

BT- MB-AC-S

快速启动手册

BEACON GLOBAL TECHNOLOGY

目 录

BT-MB-AC-S 简介	2
模块初始配置	2
配置模块做 MODBUS RTU 主站	4
Modbus 命令使能控制介绍	12
配置模块做 MODBUS RTU 从站	14
Modbus RTU 诊断方式	17
配置模块 ASCII 码协议端口参数	18
举例 1. Modbus 设备和 ASCII 设备交换数据	20
联系我们	26

BT-MB-AC-S 简介

BT-MB-AC-S系列网关是Modbus RTU®— Generic ASCII通讯网关模块，支持在Modbus RTU和Generic ASCII网络中的设备之间的双向数据交换，最大2000个字数据交换区。



E1 以太网端口==配置组态。

S1 上端口==Modbus RTU 可配置为 Modbus RTU 主站或者从站, (RS232/422/485)。

S2 下端口==Generic ASCII 码 可配置为 ASCII 接收和发送, (RS232/422/485)。

模块初始配置

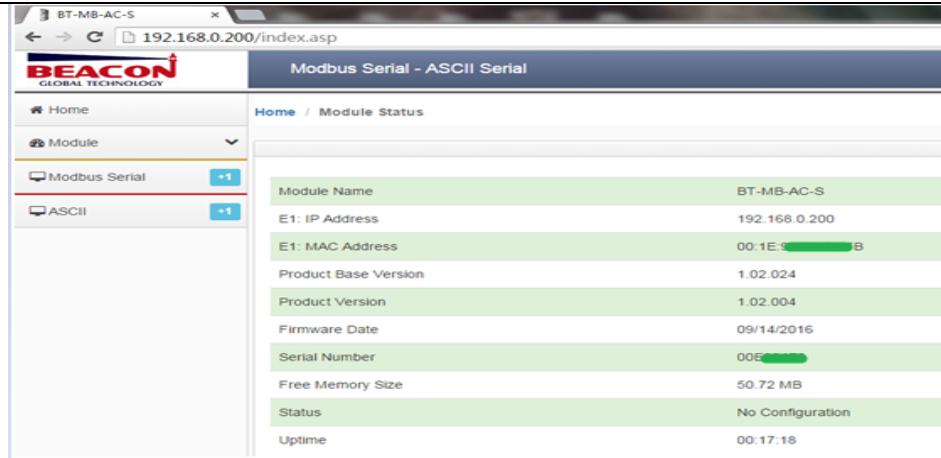
E1 以太网接口 IP 地址 192.168.0.200。 (OLED 显示屏上显示 IP 地址信息)

模块上电后，OLED显示屏上会滚动显示IP地址，方便查找模块不同接口的IP地址。本案例中网关使用IP地址为192.168.0.200。

BT系列网关全部采用网页配置形式组态，无需安装其他多余的组态软件，推荐采用如下浏览器及以上版本（更好的支持HTML5的功能）对于网关进行配置：IE10, GOOGLE Chrome 35, FIREFOX 35, Safari 7及以上的版本。

通过以太网配置 BT-MB-AC-S 模块：

1. 把本地电脑的IP地址与所连接的模块端口配置成相同的IP网段，例如本案例采用E1接口进行配置，本地电脑配置成192.168.0.177，然后在GOOGLE Chrome浏览器的地址框里面输入192.168.0.200，点击回车键后，进入到 BT-MB-AC-S模块的配置页面如下图：



2. 在配置页面的导航条内，点击Login
3. 按照界面提示，输入用户名和密码进入模块配置。

用户名(Username) : admin

密码(Password) : admin

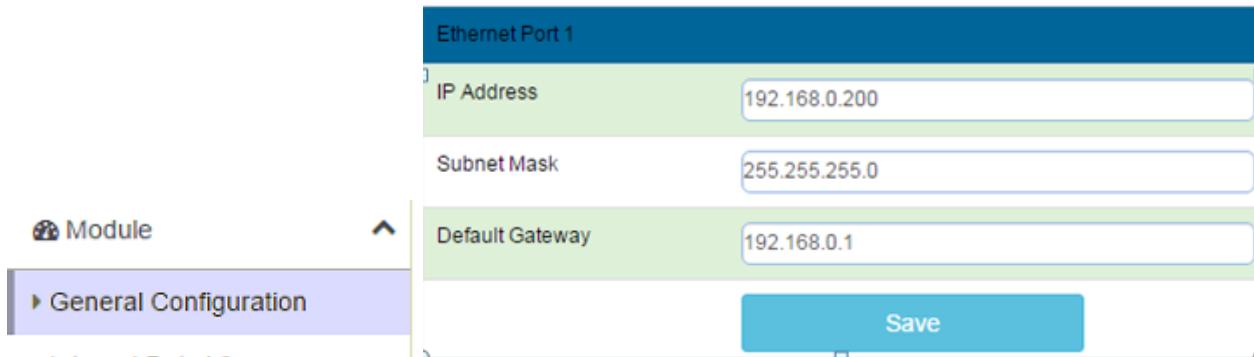
点击登录(Sign In)

请注意：如果不登录，只能浏览配置，无法进行配置修改。

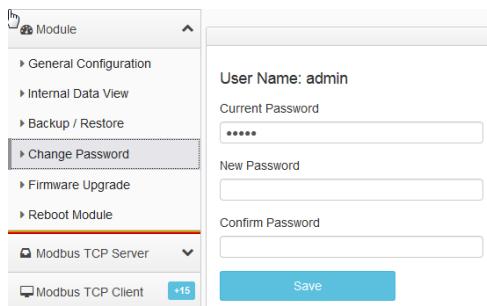
4. 登录后看到导出配置文件 **Export Config** 和恢复配置文件 **选择文件 未选择任何文件**



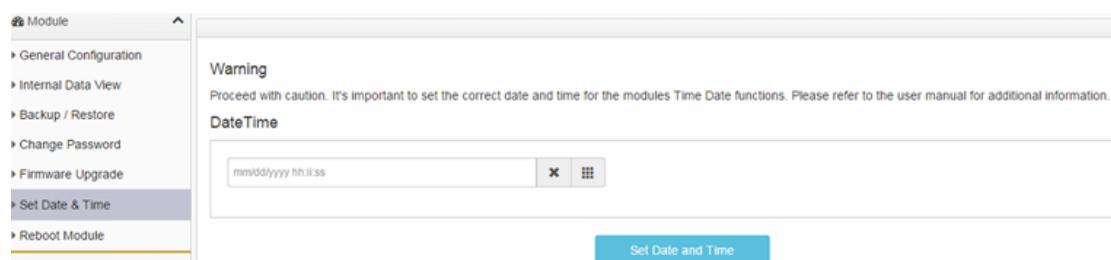
5. 查看模块 IP 地址，点击 **General Configuration**，修改模块的 IP 地址。



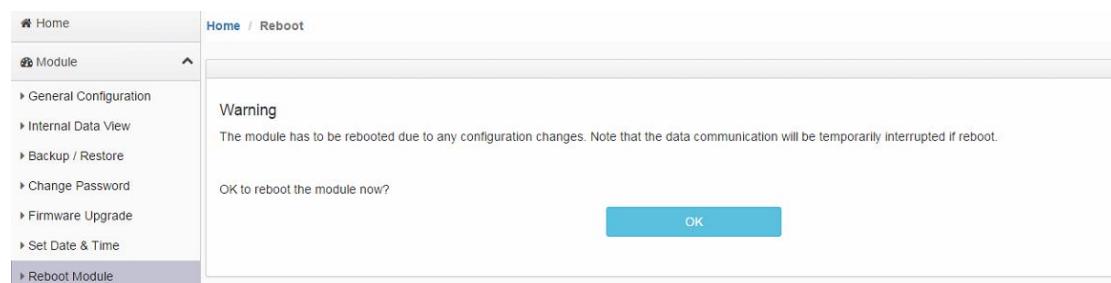
6. 点击修改密码，可以修改模块的登录密码。▶ **Change Password**



7. 点击 **Set Date & Time** 可以设置模块的日期和时间。



8. 点击 **Reboot Module** 表示重启模块。(不是复位)

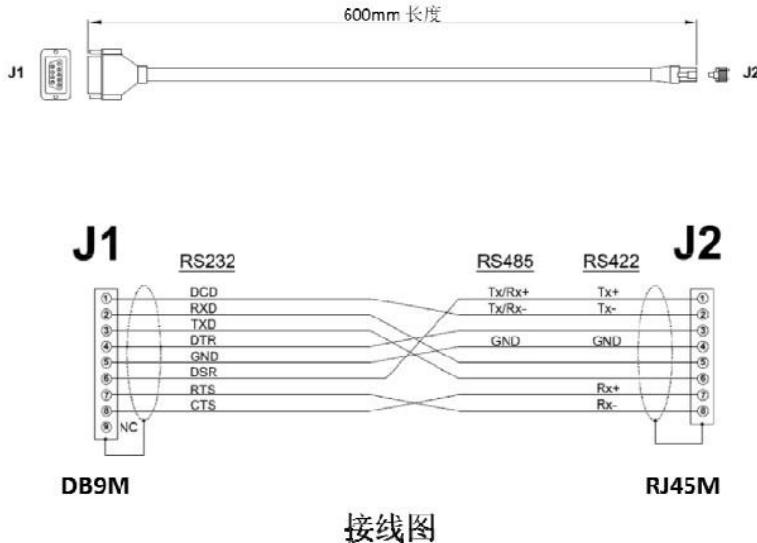


配置模块做 MODBUS RTU 主站

Modbus RTU 主站可以连接 31 个从站，RS485 接线方式长度在 1200 米以内。工程师设计连接每个主站连接从站个数可参考如下原则：

- 1、遵循 MODBUS RTU 通讯规约。
- 2、主站只读取从站数据，每个 RS485 串口主站可以接 31 个从站，MODBUS RTU 是令牌轮询方式，连接从站越多，或者距离越长，延时越大。

