1 Modbus/TCP 通讯概述

MODBUS/TCP 是简单的、中立厂商的用于管理和控制自动化设备的 MODBUS 系列通讯协议的派生产品，它覆盖了使用 TCP/IP 协议的“Intranet”和“Internet”环境中 MODBUS 报文的用途。协议的最通用用途是为诸如 PLC’s，I/O 模块，以及连接其它简单域总线或 I/O 模块的网关服务的。

MODBUS/TCP 使 MODBUS_RTU 协议运行于以太网，MODBUS/TCP 使用 TCP/IP 和以太网在站点间传送 MODBUS 报文，MODBUS TCP 结合了以太网物理网络和网络标准 TCP/IP 以及以 MODBUS 作为应用协议标准的数据表示方法。MODBUS/TCP 通信报文被封装于以太网 TCP/IP 数据包中。与传统的串口方式，MODBUS/TCP 插入一个标准的 MODBUS 报文到 TCP 报文中，不再带有数据校验和地址。

1．1 通讯所使用的以太网参考模型

Modbus/TCP 传输过程中使用了 TCP/IP 以太网参考模型的 5 层：

第一层：物理层，提供设备物理接口，与市售介质/网络适配器相兼容

第二层：数据链路层，格式化信号到源/目硬件址数据帧

第三层：网络层，实现带有 32 位 IP 址 IP 报文包

第四层：传输层，实现可靠性连接、传输、查错、重发、端口服务、传输调度

第五层：应用层，Modbus 协议报文.

1．2 通讯所使用的参考模型

Modbus 数据在 TCP/IP 以太网上传输，支持 Ethernet II 和 802.3 两种帧格式。Modbus TCP 数据帧包含报文头、功能代码和数据 3 部分，MBAP 报文头(Modbus Application Protocol、Modbus 应用协议)分 4 个控制域，共 7 个字节，如图 1 所示：

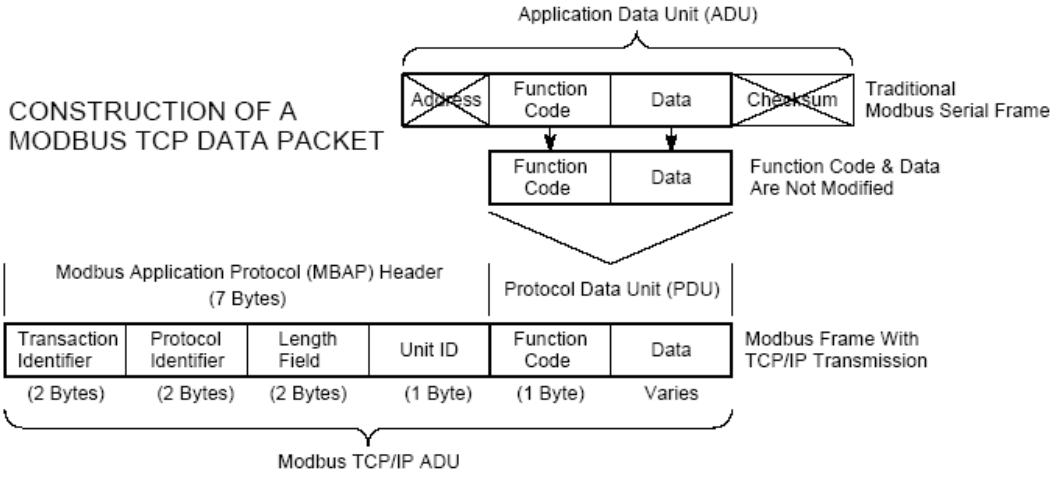


图 1：

MODBUS TCP 报文

由于使用以太网 TCP/IP 数据链路层的校验机制保证了数据的完整性，MODBUS/TCP 报文中不再带有数据校验”CHECKSUM”，原有报文中的“ADDRESS”也被“UNIT ID”替代而加在 MODBUS 应用协议报文头中。

1．3 通讯所使用的参考模型

在 Modbus 服务器中按缺省使用 Port 502 通信端口,在 Modus 客户器程序中可以设置任意通信端口，为避免与其他通讯协议的冲突一般建议使用 2000 以后的端口

1．4 Modbus/TCP 使用的功能代码

按照使用用途区分,共有 3 种类型分别为：

1) 公共功能代码：已定义好功能码，保证其唯一性，由 Modbus.org 认可；

2) 用户自定义功能代码有两组，分别为 65～72 和 100～110，无需认可，但不保证代码使用唯一性,如变为公共代码，需交 RFC 认可；

3) 保留功能代码，由某些公司使用某些传统设备代码，不可作为公共用途。

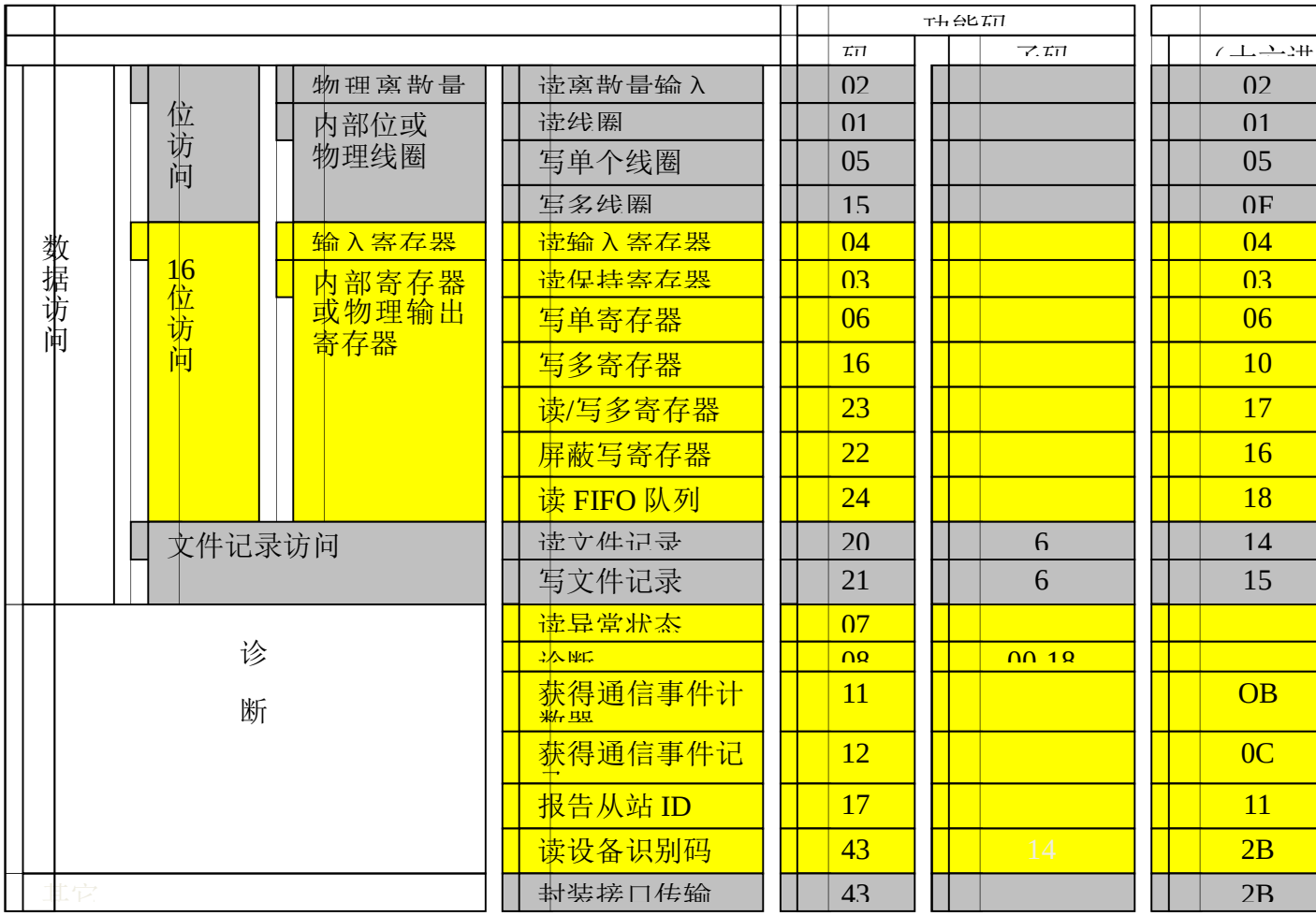
按照应用深浅，可分为 3 个类别

1) 类别 0,客户机/服务器最小可用子集：读多个保持寄存器(fc.3)；写多个保持寄存器(fc.16)。

2) 类别 1，可实现基本互易操作常用代码：读线圈(fc.1)；读开关量输入(fc.2)；读输入寄存器(fc.4)；写线圈(fc.5)；写单一寄存器(fc.6)。

3) 类别 2，用于人机界面、监控系统例行操作和数据传送功能：强制多个线圈(fc.15)；读通用寄存器(fc.20)；写通用寄存器(fc.21)；屏蔽写寄存器(fc.22)；读写寄存器(fc.23)

