电气信息工程丛书

西门子(中国)有限公司重点推荐图书

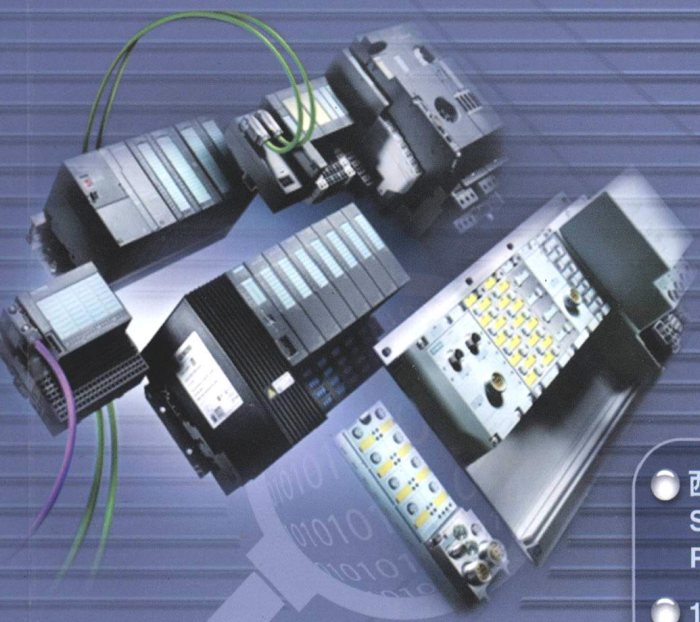
SIEMENS

西门子工业通信网络

组态编程与故障诊断

主　编　廖常初

副主编　祖正容



赠送超值 DVD 光盘：

- 西门子(中国)有限公司授权的通信软件：SIMATIC NET、Drivemonitor、iMap PDM V6.0、S7-PDIAG(不含许可证)
- 100多本中英文用户手册
- 100多个应用实例

机械工业出版社
CHINA MACHINE PRESS

电气信息工程丛书

西门子工业通信网络组态编程与故障诊断

主　编　廖常初
副主编　祖正容

机械工业出版社

本书全面介绍了西门子工业通信网络的结构、通信协议、通信服务和通信的组态编程与故障诊断。重点是应用最广的 PROFIBUS-DP 和工业以太网，对 MPI、AS-i、PROFIBUS-PA、OPC 也作了详细介绍。

本书建立在大量实验的基础上，详细介绍了实现通信最关键的组态和编程的方法，随书光盘有上百个通信例程，绝大多数例程经过硬件实验的验证。读者根据正文介绍的通信系统的组态步骤和方法，参考光盘中的例程作组态和编程练习，可以较快地掌握网络通信的实现方法。

通信的故障诊断是现场维修的难点。本书用约三分之一的篇幅和大量的实例，系统地介绍了网络通信的故障诊断方法、诊断数据的分析方法，和用人机界面、WinCC 显示故障消息的方法，包括一种功能强大、容易实现的故障诊断和显示的方法。

除了例程，随书光盘还提供了多个西门子大型通信软件和 100 多本中英文用户手册。本书各章配有适量的练习题，可供工程技术人员和维修人员自学，也可作为大专院校、培训班的教材或参考书。

图书在版编目（CIP）数据

西门子工业通信网络组态编程与故障诊断 / 廖常初主编．—北京：机械工业出版社，2009.9

（电气信息工程丛书）

ISBN 978-7-111-28256-3

Ⅰ．西…　Ⅱ．廖…　Ⅲ．工业—通信网　Ⅳ．TP393.18

中国版本图书馆 CIP 数据核字（2009）第 162962 号

机械工业出版社（北京市百万庄大街 22 号　邮政编码 100037）

责任编辑：李馨馨

责任印制：洪汉军

三河市国英印务有限公司印刷

2016 年 7 月第 1 版 • 第 5次印刷

184mm×260mm • 30.75 印张 • 761 千字

12301—14800 册

标准书号：ISBN 978-7-111-28256-3

ISBN 978-7-89451-204-8（光盘）

定价：69.00 元（含 1DVD）

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

社服务中心：（010）88361066

销 售 一 部 ：（010）68326294

销 售 二 部 ：（010）88379649

读者服务部：（010）68323821

网络服务

门户网：http:///www.cmpbook.com

教材网：http://www.cmpedu.com

封面无防伪标均为盗版

前　言

工业控制网络已经成为现代工业控制系统不可缺少的重要组成部分，从计算机、PLC 到现场的 I/O 设备、驱动设备和人机界面，网络通信无处不在。西门子是自动化领域最大的供应商，该公司支持的 PROFIBUS、PROFINET 和 AS-i 已成为 IEC 现场总线的国际标准和我国的国家标准。PROFIBUS 已经有两千多万个节点投入运行。

本书对西门子工业通信网络的结构、通信协议、通信服务和通信的组态与编程进行了全面的介绍，对通信中常用的一些基本概念和名词也作了介绍。

本书紧密结合工业通信网络的应用实践，以当前应用最广的 PROFIBUS-DP 和工业以太网为重点。第 2 章介绍了 PROFIBUS 的硬件与通信协议，第 3 章介绍了 DP 主站与 ET 200、智能从站、变频器和直流调速装置等设备之间的主从通信。以及通信处理器在主从通信中的应用。第 4 章介绍了基于 PROFIBUS 的 S7 通信和 FDL 通信。第 5 章介绍了直接数据交换通信和 DP 通信的特殊应用。第 6～8 章介绍了 PROFIBUS 通信的故障诊断与显示的方法。第 9 章介绍了 PROFIBUS-PA。第 10 章介绍了基于工业以太网的 S5 兼容通信和 S7 通信。第 11 章介绍了 PROFINET 通信与工业以太网的故障诊断。第 12 章介绍了 AS-i。第 13 章介绍了 OPC 通信。第 14 章介绍了 MPI 的全局数据通信、S7 基本通信和 S7 通信。第 15 章介绍了点对点通信和 S7 路由，对其他应用较少的通信方式作了简要的介绍。

本书对实现通信最关键的问题——组态与编程作了详细的介绍。全书的内容建立在硬件实验的基础上，随书光盘提供了上百个通信例程，绝大多数例程经过硬件实验的验证，书中对例程的组态过程、通信程序和验证通信的方法作了详细的说明。读者可以一边看书，一边用 STEP 7 打开相应的例程，通过例程了解组态和编程的方法。本书介绍的方法具有很强的可操作性，读者可以根据书中介绍的组态的步骤和方法，同时参考光盘中的例程，做组态和编程的练习，这样可以较快地掌握通信网络的组态和编程方法。有条件的读者可以在看书的同时做一些硬件实验。

现代网络控制系统越来越复杂，网络通信的故障诊断是现场电气维修人员面临的新的巨大的挑战。西门子提供了大量的用于故障诊断和显示的硬件、软件和诊断方法，但是大多数现场维修人员对此知之甚少。本书用了约三分之一的篇幅，通过大量的实例，系统地介绍了网络通信的故障诊断方法，包括用模块上的 LED 和 STEP 7 进行诊断，中断组织块在故障诊断中的应用，用中断组织块的局部变量和通信块的输出参数进行诊断，用通信处理器和专用硬件进行诊断，用诊断程序块进行诊断，诊断数据的分析方法，以及通过全集成自动化（TIA），用人机界面和 WinCC 显示故障消息的方法。另外还介绍了一种功能强大、容易实现的故障诊断和显示的方法——报告系统错误功能。

本书与当前使用的西门子的软件和硬件配套，内容新颖实用。本书的例程基于 STEP 7 V5.4.3.1 中文版和 WinCC flexible 2007 中文版，作者编写的《S7-300/400 PLC 应用技术（第 2 版）》和《西门子人机界面（触摸屏）组态与应用技术（第 2 版）》的随书光盘有这两个软件的演示版。

本书对内容、插图和程序作了优化处理，详细介绍了第一次出现的硬件和网络的组态过

程和通信程序。后面的章节涉及到类似的组态过程和程序时，只作简单的说明。详细的情况可以查看随书光盘的例程中的组态结果和程序代码。这样避免了大量的重复，减少了篇幅，使本书具有很高的性能价格比。建议对硬件组态和网络组态不太熟悉的读者，从第 3 章开始，按书上的顺序阅读组态过程并做组态的练习。

限于篇幅，本书对硬件选型、组网和硬件安装方面的内容介绍得较少。读者可以查阅随书光盘的文件夹“产品样本”中的文档获取有关的信息。

经西门子公司授权，随书光盘有常用的通信软件，以及大量的与通信有关的中英文用户手册。本书的附录有常用缩写词和随书光盘内容简介，各章配有适量的练习题。

本书的编写得到了西门子（中国）有限公司的大力支持，宋柏青先生、元娜、许艳婷女士对本书的编写提供了很大的帮助，在此表示衷心的感谢。

本书由廖常初任主编，祖正容任副主编，陈晓东、陈曾汉、范占华、杨太平、文家学、刘道芳、廖亮、左源洁、万莉、孙明秀、左渊林、王云杰、杨斌、唐永红参加了编写工作。

因作者水平有限，书中难免有错漏之处，恳请读者批评指正。

作者 E-mail：liaosun@cqu.edu.cn。

重庆大学电气工程学院 廖常初

目　录

第1章　概　　述

1.1　计算机通信的国际标准

1.1.1　开放系统互连模型

国际标准化组织 ISO 提出了开放系统互连模型 OSI，作为通信网络国际标准化的参考模型，它详细描述了通信功能的 7 个层次（见图 1-1）。

7 层模型分为两类，一类是面向用户的第 5～7 层，另一类是面向网络的第 1～4 层。前者给用户提供适当的方式去访问网络系统，后者描述数据怎样从一个地方传输到另一个地方。

发送方传送给接收方的数据，实际上是经过发送方各层从上到下传递到物理层，通过物理媒体（媒体又称为介质）传输到接收方后，再经过从下到上各层的传递，最后到达接收方的应用程序。发送方的每一层协议都要在数据报文前增加报文头，报文头包含完成数据传输所需的控制信息，只能被接收方的同一层识别和使用。接收方的每一层只阅读本层的报文头的控制信息，并进行相应的协议操作，然后删除本层的报文头，最后得到发送方发送的数据。

1．物理层

物理层的下面是物理媒体，例如双绞线、同轴电缆和光纤等。物理层为用户提供建立、保持和断开物理连接的功能，定义了传输媒体接口的机械、电气、功能和规程的特性。RS-232C、RS-422 和 RS-485 等就是物理层标准的例子。

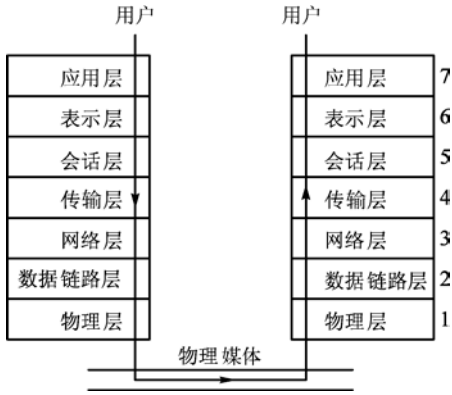


图 1-1　开放系统互连模型

2．数据链路层

数据链路层的数据以帧（Frame）为单位传送，每一帧包含一定数量的数据和必要的控制信息，例如同步信息、地址信息和流量控制信息。通过校验、确认和要求重发等方法实现差错控制。数据链路层负责在两个相邻节点间的链路上，实现差错控制、数据成帧和同步控制等。

3．网络层

网络层的主要功能是报文包的分段、报文包阻塞的处理和通信子网中路径的选择。

4．传输层

传输层的信息传送单位是报文（Message），它的主要功能是流量控制、差错控制、连接支持，传输层向上一层提供一个可靠的端到端（end-to-end）的数据传送服务。

5．会话层

会话层的功能是支持通信管理和实现最终用户应用进程之间的同步，按正确的顺序收发数据，进行各种对话。

6．表示层

表示层用于应用层信息内容的形式变换，例如数据加密/解密、信息压缩/解压和数据兼容，把应用层提供的信息变成能够共同理解的形式。

7．应用层

应用层作为 OSI 的最高层，为用户的应用服务提供信息交换，为应用接口提供操作标准。

不是所有的通信协议都需要 OSI 参考模型中的全部 7 层，例如有的现场总线通信协议只采用了 7 层协议中的第 1、2 和 7 层。

1.1.2 IEEE 802 通信标准

IEEE（国际电工与电子工程师学会）的 802 委员会于 1982 年颁布了一系列计算机局域网分层通信协议标准草案，总称为 IEEE 802 标准。它把 OSI 参考模型的底部两层分解为逻辑链路控制层（LLC）、媒体访问控制层（MAC）和物理传输层。前两层对应于 OSI 参考模型中的数据链路层，数据链路层是一条链路（Link）两端的两台设备进行通信时必须共同遵守的规则和约定。

媒体访问控制层（MAC）的主要功能是控制对传输媒体的访问，实现帧的寻址和识别，并检测传输媒体的异常情况。逻辑链路控制层（LLC）用于在节点间对帧的发送、接收信号进行控制，同时检验传输中的差错。MAC 层对应于三种已经建立的标准，即带冲突检测的载波侦听多路访问（CSMA/CD）通信协议、令牌总线（Token Bus）和令牌环（Token Ring）。

1．CSMA/CD

CSMA/CD 通信协议的基础是 Xerox 等公司研制的以太网（Ethernet），早期的 IEEE 802.3 标准规定的波特率为 10 Mbit/s，后来发布了 100 Mbit/s 的快速以太网 IEEE 802.3u，1000 Mbit/s 的千兆以太网 IEEE 802.3z，以及 10000 Mbit/s 的 IEEE 802ae。

CSMA/CD 各站共享一条广播式的传输总线，每个站都是平等的，采用竞争方式发送信息到传输线上，也就是说，任何一个站都可以随时广播报文，并为其他各站接收。当某个站识别到报文上的接收站名与本站的站名相同时，便将报文接收下来。由于没有专门的控制站，两个或多个站可能因为同时发送信息而发生冲突，造成报文作废，因此必须采取措施来防止冲突。

发送站在发送报文之前，先监听一下总线是否空闲，如果空闲，则发送报文到总线上，称之为“先听后讲”。但是这样做仍然有发生冲突的可能，因为从组织报文到报文在总线上传输需要一段时间，在这段时间内，另一个站通过监听也可能会认为总线空闲，并发送报文到总线上，这样就会因为两个站同时发送而发生冲突。

为了防止冲突，在发送报文开始的一段时间，仍然监听总线，采用边发送边接收的办法，

把接收到的信息和自己发送的信息相比较，若相同则继续发送，称之为“边听边讲”；若不相同则说明发生了冲突，立即停止发送报文，并发送一段简短的冲突标志（阻塞码序列），来通知总线上的其他站点。为了避免产生冲突的站同时重发它们的帧，采用专门的算法来计算重发的延迟时间。通常把这种“先听后讲”和“边听边讲”相结合的方法称为 CSMA/CD（带冲突检测的载波侦听多路访问技术），其控制策略是竞争发送、广播式传送、载体监听、冲突检测、冲突后退和再试发送。

以太网首先在个人计算机网络系统，例如办公自动化系统和管理信息系统（MIS）中得到了极为广泛的应用，以太网的硬件（例如网卡、集线器和交换机）非常便宜。

在以太网发展的初期，通信速率较低。如果网络中的设备较多，信息交换比较频繁，可能会经常出现竞争和冲突，影响信息传输的实时性。随着以太网传输速率的提高（100～1000 Mbit/s）和采用了相应的措施，这一问题已经解决，现在以太网在工业控制中得到了广泛的应用，大型工业控制系统最上层的网络几乎全部采用以太网。使用以太网很容易实现管理网络和控制网络的一体化。

以太网仅仅是一个通信平台，它包括 ISO 开放系统互联模型的 7 层模型中的底部两层，即物理层和数据链路层。即使增加上面两层的 TCP 和 IP，也不是可以互操作的通信协议。

2．令牌总线

IEEE 802 标准的工厂媒体访问技术是令牌总线，其编号为 802.4。它吸收了通用汽车公司支持的制造自动化协议（Manufacturing Automation Protocol，MAP）的内容。

在令牌总线中，媒体访问控制是通过传递一种称为令牌的控制帧来实现的。按照逻辑顺序，令牌从一个装置传递到另一个装置，传递到最后一个装置后，再传递给第一个装置，如此周而复始，形成一个逻辑环。令牌有“空”和“忙”两个状态，令牌网开始运行时，由指定的站产生一个空令牌沿逻辑环传送。任何一个要发送信息的站都要等到令牌传给自己，判断为空令牌时才能发送信息。发送站首先把令牌置成“忙”，并写入要传送的信息、发送站名和接收站名，然后将载有信息的令牌送入环网传输。令牌沿环网循环一周后返回发送站时，如果信息已被接收站复制，发送站将令牌置为“空”，送上环网继续传送，以供其他站使用。

如果在传送过程中令牌丢失，则由监控站向网内注入一个新的令牌。

令牌传递式总线能在很重的负荷下提供实时同步操作，传输效率高，适于频繁、少量的数据传送，因此它最适合于需要进行实时通信的工业控制网络系统。

3．令牌环

令牌环媒体访问方案是 IBM 公司开发的，它在 IEEE 802 标准中的编号为 802.5，有些类似于令牌总线。在令牌环上，最多只能有一个令牌绕环运动，不允许两个站同时发送数据。令牌环从本质上看是一种集中控制式的环，环上必须有一个中心控制站负责网络的工作状态的检测和管理。

4．主从通信方式

主从通信方式是 PLC 常用的一种通信方式。主从通信网络只有一个主站，其他的站都是从站。在主从通信中，主站是主动的，主站首先向某个从站发送请求帧（轮询报文），该从站接收到后才能向主站返回响应帧。通常主站按事先设置好的轮询表的排列顺序对从站进行周期性的查询，并分配总线的使用权。每个从站在轮询表中至少要出现一次，对实时性要求较高的从站可以在轮询表中出现几次，还可以用中断方式来处理紧急事件。

PROFIBUS-DP 的主站之间的通信为令牌方式，主站与从站之间为主从方式。

1.1.3 现场总线及其国际标准

1．现场总线的基本概念

IEC（国际电工委员会）对现场总线（Fieldbus）的定义是“安装在制造和过程区域的现场装置与控制室内的自动控制装置之间的数字式、串行、多点通信的数据总线”。它是当前工业自动化的热点之一。现场总线以开放的、独立的、全数字化的双向多变量通信取代 4～20mA 现场模拟量信号。现场总线 I/O 集检测、数据处理、通信为一体，可以代替变送器、调节器、记录仪等模拟仪表，它不需要框架、机柜，可以直接安装在现场导轨槽上。现场总线 I/O 的接线极为简单，只需一根电缆，从主机开始，沿数据链从一个现场总线 I/O 连接到下一个现场总线 I/O。使用现场总线后，可以节约配线、安装、调试和维护等方面的费用，现场总线 I/O 与 PLC 可以组成高性能价格比的 DCS（集散控制系统）。

使用现场总线后，操作员可以在中央控制室实现远程监控，对现场设备进行参数调整，还可以通过现场设备的自诊断功能诊断故障和寻找故障点。

2．IEC 61158

由于历史的原因，现在有多种现场总线标准并存，IEC 的现场总线国际标准（IEC 61158）在 1999 年底获得通过，经过多方的争执和妥协，最后容纳了 8 种互不兼容的协议，这 8 种协议对应于 IEC 61158 中的 8 种现场总线类型：

类型 1：TS 61158，原 IEC 技术报告。

类型 2：ControlNet（美国 Rockwell 公司支持）。

类型 3：PROFIBUS（德国西门子公司支持）。

类型 4：P-Net（丹麦 Process Data 公司支持）。

类型 5：FF 的 HSE（高速以太网，现场总线基金会的 H2，美国 Fisher Rosemount 公司支持）。

类型 6：SwiftNet（美国波音公司支持）。

类型 7：WorldFIP（法国 Alstom 公司支持）。

类型 8：Interbus（德国 Phoenix contact 公司支持）。

2000 年又补充了两种类型：

类型 9：FF H1（美国 Fisher Rosemount 公司支持）；

类型 10：PROFINET（西门子公司支持）。

由于以太网应用非常普及，产品价格低廉，硬件软件资源丰富，传输速率高（工业控制网络已经在使用 1000 Mbit/s 以太网），网络结构灵活，可以用软件和硬件措施来解决响应时间不确定性的问题，各大公司和标准化组织纷纷提出了各种提升工业以太网实时性的解决方案，从而产生了实时以太网（Real Time Ethernet，RTE）。

2007 年 7 月出版的 IEC 61158 第 4 版采纳了经过市场考验的 20 种现场总线（见表 1-1）。

其中的类型 1 是原 IEC 61158 第 1 版技术规范的内容，类型 2 通用工业协议（Common Industry Protocol，CIP）包括 DeviceNet、ControlNet 和实时以太网 Ethernet/IP。类型 6 SwiftNet 因为市场应用很不理想，已被撤消。

EPA（Ethernet for Plant Automation，用于工厂自动化的以太网）是我国拥有自主知识产

权的实时以太网通信标准，已被列入现场总线国际标准 IEC 61158 第 4 版的类型 14。

表 1-1　IEC 61158 第 4 版的现场总线类型

类　型	技 术 名 称	类　型	技 术 名 称
类型 1	TS61158 现场总线	类型 11	TC net 实时以太网
类型 2	CIP 现场总线	类型 12	Ether CAT 实时以太网
类型 3	PROFIBUS 现场总线	类型 13	Ethernet Powerlink 实时以太网
类型 4	P-Net 现场总线	类型 14	EPA 实时以太网
类型 5	FF HSE 高速以太网	类型 15	Modbus RTPS 实时以太网
类型 6	SwiftNet（已被撤消）	类型 16	SERCOS Ⅰ、Ⅱ现场总线
类型 7	WorldFIP 现场总线	类型 17	VNET/IP 实时以太网
类型 8	Interbus 现场总线	类型 18	CC-Link 现场总线
类型 9	FF H1 现场总线	类型 19	SERCOS Ⅲ实时以太网
类型 10	PROFINET 实时以太网	类型 20	HART 现场总线

3．IEC 62026

IEC 62026 是供低压开关设备与控制设备使用的控制器电气接口标准，于 2000 年 6 月通过。它包括：

IEC 62026-1：一般要求。

IEC 62026-2：执行器传感器接口（Actuator Sensor Interface，AS-i），德国西门子公司支持。

IEC 62026-3：设备网络（Device Network，DN），美国 Rockwell 公司支持。

IEC 62026-4：Lonworks（Local Operating Networks）总线的通信协议 LonTalk，已取消。

IEC 62026-5：智能分布式系统（Smart Distributed System，SDS），美国 Honeywell 公司支持。

IEC 62026-6：串行多路控制总线（Serial Multiplexed Control Bus，SMCB），美国 Honeywell 公司支持。

1.2　SIMATIC 通信网络简介

1.2.1　全集成自动化

传统的自动化系统大多以单元生产设备为核心，进行检测与控制，但是生产设备之间容易形成“自动化孤岛”，缺乏信息资源的共享和生产过程的统一管理，不能满足现代工业生产的要求。

随着市场竞争的加剧，需要一个完整的从现场级到工厂管理级的自控解决方案，来帮助工厂降低成本，提高产品质量，提供更佳的供应链管理，从而提高企业的市场竞争力。

信息技术成为企业成功的决定性因素，用于实现公司的供应链、企业生产现场和管理层之间的信息无缝传输，对提高投资回报、降低运营成本起到了决定性的作用。

为了提高企业的市场竞争力，实现最佳经济效益的目标，必须将自动化控制、制造执行系统（Manufacturing Execute System，MES）和企业资源规划系统（Enterprise Resource Planning，ERP）三者完美地整合在一起。

西门子公司的全集成自动化（Totally Integrated Automation，TIA）正是为顺应现代化控

制系统的变革趋势而产生的。TIA 不仅通过现场总线技术实现了系统自身与现场设备的纵向集成，同时也实现了系统与系统之间的横向联系，使通信覆盖整个企业，确保了现场实时数据的及时、精确和统一。

通过全集成自动化，可以实现从输入物流到输出物流整个生产过程的统一协同自动化，实施完整的生产现场自动化解决方案。

全集成自动化集高度的集成统一性和开放性于一身，标准化的网络体系结构，统一的编程组态环境和高度一致的数据集成，使 TIA 为企业实现了横向和纵向信息集成。

从最初的规划与设计，工程与实施，到安装与调试，运行与维护，以至系统升级改造，TIA 使企业在整个生命周期中获得最高的生产力和产品质量，并显著降低项目成本。此外，TIA 还能缩短产品上市和系统投入运行的时间，从而全面增强企业的核心竞争力。

全集成自动化具有 3 个典型的特征：

1．统一的组态和编程

STEP 7 是全集成自动化的基础，在 STEP 7 中，用项目来管理一个自动化系统的硬件和软件。STEP 7 用 SIMATIC 管理器对项目进行集中管理，它可以方便地浏览 SIMATIC S7、M7、C7 和 WinAC 的数据。实现 STEP 7 各种功能所需的 SIMATIC 软件工具都集成在 STEP 7 中。STEP 7 使系统具有统一的组态和编程方式，统一的数据管理和数据通信方式。

可以用 SIMATIC 管理器来调用编程、组态等工程工具，包括用于项目的创建、管理、保存和归档等基本应用程序。

2．统一的数据管理

以 STEP 7 为操作平台，所有软件组件都访问同一个数据库。这种统一的数据库管理机制，不仅可以减少系统开发的费用，还可以减少出错的概率，提高系统诊断的效率。各软件可以通过全局变量共享一个统一的符号表，在 STEP 7 中定义的变量，通过内部数据库，可以被 HMI（人机界面）的组态软件使用。因此，在一个项目中，只需在一点对变量进行输入和修改。这不仅降低了系统集成的工作量，而且可以避免出现错误。即使多人同时在项目中工作，也可以有效地保证数据的一致性。在工程系统中定义的参数，可以通过网络，向下传输到现场传感器、执行器和驱动器。

3．统一的通信

全集成自动化采用统一的集成通信技术，从公司管理级到现场控制级，使用国际通行的开放的通信标准，例如工业以太网、PROFINET、PROFIBUS、AS-i 等。TIA 支持基于互联网的全球信息流动，用户可以通过传统的浏览器访问控制信息。这样可以确保生产控制过程中采集的实时数据及时、准确、可靠、无间隙地与 MES 保持通信。由于有关的硬件和软件组件也使用这些通信标准，因此极易连接，包括跨系统或跨越不同网络的连接。

借助于西门子公司的全集成自动化系统，所有的自动化结构（直到现场的各个部件）都是清晰的和透明的。因为极其相似的组态和编程工具，以及共享数据的一致性，使得对错误的定位和处理都很容易。这样，用户能快速完成过程的优化、扩展和调整，将生产中断的可能性降到最低。

1.2.2 SIMATIC 网络结构与通信服务简介

大型的工厂自动化网络系统一般采用三级网络结构。

1．现场设备层

现场设备层的主要功能是连接现场设备，例如分布式 I/O、传感器、驱动器、执行机构和开关设备等，完成现场设备控制及设备间的连锁控制。一般来说，现场设备层的传输数据量较小，要求的响应时间为 10～100ms。主站（PLC、PC 或其他控制器）负责总线通信管理以及与从站的通信。总线上所有的设备生产工艺控制程序存储在主站中，并由主站执行。西门子的 SIMATIC NET 网络系统（见图 1-2）的现场设备层主要使用 PROFIBUS-DP。并将执行器和传感器单独分为一层，主要使用 AS-i（执行器-传感器接口）网络。AS-i 的主站与连接到其子网的执行器和传感器进行通信，其特点是对少量数据的毫秒级快速响应。

图 1-2　SIMATICNET

2．车间监控层

车间监控层又称为单元层，用来完成车间主生产设备之间的连接，实现车间级设备的监控。车间级监控包括生产设备状态的在线监控、设备故障报警及维护等。通常还具有诸如生产统计、生产调度等车间级生产管理功能。车间级监控用 PROFIBUS 或工业以太网将 PLC、PC 和 HMI 连接到一起。这一级对数据传输速率要求不高，要求的响应时间为 100ms～1s，但是应能传送大量的信息。

3．工厂管理层

车间管理网作为工厂主网的一个子网，通过交换机、网桥或路由器等连接到厂区主干网，将车间数据集成到工厂管理层。管理层处理的是对于整个系统的运行有重要作用的高级别的任务。除了保存过程值以外，还包括优化和分析过程等功能。

工厂管理层通常采用符合 IEC 802.3 标准的以太网，即 TCP/IP 通信协议标准。

4．西门子的自动化通信网络

S7-300/400 有很强的通信功能，CPU 模块都集成有 MPI（多点接口），有的 CPU 模块还集成有 PROFIBUS-DP、PROFINET 和点对点通信接口，此外还可以使用 PROFIBUS-DP、工业以太网、AS-i 和点对点通信处理器（CP）模块。通过 PROFINET、PROFIBUS-DP 或 AS-i 现场总线，CPU 与分布式 I/O 模块之间可以周期性地自动交换数据。在自动化系统之间，PLC 与计算机和 HMI（人机界面）站之间，均可以交换数据。数据通信可以周期性地自动进行，或者基于事件驱动。

图 1-3 是西门子的工业自动化通信网络的示意图。PROFINET 是基于工业以太网的现场总线，可以高速传送大量的数据。PROFIBUS 用于少量和中等数量数据的高速传送。MPI 是 SIMATIC 产品使用的内部通信协议，用于 PLC 之间、PLC 与 HMI 和 PG/PC（编程设备/计算机）之间的通信，可以建立传送少量数据的低成本网络。点对点通信用于特殊协议的串行通信。AS-i 是底层的低成本网络。

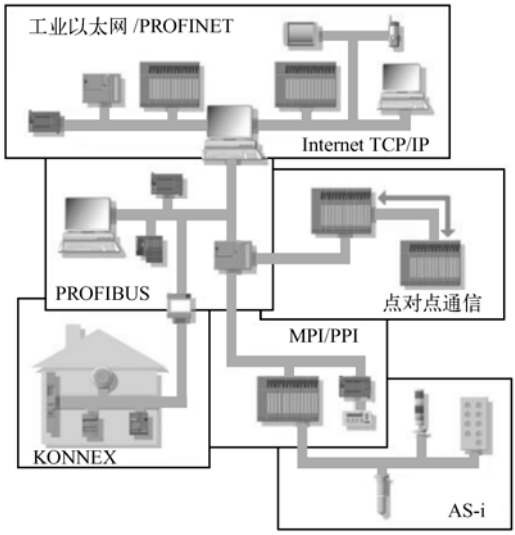


图 1-3　西门子的工业自动化通信网络

通用总线系统 KONNEX（KNX）目前在欧洲常用于楼宇自动控制。PPI（点对点接口）是主要用于 S7-200 的通信协议。

图 1-4 中的 IWLAN 是工业无线局域网的缩写，西门子对应的产品为 SCALANCE W。

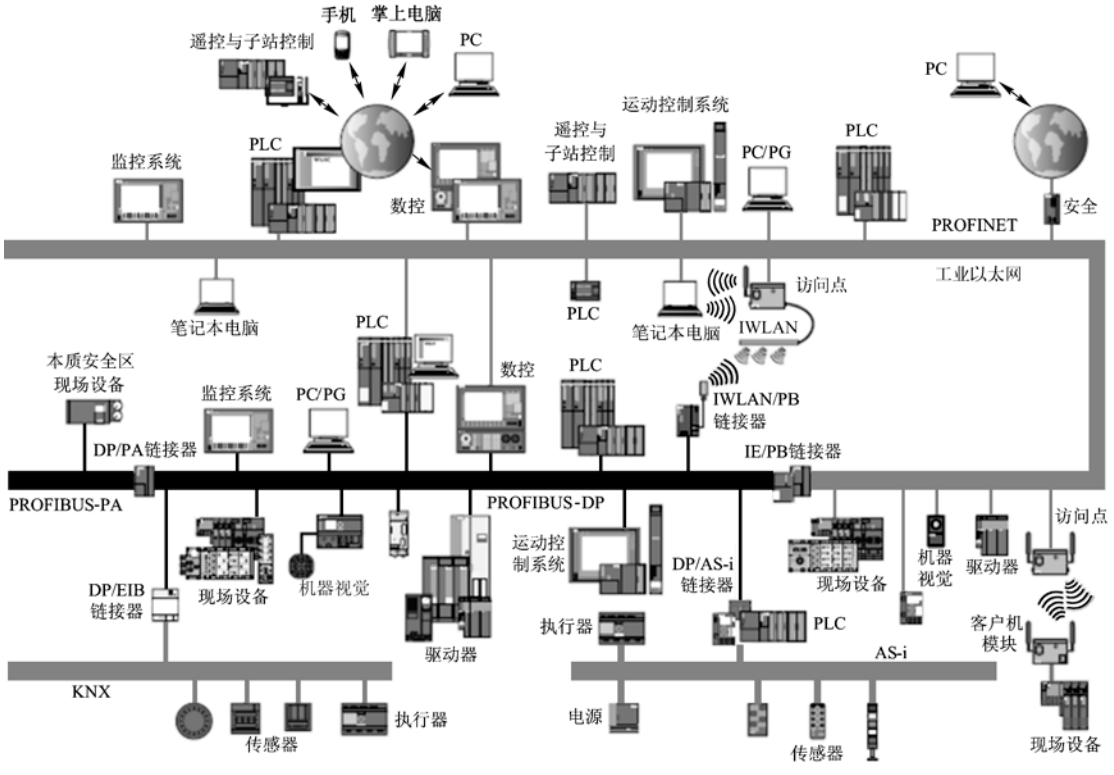


图 1-4　西门子的工业自动化通信网络

各个网络之间用链接器或有路由器功能的 PLC 连接。

5．通信服务

工业以太网、PROFIBUS 和 MPI 都可以提供 PG/OP（编程设备/操作面板）和 S7 通信服务，S7 基本通信和全局数据通信只能用于 MPI。

（1）PG/OP 通信服务

PG/OP 通信服务是集成的通信功能，用于 SIMATIC PLC 与 SIMOTION（西门子运动控制系统）、编程软件（例如 STEP 7）、HMI 设备之间的通信，下载程序和组态数据，执行测试和诊断功能，并通过 OP 实现操作员监控。工业以太网、PROFIBUS 和 MPI 均支持 PG/OP 通信服务。

由于 S7 通信功能内置在 SIMATIC PLC 的操作系统中，可以用 HMI 设备和 PG/PC 访问 PLC 内的数据。也可以用 SFB（系统功能块）和 SFC（系统功能）来产生用于 HMI 设备的报警信息。

PG/OP 通信服务支持 S7 PLC 与各种 HMI 设备或编程设备（包括编程用的 PC）的通信。PG/OP 通信服务提供以下功能：

1）PG/PC 功能：下载、上载硬件组态和用户程序，在线监视 S7 站，以进行测试和诊断。

2）OP 功能：HMI 设备和 PG/PC 使用 OP 功能读取或改写 S7 PLC 的变量，S7 PLC 在通信中是被动的，不用编写通信程序。

（2）S7 路由

S7 路由属于 PG/OP 通信服务功能。通过 S7 路由功能，可以实现跨网络的编程设备通信。PG 可以在某个固定点访问所有在 S7 项目中组态的 S7 站点，下载用户程序和硬件组态，或者执行测试和诊断功能。

网关是一个跨接在两个子网上，用来实现两个子网的数据交换的设备。某些型号的 S7 CPU、通信处理器（CP）和 HMI 可以作网关。

（3）S7 通信

S7 通信是用于 SIMATIC S7/C7 的优化的通信功能。它用于 S7 PLC 之间、S7 PLC 和 PC 之间的通信。在每个任务中，最多可以传送 64 KB 数据。S7 通信服务可以用于 MPI、PROFIBUS 和工业以太网。

（4）S5 兼容通信

S5 兼容通信用于 S7 PLC 之间的 PROFIBUS FDL 协议和工业以太网的通信。

（5）标准通信

标准通信使用数据通信的标准化协议 PROFIBUS-FMS（现场总线报文规范）和 OPC。前者已很少使用。

6．工业以太网的通信服务

1）PROFINET IO、PROFINET CBA（基于组件的自动化）通信服务；

2）S5 兼容通信服务，包括 ISO 传输、ISO-on-TCP、UDP、TCP/IP 服务；

3）IT 服务，包括 FTP、E-Mail 和 SNMP 服务。

4）OPC、PG/OP、S7、PROFIdrive、PROFIsafe 通信服务。

7．PROFIBUS 网络的通信服务

1）PROFIBUS-DP、PA、FMS、FDL、PROFIdrive、PROFIsafe 通信服务；

2）PG/OP 和 S7 通信服务。

8．其他网络的通信服务

1）AS-i 网络的通信服务包括接口服务和 ASIsave 服务。

2）MPI 网络的通信服务包括 PG/OP 通信服务、S7 通信服务、全局数据通信和 S7 基本通信服务。

1.2.3 学习网络通信的建议

西门子的通信服务分为两类：

1）不需要编程，只需要组态就可以实现数据传输。例如基于 MPI 网络的全局数据通信、PROFIBUS-DP 主站和标准从站之间的通信。

2）既需要组态，也需要编程。通过组态和调用系统功能、系统功能块和通信用的功能、功能块来实现通信。

S7-300/400 的仿真软件 PLCSIM 只能对 PROFIBUS-DP 从站的某些故障仿真，较高的版本可以对 S7 通信仿真。PLCSIM 对通信的仿真能力有限，通信过程一般需要用硬件实验来验证，但是绝大多数读者都没有做大量的硬件实验的条件。在没有实验条件的情况下，可以用 STEP 7 来练习通信网络的组态和编程。

STEP 7 主要用硬件组态工具 HW Config 和网络组态工具 NetPro 来组态通信网络。网络的组态是“可视化”的，可以在组态时形象地看到网络的结构，设置网络和各个站点的参数。

可以通过查阅产品目录和有关的手册，了解 CPU 集成的通信接口和通信处理器的通信功能。组态时选中硬件目录中的某个组件，可以在下面的小窗口看到该组件主要的性能指标。

组态工具提供了非常强的防止误操作的措施。组态时某些菜单项、单选框、复选框、按钮和选择框如果为灰色，表示对于选中的对象（例如 CPU、CP、DP 从站和模块），不能使用这些功能，从而可以有效地防止组态错误。

组态结束后，点击 HW Config 或 NetPro 工具栏上的“保存和编译”按钮  ，如果有组态错误或警告信息，将会用对话框显示出来。应改正所有的组态错误，系统才能运行，但是警告信息不会影响系统的正常运行。成功的组态是实现网络通信的必要条件。

本书的随书光盘提供了上百个网络通信的例程，绝大多数例程经过硬件调试检验通过，正文对大多数例程作了较详细的说明。读者可以一边看书，一边用 STEP 7 打开对应的例程，通过例程了解详细的组态和编程的方法。可以根据正文介绍的组态的步骤和方法，同时参考光盘中的例程，做组态和编程的练习。这样可以较快地掌握通信网络的组态和编程的方法。当然，在有条件的情况下，通过必要的硬件实验来学习通信网络的调试方法，积累必不可少的经验，也是很有必要的。

为了避免重复、减少篇幅，本书对第一次出现的组态过程和通信程序作了详细的介绍，后面的章节涉及到类似的组态过程和程序时，一般只作简单的说明。建议对硬件组态和网络组态不太熟悉的读者，从第 3 章开始，按书上的顺序阅读组态过程和作组态的练习。

限于篇幅，本书对硬件选型、组网和硬件安装方面的内容介绍得较少。读者可以查阅随书光盘的文件夹“产品样本”中的文件获取有关的信息。

1.3 练习题

1．计算机通信有哪些主要的国际标准？
2．全集成自动化有什么特点？有什么优点？
3．SIMATIC 网络主要由哪些网络组成？
4．SIMATIC 网络有哪些主要的通信服务？

第 2 章　PROFIBUS 的硬件组成与通信协议

2.1　PROFIBUS 的结构与硬件

PROFIBUS 是目前国际上通用的现场总线标准之一，它以其独特的技术特点、严格的认证规范、开放的标准、众多厂商的支持，被纳入现场总线的国际标准 IEC 61158。PROFIBUS 于 2006 年 10 月成为我国首个现场总线国家标准（GB/T 20540-2006）。

PROFIBUS 是不依赖生产厂家的、开放式的现场总线，各种各样的自动化设备均可以通过同样的接口交换信息。PROFIBUS 可以用于分布式 I/O 设备、传动装置、PLC 和基于 PC（个人计算机）的自动化系统。

PROFIBUS 技术是唯一可以满足两类通信应用（制造业和过程工业应用）的现场总线。

PROFIBUS 在 IEC 61158 中称为类型 3，PROFIBUS 的基本部分称为 PROFIBUS-V0。在 2002 年新版的 IEC 61158 中增加了 PROFIBUS-V1、PROFIBUS-V2 和 RS-485IS 等内容。新增的 PROFINET 规范作为 IEC 61158 的类型 10。

2.1.1　PROFIBUS 简介

PROFIBUS 协议主要由 3 部分组成：PROFIBUS-DP、PROFIBUS-PA 和 PROFIBUS-FMS。

1．PROFIBUS-DP

DP 是 Decentralized Periphery（分布式外部设备）的缩写。PROFIBUS-DP（简称为 DP）主要用于制造业自动化系统中单元级和现场级通信，特别适合于 PLC 与现场级分布式 I/O 设备之间的快速循环数据交换。DP 是 PROFIBUS 中应用最广的通信方式。

PROFIBUS-DP 用于连接下列设备：PLC、PC 和 HMI 设备；分布式现场设备，例如 SIMATIC ET 200 和变频器等设备（见图 2-1）。PROFIBUS-DP 的响应速度快，很适合在制造业使用。

作为 PLC 硬件组态的一部分，分布式 I/O（例如 ET 200）用 STEP 7 来组态。通过供货方提供的 GSD 文件，可以用 STEP 7 将其他制造商生产的从站设备组态到网络中。

有的 S7-300/400 CPU 配备有集成的 DP 接口，S7-200/300/400 也可以通过通信处理器（CP）连接到 PROFIBUS-DP。

2．PROFIBUS-PA

PA 是 Process Automation（过程自动化）的缩写。PROFIBUS-PA 用于 PLC 与本质安全系统的过程自动化的现场传感器和执行器的低速数据传输，特别适合于过程工业使用。PROFIBUS-PA 功能集成在起动执行器、电磁阀和测量变送器等现场设备中（见图 2-2）。

PROFIBUS-PA 由于采用了 IEC 1158-2 标准，确保了本质安全和通过屏蔽双绞线电缆进行数据传输和供电，可以用于防爆区域的传感器和执行器与中央控制系统的通信。

PA 设备可以在下列防爆区域运行：

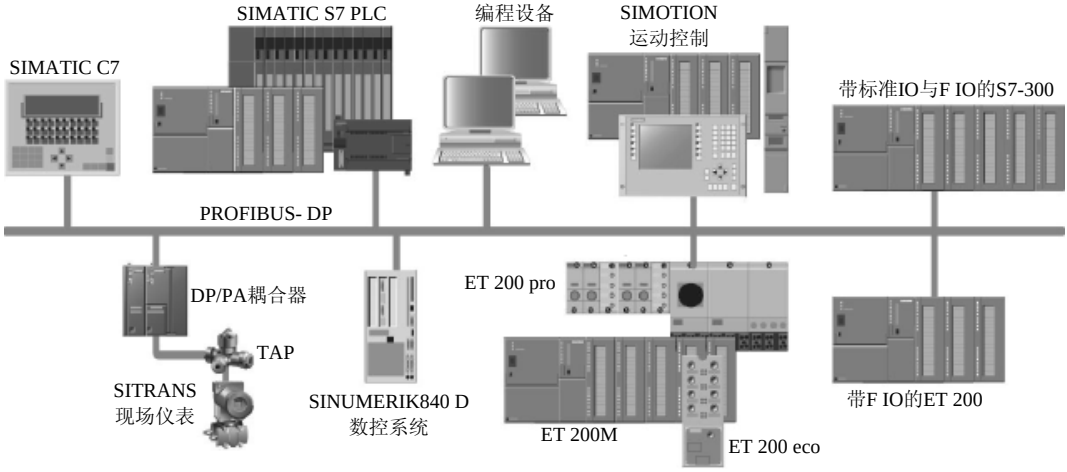


图 2-1　PROFIBUS-DP

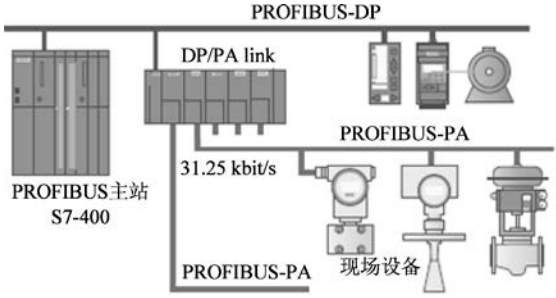


图 2-2　PROFIBUS-PA

Zone 0: 危险的瓦斯气体经常或长期存在的区域。

Zone 1: 在正常运行期间，有可能存在危险的瓦斯气体的区域。

Zone 2: 不希望在正常运行期间存在危险的瓦斯气体的区域。

传感器/执行器安装在生产现场，而耦合器和控制器等设备则安装在控制室内。即使总线上的设备不在危险现场，也必须通过适当的结构保证它们的本质安全特性。使用 DP/PA 耦合器和 DP/PA 链接器，可以将 PROFIBUS-PA 设备很方便地集成到 PROFIBUS-DP 网络中。

PROFIBUS-PA 的组态、编程与故障诊断的方法将在第 9 章介绍。

3．PROFIBUS-FMS

FMS 是 Field Message Specification（现场总线报文规范）的缩写，用于系统级和车间级不同供应商的自动化系统之间交换过程数据，处理单元级（PLC 和 PC）的多主站数据通信。

PROFIBUS-FMS 定义了主站与主站之间的通信模型，它使用 OSI 七层模型的第 1、2 层和第 7 层。

S7-300/400 使用通信 FB 来实现 FMS 服务，用 STEP 7 组态 FMS 静态连接来发送和接收数据。PROFIBUS-FMS 已经基本上被以太网通信取代，现在很少使用。

2.1.2 PROFIBUS 的物理层

ISO/OSI 参考模型的物理层是第 1 层，PROFIBUS 可以使用多种通信媒体，例如带屏蔽的双绞线、光纤、红外线、导轨以及混合方式。传输速率为 9.6kbit/s～12 Mbit/s，每个 DP 从站的输入数据和输出数据最大为 244B。使用屏蔽双绞线电缆时最长通信距离为 9.6km，使用光缆时最长通信距离为 90km，最多可以连接 127 个从站。

PROFIBUS 可以使用灵活的拓扑结构，支持线形、树形、环形结构以及冗余的通信模型。支持基于总线的驱动技术和符合 IEC 61508 的总线安全通信技术。下面介绍用于 DP 和 FMS 的 RS-485 传输和光纤传输。

1．DP/FMS 的 RS-485 传输

PROFIBUS-DP 和 PROFIBUS-FMS 使用相同的传输技术和统一的总线存取协议，可以在同一根电缆上同时运行。DP/FMS 符合 EIA RS-485 标准（也称为 H2），采用价格便宜的屏蔽双绞线电缆，电磁兼容性（EMC）条件较好时也可以使用不带屏蔽的双绞线电缆。一个总线段的两端各有一套有源的总线终端电阻。

图 2-3 中 A、B 线之间是 220Ω 终端电阻，根据传输线理论，终端电阻可以吸收网络上的反射波，有效地增强信号强度。两端的终端电阻并联后的值应基本上等于传输线相对于通信频率的特性阻抗。390Ω 的下拉电阻与数据基准电位 DGND 相连，上拉电阻与 DC 5V 电压的正端（VP）相连。在总线上没有站发送数据（即总线处于空闲状态）时，上拉电阻和下拉电阻用于确保 A、B 线之间有一个确定的空闲电位。

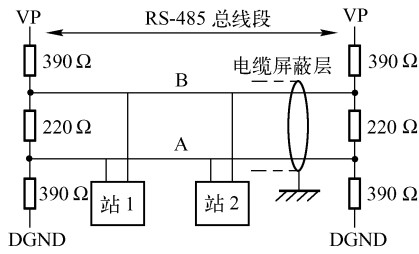


图 2-3　DP/FMS 总线段的结构

大多数 PROFIBUS 总线连接器都集成了终端电阻，连接器上的开关在 On 位置时终端电阻被连接到网络上，开关在 Off 位置时终端电阻从网络上断开。每个网段两端的站必须接入终端电阻，中间的站不能接入终端电阻。

传输速率为 9.6 kbit/s～12 Mbit/s，所选的传输速率用于总线段上的所有设备。传输速率大于 1.5 Mbit/s 时，由于连接的站的电容性负载引起导线反射，必须使用附加有轴向电感的总线连接插头。

PROFIBUS 的站地址空间为 0～127，其中的 127 为广播用的地址，所以最多能连接 127 个站点。一个总线段最多 32 个站，超过了必须分段，段与段之间用中继器连接。中继器没有站地址，但是被计算在每段的最大站数中。

每个网段的电缆最大长度与传输速率有关（见表 2-1）。

表 2-1　传输速率与总线长度的关系

传输速率/kbit/s	9.6～187.5	500	1500	3000～12000
总线长度/m	1000	400	200	100

RS-485 采用半双工、异步的传输方式，PROFIBUS 的 1 个字符帧由 8 个数据位、1 个起始位、1 个停止位和 1 个奇偶校验位组成。

2．D 型总线连接器

PROFIBUS 标准推荐总线站与总线的相互连接使用 9 针 D 型连接器。连接器的引脚分配

如表 2-2 所示。

表 2-2　D 型连接器的引脚分配

针脚号	信 号 名 称	说　　明	针脚号	信 号 名 称	说　　明
1	SHIELD	屏蔽或功能地	6	VP	供电电压正端
2	M24	24V 辅助电源输出的地	7	P24	24V 辅助电源输出正端
3	RXD/TXD-P	接收/发送数据的正端，B 线	8	RXD/TXD-N	接收/发送数据的负端，A 线
4	CNTR-P	方向控制信号正端	9	CNTR-N	方向控制信号负端
5	DGND	数据基准电位（地）			

