BT-EN-MB2-B 快速启动手册

BEACON GLOBAL TECHNOLOGY

日 录

BT-EN-MB2 简介	2
模块初始配置	2
配置模块做 EtherNet/IP Server:	3
配置模块做 EtherNet/IP Client	9
配置模块做 Modbus RTU Master(主站)	15
Modbus 命令使能控制介绍	22
配置模块做 Modbus RTU Slave(从站)	24
Modbus RTU 诊断方式	27
举例 1. Modbus 从站设备和罗克韦尔 PLC 交换整型数数据	28
举例 2. Modbus 从站设备和罗克韦尔 PLC 交换布尔量数据	32
举例 3. Modbus 主站和罗克韦尔 PLC 交换数据	34
其他版本简介	39
联系我们	40

BT-EN-MB2 简介

BT-EN-MB2系列分为基础型(-B)和领先型(-P),本手册中以基础型BT-EN-MB2-B进行举例。简介:该模块具有1 个100M/10M的以太网端口(E1),E1可以用于作为5个EtherNet/IP协议的Client和Server,具有两个串行通讯接口 S1和S2(RS-232/485/422),这两个串行接口都可以用于作为Modbus协议的Master或者Slave。

模块初始配置

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。

BT系列模块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)对于模块进行配置: IE10, GOOGLE Chrome 35, FIREFOX 35, Safari 7 及以上的版本。

把本地电脑的IP地址与所连接的模块E1端口配置成相同的IP网段,例如本案例中本地电脑配置成和E1在同一网 段下的,然后在GOOGLE Chrome浏览器的地址框里面输入E端口IP地址,点击回车键后,进入到BT-EN-MB2-B模块的 配置页,如下图:

-		EtherNet/IP - Modbus Serial	
	# Home	Home / Module Status	
	& Module ✓		
	☐ Modbus Serial →2	Module Name	BT-EN-MB2-B
-	EtherNet/IP Server	E1: IP Address	192.168.0.200
	EtherNet/IP Client	E1: MAC Address	00:1E:94:02:81:86
-		Product Base Version	1.00.181
		Product Version	1.01.044
1. 在 用户名	配置页面的右上角,点 (Username):admin	击 ^{Login} ▲ ,输入用户名和密码进入模块西	己置。
密码(P	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。	
2. 在	配置页面的左侧导航条	内,点击Module,将打开如图所示的菜单。点击 pme / Backup And Restore	Backup/Restore。

W Home	Home / Backup And Restore
B Module	Lipload configuration file to client
General Configuration	
Internal Data View	Export Config
Backup / Restore	
Change Password	Download configuration file to Module
Firmware Upgrade	
Set Date & Time	洗柽文件 未洗择任何文件
Reboot Module	The second

3. 登录后看到导出配置文件

和恢复配置文件

Ethernet Port 1	
IP Address	192.168.0.200
Subnet Mask	255.255.255.0
Default Gateway	192.168.0.1
Ethernet Port 2	
IP Address	192.168.1.200
Subnet Mask	255.255.255.0
Default Gateway	192.168.1.1
	Save

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

# Home	Home / Change Password
Module	
General Configuration Internal Data View Backup / Restore	User Name: admin Current Password
Change Password Firmware Upgrade Reboot Module	New Password
➡ Modbus Serial +2	
EtherNet/IP Server	Save

- 5. 点击 Change Password ,可以修改模块的登录密码。
- 6. 点击* Set Date & Time 可以设置模块的日期和时间。

	2 Module	
	General Configuration Internal Data View Backup / Restore	Warning Proceed with caution. It's important to set the correct date and time for the modules Time Date functions. Please refer to the user manual for additional information. DateTime
	 Change Password 	
	Firmware Upgrade	mmiddlyyyy hhiliss 🗙 🗰
	Set Date & Time	
	Reboot Module	Set Date and Time
7. 点击	* Reboot Module	表示重启模块。
	B Module	
	General Configuration Internal Data View Backup / Restore Change Password Firmware Ungrade	Warning The module has to be rebooted due to any configuration changes. Note that the data communication will be temporarily interrupted if reboot. OK to reboot the module now?
	 ▶ Set Date & Time 	OK

配置模块做 EtherNet/IP Server:

通过浏览器,登录模块主页面

BT-EN-MB2 快速启动手册

如下图。

			EtherNet/IP - Modbus Serial	
	🖀 Home		Home / Module Status	
	🚳 Module	*		
	Generation Modbus Serial	+2	Module Name	BT-EN-MB2-B
	EtherNet/IP Server	*	E1: IP Address	192.168.0.200
	EtherNet/IP Client	+5	E1: MAC Address	00:1E:94:02:81:86
			Product Base Version	1.00.181
			Product Version	1.01.044
在左侧菜单	♀ EtherNe 中,点击	et/IP Ser	er ^ , 查看 EtherNet/IP Server Co	onfiguration 的链接数

不同型号的模块的 EtherNet/IP Server Configuration链接数不同。可以看到当前模块有4组Class 1 Connections的链接,这4组Class 1 Connections的链接可以在LOGIX5000软件里进行配置全部采用或者根据需要部分采用,每组Class 1 Connections提供248个INT数据类型的输入和248个INT数据类型的输出。

BEACON GLOBAL TECHNOLOGY		EtherNet/IP - Mod	bus Serial			
🖨 Home		Home / EtherNet/IP Serve	er / Configuration			
🙆 Module	~					
Generation Modbus Serial	+2	EtherNet/IP Server Config	juration			
EtherNet/IP Server	^	Server File Size	100		7	
Configuration		0				
Comm Status		Save				
 Status of EtherNet/IP CI 1 Connections 	ass	Class 1 Connections				
EtherNet/IP Client	45	Input Data Address	Input Size	Output Data Address	Output Size	
		0	248	1000	248	
		250	248	1250	248	
		500	248	1500	248	
		0750	248	1750	248	
		Modify	240	1750	240	

上图可以看到,当前模块总共有2000个16位字的内部寄存器,模块做为EtherNet/IP Server时候,可以被多个 罗克韦尔PLC同时访问。

数据对应关系:

Input Data Address 表示罗克韦尔PLC采集模块数据(对PLC一侧为输入)的内部寄存器地址范围,0是指模块 内部第0个寄存器,输入起始地址为0,数量 248,表示模块对PLC的第一组输入数据,所占用的模块内部寄存器地址 范围。

Output Data Address 表示罗克韦尔PLC写给模块数据(对PLC一侧为输出)的内部寄存器地址范围,1000是指模块内部第1000个寄存器,输出起始地址为1000,数量 248,表示PLC对模块的第一组输出数据,所占用的模块内部寄存器地址范围。

此处248个输入寄存器的数量要与LOGIX5000里面的Class 1 Connections 对应。并且输入输出的起始位置和数 量可以任意更改。

注:模块默认做EtherNet/IP从站,不需要任何设置。

如下步骤为在Logix5000配置软件中添加模块。

将网关E1端口和电脑,以及Logix PLC以太网接口相连接。在电脑中使用RSLinx扫描模块,然后在RSLogix5000 中添加该模块的EDS文件,如下图:



选择从设备上传 EDS 文件,如下图





下一步通过添加"Generic EtherNet Bridge"完成 PLC 和模块的通讯,如下图:



Ent	er Search Text for Module T,	Den Filters		Hide Filters≎
		Module Type Categor	Module T	ype Vendor F
	Communication Communications Adapte Controller Digital	r V Allen-Bradle V Beacon Globa V Cognex Corpo	y 1 Technology, Inc. ration er	
•				•
•	' Catalog Number	Description	Vendor	Category
	Drivelogix5730 Eth E1 Plus	10/100 Mbps Ethernet Port on DriveLogix5730 Electronic Overload Relay Communications	Allen-Bradley Allen-Bradley	Communication Communication
	ETHERNET-BRIDGE	Generic EtherNet/IF CIP Bridge	Allen-Bradley	Communication
•	ETHERNET-MODULE ILX34-AENWG IND560 Ethernet/IP	Generic Ethernet Module 1734 Wireless Ethernet Adapter, Twisted-P Scale Terminal 111	Allen-Bradley Prosoft Tec Mettler-Toledo	Communication Communication Communication
E0	of 211 Module Types Fo	und		Add to Favori

设定模块的 IP 地址,该地址为 E1 端口地址

Enter S	New Module	Indule Type	C1	Pilt	Hida >	Rilters≈
	Type: Parent:	ETHERNET-BRIDGE ENBT	Generic EtherNet/	IP CIP Bridge		
	Name:	EN		Address / Host N	ame	
•	Description:		*	IP Address:	192 . 168 . 0 . 200	Þ
-			~	🔘 Host Name:		
Dr						n
E1 Et E1						n n n
EI						n
IN	🔽 Open Modu	ule Properties		ОК	Cancel Help	m T

在 Generic EtherNet Bridge 下添加一个新模块,再添加一个新的 CIP-Connection.

		6-A ENE [/A	4 3T E ENBT IDGE EN
		IJ	New Module
🖞 1756-ENBT/A ENBT	Bus Size 100	_	Discover Modules
🖃 🗊 ETHERNET-BRIDGE EN		E	Paste CtrI+V
CIP Bus		_	Print

Enter Search Text for Modu	de Type Clear Filters		Hide Filters
V	Module Type Categor	Module	Type Vendor F
0ther	Allen-Bra	adley	
Catalog Number	Description	Vendor	Category
CIP-HODVLE	Generic CIP Module	Allen-Bradley	Other

之后开始设定 PLC I/O connection 的参数,如下图:

请使用 Input 和 Output 都为 248 个字, Configuration 为 0。Comm format 需要选择 Data INT。

atalog Mo	vpe New Module	. <u>.</u>	_				×	
Enter See	Type: Parent: Name: Description: Comm Format: Address:	CIP-MODULE Generic CIP Module EN CIPO Data - INT 0	•	Connection Para Input: Output: Configuration: Status Input: Status Output:	ameters Assembly Instance: 1 2 4	Size: 248 248 0	 ▲ (16-bit) ▲ (16-bit) ▲ (16-bit) ▲ (8-bit) 	2
1 of 1 Mo	♥ Open Modul dule Types Fo	e Properties		ОК	Cano	el	Help Add to Favor	Þ
Close or	1 Create				Cr	eate	Close	Help

Assembly instances 设定方式: input 为 "1", output 为 "2", configuration 为"4"。

每一个 I/O connection 都需要进行如上的配置,之后点击 Create,来设定 I/O connection 的 RPI time 时间. 单机 PLC 结构, Use Unicast Connection over EtherNet/IP 要勾选, RPI 时间可以使用 5ms 或者 20ms。冗余 PLC 结构, Use Unicast Connection over EtherNet/IP 不要勾选, RPI 时间可以使用 20ms 或者 40ms。

Module Properties Report: EN (CIP-MODULE 1.1)]
General Connection Module Info	
Requested Packet Interval (RPI): 5.0 ms (0.2 - 750.0 ms) Inhibit Module Major Fault On Controller If Connection Fails While in Run Mode V Use Unicast Connection over EtherNet/IP	
Module Fault	
Status: Offline OK Cancel Apply Help	

以上步骤完成后, 在模块侧, 可以通过诊断来查看:

Modbus Serial +2	General Status	
EtherNet/IP Server		
	Parameter Name	Value
Configuration	Number of EtherNet/IP Class 1 Connections	0
▶ Comm Status		
Status of EthorNot/ID Class 1		Reset Counter
Connections		Auto Refresh 2 - Second(s)