Modbus/TCP 使用的功能码对应列表如下图 2 所示：



数据访问	位访问	物理离散量	读离散量输入	02		02
		内部位或物理线圈	读线圈	01		01
			写单个线圈	05		05
			写多线圈	15		0F
	16 位访问	输入寄存器	读输入寄存器	04		04
		内部寄存器或物理输出寄存器	读保持寄存器	03		03
			写单寄存器	06		06
			写多寄存器	16		10
			读/写多寄存器	23		17
			屏蔽写寄存器	22		16
			读 FIFO 队列	24		18
	文件记录访问		读文件记录	20	6	14
			写文件记录	21	6	15
诊断			读异常状态	07		
			诊断	08	00-18	
			获得通信事件计数器	11		0B
			获得通信事件记	12		0C
			报告从站 ID	17		11
			读设备识别码	43	14	2B
其它			封装接口传输	43		2B

图 2：Modbus/TCP 所使用的功能码

1．5 Modbus/TCP 通讯应用举例

在读寄存器的过程中,以 Modbus TCP 请求报文为例,具体的数据传输过程如下:

1) Modbus/TCP 客户端，用 Connect()命令建立目标设备 TCP 502 端口连接数据通信过程

2) 准备 Modbus 报文，包括 7 个字节 MBAP 内请求

3) 使用 send()命令发送

4) 同一连接等待应答

5) 通过 recv()读报文，完成一次数据交换过程

6) 当通信任务结束时，关闭 TCP 连接，使服务器可以为其他服务

2 TIA 博途 V11 平台 S7-1200 Modbus/TCP 库函数概述

TIA 博途全集成自动化软件是未来西门子全集成自动化系列所有用于工程、编程和调试自动化设备和驱动系统的基础，从 TIA 博途软件 V11 SP1 开始，在 S7-1200 的库函数中免费嵌套了 Modbus/TCP 功能块库，客户可以利用该库函数完成与一个第三方设备的 Modbus/TCP 通讯，如下图 3 所示：

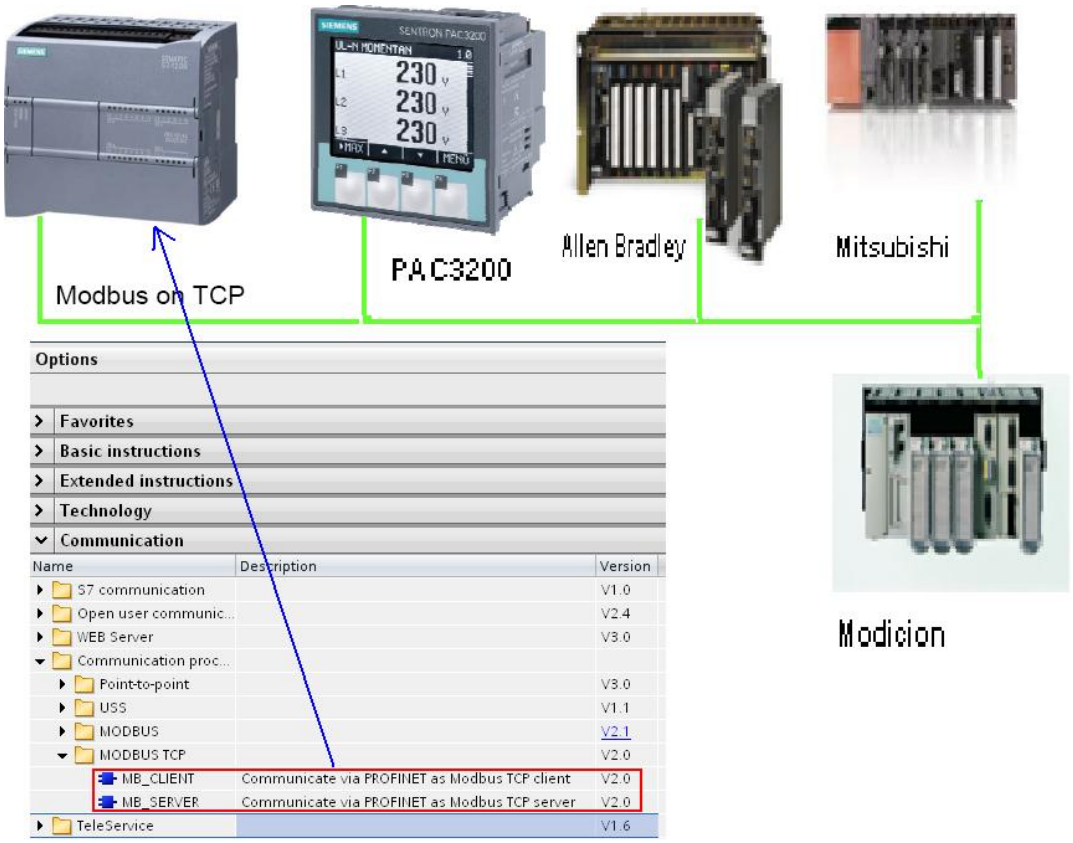


图 3：TIA 博途 V11 中 S7-1200 通过内嵌的 Modbus/TCP 库函数与第三方设备通讯

TIA 博途 V11 中内嵌的 S7-1200 Modbus/TCP 库函数具有如下特点：

1） 包含了 Server 和 Client 的库函数，同时在一个 S7-1200 中可以同时作为 Server 和 Client。

2） 通过 S7-1200 CPU 的集成 Profinet 接口进行通讯

3） 可以支持多个 Server 和 Client 的连接，但是需要注意以下几个方面：

- 最大支持 8 个连接(通过 Open IE 方式创建 TCP 连接)
- 每个连接必须调用一个“MB_Block”块
- 每个连接调用的“MB_Block”块中使用的连接 ID 和端口号必须唯一

下面将详细介绍如何使用该库函数配置 S7-1200 作为 Modbus/TCP 的 Server 和 Client 与第三方设备进行通讯。

3 测试例子中使用的硬件和软件

以下的配置实例中所使用到的硬件配置如下表 1 所示：

表 1：配置实例中用到的硬件列表配置

名称	数量	订货号
SIMATIC S7-1200, PM1207, 2,5A	1	6EP1332-1SH71
SIMATIC S7-1200 CPU1214C 固件版本 V2.1	1	6ES7214-1BD30-0XB0
以太网连接电缆	若干	
编程器兼 PC 软件测试机	1	

实例中用到的软件如下表 2 所示:

表 2：配置实例中用到的软件列表

名称	数量	订货号
SIMATIC Step7 Professional V11 SP2(1)	1	6ES7822-1AA01-0YA5
SIMATIC WinCC Professional V11 SP2	1	6AV2103-0XA01-0AH5
ModScan32 V4.0，安装在 PC 机模拟 Modbus/TCP 的 Client	1	可以从相关网站免费下载
Mbslave V4.3.0，安装 PC 机模拟 Modbus/TCP 的 Server	1	可以从相关网站免费下载

备注:

(1) SIMATIC Step7 V11 包含两个版本，分为 Step7 Basic V11 和 Step7 Professional V11， 对于 S7-1200 的组态仅需要 Step7 Basic V11 即可，本例中的 PC 机安装了 Step7 Professional V11，另外如果安装了 Step7 Professional V11，将直接嵌套安装了 WinCC Basic V11，能够用于组态与 S7-1200 匹配的一些 Basic Panel

4 在 TIA 博途 V11 中配置 S7-1200 为 Modbus/TCP Client

4．1 在 Step7 Professional V11 中组态 S7-1200

在 STEP7 Professional V11 中创建一个 S7-1200 的项目，本例中项目名为 S7-1200_Modbus_TCP_Client，切换到项目视图界面，插入一个 S7-1200 CPU，从硬件目录中插入 1214 CPU AC/DC/Rly，并选择固件版本为 V2.1 对其属性组态，如下图 4 所示