3、主站同时读写从站数据，建议每个 RS485 串口最多接 10-15 个从站，避免过长通讯延时，提升通讯响应速度。



上图为 S1/S2 端口的接线图

举例：S1 口引出来 RS485 接口，端子 6+, 1-

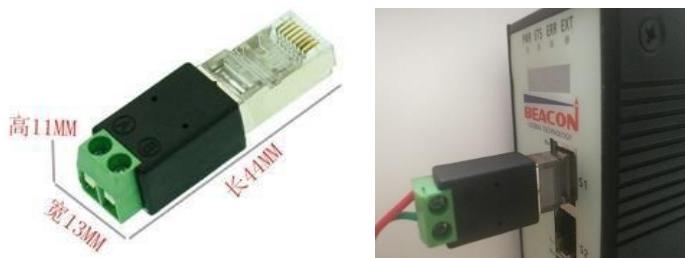
USB 转 RS485 引出来的 485 接口，端子 T/R+, T/R-

6+-----端子 T/R+

1-----端子 T/R-



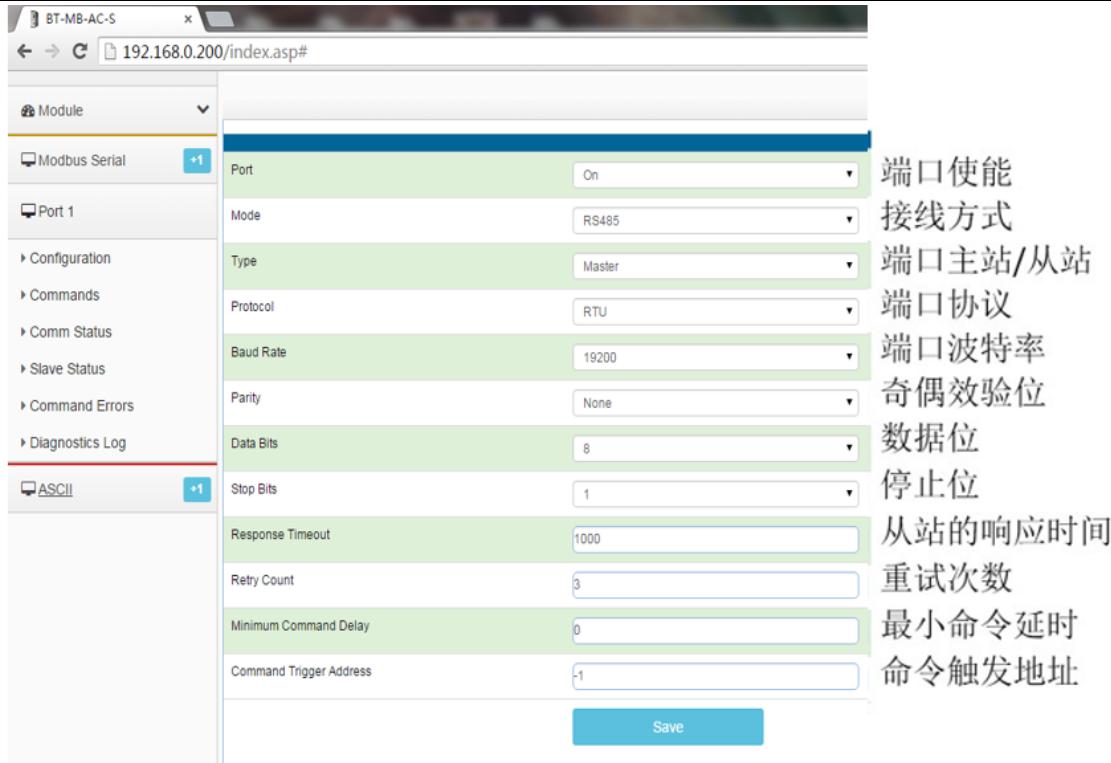
或水晶头直插网关串口。



举例：S1 口引出来 RS232 接口，端子 2RX, 3TX, 5GND 接线。

如果选用 RS232 接线方式，每个串口只能连接一个从站，接线长度不能超过 15 米。串口注意不能热插拔串口，容易对串口造成不必要的损坏。

打开浏览器，进入模块主配置页面，如下图：



此处模块作为 Modbus 主站，请根据需要连接的 Modbus 从站情况，合理在此页面配置参数。之后，点击 Port1 里面的 Commands 显示 S1 端口命令的配置页面，点击 Add。出现如下指令配置页面。

Modbus Port 1 - Modify Command

Enable	Yes	使能，禁止，内部寄存器有变化后写
Modbus Function	FC 3 - Read Holding Registers(4X)	Modbus 功能码FC1,FC2,FC3,FC4,FC5,FC6,FC15,FC16
Slave Address	1	从站地址
Modbus Data Address	0	从站读写数据Modbus起始位
Quantity	10	读或者写的数据的数量
Data Swap	No Change	数据高低位交换，字交换，字节交换，字和字节交换
Poll Interval	0	命令轮询时间
Internal Data Address	2000	模块内部寄存器，存放数据的起始地址
Cmd Errors Mapping Enabled	Yes	命令错误状态位反馈开启
Cmd Errors Mapping Address	2100	命令错误状态位反馈地址，模块内部寄存器任意位置
Desc		命令描述

Modbus 主站命令解释，采用功能码控制读写区域，注意一定要先确定模块内部数据的范围。以下举例中均采用 BT 模块支持 4000 个字数据区地址范围，实际配置不同型号模块时，请按照真实数据区大小进行指令的使用。模块内部寄存器是16位的INT格式，读写布尔量的时需要注意16倍关系。

Enable	Yes
Modbus Function	FC 3 - Read Holding Registers(4X)
Slave Address	1
Modbus Data Address	0
Quantity	100
Data Swap	No Change
Poll Interval	0
Internal Data Address	2000
Cmd Errors Mapping Enabled	Yes
Cmd Errors Mapping Address	2501
Desc	

以上指令含义如下：模块使用功能码 FC3，从站数据起始地址是 0 等于 40001. 读取数量是 100. 模块内部寄存器起始地址 2000。表示读 1 号从站，从站数据地址范围为 40001–40100 的 100 个字，放到模块内部寄存器 2000–2099，命令没有正确返回在内部寄存器 2051 报错。

如果功能码是 FC4 时（只读），从站数据起始地址是 0 等于 30001. 读取数量是 100. 模块内部寄存器起始地址 2000，表示读 1 号从站，从站数据地址范围为 30001–30100，放到模块内部寄存器 2000–2099，命令没有正确返回，会在内部寄存器 2051 报错。

Enable	Yes
Modbus Function	FC 1 - Read Coil (0X)
Slave Address	1
Modbus Data Address	0
Quantity	16
Data Swap	No Change
Poll Interval	0
Internal Data Address	32000
Cmd Errors Mapping Enabled	Yes
Cmd Errors Mapping Address	2501
Desc	

以上指令含义如下：模块使用功能码 FC1 时，从站数据起始地址是 0 等于 00001，读取数量是 16（此处读取 16 个位等于读取一个字）。模块内部寄存器起始地址 32000（此处为位地址，读取 16 个位等于读取一个字，模块内部寄存器是字，所以实际上模块内部寄存器的起始地址为 $32000/16=2000$ ）。表示读 1 号从站，从站数据地址范围为 00001–00016，放到模块内部寄存器起始地址为 2000（因为读取到 16 个位数据，等于 1 个字数据，所以只占用模块内部寄存器一个地址），命令没有正确返回在内部寄存器 2051 报错。

如果是功能码 FC2 时（只读），从站数据起始地址是 0。读取数量是 16。模块内部寄存器 32000，同上表示读 1 号从站，从站数据地址范围为 00001–00016，放到模块内部寄存器 2000，命令没有正确返回，会在内部寄存器 2051 报错。

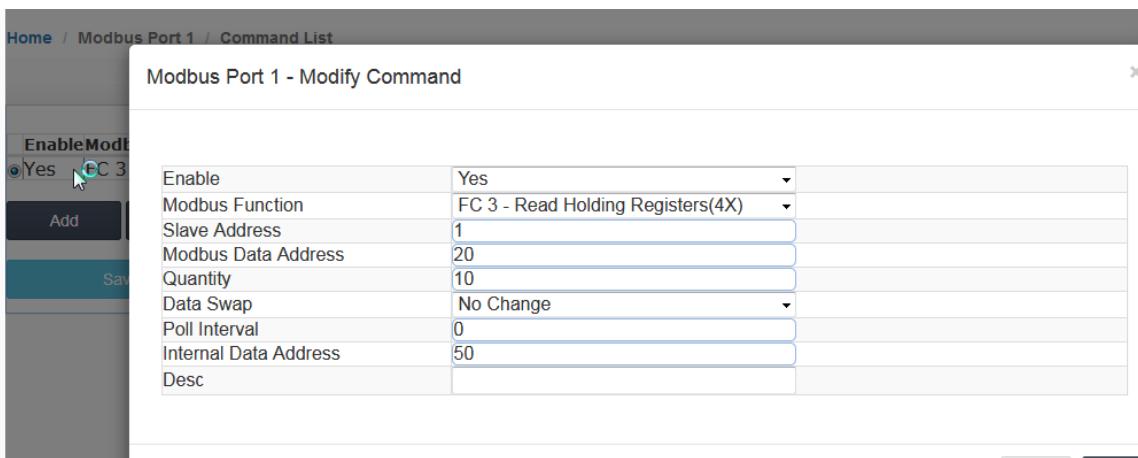
Enable	Conditional
Modbus Function	FC 16 - Preset (Write) Multiple Register
Slave Address	1
Modbus Data Address	50
Quantity	20
Data Swap	No Change
Poll Interval	0
Internal Data Address	2000
Cmd Errors Mapping Enabled	Yes
Cmd Errors Mapping Address	2501
Desc	

以上指令含义如下：Conditional 表示有条件情况下，模块使用功能码 FC6 或者 FC16 时，写出数量是 20。模块内部寄存器起始地址为 2000，表示当模块内部寄存器范围 2000–2019 的任意寄存器发生数据发生变化时候，触发一条写的命令，数据从模块写到 1 号从站，从站接收数据地址范围为 40051–40070，命令没有正确执行，会在内部寄存器 2051 报错。

Enable	Yes
Modbus Function	FC 16 - Preset (Write) Multiple Register
Slave Address	1
Modbus Data Address	50
Quantity	20
Data Swap	No Change
Poll Interval	0
Internal Data Address	2000
Cmd Errors Mapping Enabled	Yes
Cmd Errors Mapping Address	2051
Desc	

以上指令含义如下：模块功能码 FC6 或者 FC16 时，写入数量是 20。模块内部寄存器起始地址 2000。表示内部寄存器范围 2000–2019 的数据，一直连续的写出到 1 号从站，从站接收数据的地址范围为 40051–40070，命令没有正确执行，会在内部寄存器 2051 报错。

