SIEMENS

操作指南 • 10 月 2015 年

# 基于 S7-1200 CPU 集成 PN 口的 ModbusTCP 通信快速入门

**https://support.industry.siemens.com/cs/cn/zh/view/81015512**

# 目录

# 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,显而易见,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。

## 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 Modbus TCP 数据帧

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(MBAP、Modbus Application Protocol、Modbus 应用协议)分 4 个域，共 7 个字节。

## 1.3 Modbus TCP 使用的通讯资源端口号

在 Moodbus 服务器中按缺省协议使用 Port 502 通信端口,在 Modus 客户器程序中设置任意通信端口，为避免与其他通讯协议的冲突一般建议 2000 开始可以使用。

## 1.4 Modbus TCP 使用的功能代码

按照使用的通途区分,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)。

## 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 客户端实况，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) 同 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 SIMATIC S7-1200 Modbus TCP 通讯概述

S7-1200 CPU 从 Firmware V1.0.2 开始，软件 STEP7 V11 SP1 版本开始，可以直接调用 Modbus TCP 的库指令“MB_CLIENT”和“MB_SERVER”使用实现 Modbus TCP 通信功能，如下图 2-1 所示：

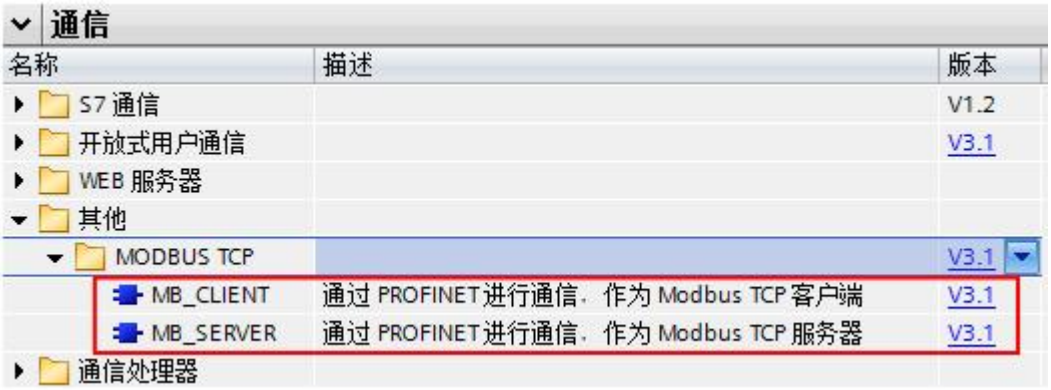



图 2-1 TIA Portal 中包含的 ModbusTCP 块库

下面将分别介绍如何配置 S7-1200 为 Modbus/TCP 的 Server，Client 与通信伙伴建立通信，测试例程中用到的软硬件如下表 1、2 所示：

| 名称 | 数量 | 订货号 |
|---|---|---|
| SIMATIC CPU1215C (固件 V3.0) | 1 | 6ES7 215-1AG31-0XB0 |
| 网线 | 若干 | |
| 编程器兼软件测试机 | 1 | |

表 1 例程中用到的硬件列表

| 名称 | 订货号 |
|---|---|
| SIMATIC STEP7 Prossional V13 | 6ES7 822-1AA01-0YA5 |
| Modscan32 用于在 PC 中模拟 Modbus Client | |
| Modsim32 用于在 PC 中模拟 Modbus Server | |

表 2 例程中用到的软件列表

# 3 配置 S7-1200 CPU 作为 Modbus TCP Server 与通信伙伴建立通讯

打开 TIA Portal V13 软件，新建一个项目，在项目中添加 CPU1215C，为集成的 PN 接口新建一个子网并设置 IP 地址，本例中为“ 192.168.70.102”，如下图 3-1 所示：