在传输期间，A 线和 B 线对“地”（DGND）的电压波形相反。信号为 1 时 B 线为高电平，A 线为低电平。各报文间的空闲（Idle）状态对应于二进制“1”信号。

3．DP/FMS 的光纤电缆传输

PROFIBUS 可以通过光纤中光的传输来传送数据。单芯玻璃光纤的最大连接距离为 15km，价格低廉的塑料光纤为 80m。光纤电缆对电磁干扰不敏感，并能确保站与站之间的电气隔离。近年来，由于光纤的连接技术大为简化，这种传输技术已经广泛地用于现场设备的数据通信。许多厂商提供专用总线插头来转换 RS-485 信号和光纤信号。

光链路模块（OLM）用来实现单光纤环和冗余的双光纤环。在单光纤环中，OLM 通过单工光纤电缆相互连接，如果光纤电缆断线或 OLM 出现故障，整个环路将崩溃。在冗余的双光纤环中，OLM 通过两个双工光纤电缆相互连接，如果两根光纤线中的一根出了故障，总线系统将自动地切换为线性结构。光纤导线中的故障排除后，总线系统返回正常的冗余环状态。

2.1.3　PROFIBUS-DP 设备的分类

PROFIBUS 网络的硬件由主站、从站、网络部件和网络组态与诊断工具组成。网络部件包括通信媒体（电缆）、总线连接器、中继器、耦合器，以及用于连接串行通信、以太网、AS-i、EIB 等网络系统的网络链接器。

PROFIBUS-DP 设备分为以下 3 种不同类型的站：

1．1 类 DP 主站

1 类 DP 主站（DPM1）是系统的中央控制器，DPM1 在预定的周期内与分布式的站（例如 DP 从站）循环地交换信息，并对总线通信进行控制和管理。DPM1 可以发送参数给 DP 从站，读取从站的诊断信息，用全局控制命令将它的运行状态告知给各从站。此外，还可以将控制命令发送给个别从站或从站组，以实现输出数据和输入数据的同步。下列设备可以作 1 类 DP 主站：

1）一个带集成 DP 接口或插入式接口子模块的 CPU，例如 CPU 315-2DP 和 CPU 417。

2）CPU 和支持 DP 主站功能的通信处理器（CP），例如 CP 342-5。

3）一个连接在 CPU 上的接口模块，例如 IM 467。

4）连接工业以太网和 PROFIBUS-DP 的 IE/PB 链接器模块。

5）ET 200S 和 ET 200X 的主站模块。

6）使用 PROFIBUS 网卡的 PC，例如 WinAC 控制器。

2．2 类 DP 主站

2 类 DP 主站（DPM2）是 DP 网络中的编程、诊断和管理设备。DPM2 除了具有 1 类主站的功能外，在与 1 类 DP 主站进行数据通信的同时，可以读取 DP 从站的输入/输出数据和当前的组态数据，可以给 DP 从站分配新的总线地址。下列设备可以作 DPM2：

（1）以 PC 为硬件平台的 2 类主站

PC 加 PROFIBUS 网卡可以做 2 类主站。西门子公司可以提供专用的编程设备，不过现在一般都用 PC 和 STEP 7 编程软件来作编程设备，用 PC 和 WinCC 等组态软件作监控操作站。

（2）操作员面板/触摸屏（OP/TP）

操作员面板（OP）和触摸屏（TP）用于操作人员对系统的控制和操作，例如参数的设置与修改、设备的起动和停机，以及在线监视设备的运行状态等。它们在工业控制中得到了广泛的应用。西门子公司提供多种型号的 TP 和 OP 供用户选用。

3．DP 从站

DP 从站是 PROFIBUS 网络上的被动节点，是低成本的 I/O 设备，用于输入信息的采集和输出信息的发送，DP 从站只与它的 DP 主站交换用户数据，向主站报告本地诊断中断和过程中断。典型的从站设备是传感器、执行器和测量变送器。在 DP 通信过程中，从站是被动的。

非智能型的分布式 I/O 没有程序存储和程序执行功能，通信适配器或接口模块用来接收主站的指令，按主站指令驱动 I/O，并将 I/O 输入及故障诊断等信息返回给主站。

（1）标准 DP 从站

西门子的 ET 200 是非智能的标准 DP 从站，将在 2.1.5 节介绍。

（2）PLC 智能 DP 从站与通信处理器

PLC 可以作为 PROFIBUS 的智能从站（I 从站）。DP 主站不是直接访问智能从站的 I/O 模块，而是通过从站组态时指定的通信双方的输入、输出地址区来交换数据。某些 PROFIBUS 通信处理器（CP）也可以作 DP 从站。

（3）具有 PROFIBUS-DP 接口的其他现场设备

西门子的 SINUMERIK 数控系统、SITRANS 现场仪表、变频器和直流传动装置都有 PROFIBUS-DP 接口或可选的 DP 接口卡，可以作 DP 从站。其他公司支持 DP 接口的输入/输出、传感器、执行器或其他智能设备，也可以接入 PROFIBUS-DP 网络。

可以将 1 类、2 类 DP 主站或 DP 从站组合在一个设备中，形成一个 DP 组合设备。

2.1.4 PROFIBUS 通信处理器

1．通信处理器的作用

通信处理器（CP）用于将 SIMATIC PLC 连接到 PROFIBUS 网络，可以用于恶劣的工业环境和较宽的温度范围。通信处理器允许标准 S7 通信、S5 兼容通信以及 PG/OP 通信。它们减轻了主 CPU 的通信任务，提高了通信的效率和可靠性。

通信处理器可以扩展 PLC 的过程 I/O，实现 SYNC/FREEZE（同步/冻结）和恒定总线周期功能。通信处理器和集成在 STEP 7 的 NCM S7 有很强的诊断功能。通过 S7 路由功能，可以实现不同网络之间的通信。不需要编程器就可以更换 CP 模块。

CP 443-5 有时间同步功能，可以在 H 系统中实现冗余的 S7 通信或 DP 主站通信。CP 443-5 扩展型允许在运行过程中添加分布式 I/O。

2．PLC 的 PROFIBUS 通信处理器

S7-200 的 PROFIBUS 通信处理器为 EM 277，在网络中只能作从站。

S7-300 的 PROFIBUS 通信处理器为 CP 342-5、CP 343-5 和有光纤接口的 CP 342-5 FO。

S7-400 的 PROFIBUS 通信处理器为 CP 443-5 基本型、CP 443-5 扩展型、IM 467 和 IM 467-FO。CP443-5 支持冗余的总线拓扑结构。

3．用于计算机的 PROFIBUS 通信处理器

（1）不带微处理器的通信处理器

不带微处理器的通信处理器价格较低，可以作 1 类、2 类 PROFIBUS-DP 主站或 DP 从站。作为运行 STEP 7 和 NCM PC 的编程器接口，可以用于工业环境。

CP 5611 和 CP 5621 用于台式计算机（PCI 总线），CP 5511 和 CP 5512 用于带有 PCMCIA 插槽的笔记本电脑。它们支持 PROFIBUS-DP 和 MPI，可以作 PROFIBUS-DP 主站或从站，有 PG/OP 和 S7 通信功能。

（2）带微处理器的通信处理器

带微处理器的通信处理器可以通过双端口 RAM 快速访问过程数据，以减轻主站 CPU 的负载，提高工控机的计算性能。OPC 作为标准接口，其服务器软件包已包含在通信软件的供货范围内。通过即插即用和诊断工具，可以缩短调试时间。CP 支持等时线模式。

CP 5613 和 CP 5613 FO 有一个 PROFIBUS 接口，CP 5614 和 CP 5614 FO 有两个 PROFIBUS 接口，它们均支持 DP 主站、PG/OP 和 S7 通信功能。

2.1.5 ET 200

西门子的 ET 200 是基于现场总线 PROFIBUS-DP 或 PROFINET 的分布式 I/O，可以与经过认证的非西门子公司生产的 PROFIBUS-DP 主站协同运行。

全集成自动化概念和 STEP 7 使 ET 200 能与西门子的其他自动化系统协同运行，实现了从硬件配置到共享数据库等所有层次上的集成。

ET 200 只需要很小的空间，能使用体积更小的控制柜。集成的连接器代替了过去密密麻麻、杂乱无章的电缆，加快了安装过程，紧凑的结构使成本大幅度降低。

某些型号的 ET 200 能在非常恶劣的环境（例如酷热、严寒、潮湿或多粉尘）中使用，有的提供连接光纤的接口，可以节省费用昂贵的抗电磁干扰措施。

在组态时，STEP 7 自动分配紧凑型 DP 从站和模块式 DP 从站的输入/输出地址。就像访问主站主机架上的输入/输出模块一样，DP 主站的 CPU 通过 DP 从站的地址直接访问它们。因此使用标准 DP 从站不会增加编程的工作量。

1．安装在控制柜内的 ET 200

（1）ET 200S

ET 200S（见图 2-4）是一种多功能按位模块化的 I/O 系统，可以配备 PROFIBUS-DP 和 PROFINET 接口模块，可以提供集成光纤接口。模块的种类丰富，有数字量 I/O 模块、模拟量 I/O 模块、技术功能模块、电动机起动器和变频器、IQ-Sense（智能传感器）模块、气动接口模块、故障安全模块。每个站最多可以使用 63 个 I/O 模块，或 20 个最大 7.5kW 的电动机起动器、最大 4kW 的变频器。有 2、4、8、32 点的 I/O 模块。能在运行时更换 I/O 模块（有热插拔功能），可以用于危险区域 Zone 2。

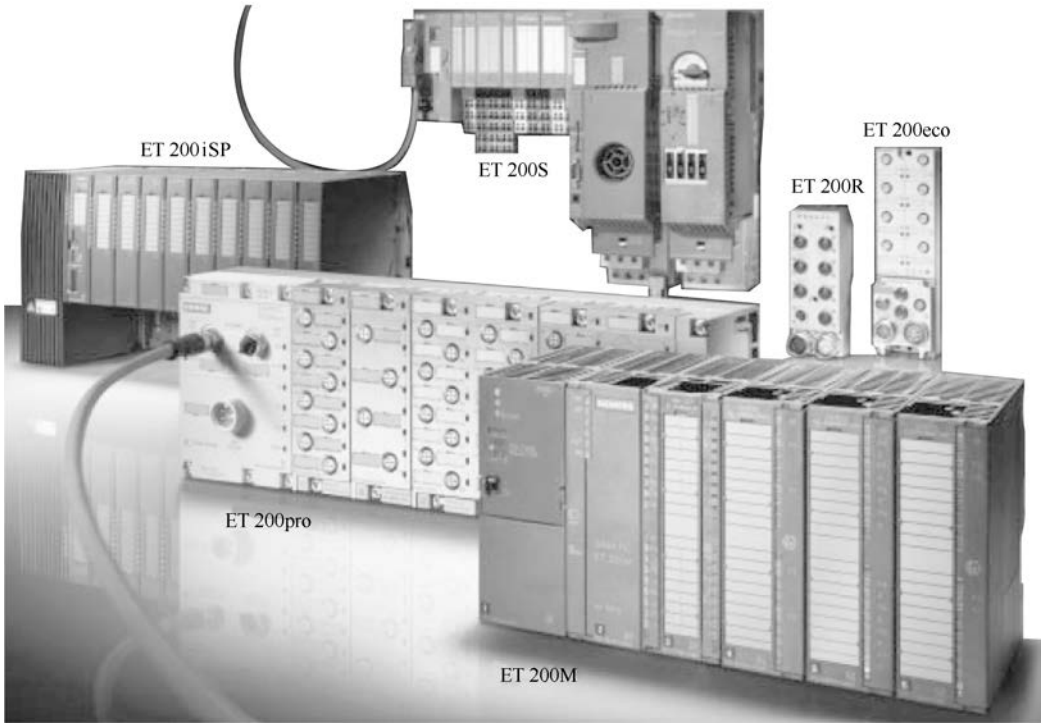


图 2-4　ET 200

ET 200S COMPACT（紧凑型）有 32 点数字量 I/O，可以扩展 12 个 ET 200S 的 I/O 模块。IM 151-7 CPU 接口模块的功能与 CPU 314 相当。使用 Y-Link，可以将 ET 200S 用于容错系统。图 2-4 中的 ET 200S 配有 I/O 模块、电动机起动器和变频器。

（2）ET 200M

ET 200M 是多通道模块化的分布式 I/O，使用 S7-300 全系列模块，适用于大点数、高性能的应用。最多可以扩展 8 个模块，新型号的 ET 200M 可以扩展 12 个模块，用接口模块 IM 153 来实现与主站的通信。

ET 200M 可以提供与 S7-400H 系统相连的冗余接口模块和故障安全型 I/O 模块。可以用于 Zone 2 的危险区域中，传感器和执行器可以用于 Zone 1。有可以带电热插拔的模块，可以在运行中修改组态。

S7-400 的 I/O 模块平均每点的价格比 S7-300 的贵得多，较大型的控制系统常用功能强大的 S7-400 的 CPU 和 ET 200M 来组成系统，这样可以使用价格便宜的 S7-300 的模块，使系统具有很高的性能价格比。这是一种常见的硬件配置方案。

ET 200M 的 SIPLUS 版本是为户外应用设计的，其环境温度范围可达−25～+60°C。

（3）ET 200iSP

ET 200iSP 是本质安全 I/O 系统，适用于有爆炸危险的区域。模块化 I/O 可以直接安装在 Zone 1，可以连接来自 Zone 0 的本质安全的传感器和执行器。

ET 200iSP 可以扩展多种端子模块，有热插拔功能，最多可以插入 32 块电子模块。ET 200iSP 也可以用于容错系统的冗余运行。ET 200iSP 有支持 HART（可寻址远程传感器高速通

道的开放通信协议）的模块，可以将 HART 仪表接入现场总线。

2．不需要控制柜的 ET 200

由于有很高的保护等级，具有抗冲击、防尘和不透水性，ET 200pro、ET 200eco 和 ET 200R 能适应恶劣的工业环境，可以用于没有控制柜的 I/O 系统。它们只需要很少的附加部件。

ET 200 无控制柜系统安装在一个坚固的玻璃纤维加强塑壳内，耐冲击和污物，不透水。而且附加部件少，节省布线，响应快。

（1）ET 200pro

ET 200pro 是防护等级为 IP 65/67 的多功能模块化分布式 I/O，可以直接安装在现场，有一个 DP 接口或一个 PROFINET 接口。ET 200pro 由接口模块和最多 16 个 I/O 模块组成，可以扩展数字量和模拟量 I/O 模块、电动机起动器、变频器、安全系统和气动模块等，支持故障安全功能。ET 200pro 可以热插拔和独立布线。

（2）ET 200eco

ET 200eco 是可以直接安装在现场的一体化经济实用的 I/O，能在运行时更换模块，不会中断总线或供电。

（3）ET 200R

ET 200R 是直接安装在现场的加固型 I/O，特别适合在机器人所处的恶劣的工业环境中使用，坚固的金属外壳使它能抗强电磁干扰和抗焊接火花的飞溅。ET 200R 最多有 32 个 I/O 点。

3．其他型号的 ET 200

（1）ET 200L

ET 200L 是一种小巧、紧凑、价格低廉的分布式 I/O 模块，适用于狭小的场合，可以十分方便地安装在 DIN 导轨上。ET 200L 的端子模块用于接线，有多种 16 点和 32 点的数字量 I/O 电子模块。

（2）ET 200B

ET 200B 是紧凑的分布式 I/O，具有不可更改的固定的输入区和输出区。提供不同电压范围和不同通道数的数字量和模拟量 I/O 模块。

2.1.6 其他网络部件与 GSD 文件

1．RS-485 中继器

下列情况需要使用 RS-485 中继器：多于 32 个站（包含中继器）连接到总线上，或者超过了网段允许的最大长度（与传输速率有关）。

RS-485 中继器用于将 PROFIBUS 网络中的两段总线连在一起，以增加站点的数目。中继器用于信号恢复和总线段之间的电气隔离，最高传输速率为 12 Mbit/s。两个节点之间最多可以安装 9 个中继器。不需要对 RS-485 中继器组态，但是在计算总线参数时应考虑它。

2．诊断中继器

诊断中继器是 RS-485 中继器，传输速率为 9.6 kbit/s～12 Mbit/s，它用于在 RS-485 系统中连接 PROFIBUS-DP 网段，此外还提供铜质总线电缆的物理在线监控，可以侦测总线系统的拓扑结构，在出现故障时，可以自动检测故障类型和故障位置（精确到米）。可以发送诊断报文到 DP 主站，包括详细的故障类型和故障位置数据。

对于 DP 主站来说，诊断中继器相当于一个 DP 从站。诊断中继器用 STEP 7 来组态，也可以作为普通中继器使用，其详细的使用方法见 6.6 节。

3．DP/DP 耦合器

DP/DP 耦合器用来将两条 PROFIBUS 子网络连接在一起，在 DP 主站之间交换数据。这两个子网络在电气上是隔离的，它们可以有不同的传输速率。在两个子网络中，DP/DP 耦合器对于每个 DP 主站来说，都是一个可以自由选择站地址的 DP 从站。可以交换的最大输入、输出数据均为 244B。DP/DP 耦合器用 STEP 7 来组态。

DP/DP 耦合器连续不断地将一个网络的输出数据复制到另一个网络的输入数据，反之亦然。耦合器用顶部的两个 DIP 开关来设置 DP 地址。

4．GSD 文件

GSD（General Station Description，常规站说明）文件是可读的 ASCII 码文本文件，包括通用的和与设备有关的通信的技术规范。为了将不同厂家生产的 PROFIBUS 产品集成在一起，生产厂家必须以 GSD 文件的方式提供这些产品的功能参数，例如 I/O 点数、诊断信息、传输速率、时间监视等。GSD 文件分为 3 个部分：

1）总规范：包括生产厂商和设备名称、硬件和软件版本、传输速率、监视时间间隔、总线连接器的信号分配等。

2）主站规范：包括适用于主站的各项参数，例如最大可以连接的从站个数和上载/下载的选项。

3）与 DP 从站有关的规范：例如 I/O 通道个数、类型和诊断数据等。

可以在制造商的网站下载 GSD 文件，在西门子自动化与驱动集团的中文网站的下载中心搜索“GSD”后，可以下载需要的 GSD 文件。

在 STEP 7 的 SIMATIC 管理器中打开硬件组态工具 HW Config，如果硬件目录窗口没有需要的 DP 从站，应安装制造商提供的 GSD 文件。

在硬件组态工具中，执行菜单命令“选项”→“安装新 GSD 文件”，在出现的对话框中点击“浏览”按钮，打开 GSD 文件所在的文件夹，安装 GSD 文件。安装成功后，在硬件目录窗口的“\PROFIBUS-DP”文件夹中，可以找到刚安装了 GSD 文件的 DP 从站，并将它用于组态。STEP 7 将 GSD 文件存储在“…\Siemens\Step7\S7 DATA\GSD”文件夹中。

2.2 PROFIBUS 的通信协议

2.2.1 PROFIBUS 的数据链路层

PROFIBUS 的协议结构见图 2-5，图中，第 2 层称为现场总线数据链路层（Fieldbus Data Link，FDL），规定了总线访问控制、数据安全性以及传输协议和报文的处理。

PROFIBUS 协议的设计满足了媒体控制的两个基本要求：

1）复杂的自动化系统（主站）之间的通信，必须保证在确切限定的时间间隔中，任何一个站点都有足够的时间来完成通信任务。

2）PLC 或 PC 与 I/O 外围设备（从站）之间的通信，应尽可能简单快速地完成数据的实时传输，因通信协议增加的数据传输时间应尽量少。

<table>
<tr><th></th><th>PROFIBUS-DP</th><th>PROFIBUS-FMS</th><th>PROFIBUS-PA</th></tr>
<tr><td rowspan="4">用户接口层</td><td>DP设备行规</td><td rowspan="2">FMS设备行规</td><td>PA设备行规</td></tr>
<tr><td>基本功能与扩展功能</td><td>基本功能与扩展功能</td></tr>
<tr><td>DP用户接口</td><td>应用层接口</td><td>DP用户接口</td></tr>
<tr><td>直接数据链路映像 DDLM</td><td>ALI</td><td>直接数据链路映像DDLM</td></tr>
<tr><td>第7层(应用层)
第3～6层</td><td colspan="2">现场总线报文规范FMS
未使用</td><td></td></tr>
<tr><td>第2层(数据链路层)</td><td colspan="2">现场总线数据链路FDL</td><td>IEC接口</td></tr>
<tr><td>第1层(物理层)</td><td colspan="2">RS-485/光纤</td><td>IEC 1158-2</td></tr>
</table>

图 2-5　PROFIBUS 协议结构

PROFIBUS-DP 采用混合的总线访问控制机制来实现上述目标（见图 2-6）。它包括主站之间的令牌（Token）传递方式和主站与从站之间的主-从方式。令牌实际上是一条特殊的报文，它在所有的主站上循环一周的时间是事先规定的。主站之间构成令牌逻辑环，令牌传递仅在各主站之间进行。令牌按令牌环中各主站地址的升序在各主站之间依次传递。

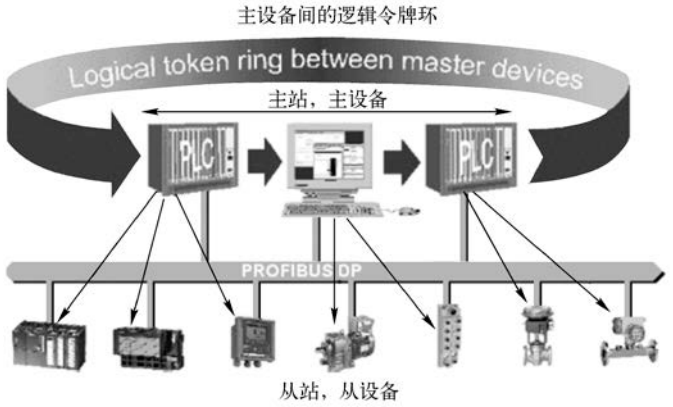


图 2-6　PROFIBUS-DP 的总线访问方式

某个主站得到令牌报文后，该主站可以在一定的时间内执行主站工作。在这段时间内，它可以依照主-从通信关系表与所有的从站通信，也可以依照主-主通信关系表与所有的主站通信。令牌传递程序保证每个主站在一个确切规定的时间内得到总线访问权（即令牌），来访问分配给该主站的从站。主站可以向从站发送数据，接收来自从站的数据。

PROFIBUS-DP 使用第 1、2 层和用户接口层，第 3～7 层未使用，这种精简的结构确保了高速的数据传输。用户接口规定了设备的应用功能、PROFIBUS-DP 系统和设备的行为特性。直接数据链路映像程序 DDLM 提供对第 2 层的访问。

在总线初始化和起动阶段，主站媒体访问控制（MAC）通过辨认主站来建立令牌环，首先自动地判定总线上所有主站的地址，并将它们的节点地址记录在主站表中。在总线运行期间，从令牌环中去掉有故障的主站，将新上电的主站加入到令牌环。

PROFIBUS 媒体访问控制还可以监视传输媒体和收发器是否有故障，检查站点地址是否

出错（例如地址重复），以及令牌是否丢失或有多个令牌。

PROFIBUS 在第 2 层按照非连接的模式操作，除提供点对点逻辑数据传输外，还提供多点通信，其中包括广播及选择广播功能。

DP 主站与 DP 从站之间的通信基于主-从原理，DP 主站按轮询表依次访问 DP 从站，主站与从站间周期性地交换用户数据。DP 主站与 DP 从站之间的一个报文循环由 DP 主站发出的请求帧（轮询报文）和由 DP 从站返回的应答或响应帧组成。

2.2.2 PROFIBUS-DP

在 PROFIBUS 现场总线中，PROFIBUS-DP 的应用最广。DP 协议主要用于 PLC 与分布式 I/O 和现场设备的高速数据通信。典型的 DP 配置是单主站结构，也可以是多主站结构。

DP 的功能经过扩展，一共有 3 个版本：DP-V0、DP-V1 和 DP-V2。有的用户手册将 DP-V1 称为 DPV1。

1．DP-V0 的基本功能

（1）总线访问方法

各主站之间为令牌传送，主站与从站之间为主-从循环传送，支持单主站或多主站系统，总线上最多 126 个站。可以采用点对点用户数据通信、广播方式和循环主-从用户数据通信。

（2）循环数据交换

DP-V0 可以实现中央控制器（PLC、PC 或过程控制系统）与分布式现场设备（从站，例如远程 I/O、阀门、变送器和变频器等）之间的快速循环数据交换，主站发出请求报文，从站收到后返回响应报文。每个从站最多可以传送 224B 的输入或输出。

（3）诊断功能

能对站级、模块级、通道级这 3 级故障进行诊断和快速定位。

（4）保护功能

DP 主站用监控定时器监视与从站的通信，对每个从站都设置有独立的监控定时器。在规定的监视时间间隔内，如果没有执行用户数据传送，监控定时器将会超时，通知用户程序进行处理。DP 从站用监控定时器检测与主站的数据传输，如果在设置的时间内没有完成数据通信，从站自动地将输出切换到故障安全状态。

（5）网络的组态功能与控制功能

通过网络可以实现下列功能：动态激活或关闭 DP 从站，对 1 类 DP 主站进行组态，可以设置站点的数目、DP 从站的地址、输入/输出数据的格式、诊断报文的格式等，以及检查 DP 从站的组态。控制命令可以同时发送给所有的从站或部分从站。

（6）同步与冻结功能

主站可以发送命令给一个从站或同时发送给一组从站。接收到主站的同步命令后，从站进入同步模式。这些从站的输出被保持在当前状态。在这之后的用户数据传输中，输出数据存储在从站，但是它的输出状态保持不变。需要用 UNSYNC 命令来解除同步模式。

冻结（FREEZE）命令使指定的从站组进入冻结模式，即将各从站的输入数据冻结在当前状态，直到主站发送下一个冻结命令时才刷新。需要用 UNFREEZE 命令来解除冻结模式。

（7）DPM1 和组态设备之间的循环数据传输

PROFIBUS-DP 允许 DPM1（1 类 DP 主站）和 DPM2（2 类 DP 主站）之间的数据交换。

该功能使组态和诊断设备可以通过总线对系统进行组态，改变 DPM1 的操作方式，动态地允许或禁止 DPM1 与某些从站之间交换数据。

2．DP-V1 的扩展功能

（1）非等时数据交换

除了 DP-V0 的功能外，DP-V1 最主要的特征是具有主站与从站之间的非等时数据交换功能，可以用它来进行参数设置、诊断和报警处理。它与循环数据交换是并行执行的，但是优先级较低。

（2）扩展的诊断功能

DP 从站通过诊断报文将突发事件（报警信息）传送给主站，主站收到后发送确认报文给从站。从站收到后才能发送新的报警信息，这样可以防止多次重复发送同一报警报文。状态报文由从站发送给主站，不需要主站确认。

3．DP-V2 的主要扩展功能

（1）从站与从站之间的通信

从站之间的直接数据交换（DX）通信采用广播式通信方式，从站作为生产者（Publisher），可以不经过主站直接将信息发送给作为消费者（Subscribers）的从站。这样从站可以直接读取别的从站的数据。

（2）等时同步模式

同步（Isochronous）功能可以实现主站与从站中的时钟同步，而与总线负载无关。此功能可以实现高精度定位处理，其时钟误差小于 1μs。通过全局控制（Global control）广播报文，使所有有关的设备循环与总线主循环同步。

（3）时钟控制与时间标记

通过用于时钟同步的新的连接 MS3，实时时间（Real Time）主站将时间标记（Time Stamp）发送给所有的从站，将从站的时钟同步到系统时间，误差小于 1ms。在有大量主站的网络中，利用这一功能可以实现高精度的事件跟踪，和实现事件顺序记录。

2.2.3 PROFIBUS 的通信服务

除了 PROFIBUS-DP、PROFIBUS-PA 和 PROFIBUS-FMS、PG/OP 和 S7 通信服务之外，PROFIBUS 还提供下列通信服务：

1．PROFIdrive

PROFIdrive 用于将驱动设备（从简单的变频器到高级的动态伺服控制器）集成到自动控制系统中。PROFIdrive 定义了用 PROFIBUS 访问驱动器数据的设备性能和方法。

为了完成现代驱动器的各种任务，PROFIdrive 定义了 6 个应用类别：

1）类别 1 定义了用速度设定值控制的标准驱动器。

2）类别 2 定义了具有技术功能的标准驱动器。过程被划分为一些子过程，主站将驱动任务发送给驱动设备，请求在各个驱动器之间直接进行数据交换。

3）类别 3 定义了包括位置控制器的定位驱动器，通过 PROFIBUS 启动和传输定位请求。

4）类别 4 和类别 5 定义了可以在多个驱动器之间实现协调运动顺序的中央运动控制。PROFIBUS 用于位置闭环控制和同步时钟周期。

5）类别 6 包括时钟处理和使用电子轴的分布式自动化，例如通过直接数字交换和同步通

信实现“电子齿轮传动”或“电子凸轮”功能。

PROFIdrive 定义了访问驱动器参数和与制造商有关的配置文件的参数的机制。对其他参数的非循环访问通过一个符合 DPV1 的参数通道进行。

2．PROFIsafe

PROFISafe（见图 2-7）用于 PROFIBUS 和 PROFINET 面向安全设备的故障安全通信。可以用 PROFIsafe 很简单地实现安全的分布式解决方案。可以在同一条物理总线上同时传输标准数据和故障安全数据，不需要对故障安全 I/O 进行额外的布线。

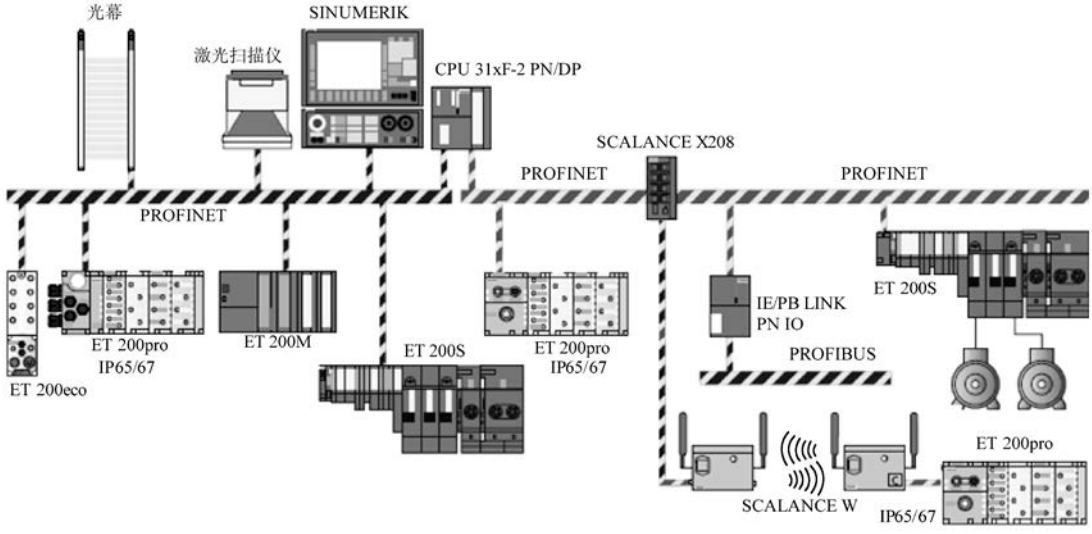


图 2-7　PROFISafe

PROFIsafe 是一种软件解决方案，在 CPU 的操作系统中以附加的 PROFIsafe 层的形式实现故障安全通信。安全数据作为标准数据的附加部分打包，从而形成 PROFIsafe 报文。

PROFIsafe 考虑了数据的延迟、丢失、不正确的时序、地址和数据的损坏。采用下列措施来保证故障安全数据传输的完整性：

1）安全报文的连续编号。

2）报文帧的时间监视和确认。

3）发送站与接收站之间使用密码来辨识。

4）采用循环冗余校验（CRC）。

使用 PROFIsafe 时，用户程序中的故障安全块将被标记为黄色，与指示物理故障安全 I/O 模块的黄色相对应。

3．PROFIBUS FDL

FDL 是 Fieldbus Data Link（现场总线数据链路）的缩写，通信伙伴可以是 S7、S5 系列 PLC 或 PC。FDL 服务由 PROFIBUS 协议的第 2 层提供，允许发送和接收最多 240B 的数据块。只有 CP（通信处理器）才能提供 FDL 服务。

S7-300/400 调用通信功能块 AG_SEND 和 AG_REC 来实现 FDL 服务。

4．PROFIBUS 在冗余控制系统中的应用

可以将 PROFIBUS 用于冗余控制系统。例如，通过两个接口模块，将 ET 200 I/O 设备连

接到冗余自动化系统的两个 PROFIBUS 子网（见图 2-8）。

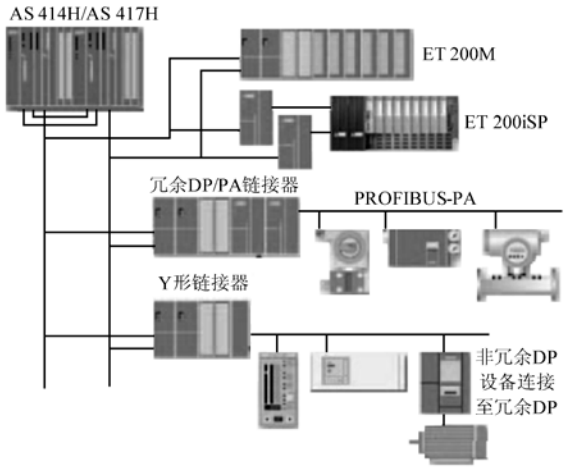


图 2-8　PROFIBUS 在冗余系统中的应用

PROFIBUS PA 线路可以通过一个冗余 DP/PA 链接器和两个接口模块进行耦合。也可以通过所谓的 Y 形链接器在冗余 PROFIBUS 中操作非冗余设备。

2.3　练习题

1．PROFIBUS 有哪 3 种通信协议？
2．PROFIBUS-DP 主要用于什么场合？
3．PROFIBUS-DP 的物理层使用什么通信接口？最大传输速率是多少？
4．PROFIBUS-DP 网络为什么需要安装终端电阻？怎样连接终端电阻？
5．1 类和 2 类 DP 主站各有什么功能？
6．PROFIBUS 通信处理器有什么功能？
7．ET 200S 有什么特点？
8．ET 200M 有什么特点？
9．PROFIBUS-DP 采用什么样的总线访问控制机制？
10．在西门子工控网站下载 EM 277 的 GSD 文件后，在 STEP 7 中安装它。

第 3 章　PROFIBUS-DP 主从通信

3.1　主站与标准 DP 从站通信的组态

某些分布很广的系统，例如大型仓库、码头和自来水厂等，可以采用基于 PROFIBUS-DP 网络的分布式 I/O，将它们放置在离传感器和执行机构较近的地方。分布式 I/O 可以减少大量的接线。

PROFIBUS-DP 最大的优点是使用简单方便，在大多数甚至绝大多数实际应用中，只需要对网络通信作简单的组态，不用编写任何通信程序，就可以实现 DP 网络的通信。用户程序对远程 I/O 的访问，就像访问中央机架的 I/O 一样。因此对远程 I/O 的编程，与对集中式系统的编程基本上没有什么区别。上述优点是 PROFIBUS-DP 得到广泛应用的主要原因之一。

使用得最多的分布式 I/O 是西门子的 ET 200，通过安装 GSD 文件，符合 PROFIBUS-DP 标准的其他厂家的设备也可以在 STEP 7 中组态。

非智能从站又称为“标准”从站，它没有 CPU 模块，通过接口模块（IM）与 DP 主站通信。本节将通过实例，介绍组态 DP 网络和组态标准 DP 从站的方法。

在实际系统中，主站与标准 DP 从站的通信用得最多。建议对硬件和网络组态不太熟悉的读者将 3.1、3.2 节作为阅读的重点，一边阅读一边用 STEP 7 对书中的例程组态，以掌握主从方式的 DP 通信的组态和编程的方法。

3.1.1　项目的生成与硬件组态

本节将通过一个例子，对项目的生成和硬件、网络组态的过程作详细介绍。

1．使用的软件

随书光盘的例程项目是用 STEP 7 V5.4 中文版 SP3.1 创建的，作者主编的《S7-300/400 PLC 应用技术（第 2 版）》的随书光盘中有该软件和仿真软件 PLCSIM 的演示版，该书介绍了 STEP 7 的安装和使用方法。

第一次使用软件时，激活临时许可证，可以获得 14 天试用期。如果没有许可证密钥（即授权），安装 STEP 7 之后，用软件 Ghost 将计算机的 C 盘备份。临时许可证到期后，用 Ghost 和备份的文件将 C 盘复原，STEP 7 又可以获得 14 天的有效使用期。

2．用新建项目向导创建项目

双击桌面上的 STEP 7 图标，打开 SIMATIC Manager（管理器）窗口，弹出“STEP 7 向导：‘新建项目’”对话框。

点击“取消”按钮，将打开上一次退出 STEP 7 时打开的所有项目。

点击“预览”按钮，可以打开或关闭该按钮下面的项目预览窗口。

点击“下一个>”按钮，在下一个对话框中选择 CPU 模块的型号为 CPU 414-2DP（见图 3-1），可以修改 CPU 在 MPI 网络中的站地址（默认值为 2）。对于实际的控制系统，CPU 的型号与订货号应与实际的硬件相同，CPU 列表框的下面是所选 CPU 的基本特性。

图 3-1　新建项目向导

点击“下一个>”按钮，在“项目名称”文本框，将项目的名称改为 PB_MS_1。本章的例程在随书光盘的文件夹“\Project\PB_MS”中。

点击“完成”按钮，开始创建项目。创建完成后，自动返回 SIMATIC 管理器（见图 3-2）。

图 3-2　SIMATIC 管理器

3．硬件组态工具 HW Config

硬件组态的任务就是在 STEP 7 中生成一个与实际的硬件系统完全相同的系统，例如生成 DP 主站、DP 网络和网络中的 DP 从站，在机架中插入模块，以及设置各组件的参数，即给参数赋值。

选中 SIMATIC 管理器左边窗口出现的“SIMATIC 400 站点”图标，用鼠标左键双击（简称双击）右边窗口中的“硬件”图标（见图 3-2），打开硬件组态工具 HW Config（见图 3-3）。可以看到自动生成的机架和 4 号槽中的 CPU 模块，需要在机架中插入其他模块。

可以用菜单命令“查看”→“目录”，或工具栏上的按钮打开或关闭右边的硬件目录窗口。选中硬件目录中的某个对象，在硬件目录下面的小窗口可以看到它的简要信息，包括订货号和对象的主要功能。

硬件目录中的 CP 是通信处理器，FM 是功能模块，IM 是接口模块，PS 是电源模块，RACK 是 S7-400 的机架或 S7-300 的导轨（Rail）。SM 是信号模块，其中的 DI、DO 分别是数字量输入模块和数字量输出模块，AI、AO 分别是模拟量输入模块和模拟量输出模块。

图 3-3 左上方的窗口是硬件组态窗口，可以在该窗口放置主机架和扩展机架，并用接口模块将它们连接起来。也可以在该窗口生成 DP 网络，并在网络上放置 DP 从站。

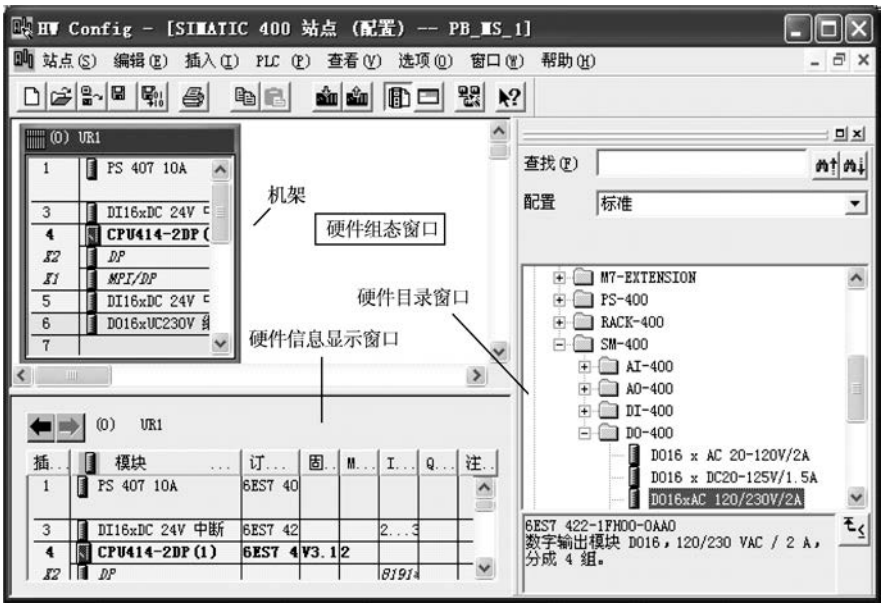


图 3-3 HW Config

选中硬件组态窗口中某个机架或 DP 从站，下面的硬件信息显示窗口将显示选中对象的详细信息，例如模块的订货号、CPU 的固件版本号和 CPU 在 MPI 网络中的站地址、I/O 模块的地址和注释等。

4．组态 S7-400 站点

用鼠标打开硬件目录中的文件夹“\SIMATIC 400\PS-400\Stantard PS-400”（S7-400 的标准电源），点击其中的“PS 407 10A”（交流输入的电源模块），该模块被选中，其背景变为深蓝色。此时硬件组态窗口的机架中允许放置该模块的 1 号插槽变为绿色，其他插槽为灰色。

可以用“拖放”的方法将选中的电源模块插入 1 号插槽。具体的方法如下：用鼠标左键按住该模块不放，移动鼠标，将选中的模块“拖”到机架中的 1 号插槽。没有移动到允许放置该模块的插槽时，光标的形状为（禁止放置）。反之光标的形状变为（允许放置）。此时放开鼠标左键，拖放的模块被放置到光标所在的插槽。S7-400 的电源模块只能放在 1 号槽，“PS 407 10A”占两个槽的位置。

除了上述“拖放”的方法之外，也可以用下面更简单的方法来插入模块，具体的方法如下：首先选中 1 号槽，即用鼠标单击机架的 1 号槽（表格中的第 1 行），使该行的背景变为深

蓝色。用鼠标双击硬件目录窗口中的“PS 407 10A”，1 号槽和 2 号槽所在的行将会出现“PS 407 10A”。可以用同样的方法将 DP 从站“挂”到 DP 网络上。

用上述的方法，将文件夹“\SIMATIC 400\SM-400\DI-400”中的 16 点数字量输入模块插入 5 号槽，将文件夹“\SIMATIC 400\SM-400\DO-400”中的 16 点数字量输出模块插入 6 号槽。它们分别占用输入、输出区中的 0 号和 1 号字节。

可以用“拖放”的方法，在机架中将模块移动到别的允许的槽位。

3.1.2 PROFIBUS-DP 网络的组态

1．生成 PROFIBUS 子网络

用鼠标双击机架中 CPU 414-2DP 下面“DP”所在的行（见图 3-3），在出现的“属性-DP”对话框的“工作模式”选项卡中，可以看到默认的工作模式为“DP 主站”。点击“常规”选项卡中的“属性”按钮，在出现的“属性 – PROFIBUS 接口 DP”对话框中，可以设置 CPU 在 DP 网络中的站地址，默认的站地址为 2（见图 3-4）。

点击“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中（见图 3-5），可以用列表框设置 PROFIBUS 子网的参数。系统推荐的默认参数如下：传输速率为 1.5Mbit/s，配置文件为 DP，一般采用默认的参数。传输速率和总线配置文件将用于整个 PROFIBUS 子网络。

“最高的 PROFIBUS 地址”用来优化多主站总线存取控制（令牌管理），可以使用默认值，或者设置为比实际的主站总数高一些的数值。选中复选框“改变”后可以修改该参数。

点击图 3-4 中的“删除”按钮，可以删除选中的子网列表框中的子网络。点击图 3-4 中的“属性”按钮，将打开选中的 PROFIBUS 网络的“属性 – PROFIBUS”对话框。

图 3-4　PROFIBUS 接口属性对话框

点击图 3-5 中的“确定”按钮，返回“属性–PROFIBUS 接口”对话框（见图 3-4）。可以看到“子网”列表框中出现了新生成的名为“PROFIBUS（1）”的子网。两次点击“确定”

按钮，返回 HW Config 窗口，此时只能看到 S7-400 的机架和新生成的 PROFIBUS（1）网络线。图 3-6 是已经组态好的 PROFIBUS 网络。

图 3-5　组态 PROFIBUS 网络参数

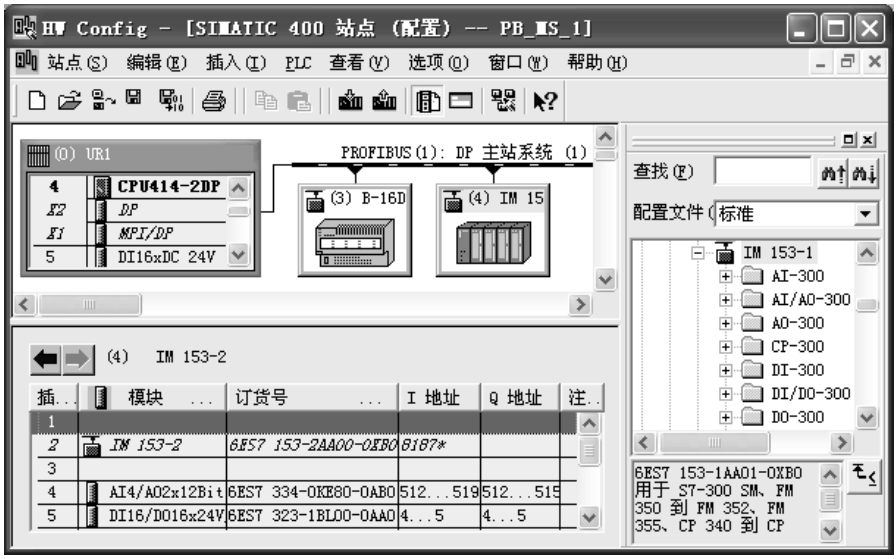


图 3-6　组态好的 PROFIBUS 网络

点击图 3-5 中的“选项”按钮，打开“选项”对话框（见图 3-7）。其中的参数用于优化 PROFIBUS 的总线参数，站点很少的简单 PROFIBUS 网络可以采用默认的参数，不用打开该对话框。

如果选中图 3-5 中的“DP”配置文件，则“选项”对话框中有“恒定的总线周期”和“电缆”选项卡。如果选中图 3-5 中的“标准”配置文件，则“选项”对话框中有“恒定的总线周期”、“网络站点”和“电缆”选项卡。

在“恒定的总线周期”选项卡，如果选中“激活恒定的总线周期”复选框，可以确保

PROFIBUS-DP 的总线周期恒定，即 DP 主站与从站以完全相同的时间间隔交换它们的数据。

如果网络中有光链接模块（OLM）、光总线终端（OBT）和 RS-485 中继器，则应在“选项”对话框的“电缆”选项卡中激活“考虑下列电缆组态”选项（见图 3-7 的右图），这样即可设置铜质电缆的长度和 RS-485 中继器的个数，或设置光纤电缆的长度，以及所用的 OLM、OBT 的个数。在计算 STEP 7 总线参数时将会用到这些信息。

图 3-7　PROFIBUS 网络选项对话框

点击图 3-5 中的“总线参数”按钮，可以查看总线参数选项卡（见图 3-8）。只有选中图 3-5 的“自定义”配置文件，才能修改总线参数。

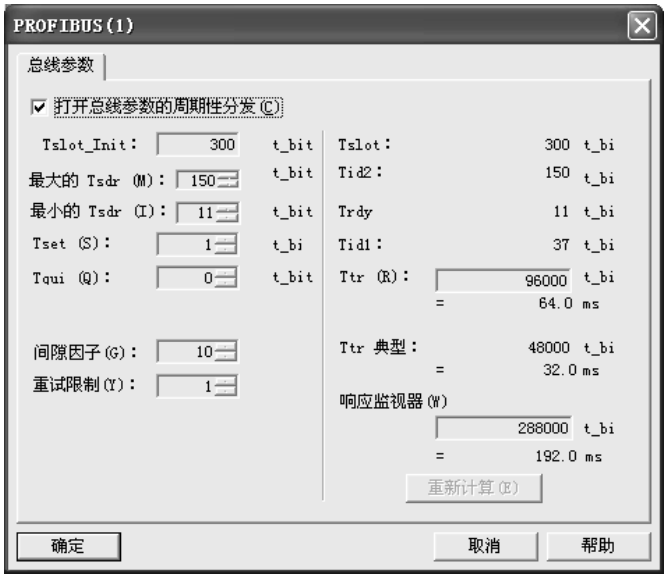


图 3-8　总线参数选项卡

2．配置文件

配置文件为不同的 PROFIBUS 应用提供基准（即默认的设置），每个配置文件包含一个 PROFIBUS 总线参数集。这些参数由 STEP 7 计算和设置，并考虑到特殊的配置文件和传输速率。这些总线参数适用于整个总线和连接在该 PROFIBUS 子网络上的所有节点。

应为 PROFIBUS-DP 网络上不同的硬件配置提供不同的配置文件：

（1）“DP”配置文件

纯 PROFIBUS-DP 单主站系统，或包含 SIMATIC S7 和 M7 装置的多主站系统选用 DP 配置文件。符合 PROFIBUS 标准的设备（SIMATIC S7、M7、C7 和 S5 和其他厂家生产的分布式 I/O）可以连接到 PROFIBUS 子网上。这些节点必须是 STEP 7 项目的组成部分，并且已经被组态。