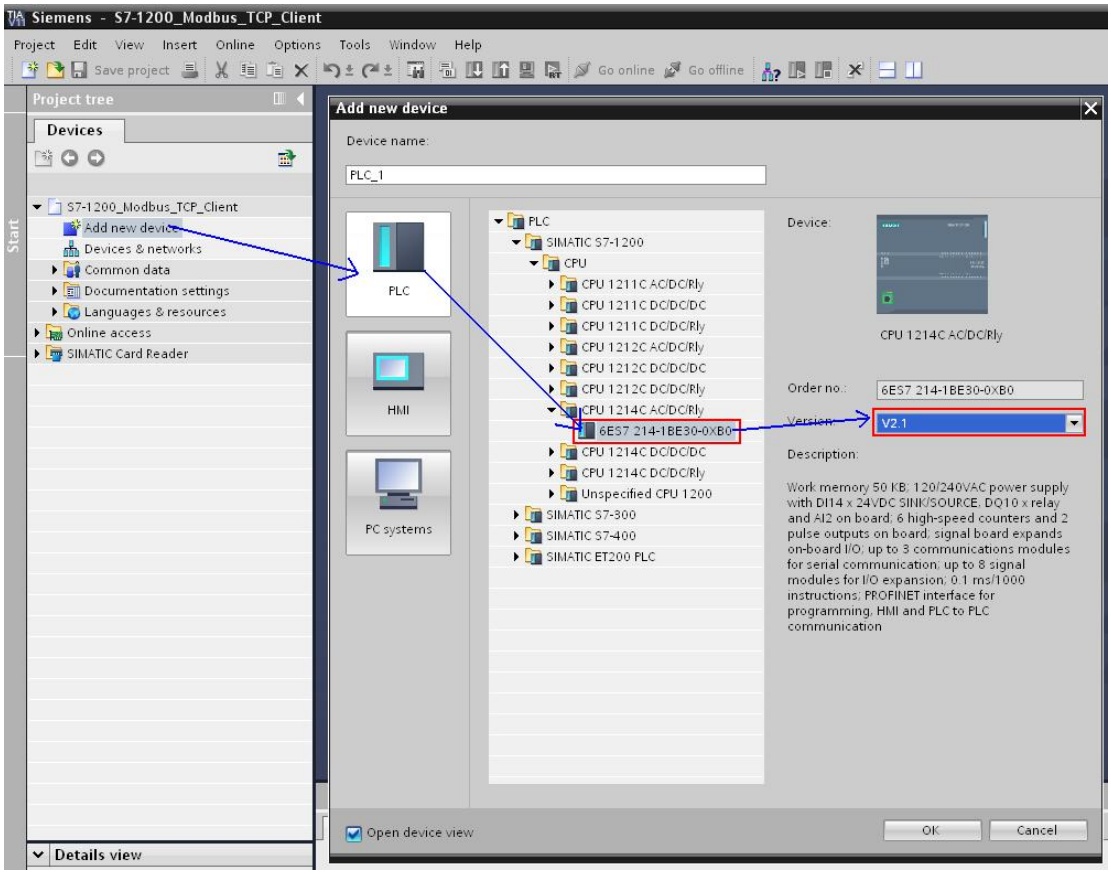


图 4：插入一个 S7-1200 项目

点击“ OK”后进入到“ Device view”界面，在“ Device view”界面中选中 CPU1214C 的以太网口，在下面的接口属性中设置 CPU1214C 的以太网接口的 IP 地址为 192.168.0.10，并将其连接到一个新建的以太网上，如下图 5 所示 ：

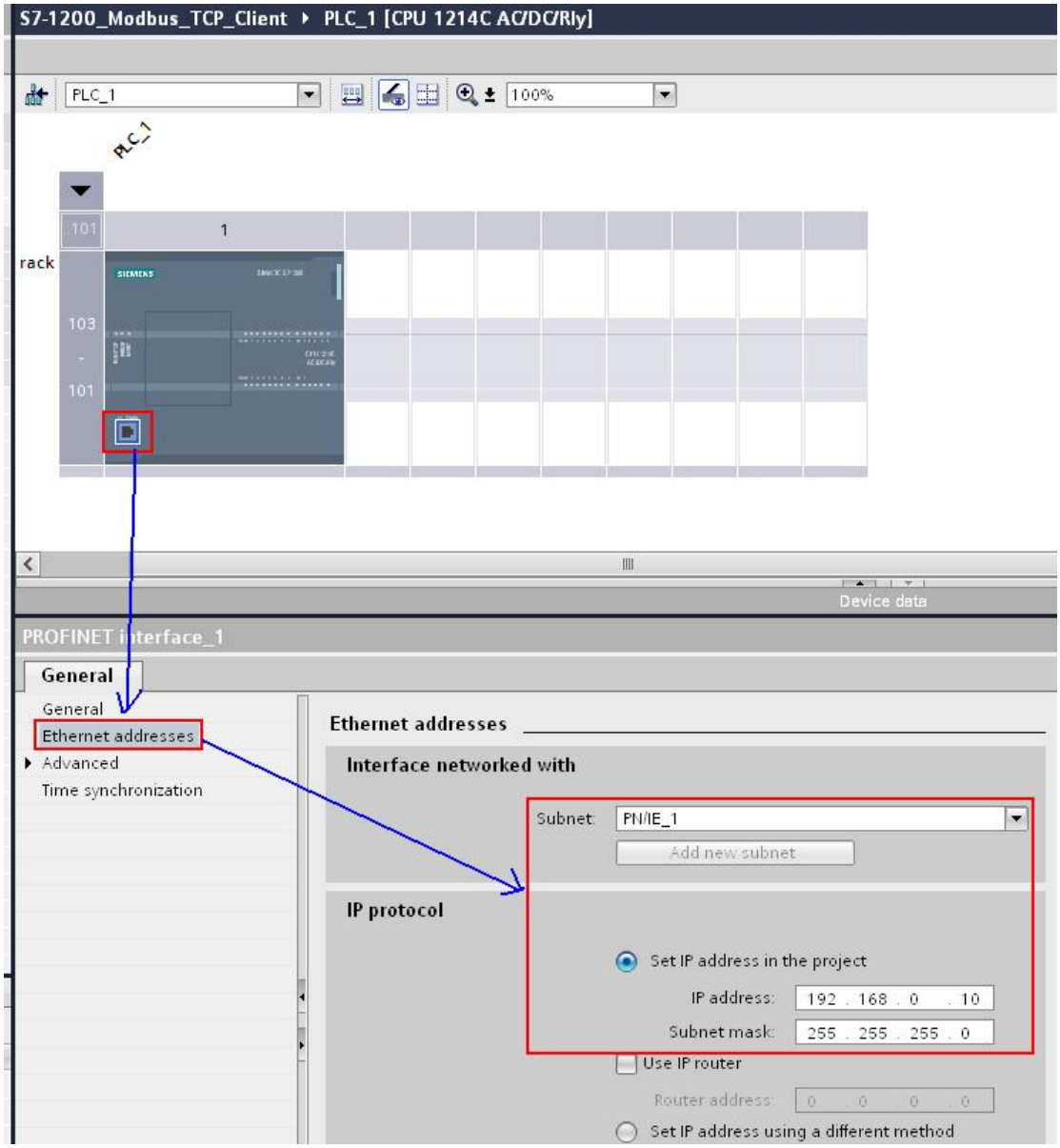


图 5：设置 S7-1200 CPU 以太网接口的 IP 地址

4．2 Step7 Professional V11 中调用 S7-1200 Modbus/TCP Client 库函数

在项目树中打开 S7-1200 项目的“ Program block”，之后打开主循环块 OB1，从右边的程序库函数中的“ Instruction->Communication->Communication processor->Modbus TCP”下将“ MB_CLIENT ” 拖拽到 OB1 界面中，之后将会自动弹出新的界面，设置调用 FB“ MB_CLIENT”功能块的背景数据块，本例中设置为 DB1，如下图 6 所示：

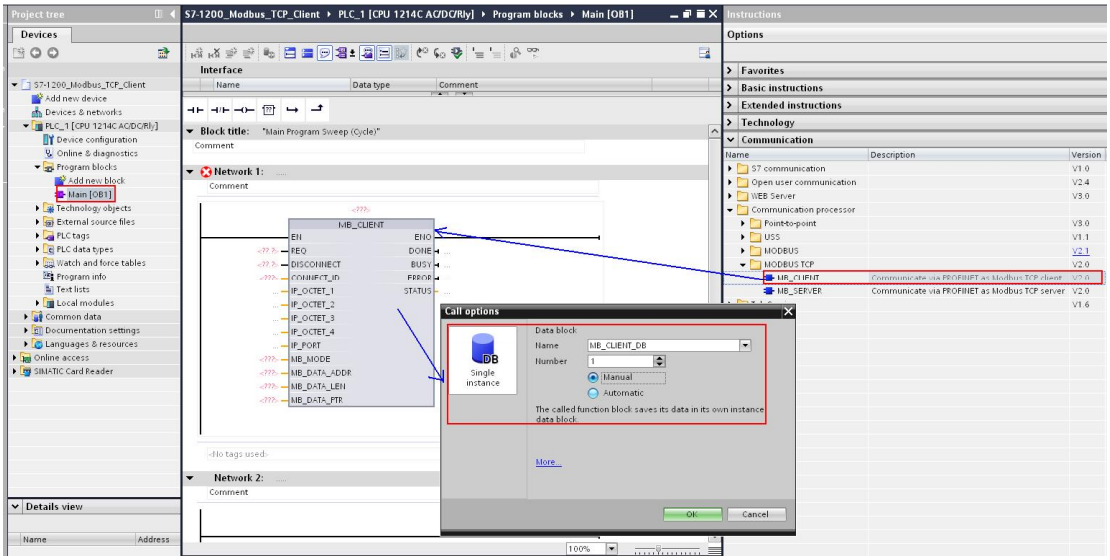


图 6：拖拽 Modbus/TCP 的“ MB_CLIENT”块到 OB1 中

之后需要设置“ MB_CLIENT”的各管脚参数，本例中将“ MB_CLIENT”功能块的控制管脚参数均放在 DB2(Client_Control)中，通讯测试数据区放在 DB3 中(Client_Data)，另外为了便于进行故障诊断，通常情况下可以将“ MB_CLIENT”功能块的“ STATUS”通过“ ERROR”信号输出进行缓存，如下图 7 所示：

