

操作指南 • 12/2018

S7-400H 与 S7-200 smart 通过 以太网进行通讯

<http://support.automation.siemens.com/CN/view/zh/109762803>

Unrestricted

目录

- 1 **S7-400H 硬件组态4**
 - 1.1 S7-400H 组态 PN 口地址4
 - 1.2 S7-400H 组态网络连接4
- 2 **S7-400H 程序编写8**
 - 2.1 通过 SFC51 读取 CPU 主从状态8
 - 2.2 编写通讯程序..... 10
- 3 **通讯测试..... 12**

S7-400H 是西门子 S7 产品的高端系列，可以实现硬件冗余自动化方案，并可满足故障安全功能的需求，S7-200 smart 系列 PLC 是全新的针对经济型自动化市场的自动化控制产品。S7-400H 系统和 S7-200 smart 之间的通讯可以通过两种方式实现：第一种方式是通过以太网通讯编写通讯程序；第二种方式是通过 Y-Link 和 EM DP01 扩展模块实现 DP 总线通讯。本文主要介绍基于以太网通讯的实现方式。

S7-400H 冗余控制器通过 CPU 的集成口或以太网通讯模板和 S7-200 smart 之间的冗余通讯的基本思路是：分别建立 RACK0 和 RACK1 与 S7-200 smart 的通讯链接，通过 SFC51 读取 400H 的主从状态，主 CPU 通过对应的通讯链接向 S7-200 smart 发送数据。

下面以 410-5H 控制器与 CPU SR60 为例，介绍一下具体的组态过程。项目中采用 PUT/GET 指令，S7-200 smart 从 V2.0 版本才开始支持 PUT/GET 通讯，V1.0 版本的 CPU 需要升级固件后才能支持 PUT/GET，S7-200 smart 侧不需要编程。

项目测试环境如下：

S7-400H 侧：6ES7 410-5HX08-0AB0，软件 PCS7 V9.0 SP1(集成 Step7 V5.6 HF3)

S7-200 smart 侧：6ES7 288-1SR60-0AA0，软件 STEP 7 MicroWIN SMART V2.3.0.2

1 S7-400H 硬件组态

1.1 S7-400H 组态 PN 口地址

在 SIMATIC Manager 中组态 S7-400H，设置 CPU 集成 PN 口的 IP 地址
RACK0:192.168.0.141; RACK1:192.168.0.142

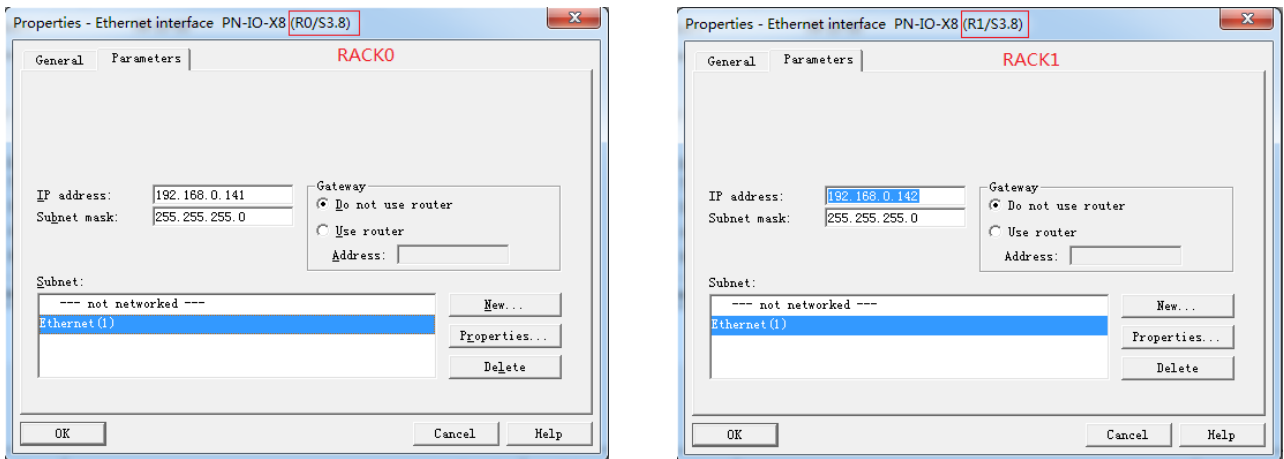


图 1-1 组态 400H 以太网地址

1.2 S7-400H 组态网络连接

在 SIMATIC Manager 中选择 “NetPro” 进行组态，首先选择 RACK0，鼠标右键选择 “Insert New Connection”，在弹出的对话框中，选择连接方为 “Unspecified”，连接类型为 “S7 connection”，然后选择 “Apply” 按钮，如下图所示：