（2）“标准”配置文件

网络中如果有其他 STEP 7 项目的节点或未在 STEP 7 中组态的节点，可以选用“标准”配置文件，STEP 7 将根据一种简单的非优化的算法来计算总线参数。

（3）“通用（DP/FMS）”配置文件

如果个别的 PROFIBUS 子网节点使用 PROFIBUS-FMS 服务，对于 CP 343-5（S7 系列）、CP 5431（S5 系列）和其他厂家生产的 PROFIBUS-FMS 设备，可以选择 DP/FMS 配置文件。使用 DP/FMS 配置文件的网络可以使用 SIMATIC S5 系列的部件，也可以使用其他 STEP 7 项目中的附加节点。PROFIBUS-FMS 已基本上被工业以太网代替，现在很少使用。

（4）“自定义”配置文件

可以为特殊的应用定义专用的自定义配置文件。首先选用 DP、标准或 DP/FMS 配置文件的总线参数设置，作为用户定义的配置文件，然后根据需要修改它们。这种调整和修改只能由具有网络使用经验的工程师来完成。

3．设置主站的启动属性

生成新的 DP 网络，返回 HW Config 后，只有 S7-400 的机架和新生成的 DP 网络。

在分布式控制系统中，不可能同时接通所有的设备或系统部件的电源。在 DP 主站启动时，不是所有的 DP 从站都可供使用。因此 DP 主站在装载从站参数和开始与 DP 从站循环地交换用户数据之前，需要一定的启动时间。

如果有必要，可以修改上述的启动时间，方法如下：双击机架中 CPU 所在的行，打开 CPU 属性对话框，用“启动”选项卡中的参数“来自模块的‘完成’消息[100ms]”，设置上电后所有 DP 从站发送“准备就绪”消息的时间。

3.1.3 主站与 ET 200 通信的组态

1．PROFIBUS-DP 从站

支持 DPV1 的 DP 从站常被称为“标准从站”，或称为非智能从站。ET 200 是典型的标准从站。在 PROFIBUS-DP 网络中，某些型号的 CPU 可以作 DP 从站。它们称为“智能（Intelligent）从站”，简称为“I 从站”。

2．DP 网络中的 I/O 地址分配

在 PROFIBUS 网络系统中，主站和非智能从站的 I/O 自动统一编址。下面是模块地址分配的原则：

1）I/O 分为 4 类，即数字量输入、数字量输出、模拟量输入和模拟量输出。按组态的先后次序，同类 I/O 模块的字节地址依次顺序排列。

2）数字量 I/O 模块的起始地址从 0 号字节开始分配。S7-300 和 S7-400 的模拟量 I/O 模块的起始地址分别从 256 号和 512 号字节开始分配，每个模拟量 I/O 点占 2B（两个字节）的地址。模块地址与模块所在的机架号和插槽号无关。

3）HW Config 自动统一分配 DP 主站和它的标准从站（非智能从站）的 I/O 的起始字节地址，用户也可以在模块的属性对话框的“地址”选项卡中修改它。不过一般都使用自动分配的地址。

智能 DP 从站内部的 I/O 地址独立于主站和其他从站。主站和智能从站之间通过组态时设置的输入/输出区来交换数据。

3．组态 DP 从站 ET 200 B

下面以 ET 200B 和 ET 200M 为例，介绍 DP 主站与标准 DP 从站的组态方法。首先组态 ET 200B 从站。打开图 3-6 右边硬件目录窗口的文件夹“\PROFIBUS DP\ET 200B”。用鼠标将其中的 B-16DI/16DO DP（见图 3-9 的左图）拖放到 HW Config 的 PROFIBUS 网络线上，这样就生成了 DP 从站，并将它连接到了 DP 主站系统。在自动打开的“属性 － PROFIBUS 接口”对话框中，设置该 DP 从站的站地址为 3，点击“确定”按钮，返回 HW Config。

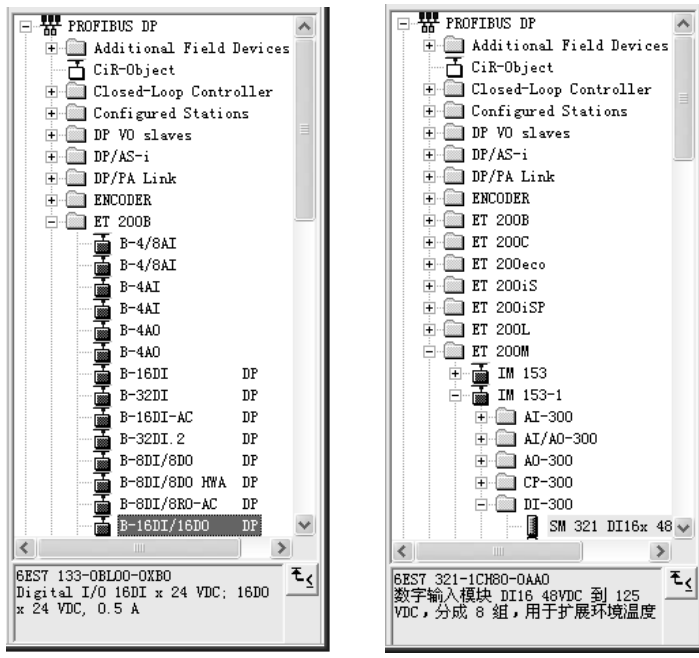


图 3-9　HW Config 右边的硬件目录窗口

选中该从站，在下面的窗口中，可以看到自动分配给它的输入、输出地址均为 2 号和 3 号字节。用 ET 200B 自带的拨码开关设置的从站地址应与 HW Config 设置的站地址相同。

双击某个 DP 从站的图标，打开“属性 － DP 从站”对话框，在“常规”选项卡（见图 3-10）中，可以看到已组态的 DP 从站的一些参考信息，例如订货号、设备系列、从站类型、诊断地址和站地址等。运行时 DP 从站通过“诊断地址”向主站报告从站的故障或返回信息。然后，CPU 将启动 OB86（机架/DP 从站故障）。SFC 13（DPNRM_DG）用该地址来从

DP 从站获取完整的诊断信息。诊断地址由 STEP 7 自动生成，用户也可以更改它。

图 3-10　DP 从站属性对话框

“SYNC/FREEZE 能力”（同步/冻结功能）复选框指出 DP 从站是否能执行由 DP 主站发出的同步和冻结控制命令。HW Config 从 DP 从站的 GSD 文件中得到有关的信息，用户不能更改此设置。如果选中“响应监视器”（监控定时器）功能，在预定义的响应监视时间内，如果 DP 从站与主站之间没有数据通信，DP 从站将切换到安全状态，所有输出被设置为 0 状态，或输出一个替代值。如果关闭监控定时器（没有选中“响应监视器”），出错时 DP 从站的输出可能不会被置为 0 状态，所以建议只是在调试时才关闭监控定时器。

在图 3-10 的“参数赋值”选项卡中，可以设置 DP 从站的参数。有关数据的内容和含义，请查阅 DP 从站设备的使用手册。

选中图 3-6 上面的组态窗口中的某个从站后，屏幕左下部的表格将显示它的详细资料，例如分配给它的 I/O 字节地址。双击表中某一行的输入或输出，在打开的从站或模块属性对话框中，可以更改输入/输出地址。

4．组态模块式 DP 从站 ET 200 M

ET 200M 是模块式远程 I/O，打开硬件目录的文件夹“\PROFIBUS-DP\ET 200M”，将其中的接口模块 IM 153-1（见图 3-9 的右图）拖放到 PROFIBUS 网络线上，就生成了 ET 200M 从站。在出现的“属性 – PROFIBUS 接口 IM 153-1”对话框中，设置它的站地址为 4。用 IM 153-1 模块上的 DIP 开关（见图 3-11）设置的站地址应与 STEP 7 组态的站地址相同。

选中图 3-6 上面窗口中的该从站，下面窗口是它的机架中的槽位，其中的 4～11 号槽最多可以插入 8 块 S7-300 系列的模块。打开硬件目录中的“IM 153-1”子文件夹，它里

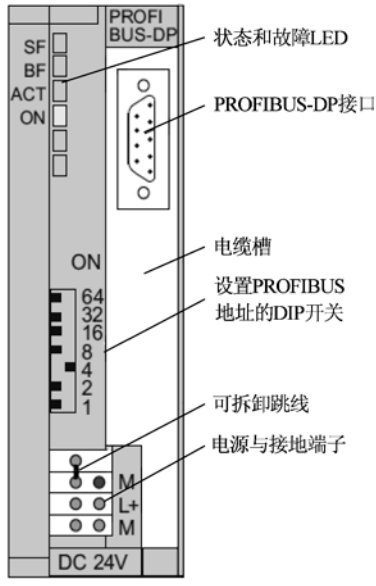


图 3-11　IM 153 的正面视图

面的各子文件夹列出了可用的 S7-300 模块，其组态方法与普通的 S7-300 的相同。将模拟量模块“SM 334 AI4/AO2”拖放到下面窗口的 4 号槽（见图 3-6），AI、AO 的起始字节地址均为 512。将数字量模块“SM 323 DI16/DO16”拖放到 5 号槽，DI、DO 的字节地址均为 4 和 5。

组态任务完成后，点击工具栏上的按钮，保存当前的组态。点击按钮（编译并保存），首先对组态信息进行编译。如果组态存在问题，将会显示错误或警告信息。改正错误后，才能成功地编译，警告信息并不影响运行。编译成功后，组态信息保存在系统数据中。

可以在 HW Config 中用按钮下载组态信息，也可以在 SIMATIC 管理器中下载“块”文件夹中的系统数据。系统数据包含硬件和网络组态的信息。

5．通信的验证

为了验证 CPU 与 DP 从站之间的通信，可以在主程序 OB1 中编写下面的简单程序：

```
L     IW    4
T     QW    4
```

即用 ET 200M 的数字量输入来控制它的数字量输出。

选中 SIMATIC 管理器左边窗口中的“块”文件夹，点击工具栏上的按钮，下载“块”文件夹中包含组态信息的“系统数据”和 OB1。用 PROFIBUS 电缆连接 CPU 模块和 ET 200 从站的 DP 接口，接通主站和从站的电源。将 CPU 切换到 RUN 模式，通过通信，ET 200M 的数字量输入模块的 IW4 的值被传送到 CPU，上述程序将 IW4 的值传送给 CPU 存储区的过程映像输出 QW4。通过通信，QW4 的值被传送给 ET 200M 的输出模块。用接在输入端的小开关改变 IW4 的值，可以看到 QW4 的状态随之改变。

在 CPU 处于 RUN 模式时，也可以用变量表来监控 DP 从站的输入值和输出值，以此来验证主站与从站之间的通信。

3.1.4 主站通过 EM 277 与 S7-200 通信的组态

PROFIBUS-DP 是通用的国际标准，符合该标准的第三方设备可以作 DP 网络的主站或从站。第三方设备作主站时，用于组态的软件由第三方提供。第三方设备作从站时，可以用 STEP 7 组态。需要在 HW Config 中安装 GSD 文件，才能在硬件目录窗口看到第三方设备和对它进行组态。下面以 S7-200 的 PROFIBUS 从站模块 EM 277 为例，介绍支持 PROFIBUS-DP 协议的第三方设备的组态方法。

1．PROFIBUS-DP 从站模块 EM 277

PROFIBUS-DP 从站模块 EM 277 用于将 S7-200 CPU 连接到 PROFIBUS-DP 网络，波特率为 9.6kbit/s～12Mbit/s。建议在与 S7-300/400 或其他系统通信时，尽量使用这种通信方式。EM 277 是智能模块，能自适应通信速率，其 RS-485 接口是隔离型的。作为 DP 从站，EM 277 接收来自主站的 I/O 组态，向主站发送数据，接收来自主站的数据。主站可以读写 S7-200 的 V 存储区，每次可以与 EM 277 交换 1～128B 的信息。EM 277 只能作 DP 从站，不需要在 S7-200 一侧对 PROFIBUS-DP 通信组态和编程。

EM 277 除了作 DP 从站外，还能作为 MPI 从站，与同一网络中的编程计算机或 S7-300/400 CPU 等其他主站进行通信。S7-200 的编程软件可以通过 EM 277 对 S7-200 编程。EM 277 共有 6 个连接，其中的两个分别保留给编程器（PG）和操作员面板（OP）。EM 277 实际上是通

信端口的扩展，可以用于连接人机界面（HMI）等设备。

2．组态 S7-300 站

在下面的例子中，S7-300 与 S7-200 通过 EM 277 进行 PROFIBUS-DP 通信，需要用 STEP 7 对 S7-300 和 EM 277 组态。在 S7-200 的程序中，只需将待发送的数据传送到组态时指定的 V 存储区，或者在组态时指定的 V 存储区中读取接收的数据就可以了。

在 STEP 7 的 SIMATIC 管理器中，生成一个项目（见随书光盘中的例程 DP_EM277），CPU 模块的型号为 CPU 315-2DP。选中该站后，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 3-13），双击“DP”所在的行，点击打开的对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成一条 PROFIBUS-DP 网络，采用默认的网络参数和默认的站地址 2。点击“确定”按钮返回 HW Config。

3．安装 EM 277 的 GSD 文件

EM 277 作为 PROFIBUS-DP 从站模块，其有关参数是以 GSD 文件的形式保存的。在对 EM 277 组态之前，需要安装它的 GSD 文件。EM 277 的 GSD 文件“siem089d.gsd”在随书光盘的文件夹“\Project\DP_MS”中。

执行 HW Config 中的菜单命令“选项”→“安装 GSD 文件”，在出现的“安装 GSD 文件”对话框中（见图 3-12），用最上面的选择框选中 GSD 文件“来自目录”。点击“浏览”按钮，用出现的“浏览文件夹”对话框选中随书光盘中的文件夹“\Project\DP_MS”，点击“确定”按钮，该文件夹中的 GSD 文件“siem089d.gsd”等出现在列表框中。选中需要安装的 GSD 文件，点击“安装”按钮，开始安装。

图 3-12 “安装 GSD 文件”对话框

4．不能安装 GSD 的处理方法

随书光盘中的项目 DP_EM277 带有 EM 277 的 GSD 文件，读者打开该项目后，在 HW Config 的硬件窗口中看不到 EM 277。如果安装来自目录的 GSD 文件（见图 3-12），将会出现无法安装的警告信息。这是因为打开该项目时，EM 277 的 GSD 文件被引用。必须重新启动计算机，关闭被引用的 EM 277 的 GSD 文件后，才能在别的项目中安装 EM 277 的 GSD 文件。

安装结束后，在 HW Config 右边的硬件目录窗口的“\PROFIBUS DP\Additional Field Devices\PLC\SIMATIC”文件夹中，可以看到新安装的 EM 277（见图 3-13）。

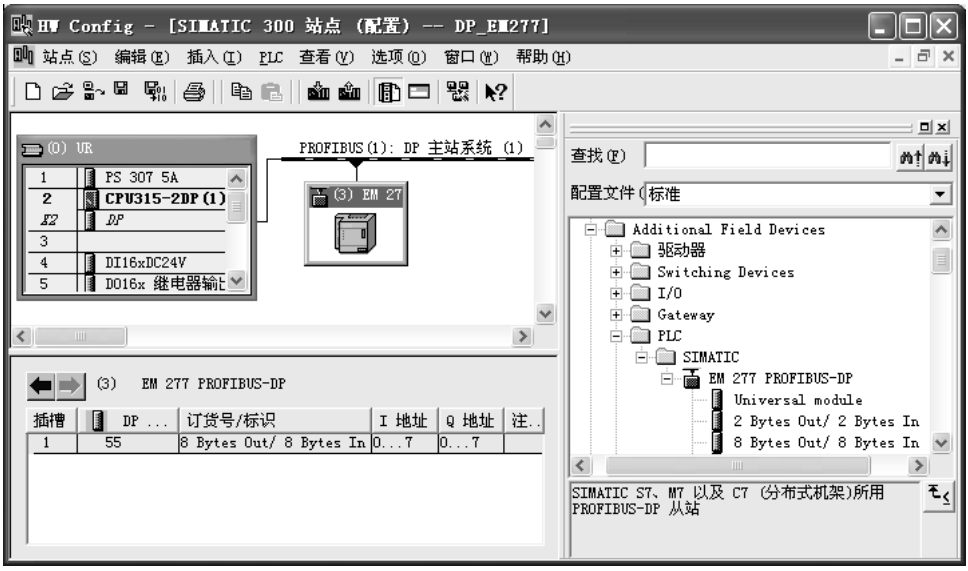


图 3-13　组态 PROFIBUS 从站

5．安装来自项目的 GSD 文件

STEP 7 将项目中所有 DP 设备的 GSD 文件保存在该项目中。以项目 DP_EM277 为例，可以将该项目中的 GSD 文件“siem089d.gsd”导入 STEP 7 的通用 GSD 文件夹，以供其他项目使用。打开另一个项目，用图 3-12 最上面的选择框选中“来自 STEP 7 项目”。点击“浏览”按钮，在出现的“打开项目”对话框中，选中项目 DP_EM277。点击“确定”按钮，返回“安装 GSD 文件”对话框，该项目中的 GSD 文件“siem089d.gsd”出现在列表框中。选中它以后，点击“安装”按钮，开始安装。

6．组态 EM 277 从站

导入 GSD 文件后，将 HW Config 右侧窗口的设备列表中的“EM 277 PROFIBUS-DP”拖放到左边窗口的 PROFIBUS-DP 网络上。用鼠标选中生成的 EM 277 从站，打开右边窗口的设备列表中的“\EM 277 PROFIBUS-DP”子文件夹，根据实际系统的需要选择传送的通信字节数。本例选择的是 8 字节输入/8 字节输出方式，将图 3-13 中的“8 Bytes Out/ 8 Bytes In”拖放到下面窗口的表格中的 1 号槽。STEP 7 自动分配远程 I/O 的输入/输出地址，因为最先组态的是 EM 277，本例分配给 EM 277 模块的输入、输出字节地址分别为 IB0～IB7 和 QB0～QB7。

双击网络上的 EM 277 从站，打开 DP 从站属性对话框。点击“常规”选项卡中的“PROFIBUS…”按钮，在打开的接口属性对话框中，设置 EM 277 的站地址为 3。用 EM 277 上的拨码开关设置的站地址应与 STEP 7 中设置的站地址相同。

在“参数赋值”选项卡中（见图 3-14），设置“I/O Offset in the V-memory”（V 存储区中的 I/O 偏移量）为 100，即用 S7-200 的 VB100～VB115 与 S7-300 的 QB0～QB7 和 IB0～IB7 交换数据。组态结束后，应将组态信息下载到 S7-300 的 CPU 模块。

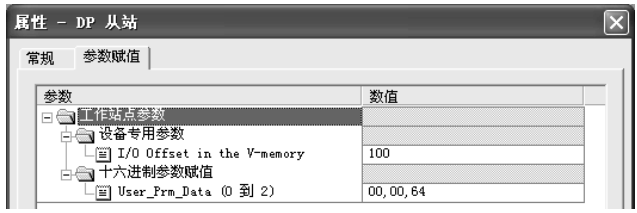


图 3-14　DP 从站属性对话框

7．S7-200 的编程

本例的 S7-200 通过 VB100～VB115 与 DP 主站交换数据。VB100～VB107 是 S7-300 写到 S7-200 的数据，对应于 S7-300 的 QB0～QB7；VB108～VB115 是 S7-300 从 S7-200 读取的数据，对应于 S7-300 的 IB0～IB7。

如果要把 S7-200 的 MB3 的值传送给 S7-300 的 MB10，应在 S7-200 的程序中，用 MOVB 指令将 MB3 传送到 VB108～VB115 中的某个字节，例如 VB108。通过通信，VB108 的值传送给 S7-300 的 IB0，在 S7-300 的程序中将 IB0 的值传送给 MB10。

在运行时可以用 STEP 7 的变量表和 STEP 7-Micro/Win 的状态表来监控通信过程中的数据传送。

3.2　DP 主站与智能从站通信的组态与编程

3.2.1　DP 主站与智能从站主从通信的组态

可以将自动化任务划分为用多台 PLC 控制的若干个子任务，这些子任务分别用几台 CPU 独立地和有效地进行处理，这些 CPU 在 DP 网络中作 DP 主站和智能从站。

DP 主站不是直接访问智能从站的物理 I/O 区，而是通过从站组态时指定的通信双方的 I/O 区来交换数据。该 I/O 区不能占用分配给 I/O 模块的物理 I/O 地址区。

主站与从站之间的数据交换是由 PLC 的操作系统周期性自动完成的，不需要用户编程，但是用户必须对主站和智能从站之间的通信连接和用于数据交换的地址区组态。这种通信方式称为主/从（Master/Slave）通信方式，简称为 MS 方式。

1．组态 DP 主站和 PROFIBUS 网络

在 STEP 7 中执行菜单命令“文件”→“新建”，创建一个项目。在“新建项目”对话框中，设置项目的名称为 PB_MS_2（见随书光盘中的同名例程）。可以直接输入保存项目的路径，或者点击“浏览”按钮，在打开的“选择目录”对话框中，选择保存项目的路径。

点击“确定”按钮后，返回 SIMATIC 管理器，用鼠标右键点击左边窗口最上面一层的项目图标（见图 3-15），执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 400 站点”，生成一个 SIMATIC 400 站点。

选中生成的“SIMATIC 400(1)”站对象，双击右边窗口中的“硬件”图标，打开 HW Config（见图 3-16）。将右边的硬件目录窗口的文件夹“\SIMATIC 400\RACK 400”中的通用机架 UR1 拖放到左边的硬件组态窗口。将电源模块插入 1 号槽，CPU 413-2DP 插入 4 号槽，从 8

号槽开始插入信号模块。

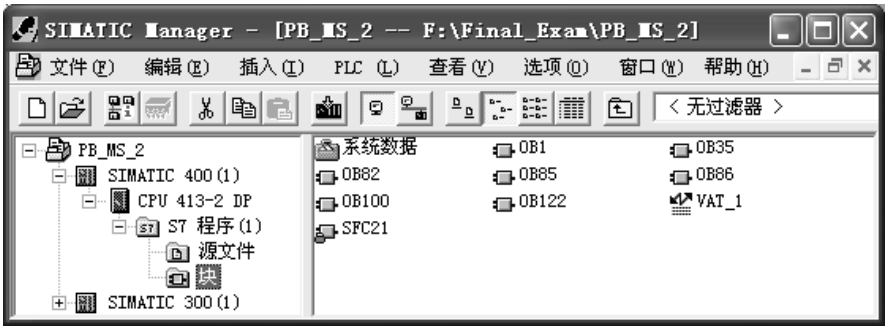


图 3-15　SIMATIC 管理器

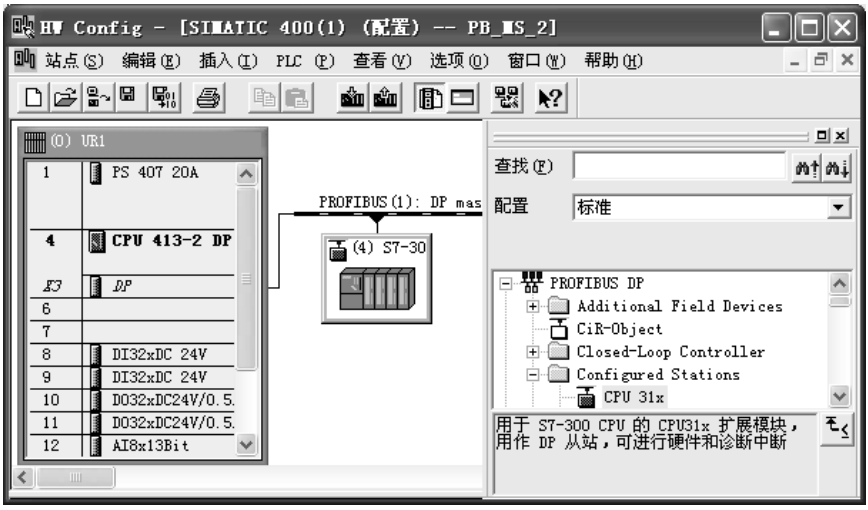


图 3-16　组态 PROFIBUS 智能从站

将 CPU 413-2DP 插入机架时，自动打开“属性–PROFIBUS 接口”对话框，采用默认的站地址 2。点击“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，可以用列表框设置 PROFIBUS 子网络的参数。采用系统推荐的默认参数，传输速率为 1.5 Mbit/s，配置文件为“DP”。两次点击“确定”按钮，返回 HW Config，可以看到生成的 PROFIBUS 网络线，此时还没有图 3-16 中的 3 号从站。

用鼠标双击机架中 CPU 413-2DP 下面“DP”所在的行（见图 3-16），在出现的 DP 属性对话框的“工作模式”选项卡中，可以看到默认的工作模式为“DP 主站”。

点击工具栏上的按钮，编译与保存组态信息。最后关闭 HW Config，返回 SIMATIC 管理器。

2．组态智能从站

用鼠标右键点击 SIMATIC 管理器左边窗口最上面的项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”。选中左边窗口中新出现的“SIMATIC 300（1）”图标，用鼠标左键双击右边窗口中的“硬件”图标，打开 HW Config。将硬件目录窗口的文

件夹\SIMATIC 300\RACK-300 中的导轨（Rail）拖放到硬件组态窗口。

S7-300 各种类型的模块的位置是固定的。电源模块占用 1 号槽，CPU 模块占用中央机架的 2 号槽，3 号槽只能用于接口模块，4～11 号槽用于其他模块。如果信号模块、功能模块和通信处理器不止 8 块，需要增加扩展机架。

将目录窗口的文件夹“\SIMATIC 300\PS 300”中的“PS 307 5A”插入 1 号槽。选中文件夹“\SIMATIC 300\CPU-300\CPU 313C-2DP”中的订货号为 6ES7 313-6CE00-0AB0 的 CPU，将它插入 2 号槽。将 CPU 放到机架上时，将会自动打开 DP 接口属性对话框的“参数”选项卡。设置 PROFIBUS 站地址为 3，不连接到 PROFIBUS（1）网络。点击“确定”按钮返回 HW Config。

因为只有主机架，没有扩展机架，不需要接口模块，3 号槽空着。但是在实际的硬件系统中，CPU 模块与 4 号槽的模块是紧挨着的。将 CPU 的 MPI 地址设置为 3。

打开硬件目录窗口中的文件夹\SIMATIC 300\SM-300，在 4～7 号槽插入 I/O 模块。

双击机架中 CPU 313C-2DP 下面的“DP”所在的行，打开 DP 属性对话框。在“工作模式”选项卡中将该站设置为 DP 从站（见图 3-17），点击“确定”按钮返回 HW Config。

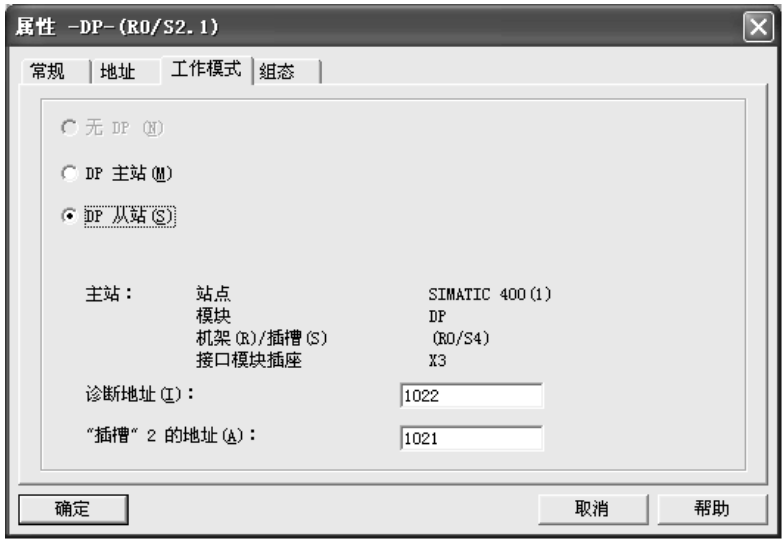


图 3-17　组态智能从站的工作模式

不是所有的 CPU 都能作 DP 从站，具体的情况可以查阅有关的手册或产品样本。在 HW Config 的硬件目录窗口下面的小窗口中，可以看到对选中的硬件的简要介绍。

因为此时从站与主站的通信组态还没有结束，不能成功地编译 S7-300 的硬件组态信息。点击按钮，保存组态信息。最后关闭 HW Config。

3．将智能 DP 从站连接到 DP 主站系统

选中 SIMATIC 管理器中的 S7-400 站，双击右边窗口的“硬件”图标，打开 HW Config。打开右边的硬件目录窗口中的“\PROFIBUS DP\Configured Stations”（已组态的站）文件夹（见图 3-16），将其中的“CPU 31x”拖放到屏幕左上方的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡（见图 3-18）被自动打开，选中从站列表中的“CPU 313-2DP”，点击“连接”按钮，该从站被连接到 DP 网络上。

图 3-18　连接 DP 从站

连接好后，“断开连接”按钮上的字符由灰色变为黑色。点击该按钮，可以将从站从网络上断开。最后点击“确定”按钮，关闭 DP 从站属性对话框，返回 HW Config。

4．主站与智能从站主从通信的组态

用鼠标双击已连接到 PROFIBUS 网络上的 DP 从站，打开 DP 从站属性对话框中的“组态”选项卡（见图 3-19），为主-从通信设置双方用于通信的输入/输出地址区。这些地址区实际上是用于通信的数据接收缓冲区和数据发送缓冲区。

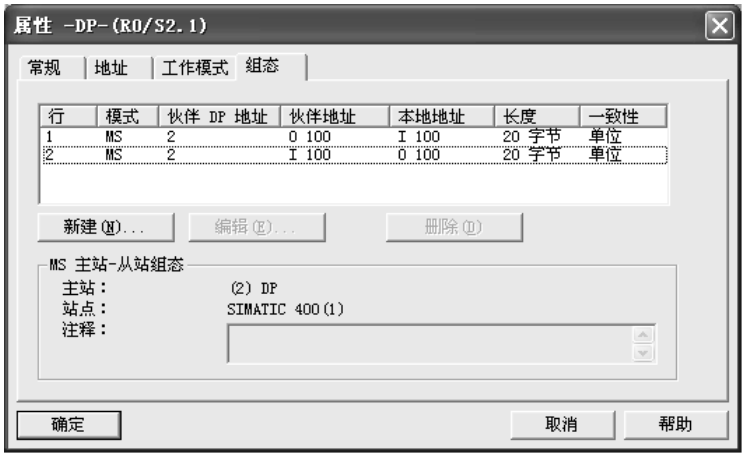


图 3-19　组态 DP 主从通信的输入/输出地址区

点击图中的“编辑”按钮，可以编辑选中的行。点击“删除”按钮，将删除选中的行。

图 3-19 中的“模式”可选“MS”（主从）或 DX（直接数据交换）；伙伴（主站）地址和本地（从站）地址是输入/输出地址区的起始地址，“长度”的单位可以选字节和字。数据的“一致性”的定义和实现的方法将在下一节介绍。

点击“新建”按钮，在出现的对话框中（见图 3-20）设置组态表第 1 行的参数。每次可

以设置智能从站与主站一个方向的通信使用的 I/O 地址区。设置好以后点击“确定”按钮，返回 DP 从站属性对话框的“组态”选项卡。

图 3-20　组态 DP 主从通信的输入/输出地址区

“组态”选项卡的第 1 行表示从站的通信伙伴（即主站）用 QB100～QB119 发送数据给从站（本地）的 IB100～IB119。第 2 行表示主站用 IB100～IB119 接收从站的 QB100～QB119 发送给它的数据。每一行最多能组态 32B，如果超出允许的字节数，点击“确定”按钮时将会出现提示信息。

组态第 2 行的通信参数时，将“DP 从站属性 – 组态 – 行 2”对话框中的 DP 伙伴（主站）的“地址类型”改为“输入”，本地（从站）的地址类型自动变为“输出”。其余的参数与图 3-20 中的相同。

图 3-19 中组态的通信双方使用的输入/输出区的起始字节地址均为 100（IB100 和 QB100），并不要求一定要将它们设置得相同。但是用于通信的数据区不能与主站和非智能从站的输入/输出区重叠。

设置完全部参数后，返回 HW Config，点击工具栏上的按钮，编译与保存 SIMATIC 400 站点的组态信息。返回 SIMATIC 管理器后，选中 SIMATIC 300（1）站点，打开 HW Config。因为已完成了所有的组态任务，点击工具栏上的按钮，可以成功地编译与保存组态信息。

点击工具栏上的按钮，打开网络组态工具 NetPro（见图 3-21），可以看到两个站点都连接到 PROFIBUS 网络上。

点击站点左边方框中的 PLC 图标，可以打开 HW Config，为该站点的硬件组态。

如果同时打开 HW Config 和 NetPro，可能会因为它们的组态相互冲突，导致不能成功地

编译和保存组态信息。此时应关闭二者之一，才能顺利编译。

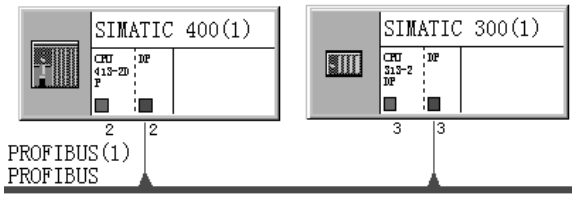


图 3-21　NetPro

3.2.2　设计验证通信的程序

1．检查通信是否成功的方法

为了检验通信是否成功，可以在初始化组织块 OB100 中，将组态时指定的通信双方的数据发送区 QB100～QB119 全部预置为某个初始值，将组态时指定的通信双方的数据接收区 IB100～IB119 清零。在运行时用变量表同时监控通信双方接收到的数据。

为了观察周期性数据传输的动态效果，可以在周期性中断组织块OB35 中，每 100ms（OB35 中断的时间间隔的默认值）将数据发送区中的某个字（例如 QW100）增加一定的值。如果通信正常，可以看到变量表中该数据在不断地变化。除此之外，可以用本地站的 ID0 来控制通信伙伴的 QD0（CPU 413-2DP）或 QD4（CPU 313C-2DP）。在运行时用接在输入端子的小开关来改变 ID0 的值，观察通信伙伴对应的输出点的状态是否随之而变。

上述程序只是用来检验通信是否成功。在编写双方实际的用户程序时，应将需要发送的数据传送到组态时设置的本站的输出区（例如本例的 QB100~QB119），将设置的本站输入区（例如本例的 IB100~IB119）接收到的数据用于需要它们的指令中。

2．初始化程序

选中 SIMATIC 管理器左边窗口中 CPU 413-2DP 的“块”文件夹，用鼠标右键点击右边的窗口，执行出现的快捷菜单中的命令，插入组织块 OB100。下面是 OB100 中的程序：

```
程序段 1：初始化发送数据区 QW100~QW118 为 16#1111
        L     W#16#1111
        T     LW    20                  //LW 是 OB100 的局部数据区中的字
        CALL  "FILL"                    //调用 SFC 21
          BVAL     :=LW20               //源数据
          RET_VAL  :=LW22               //错误代码
          BLK      :=P#Q 100.0 BYTE 20  //地址区
程序段 2：将接收数据区 IW100~IW118 清零
        L     W#16#0
        T     LW    20
        CALL  "FILL"                    //调用 SFC 21
          BVAL     :=LW20               //源数据
          RET_VAL :=LW22                //错误代码
          BLK      :=P#I 100.0 BYTE 20  //地址区
```

CPU 313C-2DP 的 OB100 的程序与 CPU 413-2DP 的基本上相同，其区别在于发送数据区被预置为 W#16#2222。

PLC 的过程映像输入区（I 区）如果用于数字量输入（DI）模块，各输入点的值唯一地取决于外接输入电路的通断，不能改写它的值，在梯形图中也不能出现输入点的线圈。过程映像输入区如果用于 DP 主站与智能从站的通信，它只是作为普通的存储区使用，可以用程序对它进行读写操作。

3．OB1 与 OB35 中的程序

下面是 CPU 413-2DP 的 OB1 中的程序：

```
程序段 1：
        L     ID     0
        T     QD     102                    //用本站的 ID0 控制对方的 QD4
程序段 2：
        L     ID     102
        T     QD     0                      //用对方的 ID0 控制本站的 QD0
```

CPU 313C-2DP 的 OB1 的程序与 CPU 413-2DP 的基本上相同，只是将上面程序中的 QD0 改为 QD4。下面是通信双方的 OB35 中的程序：

```
程序段 1：每 100ms 将 QW100 加 1
        L     QW     100
        +     1
        T     QW     100
```

为了防止通信伙伴出现故障和断电时造成 CPU 停机，为通信的双方生成 OB82、OB86 和 OB122，为 CPU 413-2DP 生成 OB85，其作用见 6.4 节。

4．修改 CPU 的 MPI 地址的方法

CPU 313-2DP 组态的 MPI 地址为 3，假设原来下载的 MPI 地址为 2。如果在 SIMATIC 管理器中下载组态信息，将会出现“在线：无法建立连接。连接伙伴未响应”的信息。必须在 HW Config 中用下面的方法下载组态信息。

点击 HW Config 工具栏上的  按钮，出现“选择目标模块”对话框。点击“确定”按钮，出现“选择节点地址”对话框（见图 3–22），“输入到目标站点的连接”列表中的 MPI 地址为组态时为 CPU 313-2DP 指定的 3。点击“显示”按钮，几秒钟后，在“可访问的节点”列表中，显示出 MPI 网络上的所有节点，同时“显示”按钮上的字符变为“更新”。可以看到 CPU 313-2DP 中原有的 MPI 地址为 2，CPU 型号为 CPU 313C-2DP。点击“可访问的节点”列表中的 CPU 313C-2DP，“输入到目标站点的连接”列表中的 MPI 地址变为 2。如果知道 CPU 原来的 MPI 地址，也可以直接将“输入到目标站点的连接”列表中的 MPI 地址修改为 2。完成这一操作后，才能将硬件组态信息和新的 MPI 地址下载到 CPU 313-2DP。点击“确定”按钮，开始下载。下载以后，CPU 中的 MPI 地址变为 3。

下一次下载组态信息或下载程序时，因为 CPU 中的 MPI 地址与组态的地址一致，出现“选择节点地址”对话框后，不需要点击“显示”按钮，直接点击“确定”按钮就可以下载。保存并编译组态信息后，也可以在 SIMATIC 管理器中下载系统数据。

图 3-22　下载硬件组态信息

用 PROFIBUS 电缆连接 CP 5613 和多个 CPU 的 MPI 接口后，不用在网络组态工具 NetPro 中将各个站连接到 MPI 网络上，也可以对每个 CPU 进行下载和监控的操作。但是各个站的 MPI 地址不能重复。

如果 MPI 网络上有两个站原来的 MPI 地址相同，对某个站下载时，为了避免冲突，应临时关闭另一个站的电源。

5．通信过程的监控

下载结束后，用电缆连接两块 CPU 集成的 DP 接口，将 CPU 切换到 RUN 模式。用 MPI 或 DP 网络监控系统的运行。

用鼠标右键点击 SIMATIC 管理器左边窗口中某个站的“块”图标，用出现的快捷菜单中的命令生成一个变量表，变量表默认的名称为 VAT_1。双击 SIMATIC 管理器右边窗口中出现的变量表图标，打开变量表，生成需要监控的变量的地址。可以只监视数据接收区的起始字和结束字的数据。在本例中，监视接收到的动态变化的 IW100、接收预置的初值的 IW106、IW118，和 ID0、QD0（或 QD4）。ID102 用于接收通信伙伴的 ID0。

同时打开通信双方的变量表，将它们调节到适当的大小。运行时选中某个站的变量表，点击工具栏上的按钮，使该变量表进入监控状态，“状态值”列显示的是 PLC 中变量的值。用同样的方法，使另一个变量表也进入监控状态。图 3-23 和图 3-24 是运行时复制的变量表。由于双方动态变化的 QW100 被传送给对方的 IW100，可以看到后者的值在不断变化。

用接在输入模块的输入端的小开关改变 ID0 的值，通信伙伴的 QD0 或 QD4 的值随之而变。在变量表中，ID0 与通信伙伴的 QD0 或 QD4 的值完全相同。

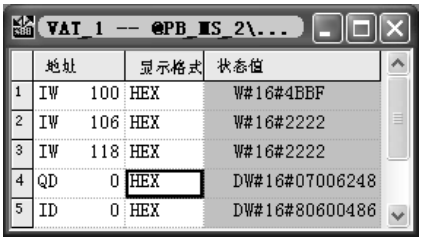


图 3-23　CPU 413-2DP 的变量表

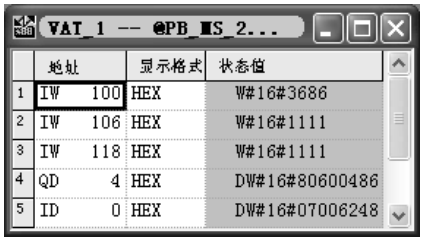


图 3-24　CPU 313C-2DP 的变量表

3.2.3　用 SFC 14 和 SFC 15 传输一致性数据

1．数据的一致性

数据的一致性（Consistency）又称为连续性。通信块被执行、通信数据被传送的过程如果被一个更高优先级的 OB 块中断，将会使传送的数据不一致（不连续）。即被传输的数据一部分来自中断之前，一部分来自中断之后，因此这些数据是不连续的。

在通信中，有的从站用来实现复杂的控制功能，例如模拟量闭环控制或电气传动等。从站与主站之间需要同步传送比字节、字和双字更大的数据区，这样的数据称为一致性数据。需要绝对一致性传送的数据量越大，系统的中断反应时间越长。可以用系统功能 SFC 14“DPRD_DAT”和 SFC 15“DPWR_DAT”来访问要求具有一致性的数据。

2．组态硬件和主从通信

在 STEP 7 中生成一个项目（见随书光盘中的例程 PB_MS_3），CPU 413-2DP 是 S7 DP 主站，CPU 313C-2DP 是智能 DP 从站。主站和从站的组态与前面的项目 PB_MS_2 基本上相同，数据长度为 20B。其区别在于参数“一致性”被组态为“全部”（见图 3-25），因此需要在用户程序中调用 SFC 15“DPWR_DAT”，将数据“打包”后发送，调用 SFC 14“DPRD_DAT”，将接收到的数据“解包”。SFC 15 用于将 RECORD 指定的数据连续地传送到 DP 从站。SFC 14、15 的参数 RECORD 指定的地址区和长度应与组态的参数一致。

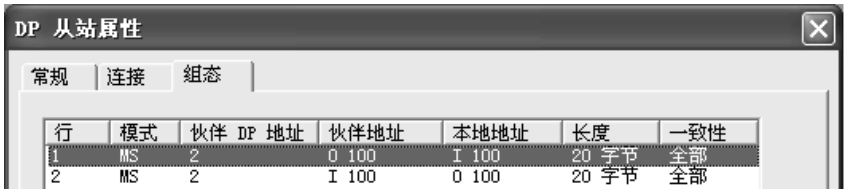


图 3-25　主从通信组态表

可以传送的数据长度与 CPU 的型号有关。对于 S7-400 CPU，最大长度是 32B。

S7-300 CPU 和 ET 200 的接口模块可以传送的最大数据长度可以查阅有关的硬件与安装手册。

如果从具有模块化设计或具有多个 DP 标识符的 DP 标准从站读取数据，通过组态时指定的起始地址，每次调用 SFC 14 只能访问一个模块或一个 DP 标识符的数据。

DP 主站用 SFC 15 发送的输出数据被智能从站用 SFC 14 读出，并作为其输入数据保存。反之也适用于智能从站发送给主站的数据的处理。用于通信的输入/输出数据区的起始地址 LADDR 应使用十六进制数格式。100 对应的十六进制数为 16#64。

3．生成数据块

选中 SIMATIC 管理器左边窗口中 CPU 413-2DP 的“块”文件夹，用鼠标右键点击右边的窗口，执行出现的快捷菜单中的命令，插入数据块 DB 1。

数据块的大小取决于数据块中定义的变量的类型和个数。数组是同一数据类型的变量的集合，用数组可以方便地定义数据块的大小。打开数据块，可以看到只有一个临时生成的 INT 型的占位符变量，用下面的方法将它修改为数组。设置数组的名称（也可以使用默认的名称），用鼠标右键点击“类型”单元，选中出现的菜单的“复杂类型”中的“ARRAY”（数组），在出现的 ARRAY 后面的方括号中输入“1..20”（见图 3-26），该数组有 20 个元素，数组元素的序号为 1～20。数组元素序号的起始值可以是 0 或别的整数。删除原有的初始值，在注释列按回车键后，ARRAY 的下面出现一个空白单元，输入数组元素的数据类型 BYTE（字节），保存数据块后关闭它。用同样的方法生成数据块 DB 2，在数据块中创建一个名为 ARAY，有 10 个字元素的数组。也可以用复制和修改名称的方法来创建内部结构相同的数据块。

图 3-26　在数据块中生成数组

4．OB1 的程序

在双方的主程序 OB1 中，调用 SFC 15“DPWR_DAT”，将 DB 1 中的数据“打包”后发送，调用 SFC 14“DPRD_DAT”，将接收到的数据“解包”后存放到 DB 2 中。

输入程序时，将程序编辑器左边窗口的文件夹“\库\Standard Library\System Function Blocks”中的 SFC 14“拖放”到右边窗口的程序段中，将会自动生成调用 SFC 14 的 CALL 指令，“:=”号之前是 SFC 的形式参数（形参），在“:=”号的后面输入各形参的实际参数（实参），“//”号的右边是对该行语句的注释。

DP 主站和智能从站的 OB1 中的用户程序基本上相同，下面是主站 OB1 的程序：

```
程序段 1：解开 IB100～IB119 接收到的数据包，并将数据存放在 DB 2
      CALL   "DPRD_DAT"                  //调用 SFC 14
        LADDR    :=W#16#64               //接收通信数据的过程映像输入区的起始地址 IB100
        RET_VAL  :=MW2                   //错误代码
        RECORD   :=DB2.ARAY              //存放接收的用户数据的目的数据区
      L      DB2.DBD      2
      T      QD           0              //用对方的 ID0 控制本站的 QD0
程序段 2：将 DB 1 的数据打包后通过 QB100～QB119 发送出去
      L      ID           0
      T      DB1.DBD      2              //用本站的 ID0 控制对方的 QD4
      CALL   "DPWR_DAT"                  //调用 SFC 15
```

```
LADDR    :=W#16#64          //发送数据的过程映像输出区的起始地址 QB100
RECORD   :=DB1.ARAY         //存放要发送的用户数据的源数据区
RET_VAL  :=MW4              //错误代码
```

因为 DB 2 中的数组的大小刚好为 20B，输入 RECORD 的实参 P#DB2.DBX0.0 BYTE 20 后，自动变为 DB2.ARAY。当然也可以直接输入 DB2.ARAY。

图 3-27 是程序段 1 的 CALL 指令对应的梯形图。梯形图和语句表中的功能（FC）和功能块（FB）包含的信息基本相同。梯形图中的 FC、FB、SFC 和 SFB 的输入参数在左边，输出参数在右边；方框里面是形参，方框外面是实参。语句表的优点是可以给每一行加上“//”右边的注释，便于程序的阅读和理解，语句表的功能比梯形图更强，有的功能只能用语句表实现。本书的程序基本上用语句表的形式给出。

图 3-27 中的梯形图可以转换为语句表，但是程序段 1 的 CALL 指令不能转换为梯形图。

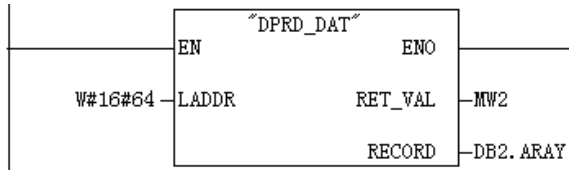


图 3-27　调用 SFC 14 的梯形图

选中梯形图或语句表中的某条指令或某个 FC、FB、SFC、SFB，按〈F1〉键可以获得选中对象的在线帮助。选中某条菜单命令，或者打开某个对话框的选项卡，按〈F1〉键也可以获得该对象的在线帮助。

从站 OB1 中的程序与主站的基本上相同，其区别仅在于将 QD0 改为 QD4。图 3-28 给出了通信双方的信号关系图。

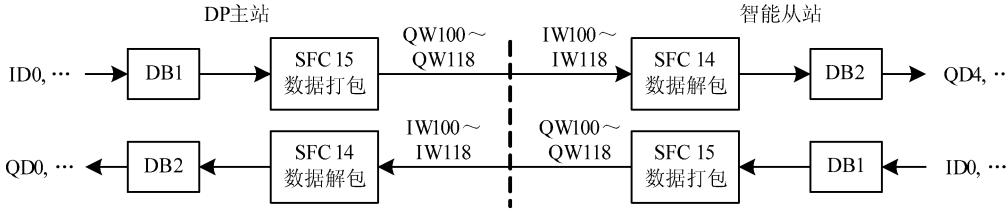


图 3-28　DP 主站与智能从站的数据传输

5．初始化程序

在主站和从站的初始化程序 OB100 中，用 SFC 21 将 DB 1 的数据发送区中的各个字分别预置为 16#1111 和 16#2222。将 DB 2 的数据接收区中的各个字清零。

6．通信的监控

在运行时用变量表同时监控通信双方接收到的 DB 2 的部分数据（见图 3-29 和图 3-30）。为了动态地观察周期性数据传输的效果，在通信双方的周期性中断组织块 OB35 中，每 100ms 将 DB1.DBW0 加 1。该数据被通信伙伴接收后存放在 DB2.DBW0，通信正常时可以看到变量