前文已经提到过,数据对应关系如下,从 AB 的 PLC 对模块 internal data base 进行读写。

EN:0:I.Data[0]-	EN:0:I.Data[247]对应模块内部寄存器	0-247 的地址	输入
EN:0:0.Data[0]-	EN:0:0.Data[247]对应模块内部寄存器	1000-1247 的地址	输出
EN:1:I.Data[0]-	EN:1:I.Data[247]对应模块内部寄存器	250-497 的地址	输入
EN:1:0.Data[0]-	EN:1:0.Data[247]对应模块内部寄存器	1250-1497 的地址	输出

File Edit View Search Logic Communications Tools	Wind	low Help			
🖹 🎽 🖶 🏯 🎄 🖻 💼 🕫 ロロ msg	÷	# 4 % Fr V V Q Q Seec	t a Language 👻 🧕		
em Run I Env Mode 6 Forces F States Controller OK 5 Edits States I/O OK solundancy 5 of	AB_ETHI rites ,	HP-1\192.16.8.0.11\Backplane\0* ■ → H → H ≺ > - U> ↓ ↓ H → H ≺ > - U> ↓ ↓ Add-On ▲ Safety ▲ Alarms ▲ BE ▲ Timer/C			
Controller Organizer 🗸 🗜	×	Scope: DEN_EN - Show: All Tags		👻 🏹 Enter Nan	e Filer
- Carl Unscheduled Programs / Phases		Name ===	Value +	Force Mask + Style	Data Type
🖶 🔄 Motion Groups		E-EN11:0:0.Data	{}	{} Decimal	INT[248]
Ungrouped Axes		+ EN11:0:0.Data[0]	8888	Decimal	INT
🗀 Add-On Instructions		+ EN11:0:0.Data[1]	8888	Decimal	INT
🗄 🚔 Data Types		+ EN11:0:0.Data[2]	8888	Decimal	INT
		EN11:0:0.Data[3]	8888	Decimal	INT
👜 🔙 Strings		+ EN11:0:0.Data[4]	8888	Decimal	INT
Add-On-Defined		+ EN11:0:0.Data[5]	8888	Decimal	INT
Predefined		+ EN11:0:0.Data[6]	8888	Decimal	INT
Module-Defined		+ EN11:0:0.Data[7]	8888	Decimal	INT
Trends		+ EN11:0:0.Data[8]	8888	Decimal	INT
		EN11:0:0.Data[9]	8888	Decimal	INT
1756 Packalana 1756-04		+ EN11:0:0.Data[10]	8888	Decimal	INT
1756 Backplane, 1756 H4			0	Decimal	INT
	=	EN11:0:0.Data[12]	0	Decimal	INT
Bright Strengt		EN11:0:0.Data(13)	0	Decimal	INT
E and show shit		EN11:0:0.Data[14]	0	Decimal	INT
1/56-EN21 EN1		EN11:0:0.Data[15]	0	Decimal	INT
D ETHERNET-BRIDGE EN11		EN11:0:0.Data(16)	0	Decimal	INT
E- CIP Bus		+ EN11:0:0.Data[17]	0	Decimal	INT
0 CIP-MODULE EN000		EN11:0:0.Data[18]	0	Decimal	INT
1 CIP-MODULE EN111		EN11:0:0.Data(19)	0	Decimal	INT
2 CIP-MODULE EN222		+ EN11:0:0.Data[20]	0	Decimal	INT
💷 🖞 3 CIP-MODULE EN333	-	EN11:0:0.Data[21]	0	Decimal	INT
1		E EN11 0 0 D ++ (22)		Destinut	DIT

以此类推。如下图,在RSLogix5000第一个CIP I/0链接的输出标签的开头写一些数据。

网关Internal Data Base 从 地址1000开始的数据的变化。

ione / interna	ai Data view									
Decimal Dis	splay Hex	adecimal Displa	iy Float Di	splay ASCI	I Display					
Address	0	1	2	3	4	5	6	7	8	9
1000	8888	8888	8888	8888	8888	8888	8888	8888	8888	8888
1010	8888	0	0	0	0	0	0	0	0	0
1020	0	0	0	0	0	0	0	0	0	0
1030	0	0	0	0	0	0	0	0	0	0
1040	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0
1060	0	0	0	0	0	0	0	0	0	0
1070	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0
1090	0	0	0	0	0	0	0	0	0	0
Prev 1 2	6 7	8 9 10 1	1 12 13	14 15 1!	9 20 Next					

在RSLogix 5000第4个CIP I/0链接的输出标签的结尾写一些数据。

0 9 1	CL 5000 - EN MT (1755 162 20 12)							
Ella	SLOgix 5000 - EN_MIT [1750-L65 20.12]							
rite								
		•	••••••••••••••••••••••••••••••••••••••	ス Select a Langu	iage 🔻 💹			
Ren	Run 🚺 🗖 Run Mode 🛛 🙀 Path: AB_ETH	HIP-1	192.168.0.11\Backplane\0*	▼ 品				
No F	proes							
No B	dits 🔒 Battery OK 🛛 🖌 H Her Her		⊢ +/+ -()· -(U)· -(L)·					
Red	ndancy 5.0 Favorites	Ad	i-On 🖌 Safety 🖌 Alarms 🖌 Bit ,	Timer/C				
	Controller Organizer							
Se la	the Madula Defend		Controller Tags - EN_MT(contro	oller)				
art	Tranda	9	cope: 🎁 EN_MT 🛛 🗸 Sho	w: All Tags		• Y. É	nter Name Filter	
Pag		⊩	Name	-9[A	Value •	Force Mark	Stule	Data I
ē	1756 Backplane 1756-M	⊩	E-EN11:3:0 Data[226]		1 diuo 0	TOICE MOSK	Decimal	INT
	P 1730 Backplane, 1730-44		+ EN11:3:0 Data[227]		ů.		Decimal	INT
			E-EN11:3/0 Data[228]		0		Decimal	INT
			+ EN11:3:0 Data[229]		0		Decimal	INT
	Ethernet		+ EN11:3:0.Data[230]		0		Decimal	INT
	1/56-EN21 EN1		+ EN11:3/II Data[231]		0		Decimal	INT
	ETHERNET-BRIDGE EN11		+ EN11:3:0.Data[232]		0		Decimal	INT
			+ EN11:3:0.Data[233]		0		Decimal	INT
	E D CIP-MODULE EN000		+ EN11:3:0.Data[234]		0		Decimal	INT
	. 9 1 CIP-MODULE EN111		+ EN11:3:0.Data[235]		0		Decimal	INT
	2 CIP-MODULE EN222		+ EN11:3:0.Data[236]		0		Decimal	INT
	L- 1 3 CIP-MODULE EN333		+ EN11:3:0.Data[237]		0		Decimal	INT
	· · · · · · · · · · · · · · · · · · ·		+ EN11:3:0.Data[238]		0		Decimal	INT
	Module Defined Tags		+ EN11:3:0.Data[239]		0		Decimal	INT
			+ EN11:3:0.Data[240]		0		Decimal	INT
	EN11:3:C		± EN11:3:0.Data[241]		0		Decimal	INT
	Description		+ EN11:3:0.Data[242]		0		Decimal	INT
	Status Running		+ EN11:3:0.Data[243]		0		Decimal	INT
	Module Fault		+ EN11:3:0.Data[244]		0		Decimal	INT
			+ EN11:3:0.Data[245]		0		Decimal	INT
			+ EN11:3:0.Data[246]		9999		Decimal	INT
			+ EN11:3:0.Data[247]		9999		Decimal	INT
		a statement						

网关Internal Data Base 地址1996和1997的数据值的变化:

Home / Inter	nal Data Vie	w									
Decimal D)isplay H	exadecimal Display	Float	Display AS	CII Display						
Address	0	1	2	3	4	5	6	7	8	9	
1900	0	0	0	0	0	0	0	0	0	0	
1910	0	0	0	0	0	0	0	0	0	0	
1920	0	0	0	0	0	0	0	0	0	0	
1930	0	0	0	0	0	0	0	0	0	0	
1940	0	0	0	0	0	0	0	0	0	0	
1950	0	0	0	0	0	0	0	0	0	0	
1960	0	0	0	0	0	0	0	0	0	0	
1970	0	0	0	0	0	0	0	0	0	0	
1980	0	0	0	0	0	0	0	0	0	0	
1990	0	0	0	0	0	0	9999	9999	0	0	

Prev 1 2 ... 11 12 13 14 15 16 17 18 19 20 Next

配置模块做 EtherNet/IP Client

模块正常和 Logix 系列 PLC 通讯都是作为 server 从站,不过也可以同时支持作为 Client 和 Server 和 PLC 交换数据。在前一章介绍"模块做 Ethernet/IP server"的时候,很重要的一点是介绍了如何分配模块内部数据区的内容。如果模块同时作为 EtherNet/IP 的 Client 和 Server 则要特别注意,读写数据区冲突的问题,以免造成数据 混乱。

EthorNot/ID Client	Air Commands
	Enable Function IP SlotOty Poll Data Internal Data Tag Tag Word File File Element Sub File CisTosAttDesc
Client 1	Type Name Offset Addr Type Num Num Element String
▶ Configuration	Add Modify Delete
Commands	
Comm Status	Save list to Flash
Command Errors	

如上图, 点击 EtherNet/IP Client ----Client1----Commands。

点开Configuration, 查看默认的配置:

Message Type:

连接罗克韦尔1756系列,1769系列,1746系列,PLC-2系列,PLC-5系列,SLC500系列,Micrologix PLC系列, PowerFlex变频器系列,连接E300智能马达保护器,PowerMonitor智能电力监控仪等需要选择Connected。

Home / EtherNet/IP Client 1 / Configuration	
Message Type	Connected
Minimum Command Delay	50
Response Timeout	1000
Retry Count	3
	Save

此处用于连接 1756PLC,因此选择 Connected。

Unconnected Send

Minimum Command Delay: 每个 Client 执行指令的轮询时间,单位 ms, 范围 0-65535

注: 该时间越小,发送命令越快,但并非越小越好,需要先查看从站设备的说明书,确定从站响应时间是否能 及时接受和反馈,主站发送命令的间隔。

Response Timeout: 所连接设备的响应时间, 单位 ms,

Retry Count: 重新尝试连接次数,

范围 0-65535

范围 0-65535

之后选择指令的类型:

EtherNet/IP Server		
_	All Commands	•
EtherNet/IP Client +15	All Commands	. Poll Da
Client 1	Basic Controller Tag PLC5 ASCII	SlotQty Interval Sw
Configuration	PLC5 Binary	Delete
▶ Commands	Generic	
▶ Comm Status	Save list to Flash	
Command Errors		

Basic 命令用于罗克韦尔 PLC-5, ControlLogix 数据的读写;

Controller Tag 命令用于罗克韦尔 CompactLogix, ControlLogix 数据标签或标签数组的读写

PLC5 ASCII 命令用于罗克韦尔 PLC-5, ControlLogix 数据的读写;

PLC5 Binary 命令用于罗克韦尔 PLC-5, ControlLogix 数据的读写;

SLC500 命令用于罗克韦尔 SLC500, MicroLogix, PowerFlex 变频器数据的读写;

Generic 命令用于罗克韦尔 PowerFlex 变频器, E300 智能马达保护器, PowerMonitor 智能电力监控仪数据的读

写。

选择要连接的种类,选择相应的命令。点击 Add 可以增加命令行。

Controller Tag	•	
Enable	Yes	▼ 使能、禁止、内部寄存器有变化后写
Function Type	CIP Data Table Write	 → 功能码读或写
IP Address	1.1.1.1	要连接设备的IP地址
Slot	0	Micro800填-1,其他PLC根据RSLINX读取CPU实际位置填出
Quantity	[1	读或者写的数据的数量
Poll Interval	0	指令轮询时间
Data Swap	No Change	 数据高低位交换,字交换,字节交换,字和字节交换
Internal Data Address	0	模块内部寄存器,存放数据的起始地址。
Data Type	INT	▼数据类型,读命令INT无效,命令会跟随读取标签的类型
Tag Name	SampleTag	标签名,可以是单个标签或数组标签直接填写数组总名
Tag Offset	0	数组标签偏移地址
Cmd Errors Mapping Enabled	No	 → 命令错误状态位反馈开启
Cmd Errors Mapping Address	0	命令错误状态位反馈地址,填写模块内部寄存器任意位
Desc		命令描述