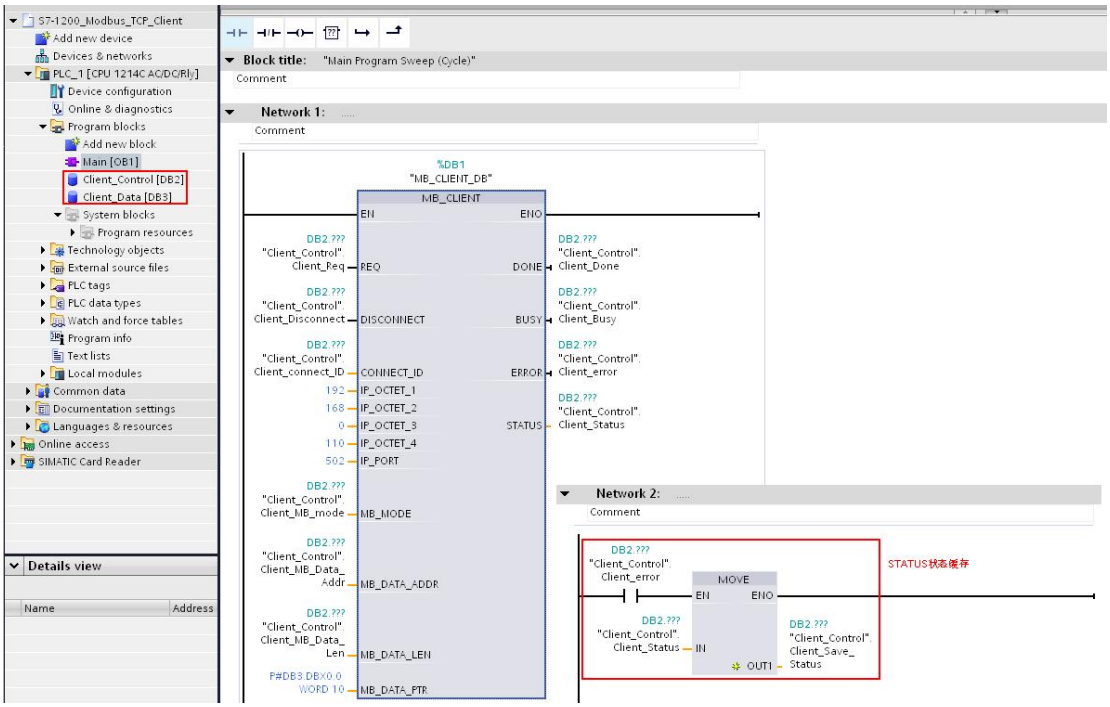


图 7：设置“ MB_CLIENT”功能块管脚

“ MB_CLIENT”功能块各管脚参数的详细含义如下表 3 所示：

参数	声明类型	数据类型	详细描述
REQ	Input	BOOL	向 Modbus/TCP Server 端连接请求，上升沿有效，本例中将参数放在 DB2 中
DISCONNECT	Input	BOOL	用于控制与 Modbus/TCP Server 端连接，其中 0 为保持连接；1 为断开连接，本例中将参数放在 DB2 中
CONNECT_ID	Input	WORD	连接 ID 号，每个连接的 ID 必须为 1，本例中将参数放在 DB2 中，并设置缺省值为 1
IP_OCTET_1	Input	BYTE	Modbus/TCP Server 的 IP 地址第 1 部分，本例中为 192
IP_OCTET_2	Input	BYTE	Modbus/TCP Server 的 IP 地址第 2 部分，本例中为 168
IP_OCTET_3	Input	BYTE	Modbus/TCP Server 的 IP 地址第 3 部分，本例中为 0
IP_OCTET_4	Input	BYTE	Modbus/TCP Server 的 IP 地址第 4 部分，本例中为 10
IP_PORT	Input	WORD	Server 端端口号，本例中为使用缺省端口号 502
MODE	Input	USINT	请求模式，包括读写及诊断，具体含义可以参考表 4，本例中将参数放在 DB2 中，并设置缺省值为 0
MB_DATA_ADDR	Input	UDINT	Modbus 起始地址，具体含义可以参考表 4，本例中将参数放在 DB2 中，并设置缺省值为 40001
MB_DATA_LEN	Input	UINT	请求的寄存器或位的长度，具体含义可以参考表 4，本例中将参数放在 DB2 中，并设置缺省值为 5
MB_DATA_PTR	Input	VARIANT	与 Modbus/TCP Server 进行通讯的数据存储区，支持指针，本例中将数据放在 DB3 中，并设置缺省值为 P#DB3.DBX0.0 WORD 10
DONE	Output	BOOL	当为 1 时表示完成一次作业，本例中将参数放在 DB2 中
BUSY	Output	BOOL	作业进程标志，0 表示当前没有任何作业，1 表示正在进行作业
ERROR	Output	BOOL	作业是否出错，0 表示正常，1 表示故障
STATUS	Output	WORD	作业状态字，用于故障诊断，具体可以参考表 5，本例中将参数放在 DB2 中

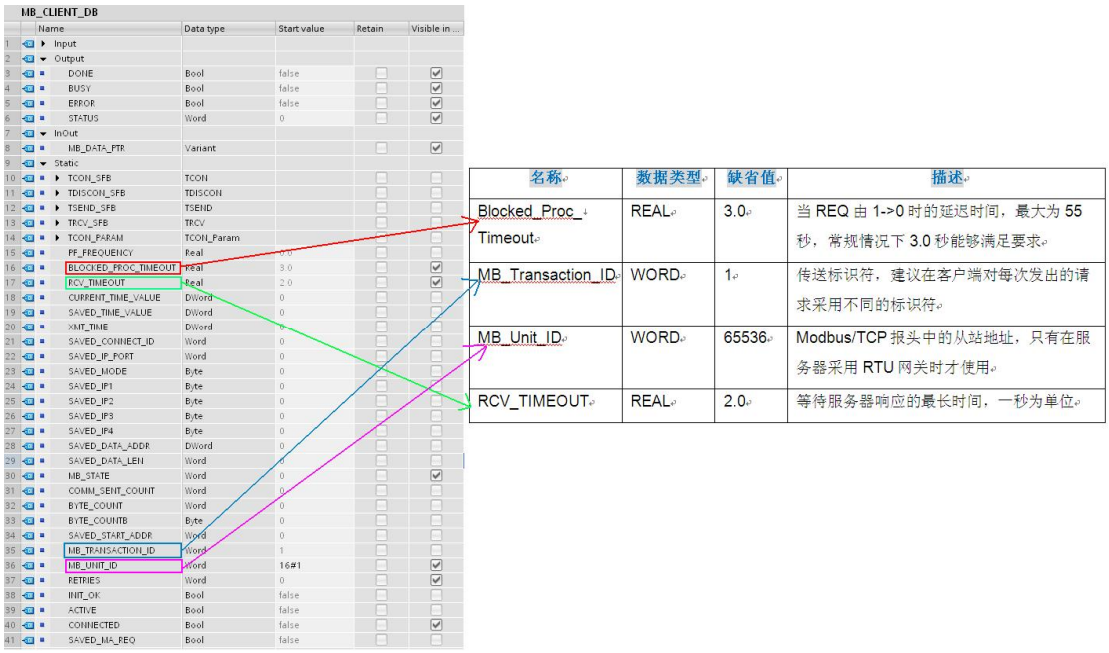
表 3：“ MB_CLIENT”功能块各管脚参数

从上述参数中可以看到，“ MB_CLIENT”功能块中并没有定义功能码的直接相关参数，具体功能码将由参数 MODE、MB_DATA_ADDR、MB_DATA_LEN 的组合共同定义，此外还包含了一些诊断功能，如下表 4 所示：

MB_MODE	Modbus 功能码	MB_DATA_LEN	功能和数据类型	MB_DATA_ADDR
0	01	1-2000	读线圈：1-2000Bit	1-9999
0	02	1-2000	读离散输入：1-2000Bits	10001-19999
0	03	1-125	读保持寄存器：1-125Word	40001-49999
0	04	1-125	读输入寄存器：1-125Word	30001-39999
1	05	1	写 1 个线圈	1-9999
1	06	1	写 1 各保持寄存器	40001-49999
1	15	2-1968	写多个线圈：2-1968Bit	1-9999
1	16	2-123	写多个保持寄存器：2-123 Word	40001-49999
2	15	1-1968	写一个或多个线圈：1-1968Bit	-
2	16	1-123	写一个或多个保持寄存器：1-123Word	-
11	11	0	获取 Modbus/TCP 服务器的状态和通信事件计数器： • 其中状态字表示了服务器的处理进程(0 表示无任务处理；0xFFFF 表示正在作业) • 通信事件计数器主要表示服务器成功响应客户端请求的次数 此模式下参数 MB_DATA_ADDR、MB_DATA_LEN 无任何意义	-
80	08	1	诊断，使用错误码 0X0000 来获取服务器的状态(用于循环检测，服务器返回请求)，1 个 Word	-
81	08	1	诊断，使用错误码 0X000A 来复位服务器的通信事件计数器，1 个 Word	-
3-10 12-79 82-255	-	-	保留	

表 4：功能块“ MB_CLIENT”功能码的表示

需要注意的是，除了功能块“ MB_CLIENT”管脚参数外，在其背景数据块中也有一部分参数需要根据项目的实际情况进行调整，如下图 8 所示：



名称	数据类型	缺省值	描述
Blocked_Proc_Timeout	REAL	3.0	当 REQ 由 1->0 时的延迟时间，最大为 55 秒，常规情况下 3.0 秒能够满足要求。
MB_Transaction_ID	WORD	1	传送标识符，建议在客户端对每次发出的请求采用不同的标识符。
MB_Unit_ID	WORD	65536	Modbus/TCP 报头中的从站地址，只有在服务器采用 RTU 网关时才使用。
RCV_TIMEOUT	REAL	2.0	等待服务器响应的最长时间，一秒为单位。