表中的 DB2.DBW0 在不断地变化。

图 3-29　CPU 413-2DP 的变量表

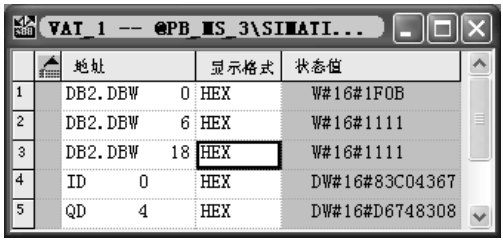


图 3-30　CPU 313C-2DP 的变量表

在运行时也可以用 DB 2 来监控数据块中定义的数组各元素的值，查看通信双方 DB 1 中的数据是否传送给了通信伙伴的 DB 2。

运行时在 SIMATIC 管理器中双击打开 DB 2，点击工具栏上的按钮，启动监控功能，出现图 3-31 所示的对话框，点击“是”按钮，DB 2 首先切换到数据视图显示方式，然后进入在线监控状态（见图 3-32）。选中对话框中的复选框“不再显示该信息”，以后点击按钮，将会直接进入数据视图监控状态。

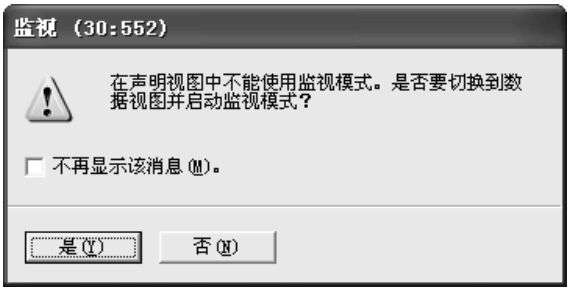


图 3-31　数据块监视的提示信息

地址	名称	类型	初始值	实际值
0.0	ary[0]	WORD	W#16#0	W#16#3E42
2.0	ary[1]	WORD	W#16#0	W#16#83C0
4.0	ary[2]	WORD	W#16#0	W#16#4367
6.0	ary[3]	WORD	W#16#0	W#16#2222
8.0	ary[4]	WORD	W#16#0	W#16#2222
10.0	ary[5]	WORD	W#16#0	W#16#2222
12.0	ary[6]	WORD	W#16#0	W#16#2222
14.0	ary[7]	WORD	W#16#0	W#16#2222
16.0	ary[8]	WORD	W#16#0	W#16#2222
18.0	ary[9]	WORD	W#16#0	W#16#2222

图 3-32　主站的 DB 2 接收的数据

3.3　PLC 与变频器 DP 通信的组态与编程

3.3.1　S7-300 与 SIMOVERT MASTERDRIVES 通信的组态

SIMOVERT MASTERDRIVES 是当前应用较广的变频器，它采用 IGBT 逆变器、全数字技术的矢量控制，是全系列通用和模块化的产品，功率范围为 0.55～2300 kW。

可以用随书光盘中的软件 Drivemonitor 或 Drive ES 来组态和监控西门子的驱动设备。

1．西门子驱动设备与 PLC 的通信方式

西门子驱动设备包括多种系列的变频器和直流调速装置。它们可以使用 PROFIBUS-DP、USS 和 SIMOLINK 这 3 种通信协议。

USS 协议属于主-从通信，PLC 作主站，驱动设备作从站。USS 协议的接口集成在基本装置中，不需要增加硬件成本。但是通信速度较慢，只有基本通信功能，最多可以连接 31 个从站。SIMOLINK 协议主要用于驱动设备之间的主从通信。

PROFIBUS-DP 协议的通信速度快，有附加功能（例如非循环通信和交叉通信），站点数

更多，但是需要添加驱动设备的 DP 通信板。

图 3-33 是 PROFIBUS-DP 通信的系统示意图，CPU 通过 MPI 接口与编程用的计算机连接，CPU 集成的 DP 接口通过 PROBIBUS 电缆与变频器的 CBP 或 CBP2 通信板上的 DP 接口连接。

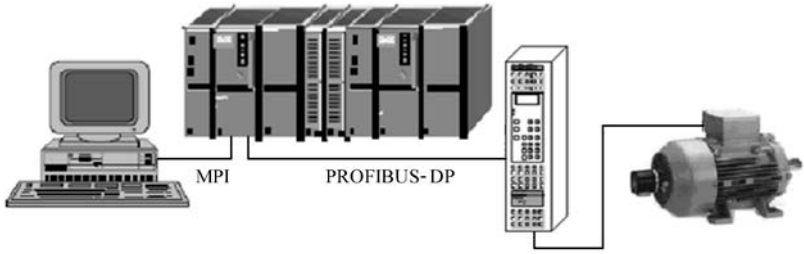


图 3-33　PLC 与变频器通信示意图

2．组态主站和 PROFIBUS 网络

在 STEP 7 中用新建项目向导创建一个项目（见随书光盘中的例程 Convert），CPU 模块为 CPU 315-2DP。选中 SIMATIC 管理器的 300 站点，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 3-34），将电源模块和信号模块插入机架。

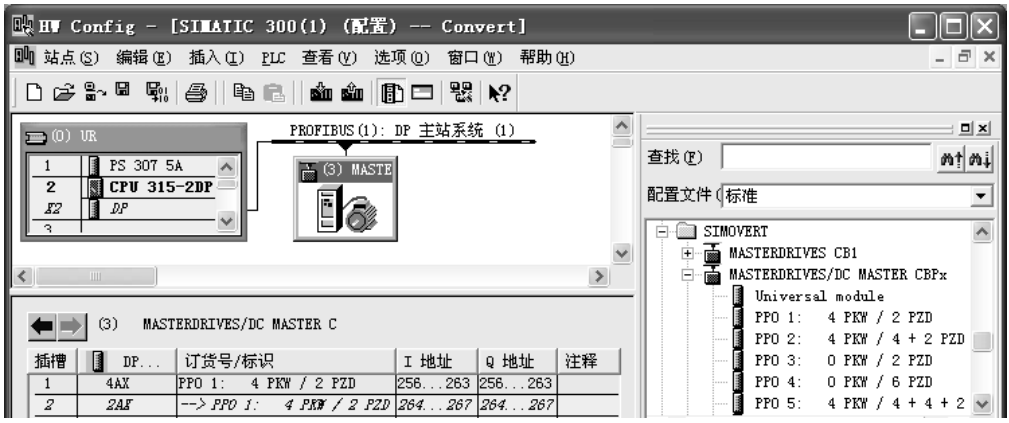


图 3-34　组态 DP 网络中的变频器

双击 CPU 模块中“DP”所在的行，点击打开的对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中点击“新建”按钮，生成一个 PROFIBUS-DP 网络。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮返回 HW Config。

3．生成 MASTERDRIVE 从站

CBP（Communication Board PROFIBUS）接口板是 SIMOVERT MASTERDRIVES 的 DP 通信扩展板，CBP2 是较新的版本。它们有一个电位隔离的 RS-485 接口，可以使用 PROFIBUS 协议或 USS 协议进行通信。

打开 HW Config 右边的硬件目录窗口的文件夹“\PROFIBUS DP\SIMOVERT”，将其中的“MASTERDRIVES/DC MASTER CBPx”或“MASTERDRIVES/DC MASTER CBP2 DPV1”

拖放到 DP 网络上（见图 3-34），作者使用的是 CBP 板。在自动打开的“属性 － PROFIBUS 接口”对话框中，设置从站地址为 3。两种 CBP 板的区别在于 CBPx 只能选择 PPO 类型的报文结构，CBP2 DPV1 还能选择更多的报文结构，以支持 CBP2 扩展的 DPV1 功能。

如果安装了 TIA（全集成自动化）软件 Drive ES，可以将硬件目录窗口的\SIMOVERT \MASTERDRIVES CBP 中增加的“Vector Control CUVC”拖放到左边的硬件组态窗口的 DP 网络上。CBP2 的某些扩展功能需要借助于软件 Drive ES 来实现。

4．变频器的通信区选择

双击打开硬件目录中的子文件夹“MASTERDRIVES/DC MASTER CBPx”，文件夹内是 CBP 板的通信区选项（见图 3-34）。

过程数据 PZD 用于 PLC 控制和监视变频器，参数数据 PKW 用于读写变频器的参数。PKW 和 PZD 总称为参数过程数据对象（PPO）。组态时一般选择 PPO1 和 PPO3。PPO1 有 4 个字的参数数据 PKW 和两个字的过程数据 PZD。系统调试好后交付用户使用时，一般选择 PPO3，它只有两个字的过程数据 PZD，可以监控变频器和电动机的运行，但是不能修改组态的参数。

选中硬件组态窗口中的变频器，就像将模块插入 ET 200M 的插槽一样，将图 3-34 中的“PPO1：4 PKW / 2 PZD”拖放到下面的窗口的第 1 行。下面的窗口自动生成两行信息，第 1 行是 PKW，第 2 行是 PZD，可以看到自动分配给它们的输入、输出地址。

双击某一行，可以看到该行参数的属性（见图 3-35）。一致性被设置为“总长度”，表示通信的数据是一致性数据，主站需要调用 SFC 14 和 SFC 15 将数据打包后发送，将接收到的数据解包。因为是灰色的字和背景色，不能修改一致性属性。

图 3-35　DP 从站属性对话框

5．变频器的参数设置

变频器在运行之前需要设置大量的参数，首先设置参数 P60=1，P366=0，P970=0，恢复工厂设置，各参数被设置为默认值。然后根据电动机、变频器和反馈元件（例如增量式编码

器）的具体情况，设置必要的参数。下面主要介绍与通信有关的参数设置：

- P53=7，允许使用 CBP 通信板、参数设置单元和串行通信接口来修改参数。
- P107=50Hz，电动机额定频率。
- P443.001=K3002，主设定值来自 PZD2。
- P554.001=B3100，用控制字的第 0 位来控制电动机的起停。
- P734.001=K32，PZD1 为状态字。
- P734.002=KK151，PZD2 为 n/f 模式的频率实际值。
- P918=3，通信板的 DP 站地址。
- P722=10ms，通信板（CBx）或工艺板（TB）的输入报文故障时间。在设置的时间内如果没有接收到有效的报文，说明通信出现故障。
- P571=B3101，P572 恒为 1，用控制字的第 1 位 B3101 控制电动机的正、反转。

3.3.2 SIMOVERT MASTERDRIVES DP 通信的数据区结构

SIMATIC 变频器 DP 通信协议的通信数据包括参数区 PKW 和过程数据区 PZD（见图 3-36）。

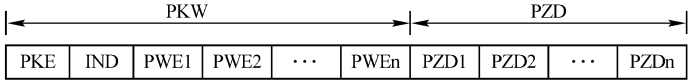


图 3-36　变频器通信数据区的结构

SIMOVERT MASTERDRIVES 有 5 种 PPO（PPO1～PPO5，见图 3-34）。以 PPO1 为例，“4PKW / 2PZD”表示有 4 个 PKW 字，两个 PZD 字。

过程数据区 PZD 包括主站发送给变频器的控制字和主设定值，以及变频器返回给主站的状态字和主实际值。表 3-1 给出了 PZD1 和 PZD2 的意义。

表 3-1　PZD 区的数据

	PZD1	PZD2
主站→变频器	控制字 1	主设定值
变频器←主站	状态字 1	主实际值

控制字 1 各位的意义见表 3-2，状态字 1 各位的意义见表 3-3。除了控制字 1 和状态字 1 外，还有用得较少的控制字 2 和状态字 2。它们各位的意义见驱动设备的使用手册。

表 3-2　过程数据中的控制字

位	意　义	位	意　义
0	上升沿起动/为 0 时为 OFF1（斜坡下降停车）	6	设定值使能
1	OFF2，为 0 时惯性自由停车	7	上升沿时确认故障
2	OFF3，为 0 时快速停车	8	正向点动，第 0 位应为 0
3	逆变器脉冲使能，运行的必要条件	9	反向点动，第 0 位应为 0
4	斜坡函数发生器使能	10	由 PLC 进行控制
5	为 0 时斜坡函数发生器保持	11	顺时针旋转磁场使能

（续）

位	意　义	位	意　义
12	反时针旋转磁场使能	14	用电动电位计降速
13	用电动电位计升速	15	为 0 时为外部故障命令

第 15 位为 0 时，如果有故障信号，将封锁逆变器脉冲，断开主接触器/旁路接触器。

【例 3-1】 用 P107 设置的额定频率为 50.00Hz，额定值对应于 16#4000。如果设定频率为 40.00Hz，试确定 PZD2（主设定值）的值。

主设定值为

$$PDZ2=\frac{40.00}{50.00}\times 16\#4000=16\#3333$$

控制字 STW、主设定值 HSW、状态字 ZSW 和主运行参数 HIW 的详细情况见随书光盘中的文件《SIMOVERT MASTERDRIVES 使用大全》（上、下两册）。

表 3-3　过程数据中的状态字

位	意　义	位	意　义
0	开机准备好	8	为 0 时频率设定值与实际值偏差过大
1	运行准备就绪	9	PZD 控制请求
2	正在运行	10	实际频率大于等于设定值
3	故障信号	11	中间回路低电压故障
4	为 0 时已发出 OFF2 关机命令	12	主接触器合闸
5	为 0 时已发出 OFF3 关机命令	13	斜坡函数发生器被激活
6	开机封锁信号	14	为 1 时为顺时针旋转磁场
7	有报警信息	15	动能缓冲（KIP）或柔性跳闸（FLN）激活

3.3.3　S7-300 与 SIMOVERT MASTERDRIVES 的 DP 通信实验

1．读写过程数据区的程序

因为是一致性数据，调用 SFC 14 和 SFC 15 来读写过程数据区 PZD 中的数据。由图 3-34 可知，PZD 区的起始地址为 264（即 W#16#108），长度为 4B。下面是 OB1 中的程序，在 M0.1 为 1 时，发送和接收数据。

程序段 1：读写过程数据

```
      A     M     0.1                  //M0.1 为 1 时发送数据
      JNB   _001                       //未满足条件则跳转
      CALL  "DPWR_DAT"                 //调用 SFC 15，将数据打包后发送
        LADDR   :=W#16#108             //PZD 输出区的起始地址（264）
        RECORD  :=P#M30.0 BYTE 4       //存放要发送的用户数据的源数据区
        RET_VAL :=MW6                  //错误代码
_001:   NOP   0
      CALL  "DPRD_DAT"                 //调用 SFC 14，将接收的数据解包
        LADDR   :=W#16#108             //PZD 输入区的起始地址（264）
        RET_VAL :=MW8                  //错误代码
```

```
RECORD  :=P#M34.0 BYTE 4          //存放读取的用户数据的目的数据区
```

因为 PPO1 只有 4B 的过程数据，也可以用语句表中的 L 和 T 指令或梯形图中的 MOVE 指令来按双字直接读写过程数据。

2．用过程数据区监控变频器的实验

控制字的第 10 位必须为 1，表示控制字来自 PLC。下面仅介绍用控制字的第 0 位控制电动机的起动和停止，用控制字的第 1 位控制电动机的旋转方向的方法。

用变量表将控制字 16#0403（反时针起动）写入 PZD 的第一个字 MW30（见图 3-37），将频率设定值 16#2000（50%的额定频率）写入 PZD 的第二个字 MW32。点击工具栏上的按钮，数据被写入 CPU，“状态值”列显示的是 CPU 中的数据。用变量表将 M0.1 置为 1 状态（true），控制字和设定值被发送到变频器，电动机开始旋转。MW36 返回的频率实际值逐渐增大，上升的速度取决于参数 P462.001 设置的加速时间的值，最后实际值在设定值 16#2000 上下窄幅波动。变频器的参数设置单元（PMU）显示的频率值在 25.0 Hz 上下波动。

变量表中的 MW34 返回的状态字 16#5334 的意义如下：变频器正在运行，未发出 OFF2 和 OFF3 关机命令，频率偏差未超出运行值，PZD 控制请求，主接触器合闸，顺时针旋转。

用变量表将控制字 16#0400（见图 3-38）写入 PLC，电动机首先减速，减速时间取决于参数 P464.001 的值，最后停止转动，返回的频率实际值为 16#FFFF（对应于数值 0）。变频器的参数设置单元显示 o009（开机准备好）。

VAT_1 -- @Convert\SIMATIC...

	地址		显示格式	状态值	修改数值
1	MW	30	HEX	W#16#0403	W#16#0403
2	MW	32	HEX	W#16#2000	W#16#2000
3	M	0.1	BOOL	true	true
4	MW	34	HEX	W#16#5334	
5	MW	36	HEX	W#16#2005	

图 3-37　用变量表监控过程数据 PZD

VAT_1 -- @Convert\SIMATIC...

	地址		显示格式	状态值	修改数值
1	MW	30	HEX	W#16#0400	W#16#0400
2	MW	32	HEX	W#16#2000	W#16#2000
3	M	0.1	BOOL	true	true
4	MW	34	HEX	W#16#4331	
5	MW	36	HEX	W#16#FFFF	

图 3-38　用变量表监控过程数据 PZD

MW34 返回的状态字为 16#4331，第 0 位为 1，表示合闸准备好；第 2 位为 0，表示变频器停止运行；第 12 位为 0，表示主接触器断开。其余各位的状态与正转运行时的状态字 16#5334 的相同。

用变量表发送控制字 16#0401（反转起动）和主设定值 16#1000（见图 3-39），电动机顺时针旋转。MW36 返回的转速实际值在 16#F000（即-16#1000）上下窄幅波动。变频器的参数设置单元显示的频率值在-12.5Hz 上下波动。MW34 返回的状态字 16#1334 的第 14 位为 0，表示顺时针旋转。其余各位与反时针旋转时的状态字 16#5334 的相同。

用变量表发送控制字 16#0403（正转起动）和设定值 16#F000（速度值为负值-16#1000，见图 3-40），也可以使电动机顺时针旋转。MW34 返回的状态字 16#1334 与图 3-39 中的相同。

用变量表发送控制字 16#0400，电动机减速后停止转动，返回的频率实际值为 16#0000。变频器的参数设置单元显示 o009（开机准备好）。

图 3-39　用变量表监控过程数据 PZD

图 3-40　用变量表监控过程数据 PZD

3．参数区 PKW 的结构

参数区的第 1 个字 PKE 和第 2 个字 IND 的结构见图 3-41。PKE 最高的 4 位 AK 是任务标识符或应答标识符，其意义分别见表 3-4 和表 3-5。

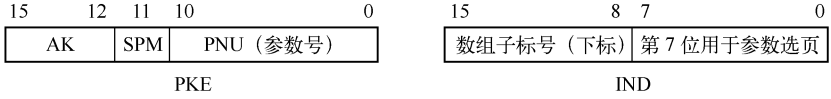


图 3-41　参数区 PKW 的结构

表 3-4　主站请求的任务标识符

编　号	意　义	编　号	意　义
0	没有任务	8	修改字数组中的双字参数值
1	请求读参数值	9	请求读数组元素的序号
2	修改单字参数值	10	保留未用
3	修改双字参数值	11	修改数组中的双字参数，保存在 EEPROM 中
4	请求读说明元素	12	修改数组中的字参数，保存在 EEPROM 中
5	修改说明元素	13	修改双字参数，保存在 EEPROM 中
6	请求读数组中的参数值	14	修改字参数，保存在 EEPROM 中
7	修改数组中的字参数值	15	读出或修改报文

表 3-5　从站的应答标识符

编　号	意　义	编　号	意　义
0	没有应答	8	没有控制和修改 PKW 接口的权限
1	传送字参数值	9	字参数变更报告
2	传送双字参数值	10	双字参数变更报告
3	传送说明元素	11	数组中的字参数变更报告
4	传送单字数组参数值	12	数组中的双字参数变更报告
5	传送双字数组参数值	13	预留
6	传送数组元素的编号	14	预留
7	任务不能执行，有错误号	15	传送报文

PKE 的第 0～10 位是二进制的基本参数号。第 11 位 SPM 是报告参数变化的触发位，SIMOVERT MASTERDRIVES 不支持它，该位总是为 0。

使用 PPO 进行周期性通信时，PKW 区的第 2 个字 IND 的第 8～15 位（高字节）为数组参数的子标号（或称下标、索引号）。IND 的第 7 位用于参数页的选择。读写参数 P000～P999 时，IND 的第 7 位和低位字节为 0。参数 U000～U999 的参数号实际上为 2000～2999，读写这些参数时，IND 的第 7 位为 1，低位字节为 16#80。

PPO 报文用双字来传送 1 个参数，PKW 区的第 3 个字 PWE1 和第 4 个字 PWE2（见图 3-36）分别是双字的高位字和低位字。数据为 16 位的字时用 PWE2 来传送，此时 PWE1 为零。

4．读写参数区数据的程序

读写参数数据区的程序与读写过程数据区的程序基本上相同。其区别主要在于数据区的起始地址和字节数不同。下面是 OB1 中的程序：

程序段 2：读写变频器的参数

```
      A     M      0.0                  //M0.0 为 1 时发送和接收数据
      JNB   _002                        // M0.0 为 0 则跳转
      CALL  "DPWR_DAT"                  //调用 SFC 15
        LADDR  :=W#16#100               //PKW 输出区的起始地址（256）
        RECORD :=P#M10.0 BYTE 8         //存放要发送的用户数据的源数据区
        RET_VAL :=MW2                   //错误代码
      CALL  "DPRD_DAT"                  //调用 SFC 14
        LADDR  :=W#16#100               //PKW 输入区的起始地址（256）
        RET_VAL :=MW4                   //错误代码
        RECORD :=P#M20.0 BYTE 8         //存放读取的用户数据的目的数据区
_002:   NOP   0
```

【例 3-2】 读取斜坡发生器的加速时间 P462.001（0.0～999.9ms，单位为 0.1ms）。

462 对应的十六进制数为 W#16#1CE，发送的第 1 个字 PKE 为 W#16#61CE（见图 3-42）。最高位的 16#6 表示请求读数组中的参数值（见表 3-4）。第 2 个字为 W#16#0100，高字节为数组子标号 1。用变量表将它们写入 MD10。第 3、4 个字为 0，用变量表将它们写入 MD14。M0.0 为 1 状态（true）时，数据被发送给变频器。

变量表中的 MD20 的高位字 16#41CE 是返回的第 1 个字。最高位的 16#4 表示返回的是单字数组参数（见表 3-5），前两个字的其余部分与发送的内容相同。第 4 个字是接收到的加速时间（3s），16#1E=30，单位为 0.1s。

【例 3-3】 将参数 P462.001 改写为 8s，单位为 0.1s。

发送的第一个字为 16#71CE（见图 3-43），最高位的 16#7 表示修改数组中的单字参数。8s（80）对应的十六进制数为 16#50，第 4 个字是新的参数值 16#0050。

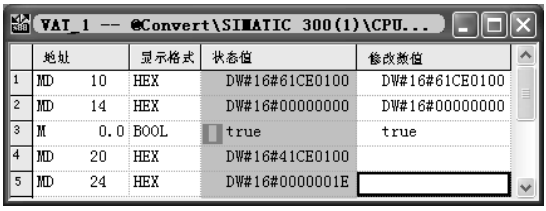


图 3-42　读参数 P462.001 的变量表

图 3-43　写参数 P462.001 的变量表

返回的第 1 个字的最高位 16#4 表示返回的是单字数组参数，其余的数据与发送的相同。

【例 3-4】 将简单斜坡函数发生器的上升时间 U383.001 改写为 2.56s，单位为 0.01s。

383 对应的十六进制数为 W#16#17F，发送的第 1 个字为 W#16#717F（见图 3-44），最高位的 16#7 表示修改数组中的单字参数。第 2 个字为 W#16#0180，高字节为数组子标号 001。低字节 16#80 的第 7 位为 1，表示该参数的类型为 U，参数的实际编号为 2383。第 3 个字为 0，第 4 个字是要修改的数值 16#100，对应的十进制数为 256（即 2.56s）。

返回的第 1 个字的最高位 16#4 表示返回的是单字数组参数，其余的数据与发送的相同。

【例 3-5】 将工艺调节器给定的二进制连接器（BiCo）参数 U352 改写为 16#3002。

352 对应的十六进制数为 W#16#160，发送的第 1 个字为 W#16#2160（见图 3-45），最高位的 16#2 表示要改写单字参数值（见表 3-4）。因为是非数组单字参数，没有数组子标号，第 2 个字的高位字节为 0，低位字节 16#80 的第 7 位为 1，表示该参数的类型为 U。第 3 个字为 0，第 4 个字为新的参数值 16#3002。

图 3-44　改写参数 U383.001 的变量表

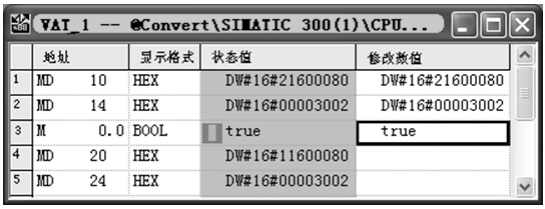


图 3-45　改写参数 U352 的变量表

返回的第一个字的最高位 16#1 表示传送的是单字参数值，其余部分与发送的相同。

3.3.4　S7-300 与 MM440 变频器的 DP 通信

1．组态主站和 PROFIBUS 网络

在 STEP 7 中用新建项目向导创建一个项目（见随书光盘中的例程 MM440），CPU 模块为 CPU 315-2DP。选中 SIMATIC 管理器的 300 站点，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 3-46），将电源模块和信号模块插入机架。

双击 CPU 模块中“DP”所在的行，点击打开对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中点击“新建”按钮，生成一条 PROFIBUS-DP 网络。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮返回 HW Config。

2．生成 MICROMASTER 从站

打开 HW Config 右边的硬件目录窗口的文件夹“\PROFIBUS DP\SIMOVERT”，将其中的“MICROMASTER 4”拖放到 DP 网络上（见图 3-46）。在自动打开的“属性 － PROFIBUS 接口”对话框中，设置从站地址为 3。

选中生成的 3 号从站，打开硬件组态窗口中的子文件夹“MICROMASTER 4”，将硬件目录窗口中的“4 PKW, 2 PZD（PPO1）”拖放到下面的窗口的第 1 行。下面的窗口自动生成两行信息，双击第 1 行的 PKW，将输入、输出起始地址修改为 60。双击第 2 行的 PZD，将输入、输出起始地址修改为 68。

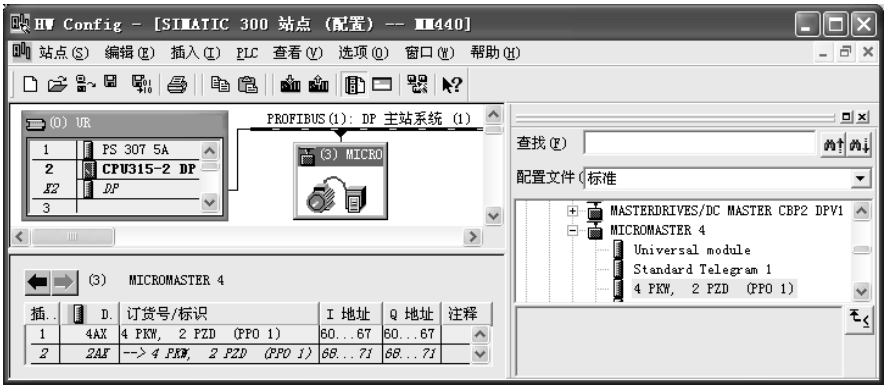


图 3-46　组态 DP 网络中的变频器

3．用过程数据监控 MM440

MM440 的控制字与 SIMOVERT MASTERDRIVES 的基本上相同（见表 3-2），其区别在于第 11 位用于设定值反向，第 12 位未用。第 15 位为本机/远程控制。

MM440 的状态字的第 0～10 位与 SIMOVERT MASTERDRIVES 的相同（见表 3-3）。第 11～15 位的意义如下：电动机电流极限报警、电动机抱闸制动投入、电动机过载、电动机正向运行、变频器过载。控制字 STW 和状态字 ZSW 的详细情况见随书光盘中的手册《MM440 使用大全》。

用过程数据监控 MM440 的方法与监控 SIMOVERT MASTERDRIVES 的相同（见 3.3.3 节）。下面是 OB1 中用双字读写过程数据的程序：

```
        A     M      0.1
        JNB   _001                  //M0.1 为 0 则跳转
        L     MD     30
        T     QD     68             //控制字和频率设定值送变频器
        L     ID     68
        T     MD     34             //保存状态字和频率实际值
_001:   NOP   0
```

4．变频器的参数设置

下面是 MM440 与通信有关的参数设置：

- P700=6，选择命令源为来自通信板 CB 的远程控制。
- P918=3，通信板的 DP 站地址。变频器上的 DIP 开关的地址设置为 0 时，DP 地址由参数 P918 提供。
- P1000=6，频率设定值来自远程 CB。
- P927=15，允许用所有的通信接口和 BOP（基本操作面板）修改参数。

5．参数区 PKW 的结构

参数区的第 1 个字 PKE 和第 2 个字 IND 的结构与 SIMOVERT MASTERDRIVES 的相同（见图 3-41），参数编号为十六进制数。PKE 的第 12～15 位为主站的任务标识符或从站的应答标识符，与 SIMOVERT MASTERDRIVERT 的基本上相同（见表 3-4 和表 3-5），未使用第 9～14 位。PKW 区的第 2 个字 IND 的格式和各部分的作用与 SIMOVERT MASTERDRIVERT

的完全相同，高字节为数组参数下标。

PKW 区的第 3 和第 4 个字 PWE1 和 PWE2 是被访问参数的数值，PPO 报文用双字来传送 1 个参数，PKW 区的第 3 和第 4 个字 PWE1 和 PWE2 分别是双字的高位字和低位字。数据为 16 位的字时，用 PWE2 来传送，此时 PWE1 为零。

【例 3-6】 读出参数 P0700（选择命令源）。

700=16#2BC，PLC 发送给变频器的 PKW 为 16#12BC 0000 0000 0000，请求读参数 P0700 的数值。最高位的 16#1 表示读参数值。因为是无下标变量，第 2 个字 IND 的高字节为 0。读写参数 P000～P1999 时，IND 的低字节为 0。

变频器返回给 PLC 的应答报文为 16#12BC 0000 0000 0006，第 1 个字的最高位的 16#1 表示传送的是字参数。P0700 是一个单字长的参数，读出的数值为 6（来自 COM 链路的通信板）。

【例 3-7】 读取参数 P2000（基准频率）的数据。

PLC 发送给变频器的 PKW 为 16#1000 0080 0000 0000，请求读参数 P2000 的数值。第 1 个字的最高位的 16#1 表示读参数值，0～10 位为参数号与 2000 的差值 0。

因为是无下标变量，第 2 个字 IND 的高字节为 0。读写参数 P2000～P3999 时，第 2 个字 IND 的低字节为 16#80。变频器返回给 PLC 的应答报文为 16#2000 0080 4248 0000，读取的是浮点数参数值 16#4248 0000，即 50.00 Hz。

【例 3-8】 读取下标参数 P2010[1]（USS 波特率）的数值。

方括号中是下标值。PLC 发送给变频器的 PKW 为 16#100A 0180 0000 0000，第 1 个字的最高位的 16#1 表示读参数值，第 0～10 位为参数号 2010 与 2000 的差值 16#0A。第 2 个字 IND 的高字节为下标值 16#01。读写参数 P2000～P3999 时，IND 的低字节为 16#80。

变频器返回给 PLC 的应答报文为 16#100A 0180 0000 0006，最后一个字的值为 6，查手册可知读取的 USS 波特率为 9600 bit/s。

用 SFC 14 和 SFC 15 读取和改写变频器参数的程序和实验过程与 3.3.3 节的相同。SFC 14 和 SFC 15 的参数 LADDR（变频器参数区的起始地址）为 16#3C（十进制数 60）。

3.3.5 S7-300 与其他厂家变频器的 DP 通信

本节以丹麦丹佛斯公司的 VLT 5000 系列变频器为例，介绍其他厂家的变频器作 PROFIBUS-DP 从站的组态与编程的方法。

1．组态主站

在 STEP 7 中用新建项目向导创建一个项目（见随书光盘中的例程 Danfoss），CPU 模块为 CPU 313C-2DP。在 SIMATIC 管理器左边的窗口中选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 3-47），将电源模块和信号模块插入机架。

双击 CPU 模块中“DP”所在的行，点击打开的 DP 属性对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中点击“新建”按钮，生成 PROFIBUS-DP 网络。点击“确定”按钮，返回 HW Config。采用默认的参数，CPU 313C-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。

2．安装 GSD 文件

VLT 5000 的 GSD 文件 da040402.GSD 在随书光盘的文件夹“\Project\ PB_MS”中。

执行菜单命令“选项”→“安装 GSD 文件”，点击“浏览”按钮，打开要安装的 GSD 文

件所在的文件夹，点击“确定”按钮后，该文件夹中的 GSD 文件出现在列表框。选中要安装的 GSD 文件，点击“安装”按钮，开始安装。安装结束后，可以在 HW Config 右边的硬件目录窗口的“PROFIBUS DP”文件夹中，找到新安装的设备（见图 3-47）。

图 3-47　组态 DP 网络中的变频器

3．生成从站

在硬件目录窗口中找到新安装的 VLT 5000，将它拖放到左边的硬件组态窗口中的 DP 网络线上。在自动打开的“属性 －PROFIBUS 接口”对话框中，设置变频器的站地址为 3。

4．组态变频器的通信区

选中硬件组态窗口中的变频器，将硬件目录窗口的子文件夹“VLT 5000”中的“PPO Typ 1 Module consisitent PCD”拖放到图 3-47 下面的窗口的第一行。下面窗口出现两行，第 1 行为 4 个字的参数数据，第 2 行为两个字的过程数据。分别双击表格的第 1 行和第 2 行，在出现的对话框中，可以看到数据的一致性均为“总长度”，表示需要调用 SFC 14 和 SFC 15，将参数数据和过程数据打包和解包。

5．通信数据区结构

因为都遵循 PROFIdrive 标准，VLT 5000 与 SIMOVERT MASTERDRIVES 的 DP 通信的组态和编程基本上相同。它们的通信数据区的结构相同（见图 3-48），只是数据的名称不同，VLT 使用的是英文的缩写。

通信数据各单元的意义如下：

（1）参数特性值区（Parameter Characteristics Value，PCV）

1）PCA（Parameter Characteristics）：参数特性字（见图 3-49）。第 12～15 位 RC 为请求/响应标识符（0～15），RC 的意义与表 3-4 和表 3-5 的基本上相同，详细的情况请参阅随书光盘中的文件《VLT 5000 PROFIBUS》。第 11 位 SPM 是自发信息（Spontaneous Message）触发位，第 0～10 位 PNU 为十六进制的参数号（1～990）。

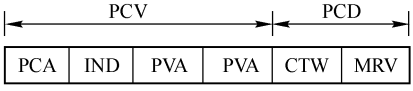


图 3-48　变频器通信数据区的结构

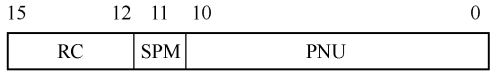


图 3-49　PCA 的结构

2）IND（Subindex）：第 3 字节为下标值，第 4 字节未用。

3）PVA（Parameter Value）：参数值，由两个字组成。

（2）过程数据（Process Data，PCD）

1）CTW（Control Word）为控制字，STW（Status Word）为状态字。

2）MRV（Main Reference Value）为主设定值，MAV（Main Actual Value）为主实际值。

【例 3-9】 将参数 207（加速时间 1）修改为 10.00s 后，启动变频器，速度设定值为 50%。

（1）修改参数

因为参数 207 为双字，参数特性值区 PCA 中的 RC 为 3。SPM 为 0，参数号 207 = 16#0CF，PCA 为 16#30CF。IND 为下标值，因为该参数没有下标，IND 为 0。

双字参数值 PVA 是以 0.01s 为单位的整数 1000（10.00s），对应的十六进制数为 16#0000 03E8。综上所述，PCV 为 16#30CF 0000 0000 03E8。

VLT 返回的数据为 16#20CF 0000 0000 03E8。

（2）发送过程数据

控制字 CTW 为 W#16#047F 或 2#0000 0100 0111 1111。

100%的主设定值 MRV 对应的十六进制数为#4000，50%对应的十六进制数为#2000。参数 202 用于设置最大频率，实际的频率设定值为最大频率的 50%。

综上所述，PCD 为 16#047F 2000。

6．通信程序设计

本节的例程中，需要传送的参数数据（8B）和过程数据（4B）的字节数和数据结构均与 3.3.2 节中的相同，都是一致性数据，因此程序的结构和设计方法与 3.3.3 节中的相同，需要调用 SFC 15 来将数据打包后发送，调用 SFC 14 将接收到的数据解包。具体的程序见随书光盘中的项目 Danfoss。

3.4 S7 PLC 与西门子直流调速装置的 DP 通信

3.4.1 系统组态与直流调速装置参数设置

1．SIMOREG DC-MASTER 简介

SIMOREG DC-MASTER 是全数字直流调速装置，它的输入为三相交流电源，可以通过调节直流电动机的电枢电流和励磁电流来调节 6～2500kW 电动机的转速，实现单/双象限或四象限传动。SIMOREG DC-MASTER 的电子箱可以插入各种工艺模块和通信模块。通过自动优化运行，可以实现 PID 参数自整定。它集成了工艺控制功能，可以进行转速计算、开关量和模拟量信号处理、算术运算，可以实现补偿控制、位置控制和压力控制。

2．组态主站

在 STEP 7 中新建一个项目（见随书光盘中的例程 DCMaster），用鼠标右键点击项目的图标，用出现的快捷菜单中的命令创建一个 S7-400 站点。在 SIMATIC 管理器左边的窗口选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 3-50），将 CPU 413-2DP、电源模块和信号模块插入机架。

双击 CPU 模块中“DP”所在的行，点击打开的 DP 属性对话框的“常规”选项卡中的“属

性”按钮，在出现的对话框的“参数”选项卡中点击“新建”按钮，生成一条 PROFIBUS-DP 网络，点击“确定”按钮返回 HW Config。采用默认的参数，CPU 413-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。

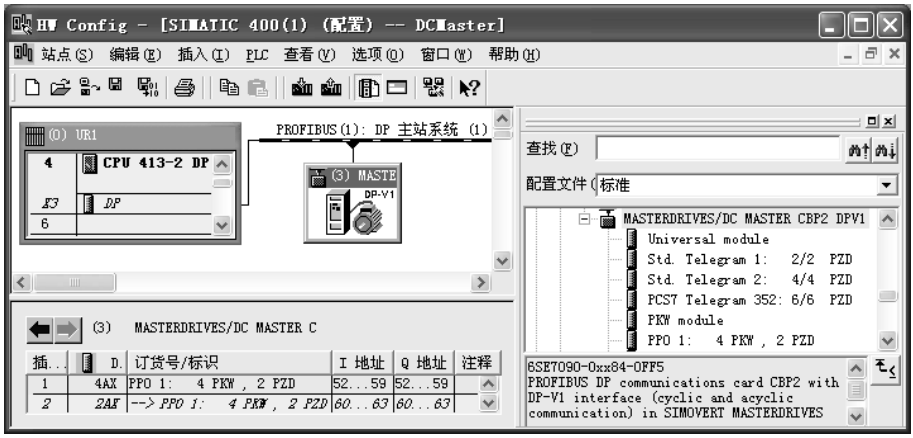


图 3-50　组态 DP 网络中的直流调速装置

3．生成从站

打开 HW Config 右边的硬件目录窗口的文件夹“\PROFIBUS DP\SIMOVERT”，将其中的“MASTERDRIVES/DC MASTER CBP2 DPV1”拖放到左边的硬件组态窗口中的 DP 网络线上（见图 3-50）。该通信板可以用于 SIMOVERT MASTERDRIVES 交流变频器和 DC MASTER 直流调速装置。

在自动打开的“属性 －PROFIBUS 接口”对话框中，设置 DC MASTER 的站地址为 3。

4．设置 DC Master 的通信区

选中硬件组态窗口中的 DC MASTER，打开子文件夹“MASTERDRIVES/DC MASTER CBP2 DPV1”，将其中的“PP01：4 PKW, 2 PZD”拖放到图 3-50 下面的窗口。第 1 行是 PKW，第 2 行是 PZD。双击第 1 行，在出现的 DP 从站属性对话框的“地址/ID”选项卡中，将 PKW 的输出和输入的起始地址改为 52。双击第 2 行，将 PZD 的输出和输入的起始地址改为 60。注意上述的地址区不能与主站和其他从站的输入、输出地址区重叠。

DC MASTER 传送的是一致性数据，主站需要调用 SFC 14 和 SFC 15 将数据打包后发送，将接收到的数据解包。

5．DC Master 的基本参数设置

DC Master 需要设置大量的参数，下面是与 DP 通信有关的几个主要的参数：首先设置 P51=21，恢复工厂设置，各参数被设置为默认值。用户需要设置电动机、反馈元件（例如增量式编码器）的额定参数，电动机电流、转矩的限幅值，加、减速时间等参数。

6．参数最优化运行

通过设置参数 P51，进行最优化运行，接通起动开关，开始优化。优化结束时显示 P51，断开起动开关。

1）令 P51=25，进行电流环优化（堵转实验）。

2）令 P51=26，进行速度环优化（空载实验）。

3）令 P51=27，进行弱磁优化。

4）令 P51=28，P223=1（摩擦和转动惯量补偿使能），进行摩擦和转动惯量补偿实验。

7．DC Master 与通信有关的参数设置

1）P918=3，CBP2 通信板的 DP 站地址。

2）P927=7，用于参数化使能，参数值只能通过使能的接口修改。P927=7 时使能了 DP 通信板、简易操作控制面板（PMU）和 DC MASTERG 面板上的 G-SST1 串行通信接口（X300）。

3）P644.001=K3002，主给定值来自过程数据的第 2 个字 PZD2。

4）U734.001=K32，返回的第 1 个字 PZD1 是状态字 1。

5）U734.002=K167，返回的第 2 个字 PZD2 是转速实际值。

参数中的 B 和 K 分别表示参数为位（bit）和 16 位的字。

DC Master 的控制字和状态字与 SIMOVERT MASTERDRIVES 的基本上相同（见表 3-2 和表 3-3），DC Master 未使用状态字的第 15 位。

3.4.2 S7 PLC 与直流调速装置通信的实验

1．控制字按位进行控制

控制字按位进行控制时应设置 P648=K9，参数 P654～P675 对应于控制字的各位，可以用它们来分别独立控制 DC Master。若 P648≠K9，只能以字为单位设置控制字 1。令 P648=K3101，控制字来自过程数据的第 1 个字 PZD1。

下面的实验中，只使用了控制字的第 0 位。令 P654=B3100，将参数 B3100（DC Master 接收到的第 1 个字的第 0 位）作为控制字的第 0 位。控制字的其他位（不包括第 10 位）使用默认值。各默认值具体的值见 DC Master 的手册对控制字各位的说明。

控制字的第 10 位比较特殊，不能单独进行控制。接收到的第 1 个 PZD 字的第 10 位应为 1，以保证过程数据作为有效数据被接收。因此控制字 1 必须作为 PZD 的第 1 个字来传送。

如果控制字的第 10 位为 0，接收到的第 1 个字的其他位和第 2～第 16 个字，都不能写入它们对应的参数 K3001～K3016 或 B3100～B3915 中。这些参数将保持它们原有的值。

DC Master 的端子 37（起动/停车开关）和端子 38（脉冲使能开关）总是激活的。它们分别与控制字的第 0 位（运行/停车）和第 3 位（脉冲使能）相“与”，因此在用 CBP2 通信板控制 DC Master 时，上述两个端子外接的触点应处于接通状态。否则不能用 CBP2 板控制 DC Master 的起动和停车。

在 OB1 中编写图 3-51 中的程序，分别用 I0.0 和 I0.1 来控制发送的控制字的第 0 位和第 10 位。做实验时，将端子 37 和 38 外接的触点接通，为控制作好准备。用外接的小开关使 I0.1 为 1，可以用 I0.0 控制电动机的起动和停车。如果令 I0.1 为 0，I0.0 不能控制电动机的起停。

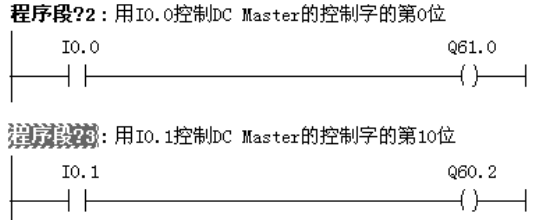


图 3-51　梯形图

2．控制字以字为单位进行控制

从以位为单位进行控制切换到以字为单位进行控制之前，需要作下面的准备工作：令P51=40（允许修改参数）；P648=K3001（用接收到的 PZD 的第 1 个字作控制字）；P654=1；还应删除图 3-51 中的程序。

做实验时，将端子 37 和 38 外接的触点接通，用变量表将 W#16#0401 写入 QW60（控制字 1），将给定值 16#2000 写入 QW62。即令控制字 1 的第 0 位 B3100 和第 10 位为 1，电动机起动运行。

用变量表将 W#16#0400 送给控制字 1，电动机停机。如果令控制字 1 的第 10 位为 0（发送 W#16#0001 或 W#16#0000），不能使电动机起动或停机。

3．读取和修改参数

下面是 OB1 中读取和修改参数的程序。

```
      A     M      0.1
      JNB   _001                    //M0.1 为 0 状态则跳转
      CALL  "DPWR_DAT"              //调用 SFC 15
        LADDR  :=W#16#34            //PKW 输出区的起始地址 QB52
        RECORD :=P#M 10.0 BYTE 8    //存放要发送的用户数据的源数据区
        RET_VAL :=MW2               //错误代码
_001:   CALL  "DPRD_DAT"            //调用SFC 14
        LADDR  :=W#16#34            //接收通信数据的 PKW 输入区起始地址 IB52
        RET_VAL :=MW4               //错误代码
        RECORD :=P#M 20.0 BYTE 8    //存放读取的用户数据的目的数据区
```

【例 3-10】 读取参数 P648.001（控制字的来源）的值。

发送的第 1 个字 PKE 为 W#16#6288，最高位的 6 表示要读取数组变量（或称下标变量），十六进制数的低 3 位 W#16#288 对应于 648。

因为子标号（或称下标）的值为 1，参数编号小于 2000，第 2 个字 IND 为 W#16#0100。第 3、4 个字（PWE1 和 PWE2）与读取操作无关，令它们为 0。

在变量表（见图 3-52）中将上述数据写入 MD10 和 MD14，在 M0.1 为 1 时将数据打包后发送给变频器。MD20 和 MD24 为返回值，第一个字 16#4288 的最高位为 4，表示返回的是数组变量。读取的参数值为 9（控制字 1 每一个单独的位由一个开关量连接器输入）。

【例 3-11】 读取无下标参数 P927（参数化接口使能）的值。

发送的第 1 个字为 W#16#139F（见图 3-53），最高位的 1 表示要读取无下标参数，十六进制数的低 3 位 W#16#39F 对应于 927。第 2～4 号字均为 0。返回的第一个字 16#139F 的最高位为 1，表示返回的是无下标参数。读取的参数 P927 的值为 7。

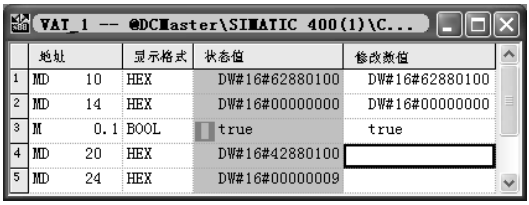


图 3-52　变量表

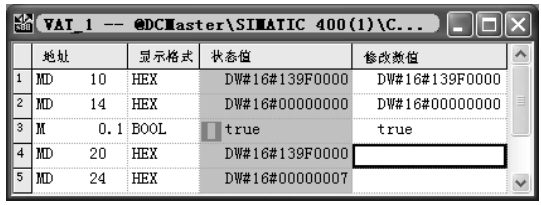


图 3-53　变量表

【例 3-12】 读取参数 U734.002 的值。

发送的第 1 个字为 W#16#62DE（见图 3-54），最高位的 6 表示要读取下标变量，十六进制数的低 3 位 W#16#2DE 对应于 734。第 2 个字为 W#16#0280。其中的 2 是下标编号，16#80 表示参数类型为 U。读取的参数 U734.002 的值为 16#167。

【例 3-13】 将参数P644.001 的值由 16#3002(给定值来自接收到的第 2 个字)改为 16#0206（由控制面板上的电位器给定）。

修改参数之前，应将 P51 设置为 40（允许修改参数），P927 的最低位应为 1（允许通过 CBP2 通信板修改参数)。

发送的第 1 个字为 W#16#7284（见图 3-55），最高的 7 表示要写下标变量，十六进制数的低 3 位 W#16#284 对应于 644。因为下标的值为 1，参数类型为 P，第 2 个字为 W#16#0100。第 3、4 个字为参数值，分别为 0 和 16#0206。

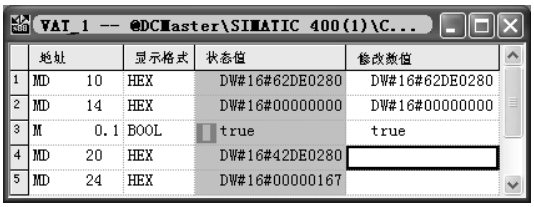


图 3-54 变量表

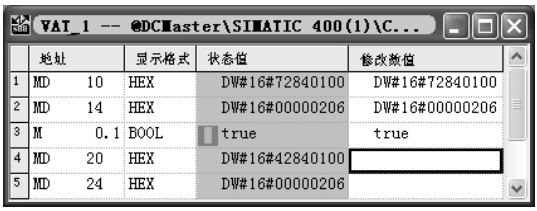


图 3-55 变量表

在变量表中将上述数据写入 MD10 和 MD14，在 M0.1 为 1 时将数据打包后发送给从站，MD20 和 MD24 是读取的该参数修改后的值（W#16#0206）。实验表明，电动机的速度不再受接收到的第 2 个字的控制，而是受外接的电位器输出的给定电压的控制。