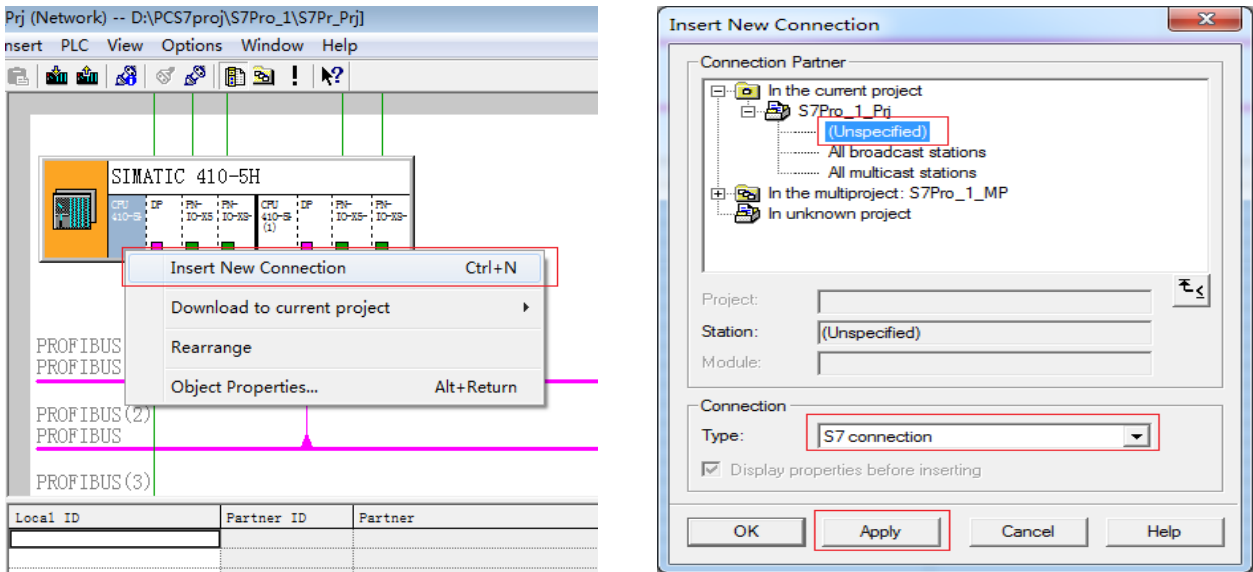


图 1-2 建立以太网连接

在弹出来的对话框中，生成的 Local ID 将在程序中使用，选择“Establish an active connection”建立主动连接，设置需要通讯的 S7-200 smart 的 IP 地址，点击“Address Detail”按钮，如下图所示：