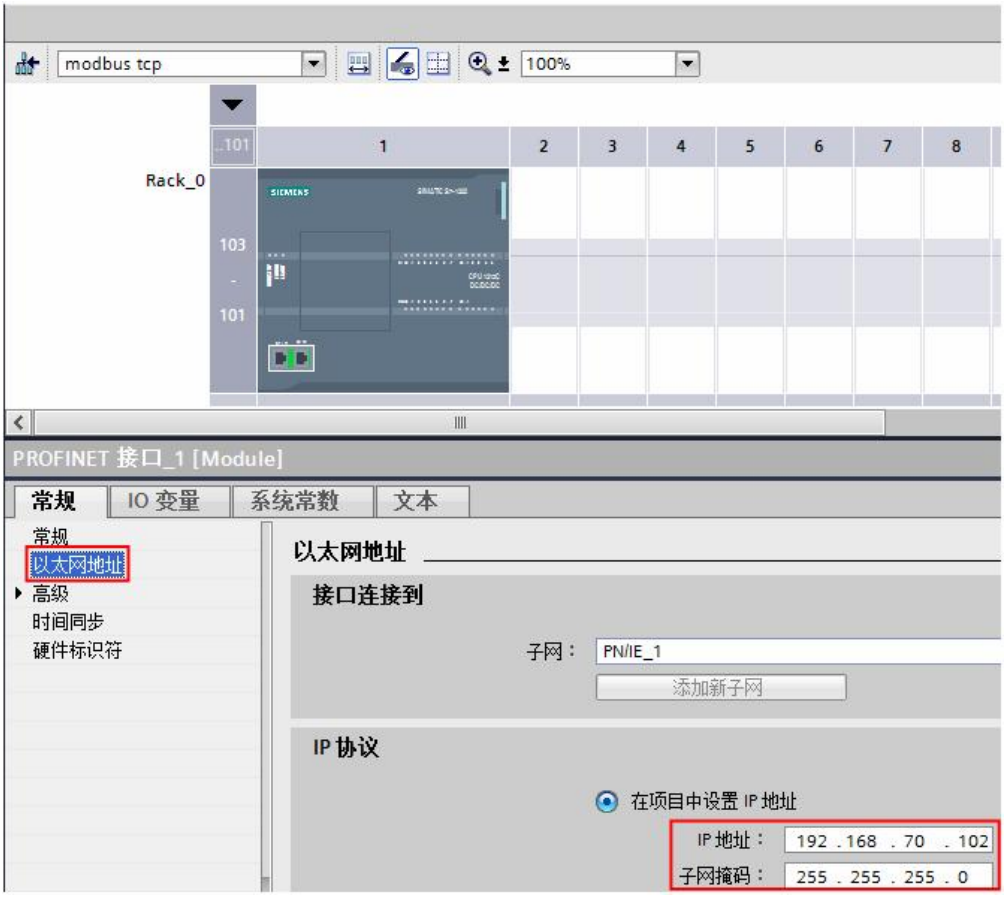



图 3-1 新建一个 S7-1200 项目并配置 IP 地址

在 CPU1215C 的 OB1 组织块中添加 Modbus TCP Server 功能块“ MB_SERVER”，软件将提示会为该 FB 块增加一个背景数据块，本例中为 DB1“ MB_SERVER_DB”，如下图 3-2 所示：

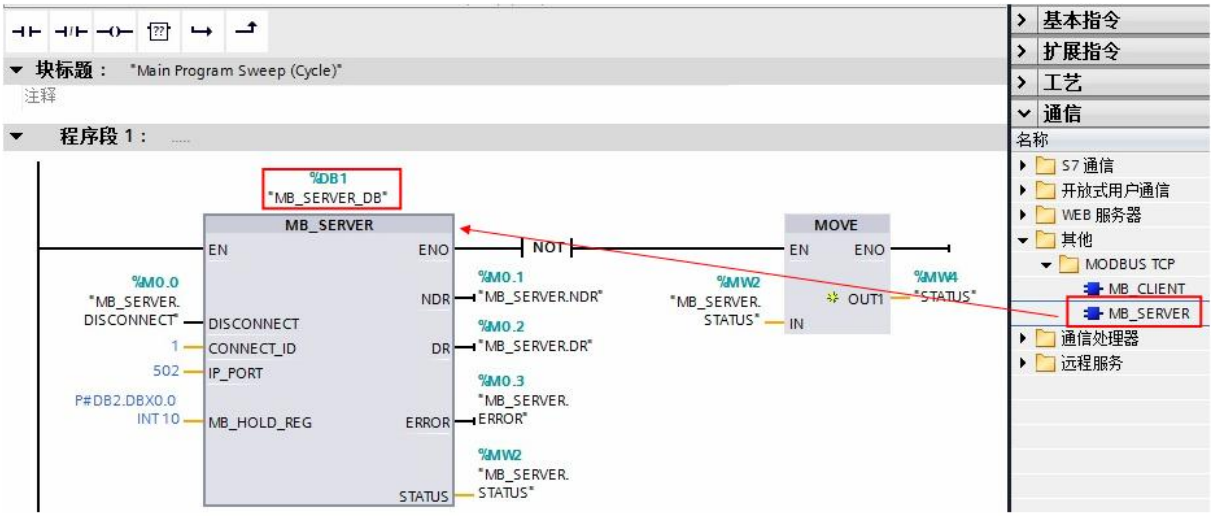
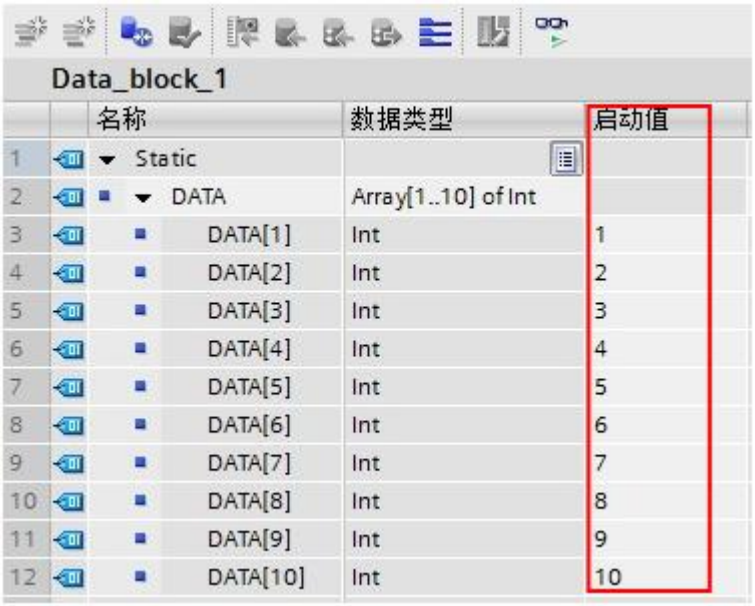



