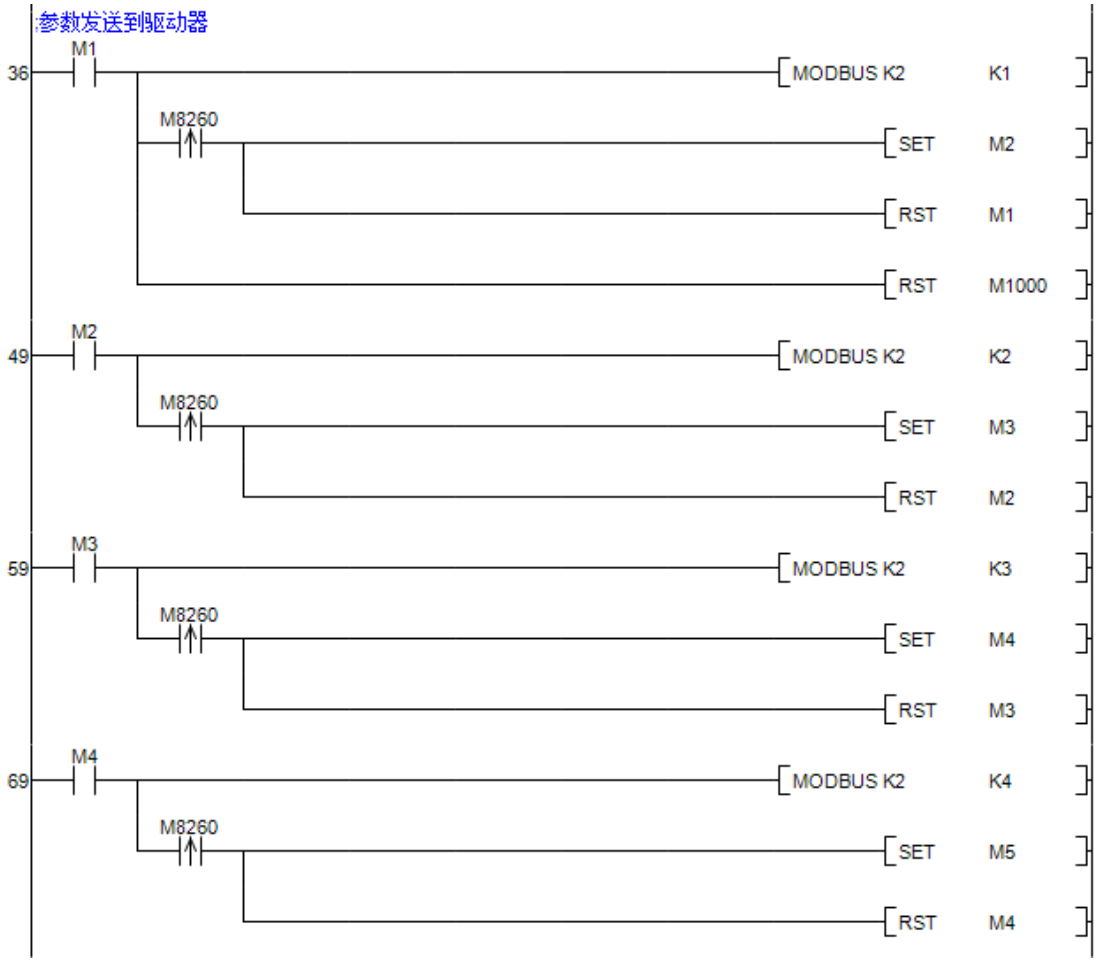
3、程序举例

MODBUS 主站表格配置

行号	命令字	从站站号	主站地址	从站地址	长度	跳转行	IP地址	IP端口	注释
1	0x06 写单个寄存器	2	D0	25088	1	0	0.0.0.0	0	运行模式
2	0x06 写单个寄存器	2	D1	25090	1	0	0.0.0.0	0	位置低16位
3	0x06 写单个寄存器	2	D2	25089	1	0	0.0.0.0	0	位置高16位
4	0x06 写单个寄存器	2	D3	25091	1	0	0.0.0.0	0	运行速度
5	0x06 写单个寄存器	2	D4	25092	1	0	0.0.0.0	0	加速时间
6	0x06 写单个寄存器	2	D5	25093	1	0	0.0.0.0	0	减速时间
7	0x06 写单个寄存器	2	D6	25094	1	0	0.0.0.0	0	停顿时间
8	0x06 写单个寄存器	2	D10	24578	1	0	0.0.0.0	0	控制字16
9	0x06 写单个寄存器	3	D0	25096					
10	0x06 写单个寄存器	3	D1	25098					
11	0x06 写单个寄存器	3	D2	25097					
12	0x06 写单个寄存器	3	D3	25099					
13	0x06 写单个寄存器	3	D4	25100					
14	0x06 写单个寄存器	3	D5	25101					
15	0x06 写单个寄存器	3	D6	25102					
16	0x06 写单个寄存器	3	D11	24578					17
17	0x06 写单个寄存器	4	D0	25104					
18	0x06 写单个寄存器	4	D1	25106					
19	0x06 写单个寄存器	4	D2	25105					



程序



4、注意事项

无

1.14 字符集

1.14.1 OPENS 指令(打开端口)

1、指令格式

16bit 6 步		32bit 6 步		指令格式
OPENS	\	\	\	OPENS “S”

操作数的数据类型如下表

操作数	内容	类型
S	字符集表格内的“配置名称”	

1.14.2 SENDS 指令(发送)

1、指令格式

16bit 6 步	32bit 6 步	指令格式

SENDS	\	\	\	SENDS “S”

操作数的数据类型如下表

操作数	内容	类型
S	字符集表格内的“配置名称”	