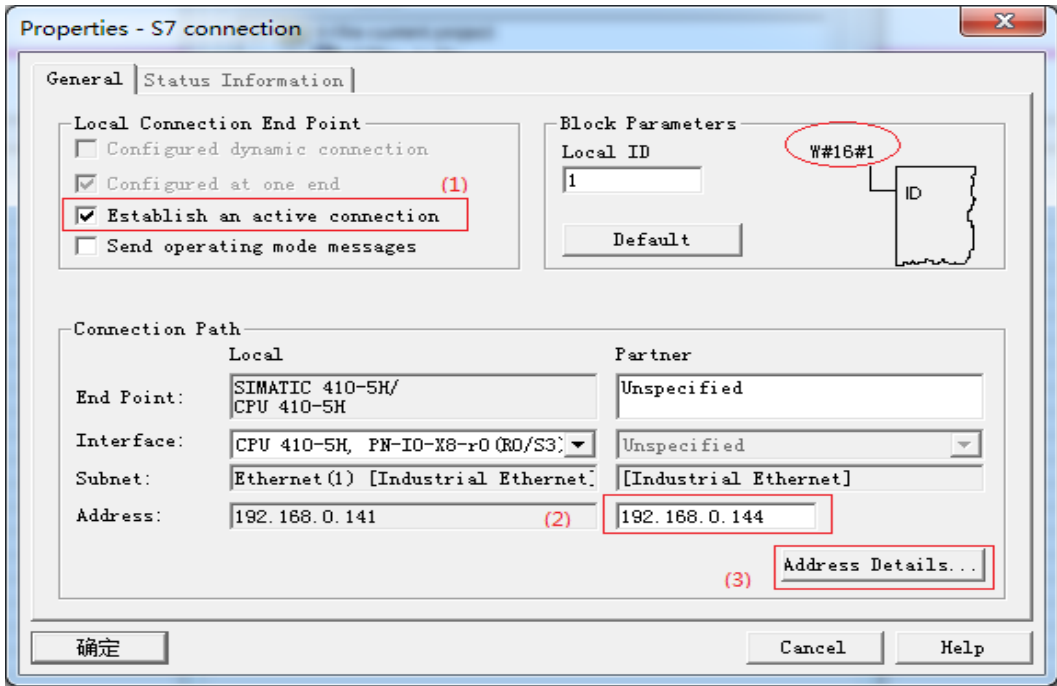


图 1-3 设置 RACK0 的连接参数

在弹出来的对话框中设置连接的 TSAP 号，选择 Partner 的 Slot 为 1，如下图所示

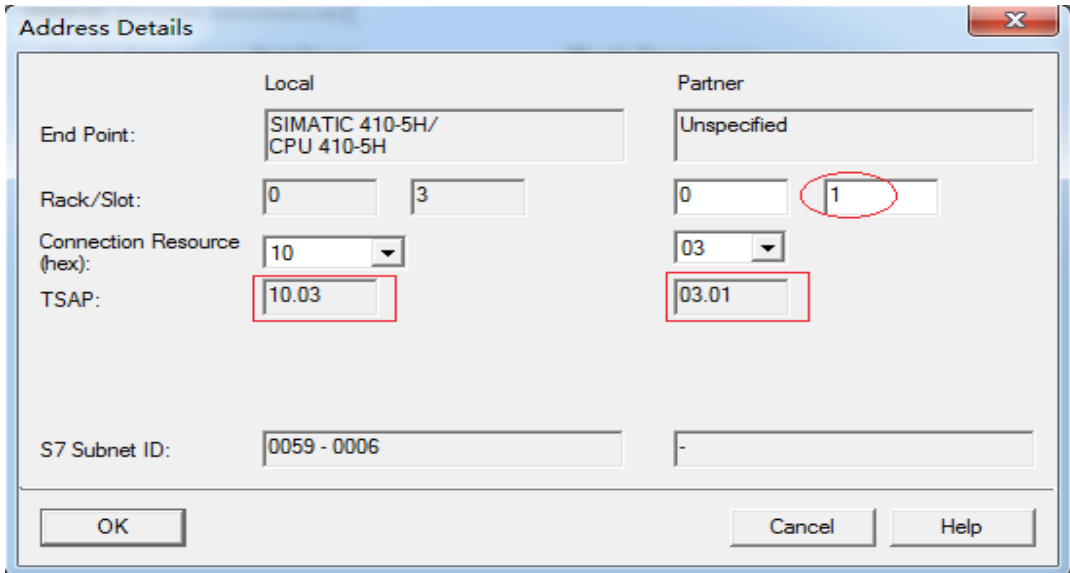


图 1-4 设置 RACK0 的 TSAP 参数

按照相同的步骤，选择 RACK1，插入连接，如下图所示

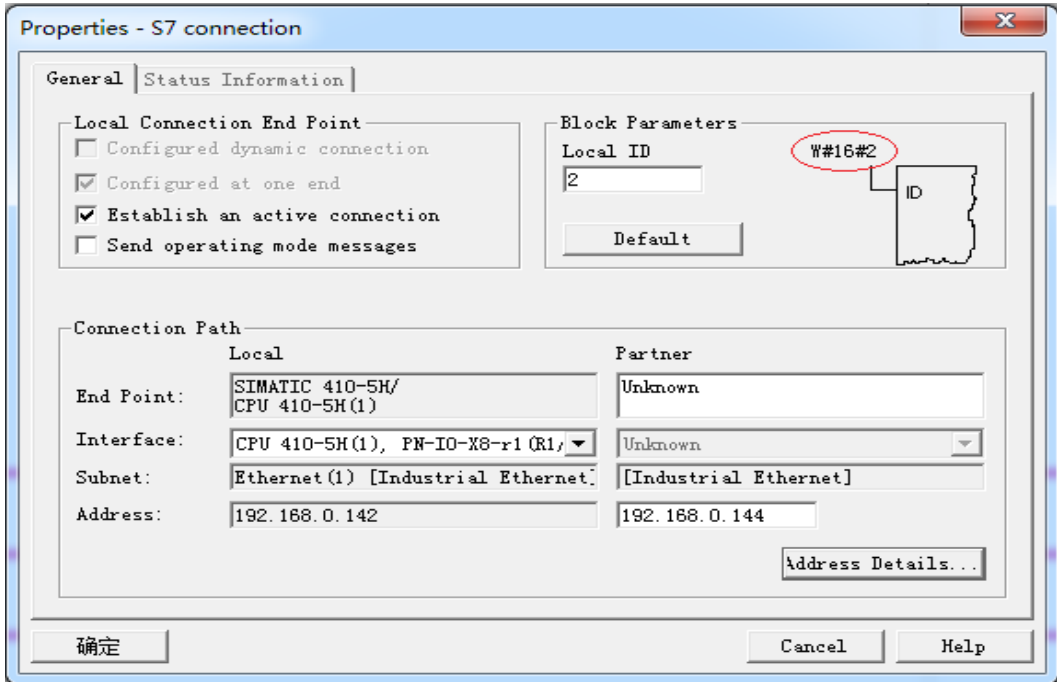


图 1-5 设置 RACK1 的连接参数

在弹出来的对话框中设置连接的 TSAP 号，选择 Partner 的 Slot 为 1，如下图所示

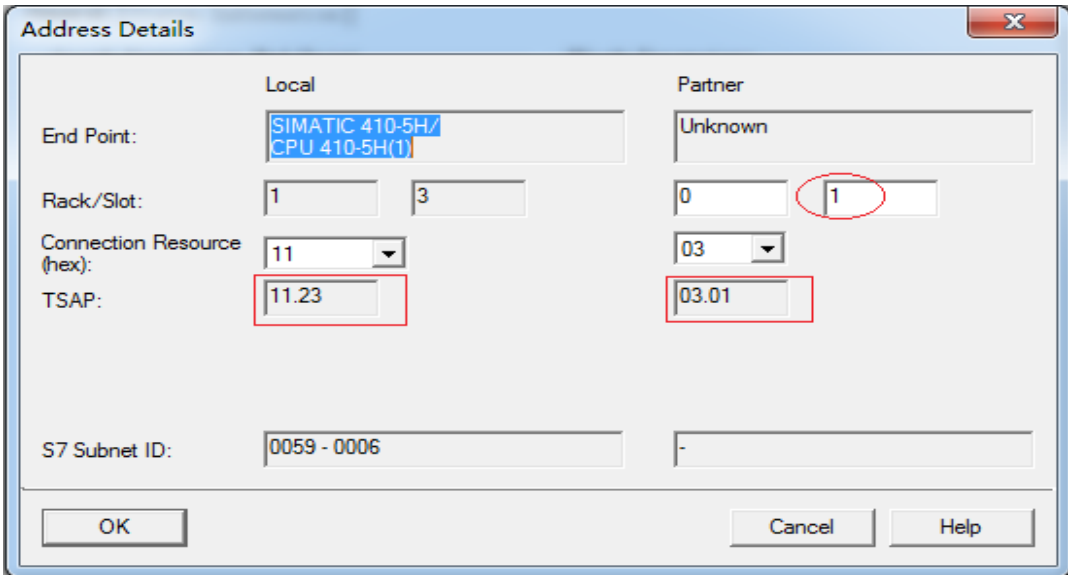