举例：新建一条命令：



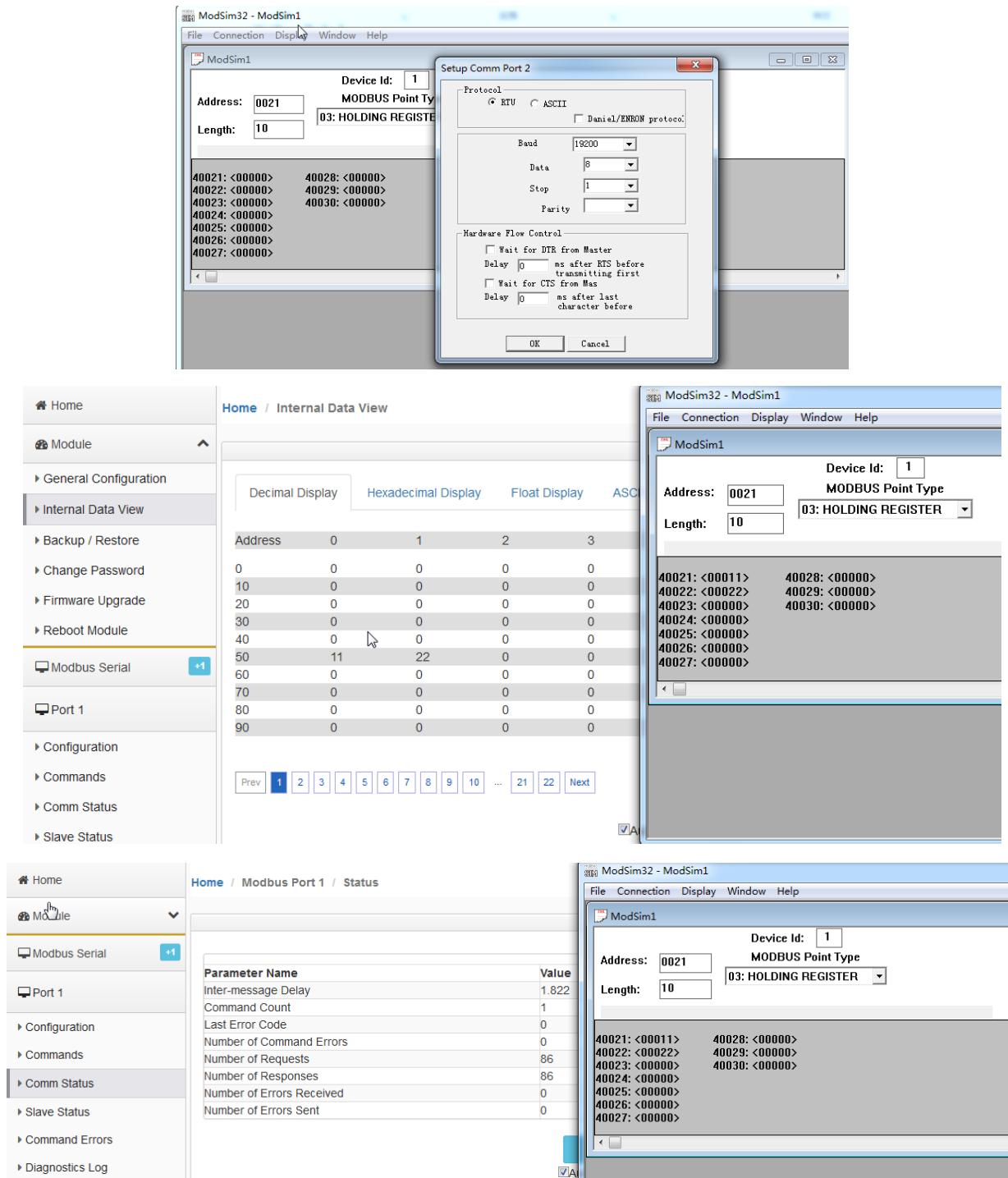
The screenshot shows the 'Modbus Port 1 - Modify Command' dialog box. On the left, there's a sidebar with navigation links like Home, Module, Modbus Serial, Port 1, Configuration, Commands, Comm Status, Slave Status, Command Errors, and Diagnostics Log. The 'Commands' link is currently selected. In the main dialog, a dropdown menu shows 'Enable Modbus' with 'Yes' selected. Below it, there are fields for Modbus Function (set to 'FC 3 - Read Holding Registers(4X)'), Slave Address (1), Modbus Data Address (20), Quantity (10), Data Swap (No Change), Poll Interval (0), Internal Data Address (50), and Desc (empty). Buttons for 'Add' and 'Save' are visible at the bottom of the dialog.

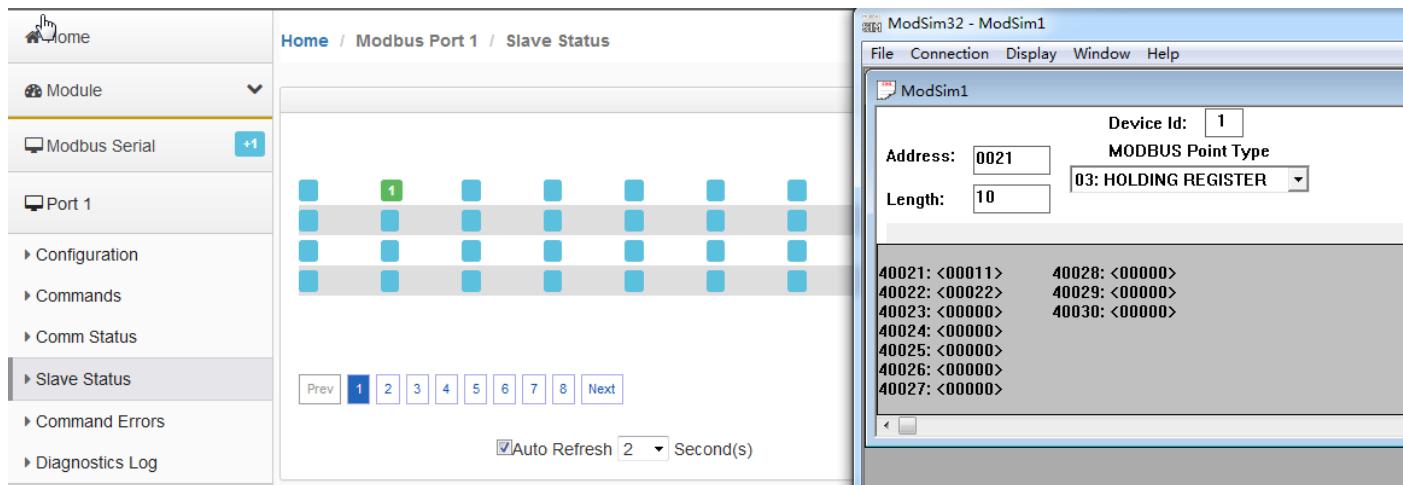
命令含义：把 MODBUSRTU 从站的 40021–40030 这 10 个 16 位的数值读到内部数据库 50–59 里面。

配置完点 SAVE，然后关闭这个命令，再点击 SAVE，把这个命令保存到命令列表里面，然后根据提示重启模块。

用户在配置好模块 MODBUS RTU 主站端口后，可以利用 MODBUSRTU 仿真软件 MODSIM32，作为 MODBUS RTU 从站，仿真测试与模块主站端口通讯。

打开 MODSIM32 软件，配置端口 2 参数，从 40021-40030 写入十个数据。点击 OK。可以看到模块主站对应的内部数据区也相应的显示出从站的数值变化。





举例介绍 Cmd Errors Mapping Enabled 和 Cmd Errors Mapping Address 这两个参数。

Cmd Errors Mapping Enabled 表示命令错误是否映射，选择 YES 表示使用，选择 NO，表示不使用。

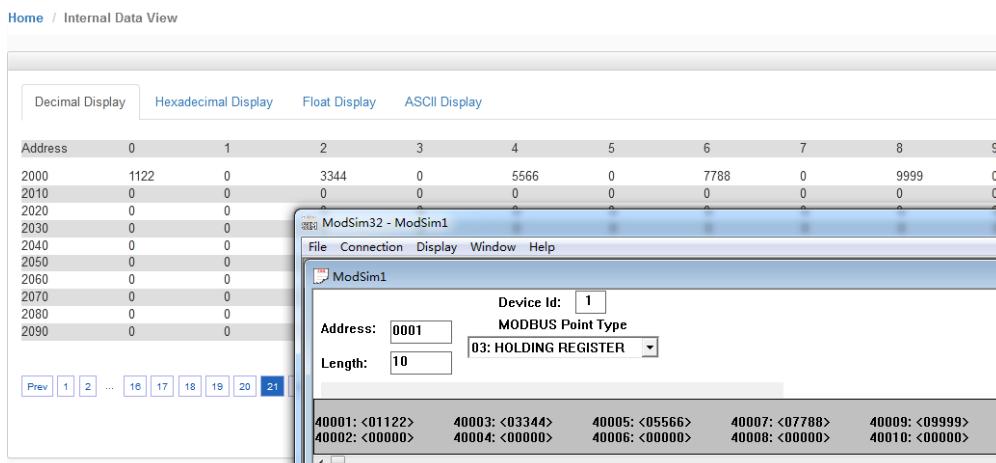
Cmd Errors Mapping Address 表示命令错误映射的地址。

Modbus Port 1 - Modify Command

Enable	Yes	使能，禁止，内部寄存器有变化后写
Modbus Function	FC 3 - Read Holding Registers(4X)	Modbus 功能码FC1,FC2,FC3,FC4,FC5,FC6,FC15,FC16
Slave Address	1	从站地址
Modbus Data Address	0	从站读写数据Modbus起始位
Quantity	10	读或者写的数据的数量
Data Swap	No Change	数据高低位交换，字交换，字节交换，字和字节交换
Poll Interval	0	命令轮询时间
Internal Data Address	2000	模块内部寄存器，存放数据的起始地址
Cmd Errors Mapping Enabled	Yes	命令错误状态位反馈开启
Cmd Errors Mapping Address	2100	命令错误状态位反馈地址，模块内部寄存器任意位置
Desc		命令描述

上图命令表示：读取 1 号从站，从站数据地址范围 40001-40010，这 10 个数放到内部起始地址为 2000 的连续 10 个寄存器内（2000-2009），如果发送错误，错误反馈会放到内部寄存器 2100 里面。