3.5 通信处理器在 DP 主从通信中的应用

3.5.1 CP 342-5 作 DP 从站

1．CPU 集成的 DP 接口与 PROFIBUS 通信处理器

大多数 S7-300/400 CPU 都集成了 DP 接口。S7-300 集成的第一个通信接口是 MPI，S7-400 集成的第一个通信接口可以选择作 MPI 或 DP 接口。与使用 PROFIBUS 通信处理器（下面简称为 PROFIBUS CP）相比，带集成 DP 接口的 CPU 的硬件成本要低得多。

PROFIBUS CP 可以扩展 PLC 的通信接口，除此之外，PROFIBUS CP 的功能比集成的 DP 接口的功能更强。所有的 PROFIBUS CP 都支持 PG/OP（编程器/操作面板）通信、S7 通信和 S5 兼容的通信（FDL 通信），CP 343-5 还有 PROFIBUS-FMS 通信功能。CPU 集成的通信接口没有 FDL 通信和 PROFIBUS-FMS 通信功能。

CP 342-5 可以作 DP 网络的主站，也可以作从站，只能在 S7-300 的中央机架上使用，不能在分布式从站（例如 ET 200M）上使用。有的 CP（例如 CP 342-5 和 CP 443-5 Ext）没有 PROFIBUS-FMS 通信功能。CP 343-5 和 CP 443-5 基本型不支持 PROFIBUS-DP 协议通信。

如果只是用 CP 342-5 连接上位机软件或操作面板（OP），这时通信采用的是 S7 协议。CP 342-5 进行 PROFIBUS-DP 通信时，需要调用 FC 1（DP_SEND）和 FC 2（DP_RECV）。如

果选择“No DP”模式，不需要调用 FC 1 和 FC 2。

CP 342-5 作为 DP 主站时，最多链接 124 个从站，与每个从站最多可以交换 244 个输入字节和 244 个输出字节，与所有从站总共最多交换 2160 个输入字节和 2160 个输出字节。CP 342-5 作为从站时，与主站最多可以交换 240 个输入字节和 240 个输出字节。CP 342-5 最多可以连接 16 个操作面板（OP）。

2．组态主站

在本节的例程中，CPU 模块分别为 CPU 413-2DP 和 CPU 315-2DP，用前者集成的 DP 接口作主站，CP 342-5 作从站。

在 SIMATIC 管理器中创建一个新的项目，CPU 为 CPU 413-2DP，项目名称为“PB_MS_4”（见随书光盘中的同名例程）。

选中管理器中的“SIMATIC 400（1）”站对象，双击右边窗口中的“硬件”图标，打开 HW Config 窗口，将电源模块和信号模块插入机架。采用默认的设置，CPU 的 MPI 接口地址为 2。双击机架中 CPU 模块的“DP”所在的行，在打开的 DP 属性对话框中，点击“属性”按钮。在出现的对话框中，点击“新建”按钮，新建的 DP 网络采用默认的设置，传输速率为 1.5 Mbit/s，配置文件为“DP”。DP 接口的站地址为默认值 2，接口的工作模式为默认的主站模式。多次点击“确定”按钮，返回 HW Config，可以看到生成的 PROFIBUS 网络线。

3．组态从站

用鼠标右键点击 SIMATIC 管理器左边窗口最上面的项目对象，在打开的快捷菜单中执行命令“插入新对象”→“SIMATIC 300 站点”，插入新的 S7-300 站。

选中该从站后，双击右边的“硬件”图标，打开 HW Config，将机架拖放到左边的窗口，将 CPU 315-2DP 插入机架。在自动打开的“属性 – PROFIBUS 接口 DP”对话框中，设置其 MPI 地址和集成的 DP 接口的地址均为 3，不连网。点击“确定”按钮返回 HW Config，在机架中插入一块 CP 342-5，设置其 DP 站地址和 MPI 站地址均为 4（见图 3-56）。

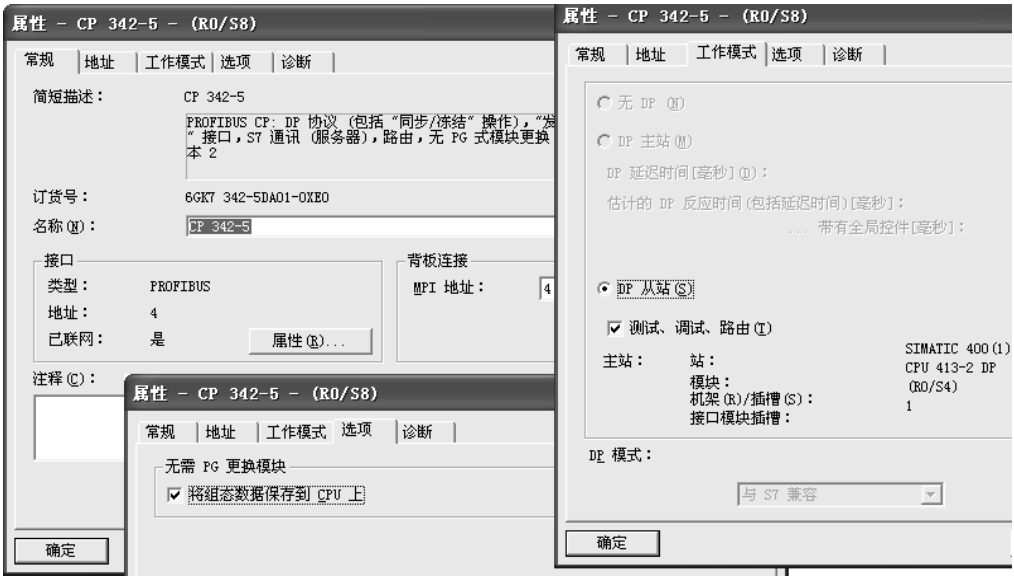


图 3-56　CP 342-5 属性对话框

打开“选项”选项卡，选中复选框“将组态数据保存到 CPU 上”（见图 3-56），CP 342-5 的组态信息将存储在 CPU 的装载存储区。CPU 掉电后再次上电时，CPU 将组态信息传送给 CP。如果没有选中该选项，组态信息保存在 CP 342-5 中，更换 CP 342-5 后，必须重新下载组态信息，否则 CPU 不能识别 CP 342-5。

点击“确定”按钮返回 HW Config 后，双击 CP 342-5，打开它的属性对话框。在“工作模式”选项卡（见图 3-56），可以选择 3 种工作模式：

1）“无 DP”模式：可以用 CP342-5 通信口进行 S7 编程或建立 PROFIBUS 的 FDL 连接。

2）“DP 主站”模式：CP342-5 除了作为网络中的 PROFIBUS 主站之外，也可以用于 S7 编程、FDL 连接和连接人机界面。

3）“DP 从站”模式，本项目选中了从站模式。

如果选中了复选框“测试、调试、路由”，除了作 DP 从站之外，CP 342-5 可以用于编程（例如下载）、测试（状态监视与修改变量的值）、S7 路由（作为网关）、S5 兼容的通信、作为服务器的 S7 通信。选中此选项后，可能会增加令牌循环时间。在对时间要求苛刻的场合，以及通信不需要 S7 路由和客户机功能时，不要激活此选项。

如果没有选中此复选框，CP 称为“被动的 DP 从站”。DP 接口只能作 S7 通信的服务器，即只有通信伙伴（例如 PG、OP 或自动化系统）才能启动通信。

在“地址”选项卡可以设置 CP 的输入/输出起始字节地址，默认值与 CP 所在的槽数有关，本例为 320（W#16#140），输入/输出地址区的长度分别为 16B。这个 16B 长度的地址区是 CPU 分配给 CP 342-5 的硬件地址区，是 CPU 和 CP 342-5 之间在主站内部进行数据交换的缓存，CPU 就是通过这个硬件地址区访问 CP 342-5 模块的。这 16B 的地址数据区与 CP 342-5 连接的 PROFIBUS 从站没有直接的关系，它并不影响主站所带的从站个数，以及主站和从站交换数据的长度。CP 342-5 与 PROFIBUS 从站进行数据交换使用的是另外一个独立的数据存储区，输入、输出区均为 2160B。

打开 S7-400 站的硬件组态窗口，将硬件目录窗口的文件夹“\PROFIBUS DP\Configured Stations\S7-300 CP 342-5 DP”中的“6GK7 342-5DA0x-0XE0”拖放到左边窗口的 DP 网络线上（见图 3-57）。本例中 CP 342-5 的订货号为 6GK7 342-5DA01-0XE0。在自动打开的 DP 从站属性对话框的“连接”选项卡中，点击“连接”按钮，将 CP 342-5 连接到 DP 网络上。

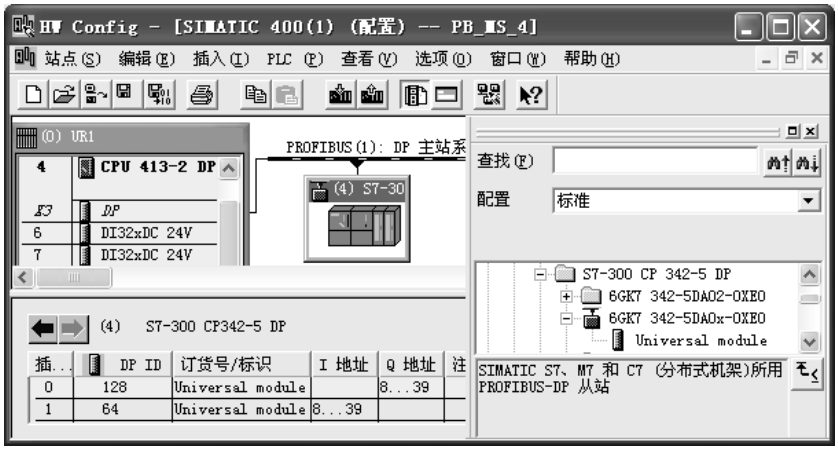


图 3-57　组态 DP 从站

选中硬件组态窗口中的从站（见图 3-57），将右边的硬件目录窗口的子文件夹“……\S7-300 CP 342-5 DP \6GK7 342-5DA0x-0XE0”中的两块“Universal module”（通用模块）插入 HW Config 左下侧窗口的 0 号和 1 号插槽中。

双击 CP 342-5 的 0 号槽的模块，在出现的 DP 从站属性对话框中，选择插入的模块的类型为“输出”（见图 3-58）。因为 S7-400 的中央机架中已经插入的输入、输出模块分别占用了 8 个字节，插入的模块的起始字节地址被自动设置为 8。设置模块的长度为 32B，传送的数据的一致性为“单位”（Unit），表示数据按单元（字节）组装数据包。

图 3-58　设置 CP 342-5 的地址

如果选择一致性为“总长度”（Total Length），表示数据整体组装为一个数据包。此时需要在主站的 OB1 中调用 SFC 14 和 SFC 15，来对传输的数据进行打包和解包处理。

如果设置的“长度”字节数超过模块允许的范围，将会出现错误信息，并显示所选的 CP 模块允许的输入、输出的最大字节长度。

双击 CP 342-5 的 1 号槽的模块，在出现的 DP 从站属性对话框中，选择插入的模块的类型为输入。模块的起始字节地址被自动设置为 8，设置模块的长度为 32B，传送的数据的一致性为“单位”。

从图 3-57 所示对话框下方的窗口可以看出，主站通过数据区 QB8~QB39 发送数据到从站，通过数据区 IB8~IB39 接收从站的数据。主站用上述 I/O 地址直接访问从站中的数据区，不需要编写通信程序。组态好 CP 342-5 后，如果将信号模块插入中央机架，可以看到，新插入的模块的起始字节地址为 40。由此可知，主站与从站的 I/O 地址是统一分配的。

点击 按钮，编译和保存组态信息。图 3-59 是组态好硬件后，在 NetPro 中看到的网络结构和站地址。

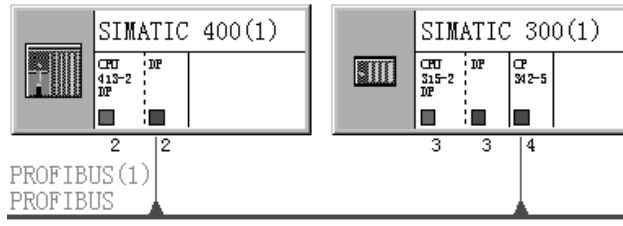


图 3-59　NetPro 中的 DP 网络

4．数据交换原理

DP 主站和 DP 从站之间通过 CP 的 DP 数据缓冲区进行的数据交换是周期性的，其周期

称为 DP 轮询周期。数据交换由 DP 主站启动，将输出数据发送到从站的 DP 数据缓冲区（见图 3-60），并接收来自从站的 DP 数据缓冲区的数据。

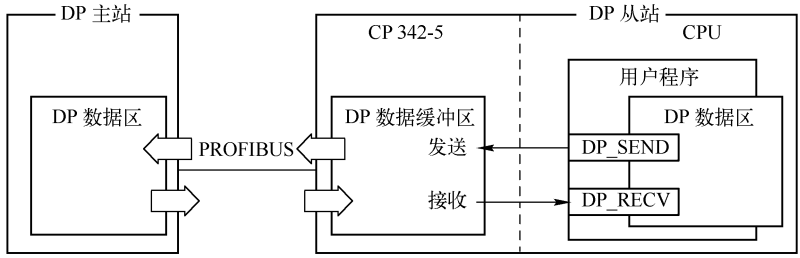


图 3-60　数据交换原理

从站的 DP 数据缓冲区和 CPU 之间的数据交换是通过调用下述的通信功能（FC）实现的：

1）DP 从站调用 FC 1（DP_SEND），将参数 SEND 指定的数据区中的数据传送到 CP 342-5，以便发送到 DP 主站。

2）DP 从站调用 FC 2（DP_RECV），将 CP 342-5 接收的 DP 主站发送的数据，保存到参数 RECV 指定的接收数据区。

3）参数 SEND 和 RECV 指定的数据区可以是过程映像区（I/Q）、位存储区（M）或数据块区（DB）。

5．编写验证通信的程序

FC 1 和 FC 2 在程序编辑器左边窗口的文件夹“\Libraries \SIMATIC_NET_CP\CP 300”中。在下面的例程中，FC 1 将 DB 1 中的 32B 数据打包后发送给 CPU 413-2DP 的 IB8~IB23（见图 3-61），FC 2 将来自 CPU 413-2DP 的 QB8~QB23 的数据存放到 DB2.DBB0~DBB31。

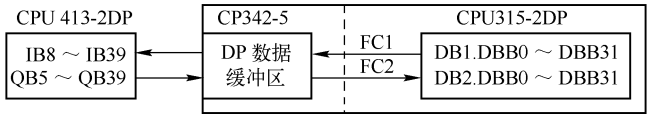


图 3-61　主站与从站之间的数据交换

下面是从站的 OB1 中的程序：

```
Network 1：将要发送的数据打包
        L       ID      0
        T       DB1.DBD     2                   //用本站的 ID0 控制对方的 QD0
        CALL    "DP_SEND"                       //调用 FC 1
          CPLADDR :=W#16#140                    //CP 342-5 的起始地址 320
          SEND    :=P#DB1.DBX0.0 BYTE 32        //保存要发送的数据的地址区
          DONE    :=M10.0                       //发送正确完成时产生一个脉冲
          ERROR   :=M10.1                       //错误标志位
          STATUS  :=MW12                        //状态字
Network2：将接收到的数据解包
        CALL    "DP_RECV"                       //调用 FC 2
          CPLADDR :=W#16#140                    //CP 342-5 的起始地址 320
```

```
RECV      :=P#DB2.DBX0.0 BYTE 32      //保存接收到的数据的地址区
 NDR      :=M0.0                      //接收正确完成时产生一个脉冲
ERROR     :=M0.1                      //错误标志位
STATUS    :=MW2                       //状态字
DPSTATUS:=MB4                         //DP 状态代码
L     DB2.DBD    2
T     QD         4                    //用对方的 ID0 控制本站的 QD4
```

FC 1 中的 SEND 和 FC 2 中的 RECV 的数据类型是以字节（BYTE）为单位的 ANY 指针。

为了观察数据传输的效果，在每 100ms 调用一次的 OB35 中，将 DB1.DBW0 加 1。

下面是 CPU 413-2DP 的 OB1 中的程序，CPU 413-2DP 的 ID10 和 QD10 对应于 CPU 315-2DP 的 DB1.DBD2 和 DB2.DBD2。

```
L     ID         10
T     QD         0                    //用对方的 ID0 控制本站的 QD0
L     ID         0
T     QD         10                   //用本站的 ID0 控制对方的 QD4
```

在 CPU 413-2DP 的 OB35 中，每 100ms 将发送给 CP 342-5 的第一个字 QW8 加 1。

在 CPU 413-2DP 的 OB100 中，将 QB8 开始的 16 个字（32B）赋初值 16#4444，将 IB8 开始的 32B 清零。在 CPU 315-2DP 的 OB100 中，将数据块 DB1 中的 16 个字（32B）赋初值 16#3333，将 DB2 的 32B 清零。

将程序和组态数据下载到 CPU 后，用 PROFIBUS 电缆连接 CPU 413-2DP 集成的 DP 接口和 CP 342-5 的 DP 接口。在系统运行时，除了将 CPU 的模式开关切换到 RUN 模式外，还应将 CP 的模式开关扳到 RUN 位置。

同时打开主站和从站的变量表，图 3-62 和图 3-63 的前两行是被监控的两个站接收的数据中的第一个字和最后一个字。可以看到主站接收到的来自从站的 DB1.DBW0 的 IW8 的值在不断地变化。从站接收的来自主站的 QW8 的 DB2.DBW0 的值也在不断地变化。用双方的 ID0 可以控制对方的 QD0 或 QD4。

图 3-62　主站的变量表

图 3-63　从站的变量表

3.5.2　CP 443-5 Ext 与 CP 342-5 的 DP 通信

1．硬件组态

在 SIMATIC 管理器中创建一个新项目，CPU 分别为 CPU 413-2DP 和 CPU 315-2DP，项目名称为“PB_MS_5”（见随书光盘中的同名例程）。在 DP 通信中，CP 443-5 Ext 为主站，CP 342-5 为从站。

在组态 S7-400 站点时，将电源模块、信号模块插入机架。采用默认的设置，CPU 的 MPI 接口和 DP 接口的地址均为 2。将 CP 443-5 Ext 插入机架的 13 号槽时，在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，设置它的 DP 站地址为 3。点击“新建”按钮，生成一条新的 PROFIBUS 子网络，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，采用默认的传输速率（1.5M bit/s）和配置文件（DP），点击“确定”按钮返回 CP 属性对话框，将 CP 连接到 DP 网络上。

点击“确定”按钮，返回 HW Config，双击 CP 433-5Ext，打开它的属性对话框，在“工作模式”选项卡中，可以看到默认的设置为 DP 主站。

在 SIMATIC 管理器中生成一个 S7-300 站，打开 HW Config，将导轨（Rail）拖放到左边的窗口，将 CPU 315-2DP 插入机架，在自动打开的“属性 – PROFIBUS 接口 DP”对话框中，设置集成的 DP 接口的地址为 4，不连网，点击“确定”按钮返回 HW Config。双击机架中 CPU 所在的行，在打开的 CPU 属性对话框中，点击“常规”选项卡中的“属性”按钮，设置 CPU 的 MPI 地址为 3，不连网。

点击“确定”按钮返回 HW Config，插入一块 CP 342-5，设置其 DP 接口的地址为 5，MPI 接口的地址为 4，不连网。工作模式设置为 DP 从站。CP 默认的输入/输出起始字节地址均为 320。最后点击工具栏上的按钮，保存组态数据。

打开 S7-400 站的硬件组态窗口，将硬件目录窗口的文件夹“\PROFIBUS DP\Configured Stations\S7-300 CP 342-5 DP”中的“6GK7 342-5DA0x-0XE0”拖放到左边窗口的 DP 网络线上（见图 3–64）。在出现的 DP 从站属性对话框的“连接”选项卡中，点击“连接”按钮，将 CP 342-5 连接到 DP 网络上。

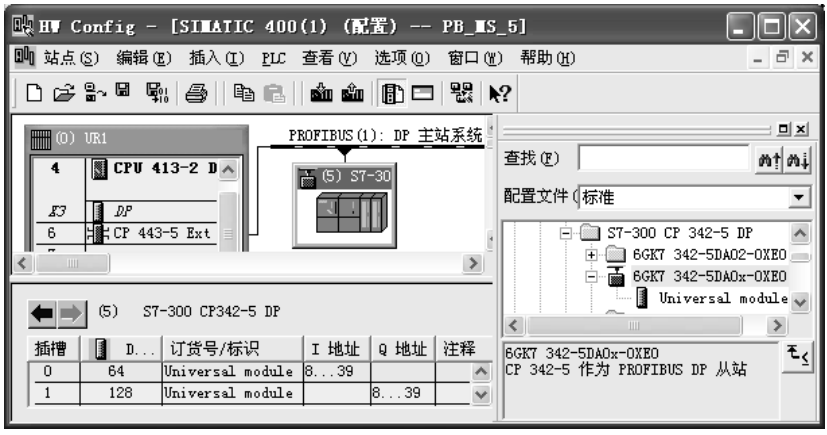


图 3–64　组态 DP 从站

选中硬件组态窗口中从站的图标，将硬件目录窗口的子文件夹“……\S7-300 CP 342-5 DP \6GK7 342-5DA0x-0XE0”中的两块“Universal module”（通用模块）插入 HW Config 左下侧窗口的插槽中。

双击 0 号槽的模块，在出现的 DP 从站属性对话框中，选择插入的模块的类型为“输入”。因为 S7-400 的中央机架中已经插入的输入、输出模块分别占用了 8 个字节，插入的模块的起始字节地址被自动设置为 8。设置模块的长度为 32B，传送的数据的一致性为“单位”（Unit）。

用同样的方法设置 1 号槽的模块为输出，其余的参数与 0 号槽的模块相同。

从图 3-64 下面的窗口可以看出，主站通过数据区 QB8~QB39 发送数据到从站，通过数据区 IB8~IB39 接收从站的数据。图 3-65 是组态好硬件后，NetPro 中的网络结构和站地址。

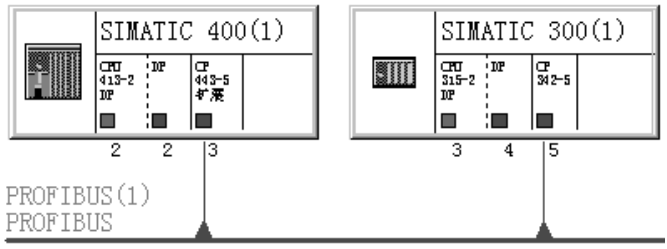


图 3-65　NetPro 中的 DP 网络

2．通信程序

本节的项目“PB_MS_5”和 3.5.1 节的项目“PB_MS_4”除了 DP 主站分别是 CP 443-5 Ext 和 CPU 413-2DP 集成的 DP 接口之外，其他硬件组态基本上相同。

与 CPU 集成的 DP 接口一样，主站的中央机架的 I/O 地址和 CP 443-5 Ext 的从站的 I/O 地址是统一分配的。CP 443-5 Ext 在 DP 主站模式时，CPU 通过 IB8～IB39 和 QB8～QB39 直接访问或通过系统功能块 SFC 14/15 访问从站用 FC 1 和 FC 2 指定的数据区（见图 3-66）。

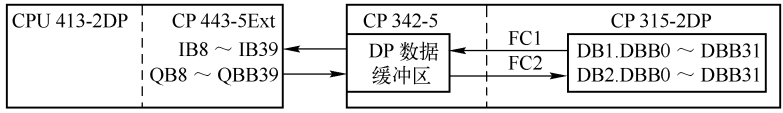


图 3-66　主站与从站之间的数据交换

因为从站通过 CP 342-5 通信，从站需要在 OB1 中调用 FC 1，将 DB1.DBB0~DBB31 打包发送给 CPU 413-2DP 的 IB8~IB39；调用 FC 2，将来自 CPU 413-2DP 的 QB8~QB39 的数据解包后存放到 DB2.DBB0~DBB31。

对于数据通信来说，CP 443-5 Ext 是“透明”的，在编程时并不需要考虑它的存在。因此本节的项目 PB_MS_5 与前一节的项目 PB_MS_4 的程序、变量表、实验方法和实验结果基本上相同。

3.5.3　CP 342-5 作 DP 主站

1．硬件组态

在下面的例子中，CP 342-5 是 DP 网络中的主站，从站是 ET 200B-16 DI、ET 200B-16DO 和 ET 200M。

在 STEP 7 中新建一个项目（见随书光盘中的例程 PB_MS_6），CPU 模块为 CPU 315-2 DP。在 HW Config 中，将电源模块和输入/输出模块插入机架。

将 CP 342-5 插入机架时，在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，设置它的 DP 站地址为 3。点击“新建”按钮，生成一条新的 PROFIBUS 子网络，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，采用默认的传输速率（1.5M bit/s）和配置文件（DP），点击“确定”按钮返回 CP 属性对话框，将 CP 连接到

DP 网络上。点击“确定”按钮返回 HW Config，双击 CP 342-5，打开它的属性对话框，在“常规”选项卡中设置它的 MPI 地址为 3。在“工作模式”选项卡，将它设置为 DP 主站。

在“地址”选项卡，可以看到默认的输入、输出的字节数为 16B，起始字节地址均为 320。

返回 HW Config 后，可以看到新创建的 PROFIBUS DP 网络，默认的 DP 主站系统的编号为 180（见图 3-67）。

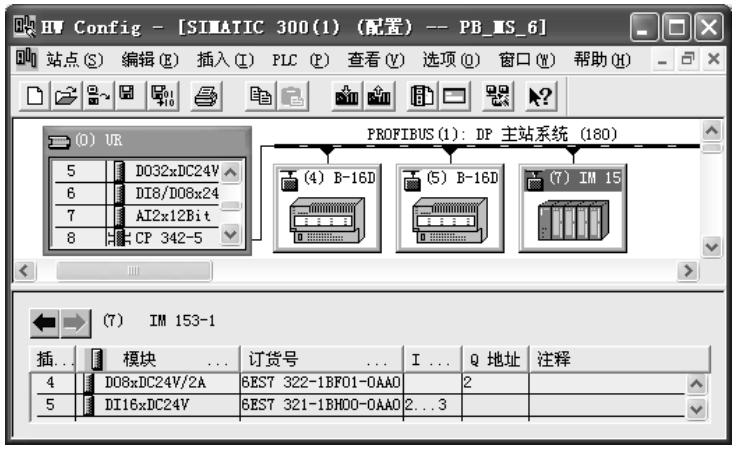


图 3-67　CP 342-5 作主站的 DP 网络

将右边硬件目录窗口的“\PROFIBUS DP\ET 200B”文件夹中的“B-16DO”和“B-16DI”拖放到 DP 网络上，在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，设置从站的地址为 4 和 5，自动分配的 4 号从站的输出地址和 5 号从站的输入地址均为 0 号和 1 号字节。

将“\PROFIBUS DP\ET 200M”文件夹中的“IM 153-1”拖放到 DP 网络上，在自动打开的“属性 –PROFIBUS 接口”对话框的“参数”选项卡中，设置从站的地址为 7。

选中该从站后，在下面的“插槽”插入一块 8 点 DO 模块和一块 16 点 DI 模块。自动分配的 I、Q 地址如图 3-67 所示。

CPU 集成的 DP 接口和 CP 443-5 作 DP 主站时，各非智能从站和中央机架的 I/O 地址是统一分配的。CP 342-5 作 DP 主站时，它的 DP 主站系统中的 I/O 地址区是虚拟的地址映像区，独立于 CPU 集成的 DP 接口和 CP 443-5 为主站的 DP 主站系统中的 I/O 地址区。虽然中央机架和作为 CP 342-5 的从站的 ET 200 的 DI/DO 均使用了 0～3 号输入字节，它们也不会冲突。

由于 CP 342-5 是通过调用 FC 1 和 FC 2 访问 DP 从站，而不是直接访问 I/Q 区，所以 ET 200M 不能使用功能模块，例如 FM 350-1 和 FM 352 等。

2．通信的编程

与 CPU 集成的 DP 接口不同，CP 342-5 作主站时，不能通过 I、Q 区直接读写 ET 200 的 I/O，需要在 OB1 中调用 CP 通信功能 FC 1“DP_SEND”和 FC 2“DP_RECV”，建立虚拟的通信接口区来访问从站。

CP 342-5 有一个内部的输入缓冲区和输出缓冲区，用来存放所有 DP 从站的 I/O 数据，较新版本的 CP 342-5 模块内部的输入、输出缓冲区分别为 2160B。输出缓冲区的数据周期性地写到从站的输出通道上，周期性读取的从站输入通道的数值存放在输入缓冲区，整个过程是

CP 342-5 与 PROFIBUS 从站之间自动协调完成的，不需编写程序。但是需要在 PLC 的用户程序中调用 FC 1 和 FC 2，来读写 CP 342-5 内部的缓冲区。

CPU 调用 FC 1（DP_SEND），将参数 SEND 指定的发送数据区的数据传送到 CP 342-5 的输出缓冲区（见图 3–68），以便将数据发送到 DP 从站。

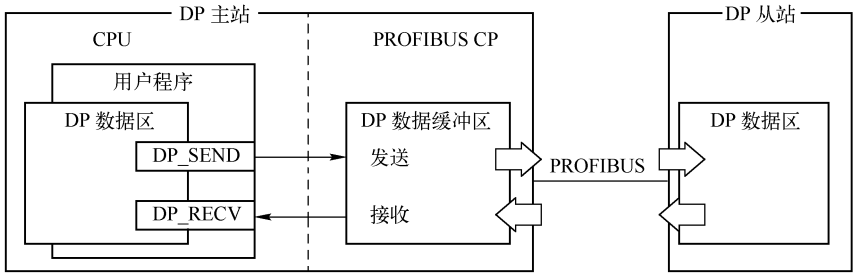


图 3–68　数据交换原理

CPU 调用 FC 2（DP_RECV），将 CP 342-5 的输入缓冲区中接收的 DP 状态信息和来自分布式 I/O 的过程数据，存入参数 RECV 指定的 CPU 中的接收数据区。

参数 SEND 和 RECV 指定的 DP 数据区可以是过程映像区（I/Q）、存储器区（M）或数据块（DB）区。输出参数 DONE 为 1、ERROR 和 STATUS 为 0 时，可以确认数据被正确地传送到了通信伙伴。

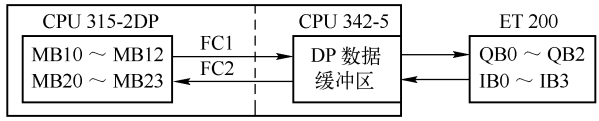


图 3–69　主站与从站之间的数据交换

在例程 PB_MS_6 的 CPU 315-2DP 的 OB1 中，调用 FC 1 将 MB10～MB12 打包后发送给 ET 200 的 QB0～QB2（见图 3–69）。调用 FC 2 将来自 ET 200 的 IB0～IB3 的数据存放到 MB20～MB23。下面是 OB1 中的程序：

```
程序段 1：数据打包后，通过 CP 发送到远程 DO
        CALL   "DP_SEND"                    //调用 FC 1
          CPLADDR  :=W#16#140               //CP 342-5 的起始地址 320
          SEND     :=P#M 10.0 BYTE 3        //S7-300 存放待发送数据的地址区
          DONE     :=M0.0                   //发送完成产生一个脉冲
          ERROR    :=M0.1                   //错误标志位
          STATUS   :=MW2                    //通信状态字
程序段 2：读取 CP 接收的来自远程 DI 的数据
        CALL   "DP_RECV"                    //调用 FC 2
          CPLADDR  :=W#16#140               //CP 342-5 的起始地址 320
          RECV     :=P#M 20.0 BYTE 4        //S7-300 存放接收到的数据的地址区
          NDR      :=M0.2                   //接收完成产生一个脉冲
          ERROR    :=M0.3                   //错误标志位
          STATUS   :=MW4                    //通信状态字
```

DPSTATUS :=MB6　　//DP 网络的状态字节

CP 342-5 的从站的输入/输出默认的起始地址为 0 号字节，一般采用默认的地址。如果起始地址非 0，例如输入点的地址设置为 QB4～QB7，FC 1 的参数 SEND 的字节数应为 8。

DP 主站模式的 DPSTATUS（见表 3-6）的第 1 位为 0 时，所有 DP 从站都处于数据传送状态。第 6 位为 1 时，接收的数据溢出，即 DP 从站接收数据的速度大于 DP 主站在 CPU 中用块调用获取数据的速度。读取的已接收数据总是 DP 从站接收的最后一个数据。

表 3-6　DPSTATUS 的意义

位	DP 主站模式	DP 从站模式
7	未用	未用
6	接收的数据溢出	未用
5	主站的 DP 状态：00 为 RUN，01 为 CLEAR，10 为 STOP，11 为 OFFLINE	未用
4		输入数据溢出
3	周期性同步被激活	DP 从站没有在监视时间内接收到来自 DP 主站的帧
2	诊断列表有效，至少一个站有新的诊断数据	1 类 DP 主站处于 CLEAR 状态
1	站列表有效	未完成组态/参数分配
0	DP 主站模式时为 0	DP 从站模式时为 1

DP 从站模式的 DPSTATUS 第 2 位为 1 时，1 类 DP 主站处于 CLEAR 状态，DP 从站接收到的 DP 输出数据为数值 0。

第 4 位为 1 时，DP 主站更新输入数据的速度大于 DP 从站在 CPU 中调用 FC 2 获取数据的速度，输入数据溢出。读取的输入数据总是从 DP 主站接收的最后一个输入数据。

3．通信的验证

用变量表监控对应于从站的 IW0 和 IW2 的 MW20 和 MW22，和对应于从站的 QW0 和 QB2 的 MW10 和 MB12。

将程序和组态数据下载到 CPU 后，用 PROFIBUS 电缆连接 CP 342-5、ET 200B 和 ET 200M 的 DP 接口。在系统运行时，除了将 CPU 的模式开关切换到 RUN 模式外，还应将 CP 342-5 的模式开关扳到 RUN 位置。打开变量表，点击工具栏上的按钮，使变量表进入监控状态。扳动接在从站输入模块外接的小开关，观察变量表中 MW20 或 MW22 的值是否变化（见图 3-70）。用变量表的“修改数值”列修改 MW10 和 MB12 的值，点击按钮，将修改值写入 PLC，数据被发送到从站对应的输出模块。观察 QW0 和 QB2 的输出点的状态是否符合修改值。

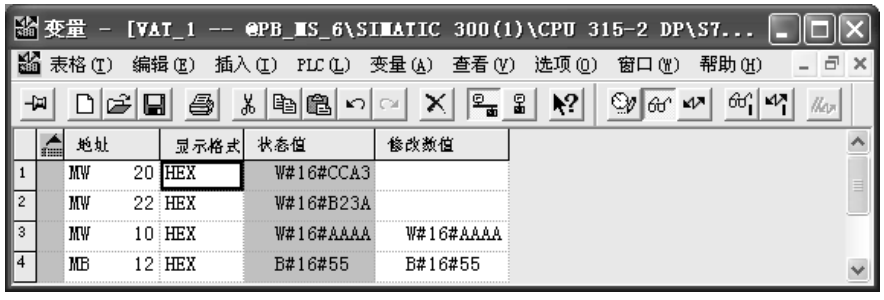


图 3-70　变量表

4．CP 342-5 同时作 DP 主站和 DP 从站

如果 CP 342-5 在通信中同时作 DP 主站和 DP 从站（见图 3-71），组态和编程的方法可以参考项目 PB_MS_4 和 PB_MS_6，通信双方的用户程序的 OB1 都需要调用 FC 1 和 FC 2。

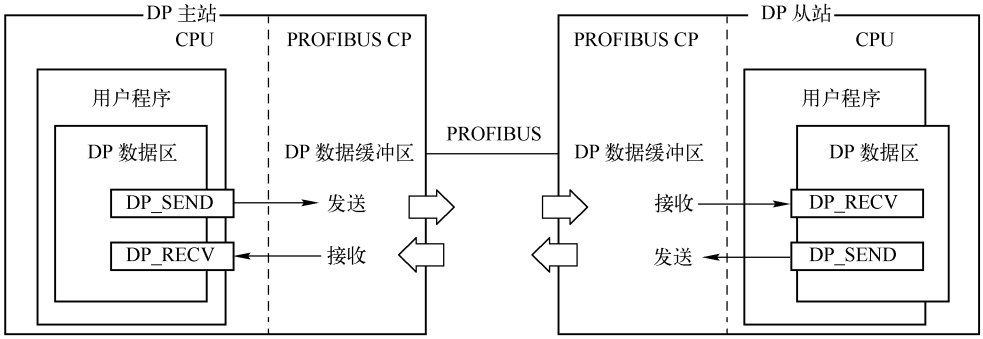


图 3-71　数据交换原理

3.5.4　使用 FC 4 控制 CP 342-5 为主站的 DP 网络

1．FC 4 DP_CTRL 的作业类型

FC 4“DP_CTRL”用于将控制作业传送给 S7-300 的 PROFIBUS CP。FC 4 在程序编辑器左边窗口的“\Libraries \SIMATIC_NET_CP\CP 300”文件夹中。

FC 4 的参数 CONTROL 用来指定 CONTROL 控制作业域的数据区地址，数据区的字节数为 1～240B，应不少于参数所需的字节数。CONTROL 的作业域第 1 个字节 CTYPE 的值确定了作业的类型（见表 3-7），从第 2 个字节开始是作业的参数。

DP_CTRL 只能在 DP 主站模式使用，正在运行 FC 4 时，不能给它提供新的作业数据。

CTYPE 为 0 和 1 时，参数的第 1 字节“命令模式”和第 2 字节“组选择”的具体结构见图 3-72 和图 3-73。命令 SYNC（同步）、UNSYNC（解除同步）、FREEZE（冻结）和 UNFREEZE（解除冻结）的意义、组态和编程的方法见 5.2.2 节。

表 3-7　DP_CTRL 的作业类型

CTYPE	对应的作业	作业域的参数
0	触发全局控制	第 1 字节为命令模式，第 2 字节为组选择
1	触发周期性全局控制	第 1～3 字节分别为命令模式、组选择和自动清除
3	删除较早的 DP 从站的诊断数据	第 1 字节为 1～126 时，为从站地址，为 127 时，为所有从站
4	设置当前 DP 模式	第 1 字节为 0～5 时，分别为 RUN、CLEAR、STOP、OFFLINE、带 AUTOCLEAR 的 RUN 和不带 AUTOCLEAR 的 RUN
5	设置当 CPU 变为 STOP 模式时，CP 切换到哪个 DP 模式，默认值为 CLEAR	第 1 字节为 0～3 时，CP 分别为 RUN、CLEAR、STOP 和 OFFLINE 模式
6	设置当 CP 变为 STOP 模式时，CP 切换到哪个 DP 模式	第 1 字节为 2 和 3 时，CP 分别为 STOP 和 OFFLINE
7	2 类 DP 主站周期性读取输入数据	第 1 字节为从站地址 1～125
8	2 类 DP 主站周期性读取输出数据	第 1 字节为从站地址 1～125
9	由 1、2 类 DP 主站终止循环处理 DP 从站	第 1 字节为从站地址 1～125
10	作为 1 类 DP 主站启动周期性的数据传输	第 1 字节为从站地址 1～125

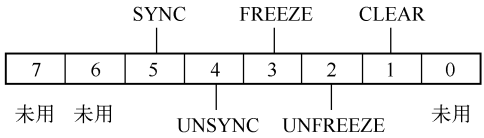


图 3-72　命令模式字节

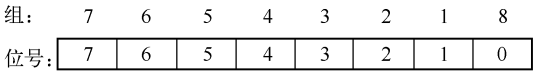


图 3-73　组选择字节

CTYPE 为 0 时，将单个全局控制作业发送到用“组选择”字节指定的 DP 从站。可以在命令模式参数中指定多个作业。

CTYPE 为 1 时，CP 将周期性全局控制作业发送到“组选择”字节指定的 DP 从站。如果在选择的组中至少有一个 DP 从站没有正常传输数据，并且第 3 个字节（自动清除）设置为 1，则将激活 CLEAR 模式，DP 从站的输出数据将被设置为 0。

可以在命令模式参数中激活下列全局作业：SYNC、FREEZE 和 CLEAR（CLEAR 位= 1），或执行取消激活的全局作业：UNSYNC、UNFREEZE 和 UNCLEAR（CLEAR 位= 0）。可以在命令模式参数中指定多个作业。

只能由另一个全局控制作业（周期性或非周期性）终止激活的周期性控制作业。要终止在命令模式中设置的作业，必须取消该作业。例如用 UNSYNC 作业取消 SYNC 作业。

2．系统的硬件结构

用 STEP 7 的新建项目向导创建一个新的项目（见随书光盘中的例程 FC4_CTRL），其硬件结构和组态方法与项目 PB_MS_6 相同，CPU 模块为 CPU 315-2 DP。CP 342-5 在 8 号槽，模块的起始地址为 320（16#140），其 DP 地址为 3，工作模式为 DP 主站。

在 HW Config 中为 CP 342-5 生成一个 DP 主站系统，默认的 DP 主站系统的编号为 180。为 CP 342-5 组态 3 个从站，ET 200B-16DO 和 ET 200B-16DI 分别是 4 号和 5 号从站。ET 200M 是 7 号从站，选中该从站后，给它分配一块 8 点 DO 模块、一块 16 点 DI 模块和一块 2AO 模块。3 个从站共有 4B 输入和 7B 输出。

3．程序设计

在 OB1 中，调用 FC 2 来读取从站的输入模块的数据和 DP 网络的状态字节，用 FC 1 将数据发送到从站的输出模块。用 FC 4 控制 DP 网络的状态。下面是 OB1 中的程序：

```
程序段 1：接收来自远程输入模块的数据
      CALL   "DP_RECV"                    //调用 FC 2
        CPLADDR :=W#16#140                //CP 342-5 的起始地址 320
        RECV    :=P#M 20.0 BYTE 4         //S7-300 的接收数据区
        NDR     :=M0.0                    //接收完成产生一个脉冲
        ERROR   :=M0.1                    //错误标志位
        STATUS  :=MW2                     //通信状态字
        DPSTATUS:=MB4                     //DP 网络的状态字节
程序段 2：发送数据到远程输出模块
      CALL   "DP_SEND"                    //调用 FC 1
        CPLADDR :=W#16#140                //CP 342-5 的起始地址 320
        SEND    :=P#M 24.0 BYTE 7         //S7-300 的发送数据区
        DONE    :=M0.2                    //发送完成产生一个脉冲
        ERROR   :=M0.3                    //错误标志位
```

```
  STATUS    :=MW6                    //通信状态字
```

程序段 3：将控制作业传送给 CP 342-5

```
  CALL   "DP_CTRL"                   //调用 FC 4
    CPLADDR :=W#16#140               //CP 342-5 的起始地址 320
    CONTROL := P#M 10.0 BYTE 4       //控制作业域的地址缓冲区
    DONE    :=M0.4                   //正确执行作业时为 1
    ERROR   :=M0.5                   //错误标志位
    STATUS  :=MW8                    //通信状态字
```

4．从站正常运行时的 DP 状态

系统刚开始运行时，DP 网络处于 RUN 状态，DP 主站已经启动 DP 从站的 CPU 用户数据的周期性处理，各从站正常运行。

MW22 是 7 号从站的输入字，MW24 是 4 号从站的输出字，MB26 是 7 号从站的输出字节。用变量表（见图 3-74）将常数写入 MW24 和 MB26，FC 1 将它发送到从站的输出模块，模块上 LED 的状态随之而变。扳动从站输入模块外接的小开关，通过变量表中的 MW22 可以看到用 FC 2 读取的输入值的变化。

MB4 是 FC 2 中的 DP 状态字节 DPSTATUS，它的第 4、5 位的意义如下（见表 3-6）：2#00 为 RUN，2#01 为 CLEAR，2#10 为 STOP，2#11 为 OFFLINE。RUN 状态时 DPSTATUS 为 16#04。

5．控制 DP 进入 CLEAR 状态

在 CLEAR（清除）状态，DP 主站已经启动 DP 从站的周期性处理，其 DP 从站的输出数据被设置为 0。

变量表中的 MB10 是 FC 4 的参数区的第一个字节 CTYPE，它决定命令的类型，MB11 是第一个参数字节。用变量表将 4 写入 MB10（设置当前的 DP 模式）。将 1（对应的 DP 模式为 CLEAR）写入第 1 个参数字节 MB11，DP 切换到 CLEAR 模式（见表 3-7），FC 2 读取的输入字 MW22 变为 0（见图 3-75），从站输出模块的输出点的 LED 全部熄灭，CP 和从站的错误 LED 未亮。DPSTATUS（MB4）为 16#10，第 4、5 位为 2#01（CLEAR 状态）。此时变量表中的 MW24 和 MB26 不能正确反映从站的输出点的状态。

VAT_1 -- @FC4_CTRL\S...

	地址		显示格式	状态值	修改数值
1	MW	22	HEX	W#16#6D61	
2	MW	24	HEX	W#16#5555	
3	MB	26	HEX	B#16#55	
4	MB	4	HEX	B#16#04	
5	MB	10	DEC	4	4
6	MB	11	DEC	0	0

图 3-74　RUN 模式的变量表

VAT_1 -- @FC4_CTRL\S...