图 1-6 设置 RACK1 的 TSAP 参数

保存编译网络组态后，分别选中 RACK0 和 RACK1 的 CPU，鼠标右键下载选中的连接到对应的 CPU。

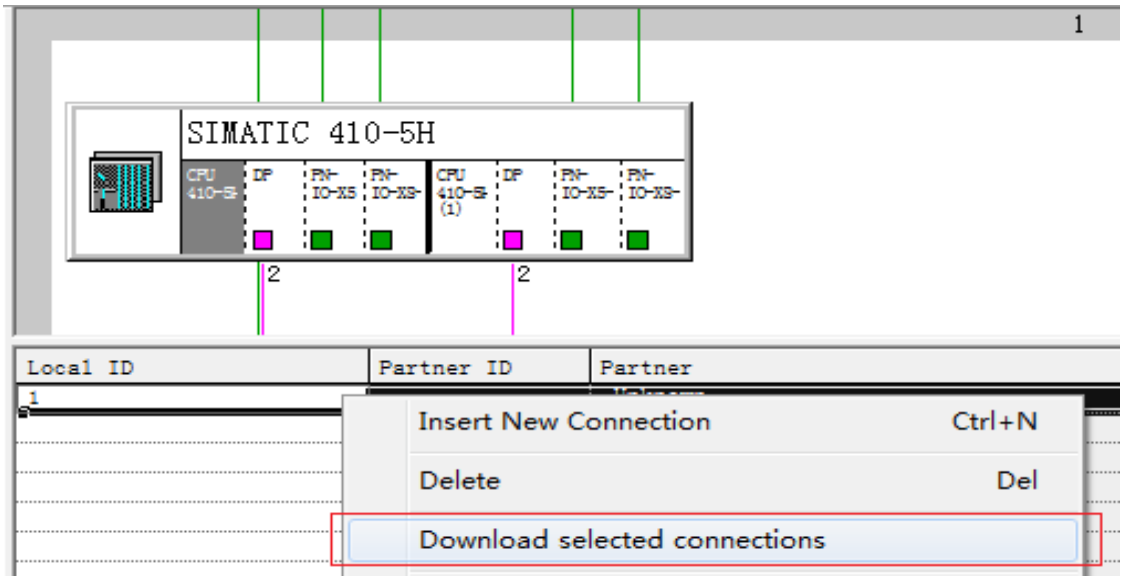


图 1-7 下载网络连接

2 S7-400H 程序编写

2.1 通过 SFC51 读取 CPU 主从状态

通过 SFC51，可以读取系统状态、从站状态，详细使用可以参考 SFC51 手册或在线帮助文档
<https://support.industry.siemens.com/cs/document/79609718/sfc-51%E5%B8%B8%E7%94%A8%E5%8A%9F%E8%83%BD%E4%BD%BF%E7%94%A8%E5%85%A5%E9%97%A8?dti=0&lc=zh-CN>,