1.14.3 REVS 指令(接收)

1、指令格式

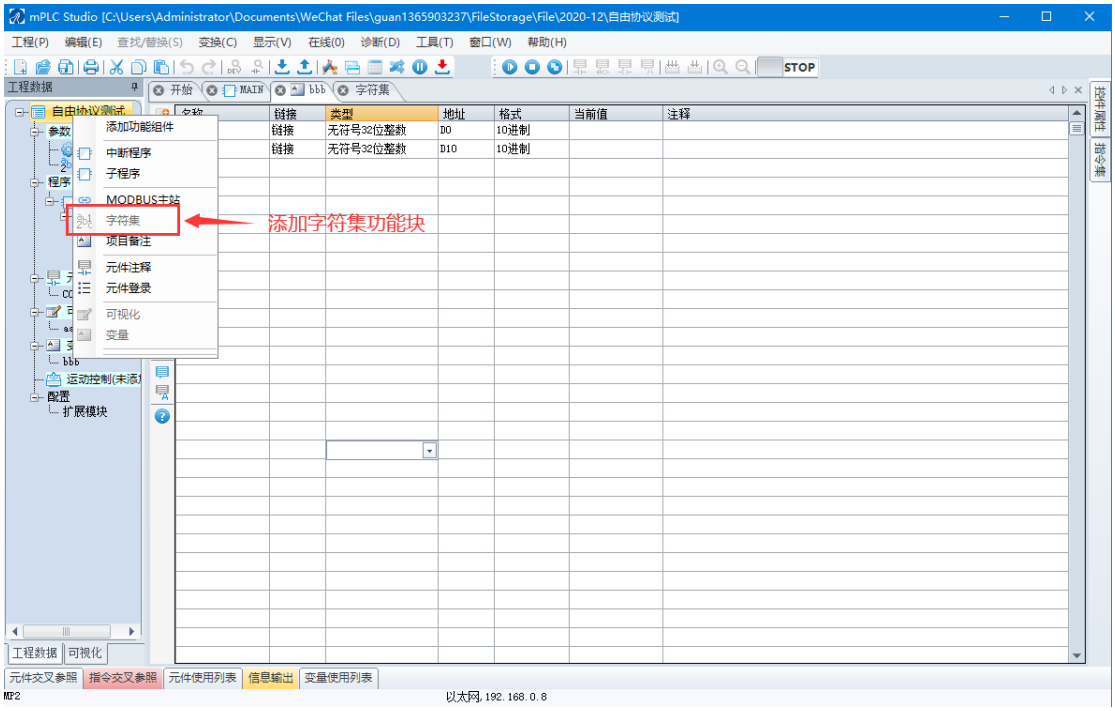
16bit 6 步		32bit 6 步		指令格式
REVS	\	\	\	REVS “S”

操作数的数据类型如下表

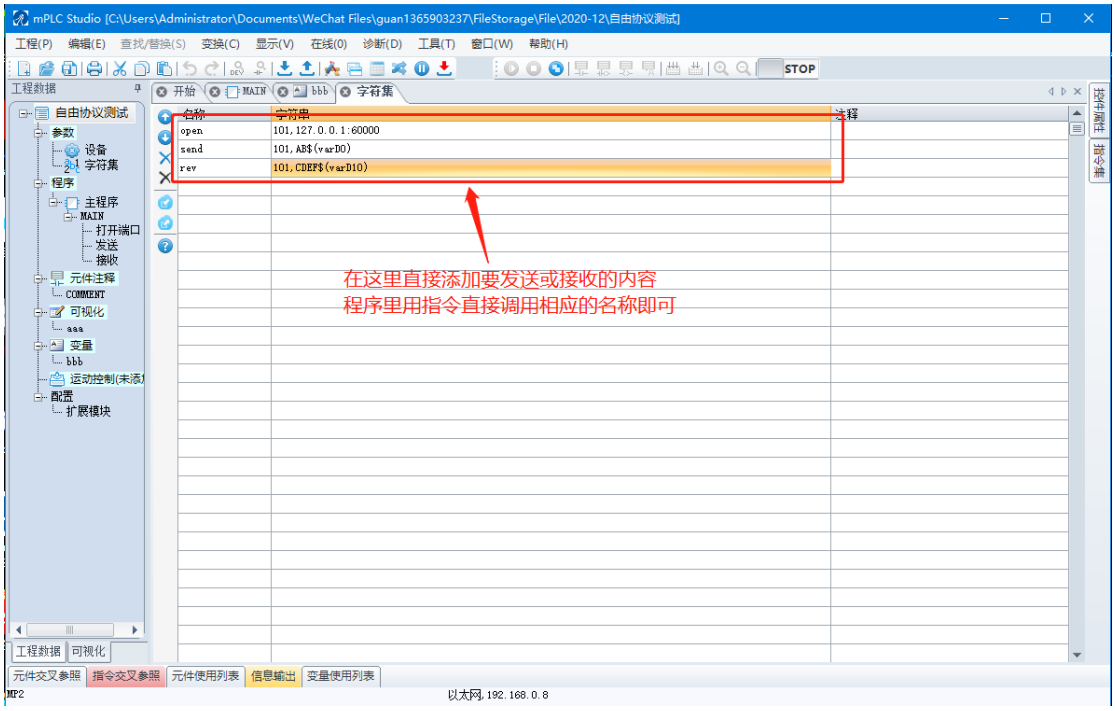
操作数	内容	类型
S	字符集表格内的“配置名称”	