以下按照和 1756 PLC 通讯举例,和其他罗克韦尔产品的通讯指令详细内容,可另外参考其他手册或者咨询 BEACON 当地经销商和办事处。

如下举例中,仅针对 EtherNet/IP Client 指令部分内容进行介绍,暂不考虑上一章中提到的 PLC CIP 标签和 模块内部数据区地址映射的关系,以及内部数据区大小范围。

在实际操作中,因为不同产品型号的模块内部数据区大小不同,请务必注意模块数据区的实际大小,并根据实际数据寄存器的地址范围来配置指令,同时还请注意相同地址是否重复被多种协议写入数据。

此选项用于罗克韦尔 PLC 在不能停机的情况下,对 Logix5000 或者 Studio 5000 软件里面标签或者标签数组进 行读或写的操作。

Controller Tag	T		
Enable	Yes	Ŧ	
Function Type	CIP Data Table Read	•	
IP Address	192.168.0.12		
Slot	0		
Quantity	100		
Poll Interval	0		
Data Swap	No Change	*	
Internal Data Address	1000		
Data Type	INT	*	
Tag Name	AA		
Tag Offset	0		
Cmd Errors Mapping Enabled	Yes	*	
Cmd Errors Mapping Address	1200		
Desc			

Click save to continue add command, click close to finish add. Close Save

举例一:如上图,读取 IP 地址为 192.168.0.12,CPU 位于 0 槽位的 L63 CPU 里面的全局变量标签数组 AA,数 组是 INT 格式,数量 100 个 INT (每条命令最大 100 个 INT,或者 50 个 DINT/REAL),放到模块内部寄存器 1000-1099 里面,如果命令检测不到 AA 的数组有 100 个 INT 或者没有 AA 数组,或者 IP 地址不对,槽位不对等,就会在模块 内部寄存器 1200 的位置报一个非零值,显示这条命令有错误,工程师可以使用 Cmd Errors Mapping 反馈来查看所 连接设备的状态。(注:对于读来说 Data: Type 始终是 INT,不可修改,但是会随着数组的类型自动调整)



±-AA		INT[100]	Read/Write	Decimal

读取 IP 地址为 92.168.0.12, CPU 位于 0 槽位的 L63 CPU 里面的全局变量标签数组 BB,数组是 REAL 格式,数量 50 个 REAL (每条命令最大 100 个 INT,或者 50 个 DINT/REAL),放到模块内部寄存器 1100-1199 里面,如果命令检测不到 BB 的数组有 50 个 REAL 数据,或者 IP 地址不对,槽位不对等,就会在模块内部寄存器 1201 的位置报一个非零值,显示这条命令有错误,工程师可以使用 Cmd Errors Mapping 反馈来查看所连接设备的状态。

chable	Yes	*		
Function Type	CIP Data Table Read	•		
IP Address	192.168.0.12			
Slot	0			
Quantity	50			
Poll Interval	0			
Data Swap	No Change			
Internal Data Address	1100			
Data Type	REAL	•		
Tag Name	68			
Tag Offset	0			
Cmd Errors Mapping Enabled	Yes	•		
Cmd Errors Mapping Address	1201			
Desc				
			Close	140
			Close	ive
			Close Sa	ive

检查命令状态,点击 Comm Status 如下图,可以看发送和接收的次数,最后的错误代码等。

n Home				
Module	~			
Modbus Serial	+2	Parameter Name	Value	
		Command Count	2	
EtherNet/IP Server	~	TNS	6354	
	_	Last Error Code	0	
EtherNet/IP Client	+5	Number of Command Errors	0	
		Number of Requests Sent	1001	
Client 1		Number of Responses Received	1001	
		Number of Errors Received	D	
 Configuration 		Number of Errors Sent	0	
Commands				
Comm Status				Reset Counter
Command Errors				

在 AA 和 BB 输入些数据:

E-BB

🖉 Controller Tags - L63(control	ler)					
Scope: 10163 Show	All Tags			•	V. Enter Name I	Filler
Name	=== △ Value	• Force Ma	sk 🔶 S	Style	Data Type	Description
⊡-AA]	{} D	Decimal	INT[100]	
AA[0]		11	C	Decimal	INT	
■ AA[1]		11	0	Decimal	INT	
E AA[2]		123	C	Decimal	INT	
		123	0	Decimal	INT	

1	Controller Tags - L63(controller)				
9	Scope: 🛐 L63 💽	Show: All Tags			▼ 7. 4	Enter Name .
	Name	그 문고	Value 🗧 🗧	Force Mask 👘 🔦	Style	Data Type
	⊟-BB		{}	{}	Float	REAL[50]
			-888.99		Float	REAL
			0.0		Float	REAL
			0.0		Float	REAL
			77.22		Float	REAL
			0.0		Float	REAL

查看内部寄存器 1000 和 1100 的数据,此处说明 1 个 REAL 的浮点数占 2 个内部寄存器,虽然命令是 50 个浮点

数,放到1100开始的内部寄存器,实际上是1100-1199这100个寄存器存放着50个浮点数。

Decimal Display	/ Hexadeci	mal Display	Float Display	ASCII Display			
ddress	0	1	2	3	4	5	6
000	11	11	123	173	0	0	0
000	0	0	0	0	0	0	0
020	0	0	0	0	0	0	0
020	0	0	0	0	0	0	
040	0	0	0	0	0	0	
040	0	0	0	0	0	0	-
000	0	0	0	0	0	0	
000	0	0	0	0	0	0	
020	0	0	0	0	0	0	
000	0	0	0	0	0	0	-
Prev 1 2	6 7 8 w	9 10 1	11 12 13 14	15 32 33	Next		

Address	0	1	2	3	4	5	6	7	8
1100	16220	-15266	0	0	0	0	28836	17050	0
1110	0	0	0	0	0	0	0	0	0
1120	0	0	0	0	0	0	0	0	0
1130	0	0	0	0	0	0	0	0	0
1140	0	0	0	0	0	0	0	0	0
1150	0	0	0	0	0	0	0	0	0
1160	0	0	0	0	0	0	0	0	0
1170	0	0	0	0	0	0	0	0	0
1180	0	0	0	0	0	0	0	0	0
1190	0	0	0	0	0	0	0	0	0

Prev 1 2 ... 7 8 9 10 11 12 13 14 15 16 ... 32 33 Next

可以看到内部寄存器 1200 和 1201 没有错误反馈:

Decimal Display	Hexadecim	al Display	Float Display	ASCII Display		
Address	0	1	2	3	4	5
1200	0	0	0	0	0	0
1210	0	0	0	0	0	0
1220	0	0	0	0	0	0
1230	0	0	0	0	0	0
1240	0	0	0	0	0	0
1250	0	0	0	0	0	0
1260	0	0	0	0	0	0
1270	0	0	0	0	0	0
1280	0	0	0	0	0	0
1290	0	0	0	0	0	D

如果我们从 Logix5000 里面删除掉 AA 或者 BB 数组标签的时候,命令检测不到有这两个数组,就会在内部寄存器 1200 和 1201 里面报错误,其他协议可以采集存放错误标签寄存器来反馈命令的执行情况。也可以查看命令状态。 这里可以看到错误代码 4 产生,这里面错误代码含义很多种,如果命令检测不到 AA 的数组有 100 个或者没有 AA 数 组,或者 IP 地址不对,槽位不对等,就会在模块内部寄存器 1200 的位置报一个非 0 值,工程师编程时,此地址不等于 0 就表示命令没有执行下去,因为错误代码组合种类非常多,例如 IP 地址不对,又没有检测不到 AA 数组,这时候就会产生 IP 和检测不到 AA 数组的错误代码组合。这里不再详细介绍。

Parameter Name			Value			
Command Count			2			
INS			15697			
ast Error Code			4			
Number of Command Errors			936			
Number of Requests Sent			10344			
Number of Responses Received			9408			
Number of Errors Received			0			
lumber of Errore Sent			0			
Home / Internal Data	a View				Res ≪Auto Refre	set Counter sh 2 V Second
Home / Internal Data	a View				Res ≇Auto Refre	set Counter sh 2 ▼ Second
Home / Internal Data	a View Hexadecima	al Display	Float Display	ASCII Display	Res ≇Auto Refre	set Counter sh 2 V Secondi
Home / Internal Data	a View Hexadecima	al Display	Float Display	ASCII Display	Res ≇Auto Refre	set Counter sh 2 • Second
Home / Internal Data Decimal Display Address 1200	a View Hexadecima 0 4	al Display 1 4	Float Display 2 0	ASCII Display 3 0	Res Mauto Refre	set Counter sh 2 V Secondi
Home / Internal Data Decimal Display Address 1200	a View Hexadecima 0 4 0	al Display 1 4 0	Float Display 2 0 0	ASCII Display 3 0 0	Res ⊠Auto Refre	set Counter sh 2 • Second
Home / Internal Data Decimal Display Address 1200 1210 1220	Hexadecima 0 4 0 0	al Display 1 4 0 0	Float Display 2 0 0	ASCII Display 3 0 0 0	Res ⊮Auto Refre	set Counter sh 2 ▼ Secondi
Home / Internal Data Decimal Display Address 1200 1210 1220 1230	A View Hexadecima 0 4 0 0	al Display 1 4 0 0	Float Display 2 0 0 0 0	ASCII Display 3 0 0 0 0	Res ⊯Auto Refre	set Counter
Home / Internal Data Decimal Display Address 1200 1210 1220 1230 1240	View Hexadecima 0 4 0 0 0 0	al Display 1 4 0 0 0	Float Display 2 0 0 0 0 0 0	ASCII Display 3 0 0 0 0 0 0 0 0	Re: ⊮Auto Refre	ket Counter sh [2 ▼] Secondi
Home / Internal Data Decimal Display Address 1200 1210 1220 1230 1240 1250	Hexadecim: 0 4 0 0 0 0	al Display 1 4 0 0 0 0 0 0	Float Display 2 0 0 0 0 0 0 0 0 0 0 0	ASCII Display 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Ret ■Auto Refre	set Counter
Home / Internal Data Decimal Display Address 1200 1210 1220 1230 1250 1550 1550	A View Hexadecima 0 4 0 0 0 0 0 0 0 0	al Display 1 4 0 0 0 0 0 0	Float Display	ASCII Display	Ref Auto Refre	ket Counter sh [2 ▼] Secondi
Home / Internal Data Decimal Display Address 1200 1210 1220 1230 1240 1250 1260 1260 1270	2 View Hexadecima 0 4 0 0 0 0 0 0 0 0 0 0 0	al Display 1 4 0 0 0 0 0 0 0 0 0 0	Float Display 2 0 0 0 0 0 0 0 0 0 0 0 0 0	ASCII Display 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Ret ■Auto Refre	set Counter

Prev 1 2 ... 8 9 10 11 12 13 14 15 16 17 ... 32 33 Next

举例:连接 E300 马达保护器,请先查看 E300 用户手册,了解关于以太网连接的方法,E300 自带有 3 个输出继电器,如果控制输出继电器 1,继电器 2,继电器 3,就需要使用 CLASS CODE9,3 个继电器分别对应着 Instance1, Instance2, Instance3。Attribute选择 3 是对这个继电器写值,0=0FF 1=0N。

Discrete Output Point Object — CLASS CODE 0x0009

The following class attributes are supported for the Discrete Output Point Object:

Instance	Name	Description
1	OutputPt00	Control Module Output 0
2	OutputPt01	Control Module Output 1
3	OutputPt02	Control Module Output 2
4	OutDigMod1Pt00	Digital Expansion Module 1 Output 0
5	OutDigMod1Pt01	Digital Expansion Module 1 Output 1
6	OutDigMod2Pt00	Digital Expansion Module 2 Output 0
7	OutDigMod2Pt01	Digital Expansion Module 2 Output 1
8	OutDigMod3Pt00	Digital Expansion Module 3 Output 0
9	OutDigMod3Pt01	Digital Expansion Module 3 Output 1
10	OutDigMod4Pt00	Digital Expansion Module 4 Output 0
11	OutDigMod4Pt01	Digital Expansion Module 4 Output 1

All instances contains the following attributes.