本例中，仅通过 SFC51 读取 S7-400H 的 CPU 的 Master 和 standby 信息。

- ①创建自定义函数块 FC61，在临时变量区创建一个变量 length，类型设置为 Struct
- ②双击 length 变量，进入结构变量成员定义，创建两个 word 类型的变量，分别为 size 和 number
- ③创建数据块 DB2，在 FC61 中调用系统函数 SFC51 读取指示灯状态，将 FC61 在指定 OB 块中调用

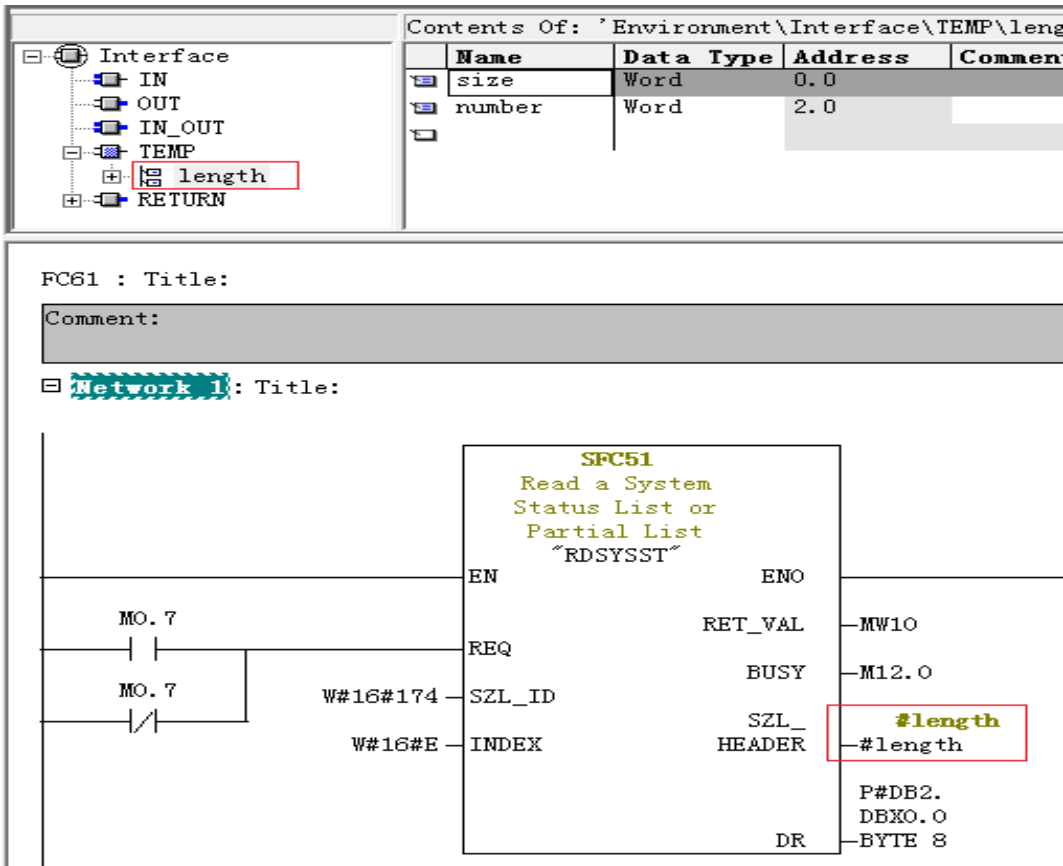


图 2-1 读取指示灯状态

SZL_ID: W#16#174 代表读取单个指示灯
INDEX: W#16#E 代表读取 MSTR 指示灯
DR: 指示灯状态读取结果保存到 DB2 的前 8 个字节 (P#DB2.DBX0.0 BYTE 8)

每个指示灯转态占用 4 字节，定义如下图

| Name | Length | Meaning |
|------------|--------|--|
| cpu_led_ID | 1 word | <ul style="list-style-type: none">Byte 0 Standard CPU: B#16#00 H-CPU: Bits 0 to 2: rack number Bit 3: 0=standby CPU, 1=master CPU Bits 4 to 7: 1111 Byte 1: LED ID |
| led_on | 1 byte | Status of the LED: 0: off 1: on |
| led_blink | 1 byte | Flashing status of the LED: 0: not flashing 1: flashing normally (2 Hz) 2: flashing slowly (0.5 Hz) |