1.14.4 字符集使用说明

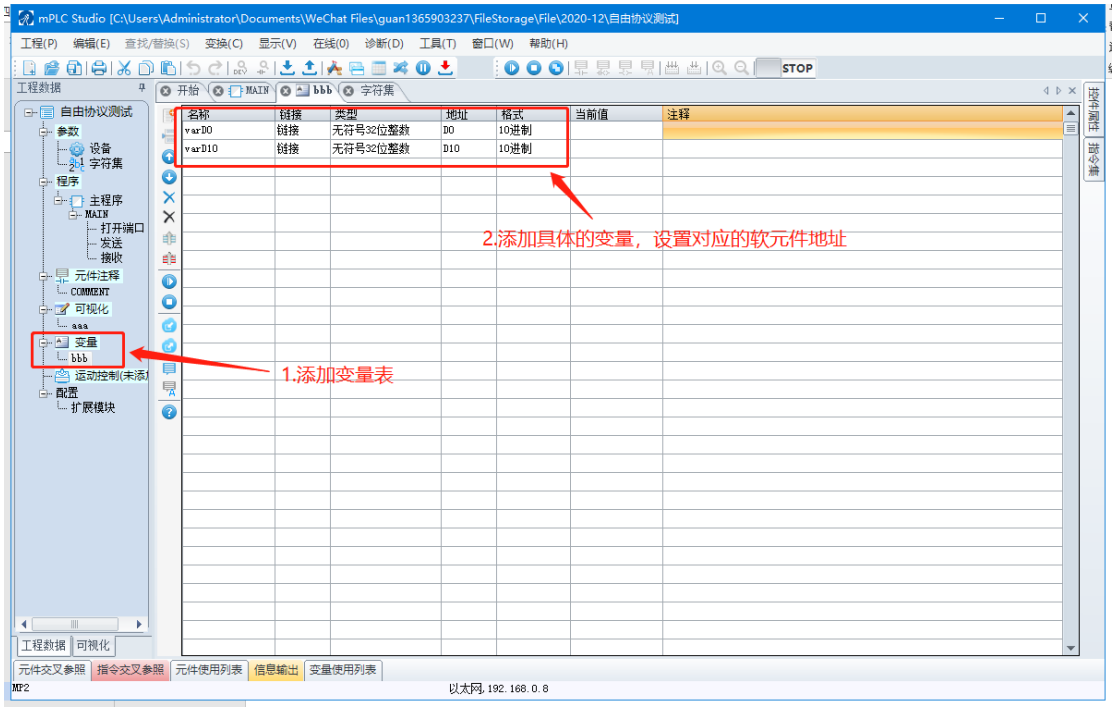
1.编写字符集程序。新建项目工程程序，在左侧项目数据中添加字符集功能。



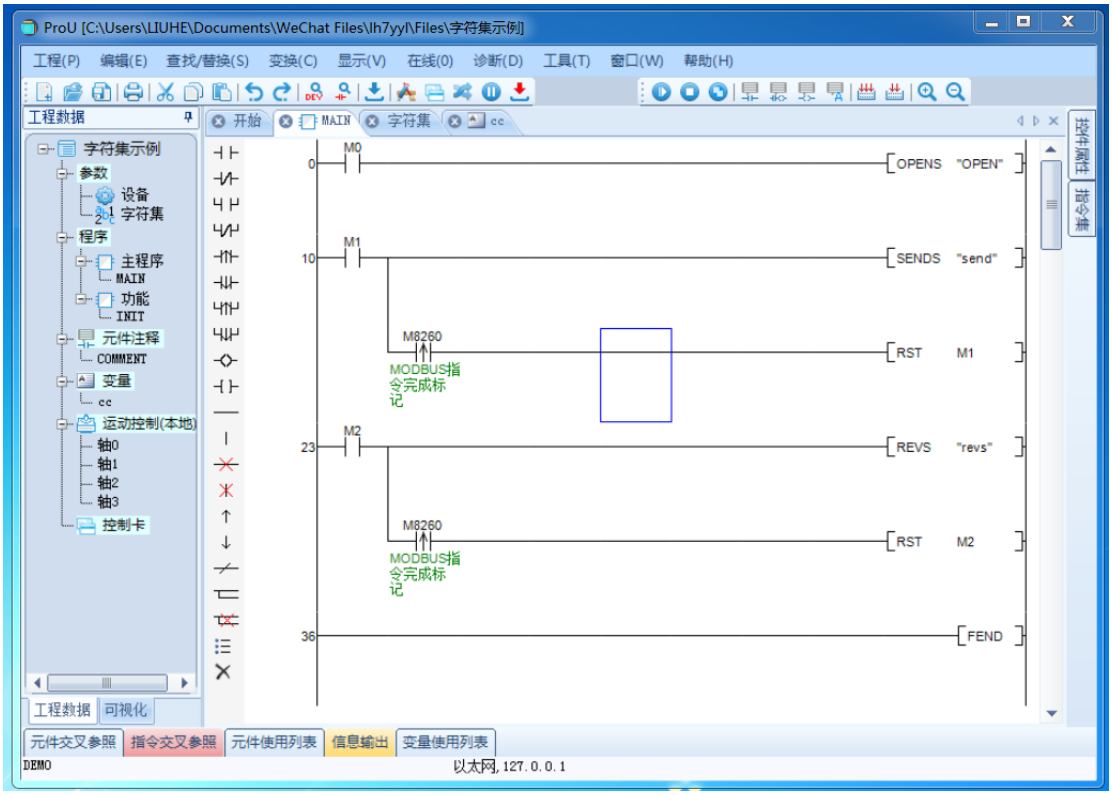
2. 打开字符集设置界面，字符集的数据发送和接收均以填表方式编写。程序中调用相应的行即可。打开端口按固定格式填写：100 表示以太网识别号（100~109），127.0.0.1 表示对应 IP 地址；8000 表示对方端口号； send 行表示：字符串 ABC 和变量 varD0 中的内容通过 100 端口发送； revs 行表示：通过端口 100 接收字符串 CDEF 和数据存放到变量 varD10 中。