指令执行效果如下图显示：



通过查看命令状态 (comm Status) 可以看到命令执行情况，通过点击 Mosim32 菜单栏显示报文，可以查看从站与主站的发送和接收报文的情况。



通过查看从站状态可以直接看到从站的状态，1-31 路都可以直观看到：

Parameter Name	Value
Inter-message Delay	1.822
Command Count	1
Last Error Code	0
Number of Command Errors	0
Number of Requests	566
Number of Responses	565
Number of Errors Received	0
Number of Errors Sent	0

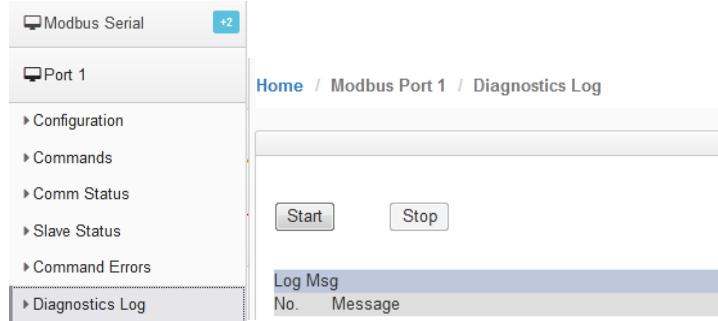
绿色表示线路数据通讯报文都正常，红色表示线路数据通讯报文都不对。

绿色和红色闪烁表示线路正常，数据通讯报文不正常。

通过查看命令错误可以看到从站报的错误值，如果开启了命令反馈功能，这个值也会送到工程师填写的命令错误存放地址里面，如下图。如果命令反馈回一个非零值（如下图寄存器地址 2100 中），则说明该指令执行出错，也可以让 PLC 从模块读取到该数值，从而知道模块指令执行失败。

Address	0	1	2
2100	-2	0	0
2110	0	0	0
2120	0	0	0
2130	0	0	0
2140	0	0	0
2150	0	0	0
2160	0	0	0
2170	0	0	0
2180	0	0	0
2190	0	0	0

通过诊断报文，可以查看主站发送和接收的报文情况。点击 Start，就可以看到下面报文发送和接受的情况。

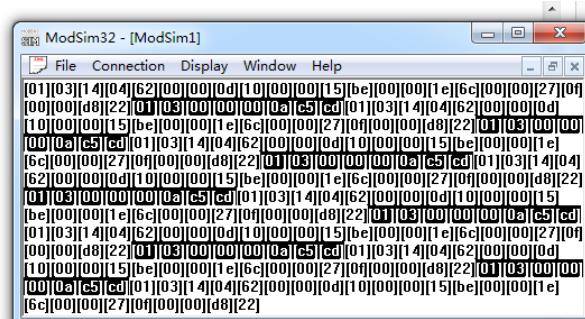


下图为主站发送和接收的报文以十六进制格式显示，Modsim32 也可以从十进制报文切换到十六进制报文显示：

```

693 00:02:14.003 >> 01 03 00 00 00 0A C5 CD
694 00:02:14.105 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
695 00:02:14.309 >> 01 03 00 00 00 0A C5 CD
696 00:02:14.432 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
697 00:02:14.637 >> 01 03 00 00 00 0A C5 CD
698 00:02:14.739 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
699 00:02:14.944 >> 01 03 00 00 00 0A C5 CD
700 00:02:15.046 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
701 00:02:15.250 >> 01 03 00 00 00 0A C5 CD
702 00:02:15.353 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
703 00:02:15.557 >> 01 03 00 00 00 0A C5 CD
704 00:02:15.680 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
705 00:02:15.885 >> 01 03 00 00 00 0A C5 CD
706 00:02:16.007 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22
707 00:02:16.212 >> 01 03 00 00 00 0A C5 CD
708 00:02:16.315 << 01 03 14 04 62 00 00 0D 10 00 00 15 BE 00 00 1E 6C 00 00 27 0F 00 00 D8 22

```



Modbus 命令使能控制介绍

新版本增加了 Modbus RTU 做主站的命令使能控制，这个作用是表示可以控制发出几个命令，比如模块连接了 15 个从站，如果有一个从站坏掉了，这时候 Modbus RTU 网络会变慢，主站每次发送命令会等待这个从站响应，解决的办法是不发送这个从站的命令，具体使用方法如下。

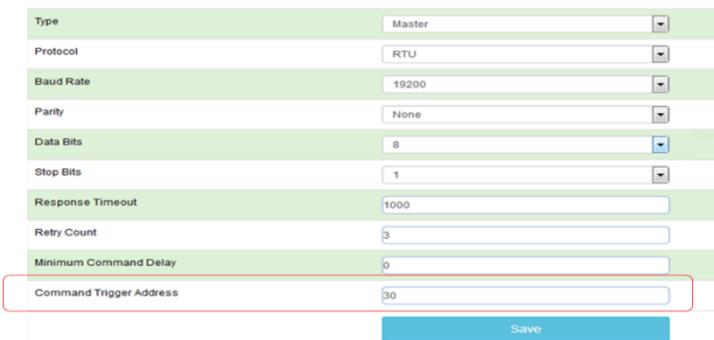
The screenshot shows the 'Command List' configuration screen. It has a table with the following columns: Enable Modbus Function, Slave Address, Modbus Data Address, Quantity, Data Swap, Poll Interval, Internal Data Address, Cmd Errors Mapping Enabled, Cmd Errors Mapping Address, and Desc. There are two rows in the table:

Enable Modbus Function	Slave Address	Modbus Data Address	Quantity	Data Swap	Poll Interval	Internal Data Address	Cmd Errors Mapping Enabled	Cmd Errors Mapping Address	Desc
<input checked="" type="checkbox"/> Yes FC 3 - Read Holding Registers(4X)	1	0	10	No Change	0	0	Yes	20	
<input checked="" type="checkbox"/> Yes FC 3 - Read Holding Registers(4X)	2	0	10	No Change	0	10	Yes	21	

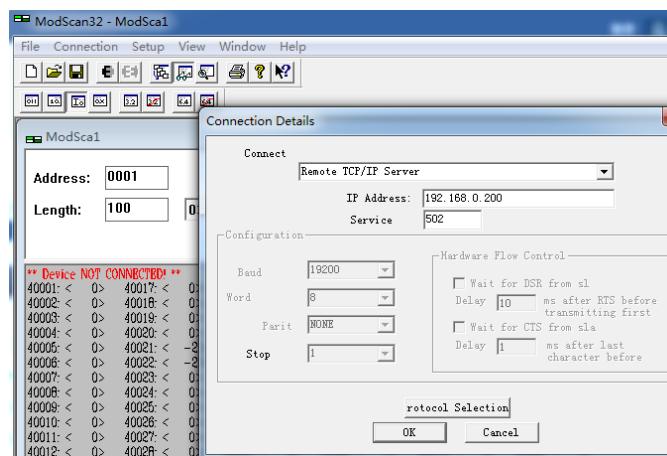
Below the table are three buttons: 'Add', 'Modify', and 'Delete'. At the bottom is a large blue 'Save' button.

上图中建立两条指令：

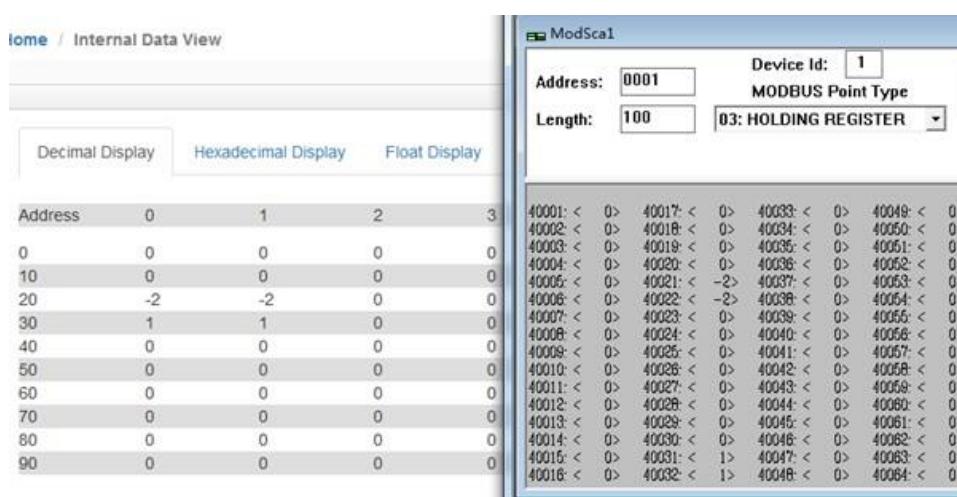
- 1- 读1号从站的40001-40010到内部寄存器0-9，错误状态放在了内部寄存器20。
- 2- 读2号从站的40001-40010到内部寄存器10-19，错误状态放在了内部寄存器21
- 3- 使能命令触发地址，在模块Modbus主站端口配置页面中，Command Trigger Address设置成30，如下图，表示使用模块内部起始地址为30的寄存器作为触发条件。然后保存，重启生效。



之后使用 ModScan 仿真作为 Modbus 从站，用 ModScan 的 40031 和 40032 可以模拟控制这两条指令的触发状态。



ModScan 的 40031 和 40032 设置为 1，可以看到模块内部寄存器地址 30 数据是 1，内部寄存器地址 31 也是 1，表示以上两条指令处于触发情况。模块内部寄存器地址 20-21 数据是-2，表示有错误代码，说明以上两条指令都没有正确执行。



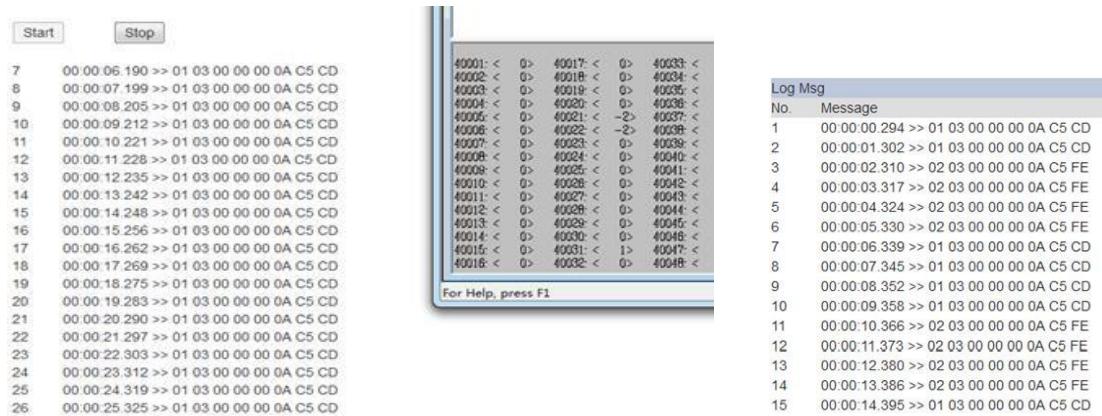
直观表现如下：检查 Port1 报文发送情况，显示第一条命令发送 4 次，第二条命令发送 4 次。因为命令本身发送 1 次，如果找不到从站设备，该命令会重新发送 3 次，共计 4 次。