图 2-2 指示灯状态位定义

读取到的指示灯状态如下图所示

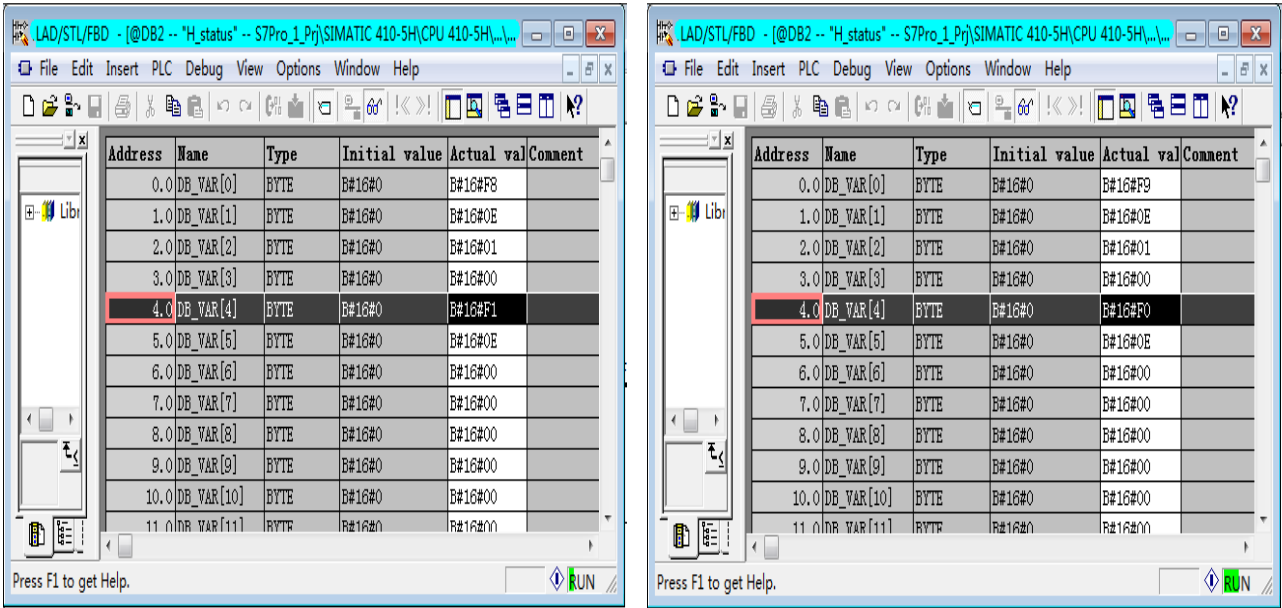


图 2-3 指示灯状态值

第一个字节代表当前的主 CPU 的信息，

B#16#F8 <==> 1111 1000 <==> Master RACK0
B#16#F9 <==> 1111 1001 <==> Master RACK1