图 3-2 添加“ MB_SERVER”功能块

创建一个全局数据块用于匹配功能块“ MB_SERVER”的管脚参数“ MB_HOLD_REG”，本例中创建数据块 DB2“ Data_block_1”，用于存储保持寄存器的通信数据，并填写初始值，如下图 3-3 所示：



Data_block_1

| | | 名称 | 数据类型 | 启动值 |
|---|---|---|---|---|
| 1 | | Static | | |
| 2 | | DATA | Array[1..10] of Int | |
| 3 | | DATA[1] | Int | 1 |
| 4 | | DATA[2] | Int | 2 |
| 5 | | DATA[3] | Int | 3 |
| 6 | | DATA[4] | Int | 4 |
| 7 | | DATA[5] | Int | 5 |
| 8 | | DATA[6] | Int | 6 |
| 9 | | DATA[7] | Int | 7 |
| 10 | | DATA[8] | Int | 8 |
| 11 | | DATA[9] | Int | 9 |
| 12 | | DATA[10] | Int | 10 |

图 3-3 创建数据块 DB2

需要注意的是该数据块必须为非优化数据块(支持绝对寻址)，在该数据块的“属性”中不勾选“优化的块访问”选项，如下图 3-4 所示：

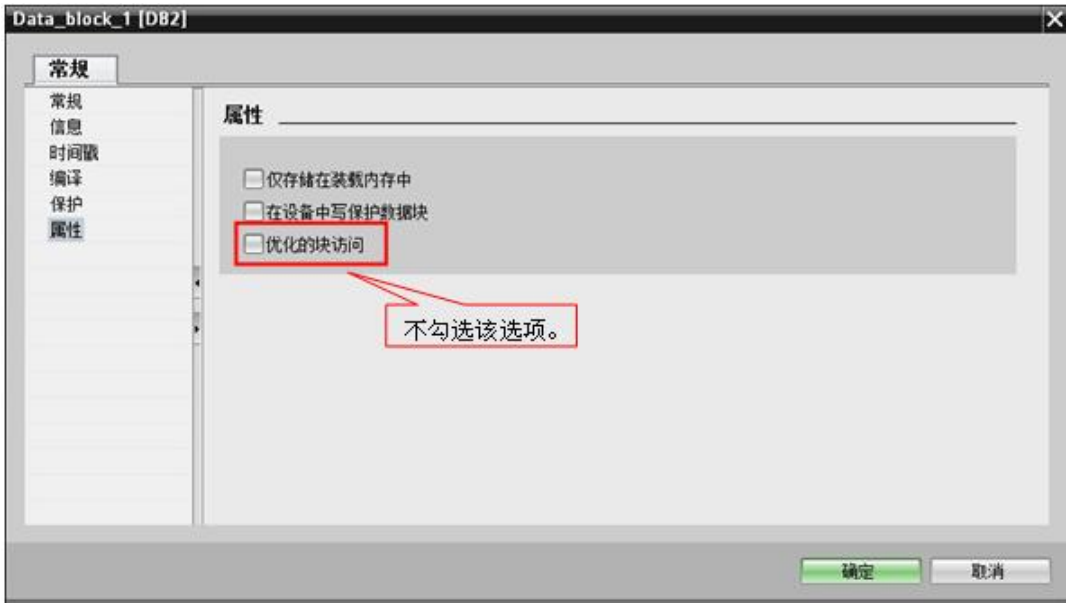



图 3-4 修改 DB 块属性

功能块“ MB_SERVER”的其它管脚参数如下表 3 所示：

| “ MB_SERVER ”的管脚参数 | 管脚声明 | 数据类型 | 含义 |
|---|---|---|---|
| DISCONNECT | 输入 | BOOL | 0: 且连接不存在时，则可启动建立被动连接。<br>1: 且连接存在时，则断开连接。 |
| CONNECT_ID | 输入 | Uint | 唯一标识 PLC 中的每个连接。 |
| IP_PORT | 输入 | Uint | 默认值=502：IP 端口号，将监视该端口是否有来自 Modbus 客户端的连接请求。 |
| MB_HOLD_REG | 输入/输出 | Variant | 指向 MB_SERVER Modbus 保持寄存器的指针：必须是一个标准的全局 DB 或 M 存储区地址。 |
| NDR | 输出 | Bool | 0: 没有新数据<br>1: 从 Modbus 客户端写入的新数据 |
| DR | 输出 | Bool | 0: 没有读取数据<br>1: 从 Modbus 客户端读取的数据 |
| ERROR | 输出 | Bool | MB_SERVER 执行因错误而终止后，ERROR 位将保持为 TRUE 一个扫描周期时间。 |
| STATUS | 输出 | Word | 通信状态信息，用于诊断；STATUS 参数中的错误代码值仅在 ERROR = TRUE 的一个循环周期内有效。 |

表 3 功能块“ MB_SERVER”的其它管脚参数

上面提到保持寄存器是由功能块“ MB_SERVER”的管脚参数“ MB_HOLD_REG”关联，对于其它数据类型，如线圈、离散输入、模拟量输入等通过功能块均已经与 S7-1200 的过程映像区进行了映射，其映射地址对应如下图 3-5 所示：

| Modbus 功能 | | | | | | S7-1200 | |
|---|---|---|---|---|---|---|---|
| 代码 | 功能 | 数据区 | 地址范围 | | | 数据区 | CPU 地址 |
| 01 | 读位 | 输出 | 1 | 到 | 8192 | 输出过程映像 | Q0.0 到 Q1023.7 |
| 02 | 读位 | 输入 | 10001 | 到 | 18192 | 输入过程映像 | I0.0 到 I1023.7 |
| 04 | 读字 | 输入 | 30001 | 到 | 30512 | 输入过程映像 | IW0 到 IW1022 |
| 05 | 写位 | 输出 | 1 | 到 | 8192 | 输出过程映像 | Q0.0 到 Q1023.7 |
| 15 | 写位 | 输出 | 1 | 到 | 8192 | 输出过程映像 | Q0.0 到 Q1023.7 |

图 3-5 S7-1200 的 Modbus 地址映射表

设置完上述各管脚参数后，下载项目到 CPU1215C 中，打开 Modsan32 应用程序，下面以保持寄存器为例介绍通信测试过程。 在 Modscan32 的数据定义界面中设置数据类型为保持寄存器，并设置 Modbus 偏移量及长度，建立与 CPU1215C 集成 PN 口的通信连接，可以看到双方可以建立通信连接并进行数据读写，如下图 3-6 所示：