	地址		显示格式	状态值	修改数值
1	MW	22	HEX	W#16#0000	
2	MW	24	HEX	W#16#5555	
3	MB	26	HEX	B#16#55	
4	MB	4	HEX	B#16#10	
5	MB	10	DEC	4	4
6	MB	11	DEC	1	1

图 3-75　CLEAR 模式的变量表

将 4 写入 MB10，0 写入 MB11，DP 返回 RUN 模式，输入、输出恢复正常，CP 和从站的错误 LED 熄灭。

6．控制 DP 进入 STOP 状态

将 4 写入 MB10，2 写入 MB11，DP 网络进入 STOP 模式（见表 3-7）。FC 2 读取的输入

字 MW22 变为 0，从站的输出模块的输出点的 LED 全部熄灭。CP 的 SF LED 闪烁，3 个从站的 BF（总线故障）LED 亮。MB4（DPSTATUS）为 16#22，第 4、5 位为 2#10（STOP 状态）。

7．控制 DP 进入 OFFLINE 状态

在 OFFLINE（离线）状态，DP 主站已经停止对 DP 从站的周期性处理（轮询），分配给 DP 主站的从站被删除。

将 4 写入 MB10，3 写入 MB11，DP 切换到 OFFLINE 状态，输入、输出模块的状态与 STOP 状态时的相同，CP 的 SF 和 IM 153-1 的 BF LED 闪烁，4 号、5 号从站的 BF LED 亮。MB4 为 16#32，第 4、5 位为 2#11（OFFLINE 状态）。

8．带自动清除（AUTOCLEAR）功能的 RUN 状态

AUTOCLEAR 的意义如下：如果需要交换数据的 DP 从站至少有一个未正常传送数据，1 类 DP 主站自动进入 CLEAR 状态。

将 4 写入 MB10，4 写入 MB11，DP 进入带 AUTOCLEAR 的 RUN 状态，从站的输入、输出模块和 LED 的状态与 RUN 状态时的相同。DPSTATUS（MB4）的值为 16#04（RUN 状态）。

断开 4 号从站的电源，其他从站的输入/输出变为 0 状态，CP 的 SF LED 闪动。接通 4 号从站的电源，各从站的输入/输出恢复正常。

9．不带自动清除功能的 RUN 状态

将 4 写入 MB10，5 写入 MB11，DP 进入不带 AUTOCLEAR 的 RUN 状态（见表 3–7），从站的输入、输出模块和 LED 的状态与 RUN 状态时相同。断开某个从站的电源，其他从站的输入/输出不会变为 0 状态，各 LED 的状态不变。DPSTATUS（MB4）的值为 16#04（RUN 状态）。

3.6 练习题

1．标准从站和智能从站各有什么特点？

2．组态一个项目，DP 主站为 CPU 315-2DP，DP 从站为 ET 200M 和 ET 200S。在主机架和从站的机架中分别插入适量的 DI、DO、AI、AO 模块，总结信号模块地址分配的规律。

3．怎样用 DIP 开关（见图 3–11）将 ET 200M 的接口模块 IM 153 的 DP 站地址设置为 14？

4．组态一个项目，CPU 315-2DP 为 DP 主站，CPU 313C-2DP 为智能从站，双方分别用 IB60 和 QB60 开始的输入、输出地址区双向传送 10B 的数据，一致性为“单位”。

5．要求同上题，区别在于数据的一致性为“全部”，编写调用 SFC 14 和 SFC 15 的程序。

6．在编程和从站 I/O 地址分配方面，CP 443-5 和 CP 342-5 有什么区别？

7．组态一个项目，CP 342-5 作主站，ET 200M 和 ET 200S 作从站，编写通信程序。

第 4 章　基于 PROFIBUS 的 S7 通信与 FDL 通信

4.1　S7 通信

4.1.1　S7 通信概述

1．连接的基本概念

数据通信协议可以分为面向连接的协议和无连接的协议，前者在进行数据交换之前，必须与通信伙伴建立连接，后者用于发送单个消息。

这两种协议在安全性方面有较大的区别，它们具有不同的传输效率。面向连接的协议具有较高的安全性，与无连接协议相比，在上层计算机中需要进行更多的处理。

连接是指两个通信伙伴之间为了执行通信服务建立的逻辑链路，而不是指两个站之间用物理媒体（例如电缆）实现的连接。连接相当于通信伙伴之间一条虚拟的“专线”，它们随时可以用这条“专线”进行通信。一条物理线路可以建立多个连接。

如果交换的信息非常重要，或者需要保证传输数据的完整性，应使用面向连接的协议。为了确保正确地建立连接，连接的一方必须是主动的，另一方是被动的，否则无法建立连接。

无连接协议传输的信息单元相当于电报报文，它们通常是一种独立完整的消息，有时也被称为数据报文。

2．动态连接和静态连接

连接分为不需要组态的动态连接和需要组态的静态连接。

（1）不需要组态的动态连接

PG（编程器）通信和 S7 基本通信不需要对连接组态，这种连接也称为动态连接，S7 基本通信将在 14.3 节介绍。

（2）需要组态的静态连接

S7 连接属于需要组态的连接，这类连接用 STEP 7 集成的网络组态工具 NetPro 组态。

组态的连接一经建立会一直保持，可以实现快速的通信，称为静态连接。与同一个通信伙伴可以建立多个连接，CPU 和 CP 同时可以使用的连接的数量受到与其型号有关的连接资源的限制。

在组态连接时，应指定通信伙伴和连接类型，以及连接的特殊属性。组态时将为每个连接自动分配一个唯一的“本地标识符”（本地 ID）。在调用通信块时，需要使用本地标识符。

3．客户机与服务器

基于连接的通信分为单向通信和双向通信。在双向通信中，通信双方都需要调用通信块，一方调用发送块来发送数据，另一方调用接收块来接收数据。

与双向通信不同，单向通信只需要通信的一方编写通信程序。编写通信程序一方的 CPU 为客户机（Client），不需编写通信程序一方的 CPU 为服务器（Server）。客户机是向服务器

请求服务的设备，它是主动的，需要调用通信块对服务器的数据进行读、写操作。服务器是提供特定服务的设备，通信服务经客户机要求启动。服务器是通信中的被动方，通信功能由它的操作系统执行。

4．S7 通信

S7 通信是专为 SIMATIC S7 和 C7 优化设计的通信协议，提供简明、强有力的通信服务。所有 S7 和 C7 PLC 都集成了 S7 通信服务，通过 S7 服务，用户程序可以读取或改写通信伙伴的数据。S7-300/400 PLC 广泛地使用 S7 通信，它主要用于 S7-300/400 CPU 之间的主-主通信，CPU 与功能模块（FM）之间、CPU 与西门子人机界面 TP/OP（触摸屏/操作员面板）和组态软件 WinCC 之间的通信。

S7 通信可以用于工业以太网、PROFIBUS 或 MPI 网络。这些网络的 S7 通信的组态和编程方法基本上相同。

S7 系统的设备可以实现下列 S7 功能（与设备的型号有关，有的只能实现部分功能）：

1）编程、测试、调试和诊断 S7-300/400 PLC 的全部 STEP 7 在线功能。

2）自动地与 HMI（人机界面）交换数据。

3）S7 站点之间的数据传输。

4）读、写别的 S7 站点的数据，通信伙伴不需要编写通信程序。

5）控制通信伙伴 CPU 的停止和起动。

6）监视通信伙伴 CPU 的运行状态。

5．用于数据交换的 S7 通信的 SFB/FB

用于数据交换的 S7 通信的 SFB/FB 见表 4-1。在 S7 单向连接中，客户机调用单向通信功能块 GET 和 PUT，读、写服务器的存储区。S7-400 可以调用 SFB 8/SFB 9 和 SFB 12/ SFB 13，进行双向通信。

表 4-1　用于 S7 通信数据交换的 SFB/FB

编　　号		助记符	可传输字节数		描　　述
S7-400	S7-300		S7-400	S7-300	
SFB 8	FB 8	U_SEND	440B	160B	与接收方通信功能（U_RCV）执行序列无关的快速的无需确认的数据交换，例如传送操作与维护消息，对方接收到的数据可能被新的数据覆盖
SFB 9	FB 9	U_RCV			
SFB 12	FB 12	B_SEND	64KB	32KB	将数据块安全地传输到通信伙伴，直到通信伙伴的接收功能（B_RCV）接收完数据，数据传输才结束
SFB 13	FB 13	B_RCV			
SFB 14	FB 14	GET	400B	160B	程序控制读取远方 CPU 的变量，通信伙伴不需要编写通信程序
SFB 15	FB 15	PUT			程序控制写变量到远方 CPU，通信伙伴不需要编写通信程序
SFB 16	—	PRINT			发送数据和指令格式到远方打印机（仅用于 S7-400）

4.1.2　CPU 与 CP 的 S7 通信功能

下列硬件支持 PROFIBUS 网络的 S7 通信：集成有 DP 接口的 S7-300/400 和 C7 的 CPU、通信处理器 CP 342-5、CP 343-5、CP 443-5 和 PC 的 PROFIBUS 通信卡。

表 4-2 是各种通信接口之间允许的 S7 通信。有 S7-300 集成的通信接口参与时，只能进行单向 S7 通信，S7-300 集成的通信接口在通信中只能作服务器。

表 4-2　不同的 DP 接口之间允许的 S7 通信功能

	S7-300 集成 DP 接口	S7-400 集成 DP 接口	CP 342-5 的 DP 接口	CP 443-5 Ext 的 DP 接口
S7-300 集成 DP 接口	不支持	单向通信	单向通信	单向通信
S7-400 集成 DP 接口		单向/双向通信	单向/双向通信	单向/双向通信
CP 342-5 的 DP 接口			单向/双向通信	单向/双向通信
CP 443-5 Ext的DP接口				单向/双向通信

S7-400 集成的 DP 接口和 CP 443-5 在单向 S7 通信中既可以作服务器，也可以作客户机。它们之间还可以进行双向 S7 通信。

S7-300 的 PROFIBUS CP 是否能参与 S7 双向通信和作 S7 单向通信的客户机，与 CP 和 CPU 的订货号和固件版本号均有关系，只有少数较高档的 S7-300 CPU 和 PROFIBUS CP 才有双向 S7 通信功能。具体的情况可以查阅模块手册，组态时选中 HW Config 右边的硬件目录窗口中的某个模块，在硬件目录下面灰色背景的小窗口中可以看到该模块简要的特性。

NetPro 有很强的防止出错的功能，它会禁止建立那些选用的硬件不支持的通信连接组态。

如果选用的 S7-300 的硬件不支持双向 S7 通信，在组态连接时将自动生成不可更改的单向 S7 连接。

4.2　基于 PROFIBUS 的单向 S7 通信

4.2.1　CPU 集成的 DP 接口的 S7 单向通信

S7-300 的集成 DP 接口在 S7 连接中只能作单向通信的服务器。在下面的例程中，S7-300 和 S7-400 分别作为服务器和客户机。S7-400 CPU 调用 SFB 14（GET）和 SFB 15（PUT），读写 S7-300 的数据。

1．硬件组态

在 STEP 7 中创建一个项目 PB_S7_A（见随书光盘的文件夹“\Project\PB_S7”中的同名例程）。在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。插入 CPU 模块时，在自动打开的 DP 接口属性对话框中，点击“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，设置传输速率为默认的 1.5 Mbit/s，配置文件为“标准”。传输速率和配置文件将用于整个 PROFIBUS 子网络。

返回 DP 接口属性对话框，可以看到生成的名为“PROFIBUS（1）”的网络。CPU 集成的 DP 接口和 MPI 接口默认的地址均为 2，默认的工作模式为 DP 主站。

在 SIMATIC 管理器中生成一个 S7-300 站。在 HW Config 中，将 CPU 313C-2DP 插入机架，在自动打开的“属性 - PROFIBUS 接口”对话框的“参数”选项卡中，设置站地址为 3，选中“子网”列表中的“PROFIBUS（1）”（见图 4-1），将 CPU 313C-2DP 连接到 DP 网络上，默认的工作方式为 DP 主站。在 CPU 属性对话框的“常规”选项卡中，设置 MPI 地址为 3。将电源模块和信号模块插入机架。组态好硬件后，点击工具栏上的  按钮，编译并保存组态信息。

图 4-2 是该项目组态和编程结束后的 SIMATIC 管理器。

图 4-1　DP 接口属性对话框

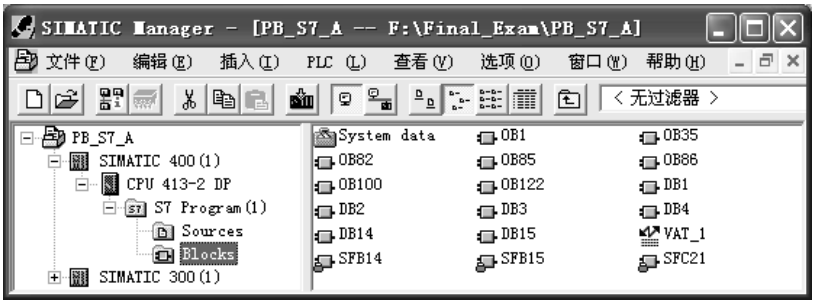


图 4-2　SIMATIC 管理器

2．S7 连接的组态

点击 SIMATIC 管理器工具栏上的按钮，打开网络组态工具 NetPro，可以看到两个站已经连接到 DP 网络上。选中 CPU 413-2 DP 所在的小方框，在 NetPro 下面的窗口出现连接表（见图 4-3）。双击连接表的第 1 行，在出现的“插入新连接”对话框中，系统默认的通信伙伴为同一项目中的 CPU 313-2 DP，默认的连接类型为 S7 连接。

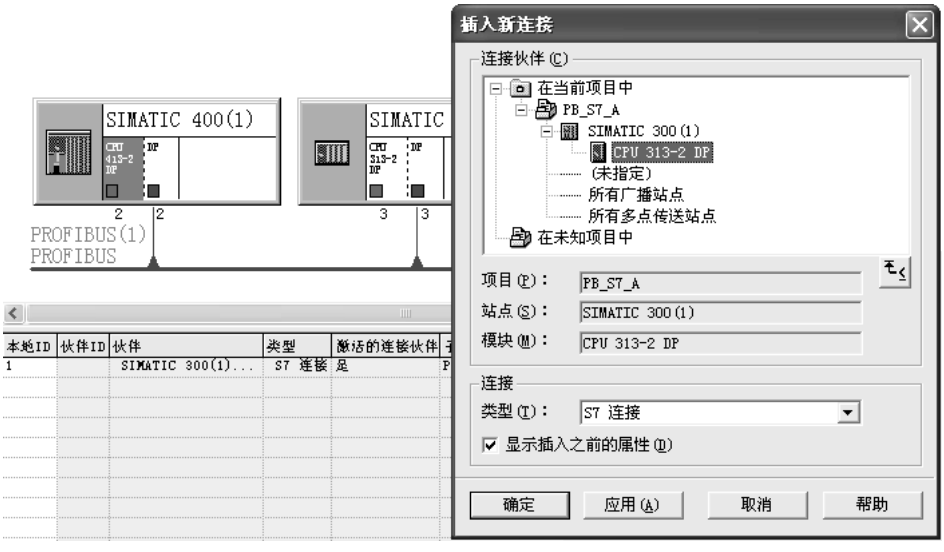


图 4-3　网络与连接的组态

点击“确定”按钮，确认默认的设置，出现“属性-S7 连接”对话框（见图 4-4），“本地

连接端点”区中的“单向”复选框被自动选中，并且不能更改，因此默认的连接方式为“单向”。在调用通信 SFB 时，将会使用“块参数”区的“本地 ID”（本地标识符）的值。

图 4–4　S7 连接属性对话框

S7-300 和 S7-400 集成的 DP 接口之间只能建立单向连接。因为是单向连接，连接表中没有通信伙伴的 ID（见图 4–3），选中 CPU 313C-2DP 所在的小方框，连接表中没有连接信息。

组态好连接后，点击工具栏上的按钮，编译并保存网络组态信息。

在单向 S7 连接中，仅需将网络组态信息下载到 S7 通信的客户机。

3．S7 通信编程

在单向 S7 连接中，CPU 413-2DP 和 CPU 313C-2DP 分别作为客户机和服务器，客户机调用单向通信功能块 GET 和 PUT，读、写服务器的存储区。服务器是通信中的被动方，不需要调用通信功能块。

S7-400 使用的 S7 通信系统功能块（SFB）在程序编辑器左边窗口的文件夹“\库\Standard Library\System Function Blocks”中，S7-300 使用的 S7 通信功能块（FB）在文件夹“\库\SIMATIC_NET_CP\CP 300 中”。SFB PUT/GET 可以读取和写入 4 个数据区，FB PUT/GET 只能读取和写入一个数据区。

GET、PUT 在通信请求信号 REQ 的上升沿时激活数据传输，这种通信方式属于事件驱动的通信方式。因为是上升沿时启动通信功能，可以将通信程序放置在主程序 OB1 或周期执行的 OB35 中。为了减少中断程序的执行时间，本书将 S7 通信的程序放在 OB1。

如果要实现周期性的数据传输，最简单的方法是用时钟存储器位提供的时钟脉冲作 REQ 信号。时钟存储器的组态方法如下：打开 HW Config，双击机架中 CPU 413-2DP 所在的行，在出现的 CPU 属性对话框的“周期/时钟存储器”选项卡中（见图 4–5），点击复选框“时钟存储器”，设置用于时钟存储器的存储器字节为 MB8。

点击“帮助”按钮，或按计算机的〈F1〉键，在出现的帮助信息中，点击绿色的“时钟存储器”。再点击时钟存储器帮助信息中的“周期性”，可以看到时钟存储器字节每一位的周期或频率。MB8 的第 1 位 M8.1 的周期为 200ms（ON 100ms，OFF 100ms）。第 0 位 M8.0 的

周期为 100ms（ON 50ms，OFF 50ms）。

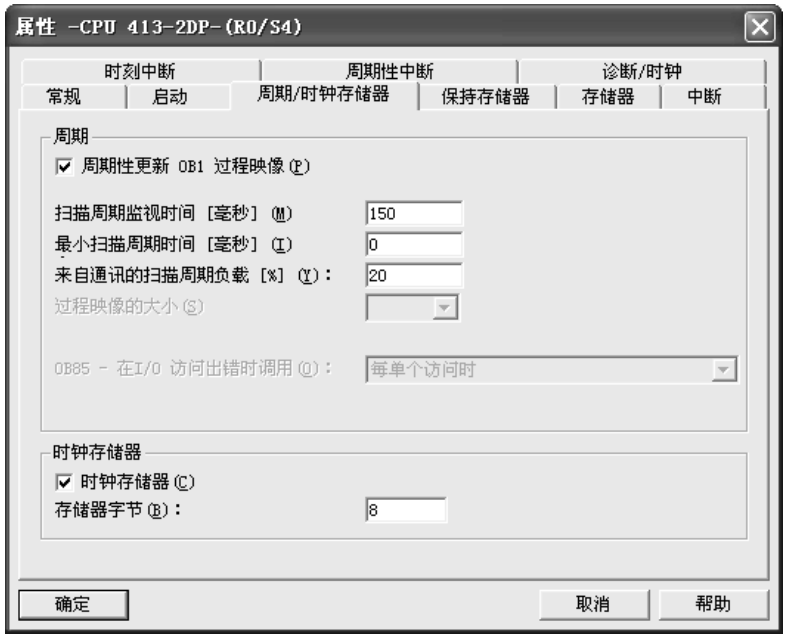


图 4-5　设置时钟存储器

CPU 413-2DP 的 OB1 的程序段 1 中的两条语句，使 M10.0 和 M8.1 的相位相反。它们分别作系统功能块 GET 和 PUT 的 REQ（通信请求）信号，它们的上升沿互差 100ms。下面是 OB1 中的程序：

```
程序段 1：将时钟脉冲信号反相
        AN    M      8.1
        =     M      10.0
程序段 2：  读取通信伙伴的数据
        CALL  "GET" , DB14                    //调用 SFB 14
          REQ     :=M8.1                      //上升沿时激活数据传输，每 200ms 读取一次
          ID      :=W#16#1                    //S7 连接号
          NDR     :=M0.1                      //每次读取完成产生一个脉冲
          ERROR   :=M0.2                      //错误标志，出错时为 1
          STATUS  :=MW2                       //状态字，为 0 时表示没有警告和错误
          ADDR_1 :=P#DB1.DBX0.0 BYTE 20       //要读取的通信伙伴的 1 号地址区
          ADDR_2 :=P#DB3.DBX0.0 BYTE 20       //要读取的通信伙伴的 2 号地址区
          ADDR_3 :=ID0                        //读取通信伙伴的 ID0
          ADDR_4 :=P#M 40.0 BYTE 20           //要读取的通信伙伴的 4 号地址区
          RD_1    :=P#DB2.DBX0.0 BYTE 20      //本站存放读取的数据的 1 号地址区
          RD_2    :=P#DB4.DBX0.0 BYTE 20      //本站存放读取的数据的 2 号地址区
          RD_3    := QD0                      //用通信伙伴的 ID0 控制本站的 QD0
          RD_4    :=P#M 20.0 BYTE 20          //本站存放读取的数据的 4 号地址区
程序段 3：向通信伙伴的数据区写入数据
        CALL  "PUT" , DB15                    //调用 SFB 15
```

```
REQ      :=M10.0                    //上升沿时激活数据交换，每 200ms 写一次
ID       :=W#16#1                   //S7 连接号
DONE     :=M10.1                    //每次写完成产生一个脉冲
ERROR    :=M10.2                    //错误标志，出错时为 1
STATUS   :=MW12                     //状态字，为 0 时表示没有警告和错误
ADDR_1   :=P#DB2.DBX0.0 BYTE 20     //要写入数据的通信伙伴的 1 号地址区
ADDR_2   :=P#DB4.DBX0.0 BYTE 20     //要写入数据的通信伙伴的 2 号地址区
ADDR_3   := QD4                     //将本站的 ID0 写入通信伙伴的 QD4
ADDR_4   :=P#M 20.0 BYTE 20         //要写入数据的通信伙伴的 4 号地址区
SD_1     :=P#DB1.DBX0.0 BYTE 20     //存放本站要发送的数据的 1 号地址区
SD_2     :=P#DB3.DBX0.0 BYTE 20     //存放本站要发送的数据的 2 号地址区
SD_3     :=ID0                      //用本站的 ID0 控制通信伙伴的 QD0
SD_4     :=P#M 40.0 BYTE 20         //存放本站要发送的数据的 4 号地址区
```

在调用 SFB PUT 和 GET 时，允许只使用 4 个地址区中的部分地址区。

ERROR 和 STATUS 均为 0 时，没有警告和错误；ERROR 为 0、STATUS 非 0 时，有警告信息；ERROR 为 1 时通信出错。SFB 的在线帮助给出了 STATUS 的警告或错误代码的意义。

下面是 CPU 413-2DP 的 OB35 中的程序：

程序段 1：

```
L     DB1.DBW    0
+     1
T     DB1.DBW    0          //每 100ms 将 DB1.DBW0 加 1
L     DB3.DBW    0
+     2
T     DB3.DBW    0          //每 100ms 将 DB3.DBW0 加 2
```

4．初始化程序

在 CPU 413-2DP 的初始化程序 OB100 中，调用 SFC 21，将 DB 1、DB 3 和 M 区中的数据发送区的各个字分别预置为 16#1111、16#3333 和 16#4444，将 DB 2、DB 4 和 M 区中的数据接收区的各个字清零。程序中的 LW20 和 LW22 是 OB100 的局部变量字。

下面是 CPU 413-2DP 的 OB100 中的程序：

程序段 1：初始化存放要发送的数据的地址区

```
L     W#16#1111
T     LW     20
CALL  "FILL"                    //调用 SFC 21
  BVAL    :=LW20                //源数据
  RET_VAL :=LW22                //故障代码
  BLK     :=P#DB1.DBX0.0 BYTE 20  //地址区
  …
```

程序段 2：将存放接收的数据的地址区清零

```
L     W#16#0
T     LW     20
CALL  "FILL"
```

```
BVAL     :=LW20                          //源数据
RET_VAL :=LW22                           //故障代码
BLK      :=P#DB2.DBX0.0 BYTE 20          //地址区
…
```

CPU 313C-2DP 和 CPU 413-2DP 的 OB100 中的程序基本上相同，其区别在于前者分别将 DB 1、DB 3 和 M 区中的数据发送区的数据初始化为 W#16#1010、W#16#3030 和 W#16#4040。CPU 313C-2DP 的 OB1 中没有通信程序，OB35 每 100ms 分别将 DB1.DBW0 和 DB3.DBW0 加 3 和加 4。

5．通信的监控

用 PROFIBUS 电缆将两块 CPU 和 CP 5613 的 MPI 接口连接到一起，将组态信息和程序分别下载到两台 PLC。运行时用电缆将两块 CPU 集成的 DP 接口连接到一起，可以用 MPI 网络或 PROFIBUS 网络对通信过程进行监控。

运行时通信双方的 OB35 使 DB1.DBW0 和 DB3.DBW0 的值不断增大，然后发送到对方的 DB2.DBW0 和 DB4.DBW0。在时钟脉冲 M8.1 的上升沿，每 200ms 读取一次 CPU 313C-2DP 的数据；在时钟脉冲 M10.0 的上升沿，每 200ms 将数据写入 CPU 313C-2DP 的数据区。

同时打开通信双方的变量表，将它们调节到适当的大小。点击工具栏上的  按钮，变量表进入监控状态，“状态值”列显示的是 PLC 中变量的值。图 4-6 和图 4-7 是在运行时复制的通信双方的变量表。在变量表中可以看到双方接收到的 DB2.DBW0 和 DB4.DBW0 的值在不断地增大，此外可以看到各数据接收区接收到的第一个字和最后一个字。

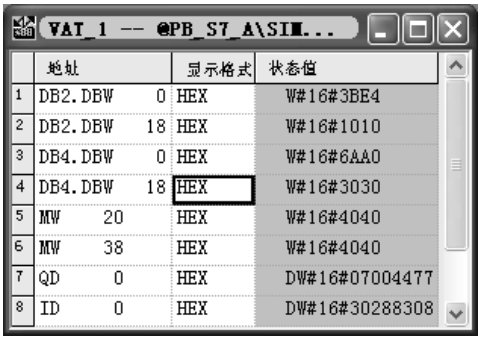


图 4-6　CPU 413-2DP 的变量表

图 4-7　CPU 313C-2DP 的变量表

通过 CPU 413-2DP 读、写 CPU 313C-2DP 中的数据，实现了用两个站的 ID0 分别控制对方的 QD0 或 QD4。在运行时用外接的小开关改变 ID0 的状态，可以看到通信伙伴对应的输出点的状态随之而变。

4.2.2　使用通信处理器的 S7 单向通信

1．CPU 413-2DP 与 CP 342-5 的单向 S7 通信

在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_S7_1）。在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。

插入 CPU 413-2DP 时，在自动打开的 DP 接口属性对话框中，点击“新建”按钮，生成 PROFIBUS-DP 网络，设置传输速率为默认的 1.5 Mbit/s，配置文件为“标准”，CPU 的工作模式为 DP 主站，MPI 地址和 DP 地址均为 2。

在 SIMATIC 管理器中插入一个 300 站点，在 HW Config 中，将电源模块、CPU 315-2DP、信号模块和 CP 342-5 插入机架。设置 CPU 的 MPI 地址和 DP 地址均为 3。

双击机架中的 CP 342-5，点击出现的 CP 属性对话框中的“属性”按钮，在出现的 CP 的 PROFIBUS 接口对话框中，将 CP 连接到 PROFIBUS 网络上，设置 CP 的工作模式为“无 DP”（见图 4–8），MPI 地址和 DP 地址均为 4。

图 4–8　组态 CP 342-5 的工作模式

组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

组态好两个站点的硬件后，点击工具栏上的按钮，打开 NetPro 窗口，CPU 413-2DP 和 CP 342-5 已连接到 PROFIBUS 网络上（见图 4–9）。

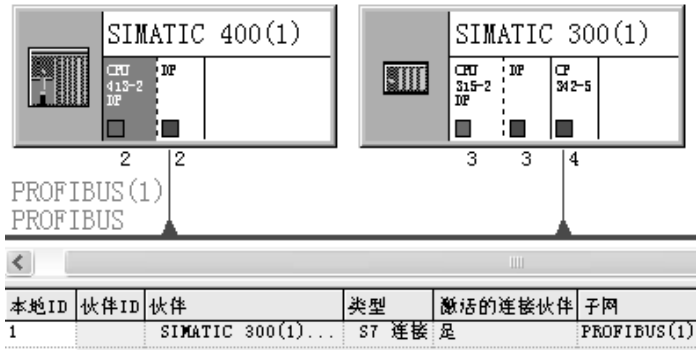


图 4–9　网络与连接的组态

选中 CPU 413-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。在出现的“插入新连接”对话框中，系统默认的通信伙伴为 CPU 315-2DP，默认的连接类型为 S7 连接。点击“确定”按钮，出现“属性–S7 连接”对话框（见图 4–10），通信伙伴的 DP 接口为 CP 342-5。该连接为单向连接，连接表中没有伙伴 ID。

本项目的程序与项目 PB_S7_A 的完全相同。在 CPU 413-2DP 的 OB1 中，调用 SFB GET 和 PUT，读取 CPU 315-2DP 地址区中的数据。SFB PUT 和 GET 中的通信请求信号 REQ 由周期为 200ms 的时钟存储器位 M8.0 和与它反相的 M10.0 提供。变量表、通信过程的监控方法与项目 PB_S7_A 的也完全相同。

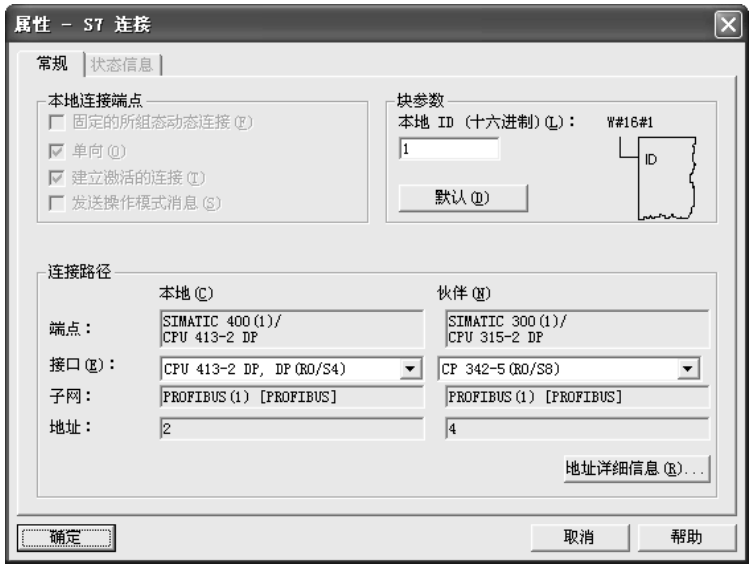


图 4-10　S7 连接属性对话框

2．CP 443-5 与 CP 342-5 的单向 S7 通信

在 SIMATIC 管理器中创建一个项目（见随书光盘中的例程 PB_S7_2），生成一个 400 站点。在 HW Config 中，将电源模块、CPU 413-2DP、信号模块和 CP 443-5 Ext 插入机架。设置 CPU 的 MPI 地址和 DP 地址均为 2。插入 CP 443-5 Ext 时，在自动打开的 DP 接口属性对话框中，点击“新建”按钮，生成 PROFIBUS-DP 网络，设置传输速率为默认的 1.5 Mbit/s，配置文件为“标准”，CP 的工作模式为“无 DP”，DP 地址为 3。

在 SIMATIC 管理器中生成一个 300 站点，在 HW Config 中，将电源模块、CPU 315-2DP、信号模块和 CP 342-5 插入机架。设置 CPU 的 MPI 地址为 3，DP 地址为 4。

双击机架中的 CP 342-5，点击出现的 CP 属性对话框中的“属性”按钮，在出现的 CP 的 PROFIBUS 接口对话框中，将 CP 连接到 PROFIBUS 网络上，设置 CP 的工作模式为“无 DP”，MPI 地址为 4，DP 地址为 5。

组态好硬件后，点击工具栏上的  按钮，编译并保存硬件组态信息。

组态好两个站后，点击工具栏上的  按钮，打开 NetPro 窗口，CP 443-5Ext 和 CP 342-5 已连接到 DP 网络上（见图 4-11）。选中“SIMATIC 400（1）”站点中 CPU 413-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。

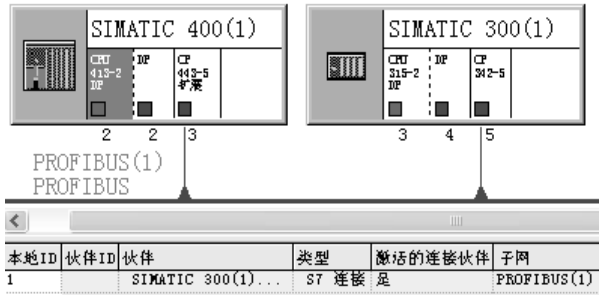


图 4-11　网络与连接的组态

在出现的“插入新连接”对话框中，系统默认的通信伙伴为 CPU 315-2DP，默认的连接类型为 S7 连接。点击“确定”按钮，出现“属性–S7 连接”对话框（见图 4–12），通信双方的 DP 接口均为通信处理器（CP）。该连接为单向连接，连接表中没有伙伴 ID。

图 4–12　S7 连接属性对话框

本项目的程序与项目 PB_S7_A 的完全相同。在 CPU 413-2DP 的 OB1 中，调用 SFB GET 和 PUT，读取 CPU 315-2DP 地址区中的数据。SFB PUT 和 GET 中的通信请求信号 REQ 由周期为 200ms 的时钟存储器位 M8.0 和与它反相的 M10.0 提供。变量表、通信过程的监控方法与项目 PB_S7_A 的也完全相同。

4.2.3　与连接有关的操作

1．自动设置块参数

为了正确地设置块调用参数，STEP 7 的程序编辑器为用户提供了程序块接收硬件组态和连接组态中的参数的功能。

在用户程序中为块分配参数值时，执行下列步骤：

1）在程序编辑器中调用通信用的块，例如 SFC 15“PUT”，暂时不要输入连接参数 ID。

2）用鼠标右键点击调用的块。

3）在出现的快捷菜单中执行“连接”命令。

4）在出现的“用于连接的块参数”对话框中（见图 4–13），选中“可用的连接”列表中的某个连接，在“块参数用于:”下面出现用于该通信块的块参数。

5）点击“确定”按钮，组态时生成的块参数 ID 的值 W#16#1 将自动出现在调用的块中。

点击图 4–13 中的“属性”按钮，将显示出连接属性对话框。点击“新建连接”按钮，在出现的“新建连接”对话框中，可以建立一个新的连接。点击“更改伙伴”按钮，在出现的

“改变连接伙伴”对话框中，可以改变连接中的通信伙伴。

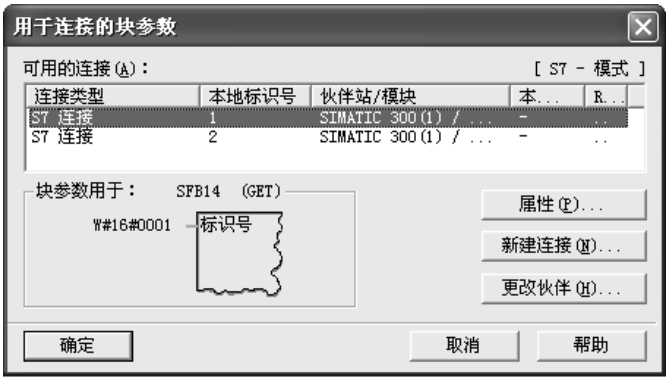


图 4-13　用于连接的块参数对话框

2．连接状态

执行 NetPro 的“PLC”菜单中的“激活连接状态”命令，“连接状态”功能被激活，连接表的最左边将会出现“连接状态”列（见图 4-14）。有 4 种可能的连接状态：已建立、未建立、正在建立、不可用。可以用菜单命令取消激活的连接状态。

连接状态	本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
▷ 已建立	1		SIMATIC 300(2)...	S7 连接	是	Ethernet(1) [IE]

图 4-14　连接状态

选中连接表中的某个连接，执行菜单命令“编辑”→“对象属性”，在出现的连接属性对话框的“状态信息”选项卡中，可以看到连接的状态信息。

4.3　基于 PROFIBUS 的双向 S7 通信

4.3.1　使用 USEND/URCV 的 S7 通信

使用 SFB/FB USEND/URCV，可以进行快速、不可靠的数据传送，例如，可以用于事件消息和报警消息的传送。

BSEND/BRCV 和 USEND/URCV 属于双向通信块，通信的双方都必须调用通信功能块。如果使用集成的 MPI 接口或集成的 DP 接口，它们只能用于两台 S7-400 之间的 S7 通信。

1．硬件组态

在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_S7_B 和图 4-15）。在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。插入 CPU 模块时，在自动打开的 DP 接口属性对话框中，点击“新建”按钮，生成 PROFIBUS-DP 网络。设置传输速率为默认的 1.5 Mbit/s，配置文件为“标准”。CPU 的 MPI 地址和 DP 地址均为 2。

在 SIMATIC 管理器中生成另一个 400 站点，在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。设置 CPU 的 MPI 地址和 DP 地址均为 3，将它连接到 DP 网络上。两个站的工作模式均为“DP 主站”。

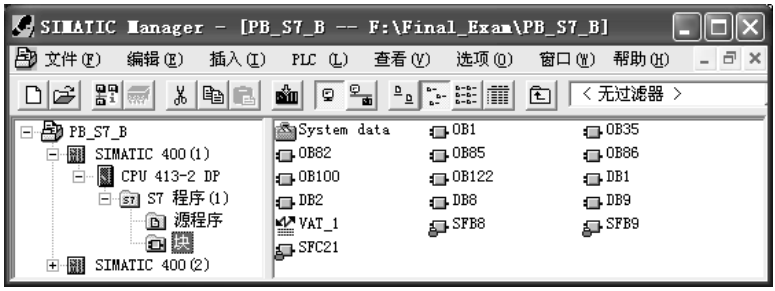


图 4-15　SIMATIC 管理器

组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

2．组态 S7 连接

组态好两个 S7-400 站后，点击工具栏上的按钮，打开 NetPro 窗口，看到连接到 PROFIBUS 网络上的两个站（见图 4-16）。选中“SIMATIC 400（1）”站点的 CPU 413-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。

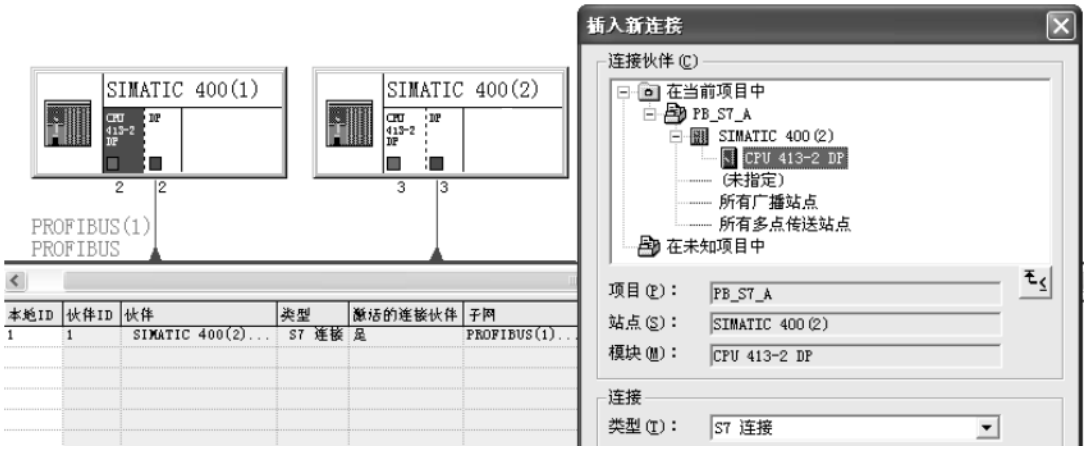


图 4-16　网络与连接的组态

在出现的“插入新连接”对话框中，系统默认的通信伙伴为站点 SIMATC 400（2）的 CPU 413-2DP，“类型”选择框中默认的连接类型为 S7 连接。

点击“确定”按钮，出现“属性-S7 连接”对话框（见图 4-17 的左图）。在“本地连接端点”区，复选框“单向”被禁止选中（该复选框为灰色），因此连接是双向的，在连接表中生成了相同的“本地 ID”和“伙伴 ID”。

复选框“建立激活的连接”是默认的设置（见图 4-17 中的左图），选中该复选框时，连接表的“激活的连接伙伴”列将显示“是”。在运行时，由本地节点建立连接。反之显示“否”，由通信伙伴建立连接。

选中 NetPro 中站点 SIMATIC 400（2）的 CPU 413-2DP 所在的小方框，下面的窗口是自动生成的该站点一侧的连接表（见图 4-18），双击连接表中的“S7 连接”，出现该站点一侧的连接属性对话框（见图 4-17 中的右图）。

图 4-17　通信双方的 S7 连接属性对话框

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
1	1	SIMATIC 400(2)...	S7 连接	否	Ethernet(1) [IE]

图 4-18　站点 SIMATIC 400（2）一侧的 S7 连接

组态好连接后，点击工具栏上的按钮，网络组态信息被编译和保存在系统数据中。

对于双向通信，必须将通信双方的连接表信息分别下载到各自的 CPU。

3．通信程序

编程时应使用组态时生成的 S7 连接的 ID 号。SFB 中的 R_ID 用于区分同一连接中不同的 SFB/FB 调用，发送方与接收方的 R_ID 应相同。为了区分两个方向的通信，令站点 SIMATIC 400（1）发送和接收的数据包的 R_ID 分别为 1 和 2，站点 SIMATIC 400（2）发送和接收的数据包的 R_ID 分别为 2 和 1。

提供发送请求信号的 M8.0 是周期为 100ms 的时钟存储器位，接收请求信号 EN_R（M0.0）为 1 状态时接收数据。下面是站点 SIMATIC 400（1）的 OB1 中的程序。

程序段 1：　发送数据

```
CALL   "USEND", DB8                  //调用 SFB 8
  REQ     :=M8.0                     //发送请求，上升沿时激活数据交换，周期为 100ms
  ID      :=W#16#1                   //S7 连接号
  R_ID    :=DW#16#1                  //发送与接收请求号
  DONE    :=M10.1                    //任务被正确执行时为 1
  ERROR   :=M10.2                    //错误标志位，为 1 时出错
  STATUS  :=MW12                     //状态字
  SD_1    :=P#DB1.DBX0.0 BYTE 20     //存放要发送的数据的 1 号地址区
```

```
        SD_2    :=P#M 40.0 BYTE 20          //存放要发送的数据的 2 号地址区
        SD_3    :=ID0                       //用本站的 ID0 控制通信伙伴的 QD0
        SD_4    :=
程序段 2：  接收数据
      CALL   "URCV" , DB9                   //调用 SFB 9
        EN_R    := M0.0                     //接收请求，为 1 时接收
        ID      :=W#16#1                    //S7 连接号
        R_ID    :=DW#16#2                   //发送与接收请求号
        NDR     :=M0.1                      //任务被正确执行时为 1
        ERROR   :=M0.2                      //错误标志位，通信出错时为 1
        STATUS  :=MW2                       //状态字
        RD_1    :=P#DB2.DBX0.0 BYTE 20      //存放接收的数据的 1 号地址区
        RD_2    :=P#M 20.0 BYTE 20          //存放接收的数据的 2 号地址区
        RD_3    := QD0                      //用通信伙伴的 ID0 控制本站的 QD0
        RD_4    :=
```

在 OB35 中，DB1.DBW0 每隔 100ms 被加 1。在初始化程序 OB100 中，用 SFC 21 将 DB 1 和 MW40～MW58 的数据发送区中的各个字分别预置为 16#4011 和 16#4014。将 DB 2 和 MW20～MW38 的数据接收区中的各个字清零。

站点 SIMATIC 400（2）与站点 SIMATIC 400（1）的程序基本上相同，在前者的 OB100 中，DB 1 和 MW40～MW58 的数据发送区中的各个字分别被预置为 16#4021 和 16#4024。

4．通信过程的监控

运行时同时打开通信双方的变量表，将它们调节到适当的大小。点击工具栏上的按钮，变量表进入监控状态。图 4-19 和图 4-20 是在运行时复制的通信双方的变量表。

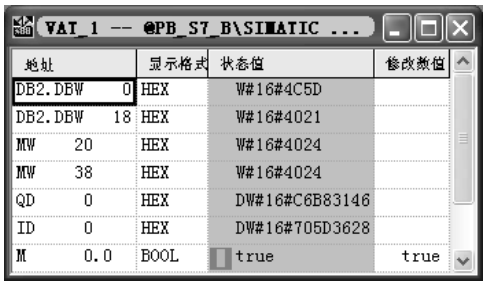


图 4-19　SIMATIC 400（1）的变量表

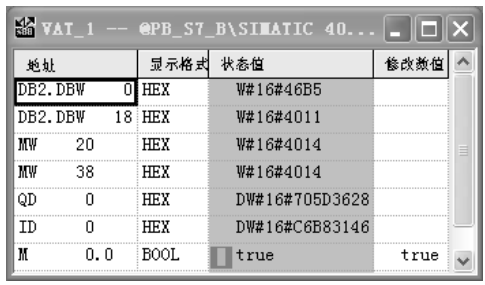


图 4-20　SIMATIC 400（2）的变量表

在时钟脉冲 M8.0 的上升沿，每 100ms USEND 发送一次数据。刚进入 RUN 模式时，接收使能位 M0.0 为 0 状态，禁止接收，双方的 DB 2、MW20～MW38 和 QD0 等数据接收区中的数据均为 0。

在接收允许信号 M0.0 的“修改数值”列输入 1，点击鼠标左键确认后 1 变为“true”。点击工具栏上的“激活修改值”按钮，“状态值”列出现绿色的指示灯符号和字符“true”，表示 CPU 中的 M0.0 变为 1 状态，允许接收数据。可以看到接收到的 DB2.DBW0 的值不断地变化，此外可以看到变量表中接收到的其他数据与发送方在 OB100 中预置的相同。

在 M0.0 为 1 状态时用外接的小开关改变 ID0 的状态，通信伙伴的 QD0 的状态随之而变。将 M0.0 的值修改为 0，写入到 CPU 后，停止接收数据，变量表中 DB2.DBW0 的值停止变化。

4.3.2 使用 BSEND/BRCV 的 S7 通信

1．硬件组态

使用 SFB BSEND/BRCV，可以进行快速的、可靠的数据传送。在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_S7_C），生成两个站，CPU 模块均为 CPU 413-2DP。系统的硬件组成与项目 PB_S7_B 相同，硬件组态和连接组态的组态过程、通信接口的地址与项目 PB_S7_B 也完全相同。

2．通信程序

SFB BSEND/BRCV 的输入参数 R_ID 用于区分同一连接中不同的 SFB/FB 调用，发送方与接收方的 R_ID 应相同。站点 SIMATIC 400（1）发送和接收的数据包的 R_ID 分别为 1 和 2，站点 SIMATIC 400（2）发送和接收的数据包的 R_ID 分别为 2 和 1。提供发送请求信号的 M8.0 是周期为 100ms 的时钟存储器位，每 100ms 发送一次数据。下面是站点 SIMATIC（1）的 OB1 中的程序。

```
程序段 1：  发送数据
        L     ID      0
        T     DB1.DBD     2                  //用本站的 ID0 控制通信伙伴的 QD0
        CALL  "BSEND" , DB12                 //调用 SFB 12
          REQ     :=M8.0                     //上升沿时激活数据交换，周期为 100ms
          R       :=M10.1                    //上升沿时中断正在进行的数据交换
          ID      :=W#16#1                   //S7 连接号
          R_ID    :=DW#16#1                  //发送与接收请求号
          DONE    :=M10.2                    //任务被正确执行时为 1
          ERROR   :=M10.3                    //错误标志位，为 1 时出错
          STATUS  :=MW12                     //状态字
          SD_1    :=P#DB1.DBX0.0 BYTE 200    //存放要发送的数据的地址区
          LEN     :=MW14                     //要发送的数据字节数
程序段 2：  接收数据
        CALL  "BRCV" , DB13                  //调用 SFB 13
          EN_R    := M0.0                    //接收请求，为 1 时允许接收
          ID      :=W#16#1                   //S7 连接号
          R_ID    :=DW#16#2                  //发送与接收请求号
          NDR     :=M0.1                     //任务被正确执行时为 1
          ERROR   :=M0.2                     //错误标志位，通信出错时为 1
          STATUS  :=MW2                      //状态字
          RD_1    :=P#DB2.DBX0.0 BYTE 200    //存放接收的数据的地址区
          LEN     :=MW4                      //已接收的数据字节数
        L     DB2.DBD     2
        T     QD      0                      //用通信伙伴的 ID0 控制本站的 QD0
```

BSEND 的输入参数 LEN 是要发送的数据的字节数，数据类型为 WORD（无符号的字）。因为不能使用常数，设置 LEN 的实参为 MW14，在初始化程序 OB100 中用下面两条语句预置它的初始值为 200：

```
        L     200
```

```
T      MW     14                    //预置要发送的字节数
```

在初始化程序 OB100 中，用 SFC 21 将 DB 1 的数据发送区的各个字预置为 16#4131。将 DB 2 的数据接收区的各个字清零。在 OB35 中，每隔 100ms 将 DB1.DBW0 加 1。

站点 SIMATIC 400（2）与站点 SIMATIC 400（1）的程序基本上相同。在前者的 OB100 中，发送区的数据被初始化为 W#16#4132。

3．通信过程的监控

图 4-21 和图 4-22 是在运行时复制的通信双方的变量表。图中接收允许信号 M0.0 被置为 true。在运行时可以看到双方接收到的 DB2.DBW0 在不断地变化，此外可以看到数据接收区的最后一个字 DBW198 的值与发送方预置的相同。

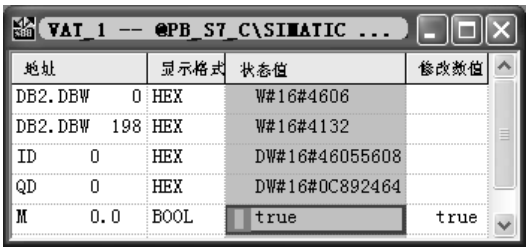


图 4-21　SIMATIC 400（1）的变量表

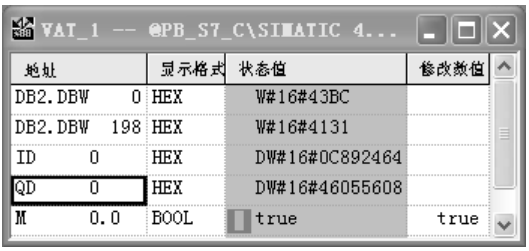


图 4-22　SIMATIC 400（2）的变量表

在运行时用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD0 的状态随之而变。

4.3.3　CP 443-5 在 S7 通信中的应用

1．S7 单向通信

在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_S7_D）。在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。设置 CPU 的 MPI 地址和 DP 地址均为 2，默认的工作模式为 DP 主站。

在 SIMATIC 管理器中生成另一个 400 站点，在 HW Config 中，将电源模块、CPU 413-2DP、信号模块和 CPU 443-5 Ext 插入机架。设置 CPU 的 MPI 地址和 DP 地址均为 3。插入 CP 443-5 Ext 时，在自动打开的 DP 接口属性对话框中，点击“新建”按钮，生成 PROFIBUS-DP 网络。设置传输速率为默认的 1.5 Mbit/s，配置文件为“标准”，CP 的工作模式为“无 DP”，DP 地址为 4。组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

组态好两个站后，点击工具栏上的按钮，打开 NetPro 窗口（见图 4-23），将“SIMATIC 400（1）”站点集成的 DP 接口连接到 PROFIBUS 网络上。

选中“SIMATIC 400（1）”站点中 CPU 413-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。

在出现的“插入新连接”对话框中，系统默认的通信伙伴为 CPU 413-2DP，默认的连接类型为 S7 连接。点击“确定”按钮，出现 S7 连接属性对话框（见图 4-24），该连接为双向连接，连接表中的“本地 ID”和“伙伴 ID”均为 1。

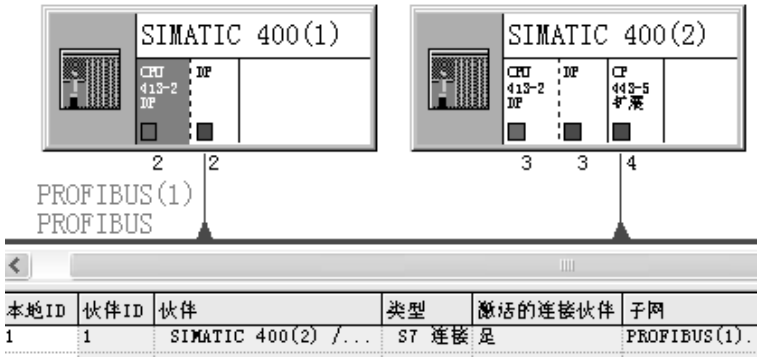


图 4-23　网络与连接组态

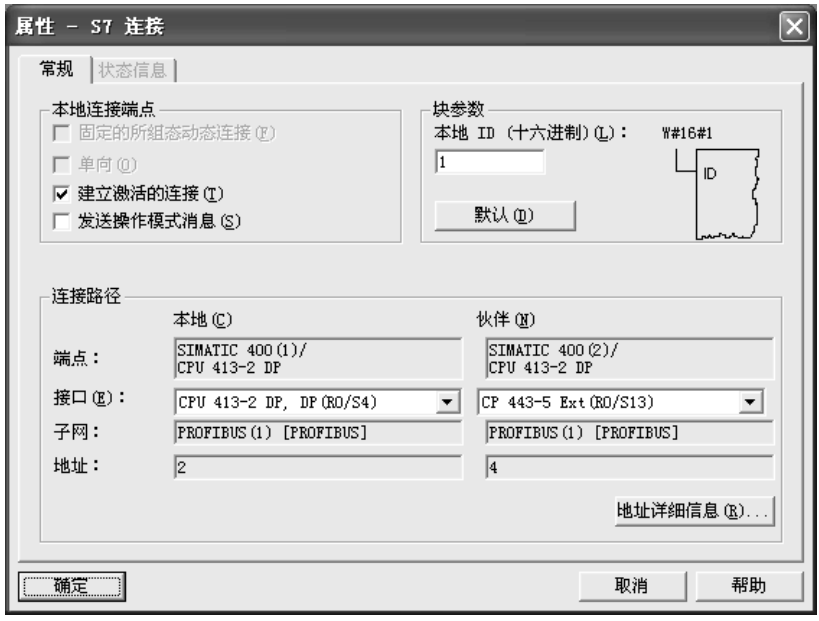


图 4-24　S7 连接属性对话框

本项目的程序与项目 PB_S7_A 的基本上相同。在站点 SIMATIC 400（1）的 OB1 中，调用 SFB GET 和 PUT，读取站点 SIMATIC 400（2）中 4 个地址区的数据。SFB PUT 和 GET 中的通信请求信号 REQ 由周期为 200ms 的时钟存储器位 M8.1 和与它反相的 M10.0 提供。变量表、通信过程的监控方法与项目 PB_S7_A 的也完全相同。