图 8：功能块“ MB_CLIENT”背景数据块参数设置

至此 S7-1200 功能块“ MB_CLIENT”库函数的所有设置完成，下载所有的硬件组态和程序到 CPU1214C 中。

4．3 配置 Modbus Slave 作为 Modbus/TCP Server

打开 Modbus Slave 软件，在 Connection->connection 中打开连接属性对话框，连接接口选择“Modbus TCP/IP”，TCP/IP Server Port 为本地服务器的端口 502，并可以勾选“Ignore Unit ID”选项，如下图 9 所示：

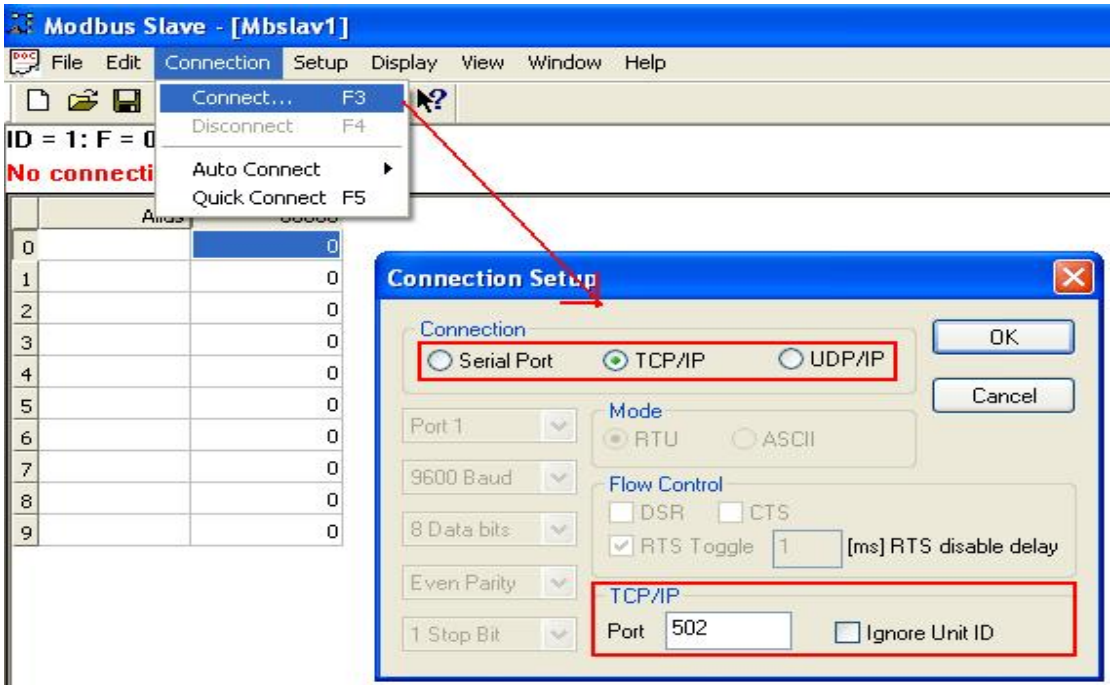


图 9：Modbus Slave 连接设置

（说明-“ Ignore Unit ID ”选项的含义如下：

Ignore Unit ID-在一些厂商的 PLC 的程序或网关中可能会用到 Unit ID 以指定处理类型）

在 Modbus Slave 的“Set up->Slave Definition“中可以设置功能码、起始地址、长度、显示的列数、数据显示格式及响应时间等，并可勾选“Hide Alias Columns”、“PLC Adresses(Base1)”、“Insert CRC/LRC error”、“Skip response”、“Return Exception 06,Busy”选项，如下图所示：

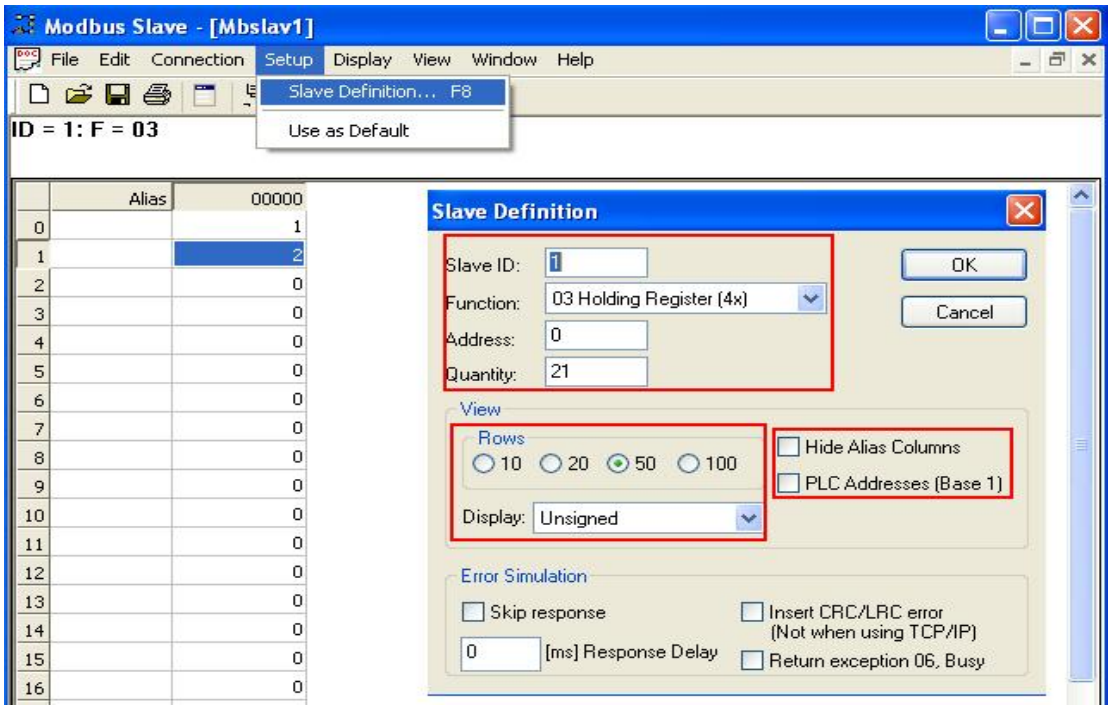


图 10：Modbus Slave 数据定义

(说明-各勾选选项的含义如下：

Hide Alias Columns –隐藏注释选项

PLC Addresses(Base1) - 选择寄存器地址是基于 PLC 地址编排(1..65535)还是基于协议编排(0-65535)

Insert CRC/LRC error - 选择是否进行 CRC/LRC 错误校验

Skip response – 选择是否忽略报文丢失响应

Return Exception 06,Busy – 选择是否返回 Slave 忙信号)

Modsbus Slave 支持读写线圈(fc.1/5)、读开关量输入(fc.2)、读输入寄存器(fc.4)、读写多个保持寄存器(fc.3/16)等多个功能码。

4．4 通讯测试

由于功能块“MB_CLIENT”和 Modsbus Slave 支持的功能均比较多，下面以 FC03/16 读取保持寄存器为例来介绍详细的通讯过程。

4．4．1 FC03 功能码(读取服务器端保存寄存器)测试

设置功能块“MB_CLIENT”和 Modsbus Slave 在起始地址和长度相对应，将功能块“MB_CLIENT”的 Mode 设置为 0(读模式)，在 TIA 博途中新建一个变量监控表，使能功能块“MB_CLIENT”的 REQ 参数后可以看到通讯已经建立，客户端能够从服务器中读取数据，如下图 11 所示：

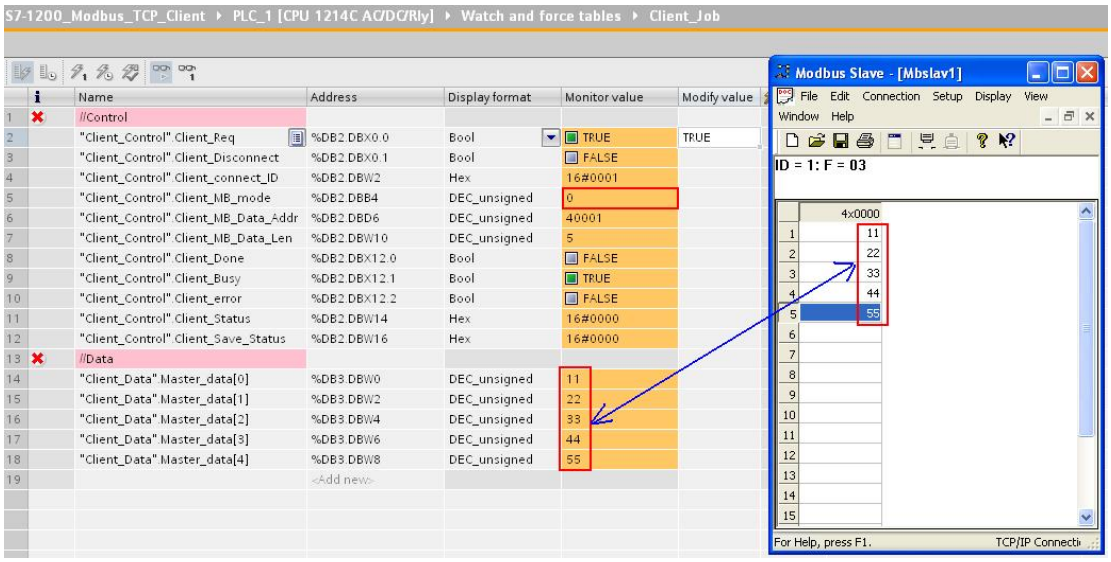


图 11：功能块“MB_CLIENT”读保持寄存器功能测试

4．4．2 FC16 功能码(写服务器端保存寄存器)测试

同样设置功能块“MB_CLIENT”和 Modsbus Slave 在起始地址和长度相对应，将功能块“MB_CLIENT”的 Mode 设置为 1(写模式)，使能功能块“MB_CLIENT”的 REQ 参数后可以看到通讯已经建立，客户端能够向服务器中写数据，如下图 12 所示：