Table 619	- Discrete	Output Point	Object	Instance	Attributes
-----------	------------	---------------------	--------	----------	------------

Attribute ID	Access Rule	Name	Data Type	Value
3	Get/Set	Value	BOOL	0=0FF, 1=0N
5	Get/Set	Fault Action	BOOL 0=Fault Value attribute, 1=Hol State	
6	Get/Set	Fault Value	BOOL	0=OFF, 1=ON
7	Get/Set	Idle Action	BOOL 0=Fault Value attribute, 1=Holo State	
8	Get/Set	Idle Value	BOOL	0=0FF, 1=0N
113	Get/Set	Pr Fault Action	BOOL	0=Pr Fault Value attribute, 1=Ignore
114	Get/Set	Pr Fault Value	BOOL	0=0FF, 1=0N
115	Get/Set	Force Enable	BOOL	0=Disable, 1=Enable
116	Get/Set	Force Value	BOOL	0=0FF, 1=0N
117	Get/Set	t/Set Input Binding STRUCT: Size of appendix I encoded Appendix I encoded path: N means attribute 3 drives th Array of USINT of USINT Otherwise, this is a path to instance of the DeviceLogix		Size of appendix I encoded path Appendix I encoded path: NULL path means attribute 3 drives the output. Otherwise, this is a path to a bit in an instance of the DeviceLogix Data Table

Home	me / EtherNet/IP Client 1 / Command List											
Cor												
Gei	Enable	Function Type	IP Address	Slot	Qty	Poll Interval	Data Swap	Internal Data Address	Cls Ins Att	Cmd Errors Mapping Enabled	Cmd Errors Mapping Address	Desc
01	Yes	Write Attribute Single	192.168.0.8	-1	1	o	No Change	1300	9 1 3	Yes	1400	
02	Yes	Write Attribute Single	192.168.0.8	-1	1	0	No Change	1301	9 2 3	Yes	1401	
©3	Yes	Write Attribute Single	192.168.0.8	-1	1	0	No Change	1302	9 3 3	Yes	1402	
	Add Modify Delete											
S٤	ve list to F	lash										

如上建立的3条指令,表示对IP地址为192.168.0.8的E300马达保护器3个输出继电器进行输出操作,如果内部 寄存器1300,1301,1302值为1的时候,3个输出继电器会进行闭合动作,如果内部寄存器1300,1301,1302值为0 的时候,3个输出继电器会进行分开动作,如果3条命令没有正确执行,内部寄存器1400,1401,1402会报一个非零 值。注:模块作为EtherNet/IP Client可以支持的内容非常多,根据需要连接的设备的不同(Logix控制器,PowerFlex 变频器,E300马达保护器,PowerMonitor电力仪表),可以和我们联系,获取进一步的详细技术支持。联系方式请 见手册最后一页。

配置模块做 Modbus RTU Master(主站)

MB指的通讯协议是MODBUS RTU,接线方式提供RS232/422/485三种可以选择。MB2代表2个MODBUS RTU接口,S1,S2 MODBUS RTU接口可以自由选择做主站或者从站。

Modbus RTU主站可以连接31个从站,两个串口S1和S2可以连接62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。



接线图

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RS485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,容易 对串口造成不必要的损坏。

打开浏览器,在左侧导航栏点击 Modbus Serial——Port1 里面的 Configuration,显示 S1 端口配置的页面,如下图:

	Port	On	・ 対	耑口使能	
	Mode	RS485	• 技	接线方式	
	Туре	Master	ت ألم الم	耑口主站/从站	
	Protocol	RTU	🖬 🚽	耑口协议	
	Baud Rate	19200	• Ì	端口波特率	
	Parity	None	• ²	奇偶效验位	Odd
	Data Bits	8	三 娄	牧据位	Even
	Stop Bits	1	• 1 ²	停止位 从站的响应时间	于效验
Modbus Serial	Response Timeout	1000	/		
Moubus Serial	Retry Count	3	I	直试次数	奇效验
Port 1	Minimum Command Delay	Ø	ı ب	录小命令 延时	
0	Command Trigger Address	-1	角	市令触友地址	
Configuration		Save			

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。以下举例中采用的 模块最多可以支持 4000 个字数据区地址范围,实际配置模块时,请按照模块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	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: 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	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:模块功能码 FC6 或者 FC16 时,写出数量是 20.模块内部寄存器起始地址 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,

# Home	Home / Modbus	Port 1 / Command List			
🚯 Module 🗸 🗸		Modbus Port 1 - Modify Comr	mand		3
General Hodbus Serial	EnableMode				
	•Yes EC 3	Enable	Yes	•	
Port 1		Modbus Function	FC 3 - Read Holding Registers(4X)	•	
• Configuration	Add	Slave Address	1		
Configuration		Modbus Data Address	20		
▶ Commands	Sav	Quantity	10		
		Data Swap	No Change	▼	
 Comm Status 		Poll Interval	0	j	
Slave Status		Internal Data Address	50		
		Desc			
Command Errors					
Diagnostics Log					

命令含义:把 MODBUS RTU 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 MODBUS RTU 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 MODSIM 32 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。

ModSim32 - ModSim1	Window Help		
ModSim1	· · · · · · · · · · · · · · · · · · ·		
Address: 0021 Length: 10 40021: <00000> 40022: <00000> 40023: <00000> 40024: <00000> 40025: <00000> 40026: <00000> 40027: <00000>	Device Id: 1 MODBUS Point Ty 03: HOLDING REGISTE 40028: <00000> 40029: <00000>	Frotocol Frotocol Frotocol Baud Data Stop Pau Hardware Flow Contr Wait for DTI Delay 0 Wait for CTS Delay 0 OK	SCII Daniel/ENRON protoco: 19200 - a 8 - p 1 - arity arity trol TR from Master ms after RTS before transmitting first TS from Mas ms after last character before
# Home Hom هه الأكياناء ب GModbus Serial 1	e / Modbus Port 1 / Status		ModSim32 - ModSim1 File Connection Display Window Help ModSim1 Device Id: 1 Address: 0021 MODBUS Point Type
Part 1 Port 1 Int Compared Co	rrameter Name ler-message Delay mmand Count list Error Code mber of Command Errors	Value 1.822 1 0 0	Length: 10 03: HOLDING REGISTER 40021: <00011> 40028: <00000> 40022 <00002> 40029 <00000>
Commitatives Commitatives Commitatives Commitatives Commitatives Command Errors	Imber of Requests Imber of Responses Imber of Errors Received Imber of Errors Sent	86 86 0 0	40023: <00000> 40030: <00000> 40024: <00000> 40025: <00000> 40025: <00000> 40026: <00000> 40027: <00000> 40027: <00000>
Diagnostics Log		v .	ZA
A Home	Home / Internal Data View		ModSim32 - ModSim1 File Connection Display Window Help ModSim1
General Configuration Internal Data View Backup / Restore	Decimal Display Hexade	cimal Display Float Disp	splay ASC Address: 0021 Device Id: 1 Address: 0021 D3: HOLDING REGISTER -
Change Password Firmware Upgrade Reboot Module	0 0 10 0 20 0 30 0 40 0 N	0 0 0 0 0 0 0 0 0 0 0 0	0 40021: <00011> 40028: <00000> 0 40022: <00022> 40023: <00000> 0 40023: <00000> 40030: <00000> 0 40024: <00000> 40030: <00000>
Modbus Serial Port 1	50 11 60 0 70 0 80 0 90 0	22 0 0 0 0 0 0 0 0 0 0 0 0 0	40026:<00000> 40027:<00000> -
 Configuration Commands Comm Status 	Prev 1 2 3 4 5 6 7	8 9 10 21 22	Next
Slave Status			

Dome	Home / Modbus Port 1 / Slave Status	ModSim32 - ModSim1
🚯 Module 🗸		ModSim1
☐ Modbus Serial +1		Address: 0021 Device Id: 1 Address: 0021 MODBUS Point Type
Port 1		Length: 10
Configuration		
Commands		40021: (00011) 40028: (00000) 40022: (00022) 40029: (00000) 40029: (00000)
▶ Comm Status		40023: <000005 40024: <000005 40025: <000005
 Slave Status 	Prev 1 2 3 4 5 6 7 8 Next	40026: <00000> 40027: <00000>
Command Errors		
Diagnostics Log	ZAuto Refresh 2 ▼ Second(s)	

Modbus Port 1 - Modify Command

Enable	Yes	▼ 使能 埜止 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

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。

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:

			-							
Decimal Di	splay Hexa	idecimal Display	Float Display	ASCII Displa	y					
Address	0	1	2	3	4	5	6	7	8	
2000	1122	0	3344	0	5566	0	7788	0	9999	
2010	0	0	0	0	0	0	0	0	0	
2020	0	0		-	-		~	-		_
2030	0	0	ModSim32	- ModSim1	_	_	_	_		
2040	0	0	File Connec	tion Display V	Vindow Help					
2050	0	0		terr eropiaj i	interest interp					_
2060	0	0	🗾 💭 ModSim:	1						
2070	0	0			Device Id:	1				
2080	0	0			Device ia.					
2090	0	0	Address:	0001	MODBUS P	oint lype				
					3: HOLDING RE	GISTER 💌				
			Length:	10						
Prev 1 2	16 17	18 19 20 21								

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。

Home / Modbus Port 1 / Status			
		ModSim32 - [ModSim1]	
		File Connection Display Window Help	_ & ×
		[000][000][216][034] Show Data	197 205 [001] [003] [020] [004]
		1098[000][000][013][
Parameter Name	Value		
Inter-message Delay	1.822	10391015100010001	
Command Count	1	[020][004][098][000][< Decimal	
Last Error Code	0	[000][039][015][000][Hex	000[000][010][197][205][001]
Number of Command Errors	0	[003][020][004][098][[021][190][000][000][030][108]
Number of Requests	566		
Number of Responses	565		
Number of Errors Paceived	0	[108][000][000][039][Floating Point	
Number of Energy Oast	0	Eloat (Swapped)	[οσο][οσο][σςτ][τασ][σσο][σσο]
Number of Errors Sent	U		
		64-Bit Floats	
		Auto Refresh 2 - Secon 64-Bit Swapped	

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:

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(内部寄存器地址2100)里面。

	Home / M	odbus Port 1 / Co	ommand Errors St	atus	Home / Intern	al Data View	/	
₽ Modbus Serial	+2							
Port 1	Decima	al Display Hexa	decimal Display		Decimal Dis	play He	adecimal Display	Float Display
	0	1	2	3	Address	0	1	2
Configuration	-2	0	0	0	2100	-2	0	0
-	0	0	0	0	2110	0	0	0
Commands	0	0	0	0	2120	0	0	0
oonnanao	0	0	0	0	2130	0	0	0
Comm Status	0	0	0	0	2140	0	0	0
Comm Status	0	0	0	0	2150	0	0	0
	0	0	0	0	2160	0	0	0
Slave Status	0	0	0	0	2170	0	0	0
	1 0	0	0	0	2180	0	0	0
Command Errors	0	0	5	0	2190	U	U	U
Diagnostics Log	Prev 1	2 Next			Prev 1 2	17 18	19 20 21 22	23 24 25