重发次数，可以在端口配置中进行修改（如下图）：



以上两条指令都没有正确执行，如果是 2 号从站有问题，我们可以把 2 号从站的命令停止发送。

需要修改 ModScan 中 40032 的数值，从 1 改成 0（如下图），这样相当于停止了触发读取 2 号从站的指令。

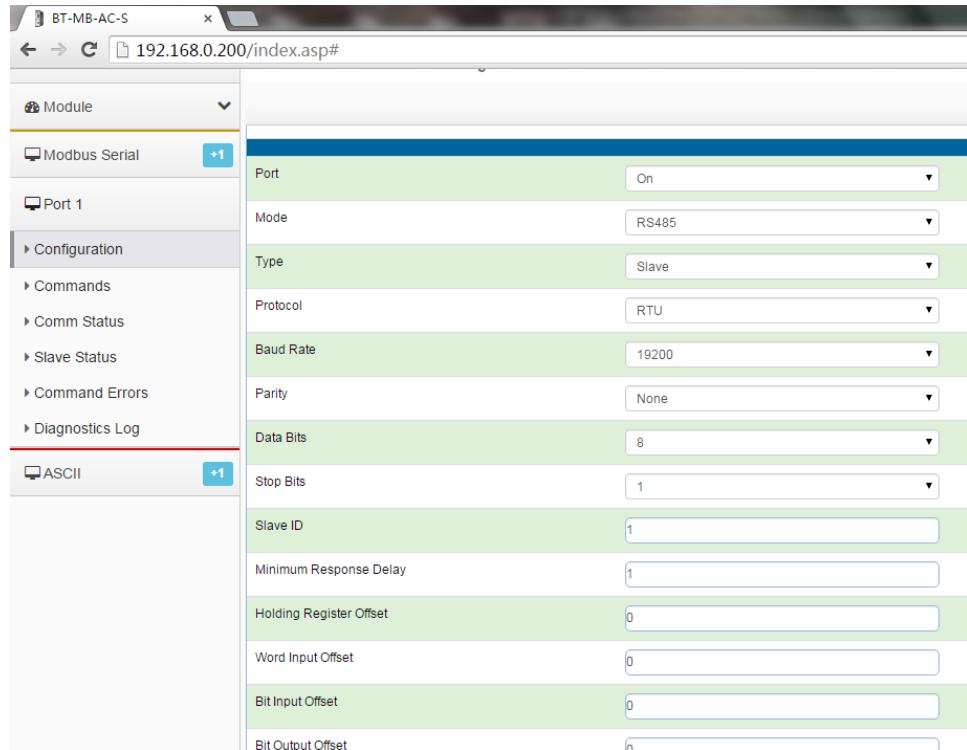


就实现了模块只读取 1 号从站数值的报文。避免了多个从站中有一个或两个掉线而影响整体 Modbus RTU 网络变慢的情况。

该功能建议配合前文提到的命令反馈功能一并使用，当其中一个命令返馈回来出现非 0 值，PLC 的程序可以自动关联这个触发值去停止 Modbus 指令的执行。

配置模块做 MODBUS RTU 从站

点击 Port1 里面的 Configuration 显示 S1 端口配置的页面：



注意事项：S1 作为 Modbus 从站，只需要配置端口参数，无需配置端口命令。S1 和 S2 共用模块内部数据区。

如下图为设置模块的 Modbus 从站端口参数：

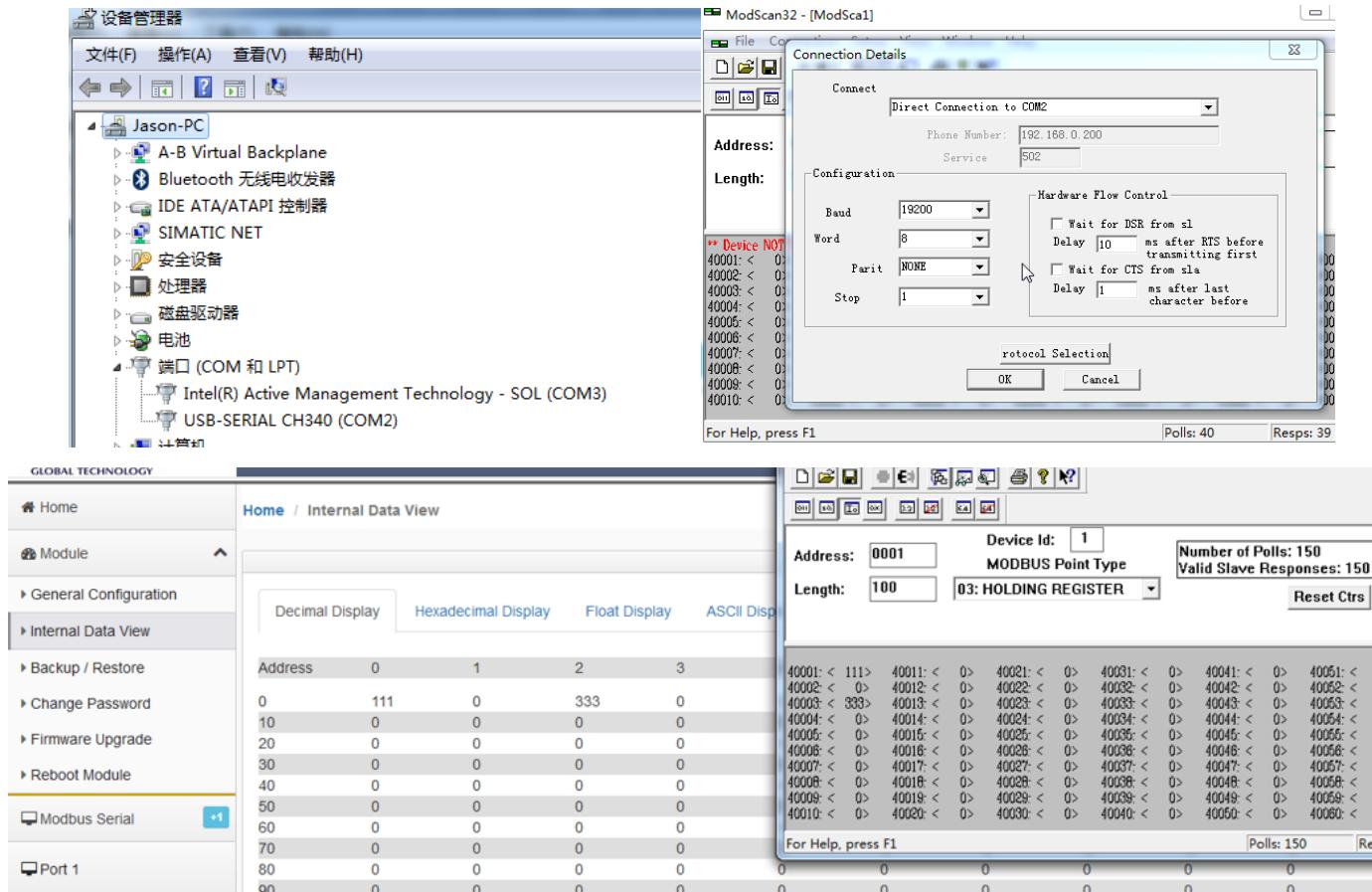
Port	On	端口使能
Mode	RS485	接线方式
Type	Slave	端口主站/从站
Protocol	RTU	端口协议
Baud Rate	19200	端口波特率
Parity	None	奇偶效验位
Data Bits	8	数据位
Stop Bits	1	停止位
Slave ID	1	从站地址
Minimum Response Delay	1	最小响应延时
Holding Register Offset	0	数据偏移
Word Input Offset	0	字输入偏移
Bit Input Offset	0	位输入偏移
Bit Output Offset	0	位输出偏移
<input type="button" value="Save"/>		

模块做 Modbus 从站时，需要搞清楚内部寄存器与 Modbus 数据对应关系：

模块内部寄存器 地址	等于	Modbus4区 地址	等于	Modbus3区 地址	等于	Modbus1区 地址	等于	Modbus1区 地址	等于	Modbus0区 地址	等于	Modbus0区 地址
0	=	40001	=	30001	=	10001	至	10016	=	00001	至	00016
1	=	40002	=	30002	=	10017	至	10032	=	00017	至	00032
10	=	40011	=	30011	=	10161	至	10176	=	00161	至	00176
11	=	40012	=	30012	=	10177	至	10192	=	00177	至	00192
20	=	40021	=	30021	=	10321	至	10336	=	00321	至	00336
30	=	40031	=	30031	=	10481	至	10496	=	00481	至	00496
99	=	40100	=	30100	=	11585	至	11600	=	01585	至	01600
100	=	40101	=	30101	=	11601	至	11616	=	01601	至	01616
220	=	40221	=	30221	=	13521	至	13536	=	03521	至	03536
1000	=	41001	=	31001	=	26001	至	26016	=	16001	至	16016
1001	=	41002	=	31002	=	26017	至	26032	=	16017	至	16032
1999	=	42000	=	32000	=	41985	至	42000	=	31985	至	32000
2000	=	42001	=	32001	=	42001	至	42016	=	32001	至	32016
2001	=	42002	=	32002	=	42017	至	42032	=	32017	至	32032
3000	=	43001	=	33001	=	58001	至	58016	=	48001	至	48016