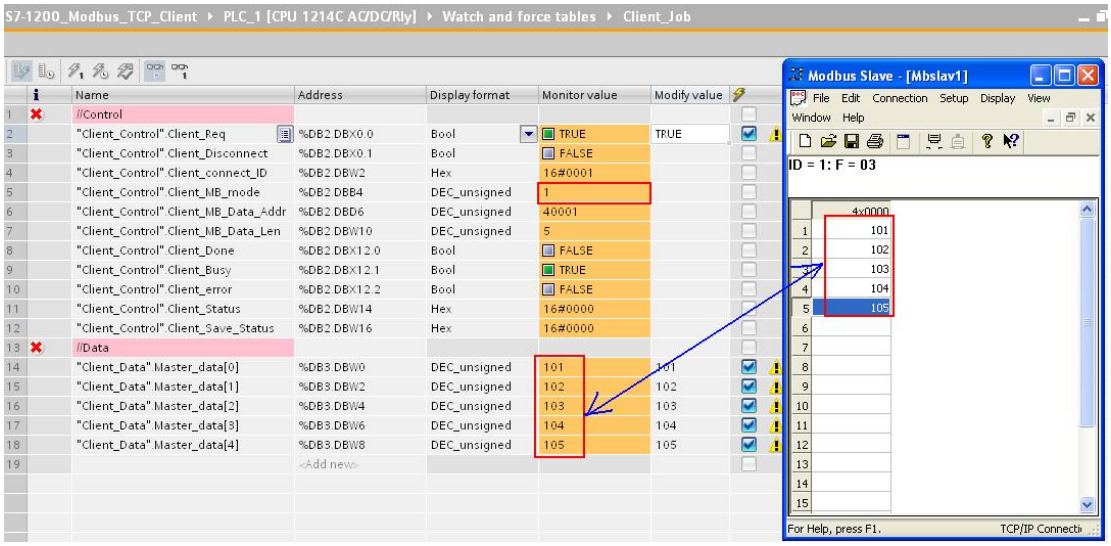


图 12：功能块“ MB_CLIENT”写保持寄存器功能测试

4．5 通讯诊断相关注意事项

当通讯故障时可以通过功能块“ MB_CLIENT”的 STATUS 输出状态字代码来进行故障诊断，详细的故障代码信息如下表 5 所示：

错误类型	STATUS(W#16#)	返回给客户端的异常码(W#16#)	描述
应用层协议错误	8381	01	功能码不支持
	8382	03	请求的数据长度错误
	8383	02	参数 MB_DATA_PTR 设置错误
	8384	03	数据错误，如数据格式不支持等
	8385	03	服务器不支持诊断功能(FC08)
参数设置	80C8	不相关	服务器无响应，检查连接设置

错误	8188		MODE 设置值无效
	818A		MB_DATA_LEN 参数设置无效
	818B		参数 MB_DATA_PTR 设置的指针无效
	818C		参数 MB_DATA_PTR 设置的数据块定义为“仅符号访问”，必须设置成标准访问格式
	8200		Modbus 请求正通过端口处理
	8380		Modbus 客户端发送的报文格式错误
	8387		连接 ID 重复，不同的连接必须使用不同的连接 ID
	8388		当使用 FC15,16 功能码时，服务器响应了一个与客户端请求不匹配的报文

表 5：功能块“ MB_CLIENT ”通讯故障代码对应表

另外对于多个客户端的连接请求，必须遵循一下规则：

1) 不同的功能块“ MB_CLIENT”的背景 DB 块必须不同
2) 功能块“ MB_CLIENT”请求的服务器端的 IP 地址必须定义
3) 服务器的端口号可以根据服务器的具体情况来定义

5 在 TIA 博途 V11 中配置 S7-1200 为 Modbus/TCP Server

5．1 在 Step7 Professional V11 中组态 S7-1200

该部分内容与 4.1 章节相同，请参考 V4.1 章节

5．2 Step7 Professional V11 中 S7-1200 Modbus/TCP Server 库函数的使用

在项目树中打开 S7-1200 项目的“ Program block”，之后打开主循环块 OB1，从右边的程序库函数中的“ Instruction->Communication->Communication processor->Modbus TCP”下将“ MB_SERVER ”拖拽到 OB1 界面中，之后将会自动弹出界面，设置调用 FB“ MB_SERVER”功能块的背景数据块，本例中设置为 DB1，如下图 13 所示：

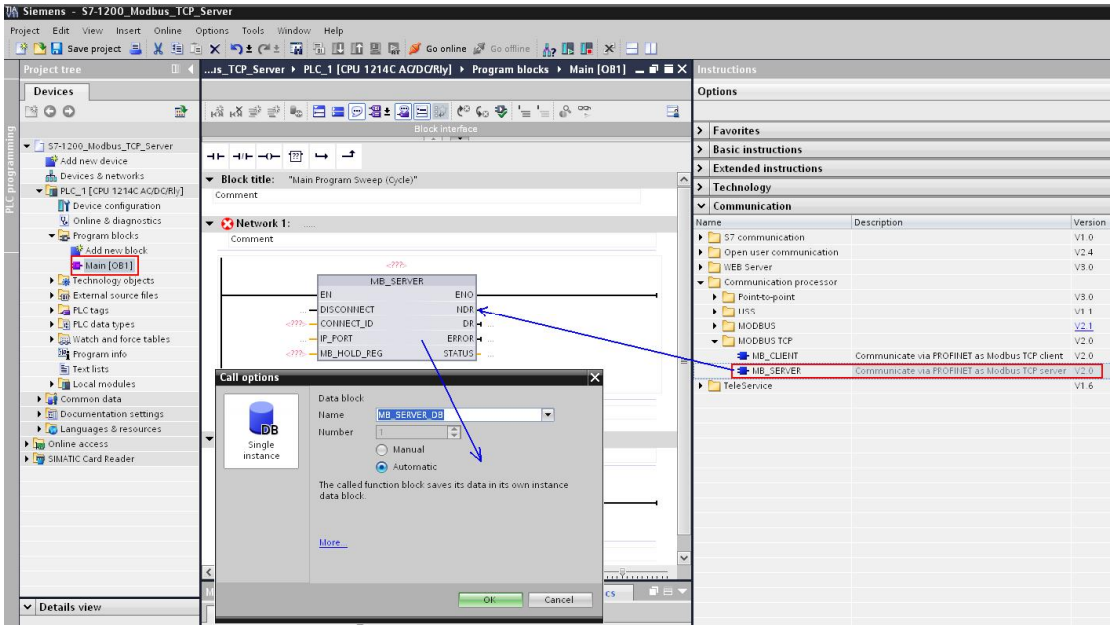


图 13：拖拽 Modbus/TCP 的“ MB_SERVER”块到 OB1 中

之后需要设置“ MB_SERVER”的各管脚参数，本例中将“ MB_SERVER”功能块的控制管脚参数均放在 DB2(Server_Control)中，通讯测试数据区放在 DB3 中(Server_Data)，另外为了便于进行故障诊断，通常情况下可以将“ MB_SERVER”功能块的“ STATUS”通过“ ERROR”信号输出进行缓存，如下图 14 所示：