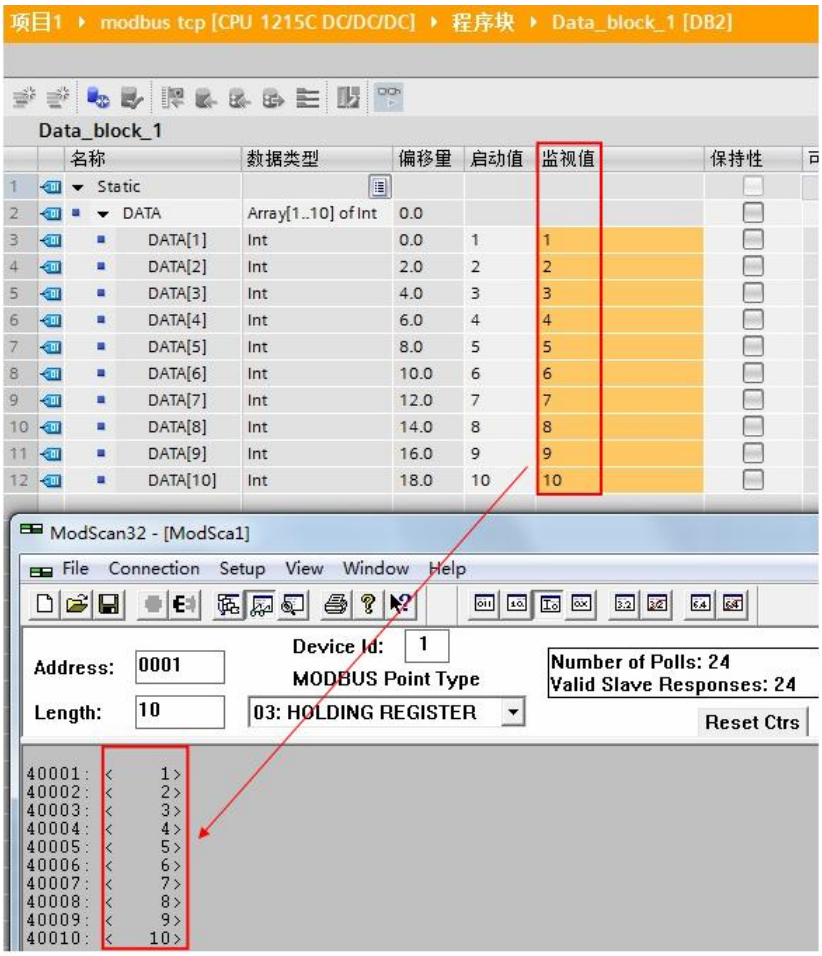



图 3-6 通信测试

对于其它数据类型，由于与 S7-1200CPU 的过程映像区进行了映射，其过程类似。

使用功能块“ MB_SERVER”的一些注意事项：

1）S7-1200 CPU 的集成 PN 口通过功能块“ MB_SERVER”支持与多个 Modbus 客户端的通信，支持的个数取决于 CPU 集成 PN 口所支持的 TCP 连接数，必须为每一个客户端连接分别调用一次功能块“ MB_SERVER”，其背景数据块、ID、端口号等参数必须唯 一。

2）S7-1200 CPU 的集成 PN 口支持多协议，除了运行 Modbus TCP 协议外，同时可以运行 PROFINET、TCP/IP、S7 等协议。