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。

Modbus Serial +2	Home / Modbus Port 1 / Diagnostics Log
₽Port 1	
▶ Configuration	
▶ Commands	
▶ Comm Status	Start Stop
▶ Slave Status	
Command Errors	Log Msg
Diagnostics Log	No. Message

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:



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。

EnableModbus Function		Slave Address	Modbus Data Address	Quantity	Data	Poll Interval	Internal Data Address	Cmd Errors Mapping Enabled	Cmd Errors Mapping Address De	
Ye	s	FC 3 - Read Holding Registers(4X)	1	0	10	No	0	0	Yes	20
Ye	s	FC 3 - Read Holding Registers(4X)	2	0	10	No Change	0	10	Yes	21

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。

Type	Master	•
Protocol	RTU	•
Baud Rate	19200	•
Parity	None	•
Data Bits	8	-
Stop Bits	1	•
Response Timeout	1000	
Retry Count	3	
Minimum Command Delay	0	
Command Trigger Address	30	
	Save	

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。

ModScan32 - ModSca1	
File Connection Setup View	Window Help
01 40 To 02 50 50 54	
Malfart	Connection Details
MINODSCAT	Connect
Address: 0001	Remote TCP/IP Server
100	IP Address: 192.168.0.200
Lengen: 100 U.	Service 502
	Loninguration - Vendeues Flam Canteral
** Device NOT CONNECTED! **	Baud 19200
40001: < 0> 40017: < 0: 40002: < 0> 40018: < 0:	Word 8 V Delay In ms after RTS before
40003: < 0> 40019: < 0	Parit NONE
40004: < 0> 40020: < 0: 40005: < 0> 40021: < -2	Delay 1 ms after last
40006: < 0> 40022: < -2	Stop 1 character before
40006: < 0> 40023: < 0: 40006: < 0> 40024: < 0:	
40009: < 0> 40025: < 0:	rotocol Selection
40010: < 0> 40020: < 0; 40011: < 0> 40027: < 0;	OK Cancel
40012 < 0> 40028 < 01	l

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。

ome / Inter	rnal Data	View			ModSca	1							
Decimal D	Display	Hexadecimal Display	Float Display		Address: Length:	[0001	03:	Device Id MODBUS HOLDING	Poin REG	l It Type ISTER		Ð
Address	0	1	2	3	40001: <	0>	40017: <	0>	40033: <	0>	40049:	<	0>
0	0	0	0	0	40002: <	0>	40018: <	0>	40034: <	0>	40050	~ ~ ~	0>
10	0	0	0	0	40004: <	0>	40020 <	-25	40030: <	0>	40053	2	0>
20	-2	-2	0	0	40006: <	0>	40022: <	-2>	40038: <	0>	40054:	<	0>
30	1	1	0	0	40007: <	<0>	40023: <	0>	40039: <	0>	40055:	<	0>
40	0	0	0	0	40008: <	0>	40024: <	0>	40040: <	0>	40056:	2	0>
50	0	0	0	0	40010: <	0>	40026: <	0>	40042 <	0>	40058	<	0>
60	0	0	0	0	40011: <	<0>	40027: <	0>	40043: <	0>	40059:	<	0>
70	0	0	0	0	40012: <	05	40020: <	05	40044: <	05	40060	5	0>
80	0	0	0	0	40014: <	0>	40030: <	0>	40046 <	0>	40062	<	0>
90	0	0	0	0	40015: < 40016: <	<0 <0	40031: < 40032: <	1> 1>	40047: < 40048: <	0> 0>	40063: 40064:	< <	0>

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):

R	Retry Count	3	

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。

Start	Stop
7	00 00 06 190 >> 01 03 00 00 00 0A C5 CD
8	00:00:07 199 >> 01 03 00 00 00 0A C5 CD
9	00:00:08:205 >> 01:03:00:00:00:0A C5 CD
10	00:00:09:212 >> 01:03:00:00:00:0A C5 CD
11	00:00 10:221 >> 01 03:00:00 00 0A C5 CD
12	00:00:11:228 >> 01:03:00:00:00:0A C5:CD
13	00:00 12:235 >> 01 03 00 00 00 0A C5 CD
14	00:00 13:242 >> 01 03 00 00 00 0A C5 CD
15	00:00 14:248 >> 01 03 00 00 00 0A C5 CD
16	00:00:15:256 >> 01:03:00:00:00 0A C5 CD
17	00:00 16.262 >> 01 03 00 00 00 0A C5 CD
18	00:00 17:269 >> 01:03:00:00:00 0A C5 CD
19	00:00:18:275 >> 01 03 00 00 00 0A C5 CD
20	00:00 19:283 >> 01 03 00 00 00 0A C5 CD
21	00:00 20:290 >> 01 03 00 00 00 0A C5 CD
22	00:00 21:297 >> 01 03 00 00 00 0A C5 CD
23	00:00:22.303 >> 01 03 00 00 00 0A C5 CD
24	00:00:23:312 >> 01 03 00 00 00 0A C5 CD
25	00.00 24.319 >> 01 03 00 00 00 0A C5 CD
26	00:00:25 325 >> 01 03 00 00 00 0A C5 CD

10001	0>	40017: <	0>	40033: <
40002: <	0>	40018: <	0>	40034: <
40003: <	0>	40019: <	0>	40035: <
40004: <	0>	40020: <	0>	40038 4
40005: <	0>	40021: <	-2>	40037: 4
40006: <	0>	40022: <	-2>	40038 -
40007: <	0>	40023 <	05	40039: «
40008: <	0>	40024: <	05	40040: -
40008 <	0>	40025- <	0>	40041: -
40010 <	0>	40026 <	6>	40042 4
40011 <	0>	40027: <	05	40043
40012 <	0>	40028 <	0>	40044 -
40013 <	6>	40029 <	(D)	40045: 4
40014 <	0>	40030 <	65	40046
40015- <	G> .	40031 <	1>	40047
10010. 1	0>	40032- <	65	4004R-

Log N	/lsg
No.	Message
1	00:00:00.294 >> 01 03 00 00 00 0A C5 CD
2	00:00:01.302 >> 01 03 00 00 00 0A C5 CD
3	00:00:02.310 >> 02 03 00 00 00 0A C5 FE
4	00:00:03.317 >> 02 03 00 00 00 0A C5 FE
5	00:00:04.324 >> 02 03 00 00 00 0A C5 FE
6	00:00:05.330 >> 02 03 00 00 00 0A C5 FE
7	00:00:06.339 >> 01 03 00 00 00 0A C5 CD
8	00:00:07.345 >> 01 03 00 00 00 0A C5 CD
9	00:00:08.352 >> 01 03 00 00 00 0A C5 CD
10	00:00:09.358 >> 01 03 00 00 00 0A C5 CD
11	00:00:10.366 >> 02 03 00 00 00 0A C5 FE
12	00:00:11.373 >> 02 03 00 00 00 0A C5 FE
13	00:00:12.380 >> 02 03 00 00 00 0A C5 FE
14	00:00:13.386 >> 02 03 00 00 00 0A C5 FE
15	00:00:14.395 >> 01 03 00 00 00 0A C5 CD

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 Slave (从站)

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

# Home	Home / Modbus Port 1 / Configuration		
∰ Module ✓			
Modbus Serial			
Port 1	Port	On	•
- Poit I	Mode	RS485	•
Configuration	Туре	Slave	•
Commands	Protocol	RTU	
Comm Status	Revel Date		
Slave Status	baug Rate	19200	•
Command Errors	Parity	None	•
Diagnostics Log	Data Bits	8	•
	Stop Bits	1	•
	Slave ID	1	
	Minimum Response Delay	1	
	Holding Register Offset	0	

点击Port1里面的Configuration显示S1端口配置的页面:

Modbus Serial	+2
Port 1	
Configuration	

注意事项: S1或者S2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
Туре	Slave	☑ 端口主站/从회
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	o	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
	Save	APPEND APPEND

使用ModScan32仿真Modbus RTU主站,可以对模块内部寄存器读写。

ModScan32 - [ModSca1]	
Pile Connection Setup View Window Help	_ 6 ×
Address: 0001 Device Id: 1 Address: 0001 MODBUS Point Type Valid Slave Responses: 0 Length: 100 03: HOLDING REGISTER Reset Ctrs	
** Deske NOT CONNECTED ** Connection Details 23 Connection Details Connection Details Connection Connection to COM2 Connect Descent Descent	Modbus Protocol Selections
10012 <	Delay Between Folls (nsees) Force modbus command 15 and 16 for single-point GF bounded in cases where the slaw down not support the single-point write functions OS support the single-point write functions OS OK Cancel
For Help, press F1	Polls: 0 Resps: 0

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。

BT-EN-MB2 快速启动手册

▲ 设备管理器				_		Mo	odScan32 - [ModSca1]
文件(F) 操作(A) 查看	(V) 帮助(H)					💼 Fil	Connection Details
♦ ♦ 〒 2 〒	1.						Connect
A lacon-PC							Direct Connection to COM2
A-B Virtual Ba	ckplane					Addre	Phone Number: 192.168.0.200
▶ 🚯 Bluetooth 无绘						Leng	gth: Configuration
D IDE ATA/ATAR	PI 控制器						Band 19200 V Hardware Flow Control
SIMATIC NET						** Doni	Word 8
> 📝 安全设备						40001:	C (V) C (V) C (V) Parit NONE Wait for CTS from sla NONE Wait for CTS from sla
▷ 🛄 处理器						40003:	Stop 1 ▼ Delay 1 ms after last character before
						40005:	
						40008:	< 0) rotocol Selection 00
▲ 一會 端口 (COM 和	LPT) tive Managem	ont Tochno		2)		40008:	< 0. 0K Cancel 10
	LI CH340 (CON	42)	10gy - 30E (COM	3)		40010:	
↓ ■ 汁質切						For He	elp, press F1 Polls: 40 Resps: 39
CLORE ILCHNOLOGI						_	
A Home	Home / Modb	ous Port 1 /	Status				
🔊 Module.						_	Address: 0001 Device Id: 1 Number of Polls: 30
- modalo						-	MODBUS Point Type Valid Slave Responses: 30
Modbus Serial +1						-	Length: 100 03: HOLDING REGISTER • Reset Ctrs
	Parameter N	lame			Value	e	
Port 1	Inter-messag	e Delay			1.822	2	
Configuration	Last Error Co	de				-	10011 C 0> 10011 C 0> 10011 C 0> 10001 C 0> 10001 C 0> 10011 C 0> 10001 C 10 1001 C 10 1000 C
• Commanda	Number of Co	ommand Erro	ors		0	_	40003; < 0> 40013; < 0> 40023; < 0> 40033; < 0> 40040; < 0> 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 0> 40054; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; < 40040; <
Commanus	Number of Re	equests			29	-	40006; < 0> 40016; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026; < 0> 40026;
Comm Status	Number of Er	rors Receive	d		0	-	. 40007: < 0> 40017: < 0> 40027: < 0> 40037: < 0> 40037: < 0> 40047: < 0> 40057: < 40008: < 0> 40018: < 0> 40028: < 0> 40038: < 0> 40038: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058: < 0> 40058:
Slave Status	Number of Er	rors Sent			0		40009: < 0> 40019: < 0> 40029: < 0> 40039: < 0> 40039: < 0> 40049: < 0> 40059: < 40010: < 0> 40020: < 0> 40030: < 0> 40040: < 0> 40050: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: < 0> 40060: <
Command Errors						_	For Help, press F1 Polls: 30 P
Diagnostics Log							
- Diagnostics Log					V	Auto R	tefresh 2 Second(s)
GLOBAL TECHNOLOGY							
						_	
A Home	Home / Inter	nal Data Vie	W				
🙆 Module 🔨 🔨							Address: 0001 Device Id: 1 Number of Polls: 150
						-	MODBOS Point Type Valid Slave Responses: 15
General Configuration	Decimal Di	splay H	exadecimal Display	Float [Display AS	SCII Disp	Reset Ctrs
Internal Data View						_	
Backup / Restore	Address	0	1	2	3		40001 < 111> 40011 < 0> 40021 < 0> 40031 < 0> 40041 < 0> 40051 <
Change Password	0	111	0	333	0		40002: < 0> 40012: < 0> 40022: < 0> 40032: < 0> 40042: < 0> 40042: < 0> 40052: < 40042: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: < 0> 40052: <
Change Password	10	0	0	0	0		40004; < 0> 40014; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024; < 0> 40024;
Firmware Upgrade	20	0	0	0	0		40006: < 0> 40016: < 0> 40026: < 0> 40036: < 0> 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 0> 40056: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 40046: < 400
Reboot Module	30 40	0	0	0	0		40007: < 0> 40017: < 0> 40027: < 0> 40037: < 0> 40037: < 0> 40047: < 0> 40057: < 40047: < 0> 40057: < 40040; < 40040; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 40050; < 0> 400
Madhua Carial	50	0	0	0	0		40009: < 0> 40019: < 0> 40029: < 0> 40039: < 0> 40039: < 0> 40049: < 0> 40059: < 0> 40059: < 0> 40059: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050: < 0> 40050:
	60	0	0	0	0		For Help, press F1
Port 1	80	0	0	0	0		
	90	0	0	0	0		0 0 0 0 0