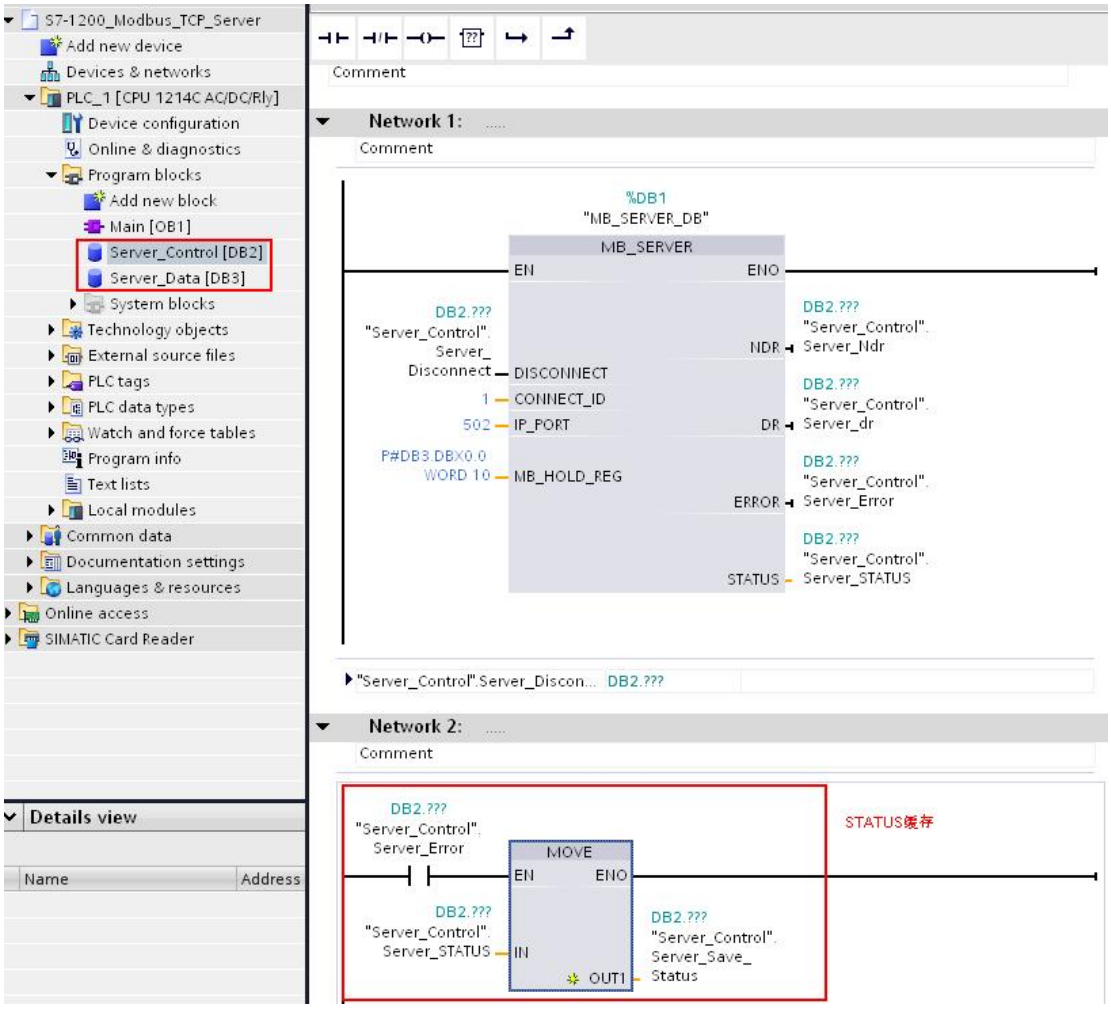


图 14：图 7：设置“ MB_SERVER”功能块管脚

“ MB_SERVER”功能块各管脚参数的详细含义如下表 6 所示：

参数	声明类型	数据类型	详细描述
DISCONNECT	Input	BOOL	用于控制与 Modbus/TCP CLient 端连接，其中 0 为保持连接；1 为断开连接，本例中将参数放在 DB2 中
CONNECT_ID	Input	UINT	连接 ID 号，每个连接的 ID 必须为 1，本例中将参数放在 DB2 中，并设置缺省值为 1
IP_PORT	Input	UINT	S7-1200 作为 Server 端使用的端口号，本例中为使用缺省值为 502
MB_HOLD_REG	Input	VARIANT	与 Modbus/TCP Client 进行通讯的数据存储区，支持指针，该数据区仅用于保持寄存器的数据通讯，支持功能码 FC03(读)、FC06(写)和 FC16(读)，本例中将数据放在 DB3 中，并设置缺省值为 P#DB3.DBX0.0 WORD 10

NDR	Output	BOOL	客户端写新数据状态指示，其中 0 表示没有新数据，1 表示有新数据写入，本例中将参数放在 DB2 中
DR	Output	BOOL	客户端读新数据状态指示，其中 0 表示没有新数据，1 表示新数据读取，本例中将参数放在 DB2 中
ERROR	Output	BOOL	作业是否出错，0 表示正常，1 表示故障，本例中将参数放在 DB2 中
STATUS	Output	WORD	作业状态字，用于故障诊断，具体可以参考表 8，本例中将参数放在 DB2 中

表 6：“ MB_SERVER”功能块各管脚参数

需要说明的是，与“ MB_Client”功能块不同，“ MB_SERVER”功能块所支持的各功能码是放在不同的区域，例如 FC03/06/16 功能码的数据区将存放在管脚参数“ MB_HOLD_REG”中，而 FC01/02/04/05/15 功能码的数据区将直接映射到 S7-1200 的过程映像区中，具体如下表 7 所示：

Modbus 功能				**S7-1200 地址区**	
功能码	功能	数据类型	地址区域	数据区	CPU 地址
01	读线圈	输出(Bit)	1-8192	过程映像输出	Q0.0-Q1023.7
02	读离散输入	输入(Bit)	10001-18192	过程映像输入	I0.0-I1023.7
04	读输入寄存器	输入(WORD)	30001-30512	过程映像输入	IW0-IW1022
05	写单个线圈	输出(Bit)	1-8192	过程映像输出	Q0.0-Q1023.7
15	写多个线圈	输出(Bit)	1-8192	过程映像输出	Q0.0-Q1023.7
03	读保持寄存器	输出(WORD)	40001-49999	DB 块	
06	写单个保持寄存器	输出(WORD)	40001-49999	DB 块	
16	写多个保持寄存器	输出(WORD)	40001-49999	DB 块	
08	诊断，带子码 0x0000，用于服务器回波测试			-	-
08	诊断，带子码 0x000A，诊断，使用错误码 0X000A 来复位服务器的通信事件计数器，1 个 Word			-	-
11	诊断，获取 Modbus/TCP 服务器的状态和通信事件计数器： ·其中服务器用一个内部状态字表示了通信事件计数器，用于指示服务器成功响应客户端请求的次数			-	-

表 7：“ MB_SERVER”功能块所支持的功能码

至此 S7-1200 功能块“ MB_SERVER”库函数的所有设置完成，下载所有的硬件组态和程序到 CPU1214C 中。

5．3 配置 Modscan32 作为 Modbus/TCP Client

打开 Modscan32 软件，在“Connection-connection”中打开连接属性对话框，连接接口选择“ Remote TCP/IP Server”， IP Adress 分别填入 S7-1200 CPU 的 IP 地址 192.168.0.10,Server Port 为远程服务器 CPU 的端口 502，在协议的选择对话框中可以定义传输模式、通讯超时响应时间，报文发送间隔及允许写多个保持寄存器等，这里分别保持缺省设置即可，如下图 15 所示：

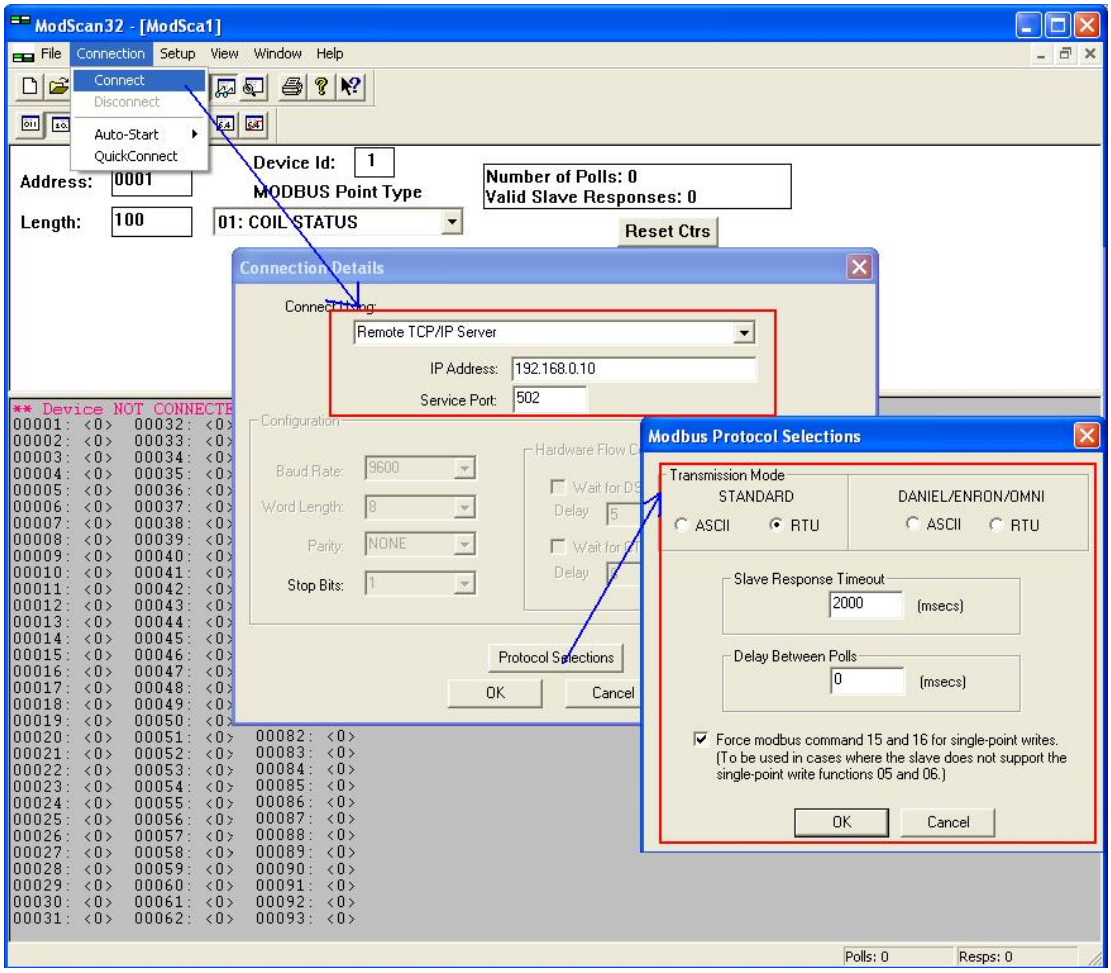


图 15：对应 TCP 通讯的 Modscan32 连接窗口

5．4 通讯测试

由于功能块“ MB_SERVER”和 Modscan32 支持的功能均比较多，下面对于读功能测试中将以 FC01(读线圈)、FC03(读保持寄存器)；写功能以 FC15(写多个线圈)、FC16(写多个保持寄存器)为例来阐述测试过程。