3）S7-1200 CPU 的集成 PN 口可以同时作为 Modbus TCP 的 Server 及 Client。

# 4 配置 S7-1200 CPU 作为 Modbus TCP Client 与通信伙伴建立通讯

在上述新建的项目中增加一个 CPU1215C 的站点，设置 PN 的 IP 地址“ 192.168.70.102”，之后在 CPU1215C 的 OB1 组织块中添加 Modbus TCP Client 功能块“ MB_CLIENT”，软件将提示会为该 FB 块增加一个背景数据块，本例中为 DB1“ MB_CLIENT_DB”，如下图 4-1 所示：

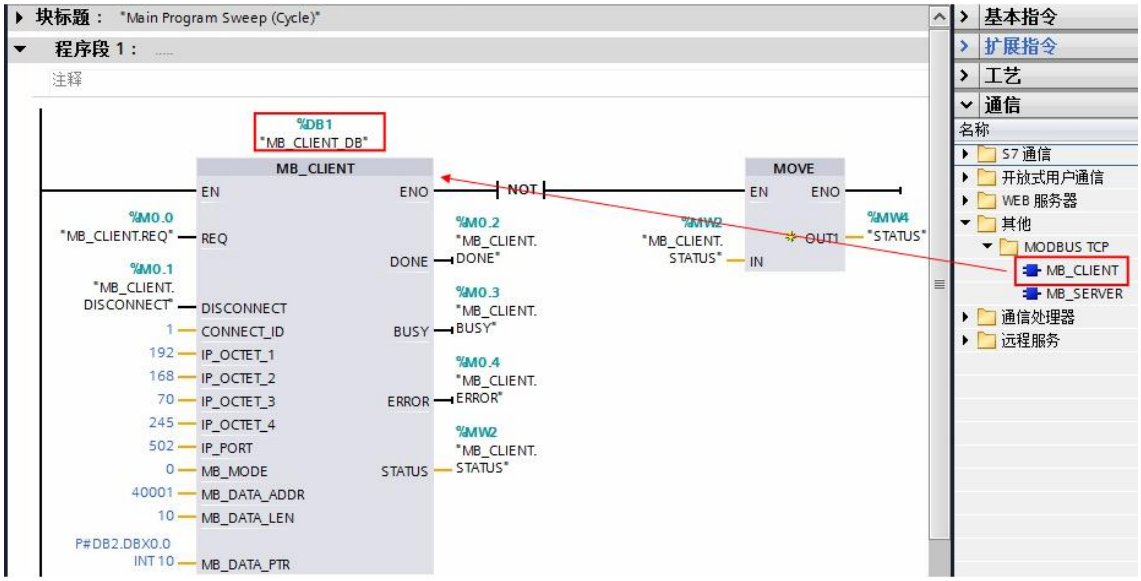
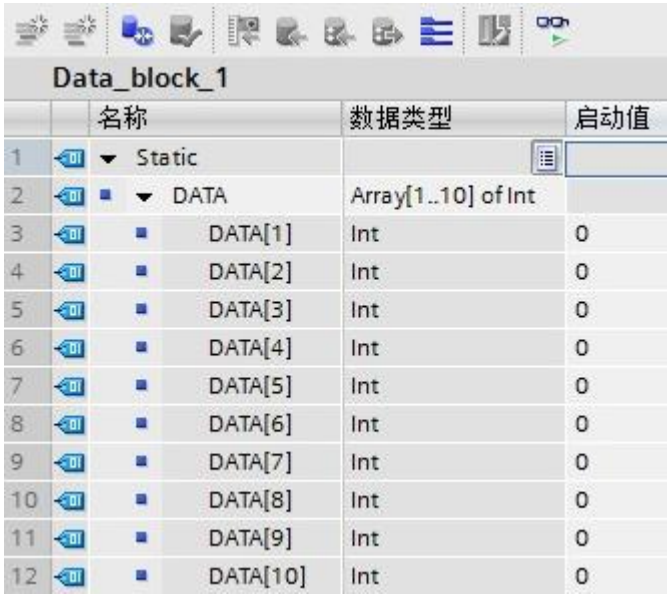