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。

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

➡ ModScan32 - [ModSca1]	Iome / Internal Data View
🖬 File Connection Setup View Window Help	
	Decimal Display Hexadecimal Display Float Display ASCII Display
	Address 0 1 2 3
Address: 0001 Device Id: 1 MODBUS Point Type Valid Slave Responses: 20 Length: 100 03: HOLDING REGISTER	0 0 0 0 0 0 10 0 0 0 0 0 20 0 0 0 0 0 30 0 0 0 0 0 40 0 0 0 0 0 50 123 333 0 0 60 0 0 0 0 70 0 0 0 0 80 0 0 0 0
40001: < 123> 40006: < 0> 40011: < 0> 40016: < 0> 40021: < 0> 40026: < 40002: < 333> 40007: < 0> 40012: < 0> 40017: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027: < 0> 40027	90 0 0 0 0 0

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区的数值,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:

# Home	Home / Modbus Port 1 / Status		
A Module	•		
Modbus Serial	Parameter Name	Value	
Port 1	Inter-message Delay	1.822	
- Poit I	Command Count	1	
▶ Configuration	Last Error Code	0	
Configuration	Number of Command Errors	0	
▶ Commands	Number of Requests	1232	
	Number of Responses	1232	
Comm Status	Number of Errors Received	0	
▶ Slave Status	Number of Errors Sent	0	
Command Errors Diagnostics Log		Reset C	punter
Diagnostics Log		Auto Refresh 2	 Second(s)

可视化查看从站状态 点击Slave Status 可以看到1号从站是绿色的。



Command Errors 查看命令行是否有错误产生点击: Decimal Display Hexadecimal Display 8 9 0 4 0 0 0 0 0 0 0 0 0 0 0

报文诊断功能:点击Diagnostics Log,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。

	Home / Modbus Port 1 / Diagnostics Log
# Home	
A Module	✓ Start Stop
General Modbus Serial	•2 638 00:00:08,719 << 01:03:14:22:B8:2
₽Port 1	640 00:00:08,746 << 01 03 14 22 B8 28 B8 25 B8 2
 Configuration 	644 00:00:08:802 << 01:03 14 22 B8 28 B8 2
▶ Commands	646 00.00.08.830 << 01 03 14 22 B8 28 B5 25 B5 2
Comm Status	649 00.00.08.64 >> 01 03 14 22 B5 25 B5 26 26 25 B5 26 25 B5 26 25 B5 26 25 B5 26 26 26 26 26 B5 26 26 26 26 26 26 26 26 26 26 26 26 26
Slave Status	651 00:00:08.892 >> 01 03 03 E8 00 0A 45 BD 652 00:00:08.916 << 01 03 14 22 B8 28 B3 F
Command Errors	653 00:00:08:921 >> 01 03 03 E8 00 0A 45 BD 654 00:00:08:944 << 01 03 14 22 B8 28
Diagnostics Log	655 00:00:08:949 >> 01 03 03 E8 00 0A 45 BD 656 00:00:08:973 << 01 03 14 22 B8 28 B3 25 B3 55 657 00:00:08:978 >> 01 03 03 E8 00 0A 45 BD

举例 1. Modbus 从站设备和罗克韦尔 PLC 交换整型数数据

本案例中,模块配置作为Modbus RTU的master,同时配置作为EtherNet/IP的server

Server File Size	100		•
Save			
Suit			
Class 4 Connections			
Class 1 Connections			
Class 1 Connections	Input Size	Output Data Address	Output Size
Class 1 Connections	Input Size	Output Data Address	Output Size
Class 1 Connections	Input Size 248 248	Output Data Address 1000 1250	Output Size
Class 1 Connections	Input Size 248 248 248	Output Data Address 1000 1250 1500	Output Size 248 248 248

Logix 5000的输入输出标签和模块内部寄存器地址对应关系如下

PLC输入CIP标签组, EN:0:I. Data[0]- EN:0:I. Data[247] 对应模块内部寄存器 0-247 的地址。

PLC输出CIP标签组, EN:0:0. Data[0]- EN:0:0. Data[247] 对应模块内部寄存器1500-1747 的地址。

以此类对,共计4组输入输出标签组:

如下图,我们在模块Modbus驱动一侧,使用功能码FC3读1号从站数据,从1号从站的40001至40010一共读10个字(INT类型),放入模块内部数据寄存器0-9,修改成以下配置。功点击Save,保存该指令的修改。

Enable		Yes		•	佑坐 林山	ட டிஷ்	安方型者	「本小戸」	e			
Modbus Function	1	FC 3 - Read Holding	Registers(4)	• ()	CRE, 示山 Modbus ひ	L,内司 1能码FC) 町1丁 前前 1 1 FC2 FC	3 FC4 FC	=9 5 EC6 EC15 EC16			
Slave Address		1			从站地址		1,102,10	0,1 04,1 0	0,1 00,1 010,1 010			
Modbus Data Ad	dress	0			从站读写数	如据Moo	bus起始	位				
Quantity		10			读或者写的	的数据的	数量					
Data Swap		No Change		-	数据高低位	立交换.	字交换.	字节交	换、字和字节交换	L		
Poll Interval		0			命令轮询时	前间						
Internal Data Add	Internal Data Address			j	模块内部者	客存器,	存放数据	舌的起始:	地址			
Cmd Errors Map	Cmd Errors Mapping Enabled		Yes			状态位反	馈开启					
Cmd Errors Mapping Address		500			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					
← → C [] 192.168.0.25	Home / Modbus P	ort 1 / Command List								Thi		
🚳 Module 🗸												
Modbus Serial			Slave	Modbu	s Data	Quantity	Data Swap	Poll	Internal Data	Desc		
Port 1	EnableModb	us Function	Address	Addres	is			Interval	Audress			
🖵 Port 1	Yes FC 3	us Function - Read Holding	Address	Addres	5	10	No	0	0			
	Yes FC 3 Regist	us Function - Read Holding ters(4X)	Address 1	Addres 0	.5	10	No Change	0	0			
Port 1 Configuration Commands	Yes FC 3 Regisi	us Function - Read Holding ters(4X) Modify Delete	Address 1	Addres 0	5	10	No Change	0	0			
Port 1 Configuration Commands Comm Status	enableModbi Yes FC 3 Regis	us Function - Read Holding ters(4X) Modify Delete	Address 1	Addres 0	5	10	No Change	0	0			
Port 1 Configuration Commands Comm Status Status Status	enableModb Yes FC 3 Regist Add	us Function - Read Holding ters(4X) Modify Delete	Address 1	Addres 0	5	10	No Change	0	0			
Port 1 Configuration Commands Comm Status Slave Status Command Errors	EnableModb Yes FC 3 Regis	us Function - Read Holding ters(4X) Modify Delete	Address 1	Addres 0	5	10	No Change	0	0			
Port 1 Configuration Commands Comm Status Status Status Command Errors Diagnostics Log	Enable Modble Ves FC 3 Regisi Add	us Function - Read Holding ters(4X) Modify Delete	Address 1	Addres 0	15	10	No Change	0	0			

配置界面弹出成功(Successful),然后点击Close(关闭)。

点击下面的蓝色的Save保存命令写入模块缓存,点击OK。重启模块。模块倒计时19秒之后,新命令生效。

Modbus Serial	Warning
🖵 Port 1	The module has to be rebooted due to any configuration changes. Note that the data communication will be temporarily interrupted if reboot.
Configuration	OK to reboot the module now?
Commands	ок
Comm Status	
 Slave Status 	

仿真:

硬件连接:把USB转RS232电缆插到电脑上(如果现场电脑没有串行接口),中间经过RS232交叉转换头,再通 过一根橙色的9针公头转RJ45水晶头,这三种接在一起,进行RS232接线方式的仿真工作。电脑会弹出装USB转232电 缆的驱动,装好后,在电脑里面可以看到如下图,本文以及COM1说明: (注:此端口可以修改)

USB Serial Port (COM1)

软件连接: 打开ModSim32,可以仿真Modbus RTU 从站。设置从站地址,从站数据量,寄存器种类。如下图:选择连接Port 1(USB转232串口在电脑里的配置),然后把波特率,数据位,停止位,奇偶效验位,设置成与主站相同的参数。

29

	Setup Comm Port 1
ModSim32 - ModSim1	Protocol
File Connection Display Window Help	
C ModSim1	Baud 19200 -
Device Id: 1 Address: 0001 MODBUS Point Type 03: HOLDING REGISTER	Data 8 - Stop 1 - Parity MODE -
* * * NOT CONNECTED! * * *	Hardware Flow Control
40001: <00000> 40008: <00000> 40002: <00000> 4009: <00000> 40003: <00000> 40010: <00000> 40004: <00000> 40005: <00000> 40006: <00000>	Vait for DTR from Master Delay 0 ms after KTS before transmitting first Wait for CTS from Mas Delay 0 ms after last character before
ModSim32 - ModSim1	
File Connection Display Window Help	
Connect > Port 1	
Disconnect Port 2	
D. (2)	

点击上面的OK,完成串口仿真连接。

在ModSim32仿真的从站里面输入数据

ModSim32 - ModSim1
File Connection Display Window Help
ModSim1
Device Id: 1
Address: 0001 MODBUS Point Type
03: HOLDING REGISTER 🔻
Length: 10
40001: 2001111 40000: 200000
40003: <00333> 40010: <06789>
40004: <00444>
40005: <00555>
40006: <00666>
40007: <00777>

因为之前我们在模块内保存的命令(如下图)。

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
Poli Interval	0	命令轮询时间
Internal Data Address	0	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	500	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