5．4．1 读功能 FC01(读线圈)、FC03(读保持寄存器)测试

在 Modscan32 的“Set up->Data Definition“中设置数据扫描周期、寄存器连接类型、起始地址、长度等，本例中分别设置测试线圈和保持寄存器测试，如下图 16 所示：

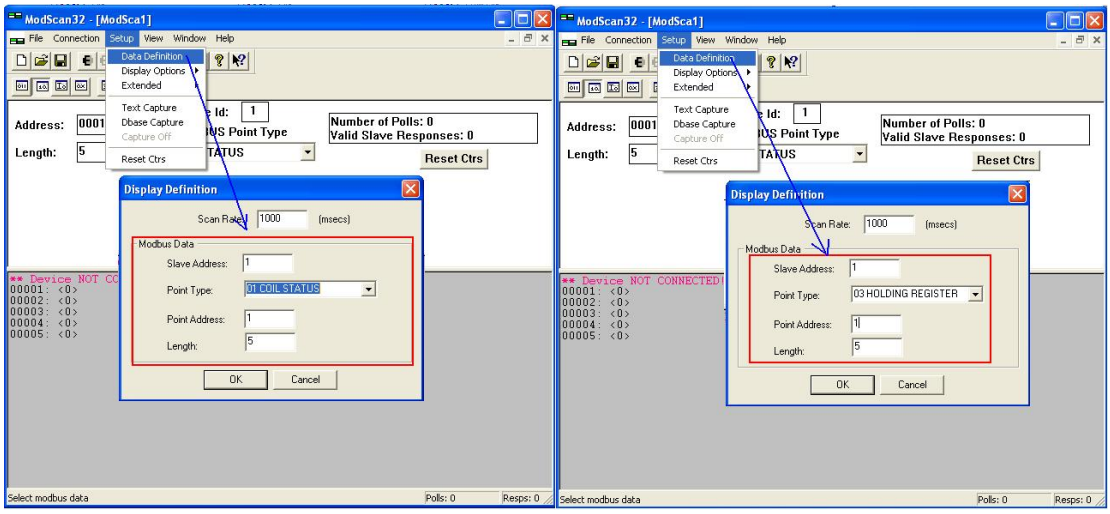


图 16:Modscan32 中 Modbus 数据参数定义

之后在 Modsan32 中使能建立连接，可以看到在 Modscan32 能够和 S7-1200 建立通讯，对于线圈，Modscan32 能够直接读取 S7-1200 的输出状态 Q0.0-Q0.4；对于保持寄存器，Modscan32 能够直接读取 DB3 中的数据，如图 17 所示：

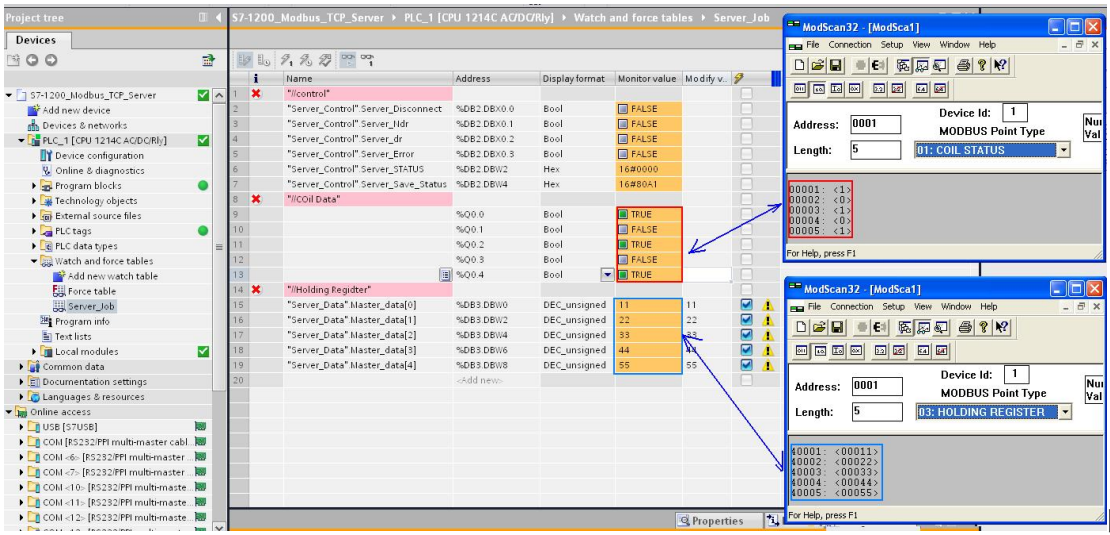


图 17：读功能测试中将以 FC01(读线圈)、FC03(读保持寄存器)测试

5．4．2 写功能 FC15(写多个线圈)、FC16(写多个保持寄存器)测试

在 Modscan32 中使能连接与 S7-1200 建立通讯后，可以看到对于线圈，Modscan32 能够直接将线圈状态写入 Q0.0-Q0.4 中；对于寄存器，Modscan32 能偶将数据直接写入 DB3 中，如图 18 所示：

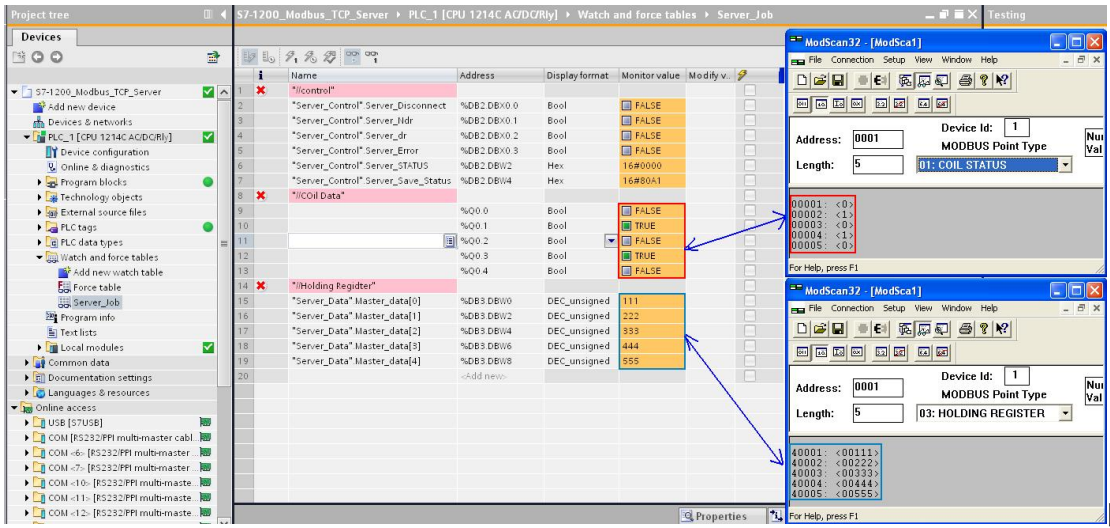


图 18：写功能以 FC15(写多个线圈)、FC16(写多个保持寄存器)测试

5．5 通讯诊断相关注意事项

当通讯故障时可以通过功能块“ MB_SERVER”的 STATUS 输出状态字代码来进行故障诊断，详细的故障代码信息如下表 8 所示：

错误类型	STATUS(	返回给客	描述

	W#16#)	户端的异常码 (W#16#)	
应用层协议错误	8187	无响应	参数“ MB_HOLD_REG”设置无效，数据区过短
	818C	无响应	1 参数“ MB_HOLD_REG”所匹配的 DB 块不能定义为仅符号访问”，必须设置成标准访问格式 2 “ MB_SERVER”功能块组织响应报文超时(超过 55 秒)
	8381	01	功能码不支持
	8382	03	请求的数据长度错误
	8383	02	客户端请求的数据地址超出了“ MB_HOLD_REG”设置的范围
	8384	03	数据错误，如数据格式不支持等
	8385	03	服务器不支持诊断功能(仅支持功能码 FC08)

表 8：功能块“ MB_SERVER ”通讯故障代码对应表

本文網址：http://support.automation.siemens.com/CN/view/zh/77322990