3. 设置变量，varD0 映射到 D0 软元件，varD10 映射到 D10 软元件



4. 编写梯形图程序。 OPENS 指令表示打开端口，需要一直保持接通状态；SENDS 指令表示将 send 字符集中的内容发送到端口，发送成功后可用 M8029 标志位进行复位；REVS 指令表示从端口接收到的内容，接收成功后可用 M8029 标志位进行复位。程序编写完成后，需要下载到 PLC，并重启。



1.15 Modbus 参数设置

设置系统块中的通讯口

在通讯口界面中有两个串口的选择，PORT1 和 PORT2，PORT1 为 RS232 串口，PORT2 为 RS485 串口，均支持 Modbus 主站和从站。

设置 Modbus 通讯协议参数

在 Modbus 通讯协议操作数界面中，有个默认值按钮，默认值是 Modbus 通讯协议推荐的通讯设置。参数设置选项如下表所示。

选项	设置内容
站号	0～247
波特率	115200，57600，38400，19200，9600，4800，2400，1200，600，300

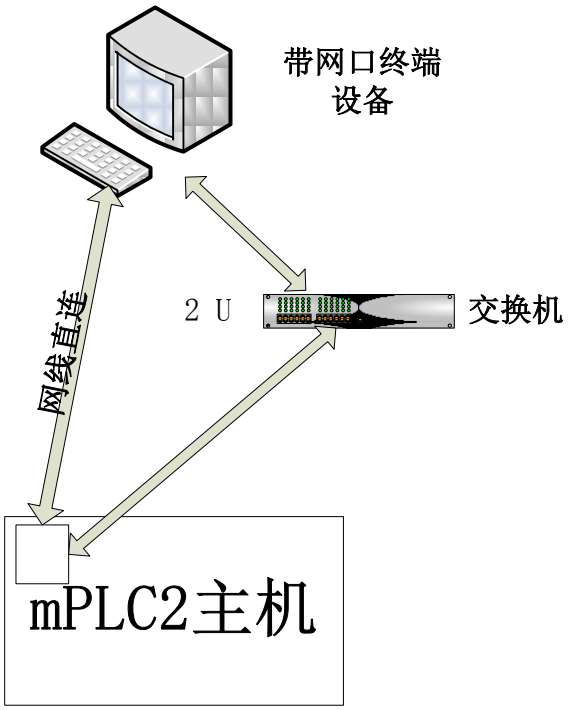
选项	设置内容
数据位	设定 7 或 8，ASCII 模式 7 位，RTU 模式 8 位
奇偶校验位	设定为无校验、奇校验、偶校验
停止位	设定 1 或 2，奇、偶校验时设定为 1，无校验时设定为 2
Modbus 主/从	均可设为主站或从站
传输模式	选择 RTU 模式或 ASCII 模式
注：当操作数在系统块中设定并下载后，不是立即有效，必须运行一次，才能生效。	

1.16 以太网通讯说明

1.16.1 Modbus/TCP

mPLC2 主机的 Modbus 协议中，带有 Modbus/TCP 的支持，硬件直接通过网口可以进行连接。

Modbus/TCP 的网络拓扑示例：



如上所示，带网络支持的终端设备可以通过网线跟 mPLC2 主机进行直连，或者通过交换机、路由器等中继设备进行间接连接，均可以构成通讯网络。

1. mPLC2 主机作为 Modbus/TCP 从站

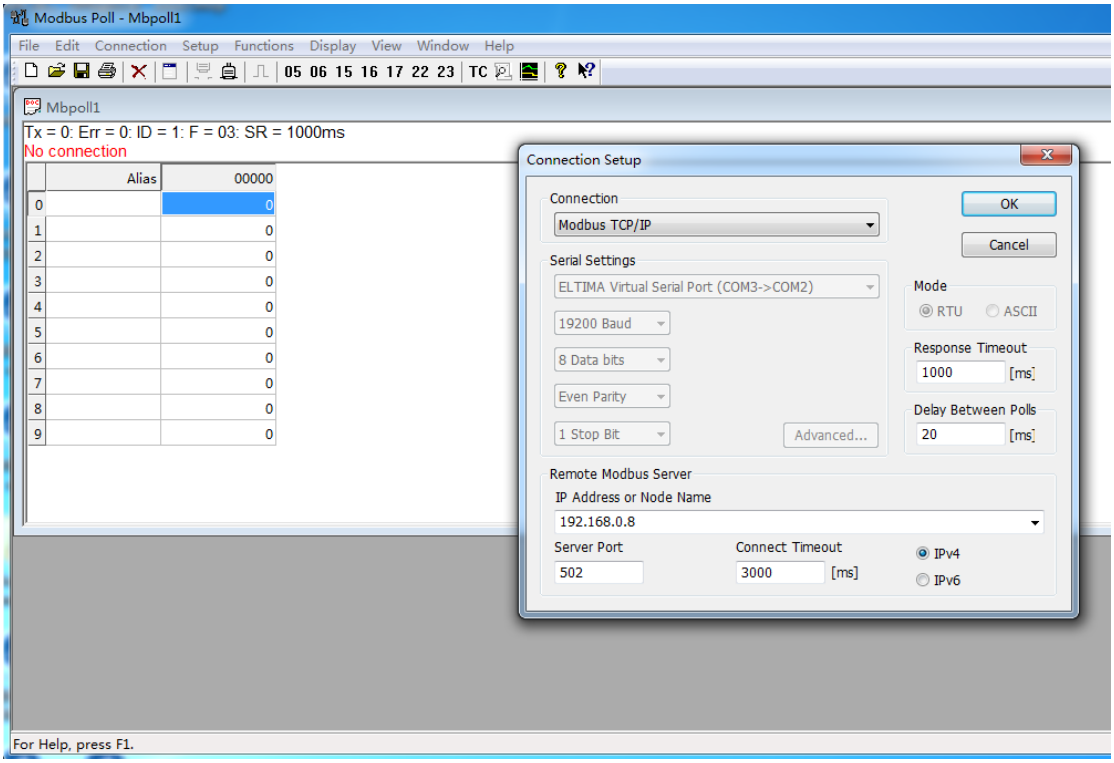
当 PLC 主机作为 Modbus/TCP 从站时，网络的另一端作为主站，可以通过发送符合 Modbus/TCP 规定的数据包来跟 PLC 主机进行交互，此时的 TCP 协议两端分别为：