图 4-1 插入一个 MB_CLIENT 功能块

创建一个全局数据块用于匹配功能块“ MB_CLIENT”的管脚参数“ MB_DATA_PTR”，本例中为 DB2“ Data_block_1”，用于存储 Modbus 通信的数据，如下图 4-2 所示：



Data_block_1

| | | 名称 | 数据类型 | 启动值 |
|---|---|---|---|---|
| 1 | | Static | | |
| 2 | | DATA | Array[1..10] of Int | |
| 3 | | DATA[1] | Int | 0 |
| 4 | | DATA[2] | Int | 0 |
| 5 | | DATA[3] | Int | 0 |
| 6 | | DATA[4] | Int | 0 |
| 7 | | DATA[5] | Int | 0 |
| 8 | | DATA[6] | Int | 0 |
| 9 | | DATA[7] | Int | 0 |
| 10 | | DATA[8] | Int | 0 |
| 11 | | DATA[9] | Int | 0 |
| 12 | | DATA[10] | Int | 0 |

图 4-2 创建数据块 DB2

需要注意的是该数据块必须为非优化数据块(支持绝对寻址)，在该数据块的“属性”中不勾选“优化的块访问”选项，如下图 4-3 所示：

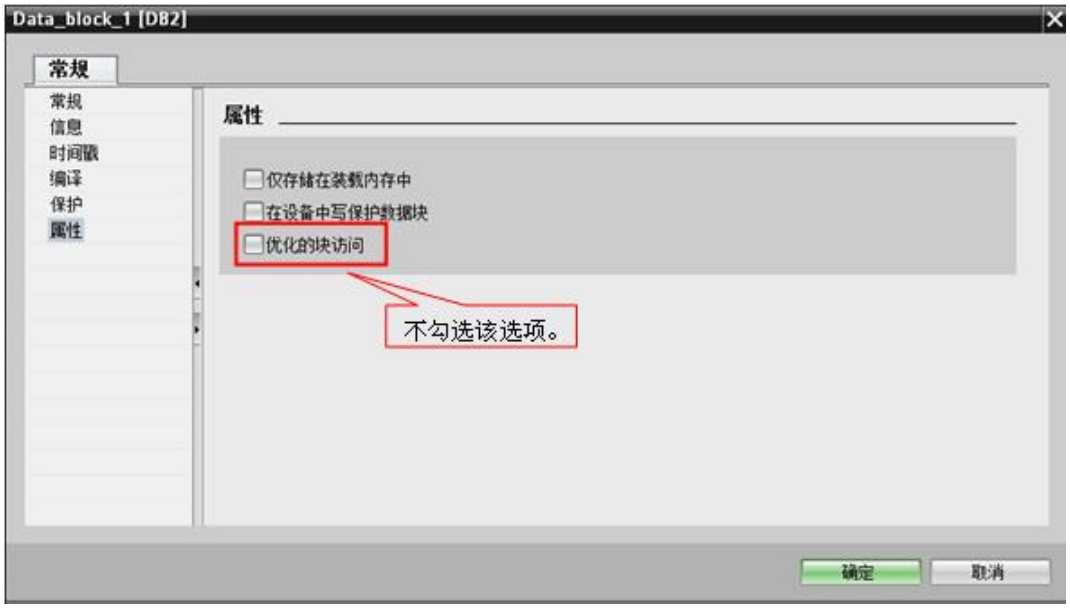



图 4-3 修改 DB 块属性

对于功能块“ MB_CLIENT”的其它参数管脚含义如下表 4 所示：

| “MB_CLIENT”的管脚参数 | 管脚声明 | 数据类型 | 含义 |
|---|---|---|---|
| REQ | 输入 | BOOL | FALSE=无 Modbus 通信请求<br>TRUE=请求与 Modbus TCP 服务器通信 |
| DISCONNECT | 输入 | BOOL | 0: 且连接不存在时，则可启动建立被动连接。<br>1: 且连接存在时，则断开连接。 |
| CONNECT_ID | 输入 | Uint | 唯一标识 PLC 中的每个连接。 |
| IP_OCTET_1 | 输入 | USint | Modbus TCP 服务器 IP 地址：八位字节 1 |
| IP_OCTET_2 | 输入 | USint | Modbus TCP 服务器 IP 地址：八位字节 2 |
| IP_OCTET_3 | 输入 | USint | Modbus TCP 服务器 IP 地址：八位字节 3 |
| IP_OCTET_4 | 输入 | USint | Modbus TCP 服务器 IP 地址：八位字节 4 |
| IP_PORT | 输入 | Uint | 默认值=502：服务器的 IP 端口号 |
| MB_MODE | 输入 | USint | 模式选择：分配请求类型（0=读、1=写） |
| MB_DATA_ADDR | 输入 | UDINT | 分配 MB_CLIENT 访问的数据的起始地址 |
| MB_DATA_LEN | 输入 | UINT | 数据长度： 数据访问的位数或字数 |