所以数据先被模块的Modbus主站驱动协议,读到模块内部数据区寄存器0-9里面。

🖨 Home		Home / Interna	ne / Internal Data View									
B Module	^											
General Configuration		Decimal Dis	nlav Her	vadecimal Display	Eloat D	ienlav ASC	II Display					
Internal Data View		Decimar Dis	pidy Tie.	auccimar Displa	Tioat D	ispidy Abo	Позрау					
Backup / Restore		Address	0	1	2	3	4	5	6	7	8	9
Change Password		0	111	222	333	444	555	666	777	888	999	6789
Firmware Upgrade		10 20	0	0	0	0	0	0	0	0	0	0

👸 RS	Logix 5000 - EN_EN in EN_MB.ACD [1756-L63 20.12]* - [Controller	Tags - EN_EN(controller)]	ten: magniture.	And Michael Prof.			
2 F	ile Edit View Search Logic Communications To	ols Wind	dow Help					
Ē	📂 🖬 🎒 👗 🛍 🛍 🗠 🗠 🛛 msg	•	- 🚜 🐴 强 🛅 🛛 🗏	Y ⊕, ⊖, Select i	a Language 👻 🥺			
Rem	Run Run Mode 🔜 🎆 Pat	α AB_ETH	IIP-1\192.168.0.11\Backplane\0*	▼ 🖁				
No Fo	ces							
NoEd	its 🔒 🗖 VOOK 4 H	h ha	++ +/+ -()(U)(L)-					
Redur	dancy 🖏	Favorites	🕻 Add-On 👗 Safety 👗 Alarms	Bit 🖌 Timer/C				
D	Controller Organizer	• 4 X	Scope: To EN_EN	 Show: All Tags 		•	Y. Enter Name Filte	×
Sta	🔤 Unscheduled Programs / Phases	^ I	Name	==! 2	Value +	Force Mask	Stule	Data Tune
	🖨 📹 Motion Groups		+ EN11:2:C		()	{}		AB:1756 MODU
ĝ	- 🗀 Ungrouped Axes		+ EN11:1:0		{}	{}		AB:1756 MODU
-	🗀 Add-On Instructions				{}	{}		AB:1756 MODU
	🖨 😑 Data Types		+ EN11:1:C		{}	{}		AB:1756 MODU
	🙀 User-Defined		± EN11:0:0		{}	{}		AB:1756_MODU
	🖶 🙀 Strings		EN11:0:I		{}	{}		AB:1756_MODU
	🚘 Add-On-Defined		- EN11:0:1.D ata		{}	{}	Decimal	INT[248]
	🕀 🙀 Predefined		+ EN11:0:I.Data[0]		111		Decimal	INT
	🗈 🚘 Module-Defined		+ EN11:0:I.Data[1]		222		Decimal	INT
	- 🗀 Trends		+ EN11:0:I.Data[2]		333		Decimal	INT
	- 🔄 I/O Configuration		EN11:0:I.Data[3]		444		Decimal	INT
	= 1756 Backplane, 1756-A4		EN11:0:I.Data[4]		555		Decimal	INT
	10 11756-L63 EN EN		+ EN11:0:1.Data[5]		666		Decimal	INT
	121 1756-EN2T EN1	=	EN11:0:I.Data[6]		777		Decimal	INT
	Ethernet		±-EN11:0:I.Data[7]		888		Decimal	INT
	1756-EN2T EN1		+ EN11:0:1.Data[8]		999		Decimal	INT
			+ EN11:0.1.Data[9]		6789		Decimal	INT
			+ EN11:0:I.Data(10)		0		Decimal	INT
			+ EN11:0:1.Data[11]		0		Decimal	INT
			EN11:0:I.Data[12]		0		Decimal	INT
	I CIP-MODULE EN111		EN11:0:I.Data[13]		0		Decimal	INT
	□ 2 CIP-MODULE EN222				0		Decimal	INT
	II 3 CIP-MODULE EN333	-	+ EN11:0:1.Data[15]		0		Decimal	INT
i in	~		+ EN11:0:1.Data[16]		0		Decimal	INT

同时被Logix PLC读取到EN:0:I.Data[0]- EN:0:I.Data[9],如下图。

模块作为Modbus主站写一组数据到仿真软件(从站)里,先明确模块内部数据区的划分,1000以后是PLC输出的数据地址。所以RSLogix5000内输出的CIP标签组EN11:0:0.Data[0]-EN11:0:0.Data[9]对应模块内部寄存器地址1000-1009。

	{}
E-EN11:0:0.Data	{}
	8888
	8888
	8888
	8888
	8888
	8888
	8888
	8888
	8888
	8888

增加一个Modbus主站的写命令,功能码使用16,将模块内部数据区地址从1000开始的10个字写入到1号Modbus从站的40101-40110中,也就是把RSLogix5000内输出的CIP标签组EN11:0:0.Data[0]-EN11:0:0.Data[9]这10个数据写到Modbus RTU的1号从站,数据地址为40101-40110里面。

Yes	•	
FC 16 - Preset (Write) Multiple Registe	•	
1		
100		
10		
No Change	•	
0		
1000		
	•	
	Fes FC 16 - Preset (Write) Multiple Registe 1 100 10 No Change 0 1000	Fes • FC 16 - Preset (Write) Multiple Registe • • 1 • 100 • 10 • 10 • 100 • 100 • 100 • 100 • 100 • 1000 • • •

先看一下模块内部寄存器1000开始的地址,有数据从Logix5000写入到模块:

# Home	Home / Inter	me / Internal Data View									
Bo Module	^										
General Configuration	Desimal	Display Hay	radecimal Displa	Elect Di	ionlay ASCI	I Dieplay					
Internal Data View	Decimar	hispidy Her	cauecimai Displa	iy Float Di	ispiay ASCI	i Dispiay					
Backup / Restore	Address	0	1	2	3	4	5	6	7	8	9
Change Password	1000	8888	8888	8888	8888	8888	8888	8888	8888	8888	8888
	1010	0	0	0	0	0	0	0	0	0	0
 Firmware Upgrade 	1020	0	0	0	0	0	0	0	0	0	0
· Cot Data & Time	1030	0	0	0	0	0	0	0	0	0	0
set Date & Time	1040	0	0	0	0	0	0	0	0	0	0
▶ Reboot Module	1050	0	0	0	0	0	0	0	0	0	0

再看一下Modsim32模拟的1号从站,40101-40110地址的数据。

ModSim32 - ModSim1	
File Connection Displa	iy Window Help
ModSim1	
Address: 01 01 Length: 10	Device Id: 1 MODBUS Point Type 03: HOLDING REGISTER •
40101: <08888> 40102: <08888> 40103: <08888> 40104: <08888> 40105: <08888> 40105: <08888> 40106: <08888> 40107: <08888>	40108: <08888> 40109: <08888> 40110: <08888>

至此,通过EtherNet/IP对于Modbus从站读和写字(WORD/INT)都已完成。

举例 2. Modbus 从站设备和罗克韦尔 PLC 交换布尔量数据

本案例中,模块配置作为Modbus RTU的master,同时配置作为EtherNet/IP的server。

特殊说明一下位(bit/bool)的操作,一般常见的应用是针对位寄存器,举一个例子来说明:

E-EN11:2:0	{}				
	{}				
- EN11:2:0.Data[0]	22				
EN11:2:0.Data[0].0	0				
EN11:2:0.Data[0].1	1				
EN11:2:0.Data[0].2	1				
EN11:2:0.Data[0].3	0				
EN11:2:0.Data[0].4	1				
	0				
	0				
-EN11:2:0.Data[0].7	0				

想把PLC中输出CIP标签中EN11:2:0. Data[0].0-EN11:2:0. Data[0].7,这8个布尔量(如上图)写入到1号Modbus 从站线圈00055-00062这8个位里面(如下图)。PLC中输出CIP标签中EN11:2:0. Data[0] EN11:2:0. Data[247]对应 模块内部寄存器地址1500-1747,因为内部寄存器是16位的字,所以其中EN11:2:0. Data[0].0-EN11:2:0. Data[0].7 对应于模块内部寄存器地址1500的前8位。

ModSim32 - ModSim2
File Connection Display Window Help
💭 ModSim2
Device Id: 1
Address: 0055 MODBUS Point Type
01: COIL STATUS
Length: 10
00055.202 00062.202
00056: <0> 00063: <0>
00057: <0> 00064: <0>
00058: <0>
00059: <0>
UUUB1: <u></u>
<

特别注意模块的内部寄存器是16位的字表示,如果要对位地址进行读写,指令中使用的模块的内部寄存器的起 始地址要乘以16才能对应上。

再看一遍对应关系。PLC输入标签组EN11:2:0. Data,对应模块内部寄存器起始地址为1500。

1个模块内部寄存器=16个位,想对从站的位线圈进行读写,要用1500*16=24000 来作为起始地址。在模块Modbus 主站一侧配置命令(如下图),可以选用功能码15 写多个线圈,从模块内部寄存器1500开始,连续调用8个布尔量, 写入到1号Modbus从站, 00055-00062这8个位里面。

Modbus Port 1 - Add Command

Enable	Yes 🔹
Modbus Function	FC 15 - Force (Write) Multiple Coils (0) •
Slave Address	1
Modbus Data Address	54
Quantity	8
Data Swap	No Change 🔹
Poll Interval	0
Internal Data Address 🛛 🛛 💥	24000

指令配置完成生效后, Modsim32模拟的从站才能正确接收到位(bit/bool)的数据,如下图:

ModSim32 - ModSim2
File Connection Display Window Help
ModSim2
Device Id: 1
Address: 0055 MODBUS Point Type
01: COIL STATUS
Length: 10
00055.307 00052.307
00058: <0>
00059: <1>
00060: <0>
00061: <0>

同理,如果是模块读取Modbus从站位数据的时候,要保存数据到模块的内部寄存器时,指令中填写的内部数据

地址,需要使用模块实际内部地址*16的方式。如下图:

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

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Modbus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。

举例 3. Modbus 主站和罗克韦尔 PLC 交换数据

本案例中,模块配置作为Modbus RTU的slave,同时配置作为EtherNet/IP的server

BT-EN-MB系列模块有多个串口的型号可选,每个串口都可以做Modbus从站,是共用一个数据区的。

如下从站测试使用两种方式举例,

第一种使用ModScan32模拟Modbus主站,模块Port1作为Modbus从站。

第二种使用Port1做Modbus主站,Port2做Modbus从站来进行数据交换。

测试方法1,如下图,把模块Port1 改成从站,从站站号Slave ID设置为 1,点击保存。从站不需要配置命令:

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

打开ModScan32,按照下面的配置设置。

🖼 ModScan32 - ModSca1	
File Connection Setup View Window Help	
🖴 ModSca1	
Address: 0001 Device Id: 1 MODBUS Point Type Valid Slave Responses: 0	
Length: 10 03: HOLDING REGISTER Reset Ctrs	
** Device NOT CONNECTED! ** 400001 < 0> 40004 < 0> 40007 < 0> 40010 < 0>	
40002: < 0> 40005: < 0> 40008: < 0>	

修改ModScan32配置参数,修改 Delay I ms after last character before 默认10,改成1。

在协议选择里面修改延时改成如下图示。点击OK。

Connection Details	Modbus Protocol Selections
Connect Direct Connection to COM1 Phone Number: 192.168.250.236	Transmizzion Mode STANDARD DANIEL/ENROM/OMNI CASCII GRU CASCII CRU
Configuration Baud 19200 V Word 8 V Farit MONE V Stop V S	Slave Response Timeout 500 (msecs) Delay Between Folls 100 (msecs) Force modbus command 15 and 16 for single-poin (To be used in cases where the slave does not support the single-point write functions 05 and 10 h 0K Cancel