打开 MODBUS RTU 仿真软件 MODSCAN32，其作用是仿真 MODBUS RTU 主站。软件连接作为 MODBUS RTU 从站的模

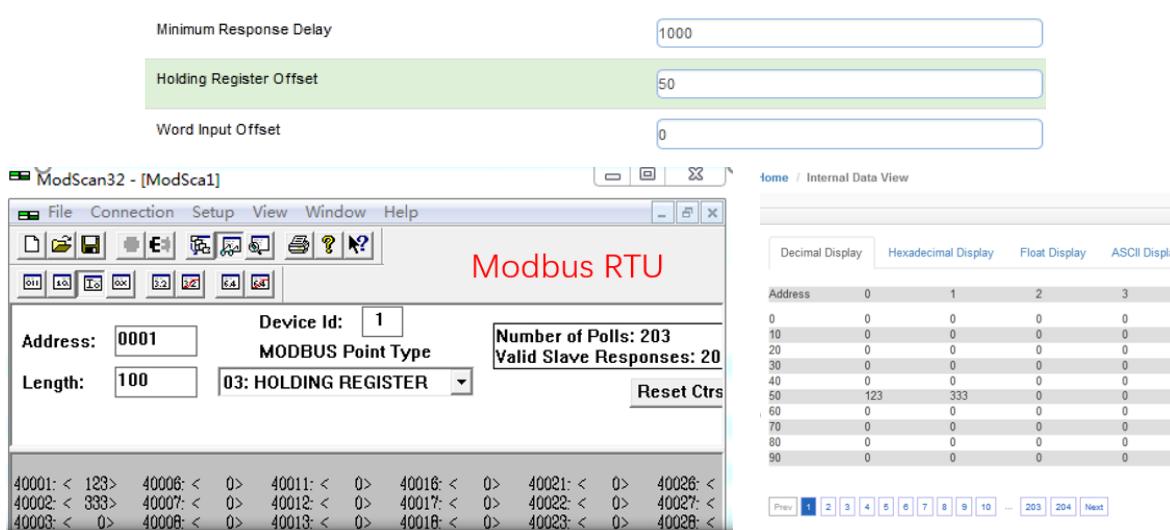
块。选择 Connection，选择电脑的 USB-485 接口 COM2，修改波特率，数据位，奇偶校验位，停止位等参数与模块的 Modbus 端口参数一致。点击 OK，可以看到连接的发送和接收次数。在 40001 等数据区进行写数据，可以看到模块内部对应寄存器同样有数据显示。



备注：40001 对应着内部寄存器 0，40100 对应着内部寄存器 99，以此类推。

Modbus RTU 配置成从站时，在主页面，可以设置接收地址偏移。

Holding Register Offset 使用方法：Modbus RTU 主站使用 FC3 功能码，在 40001 和 40002 输入两个数据，正常情况下，这两个数据应该会被写入到模块内部寄存器 0-1 当中去。如果此处偏移量设置成 50 (如下图)，则数据会直接偏移写入模块内部寄存器 50-51 里面。4 区，3 区，1 区，0 区同样遵循这个原理。



Word Input Offset 使用方法：如果此处偏移量设置成 50(如下图)，Modbus RTU 主站一侧在 3 区对 30001 和 30002 输入两个数据，数据会直接向后偏移放到模块内部寄存器 50-51 里面，ModScan32 仿真软件不能载入 3 区的数据，请以现场设备实际数据区域来填写。

Minimum Response Delay	1000
Holding Register Offset	0
Word Input Offset	50

Modbus RTU 诊断方式

查看主站端口命令是否有错误，发包和收包状态：

The screenshot shows the 'Status' section of the Modbus Port 1 configuration. On the left, a sidebar lists 'Module', 'Modbus Serial +2', 'Port 1', 'Configuration', 'Commands', **Comm Status**, 'Slave Status', 'Command Errors', and 'Diagnostics Log'. The 'Comm Status' item is selected. The main area displays a table of parameters:

Parameter Name	Value
Inter-message Delay	1.822
Command Count	1
Last Error Code	0
Number of Command Errors	0
Number of Requests	1232
Number of Responses	1232
Number of Errors Received	0
Number of Errors Sent	0

Buttons at the bottom right include 'Reset Counter' and 'Auto Refresh [2 Second(s)]'.

可视化查看从站状态点击SlaveStatus可以看到1号从站是绿色的。

The screenshot shows the 'Slave Status' section of the Modbus Port 1 configuration. The sidebar shows 'Module', 'Modbus Serial +2', 'Port 1', 'Configuration', 'Commands', 'Comm Status', **Slave Status**, 'Command Errors', and 'Diagnostics Log'. The 'Slave Status' item is selected. The main area displays a grid of 16 slave status indicators, where the first one is highlighted in green. Navigation buttons 'Prev' and 'Next' are at the bottom, along with an 'Auto Refresh [2 Second(s)]' button.

查看命令行是否有错误产生点击：

The screenshot shows the 'Command Errors' section of the Modbus Port 1 configuration. The sidebar shows 'Module', 'Modbus Serial +2', 'Port 1', 'Configuration', 'Commands', 'Comm Status', 'Slave Status', 'Command Errors', and 'Diagnostics Log'. The 'Command Errors' item is selected. The main area shows a table with columns for error type and count. Buttons at the bottom include 'Decimal Display' and 'Hexadecimal Display'.

Type	Count
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0

报文诊断功能：点击DiagnosticsLog，再点击Start端口发送和接收报文的情况。

>>符号是S1端口发送的报文，<<符号是S1端口接收的报文。

Line	Timestamp	Message
638	00:00:08.719	<< 01 03 14 22 B8 5F 3F
639	00:00:08.723	>> 01 03 03 E8 00 0A 45 BD
640	00:00:08.746	<< 01 03 14 22 B8 5F 3F
641	00:00:08.750	>> 01 03 03 E8 00 0A 45 BD
642	00:00:08.774	<< 01 03 14 22 B8 5F 3F
643	00:00:08.778	>> 01 03 03 E8 00 0A 45 BD
644	00:00:08.802	<< 01 03 14 22 B8 5F 3F
645	00:00:08.806	>> 01 03 03 E8 00 0A 45 BD
646	00:00:08.830	<< 01 03 14 22 B8 5F 3F
647	00:00:08.835	>> 01 03 03 E8 00 0A 45 BD
648	00:00:08.859	<< 01 03 14 22 B8 5F 3F
649	00:00:08.864	>> 01 03 03 E8 00 0A 45 BD
650	00:00:08.888	<< 01 03 14 22 B8 5F 3F
651	00:00:08.892	>> 01 03 03 E8 00 0A 45 BD
652	00:00:08.916	<< 01 03 14 22 B8 5F 3F
653	00:00:08.921	>> 01 03 03 E8 00 0A 45 BD
654	00:00:08.944	<< 01 03 14 22 B8 5F 3F
655	00:00:08.949	>> 01 03 03 E8 00 0A 45 BD
656	00:00:08.973	<< 01 03 14 22 B8 5F 3F
657	00:00:08.976	>> 01 03 03 E8 00 0A 45 BD

配置模块 ASCII 码协议端口参数

在模块主配置页面左侧，导航栏中点击 ASCII--Port2--configuration，设置 ASCII 相关参数，如下图：

Setting	Value
Mode	RS485
Baud Rate	19200
Parity	None
Data Bits	8
Stop Bits	1
Rx DB Location	0
Tx DB location	1000
Rx Termination Type	0
Rx Termination Count	2
Rx Termination Chars	12
Rx Timeout	200
Rx Delay	10
Rx Packet Length	20

Enabled: 表示是否启用端口；

Mode: RS232或者RS485接口选择；

Baud Rate: 与ASCII设备通讯的波特率；

Parity: 效验位；

Data Bits: 数据位；

Stop Bits: 停止位；

Rx DB Location: 表示模块端口接收 ASCII 码字符的内部寄存器起始地址；

Tx DB location: 表示模块端口发送 ASCII 码字符的内部寄存器起始地址；

Rx Termination Type: 0 to 15, 接收终止类型选项, 模块的串行端口缓存(255个字节)接收到ASCII字符串之后, 端口需要向模块处理器传输这些数据。可以设置条件暂停端口继续接收ASCII字符串, 然后端口缓存将已接收到的字符串打包向处理器进行数据的传输;

变量含义: Bit 0 = 无终止条件限制, 如果选择0, 表示端口读取到外部设备发来的字符串后, 立即向处理器进行传输, 一般默认使用该选项;

Bit1=超时终止, Bit2=字符间超时Bit3=数据包超过长度终止;

Rx Termination Chars: 可以设定终止字符, 端口接收一段连续字符串时, 当遇到该终止字符时, 则此次接收终止, 端口缓存将把已经接收到的字符串向处理器进行数据的传输。随后的字符串, 将会在下一个数据包中传输给处理器。注意: 可通过下一个参数, 设定第几次接收到终止字符的时候, 端口才结束此次的接收, 打包数据向处理器传输;

Rx Termination Count: 0 to 12, 这个设定值(接收到终止字符的最大累计值)有效的前提是Rx Termination Type 不等于0。当设定了终止字符之后, 需要设定该参数, 端口接收一段字符串, 其中接收到的终止字符的累积次数, 超过了该设定值, 则此次接收终止, 端口缓存将把已经接收到的字符串向处理器进行数据的传输。随后的字符串, 将会在下一个数据包中传输给处理器;

Rx Timeout: 0 to 65535, 如果在接收终止类型参数中设置了Bit1(超时终止), 则使用此参数。参数设置值表示在端口接收到第一个字符后开始计时, 在设定的时间到达后, 此次接收终止, 端口缓存将把接收到的字符串向处理器进行数据的传输。随后的字符串, 将会在下一个数据包中传输给处理器;

Rx Delay: 0 to 65535, 如果在接收终止类型参数中设置了Bit2(字符间超时终止), 则使用此参数。参数设置值表示在模块端口接收两个字符之间可等待的间隔时长, 如果端口接收两个字符的间隔, 超过了该设定值, 则此次接收终止, 端口缓存将把已经接收到的字符串向处理器进行数据的传输。随后的字符串, 将会在下一个数据包中传输给处理器;

Rx Packet Length: 0 to 200, 如果在接收终止类型参数中设置了Bit3(数据包超过长度终止), 则使用此参数。参数设置值代表, 模块端口每次接收到的字符串的长度, 超过该设定长度则此次接收终止, 端口缓存将把已经接收到的字符串向处理器进行数据的传输。随后的字符串, 将会在下一个数据包中传输给处理器;

Swap Rx Data Bytes: Yes or No, 这个参数决定了模块接收到的数据是否进行字节顺序的交换。YES为交换, NO为不交换。如果参数设置为Yes, 则接收到数据包的奇偶字节将会进行交换;

Tx Timeout: 0 to 65535, 此参数决定了每次发送数据的超时时间。端口发送的一个字符串数据包, 必须在设定的时间内发送出去。如果超时, 此次发送数据将被终止。其余的字符串将会在下一个数据包中发出;