| MB_DATA_PTR | 输入/输出 | Variant | 指向 Modbus 数据寄存器的指针： 寄存器缓冲数据进入 Modbus 服务器或来自 Modbus 服务器。 该指针必须分配一个标准全局DB 或一个 M 存储器地址。 |
| DONE | 输出 | BOOL | 上一请求已完成且没有出错后，DONE 位将保持为 TRUE 一个扫描周期时间 |
| BUSY | 输出 | BOOL | 0：无 MB_CLIENT 操作正在进行<br>1：MB_CLIENT 操作正在进行 |
| ERROR | 输出 | BOOL | 0: 无错误<br>1: 出错。 出错原因由参数 STATUS 指示 |
| STATUS | 输出 | WORD | 指令的详细状态信息 |

表 4 功能块“MB_CLIENT”的其它管脚参数

对于“MB_MODE”“MB_DATA_ADDR”和“MB_DATA_LEN”参数，其对应关系如下图 4-4 所示：

| MB_MODE | Modbus 功能 | 数据长度 | 操作和数据 | MB_DATA_ADDR |
|---|---|---|---|---|
| 0 | 01 | 1 到 2000 | 读取输出位：<br>每个请求 1 到 2000 个位 | 1 到 9999 |
| 0 | 02 | 1 到 2000 | 读取输入位：<br>每个请求 1 到 2000 个位 | 10001 到 19999 |
| 0 | 03 | 1 到 125 | 读取保持寄存器：<br>每个请求 1 到 125 个字 | 40001 到 49999 或<br>400001 到 465535 |
| 0 | 04 | 1 到 125 | 读取输入字：<br>每个请求 1 到 125 个字 | 30001 到 39999 |
| 1 | 05 | 1 | 写入一个输出位：<br>每个请求一位 | 1 到 9999 |
| 1 | 06 | 1 | 写入一个保持寄存器：<br>每个请求 1 个字 | 40001 到 49999 或<br>400001 到 465535 |
| 1 | 15 | 2 到 1968 | 写入多个输出位：<br>每个请求 2 到 1968 个位 | 1 到 9999 |
| 1 | 16 | 2 到 123 | 写入多个保持寄存器：<br>每个请求 2 到 123 个字 | 40001 到 49999 或<br>400001 到 465535 |
| 2 | 15 | 1 到 1968 | 写入一个或多个输出位：<br>每个请求 1 到 1968 个位 | 1 到 9999 |
| 2 | 16 | 1 到 123 | 写入一个或多个保持寄存器：<br>每个请求 1 到 123 个字 | 40001 到 49999 或<br>400001 到 465535 |

图 4-4 “ MB_MODE”、“ MB_DATA_ADDR”和“ MB_DATA_LEN”参数对应关系

之后打开上述功能块“ MB_CLIENT”的背景数据块，在“ MB_UNIT_ID”参数中表示通信服务器伙伴的从站地址，该地址与通信伙伴一致，如下图 4-5 所示：