2.2 编写通讯程序

Bit0=0 代表 RACK0 为 MASTER，Bit0=1 代表 RACK1 为 MASTER，调用相应的通讯程序 FC62、FC63

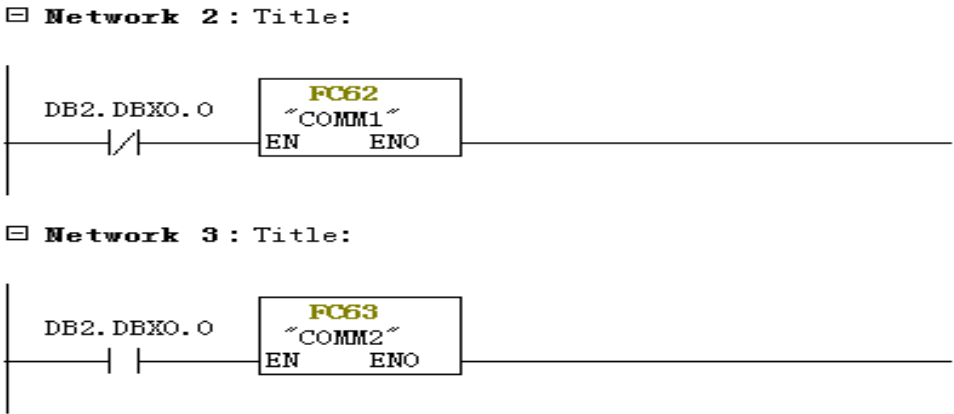


图 2-4 调用不同连接 ID 的函数块

S7-400H 通过调用系统功能块 SFB14，SFB15 完成通讯功能

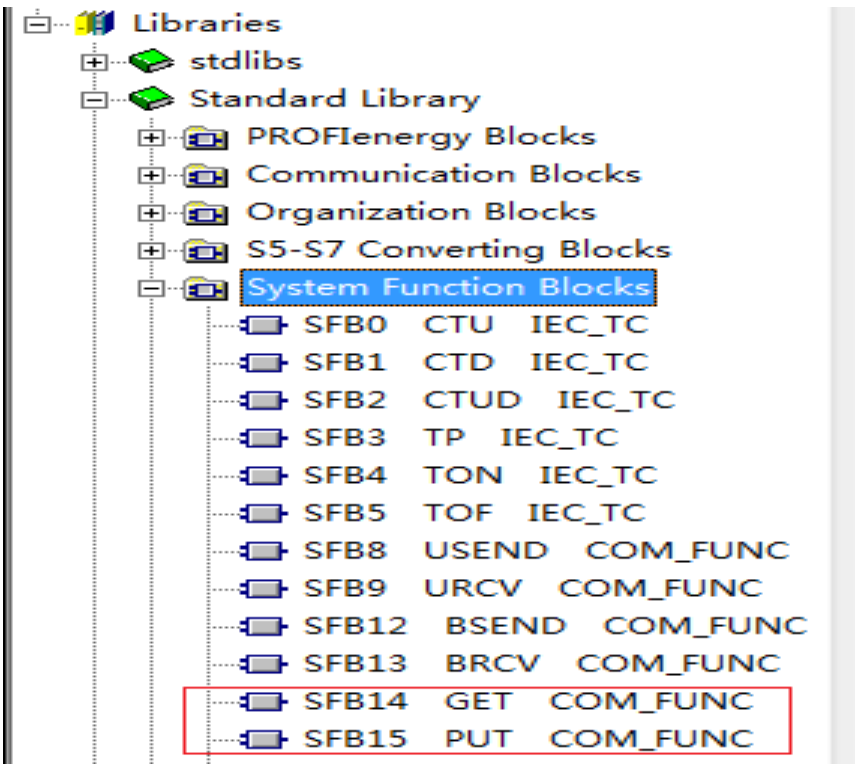


图 2-5 选择系统功能块

S7-400H 没有 V 区，需要手动建立 DB1 来代替，

M0.7: 时钟脉冲

GET: VB0~VB3 ==> DB1.DBB0~DB1.DBB3