2．使用 USEND/URCV 的 S7 通信

在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_S7_E），生成两个 SIMATIC 400 站，CPU 模块均为 CPU 413-2DP。系统的硬件组成与项目 PB_S7_D 相同，硬件组态和连接组态的组态过程、通信接口的地址与项目 PB_S7_D 也完全相同。

本项目的程序与项目 PB_S7_B 的完全相同。在通信双方的 OB1 中调用 SFB USEND 和 URCV，发送和接收数据。USEND 中的发送请求信号由周期为 100ms 的时钟存储器位 M8.0 提供，在 URCV 的接收请求信号 EN_R（M0.0）为 1 状态时接收数据。变量表、通信过程的

监控方法与项目 PB_S7_B 的也完全相同。

4.4 通过 S7 连接控制和监视远程 PLC 的运行模式

在 STEP 7 中创建一个项目（见随书光盘中的例程 PB_CTRL），生成两个站，CPU 模块均为 CPU 413-2（见图 4-25）。

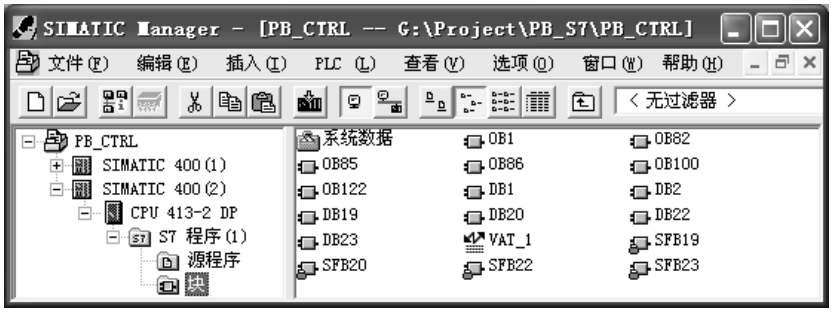


图 4-25 SIMATIC 管理器

新建一条 DP 网络，设置其配置文件为“标准”。点击 SIMATIC 管理器工具栏上的按钮，打开网络组态工具 NetPro，将两个站连接到 DP 网络上，设置它们的 MPI 和 DP 站地址分别为 2 和 3（见图 4-26），“工作模式”均为 DP 主站。

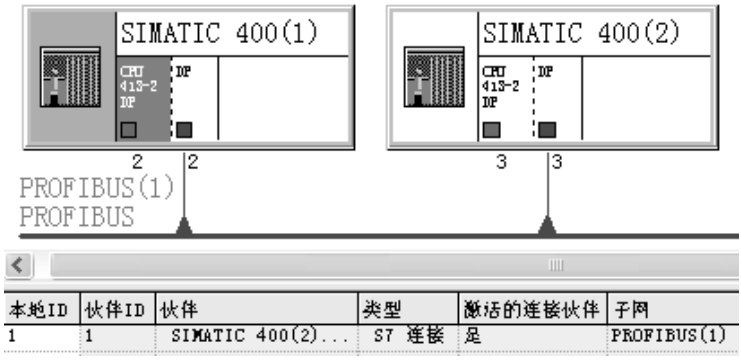


图 4-26 网络与连接的组态

选中 2 号站的 CPU，双击下面的连接表的第 1 行，生成一个双向的 S7 连接。

S7-400 可以通过 SFB 改变远程设备的运行状态，属于单边编程方式。

1．调用 SFB 19 “START”使远程设备热启动或冷启动

在 SFB 19“START”的请求信号 REQ 的上升沿，激活由 S7 连接 ID 寻址的远程设备的热启动或冷启动。远程设备应满足下列条件：

1）S7-300/400 或 C7-300 的 CPU 处于 STOP 模式。

2）CPU 的模式选择开关在 RUN 或 RUN-P 位置。

完成暖启动或冷启动后，远程设备切换到 RUN 模式，并发送一个执行成功的肯定应答。

远程的 S7-400 CPU 处于 STOP 模式时，调用 SFB 21“RESUME”，可以使它热启动。

2．调用 SFB 20“STOP”将远程设备切换到 STOP 模式

如果远程 S7/M7-300/400 或 C7-300 的 CPU 处于 RUN、HALT 或 STARTUP 模式，且 CPU 的模式选择开关在 RUN 或 RUN-P 位置，在 SFB 20“STOP”的请求信号 REQ 的上升沿，由 S7 连接 ID 寻址的远程设备将切换到 STOP 模式。

下面是 3 号站的 OB1 调用 SFB 19 和 SFB 20 的程序。

程序段 1：将远程设备切换到 RUN 模式

```
CALL   "START" , DB19                  //调用 SFB 19
  REQ        :=M20.6                   //通信请求，上升沿时激活 SFB
  ID         :=W#16#1                  //S7 连接号
  DONE       :=DB1.DBX10.1             //操作正确完成后为 1
  ERROR      :=DB1.DBX10.2             //错误标志
  STATUS     :=DB1.DBW14               //状态字
  PI_NAME    :="data".pi_name          //字符串'P_PROGRAM '
  ARG        :=                        //未设置此参数时为热启动
  IO_STATE   :=                        //S7 PLC 未用此参数
```

程序段 2：将远程设备切换到 STOP 模式

```
CALL   "STOP" , DB20                   //调用 SFB 20
  REQ        :=M20.7                   //通信请求，上升沿时激活 SFB
  ID         :=W#16#1                  //S7 连接号
  DONE       :=DB1.DBX17.1             //操作正确完成后为 1
  ERROR      :=DB1.DBX17.2             //错误标志
  STATUS     :=DB1.DBW20               //状态字
  PI_NAME    :="data".pi_name          //字符串'P_PROGRAM '
  IO_STATE   :=                        //S7 PLC 未用此参数
```

3．SFB 19 和 SFB 20 的公用参数

SFB 19 和 SFB 20 的参数 PI_NAME 是指向存储要启动的程序的名称（ASCII 代码）的数据区。对于 S7 PLC，参数 PI_NAME 应为字符串‘P_PROGRAM’。符号名为 data 的 DB 2 中定义了一个有 5 个字元素的数组 pi_name。下面是 OB100 中的程序，用于将字符串“P_PROGRAM”写入该数组。值得注意的是，最后一个字的字符“M”后面有一个空格。

```
L     'P_'
T     "data".pi_name[1]
L     'PR'
T     "data".pi_name[2]
L     'OG'
T     "data".pi_name[3]
L     'RA'
T     "data".pi_name[4]
L     'M '
T     "data".pi_name[5]
```

如果通信伙伴是 S7 系列 PLC，不给参数 IO_STATE 分配任何数值，其默认值为 B#16#0。

完成 SFB 指定的操作后，远程设备切换到指定的运行模式，并发送一个肯定的执行应答。接收到肯定应答之后，SFB 的状态参数 DONE 被置位为 1。

如果没有给 SFB 19 的参数 ARG 分配数值，则在远程设备上执行热启动。如果远程设备支持冷启动，且为 ARG 分配数值“C”，则在远程设备上执行冷启动。

4．调用 SFB 22“STATUS”来查询远程伙伴的状态

SFB 22 用于按用户的请求提供通信伙伴（S7-400 CPU/M7-300/400）的运行状态。

在输入信号 REQ 的上升沿，将任务发送到远程伙伴。通过判断响应信息，确定是否有错误。如果没有出错，在下一次调用 SFB 时，将接收到的远程设备的状态复制到变量 PHYS、LOG 和 LOCAL。下面是 3 号站的 OB1 调用 SFB 22 的程序。

程序段 3：查询远程伙伴的运行模式

```
CALL   "STATUS" , DB22          //调用 SFB 22
  REQ     :=M21.1               //通信请求，上升沿时激活 SFB
  ID      :=W#16#1              //S7 连接号
  NDR     :=DB1.DBX30.1         //操作正确完成后为 1
  ERROR   :=DB1.DBX30.2         //错误标志
  STATUS  :=DB1.DBW32           //状态字
  PHYS    :=DB1.DBB34           //物理状态，10H 正在执行功能，13H 为服务请求
  LOG     :=DB1.DBB36           //逻辑状态，00H 允许状态改变
  LOCAL   :=DB1.DBW38           //第一个字节是通信伙伴的当前状态
```

变量 LOCAL 的最小长度为 2B，用高字节来表示伙伴设备（S7 CPU）的状态标识符（见表 4-3）。

表 4-3　设备状态标识符 LOCAL

工作模式	标识符	工作模式	标识符	工作模式	标识符
STOP	00H	热启动	03H	RUN_P	09H
暖启动	01H	HOLD	04H	LINK-UP	0BH
RUN	02H	冷启动	06H	UPDATE	0CH

5．调用 SFB 23“USTATUS”接收远程通信伙伴操作模式变化的信息

在组态被读取操作模式状态的 2 号站的 S7 连接时，应选中复选框“发送操作模式消息”（见图 4-27）。2 号站的操作模式变化时，将会主动发送它的状态消息给调用 SFB 23 的 3 号站的 CPU。

如果 SFB 23 的使能输入 EN_R 为 1，并且有来自通信伙伴的数据帧，在下一次调用 SFB 时，将状态信息输入到变量 PHYS、LOG 和 LOCAL 中。状态参数 NDR 为 1 表示操作完成。

下面是 3 号站的 OB1 调用 SFB 23 的程序：

程序段 4：接收远程设备的状态变化

```
CALL   "USTATUS" , DB23         //调用 SFB 23
  EN_R    :=M21.2               //为 1 时准备接收控制参数
  ID      :=W#16#1              //S7 连接号
  NDR     :=DB1.DBX40.1         //操作正确完成后为 1
  ERROR   :=DB1.DBX40.2         //错误标志
```

```
STATUS :=DB1.DBW42          //状态字
PHYS   :=DB1.DBB44          //物理状态，10H 执行功能，13H 服务请求
LOG    :=DB1.DBB46          //逻辑状态，00H 允许状态改变
LOCAL  :=DB1.DBW48          //第一个字节是通信伙伴的当前状态
```

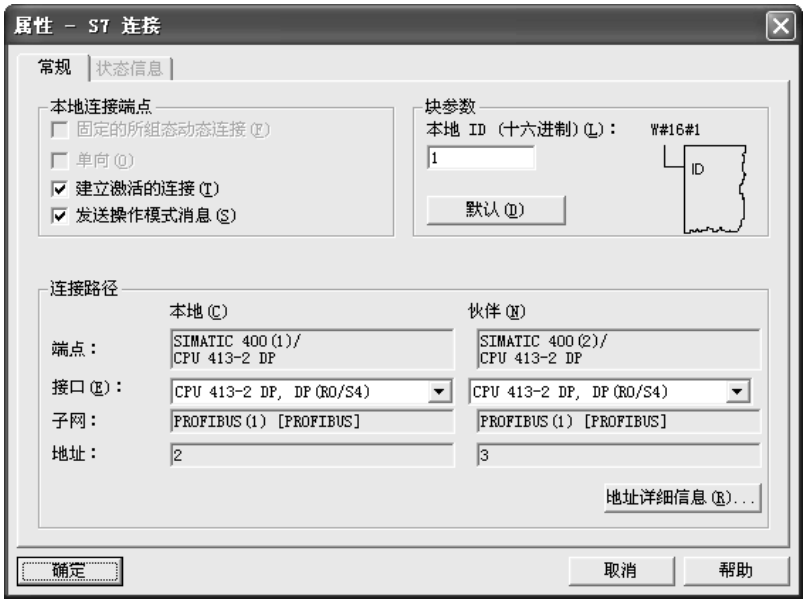


图 4–27　2 号站的 S7 连接属性对话框

6．实验结果

（1）将远程设备切换到 RUN 模式

图 4–28 和图 4–29 是 3 号站的变量表，其中 DB 1 的 DBB34、DBB36 和 DBW38 分别是 SFB 22 的 PHYS（物理状态）、LOG（逻辑状态）和 LOCAL（通信伙伴的当前状态），DBB44、DBB46 和 DBW48 分别是 SFB 23 的 PHYS、LOG 和 LOCAL。

2 号站在 STOP 模式时，用 3 号站的变量表将 1（true）写入 SFC 19 的使能信号 M20.6，将 0（false）写入 SFC 20 的使能信号 M20.7，在 M20.6 的上升沿激活 SFB 19，将 2 号站切换到 RUN 模式（见图 4–28）。

SFB 22 和 SFB 23 的 PHYS（物理状态）为 16#10，表示正在执行功能。SFB 23 的 EN_R 信号 M21.2 一直为 1 状态，3 号站进入 RUN 模式后，SFB 23 的输出参数 LOCAL（DB1.DBW48）的高字节自动变为 16#02（见图 4–28 和表 4–3），表示通信伙伴为 RUN 模式。

SFB 22 和 SFB 23 的区别如下：

远程伙伴（2 号站）的状态改变后，SFB 22 的状态数据不会自动改变，需要用 SFC 22 的使能信号 M21.1 的上升沿来读取远程伙伴的当前状态。

如果 SFB 23 的 EN_R 为 1，远程伙伴的状态改变时，它的 LOCAL 的值马上随之而变。

（2）将远程设备切换到 STOP 模式

2 号站在 RUN 模式时，用 3 号站的变量表将 0 写入 M20.6，将 1 写入 M20.7（见图 4–29），在 M20.7 的上升沿，激活 SFB 20，将 2 号站切换到 STOP 模式。SFB 23 的输出参数 LOCAL

（DB1.DBW48）的高字节自动变为 16#00（见图 4-29），表示通信伙伴为 STOP 模式。

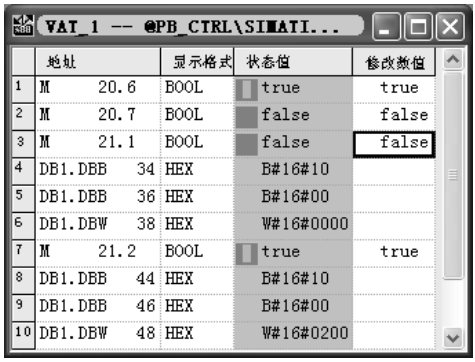


图 4-28　3 号站的变量表

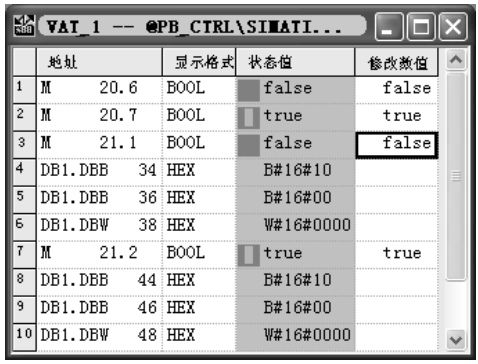


图 4-29　3 号站的变量表

7．用 S7-400 远程控制 S7-300 的运行模式

随书光盘中的项目“PB_CTRL2”的硬件结构和组态方法与项目“PB_CTROL”的基本相同，二者的区别仅在于前者被控制的是 CPU 313C-2DP，建立的是 S7 单向连接。两个项目的程序、变量表和调试方法完全相同，具体情况见随书光盘的项目。经实验验证，CPU 413-2DP 可以用 SFB 19 和 SFB 20 切换 CPU 313C-2DP 的 RUN/STOP 运行模式，但是不能用 SFB22、SFB23 查询和接收 CPU 313C-2DP 的运行模式。

8．远程站点的监控功能在 MPI 网络上的应用

SFB 19～SFB 23 可以用于 MPI、PROFIBUS-DP 和工业以太网，组态和编程的方法相同。

（1）S7-400 用 MPI 远程监控 S7-400 的运行模式

随书光盘中的项目“\Project\MPI_S7\MPI_CTRL”的硬件结构和组态方法与项目“PB_CTROL”的完全相同，二者的区别仅在于前者建立的是 MPI 网络上的连接。具体的情况见随书光盘上的项目。两个项目的变量表和调试方法、实验结果也完全相同。

（2）S7-400 用 MPI 远程控制 S7-300 的运行模式

随书光盘中的项目“\Project\MPI_S7\ MpiCtrl2”的硬件结构和程序与项目“PB_CTRL2”的完全相同，二者的区别仅在于前者建立的是 MPI 网络上的连接。具体的情况见随书光盘的该项目。两个项目的变量表和调试方法、实验结果也完全相同。

4.5　同一 DP 主站系统的 FDL 通信

4.5.1　FDL 通信的基本概念

FDL 是 PROFIBUS 的第 2 层——现场总线数据链路层（Fieldbus Data Link）的缩写，用于实现 PROFIBUS 主站之间的通信。它是安全系数很高的发送/接收双向数据通信服务，可以有效地检测出通信的错误。

PROFIBUS 网络中的 FDL 连接与工业以太网中的 ISO、ISO-on-TCP、TCP、UDP 连接统称为 S5 兼容的连接，它们使用相同的通信功能（FC）AG_RECV 和 AG_SEND。

在 PROFIBUS-DP 通信中，具有令牌功能的 PROFIBUS-DP 主站轮询无令牌功能的从站，

进行数据交换。与此不同，PROFIBUS FDL 的每一个通信站点都具有令牌功能，通信以令牌环的方式进行数据交换，某一个站得到令牌后，才能通过令牌与别的站交换数据。每一个 FDL 站点都可以和多个站点建立通信连接。FDL 既可以用于 S7 PLC 之间，也可以用于 S7 PLC 与 S5 PLC 或 PC 之间的通信。

FDL 数据传输是双向的，可以在 FDL 连接上同时进行发送和接收。两个站都具有同样的权限，每个站都可以触发发送和接收过程。

FDL 包括 MAC（媒体访问控制）和 LLC（逻辑链路控制）。MAC 控制主站之间的令牌传递和主站、从站之间的主从方式数据交换，LLC 主要负责建立和终止逻辑通信链路。FDL 支持 SDA（有确认的数据发送）和 SDN（无确认的数据发送）、自由第 2 层通信、广播通信和多点传送通信。

只有 PROFIBUS 通信处理器（CP）才支持 FDL 的数据通信，例如，用于 S7-300 系列 PLC 的 CP 342-5 和 CP 343-5，用于 S7-400 系列 PLC 的 CP 443-5，用于 S5 系列 PLC 的 CP 5431，以及用于上位机的 CP 5512、CP 5612、CP 5613 PROFIBUS 网卡。S7-200 不支持 FDL 通信。

通信处理器可以同时与多个主站建立通信连接，大多数通信处理器的 FDL 连接个数最多 16 个。S7 连接采用 OSI（开放系统互连模型）的第 1、2 和 7 层，专门用于西门子 PLC 之间的大数据量通信。因为 FDL 只采用了 OSI 模型的第 1、2 层，传输请求由硬件发起，传输速率快，但是传输的数据量较小（仅 240B）。

4.5.2 硬件组态与 FDL 连接组态

本节介绍的是指定通信伙伴的 FDL 连接，通过对连接进行组态，来指定在同一个 STEP 7 项目内唯一的连接伙伴。

1．组态 S7-400 站点

在 STEP 7 中创建一个名为 FDL_1 的项目，FDL 通信的例程在随书光盘的文件夹“\Project\PB_FDL”中。在 HW Config 中，将电源模块、CPU 和信号模块插入机架，CPU 模块为 CPU 413-2DP（见图 4-30）。CPU 的 MPI 接口和集成 DP 接口的地址均为 2，未使用集成的 DP 接口。

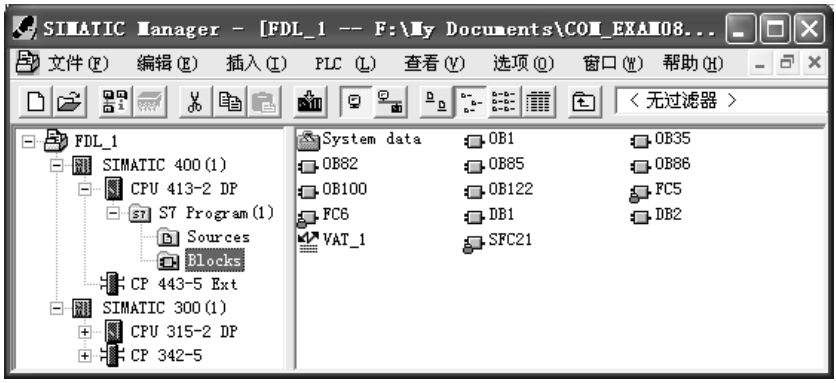


图 4-30 SIMATIC 管理器

将 CP 443-5 Ext 插入机架，点击自动打开的“属性– PROFIBUS 接口 DP”对话框的参数

选项卡中的“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，设置网络的传输速率为 1.5Mbit/s，配置文件为“标准”。点击“确定”按钮，返回“属性-PROFIBUS 接口 DP”对话框，设置 CP 443-5 Ext 的站地址为 3，将它连接到网络上。

返回 HW Config 后，双击 CP 443-5 Ext，打开 CP 443-5 Ext 的属性对话框，在“工作模式”选项卡，设置 CP 的工作模式为“无 DP”。点击“确定”按钮，返回 HW Config。组态结束后，点击工具栏上的按钮，编译并保存组态信息。CP 443-5 Ext 属性视图的“地址”选项卡中默认的输入、输出的起始字节地址为 2040，即十六进制数 16#7F8。

2．组态 S7-300 站点

在 SIMATIC 管理器中生成一个 S7-300 站。在 HW Config 中，将 CPU 315-2DP 插入机架，在自动打开的“属性 – PROFIBUS 接口 DP”对话框的“参数”选项卡中，设置集成的 DP 接口地址为 4，不连网。CPU 的 MPI 地址为 3。

将 CP 342-5 插入机架，在自动打开的“属性-PROFIBUS 接口 DP”对话框的“参数”选项卡中，设置其 DP 接口地址为 5，MPI 地址为 4。将它连接到 CP 443-5Ext 所在的 PROFIBUS（1）网络上，工作模式也设置为“无 DP”。“地址”选项卡中默认的输入、输出的起始字节地址为 320，即十六进制数 16#140。组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

3．组态 FDL 连接

组态好 S7-300 站和 S7-400 站后，关闭 HW Config，打开 NetPro，看到连接到 PROFIBUS 网络上的两个站（见图 4–31）。选中 S7-400 CPU，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。

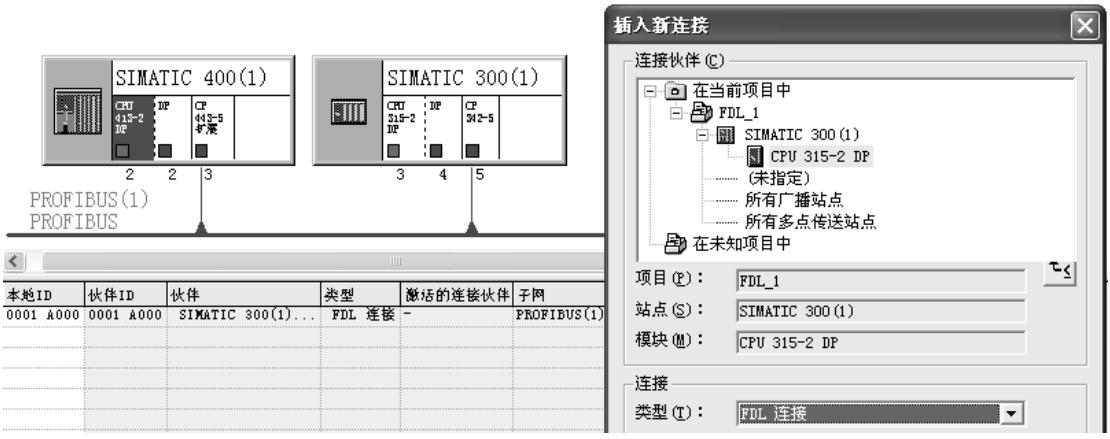


图 4–31　网络与连接组态

在弹出的“插入新连接”对话框中，选中“连接伙伴”列表框中的连接对象为与本站通信的 CPU 315-2DP（见图 4–31 中的右图），将连接类型设为“FDL 连接”。

点击“确定”按钮，自动打开“属性 – FDL 连接”对话框（见图 4–32）。在“常规信息”选项卡的“块参数”区中，自动生成了参数“标识号”（ID）和 LADDR（CP 模块的起始地址），编程时将会用到这两个参数。

在“地址”选项卡（见图 4–32）中，设置本站和通信伙伴的 LSAP（连接服务访问点）。

返回 NetPro 后，选中 CPU 315-2DP 所在的小方框，在下面的连接表中，可以看到自动生

成的 FDL 连接的信息（见图 4-33）。图 4-34 是 5 号站的 FDL 连接属性对话框。

图 4-32　3 号站的 FDL 连接属性对话框

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
0001 A000	0001 A000	SIMATIC 400(1)...	FDL 连接	-	PROFIBUS(1) [PROFIBUS]

图 4-33　5 号站的 FDL 连接表

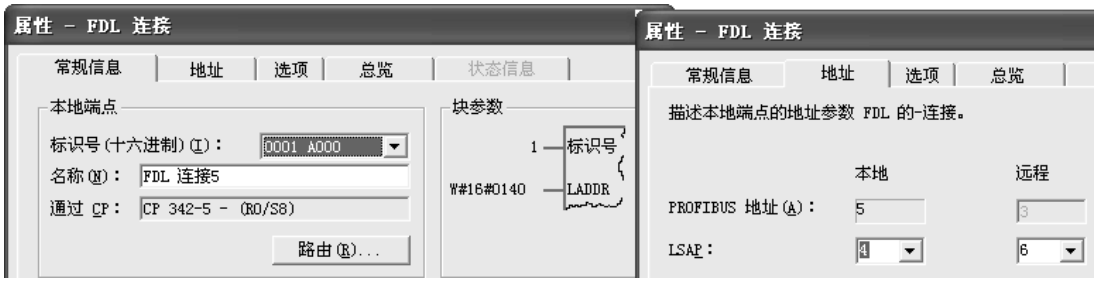


图 4-34　5 号站的 FDL 连接属性对话框

参数 LADDR 与 CP 342-5 所在的插槽有关。CP 342-5 如果在 4 号槽，它是电源模块、CPU 模块和接口模块 IM 之外从左到右的第 1 块模块，其 LADDR 的值为 W#16#0100（对应的十进制数为 256），该地址也是 S7-300 分配给 4 号槽的模拟量模块和 CP 模块的起始地址。本项目的 CP 342-5 在 8 号槽，LADDR 的值为 320（W#16#140）。

4.5.3　编写验证通信的程序

1．CPU 413-2DP 的通信程序

S5 兼容通信的双方通过调用程序编辑器左边窗口的文件夹“\库\SIMATIC_NET_CP”中的通信功能 FC 5 AG_SEND 和 FC 6 AG_RECV 来实现 FDL 服务。S7-300 和 S7-400 分别使用子文件夹“CP 300”和“CP 400”中的 FC，不能混用。AG_SEND 用于将用户数据区的数据传送给 PROFIBUS CP，再发送出去，AG_RECV 用于将 PROFIBUS CP 接收的数据存放到用户数据区。

下面是 CPU 413-2DP 的 OB35 中的程序， ACT 是 FC 5 “AG_SEND”的发送使能位，ACT 为 1 状态时发送数据，为了实现周期性的数据发送，令 ACT 一直为 1 状态（true），如果在 OB1 中调用 FC 5，将在每一个循环扫描周期发送一次数据，发送将过于频繁。因此将发送程序放在中断循环周期为 100ms 的 OB35 中，每 100ms 发送一次数据。

程序段 1：每 100ms 将 DB1.DBW0 加 1

```
L       DB1.DBW      0
+       1
T       DB1.DBW      0
```

程序段 2：发送程序

```
L       ID       0
T       DB1.DBD    2                  //用本站的 ID0 控制通信伙伴的 QD4
CALL    "AG_SEND"                     //调用 FC 5
  ACT      :=TRUE                     //发送使能位
  ID       :=1                        //连接 ID，见图 4-32 中的块参数
  LADDR    :=W#16#7F8                 //CP 443-5 Ext 的起始地址，见图 4-32 中的块参数
  SEND     :=P#DB1.DBX 0.0 BYTE 240   //存放要发送的数据的地址区
  LEN      :=240                      //发送数据的字节数
  DONE     :=M10.2                    //每次发送成功产生一个脉冲
  ERROR    :=M10.3                    //错误标志位
  STATUS   :=MW12                     //状态字
```

下面是 CPU 413-2DP 的 OB1 中的接收程序：

程序段 1：接收程序

```
CALL    "AG_RECV"                     //调用 FC 6
  ID       :=1                        //连接 ID
  LADDR    :=W#16#7F8                 //CP 443-5 Ext 的起始地址
  RECV     :=P#DB2.DBX 0.0 BYTE 240   //存放接收的数据的地址区
  NDR      :=M0.1                     //每次接收完新数据产生一个脉冲
  ERROR    :=M0.2                     //错误标志位
  STATUS   :=MW2                      //状态字
  LEN      :=MW4                      //接收的数据字节数
L       DB2.DBD    2
T       QD       0                    //用通信伙伴的 ID0 控制本站的 QD0
```

在初始化程序 OB100 中，调用 SFC 21 预置发送数据区的字的初值为 W#16#4444，将数据接收区清零。

程序段 1：初始化存放要发送的数据的地址区

```
L       W#16#4444
T       LW      20
CALL    "FILL"
  BVAL      :=LW20                    //源数据
  RET_VAL   :=LW22                    //错误代码
  BLK       :=P#DB1.DBX0.0 BYTE 240   //地址区
```

程序段 2：将存放接收到的数据的地址区清零

```
L       W#16#0
T       LW      20
CALL    "FILL"
  BVAL      :=LW20                    //源数据
  RET_VAL   :=LW22                    //错误代码
```

```
BLK       :=P#DB2.DBX0.0 BYTE 240     //地址区
```

2．CPU 315-2DP 的通信程序

CPU 315-2DP 的发送程序和接收程序与 CPU 413-2DP 的基本上相同，参数 LADDR 的值为 W#16#140。

下面是 CPU 315-2DP 的 OB35 中的程序：

```
程序段 1：每 100ms 将 DB1.DBW0 加 1
      L     DB1.DBW     0
      +     1
      T     DB1.DBW     0
程序段 2：发送程序
      L     ID      0
      T     DB1.DBD   2                  //用本站的 ID0 控制通信伙伴的 QD0
      CALL  "AG_SEND"                    //调用 FC 5
        ACT     :=TRUE                   //发送使能位
        ID      :=1                      //连接 ID
        LADDR   :=W#16#140               //CP 342-5 的起始地址
        SEND    :=P#DB1.DBX 0.0 BYTE 240 //存放要发送的数据的地址区
        LEN     :=240                    //发送数据的字节数
        DONE    :=M10.1                  //每次发送成功产生一个脉冲
        ERROR   :=M10.2                  //错误标志位
        STATUS  :=MW12                   //状态字
```

下面是 CPU 315-2DP 的 OB1 中的接收程序：

```
程序段 1：接收程序
      CALL  "AG_RECV"                    //调用 FC 6
        ID      :=1                      //连接 ID
        LADDR   :=W#16#140               //CP 342-5 的起始地址
        RECV    :=P#DB2.DBX 0.0 BYTE 240 //存放接收的数据的地址区
        NDR     :=M0.1                   //每次接收完新数据产生一个脉冲
        ERROR   :=M0.2                   //错误标志位
        STATUS  :=MW2                    //状态字
        LEN     :=MW4                    //接收数据长度
      L     DB2.DBD   2
      T     QD      4                    //用通信伙伴的 ID0 控制本站的 QD4
```

CPU 315-2DP 的 OB100 中的程序与 CPU 413-2DP 的基本上相同，只是将发送区中的数据字初始化为 W#16#3333。

将组态信息和程序分别下载到两台 PLC 后，用电缆连接两块 CP 和计算机的 CP 5613 的 DP 接口，将 CP 5613 设置为 PROFIBUS，通过 PROFIBUS 网络对 S7-300 和 S7-400 进行监控。

在 STEP 7 中同时打开主站和从站的变量表（见图 4–35 和图 4–36）。通信双方在 OB35 中将 DB1.DBW0 加 1，然后发送到对方的 DB2.DBW0。在变量表中可以看到双方接收到的 DB2.DBW0 在不断地变化，数据接收区的最后一个字 DBW238 与发送方在 OB100 中预置的相同。用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD0 或 QD4 的状态随之而变。

图 4-35　3 号站的变量表

图 4-36　5 号站的的变量表

4.5.4　S7-300 之间的 FDL 通信

1．硬件与连接组态

打开 SIMATIC 管理器，新建一个名为 FDL_2 的项目（见随书光盘中的同名例程）。该项目用于两台 CPU 315-2DP 之间的 FDL 通信。为了实现 S7-300 之间的 FDL 通信，通信的双方都需要使用 PROFIBUS 通信模块 CP 342-5。

本项目和本节后面的项目均使用相同的硬件，两台 CPU 315-2DP 的订货号为 6ES7 315-2AF03 0AB0，CP 342-5 的订货号为 6SE7 342-5DA01 0XE0。在 SIMATIC 管理器中生成一个 S7-300 站。在 HW Config 中，将电源模块、CPU 315-2DP 和信号模块插入机架。CPU 的 MPI 接口地址和集成的 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 342-5 插入机架，设置其 DP 地址和 MPI 地址均为 3。在它的属性对话框中生成一个 PROFIBUS 网络，默认的传输速率为 1.5 Mbit/s，设置配置文件为“标准”，将 CP 342-5 连接到网络上。在 CP 342-5 属性对话框的“工作模式”选项卡，设置 CP 的工作模式为“无 DP”。组态好硬件后，点击工具栏上的按钮，编译并保存组态信息。

在 SIMATIC 管理器中生成另一个 S7-300 站，组态的方法与前一个站基本上相同。CPU 的 MPI 接口和集成 DP 接口的地址均为 4，未使用集成的 DP 接口。CP 342-5 的 DP 地址和 MPI 地址均为 5，将它的 DP 接口连接到 PROFIBUS 网络上。设置 CP 的工作模式为“无 DP”。组态好硬件后，点击工具栏上的按钮，编译并保存组态信息，然后关闭 HW Config。

点击 SIMATIC 管理器工具栏上的按钮，打开网络组态工具 NetPro（见图 4-37），选中站点“SIMATIC 300（1）”的 CPU 所在的小方框，在下面的窗口出现连接表。双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”列表中的连接对象设为另一台 CPU 315-2DP，连接类型为“FDL 连接”。

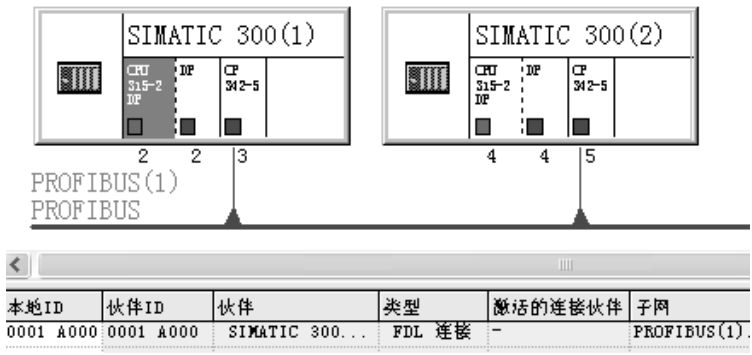


图 4-37　网络与连接组态

点击“确定”按钮，在自动打开的连接属性对话框的“地址”选项卡（见图 4-38）中，显示本地（本站）的 DP 站地址为 3，设置 LSAP（连接服务访问点）为 18。通信伙伴（远程）的站地址为 5，设置它的 LSAP 为 10，点击“确定”按钮确认。图 4-37 的下面是组态好 FDL 连接后的连接表。图 4-39 是 5 号站的 FDL 连接属性对话框。

图 4-38　3 号站的 FDL 连接属性对话框

在组态时发现，某些型号的 PROFIBUS CP 不能用于某些方式的 FDL 通信。例如，订货号为 6GK7 443-5DX00 0XE0 的 CP 443-5 Ext，不能用于通信对象未指定、广播方式、多点传送方式和自由第 2 层通信 FDL 通信。

图 4-39　5 号站的 FDL 连接属性对话框

2．FDL 通信的编程与程序运行

在通信双方的 OB1 中调用通信功能 FC 6“AG_RECV”，在 OB35 中调用 FC 5“AG_SEND”来实现 FDL 服务。通信程序的编程方法、变量表和调试方法与项目 FDL_1 中的基本上相同。

4.6　不同 DP 主站系统与不同项目的 FDL 通信

4.6.1　不同 DP 主站系统的 FDL 通信

1．硬件组态

本节介绍的项目中，两台 S7-300 PLC 在同一个 PROFIBUS 物理子网络上和同一个项目中，但是分别在不同的 PROFIBUS-DP 主站系统内，它们之间也可以用 FDL 连接进行通信。

打开 SIMATIC 管理器，用新建项目向导建立一个新的项目，项目名称为“FDL_2Net”（见随书光盘中的同名例程）。

在 SIMATIC 管理器中生成两个 S7-300 站，它们的硬件结构、MPI 和 DP 站地址均与项

目 FDL_2 中的相同，两个项目的区别在于本项目的两块 CP 342-5 分别连接在两条 DP 网络上（见图 4-40），即分别属于两个不同的 PROFIBUS 主站系统。为了实现这一要求，组态两台 S7-300 的 CP 342-5 时，分别在它们的属性对话框中生成 PROFIBUS 网络，并将 CP 342-5 连接到各自的网络上，设置 CP 的工作模式均为“无 DP”。组态好硬件后，点击工具栏上的按钮，编译并保存组态信息，然后关闭 HW Config。

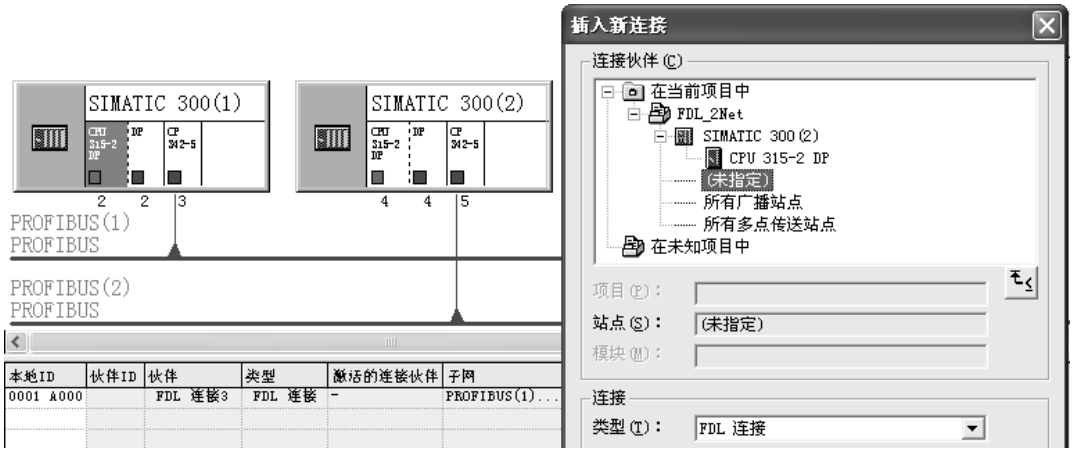


图 4-40　组态 FDL 连接

在 SIMATIC 管理器中点击工具栏上的按钮，打开网络组态工具 NetPro，可以看到分别连接到两条 PROFIBUS 网络上的两个站。

选中图 4-40 中 SIMTIC-300（1）的 CPU 所在的方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中（见图 4-40），将“连接伙伴”列表框中的连接对象设为“未指定”，连接类型设为“FDL 连接”。

点击“确定”按钮，在自动打开的连接属性窗口的“地址”选项卡中（见图 4-41），本站（本地）的 DP 站地址为 3，设置 LSAP（连接服务访问点）为 18。与项目 FDL_2 相比，通信伙伴（远程）的站地址不是自动指定的，将它设置为 5，设置其 LSAP 为 10。点击“确定”按钮确认。

图 4-41　3 号站的 FDL 连接的地址信息

图 4-42　5 号站的 FDL 连接的地址信息

选中图 4-40 中 SIMTIC-300（2）的 CPU 所在的方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接（见图 4-43）。连接对象设置为“未指定”，连接

类型为“FDL 连接”。

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
0001 A000		FDL 连接4	FDL 连接	-	PROFIBUS(2)...

图 4-43　5 号站一侧的连接表

点击“确定”按钮，在自动打开的连接属性窗口的“地址”选项卡中（见图 4-42），本站（本地）的 DP 站地址为 5，设置其 LSAP（连接服务访问点）为 10。设置远程通信伙伴的站地址为 3，LSAP 为 18。点击“确定”按钮确认，这样就建立起了两台 PLC 之间的 FDL 连接。注意在设置双方的站地址和 LSAP 时（见图 4-41 和图 4-42），应保证它们之间的对应关系，否则建立不起通信连接。

通信双方用上述方法建立起 FDL 连接后，发送数据时它们将站地址和 LSAP 提供给对方。同一物理网络中的站地址和 LSAP 应统一安排，不能重叠。

连接组态完成后，点击工具栏上的按钮，编译并保存网络组态。

2．FDL 通信的编程与验证

在通信双方的 OB1 中调用通信功能 FC 6“AG_RECV”，在 OB35 中调用 FC 5“AG_SEND”来实现 FDL 服务。变量表、通信程序的编程方法和调试方法与项目 FDL_1 中的基本上相同。

4.6.2　不同项目的 FDL 通信

有的大型控制系统由两个或多个设计人员在不同的项目中进行设计，同一个 PROFIBUS 物理子网络中的主站可以在不同的项目中。各项目的硬件组态信息和程序没有公开，具有很好的保密性。双方只需要提供自己项目中的 PROFIBUS 站地址、LSAP、传输的数据长度，就可以用 FDL 连接来实现不同的项目主站之间的通信。

本节介绍的是指定连接伙伴的 FDL 连接，通过对连接的组态，来指定唯一的连接伙伴。

1．硬件组态

在 SIMATIC 管理器中，用新建项目向导创建一个新的项目，项目名称为“FDL_Pro1”（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。在 HW Config 中，将电源模块、信号模块插入机架。CPU 的 MPI 接口和集成 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 342-5 插入机架，设置其 DP 和 MPI 站地址均为 3。生成一个 PROFIBUS 网络，采用的传输速率为默认的 1.5 Mbit/s，配置文件为“标准”，将 CP 342-5 连接到网络上。在“工作模式”选项卡中，设置 CP 的工作模式为“无 DP”。

组态好硬件后，点击工具栏上的按钮，编译并保存组态信息，然后关闭 HW Config。

点击工具栏上的按钮，打开网络组态工具 NetPro，可以看到网络上只有一个站（见图 4-44）。选中 CPU 315-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表的第一行，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”中的连接对象设为“未指定”，连接类型为“FDL 连接”。

点击“确定”按钮，在自动打开的连接属性对话框的“地址”选项卡中（见图 4-45），本站的站地址为 3，设置本站的 LSAP（连接服务访问点）为 18，远程站的地址为 5，LSAP 为 10。组态完成后点击工具栏上的按钮，编译并保存网络组态。

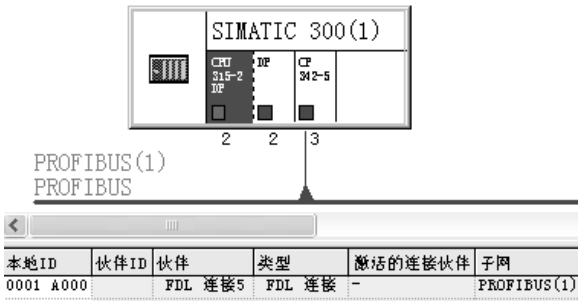


图 4-44　项目 FDL_Pro1 中的网络

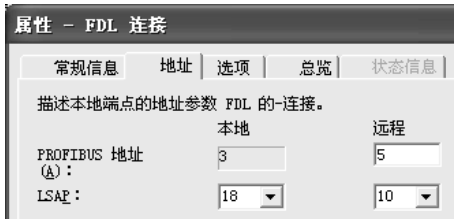


图 4-45　3 号站的 FDL 连接的地址信息

在 SIMATIC 管理器中新建一个名为“FDL_Pro2”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP，设置 CPU 的 MPI 和 DP 接口的地址均为 4（见图 4-46）。将 CP 342-5 插入机架，设置其 DP 接口地址和 MPI 地址均为 5，生成一个 PROFIBUS 网络，传输速率默认为 1.5Mbit/s，配置文件为“标准”，将 CP 342-5 连接到网络上，设置 CP 的工作模式为“无 DP”。

点击工具栏上的按钮，编译并保存组态信息。然后关闭 HW Config，打开网络组态工具 NetPro，选中 CPU 315-2DP 所在的小方框，在连接表中生成一个新的 FDL 连接，连接对象为“未指定”。

双方的连接属性对话框的“地址”选项卡中的站地址和 LSAP 见图 4-47。因为是在同一个物理网络上，两个项目中的站地址和 LSAP 不能重叠。

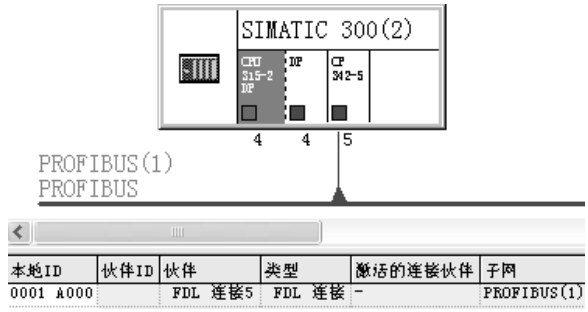


图 4-46　项目 FDL_Pro2 中的网络

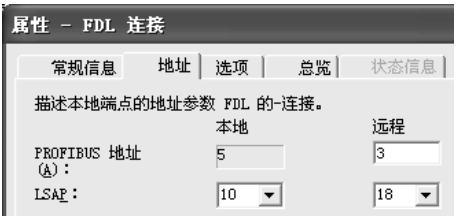


图 4-47　5 号站的 FDL 连接的地址信息

2．FDL 通信的编程与调试

在通信双方的 OB1 中调用接收数据的 FC 6“AG_RECV”，在 OB35 中调用发送数据的 FC 5“AG_SEND”。变量表、通信程序的编程方法和调试方法与项目 FDL_1 中的基本上相同。

4.7　其他 FDL 通信方式的组态与编程

4.7.1　自由第二层 FDL 通信

1．自由第二层 FDL 通信

自由第二层 FDL 通信在组态期间不指定连接伙伴的地址，不是在两个主站之间建立固定

的连接，而是通过用户程序中 4B 的报文头（Job Header），指定同一物理 PROFIBUS 网络上某个主站的站地址和 LSAP（连接服务访问点），进行数据发送。这种方式比较灵活，通过修改报文头就可以改变数据的接收方，最多可以访问 126 个支持 FDL 的站点。连接伙伴既可以在 STEP 7 项目之内，也可以在 STEP 7 项目之外。

FDL 服务允许发送和接收最多 240B 的数据，如果使用自由第二层、广播方式和多点传送方式，由于报文头的原因，最大的数据传输量为 236B。

2．组态硬件

打开 SIMATIC 管理器，用新建项目向导建立一个新的项目，CPU 为 CPU 315-2DP，项目名称为“FDLfree2”（见随书光盘中的同名例程）。

在 HW Config 中，将电源模块、信号模块和 CP 342-5 插入机架。采用默认的设置，CPU 的 MPI 接口和集成 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 342-5 插入机架，设置其 DP 站地址和 MPI 地址均为 3。生成一个 PROFIBUS 网络，设置网络的传输速率为 1.5 Mbit/s，配置文件为“标准”，将 CP 342-5 连接到网络上。在“工作模式”选项卡，设置 CP 的工作模式为“无 DP”。

在 SIMATIC 管理器中生成另一个 S7-300 站。在 HW Config 中，将 CPU 315-2DP 和电源模块、信号模块插入机架，设置 CPU 的 MPI 地址和集成的 DP 接口的地址均为 4。将 CP 342-5 插入机架，设置其 DP 地址和 MPI 地址均为 5。将 CP 342-5 连接到网络上，设置 CP 的工作模式为“无 DP”。组态结束后，点击工具栏上的按钮，编译并保存两个站的组态信息。

3．组态 FDL 连接

组态好两个 S7-300 站后，关闭 HW Config，点击 SIMATIC 管理器工具栏上的按钮，打开网络组态工具 NetPro，看到连接到 PROFIBUS 网络上的两个站（见图 4–48）。