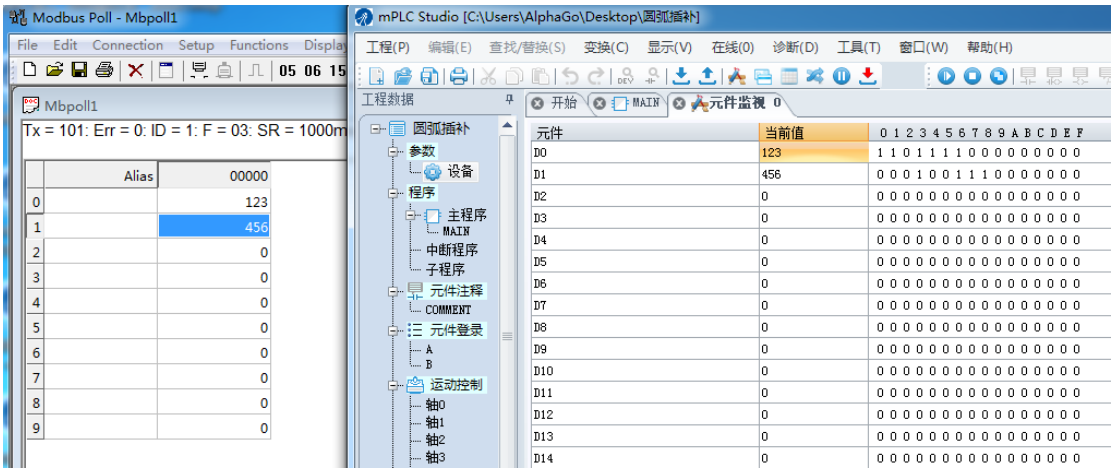
服务器端：PLC 主机，IP 地址：192.168.0.8（默认，可以更改），端口：502

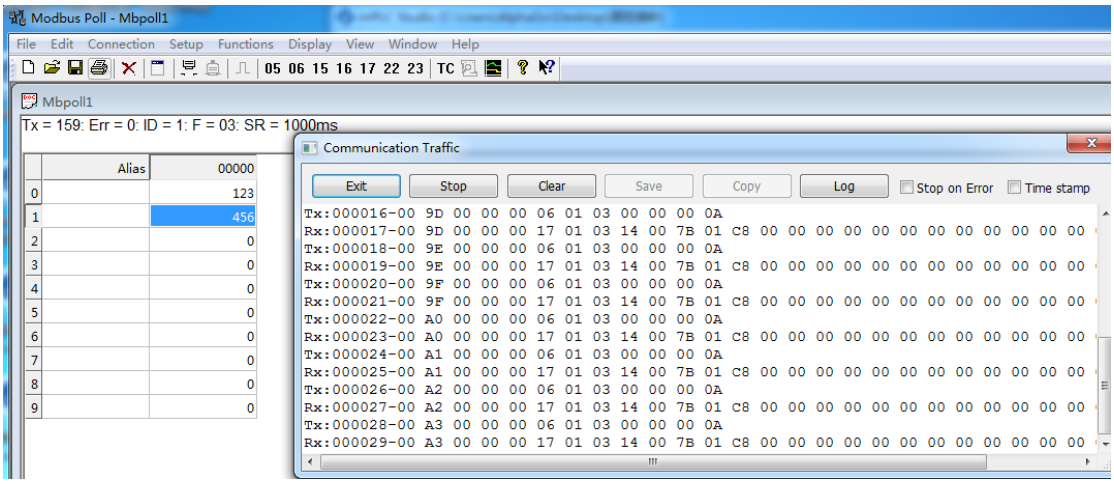
客户端：远程设备，IP 地址：192.168.0.xx（跟服务器同一网段）

远程设备可以作为客户端，通过上面给出的服务器端 IP 地址和端口，跟 PLC 主机进行 TCP 协议的链接，然后进行数据交互。

以 Modbus 的测试工具“Modbus Poll”为例，在建立连接时选择“Modbus TCP/IP”，IP 地址填入“192.168.0.8”，服务器端口填入“502”，点击【OK】，即可建立与 PLC 主机的链接。


Modbus Poll - Mbpoll1
File Edit Connection Setup Functions Display View Window Help
05 06 15 16 17 22 23 TC
Mbpoll1
Tx = 0: Err = 0: ID = 1: F = 03: SR = 1000ms
No connection
Alias 00000
Connection Setup
Connection
Modbus TCP/IP
OK
Cancel
Serial Settings
ELTIMA Virtual Serial Port (COM3->COM2)
19200 Baud
8 Data bits
Even Parity
1 Stop Bit
Advanced...
Mode
RTU ASCII
Response Timeout
1000 [ms]
Delay Between Polls
20 [ms]
Remote Modbus Server
IP Address or Node Name
192.168.0.8
Server Port
502
Connect Timeout
3000 [ms]
IPv4
IPv6
For Help, press F1.