图 4-5 在功能块“ MB_CLIENT”的背景数据块设置 Unit ID

设置完上述各管脚参数后，下载项目到 CPU1215C 中，打开 Modsim32 应用程序，下面以保持寄存器为例介绍通信测试过程。

在 Modsim32 的数据定义界面中设置数据类型为保持寄存器，依据功能块“ MB_CLIENT”设置的起始地址“ MB_DATA_ADDR”和长度“ MB_DATA_LEN”，可以看到双方可以建立通信连接并进行数据读写，如下图 4-6 所示：

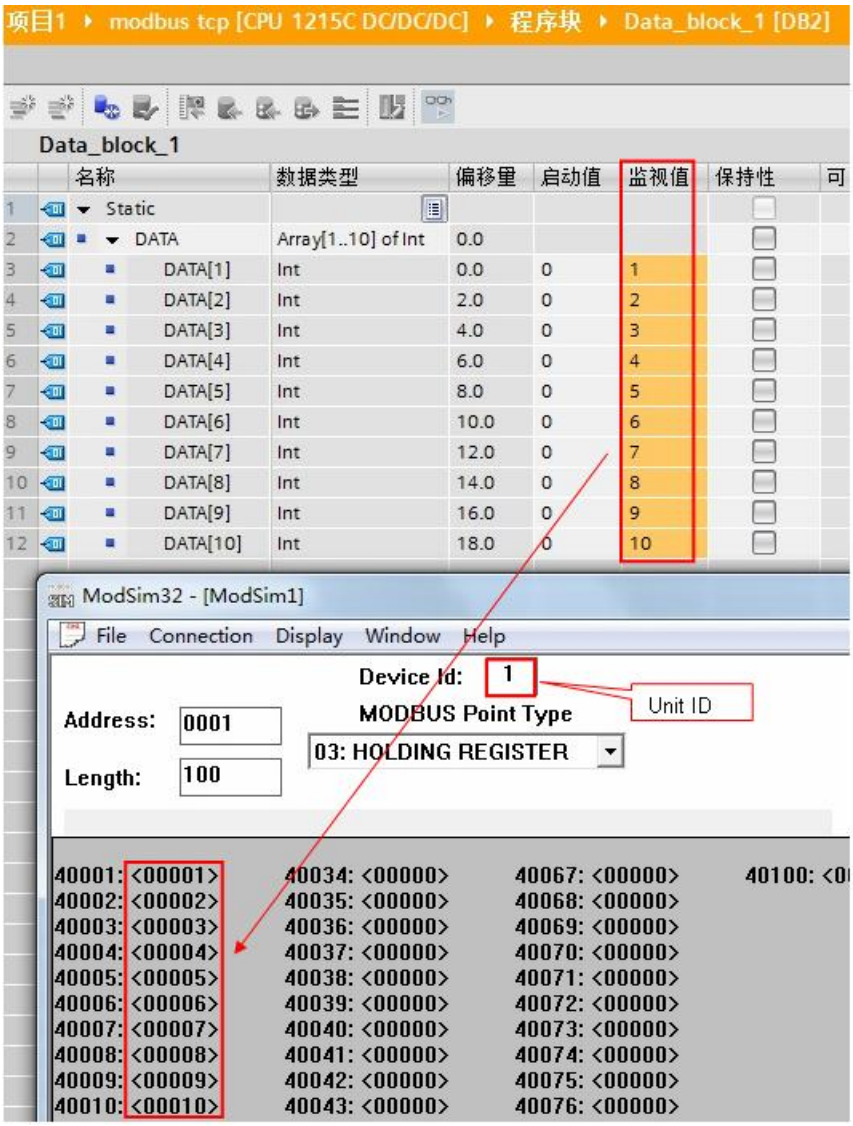



图 4-6 通信测试

对于其它数据类型，测试过程类似。

使用功能块“ MB_CLIENT”的一些注意事项：

1）S7-1200 CPU 的集成 PN 口通过功能块“ MB_CLIENT”支持与多个 Modbus 服务器的通信，支持的个数取决于 CPU 集成 PN 口所支持的 TCP 连接数，必须为每一个服务器连接需要分别调用一次功能块“ MB_CLIENT”，其背景数据块、ID 等参数必须唯一。

2）S7-1200 CPU 的集成 PN 口可以同时作为 Modbus TCP 的 Server 及 Client。

# 5 本文说明

本文所描述的内容适用于 S7-1200 V4.0 及以下 版本的 CPU 实现 Modbus TCP 通信，关于 S7-1200 V4.1 及以上版本的 CPU 实现 Modbus TCP 通信，指令使用方法与 S7-1500 相同，请参考如下链接：

S7-1500 CPU 集成 PN 口的 ModbusTCP 通信快速入门

http://support.automation.siemens.com/CN/view/zh/90974593