Tx Minimum Delay: 0 to 65535, 此参数决定了每次发送字符串数据包之间的延迟间隔, 在结束了一次数据包发送之后, 经过固定的间隔后, 模块端口再次向外部发送数据。在与某些慢速的串口设备通讯时, 需要考虑设定该参数;

Swap Tx Data Bytes: Yes or No, 这个参数决定了模块发送的数据是否进行字节顺序的交换。YES为交换,

NO为不交换。如果参数设置为Yes，则发送数据包的奇偶字节将会进行交换。

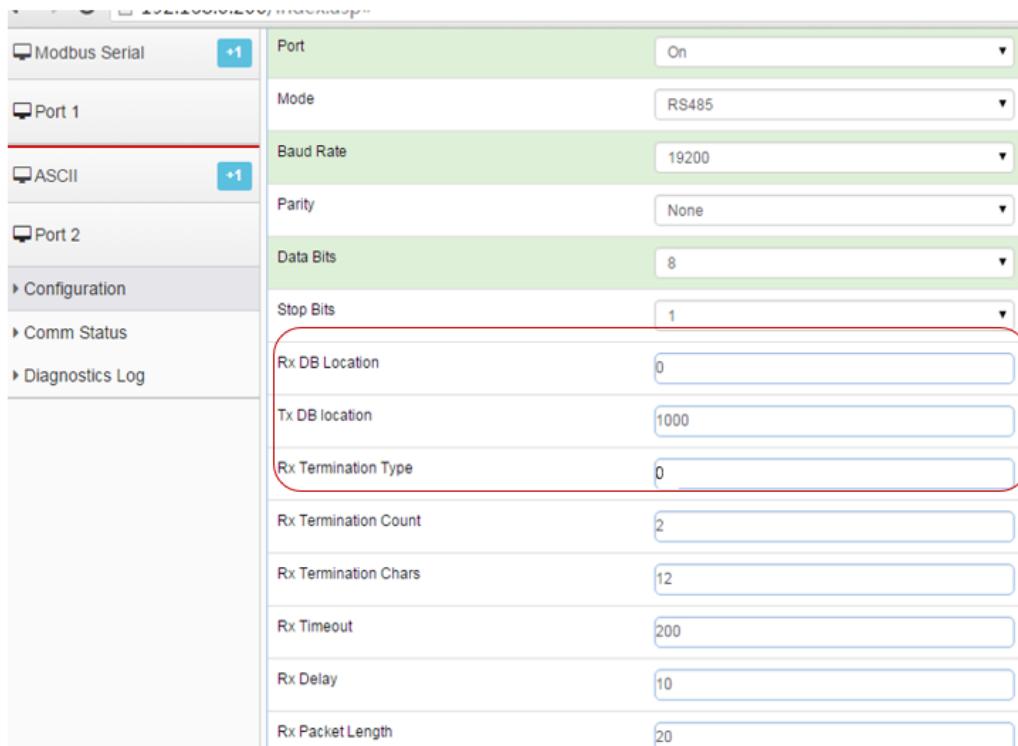
举例 1. Modbus 设备和 ASCII 设备交换数据

本案例中模块的 Modbus 端口配置成为从站，ASCII 端口可读可写。

如下图：配置 ASCII 码端口，串口 2、波特率、效验位、停止位、后点击保存，可以看到 Rx Termination Type 是 0 表示无终止条件限制，外部设备字符传输到端口，就会立即由端口传输到处理器，接收到的数据保存到模块内部寄存器起始地址为 0 的区域，另外 PORT2 端口发送的数据，设置来自于内部寄存器起始地址为 1000 的区域。

接线方式默认 RS485 接线方式，对应的接头，端子 6==+/485A/TR+，端子 1==-/485B/TR-。

可以选择 RS232 接线方式，对应的接头，端子 2 端子 3 端子 5 或者直接使用梯型串口头。



Port2端口连接电脑串口助手。发送数据，例如BEACON公司网站<http://www.beaconglobaltech.com>。



从模块数据内部数据区，查看收到的数据。

采用十进制显示：

0-24 寄存器（前 25 个寄存器）有数值，其中 0-8 寄存器（前 9 个寄存器）是状态信息。

内部寄存器 0 表示接收数据的次数。

内部寄存器 2 表示接收数据的长度。

内部寄存器 4 表示接收数据的总长度。

内部寄存器 5 表示接收数据的报文次数。

The screenshot shows the 'Internal Data View' section of the configuration interface. On the left, there's a sidebar with various module configuration options. The 'Modbus Serial' section is currently selected and has a count of '+1'. The main area displays a table of data from address 0 to 90. The columns are labeled 0 through 9. Addresses 0-8 represent status information, while 9-24 represent data. Below the table is a navigation bar with buttons for 'Prev', page numbers (1-10, 21-22), and 'Next'.

Address	0	1	2	3	4	5	6	7	8	9
0	478	0	31	1	15267	478	0	0	0	29800
10	28788	12090	30511	30583	25134	24933	28515	26478	28524	24930
20	29804	25445	11880	28515	109	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0

采用 ASCII 码显示：0-48 位（前 49 位）有数值，因为内部寄存器为 16 位字寄存器，2 个 ASCII 码占用 1 个内部寄存器地址。所以其中 0-17 位（前 18 位）是状态信息。18-48 位显示 <http://www.beaconglobaltech.com>。

This screenshot shows the same interface but with 'ASCII Display' selected. The data table now contains ASCII characters instead of decimal values. The first 18 columns (addresses 0-17) show mostly empty or placeholder characters like 'OE', 'p', and 'h'. The remaining columns (18-48) show the URL 'http://www.beaconglobaltech.com' repeated across all rows. The navigation bar at the bottom remains the same.

Address	0	1	2	3	4	5	6	7	8	9
0	OE								μ	P
10	OE								h	t
20	t	p	:	/	/	w	w	.	b	a
30	e	a	c	o	n	g	i	o	m	♦
40	l	t	e	c	h	.	c	o		
50	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
60	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
70	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
80	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
90	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
100	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
110	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
120	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
130	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
140	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
150	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
160	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
170	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
180	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦
190	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦

AC 端口接收数据状态检查。

This screenshot shows the 'Modbus Serial - ASCII Serial' monitoring interface. The left sidebar has 'Module' selected, with 'Modbus Serial' and 'ASCII' both highlighted in red. The main area displays a table of parameters for 'Port 2'. The table includes fields like 'Enabled' (1), 'Error Word' (0), and various counters for received and transmitted data. At the bottom right, there are buttons for 'Reset Counter' and 'Auto Refresh' (set to 2 seconds).

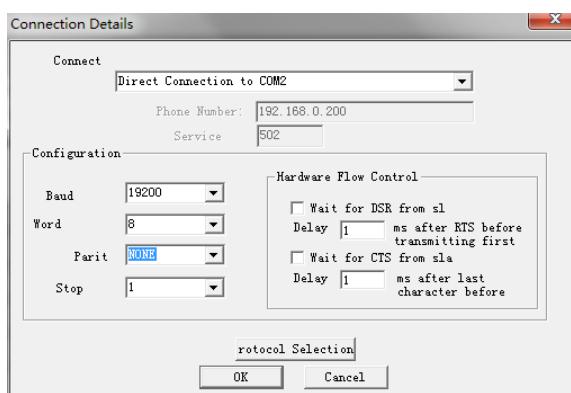
Parameter Name	Value
Enabled	1
Error Word	0
Last Error Code	0
Received Data Len	32
Receive State	1
Characters Received	25024
Messages Received	2830
Transmit Len	0
Transmit State	0
Characters Tx	0
Messages Tx	0

AC 端口接收数据报文检查。

No.	Message
1	00:00:00:247 << 68 74 74 70 3A 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D
2	00:00:01:261 << 68 74 74 70 3A 2F 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D
3	00:00:02:275 << 68 74 74 70 3A 2F 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D
4	00:00:03:288 << 68 74 74 70 3A 2F 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D
5	00:00:04:302 << 68 74 74 70 3A 2F 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D
6	00:00:05:316 << 68 74 74 70 3A 2F 2F 77 77 77 2E 62 65 61 63 6F 6E 67 6C 6F 62 61 6C 74 65 63 68 2E 63 6F 6D

点击 Port1-Configuration, 配置模块作为 Modbus RTU 从站, 初始画面如下:

Port1-连接 ModScan32:



使用ModSIM32仿真作为Modbus RTU主站, 使用FC3读取模块内部数据。前文介绍了, 模块做Modbus RTU从站时, 内部数据区起始地址0-24对应了40001-40025, 所以下图中我们可以看到Modbus RTU主站读取到了刚才模块

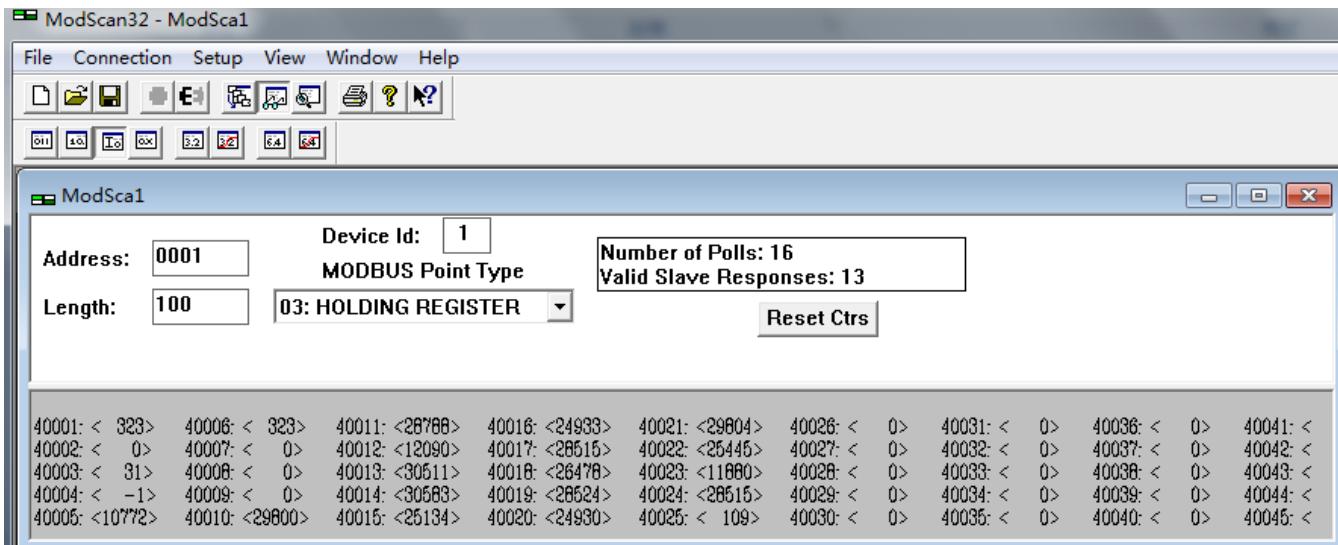
从串口助手（ASCII设备）中接收到的数据http://www.beaconglobaltech.com。