Modbus Poll - Mbpoll1
Mbpoll1
Tx = 101: Err = 0: ID = 1: F = 03: SR = 1000m
Alias 00000
mPLC Studio [C:\Users\AlphaGo\Desktop\圆弧插补]
工程(P) 编辑(E) 查找/替换(S) 变换(C) 显示(V) 在线(0) 诊断(D) 工具(T) 窗口(W) 帮助(H)
工程数据
开始 MAIN 元件监视 0
圆弧插补
参数
设备
程序
主程序
MAIN
中断程序
子程序
元件注释
COMMENT
元件登录
A
B
运动控制
轴0
轴1
轴2
轴3
元件 当前值
D0 123
D1 456
D2 0
D3 0
D4 0
D5 0
D6 0
D7 0
D8 0
D9 0
D10 0
D11 0
D12 0
D13 0
D14 0


Modbus Poll - Mbpoll1
Mbpoll1
Tx = 159: Err = 0: ID = 1: F = 03: SR = 1000ms
Alias 00000
Communication Traffic
Exit Stop Clear Save Copy Log Stop on Error Time stamp
Tx:000016-00 9D 00 00 00 06 01 03 00 00 00 0A
Rx:000017-00 9D 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000018-00 9E 00 00 00 06 01 03 00 00 00 0A
Rx:000019-00 9E 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000020-00 9F 00 00 00 06 01 03 00 00 00 0A
Rx:000021-00 9F 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000022-00 A0 00 00 00 06 01 03 00 00 00 0A
Rx:000023-00 A0 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000024-00 A1 00 00 00 06 01 03 00 00 00 0A
Rx:000025-00 A1 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000026-00 A2 00 00 00 06 01 03 00 00 00 0A
Rx:000027-00 A2 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00
Tx:000028-00 A3 00 00 00 06 01 03 00 00 00 0A
Rx:000029-00 A3 00 00 00 17 01 03 14 00 7B 01 C8 00 00 00 00 00 00 00 00 00 00 00

可以看到“Modbus Poll”软件发出的数据都是符合 Modbus/TCP 协议规范的数据帧，PLC 主机也回复了正确的数据回包。

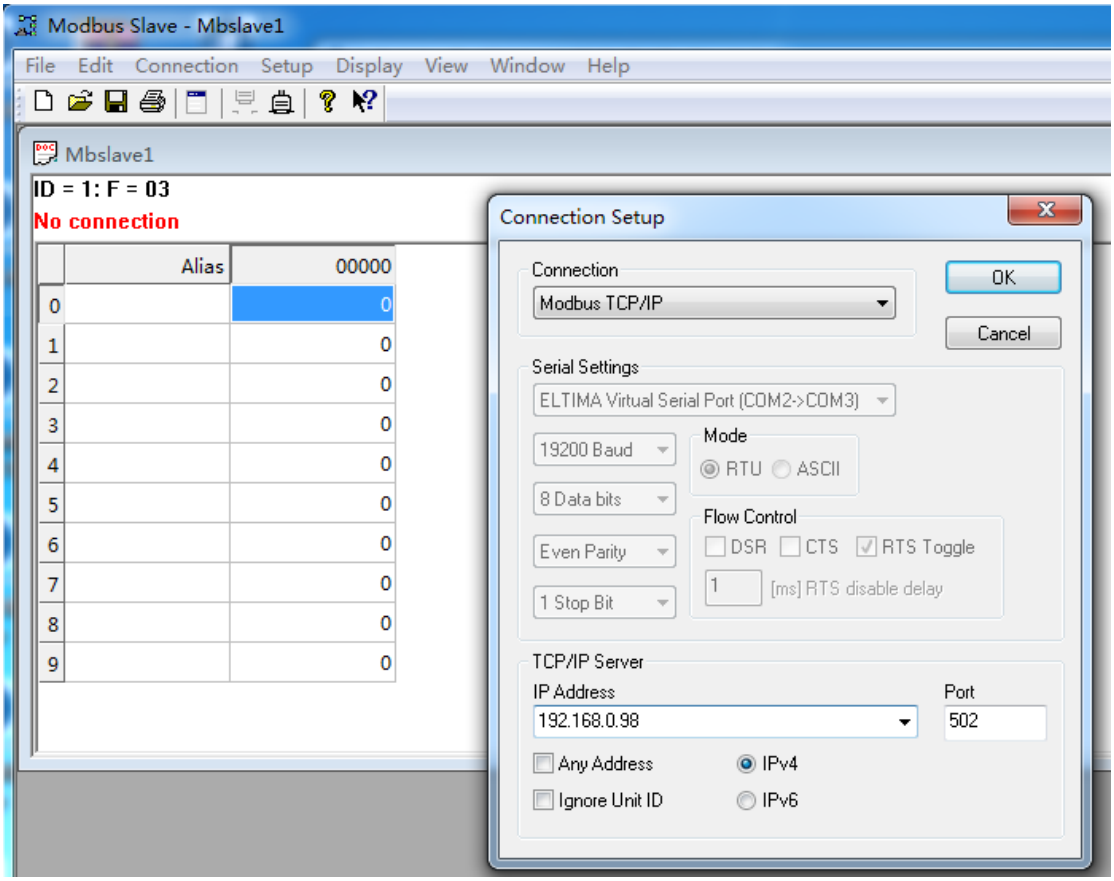
2. mPLC2 主机作为 Modbus/TCP 主站

当 PLC 主机作为 Modbus/TCP 主站去访问从站时，根据上一节的内容可以知道，此时的 TCP 协议两端分别为：

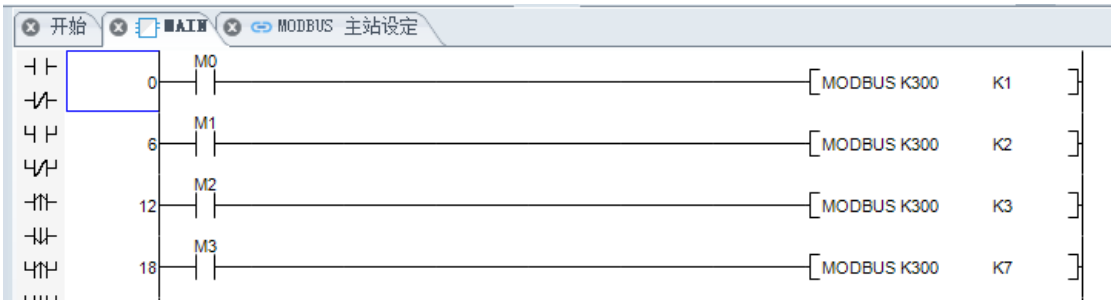
服务器端：远程设备，IP 地址：192.168.0.xx（设备自定义），端口：502