连接后在40001写入55,40002写入85。

I	🍽 ModScan32 - ModSca1
į	File Connection Setup View Window Help
ĺ	m ModSca1
	Address: 0001 Device Id: 1 MODBUS Point Type Valid Slave Responses: 18
l	Length: 10 03: HOLDING REGISTER
I	
1	
	40001: < 55> 40004: < 0> 40007: < 0> 40010: < 0>
	40001: < 55> 40004: < 0> 40007: < 0> 40010: < 0> 40002: < 85> 40005: < 0> 40008: < 0> 40003: < 0> 40006: < 0> 40009: < 0>

前文中我们提到Modbus主站和模块内部数据区的对应关系如下:

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

BT-EN-MB2 快速启动手册

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	II	11601	至	11616	=	01601	至	01616
220	=	40221	=	30221	=	13521	至	13536	=	03521	至	03536
1000	=	41001	=	31001	II	26001	至	26016	=	16001	至	16016
1001	=	41002	=	31002	=	26017	至	26032	=	16017	至	16032
1999	=	42000	=	32000	II	41985	至	42000	=	31985	至	32000
2000	=	42001	=	32001	=	42001	至	42016	=	32001	至	32016
2001	=	42002	=	32002	Ш	42017	至	42032	=	32017	至	32032
3000	=	43001	=	33001	=	58001	至	58016	=	48001	至	48016

查看模块内部数据区,可以看到40001对应的是内部寄存器0,40002对应的是内部寄存器1。

		EtherNet/IP - Mod	bus	Serial			
A Home		Home / Internal Da	ta V	īew			
🙆 Module	^						
General Configuration		Decimal Display		Hevadecimal	Display	Float Display	ASCILI
▶ Internal Data View		Decimar Display		Tiexadecimar	Display	Tioat Display	70011
Backup / Restore		Address	0		1	2	3
Change Password		0	55		85	0	0
		10	0		0	0	0

ModScan32的报文如下,看报文点击Show Traffic:

ModScan32 - ModSca1				
File Connection Set	tup View Window Help			
	Data Definition			
	Display Options Show Data			
	Extended			
➡ ModScan32 - ModSca1 File Connection Setup View Window Help				
ModSca1				
Address: 0001 Device Id: 1 Address: 0001 MODBUS Point Type Valid Slave Responses: 64 Length: 10 03: HOLDING REGISTER • Reset Ctrs				
$ \begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2$		20[010] 11[003] 55[000] 20[000] 20[000] 20[000] 22[001] 20[000] 22[000]		

🖵 Port 1	Home / Modbus Port 1 / Diagnostics Log		
Configuration			
Commands	Start Stop		
Comm Status	5 00:00 03.040 << 01 03 00 00 00 A C5 CD 6 00:00 03.043 >> 01 03 14 00 37 00 55 00 00 00 00 00 00 00 00 00 00 00	7	
Slave Status	8 00:00:04.057 >> 01 03 14 00 37 00 55 00 00 00 00 00 00 00 00 00 00 00	7	
Command Errors	10 00:00:65:071 >> 01 03 14 00 37 00 55 00 00 00 00 00 00 00 00 00 00 00	7	
Diagnostics Log	13 00:00:07.095 << 01 03:00:00:00 A C5 CD 14 00:00:07.098 >> 01 03:14:00:37:00:55:00:00:00:00:00:00:00:00:00:00:00:	7	

测试方法2, 先在RSlogix5000的CIP输出数组EN11:0:0. Data[0]-EN11:0:0. Data[9]中录入一些数据, 如下图。



前文提到了EN11:0:0. Data[0]- EN11:0:0. Data[247]对应了模块内部寄存器 1000-1247 的地址,所以在模块

内部寄存器中1000-1009中可以看到这写数据。

ome / Intern	al Data View									
Decimal Di	isplay Hex	adecimal Displa	iy Float Dis	splay ASCI	I Display					
Address	0	1	2	3	4	5	6	7	8	9
1000	8888	8888	8888	8888	8888	8888	8888	8888	8888	8888
1010	0	0	0	0	0	0	0	0	0	0
1020	0	0	0	0	0	0	0	0	0	0

接着把模块Port2设置为Modbus从站(如下图),站号为1号从站,具体设置方法参考测试方法1中内容。

🚳 Module 💊	-			
🖵 Modbus Serial 🛛 🔫	Port	On	□ 端口使能	
Port 1	Mode	R\$232	▶ 接线方式	
Port 2	Туре	Slave	■ 端口主站/从	(立古
	Protocol	RTU	▶ 端口协议	
Commands	Baud Rate	19200	▶ 端口波特率	
Commands Comm Status Slave Status Command Errors Diagnostics Log	Parity	None	• 奇偶效验位	
	Data Bits	8	> 数据位	
	Stop Bits	[t	▶ 停止位	
	Slave ID	1	从站地址	
EtherNet/IP Server	Minimum Response Delay	(1	最小响应延	时
EtherNet/IP Client +5	Holding Register Offset	o	数据偏移	
	Word Input Offset	6	字输入偏移	
	Bit Input Offset	0	位输入偏移	
	Bit Output Offset	(e	位输出偏移	
		Save		

之后把模块的Port1 改成主站(如下图)。同时完成硬件配置:要把两个串口S1和S2用橙色的RJ45水晶头连接,

中间经过一个2/3交叉的线接在一起 (RS232模式):

🖌 🖟 BT-EN-MB2-B 🛛 🗙 🦲 and a state of the state of th					
← → C 🗋 192.168.0.250/index.asp#					
	EtherNet/IP - Modbus Serial				
A Home	Home / Modbus Port 1 / Configuration				
🚳 Module 🗸 🗸					
Modbus Serial →2	Port	On	•		
	Mode	R\$232	•		
▶ Configuration	Туре	Master	•		
▶ Commands	Pretocol	- motor			
▶ Comm Status		RIU			
Islave Status	Baud Rate	19200	T		
Command Errors	Parity	None	T		
 Diagnostics Log 	Data Bits	8	۲		
Port 2	Stop Bits	1	۲		
EtherNet/IP Server	Response Timeout	1000			
EtherNet/IP Client	Retry Count	3			
	Minimum Command Delay	0			
	Command Trigger Address	-1			
		Save			

之后点击Port1 Commands,配置一条命令用模块Port1端口作为Modbus主站,读取1号从站(Port2端口)的 41001-41010这10个数据(对应模块内部寄存器地址为1000-1009),放到模块内部寄存器地址0-9里面,模块这段 内部地址区对应着RSLogix5000输入CIP数组 EN11:0:I.Data[0]-EN11:0:I.Data[9],配置如下:

C 192.100.0.25	o/muex.asp#			
BEACON GLOBAL TECHNOLOGY	EtherNet	/IP - Modbus Serial		Log
# Home	Home / Modbu	s Port 1 Command List		
🙆 Module 🗸		Modbus Port 1 - Modify Cor	mmand	
Modbus Serial	EnableMod			
D Port 1	• Yes FC 3	Enable	Yes	•
P rofe 1	1	Modbus Function	FC 3 - Read Holding Registers(4X)	
- Configuration	Add	Slave Address	1	
Computation		Modbus Data Address	1000	
▶ Commands	Sin	Quantity	10	
		Data Swap	No Change	
Comm Status		Poll Interval	0	
Slave Status		Internal Data Address	D	

1) Logix 5000的输入标签组EN11:0:I. Data[0]-EN11:0:I. Data[9], 读取模块内部数据区 0-9 的数据;

2) 模块内部数据区 0-9 的数据,来自于模块端口1作为Modbus主站,读取端口2(Modbus从站)的数据;

3) 端口2作为Modbus从站时,其41001-41010的地址区实际上对应了模块内部数据区1000-1009的数据;

4) 模块内部数据区1000-1009的数据,来自于Logix 5000的输出标签组EN11:0:0. Data[0]-EN11:0:0. Data[9]

的数据;

5)所以等于Logix 5000的输入标签组EN11:0:I. Data[0]-EN11:0:I. Data[9] 读取输出标签组EN11:0:0. Data[0]

逻辑关系如下:

- EN11:0:0. Data[9]的数据。

点击保存,重启模块。

查看RS1goix5000里面的标签,可以看到PLC的输入CIP数据区,读取到了刚才输出CIP数据区中录入的数据。

RSLogix 5000 - EN_EN in EN_MB.ACD [1756-L63 20.12]*	P DAME: PROPERTY.	and the same first start
File Edit View Search Logic Communications Tools Window	w Help	
🎦 📂 🖬 🎒 🎼 🕫 🖙 🖼 msg	🗸 🦀 🐴 强 🋅 📝 🛒 🍭 🔍 🛛 Select a L	anguage 👻 🤵
Rem Run Run Mode No Forces Controller DK No Edits Battery DK Redundancy 5.45	HIP-1\192168.011\Backplane\0*	
Controller Organizer - 🗸 🗸 🗙	Controller Tags - EN_EN(controller)	
Unscheduled Programs / Phases		
🕆 🖶 🔄 Motion Groups	Scope: BELLEN	
ungrouped Axes	Name == V	Value 🔶
Add-On Instructions		{}
🖃 🚔 Data Types	± EN11:1:C	{}
- 🕞 User-Defined	EN11:0:0	{}
🕀 🙀 Strings	+ EN11:0:0.Data	{}
- 🛱 Add-On-Defined	EN11:0:1	{}
🗄 🙀 Predefined	EN11:0:I.Data	{}
🗄 🚂 Module-Defined	+ EN11:0:I.Data[0]	8888
	+ EN11:0:I.Data[1]	8888
E 😌 I/O Configuration	+ EN11:0:I.Data[2]	8888
	+ EN11:0:I.Data[3]	8888
101 1756-163 EN EN	+ EN11:0:I.Data[4]	8888
I [2] 1756-EN2T EN1	+ EN11:0:I.Data[5]	8888
E Sthernet	+ EN11:0:I.Data[6]	8888
	+ EN11:0:I.Data[7]	8888
	+ EN11:0:I.Data[8]	8888
	+ EN11:0:I.Data[9]	8888
E-E CIP Bus	+ EN11:0:1.Data[10]	0
0 CIP-MODULE EN000	+ EN11:0:1.Data[11]	0
1 CIP-MODULE EN111	+ EN11:0:I.Data[12]	0
2 CIP-MODULE EN222	+ EN11:0:1.Data[13]	0
3 CIP-MODULE EN333	+ EN11:0:I.Data[14]	0
	+ EN11:0:1 Data[15]	0

其他版本简介

本案例分别介绍了最为常见的模块作为EtherNet/IP从站和1756系统(做主站)通讯,同时模块作为Modbus主站读取/写入现场Modbus从站数据的应用,以及模块作为Modbus从站将主站数据传回给1756 PLC的应用。

BT-EN-MB2-B(基础版)可以支持4个EtherNet/IP Class1I/0链接,数据交换区2000个字,支持作为最多5个 EtherNet/IP Client和5个EtherNet/IP Server,2个串口可以作为Modbus RTU的主站或者从站使用,做主站时每个 端口可以支持64条Modbus指令。

BT-EN-MB2-P(领先版)可以支持10个EtherNet/IP Class1 I/0链接,数据交换区5000个字,支持作为最多15个 EtherNet/IP Client和15个EtherNet/IP Server,2个串口可以作为Modbus RTU的主站或者从站使用,做主站时每 个端口可以支持128条Modbus指令。

BT-EN-MB2-B/P模块除了可以支持EtherNet/IP Class1I/0链接以外,还可以支持EtherNet/IP Class3采用MSG 命令完成数据交换,此种方式数据交换的速度会大大低于Class 1的方式,配置命令也比较繁琐,本手册中不再一一举例。

BT-EN-MB系列型号众多,有单个、两个,四个Modbus RTU端口的产品可供选择,各种模块配置方式均可参考本手册中内容。

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	13910136425
(中国大陆)	
技术支持	<pre>support@beacongt.com</pre>
亚太区销售	asia@beacongt.com
北美区销售	usa@beacongt.com