PUT: DB1.DBB4~DB1.DBB7 ==> VB4~VB7

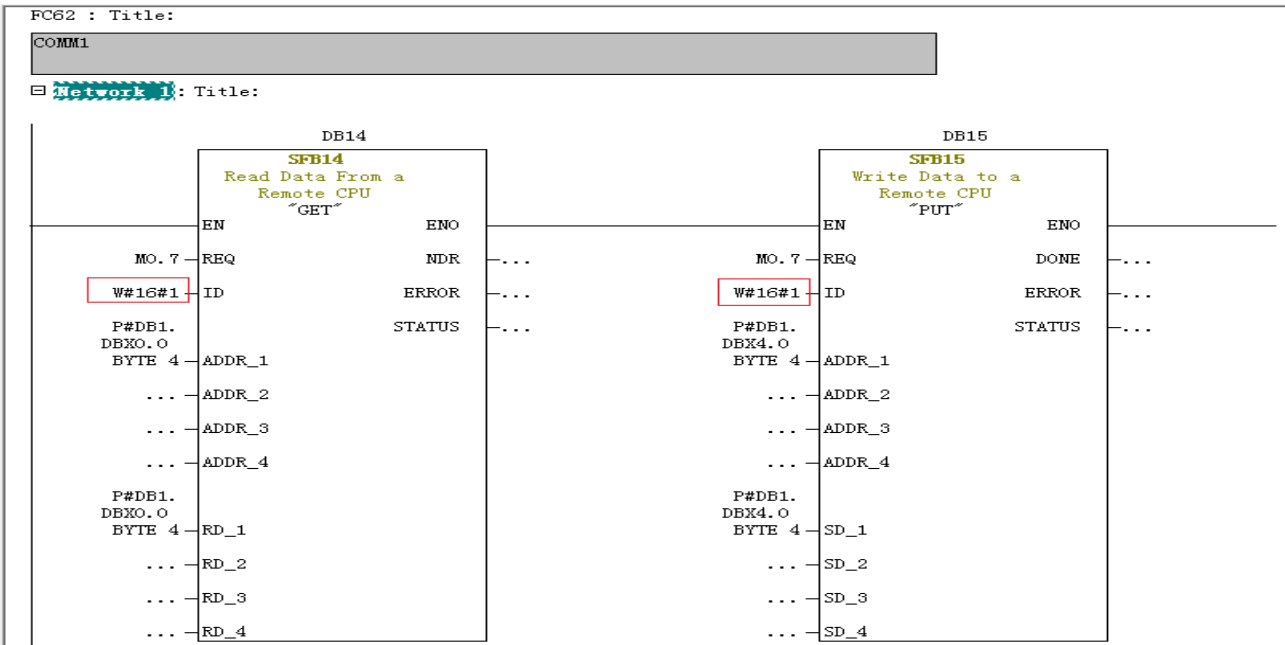


图 2-6 RACK0 通讯程序

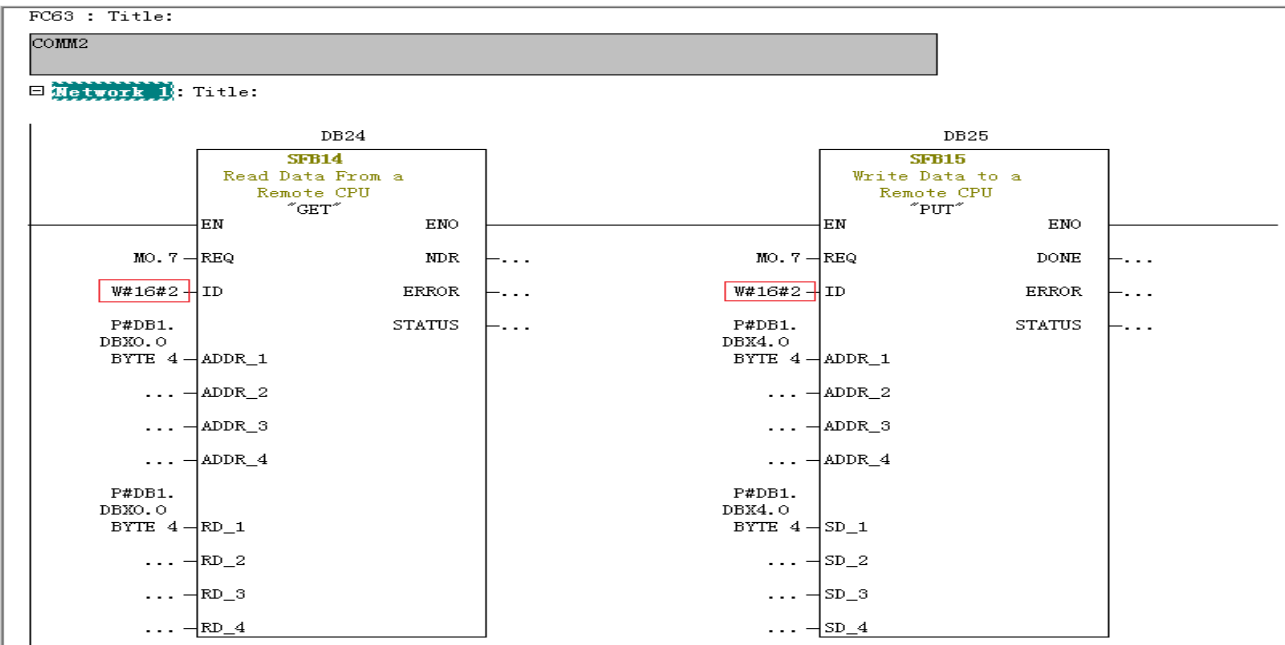


图 2-7 RACK1 通讯程序

3 通讯测试

在 S7-400H 侧写入 DB1.DBB4~DB1.DBB7，在 S7-200 smart 侧写入 VB0~VB3，同时对 S7-400H 进行主备 CPU 切换测试，通讯测试结果如下图所示：

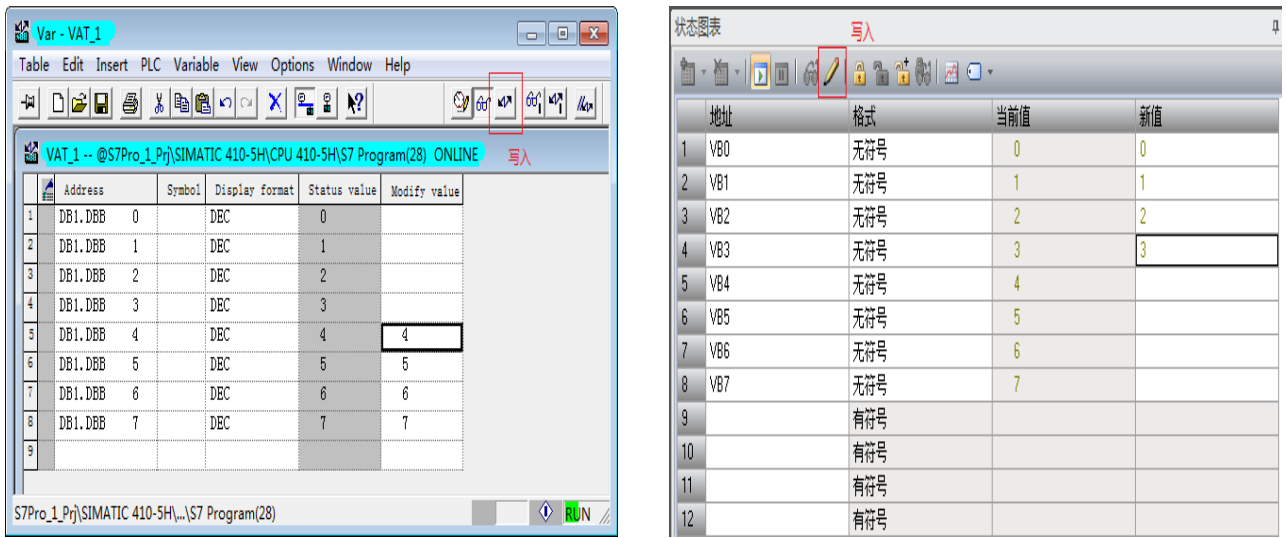


图 3-1 通讯数据监控