客户端：PLC 主机，IP 地址：192.168.0.xx（跟服务器同一网段）

以 Modbus 的测试工具“Modbus Slave”为例，在建立连接的时候选择“Modbus TCP/IP”，IP 地址填入“192.168.0.98”，服务器端口填入“502”，点击【OK】，即可建立一个服务器端。



PLC 主机端要在【MODBUS 主站】中进行通信参数和命令的设置，然后用户程序中进行指令调用来跟远程服务器进行连接，并通信。



根据上面的用户程序的逻辑，当 M0 为 ON 时，PLC 主机就会执行【MODBUS 主站】中设定的行号为 1 的命令，首先通过 IP 地址和端口建立 TCP 连接，成功后根据设置的命令和参数发送对应的数据帧给远程服务器（Modbus/TCP 从站），然后等待远程服务器的数据回包，最后对收到的数据包进行协议解析和数据操作。

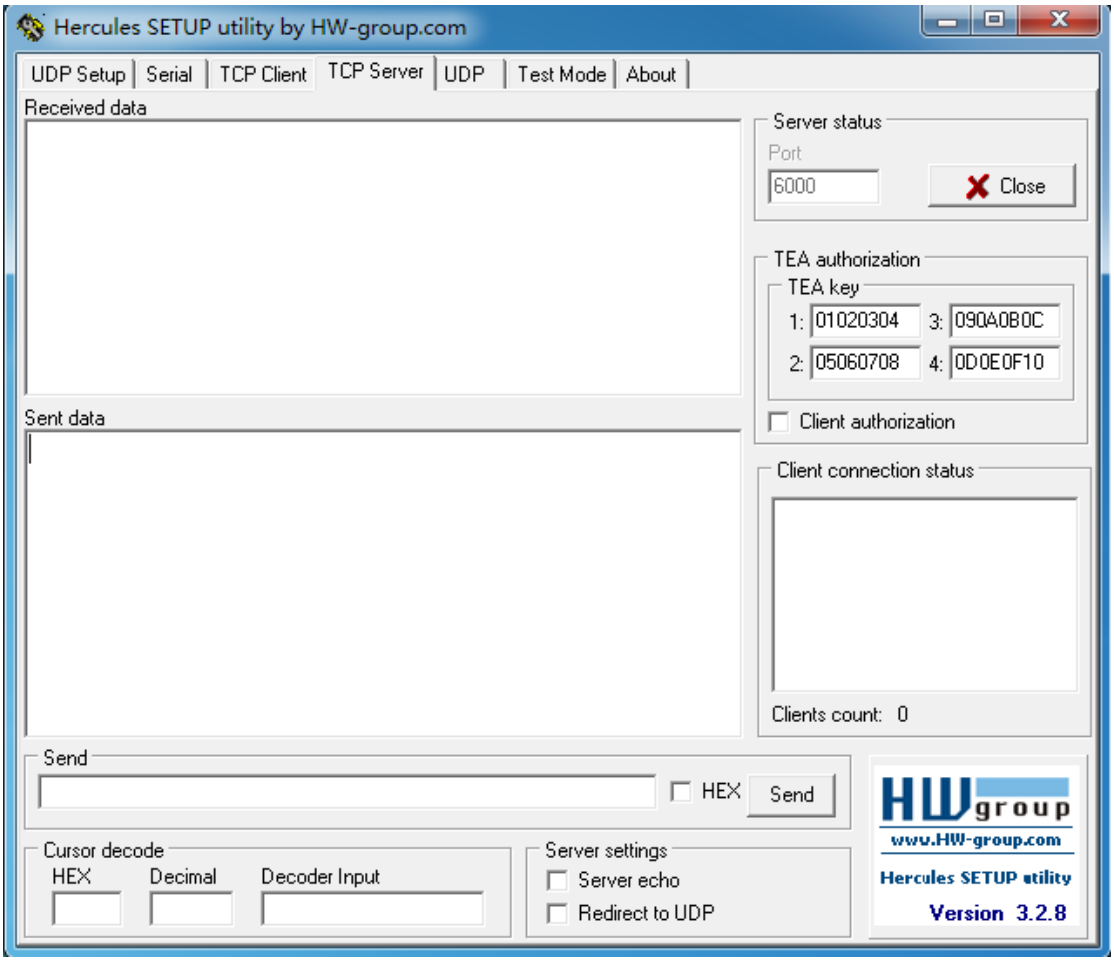
1.16.2 通过 TCP 协议使用字符集功能

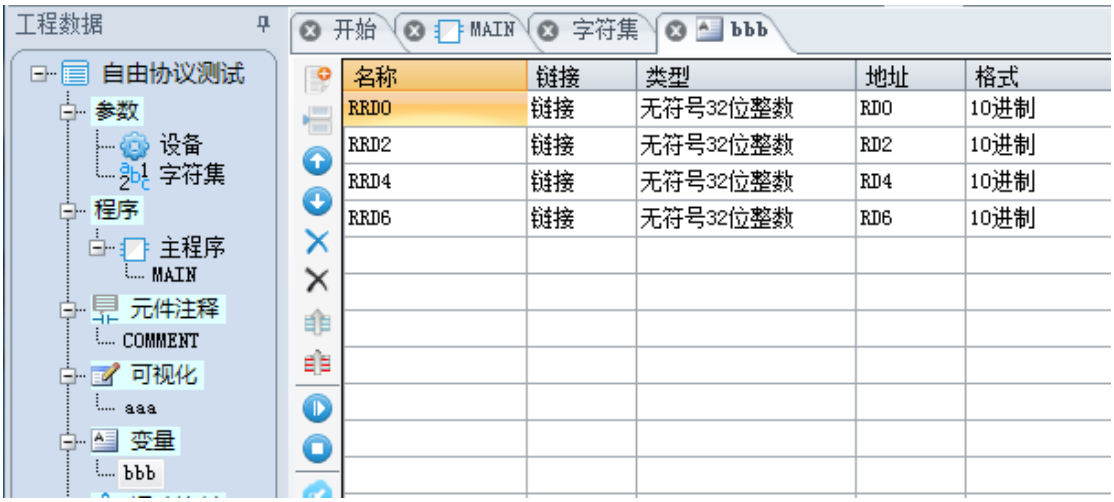
关于字符集功能的说明参考 1.14 节的内容。

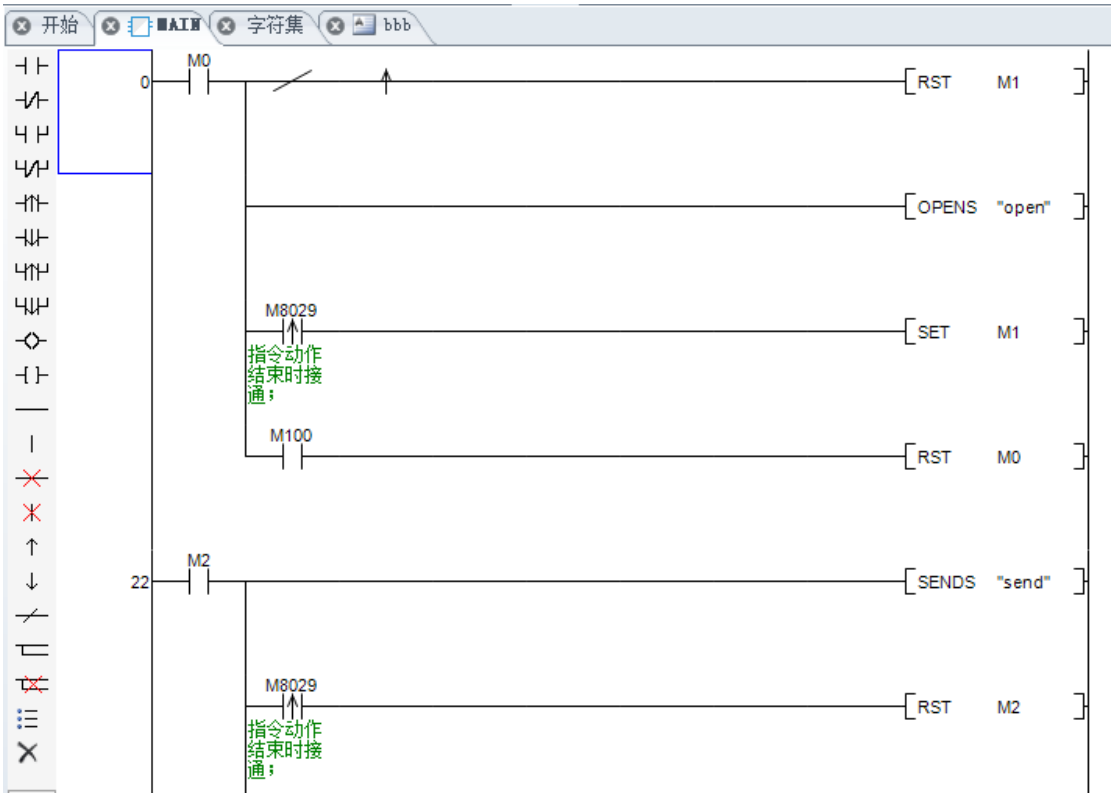