其中模块中0-24地址的寄存器中有数据显示，则ModSIM32中40001-40025也同样接收到了数据。

注：如下截图中模块内部数据区和ModSIM32，前9位状态寄存器数据不同。是因为两次截图时，两次接收数据的次数，接收数据的总长度，接收数据的报文次数均不同。

Address	0	1	2	3	4	5	6	7	8	9
0	478	0	31	1	15267	478	0	0	0	29800
10	28788	12090	30511	30583	25134	24933	28515	26478	28524	24930
20	29804	25445	11880	28515	109	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0

上图为模块内部数据寄存器截图，下图为 ModScan32 从模块采集到的数据。



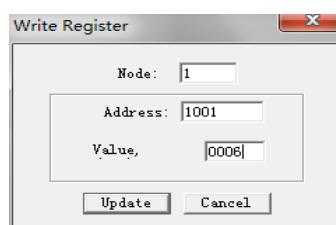
ModScan32（模拟 Modbus 主站）写数据给串口助手（模拟 ASCII 设备）。因为之前 AC 端口配置模块对外发送数据的内部寄存器起始地址是 1000 对应于 Modbus 主站的 41001，所以 ModScan32 需要从 41001 开始填写数据。

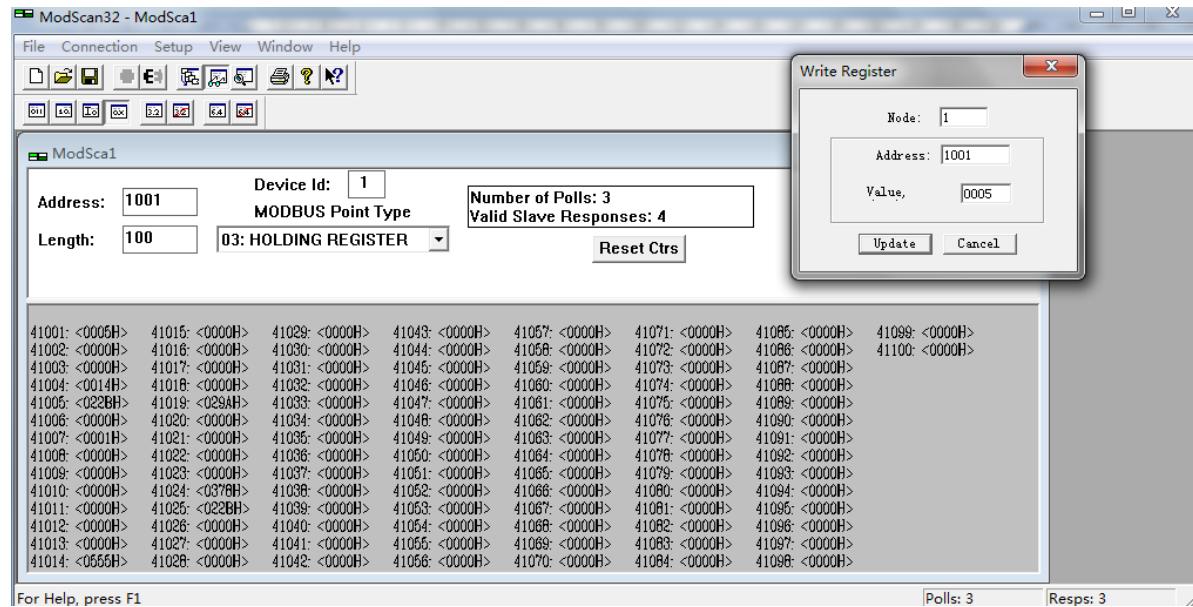
Rx DB Location	<input type="text" value="0"/>	ASCII码接受数据起始地址
Tx DB location	<input type="text" value="1000"/>	ASCII码发送数据起始地址

41001-时间轮询点，例如从 1 改成 2，触发写一次，或者从 2 改成 5，触发写一次。

41004-写数据的数量。例如 20 个 ASCII 码十进制的 20，用 16 进制表示就是 14。

41005-41014 这 10 个 INT 的数据就发送出去了，表示的还是 20 个 ASCII 码。

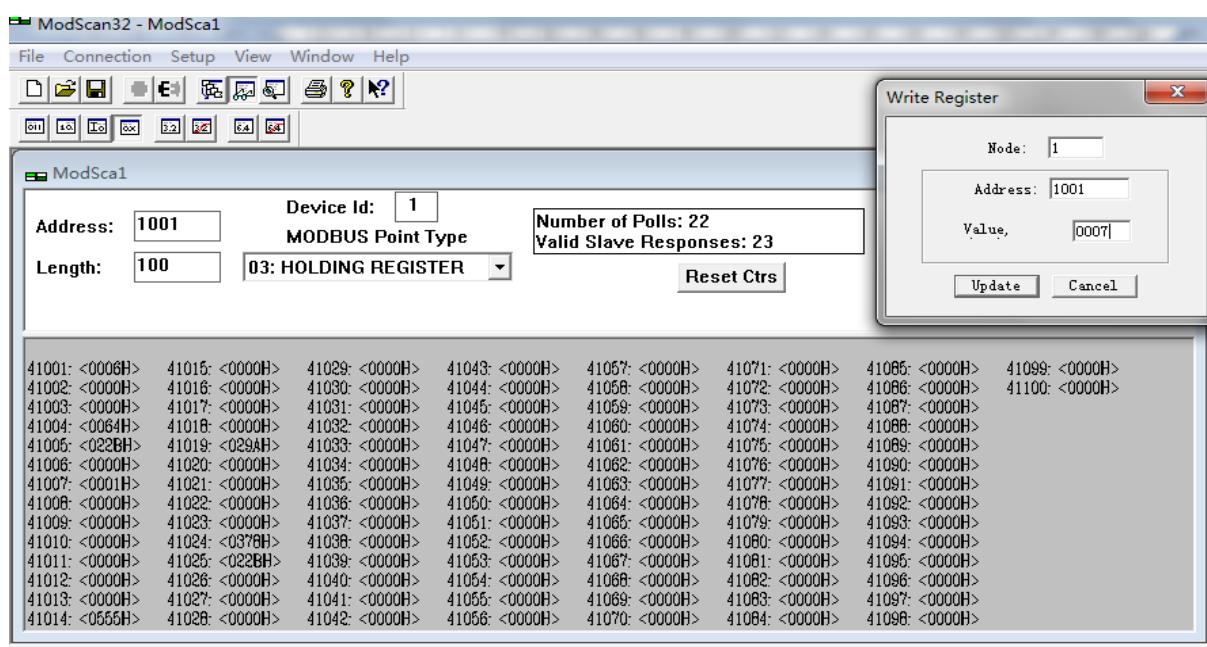




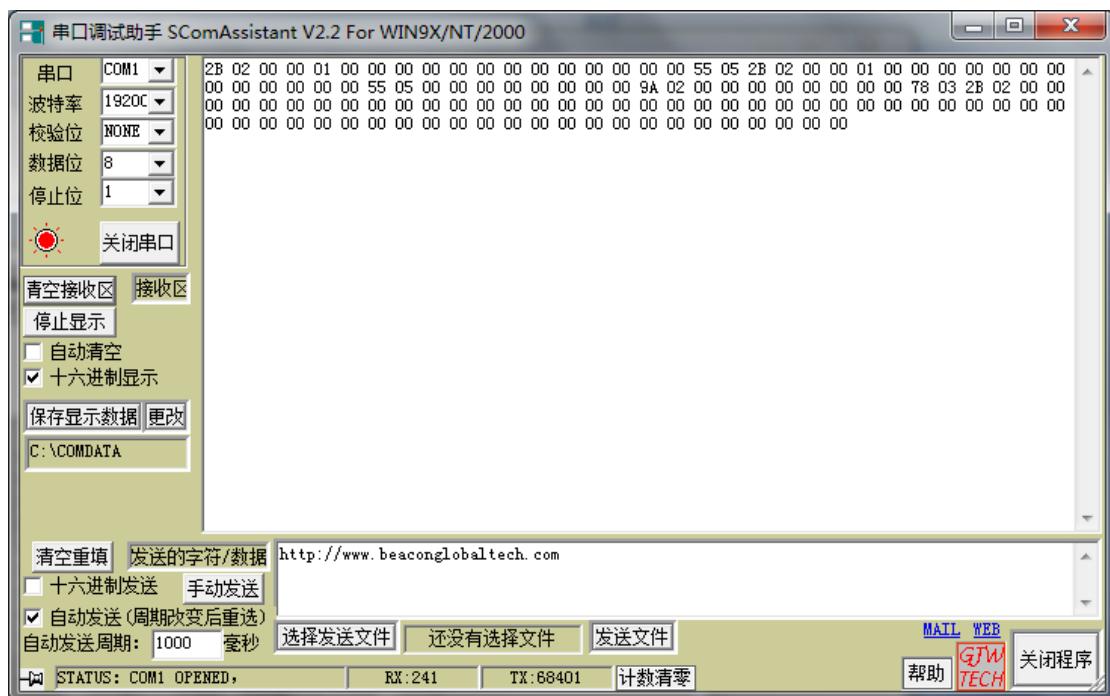
下图可以看到 ModScan32（模拟 Modbus 主站）写入模块的 20 个 ASCII 数据，通过模块发送到串口助手（模拟 ASCII 设备）。



ModScan32（模拟 Modbus 主站）增加写数据的数量，修改 41004，写 100 个 ASCII 码，十六进制显示，改成 64H，也就是十进制的 100。然后写给 41001，触发写操作。



下图可以看到 ModScan32（模拟 Modbus 主站）写入模块的 100 个 ASCII 数据，通过模块发送到串口助手（模拟 ASCII 设备）。



联系我们

如果在使用过程中有更多的问题，可以通过以下方式联系我们获得支持。

客户服务热线 (中国大陆)	13910136425
技术支持	support@beacongt.com
亚太区销售	asia@beacongt.com
北美区销售	usa@beacongt.com
微信公众平台	
网址	http://www.beaconglobaltech.com