字符集功能同时支持串口和以太网硬件连接，当使用以太网进行硬件连接时，mPLC2 主机会作为 TCP 协议中的客户端，对远程服务器进行连接和数据交互。PLC 主机的 IP 地址就是作为客户端的 IP 地址，要与远程服务器在同一个网段中才能进行通信。

在设定字符集的 OPENS 命令字符时，可以自由设定远程服务器的 IP 地址和端口。

链接建立成功后，便可以使用 SENDS 和 REVS 指令，让 PLC 主机以客户端的形式发送相关的字符串给远程服务器，和从远程服务器接收相关的字符串，PLC 主机接收到字符串后根据定义的格式进行数据提取，最后赋值给对应的 PLC 寄存器。




工程数据
自由协议测试
参数
设备
字符集
程序
主程序
MAIN
元件注释
COMMENT
可视化
aaa
变量
bbb
开始
MAIN
字符集
bbb
名称 链接 类型 地址 格式
RRD0 链接 无符号32位整数 RD0 10进制
RRD2 链接 无符号32位整数 RD2 10进制
RRD4 链接 无符号32位整数 RD4 10进制
RRD6 链接 无符号32位整数 RD6 10进制


开始
MAIN
字符集
bbb
0
M0
RST M1
OPENS "open"
M8029
SET M1
指令动作
结束时接
通;
M100
RST M0
22
M2
SENDS "send"
M8029
RST M2
指令动作
结束时接
通;