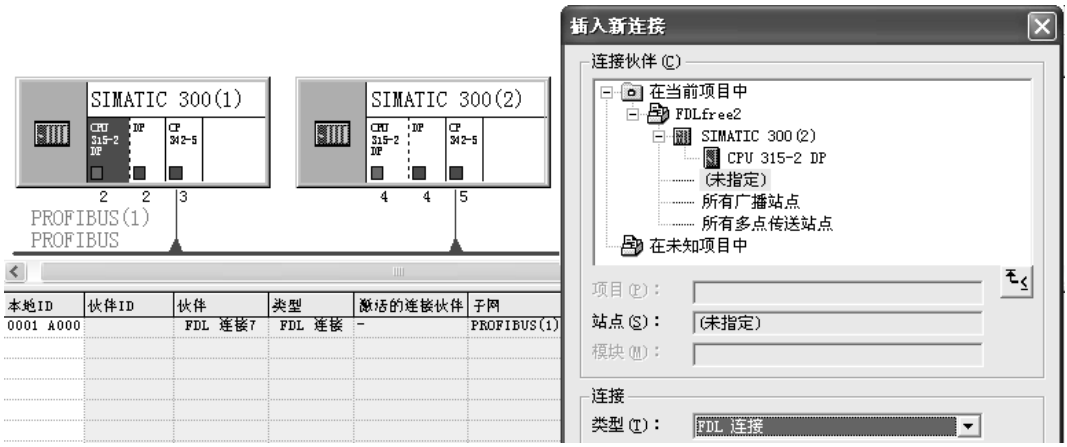


图 4–48　网络与连接组态

选中图 4–48 中 CPU 315-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框（见图 4–48 的右图）中，将“连接伙伴”中的连接对象设为“未指定”，连接类型设为“FDL 连接”。

点击“确定”按钮，在出现的 FDL 连接属性对话框的“地址”选项卡中，选中复选框“空闲第二层访问”（见图 4–49）。设置本地站的 LSAP 的值为 18，远程站的地址和 LSAP 在报文

头中给出，不能在组态时设置。

选中图 4-48 的站点 SIMATIC 300（2）中的 CPU 315-2DP，双击下面的连接表中的“FDL 连接”，在出现的 FDL 连接属性对话框的“地址”选项卡中，选中复选框“空闲第二层访问”（见图 4-50）。设置本地站的 LSAP 的值为 10。

图 4-49　3 号站 FDL 连接的地址信息

图 4-50　5 号站 FDL 连接的地址信息

4．自由第二层 FDL 通信的编程

自由第二层 FDL 通信仍然通过调用通信功能块 AG_SEND 和 AG_RECV 来实现。

在自由第二层方式下，用发送数据区和接收数据区的前 4 个字节作为报文头。报文头的第 1、2 个字节分别是通信伙伴的站地址（0～126）和连接服务访问点 LSAP（1～62），第 3 个字节是发送数据的方式，00H 为 SDA（Send Data with Acknowledge，带确认的数据发送），01H 为 SDN（Send Data with no Acknowledge，无确认的数据发送）。第 4 个字节未用，从第 5 个字节开始，才是实际接收和发送的数据。

自由第二层 FDL 通信方式报文头的第 3 个字节如果设置为带确认的数据发送方式（SDA），只有收到接收方的应答信号才能建立起通信连接，因此其安全级别较高。

因为参数 ACT 一直为 1（true），为了避免发送过于频繁，将发送程序放在循环周期为 100ms 的中断组织块 OB35 中。在 FDL 连接属性对话框的“地址”选项卡中（见图 4-49 和图 4-50），可以找到通信伙伴 CP 342-5 的站地址和 LSAP。下面是站点“SIMATIC 300（1）”OB35 中的程序：

```
程序段 1：
        L     DB1.DBW     4          //每 100ms 将 DB1.DBW4 加 1
        +     1
        T     DB1.DBW     4
        L     ID      0
        T     DB1.DBD   236          //用本站的 ID0 控制通信伙伴的 QD4
程序段 2：发送数据
        L     5                      //接收方的 CP 342-5 的站号
        T     DB1.DBB     0
        L     10                     //接收方的 CP 342-5 的 LSAP
        T     DB1.DBB     1
        L     B#16#0                 //带确认的数据发送方式
        T     DB1.DBB     2
        CALL  "AG_SEND"              //调用 FC 5
          ACT     :=TRUE             //为 1 状态时发送
```

```
      ID      :=1                          //组态时指定的连接号
      LADDR   :=W#16#140                   //组态时指定的 CP 342-5 的起始地址
      SEND    :=P#DB1.DBX0.0 BYTE 240      //存放要发送的数据的地址区
      LEN     :=240                        //发送的数据字节数
      DONE    :=M10.1                      //发送成功时产生一个脉冲
      ERROR   :=M10.2                      //错误标志字
      STATUS  :=MW14                       //状态字
```

下面是站点“SIMATIC 300（1）”的 OB1 中的接收程序。实际的系统中，可能有多个站向该站发送数据，接收到的前两个字节的数据分别是发送方的站地址和 LSAP，根据它们可以判别是哪个站发送的，然后将接收到的数据分别保存到不同的数据区中。

程序段 1：

```
      CALL    "AG_RECV"                    //调用 FC 6
      ID      :=1                          //组态时指定的连接号
      LADDR   :=W#16#140                   //组态时指定的 CP 342-5 的起始地址
      RECV    :=P#DB2.DBX0.0 BYTE 240      //存放接收的数据的地址区
      NDR     :=M0.1                       //接收到新数据时产生一个脉冲
      ERROR   :=M0.2                       //错误标志字
      STATUS  :=MW2                        //状态字
      LEN     :=MW4                        //接收的数据字节数
```

程序段 2：

```
      A       M       0.1
      JCN     LOOP                         //未接收到新数据时跳转
      L       5
      L       DB2.DBB     0
      ==I
      =       M       1.0                  //发送方的站地址为 5 时 M1.0 ON
      L       10
      L       DB2.DBB     1
      ==I
      =       M       1.1                  //发送方的 LSAP 为 10 时 M1.1 ON
      A       M       1.0
      A       M       1.1
      JCN     LOOP                         //不是 5 号站发送的数据则跳转
      CALL    "BLKMOV"                     //保存从 DBB4 开始的 236B 数据
      SRCBLK  :=P#DB2.DBX4.0 BYTE 236      //源数据区
      RET_VAL :=MW6
      DSTBLK  :=P#DB3.DBX0.0 BYTE 236      //目的数据区
      L       DB2.DBD   236
      T       QD        4                  //用通信伙伴的 ID0 控制本站的 QD4
LOOP: NOP   0
```

在初始化程序 OB100 中，调用 SFC 21，将 DB 1 的数据发送区中的各个字预置为 16#1111。将 DB 2 的数据接收区中的各字节清零。

两台 CPU 315-2DP 的程序基本上相同，其区别在于站点“SIMATIC 300（2）”的程序中

CP 地址为 W#16#150，通信伙伴的站地址为 3，LSAP 为 18。在它的 OB100 中，调用 SFC 21 将 DB 1 的数据发送区中的各个字预置为 16#2222。

图 4-51 和图 4-52 是在运行时复制的通信双方的变量表。图 4-51 中的 DB2.DBD0 是 3 号站接收到的报文头，可以看出，发送方的站地址（第 1 个字节）为 5，LSAP（第 2 个字节）为 10（16#A），第 3 个字节 16#00 表示是带确认的数据发送（SDA），第 4 个字节 16#FF 没有什么意义。

图 4-51　SIMATIC 300（1）的变量表

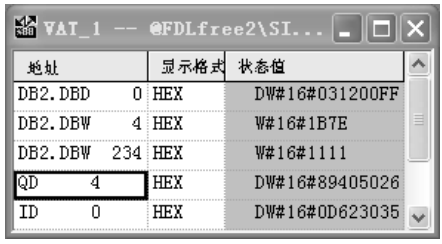


图 4-52　SIMATIC 300（2）的变量表

4.7.2　广播方式的 FDL 通信

1．FDL 广播通信方式

在 FDL 广播通信方式，某个主站通过一个虚拟站向网络中所有其他的 FDL 站发送数据。主站发送数据到虚拟站，其他站从虚拟站接收数据。广播方式采用 SDN（无确认的数据发送）方式，其他站的接收是无条件的。虚拟站的地址固定为 127，LSAP 固定为 63，不可更改。

调用 AG_SEND 发送数据时指定的发送数据区，必须保留最前面 4 个字节组成的报文头，其内容是什么没有关系。发送时可以使用的 LSAP 的范围为 1~56，网络中每一个参与 FDL 通信的主站都可以用广播方式发送数据。

接收方收到的报文头的前两个字节是发送站的站地址和 LSAP，第 3 个字节是 7FH，表示使用的是 SDN（无确认的数据发送）方式，以上数据是发送时 CPU 自动添加的。如果可能有多个站点用广播方式发送数据，接收方应根据前两个字节判断是哪个主站发送的，然后将数据存放到不同的地址区。从第 5 个字节开始为实际发送和接收的数据，由于报文头的原因，最大的数据传输量是 236B。

2．硬件组态

打开 SIMATIC 管理器，用新建项目向导建立一个新的项目，CPU 为 CPU 315-2DP，项目名称为“FDLbroad”（见随书光盘中的同名例程）。

在 HW Config 中，将电源模块、信号模块和 CP 342-5 插入机架。采用默认的设置，CPU 的 MPI 接口和集成 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 342-5 插入机架，设置其 DP 和 MPI 接口的站地址均为 3。生成一个 PROFIBUS 网络，设置网络的传输速率为 1.5 Mbit/s，配置文件为“标准”，将 CP 342-5 连接到网络上。在“工作模式”选项卡，设置 CP 的工作模式为“无 DP”，模块的起始字节地址为默认的 320（或十六进制数 16#140）。

在 SIMATIC 管理器中生成另一个 S7-300 站。在 HW Config 中，将 CPU 315-2DP 和电源模块、信号模块插入机架，设置 CPU 的 MPI 接口和集成的 DP 接口的站地址均为 4。将

CP 342-5 插入机架，设置其 DP 接口和 MPI 接口的站地址均为 5，将 CP 342-5 连接到网络上。设置 CP 的工作模式为“无 DP”，模块输入、输出的起始字节地址为默认的 336（即十六进制数 16#150）。

3．组态 FDL 连接

组态好两个站的硬件后，分别保存并编译硬件组态信息，关闭 HW Config。在 SIMATIC 管理器中点击工具栏上的按钮，打开网络组态工具 NetPro，看到连接到 PROFIBUS 网络上的两个站（见图 4-53）。

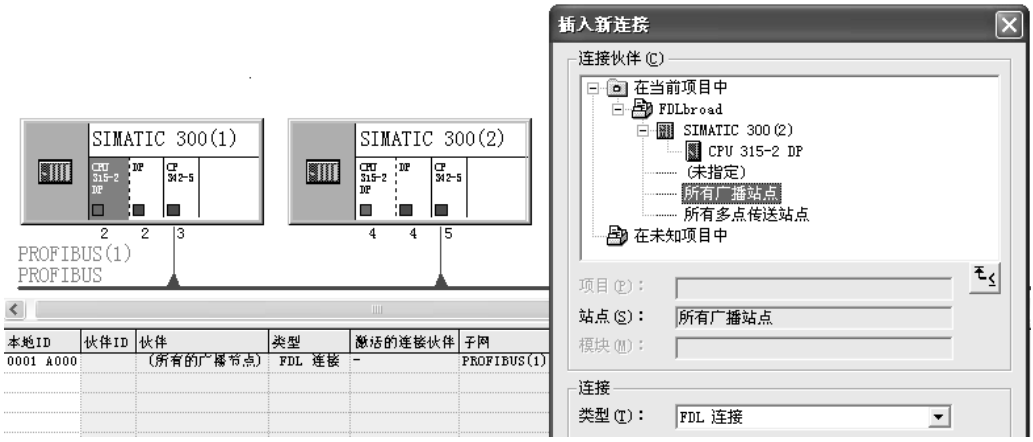


图 4-53　网络与 FDL 连接组态

选中 SIMATIC 300（1）站，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”中的连接对象设为“所有广播站点”，连接类型为“FDL 连接”。

点击“确定”按钮，在出现的 FDL 连接属性对话框的“常规信息”选项卡中，块参数“标识号”（ID）为 1，模块起始地址 LADDR 为#16#140。在“地址”选项卡中（见图 4-54），可以看到系统自动指定的远程虚拟站的站地址为 127，LSAP 为 63（不能修改）。设置本地站的 LSAP 为 18。

选中 SIMATIC 300（2）站，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”中的连接对象设为“所有广播站点”，连接类型为“FDL 连接”。

点击“确定”按钮，在出现的 FDL 连接属性对话框的“地址”选项卡中（见图 4-55），系统指定的远程虚拟站的站地址为 127，LSAP 为 63。设置本地站的 LSAP 为 10。

图 4-54　3 号站的 FDL 连接的地址信息

图 4-55　5 号站的 FDL 连接的地址信息

广播方式需要调用通信功能 AG_SEND 和 AG_RECV 来发送和接收数据，它们的编程与 FDL 自由第二层的基本上相同，发送时虽然未使用报文头的 4B 数据，但是要保留它们的位置，所以最多只能发送 236B 数据。

4．通信程序

下面是 SIMATIC 300（1）站的 OB35 中的发送程序：

程序段 1：

```
L     DB1.DBW    4                    //每 100ms 将 DB1.DBW4 加 1
+     1
T     DB1.DBW    4
```

程序段 2：发送数据

```
L     ID      0
T     DB1.DBD   236                    //用本站的 ID0 控制通信伙伴的 QD4
CALL  "AG_SEND"                        //调用 FC 5
  ACT     :=TRUE                       //为 1 时发送
  ID      :=1                          //组态时指定的连接号
  LADDR   :=W#16#140                   //组态时指定的 CP 的起始地址
  SEND    :=P#DB1.DBX0.0 BYTE 240      //存放要发送的数据的地址区
  LEN     :=240                        //发送的字节数
  DONE    :=M10.1                      //发送成功时产生一个脉冲
  ERROR   :=M10.2                      //错误标志字
  STATUS  :=MW14                       //状态字
```

OB1 中的接收程序调用 FC 6 接收到数据后，需要根据报文头，判断是哪个站发送的数据，然后将它们保存到不同的数据区。接收程序与项目“FDLfree2”的相同。

在站点 SIMATIC 300（1）的初始化程序 OB100 中，用 SFC 21 将 DB 1 的数据发送区中的各个字预置为 16#1111，将 DB 2 和 DB 3 的数据接收区中的各字节清零。

两台 CPU 315-2DP 的程序基本上相同，只是站点“SIMATIC 300（2）”的程序中的 CP 地址为 W#16#150，通信伙伴的站地址为 3，LSAP 为 18。在它的 OB100 中，用 SFC 21 将 DB 1 的数据发送区的各个字预置为 16#2222。

图 4-56 和图 4-57 是在运行时复制的通信双方的变量表。图 4-56 中的 DB2.DBD0 是 3 号站接收到的报文头，第一个字节是发送方的站地址 5，第 2 个字节 LSAP 为 10（16#A），第 3 个字节 16#7F 表示使用的是 SDN（无确认的数据发送）方式，第 4 个字节 16#FF 没有什么意义。

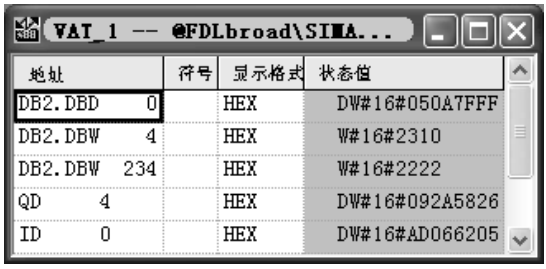


图 4-56　SIMATIC 300（1）的变量表

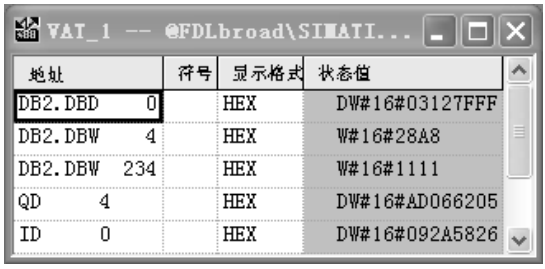


图 4-57　SIMATIC 300（2）的变量表

4.7.3 多点传送方式的 FDL 通信

1．FDL 多点传送方式

FDL 多点传送是一种分组广播方式，网络中具有相同的 LSAP（1~56）的主站为同一组，一个站可以同时向该组中的其他站发送数据。多点传送方式和广播方式均采用 SDN（无确认的数据发送）的服务方式。

多点传送方式的 FDL 发送站将数据发送到一个 DP 地址固定为 127 的虚拟站，同一多点传送组内其他的站通过虚拟站接收数据。同一组内所有的站具有相同的 LSAP（1～56），组态时可以通过设置本地 LSAP 的值来修改本组公用的 LSAP，它被自动作为同一组内各站的远程 LSAP（见图 4-59 和图 4-60）。

在同一物理网络中，可以有其他多点传送组，不同的组应使用不同的 LSAP。

与广播方式类似，在多点传送方式，发送数据区的前 4 个字节作为报文头，必须保留，其内容是什么没有关系。从第 5 个字节开始为实际接收和发送的数据。同一组内每个参与 FDL 通信的站都可以用多点传送方式发送数据。

接收方收到的报文头的前两个字节是发送站的站地址和 LSAP，第 3 个字节是 7FH，用来表示使用的是 SDN（无确认的数据发送）方式，以上数据是发送时 CPU 自动添加的。接收方根据前两个字节判断是哪个主站发送的，然后将数据存放到不同的地址区。从第 5 个字节开始为实际发送和接收的数据，由于报文头的原因，最大的数据传输量为 236B。

2．硬件组态

打开 SIMATIC 管理器，用新建项目向导创建一个新的项目，CPU 为 CPU 315-2DP，项目名称为“FDL_mul”（见随书光盘中的同名例程）。

在 HW Config 中，将电源模块、信号模块和 CP 342-5 插入机架。采用默认的设置，CPU 的 MPI 接口和集成 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 342-5 插入机架，设置其 DP 和 MPI 接口的站地址均为 3。生成一个 PROFIBUS 网络，设置网络的传输速率为 1.5 Mbit/s，配置文件为“标准”，将 CP 342-5 连接到网络上。在“工作模式”选项卡，设置 CP 的工作模式为“无 DP”，模块输入、输出的起始字节地址为默认的 320（即十六进制数 16#140）。

在 SIMATIC 管理器中生成另一个 S7-300 站。在 HW Config 中，将 CPU 315-2DP 和电源模块、信号模块插入机架，设置 CPU 的 MPI 接口和集成的 DP 接口的站地址均为 4。插入一块 CP 342-5，设置其 DP 接口和 MPI 接口的站地址均为 5，将 CP 342-5 连接到网络上。设置 CP 的工作模式为“无 DP”，模块输入、输出的起始字节地址为默认的 336（16#150）。

3．组态 FDL 连接

组态好两个站的硬件后，分别保存并编译硬件组态信息后，关闭 HW Config。在 SIMATIC 管理器中点击工具栏上的  按钮，打开网络组态工具 NetPro，看到连接到 PROFIBUS 网络上的两个站（见图 4-58）。选中 SIMATIC 300（1）站，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”中的连接对象设为“所有多点传送站点”，连接类型为“FDL 连接”。

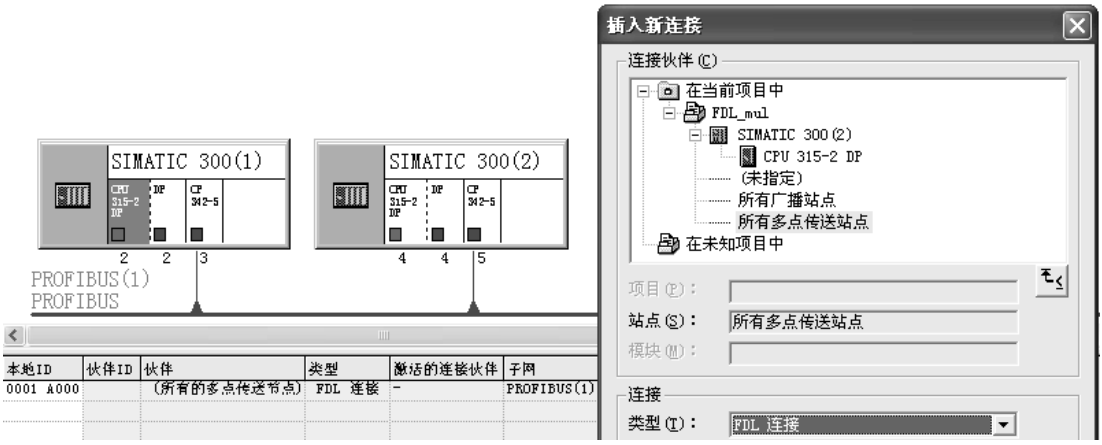


图 4-58　网络与 FDL 连接组态

点击“确定”按钮，在出现的 FDL 连接属性对话框的“地址”选项卡中（见图 4-59），可以看到系统自动指定的远程虚拟站的站地址为 127。设置本地站的 LSAP 为 18，它被自动作为该站的远程 LSAP。改变本地 LSAP，远程 LSAP 也随之而变，LSAP 相同的站点为同一组，同组的站点之间进行广播通信。

选中 SIMATIC 300（2）站，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新的连接。在弹出的“插入新连接”对话框中，将“连接伙伴”中的连接对象设为“所有多点传送节点”，连接类型为“FDL 连接”。

点击“确定”按钮，在出现的 FDL 连接属性对话框的“地址”选项卡中（见图 4-60），设置本地的 LSAP 为 18，系统自动指定的远程虚拟站的站地址为 127，LSAP 为 18。上述两个站因为具有相同的 LSAP，它们属于同一个多点传送组。

图 4-59　3 号站的 FDL 连接的地址信息

图 4-60　5 号站的 FDL 连接的地址信息

多点传送方式需要调用通信功能 AG_SEND 和 AG_RECV 来发送和接收数据，具体的程序与广播方式的基本上相同，注意双方 OB1 中的 LSAP 均为 18。

图 4-61 和图 4-62 是在运行时复制的通信双方的变量表。图 4-62 中的 DB2.DBD0 是 5 号站接收到的报文头，第 1 个字节是发送方的站地址 3，第 2 个字节 LSAP 为 18（16#12），第 3 个字节 16#7F 表示使用的是 SDN（无确认的数据发送）方式，第 4 个字节 16#FF 没有什么意义。

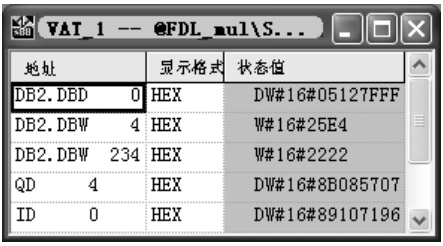


图 4-61　SIMATIC 300（1）的变量表

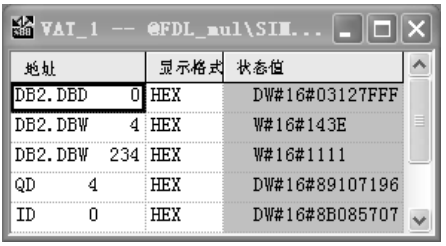


图 4-62　SIMATIC 300（2）的变量表

4.8　练习题

1．什么是连接？什么情况需要建立连接？

2．静态连接有什么特点？

3．简述客户机和服务器在通信中的作用。

4．哪些模块在 PROFIBUS S7 通信中可以作客户机？

5．组态一个项目，CPU 412-2DP 作 S7 通信的客户机，CPU 315-2DP 作服务器，编写通信程序。

6．组态一个项目，实现两台 CPU 412-2DP 之间的 S7 双向通信，编写调用 SFB 12/13 的通信程序。

7．组态一个项目，实现两台 CPU 314C-2DP 之间的 FDL 通信，编写通信程序。

8．组态一个项目，实现两台 CPU 313C-2DP 之间的 FDL 自由第二层通信，编写通信程序。

9．组态一个项目，实现 3 台 CPU 313C-2DP 之间的 FDL 广播方式的通信，编写通信程序。

第 5 章　PROFIBUS-DP 通信的其他应用

5.1　直接数据交换通信及其组态

5.1.1　直接数据交换通信

直接数据交换（Direct Data Exchange、DX）又称为交叉通信，主要用于智能从站接收 DP 从站的数据，和多主站系统的从站发送数据到其他主站。在选型时应注意某些 CPU 没有直接数据交换功能。

直接数据交换通信采用广播式通信方式，从站作为生产者（Publisher），可以不经过主站直接将信息发送给作为消费者（Subscribers）的从站（见图 5-1）。这样从站可以直接接收别的从站发送的数据。

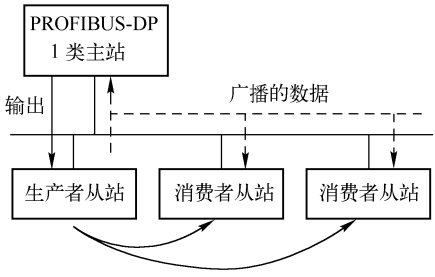


图 5-1　从站与从站的直接数据交换

实现直接数据交换通信需要下列条件：

1）从站将数据发送给主站。

2）作为消费者的从站必须是有 CPU 的智能从站。

3）参与直接数据交换的站点必须有直接数据交换通信功能。

下面是直接数据交换的几种应用场合：

（1）单主站系统中 DP 从站发送数据到智能从站

通过直接数据交换，DP 从站发送的数据可以被同一个 PROFIBUS-DP 子网的智能从站接收。所有具有直接数据发送功能的 DP 从站（包括非智能从站），都能提供用于 DP 从站之间的直接数据交换的数据，只有主站和智能 DP 从站才能接收这些数据。

（2）多主站系统中从站发送数据到其他主站

同一个物理 PROFIBUS-DP 子网中有几个 DP 主站的系统称为多主站系统（见图 5-2）。有直接数据交换功能的 DP 从站发送的数据，可以被同一个物理 DP 子网中其他 DP 主站系统的主站直接读取。

（3）从站发送数据到其他主站系统的智能从站

在多主站系统中，有直接数据发送功能的 DP 从站发送的数据，可以被同一物理 DP 网络

中其他主站系统的智能从站读取。

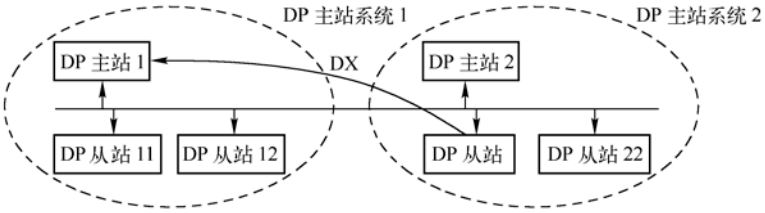


图 5-2　多主站系统中从站发送数据到其他主站

5.1.2　直接数据交换通信的组态

1．通信要求

用 STEP 7 的新建项目向导创建一个名为“DX_1”的项目，主站（2 号站）的 CPU 为 CPU 412-2DP。两个从站的 CPU 均为 CPU 315-2DP，站地址分别为 3 和 4。本章的例程在随书光盘的文件夹“\Project\PB_Others”中。

通信要求如下（见图 5-3）：

1）2 号主站发送连续的 10B 数据到 3 号从站，接收来自 3 号从站连续的 20B 数据。

2）4 号从站发送连续的 10B 数据到 2 号主站，接收来自主站的连续的 10B 数据。

3）4 号从站通过直接数据交换功能接收 3 号从站发送给主站的数据中的后 10B 数据。

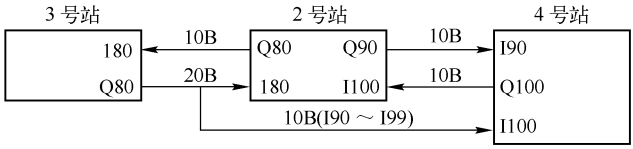


图 5-3　直接数据交换通信示意图

2．组态 DP 主站

选中 SIMATIC 管理器中的“SIMATIC 400（1）”对象，双击右边窗口中的“硬件”图标，进入 HW Config 窗口，可以看到自动生成的机架和 CPU 模块。将电源模块插入 1 号槽，从 8 号槽开始插入信号模块。

双击 CPU 下面“DP”所在的行（见图 5-4），在出现的 PROFIBUS 接口属性对话框中，点击“新建”按钮，采用默认的参数设置，站地址为 2，传输速率为 1.5 Mbit/s，配置文件为 DP。多次点击“确定”按钮，返回硬件组态窗口，在 CPU 412-2DP 的机架右侧出现 PROFIBUS-DP（1）主站系统的网络线。此时图中还没有两个从站。点击按钮，保存组态信息，关闭 HW Config。

3．组态智能从站

用鼠标右键点击 SIMATIC 管理器左侧窗口最上面的项目，在打开的快捷菜单中执行命令“插入新对象”→“SIMATIC 300 站点”。选中生成的站点，双击右边窗口的“硬件”图标，打开 HW Config，对该站的硬件组态。生成该站的机架后，将 CPU 315-2DP 插入 2 号槽，电源模块插入 1 号槽。从 4 号槽开始插入信号模块。CPU 的订货号为 6ES7 315-2AG10-0AB0。

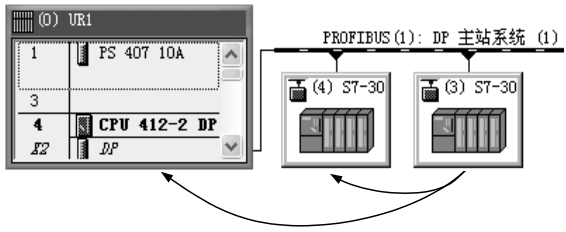


图 5-4　HW Config 中的 DP 网络

参与直接数据通信的主站和从站均应有直接数据通信功能。较低档的 CPU 315-2DP 没有直接数据交换功能。选中硬件目录中的某个 CPU 模块后，可以在下面的小窗口看到该模块是否有直接数据交换功能。

插入 CPU 315-2DP 时，在自动打开的 PROFIBUS 接口属性对话框中，将 DP 站地址设为 3，不连网。点击“确定”按钮，返回 HW Config。双击 CPU 中标有 DP 的行，在打开的对话框的“工作模式”选项卡中，将该站设置为 DP 从站，点击按钮，保存组态信息，关闭 HW Config。这时如果点击工具栏上的按钮（编译与保存），因为没有完成全部组态任务，不能成功地编译组态信息。

双击机架中“CPU 315-2 DP”所在的行，在出现的 CPU 属性对话框中，点击“属性”按钮。在出现的 MPI 接口属性对话框中设置 MPI 站地址为 3。

用同样的方法生成另一个 DP 从站，CPU 的型号与 3 号从站相同。设置该站的 DP 站地址和 MPI 站地址均为 4，不连网，工作模式为 DP 从站。点击按钮，保存组态信息后关闭 HW Config。

4．将智能从站连接到 DP 网络上

在 DP 主站的硬件组态窗口中，打开右边的硬件目录窗口的“\PROFIBUS-DP \Configured Stations”（已组态的站）文件夹，将图标“CPU 31x”拖放到左边窗口中的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡被自动打开。选中列表框中的某个从站，点击“连接”按钮，该站被连接到 DP 网络上。用同样的方法将两个从站连接到 DP 网络上。图 5-5 是组态好后在网络组态工具 NetPro 中看到的网络连接图。

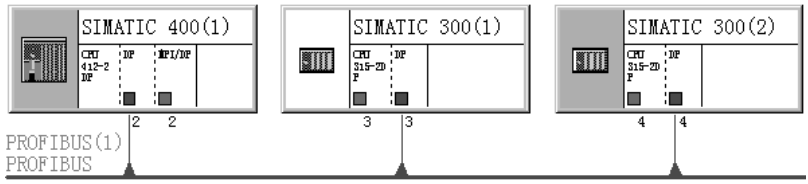


图 5-5　NetPro 中的 DP 网络

5．组态发送站的地址区

在主站的硬件组态窗口中，双击 3 号从站的图标（见图 5-4），或双击 NetPro 中 3 号从站的“DP”所在的方框（见图 5-5）。在出现的“DP 从站属性”对话框的“组态”选项卡中，点击“新建”按钮，在出现的对话框中设置 DP 主站和 3 号从站用于通信的输入/输出区的地址。

由图 5-6 可知，DP 主站（2 号站）和智能从站之间通过 IW80 和 QB80 开始的数据区交换数据。设置好后点击“确定”按钮，返回主站的 HW Config 窗口。

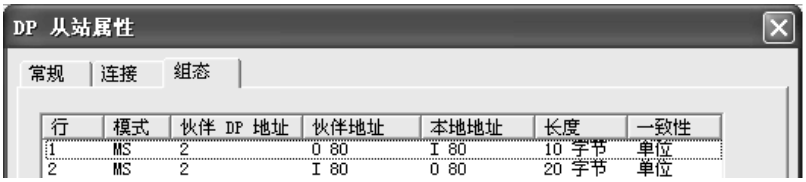


图 5-6　组态 3 号从站的通信 I/O 区

6．组态接收站的地址区

双击图 5-4 中的 4 号从站的图标，在打开的“DP 从站属性”对话框的“组态”选项卡中，按图 5-7 第 1 行和第 2 行的要求，设置 4 号从站与主站之间的主从（MS）通信的过程映像输入/输出区地址。

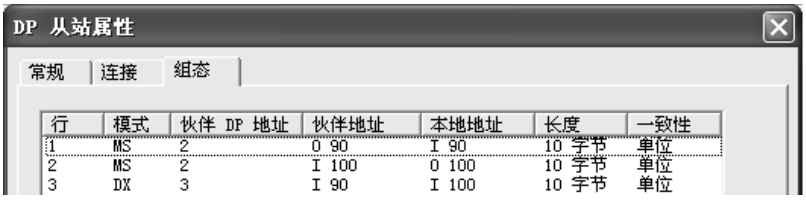


图 5-7　组态 4 号从站的通信 I/O 区

点击“新建”按钮，出现设置 DP 从站输入/输出区地址的对话框（见图 5-8），设置图 5-7 第 3 行的参数。用最上面的“模式”选择框选中“DX”模式，使 4 号从站通过直接数据交换，接收 3 号从站发送给主站的数据中的后 10 个字节。DP 伙伴被自动指定为发送数据的 3 号从站，但是自动生成的 DP 伙伴的地址 IB80 是主站接收 3 号从站的数据的起始地址。组态时将它修改为 IB90，该地址区应在主站（2 号站）接收 3 号从站发送的数据的 IB80～IB99 的范围之内。可以理解为 3 号从站向主站发送数据时，4 号从站“偷听”主站接收到的部分数据（见图 5-3）。

图 5-8　直接数据交换的参数设置

设置好后点击“确定”按钮，返回主站的 HW Config 窗口。点击工具栏上的（编译与保存）按钮。因为已经完成了全部组态任务，可以成功地编译组态信息。

7．编写验证通信的程序

在主站的 OB100，将发送给 3 号从站和 4 号从站的输出区的字初始化为 W#16#2222，将接收数据的输入区清零。在 3 号从站的 OB100，将发送给主站和 4 号从站的输出区的字初始化为 W#16#3333，将接收数据的输入区清零。在 4 号从站的 OB100，将发送给主站的输出区的字初始化为 W#16#4444，将接收主站和 3 号从站的数据的输入区清零。

具体的程序见随书光盘中的项目 PB_DX1。

8．通信的验证

用电缆连接 3 块 CPU 集成的 MPI 通信接口和计算机的 CP 5611 的 MPI 接口，将用户程序和组态信息分别下载到 3 台 PLC。用电缆连接 3 块 CPU 集成的 DP 接口，将各 CPU 切换到 RUN 模式。用 MPI 或 DP 网络监控系统的运行。

在 SIMATIC 管理器中创建 3 个从站的变量表（见图 5-9～图 5-11），用变量表监视各站接收到的数据区的第一个字和最后一个字。在运行时同时打开 3 个从站的变量表，将它们调节到适当的大小。点击工具栏上的按钮，使各变量表分别进入监控状态。由图 5-9～图 5-11 可知，各站都成功地接收到了通信伙伴发送给它的数据，4 号从站的 IW100～IW108 是用直接数据通信接收到的 3 号从站发送给主站的数据的后 10 个字节。

图 5-9　2 号站的变量表

图 5-10　3 号从站的变量表

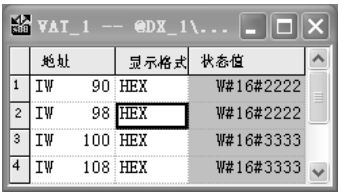


图 5-11　4 号从站的变量表

5.1.3　ET 200 发送数据给智能从站

在直接数据交换通信中，智能从站除了能接收其他智能从站发送的数据外，还可以接收具有直接数据交换发送功能的非智能从站发送给主站的数据。

1．生成主站和 DP 网络

在 STEP 7 中创建一个名为“DX_2”的项目（见随书光盘中的同名例程），CPU 为 CPU 412-2DP。选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，点击出现的对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成一条 PROFIBUS-DP 网络。采用默认的参数，CPU 412-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮，返回 HW Config。

2．组态智能从站

用鼠标右键点击 SIMATIC 管理器屏幕左边最上面的项目对象“DX_2”，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”，插入新的站。选中生成的新站后，双

击右边窗口的“硬件”图标，对该站的硬件组态。首先生成该站的机架，然后插入 CPU 315-2DP、电源模块和信号模块。

将 CPU 放到机架上时，将会自动打开 DP 接口属性对话框的“参数”选项卡。设置 PROFIBUS 站地址为 3，不连接到 PROFIBUS 网络。返回 HW Config 后，双击 CPU 中 DP 所在的行，打开 DP 属性对话框。在“工作模式”选项卡将该站设置为 DP 从站。返回 HW Config 后，双击 CPU 315-2DP 所在的行，将它的 MPI 地址设置为 3。最后点击按钮，保存对 S7-300 站的组态。

3．将智能从站连接到 DP 网络上

返回 S7-400 主站的硬件组态窗口，打开右边的硬件目录窗口中的“\PROFIBUS-DP\Configured Stations”（已组态的站）文件夹，将图标“CPU 31x”拖放到左边窗口中的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡被自动打开，选中列表框中的“CPU 315-2DP”，点击“连接”按钮，该站被连接到 DP 网络上。

4．组态模块式 DP 从站 ET 200 M

打开 HW Config 的硬件目录窗口的文件夹“\PROFIBUS-DP\ET 200M”，选中接口模块“IM 153-1”，订货号为 6ES7 153-1AA03-0XB0，在下面灰色背景的小窗口中可以看到，它具有横向通信（即直接数据通信）的发送功能（见图 5-12）。将它拖放到 PROFIBUS 网络线上，便生成了 ET 200M 从站。在出现的“属性 – PROFIBUS 接口 IM 153-1”对话框中，设置它的站地址为 4。某些低版本的 IM 153 没有直接数据通信的发送功能。

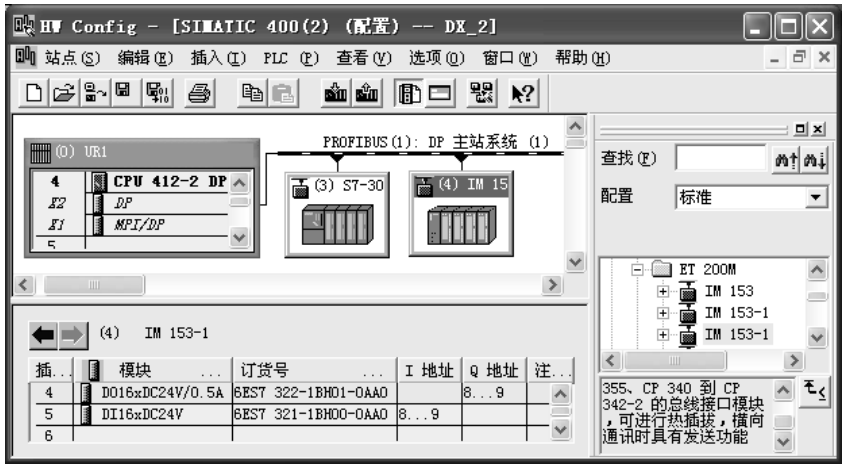


图 5-12　组态 DP 网络

选中 DP 网络上的该从站，打开硬件目录窗口中的“IM 153-1”子文件夹，将 16 点数字量输出、16 点数字量输入模块拖放到左下面窗口的 4、5 号槽（见图 5-12），它们分别占用 QW8 和 IW8。

5．组态智能从站与主站的通信地址区

双击图 5-12 中的智能从站（3 号从站）的图标，打开“DP 从站属性”对话框（见图 5-13）。点击“组态”选项卡的“新建”按钮，在出现的对话框中设置主站与 DP 从站用于交换数据的输入/输出区的地址。选择通信模式为 MS（主从），通信伙伴为它的主站（2 号站）。组态表

的第 1、2 行用于智能从站与主站之间的主从通信。由图 5-13 可知，DP 主站（2 号站）和智能从站之间通过 IW80～IW98 和 QB80～QB98 交换数据。

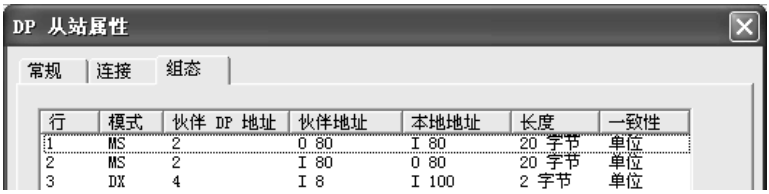


图 5-13　3 号从站的通信 I/O 区组态

6．组态从站之间的直接数据交换通信的地址区

点击组态选项卡中的“新建”按钮，在打开的“DP 从站属性 – 组态 – 行 3”对话框中（见图 5-14），选择通信模式为 DX（直接数据交换），DP 伙伴的站地址只能选 4 号从站（ET 200M），伙伴地址、长度、单位（2B）和一致性都是自动生成的。伙伴地址是主站接收 4 号从站 ET 200M 发送的数据的输入地址，相当于 4 号从站向主站发送数据时，3 号智能从站用 IW100“偷听”这些数据。点击“确定”按钮，返回“DP 从站属性”对话框。

图 5-14　直接数据交换的参数设置

设置好通信地址区后点击“确定”按钮，返回主站的 HW Config 窗口。

设置完上述参数后，点击工具栏上的按钮，编译与保存组态信息。

7．编写验证通信的程序

在主站的 OB100，将发送给 3 号从站的输出字初始化为 W#16#2222，将接收数据的输入区清零。在 3 号从站的 OB100，将发送给主站的输出字初始化为 W#16#3333，将接收数据的输入区清零。具体的程序见随书光盘中的项目 DX_2。

8．通信的验证

用电缆连接两块 CPU 集成的 MPI 通信接口和计算机的 CP 5611 的 MPI 接口，将用户程序和组态信息分别下载到两台 PLC。

用电缆连接两块 CPU 集成的 DP 接口和 IM 153-1 的 DP 接口，将各 CPU 切换到 RUN 模式。用 MPI 或 DP 网络监控系统的运行。

在 SIMATIC 管理器中分别创建 CPU 412-2DP 和 CPU 315-2DP 的变量表（见图 5-15 和图 5-16），用变量表监视它们用主从通信接收到的第一个字 IW80 和最后一个字 IW98，以及 3 号从站用直接数据交换接收 ET 200M 的数据的输入字 IW100。运行时点击工具栏上的按钮，使变量表分别进入监控状态。改变 ET 200M 的 DI 模块外接的小开关的状态，可以看到变量表中 IW100 的值随之而变。

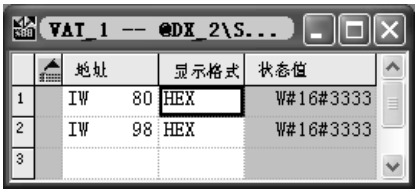


图 5-15　主站的变量表

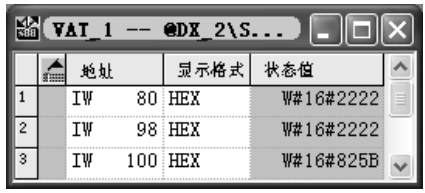


图 5-16　3 号从站的变量表

9．注意事项

1）ET 200M 的接口模块 IM 153 必须具有直接数据通信的发送功能。

2）智能从站只能接收非智能从站发送给主站的输入信号，不能接收主站发送给非智能从站的输出信号。

3）组态时使用的直接数据交换的地址 IW8 是主站用来读取 ET 200M 的输入信号的地址。

4）组态表的每一行只能组态与 ET 200M 的一块输入模块的 DX 通信。假设 ET 200M 有两块 16 点输入模块，地址分别为 IW8 和 IW10，如果组态时设置数据长度为 4B（读取 IW8 和 IW10），点击“确定”按钮时将会出现错误信息对话框。应在组态表中分两行来组态两块输入模块。

5.1.4　DP 从站发送数据到其他 DP 主站

1．组态要求

同一个 DP 网络上有两个 DP 主站（见图 5-17），3 号智能从站发送 20B 的数据给它的主站（2 号站），与此同时，另一个主站（4 号站）用直接数据交换功能接收其中的部分数据。

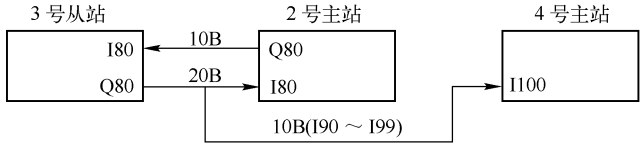


图 5-17　直接数据交换通信示意图

2．生成主站和 DP 网络

在 STEP 7 中创建一个名为“DX_3”的项目（见随书光盘中的同名例程），CPU 为 CPU 412-2DP。选中该站，双击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源

模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，打开 DP 属性对话框。点击“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成 PROFIBUS-DP 网络，点击“确定”按钮返回 HW Config。采用默认的参数，CPU 412-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。

多次点击“确定”按钮，返回硬件组态窗口，在 CPU 412-2DP 的机架右侧出现 PROFIBUS 的网络线。

3．组态智能从站

用鼠标右键点击 SIMATIC 管理器屏幕左边最上面的项目对象“DX_3”，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”，插入新的站。选中新站后，双击“硬件”图标，对该站的硬件组态。首先生成该站的机架，然后插入 CPU 315-2DP、电源模块和信号模块。

将 CPU 放到机架上时，将会自动打开 DP 接口属性对话框的“参数”选项卡。设置 PROFIBUS 站地址为 3，不连接到 PROFIBUS 网络。点击“确定”按钮返回 HW Config 后，双击 CPU 中 DP 所在的行，打开 DP 属性对话框。在“工作模式”选项卡中将该站设置为 DP 从站，点击“确定”按钮返回 HW Config。设置 CPU 315-2DP 的 MPI 地址为 3。

点击工具栏上的按钮，保存组态信息。最后关闭 HW Config。

4．将智能从站连接到 DP 网络上

返回 S7-400 主站的硬件组态窗口，打开右边的硬件目录窗口中的“\PROFIBUS-DP \Configured Stations”（已组态的站）文件夹，将图标“CPU 31x”拖放到左边窗口中的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡被自动打开，选中列表框中的 CPU 315-2DP。点击“连接”按钮，该站被连接到 DP 网络上（见图 5-18）。

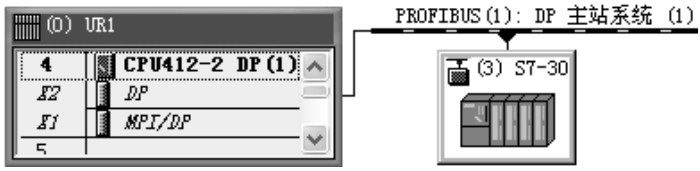


图 5-18　组态 DP 从站

5．组态智能从站的通信地址区

在主站的硬件组态窗口中，双击图 5-18 中的 3 号从站的图标，在出现的对话框的“组态”选项卡中，点击“新建”按钮。在出现的“DP 从站属性”对话框中，设置主站与 DP 从站用主从方式交换数据的输入/输出区的地址。

由图 5-19 可知，DP 主站（2 号站）和智能从站之间通过 IW80 和 QB80 开始的地址区交换数据。设置好通信地址区后，点击“确定”按钮，返回主站的 HW Config 窗口。

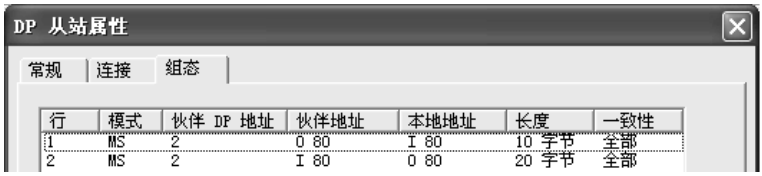


图 5-19　组态 3 号从站的通信 I/O 区

点击工具栏上的按钮，编译与保存组态信息。

6．生成和组态另一个 DP 主站

用鼠标右键点击 SIMATIC 管理器左边窗口最上面的项目对象“DX_3”，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”。选中新生成的站后，双击右边窗口的“硬件”图标，对该站的硬件组态。首先生成该站的机架，然后插入 CPU 315-2DP、电源模块和信号模块。

将 CPU 放到机架上时，将会自动打开 DP 接口属性对话框的“参数”选项卡。设置 PROFIBUS 站地址为 4，连接到 PROFIBUS 网络。点击“确定”按钮返回 HW Config。

双击 CPU 中 DP 所在的行，打开 DP 属性对话框，采用默认的工作模式（DP 主站）。在“组态”选项卡（见图 5-20）中，组态通信用的过程映像输入/输出区地址。点击“新建”按钮，出现图 5-21 中的对话框。最上面的“模式”选择框为“DX”，字符和背景色均为灰色，不可更改。

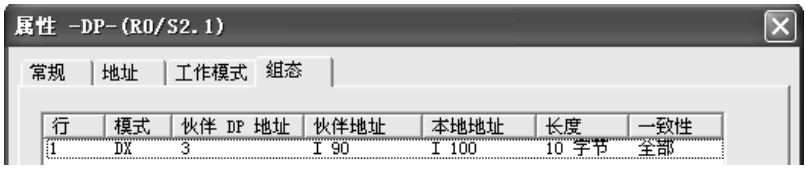


图 5-20　组态 4 号主站的通信 I/O 区

图 5-21　直接数据交换的参数设置

设置表中的参数，使 4 号主站通过直接数据交换，接收 3 号从站发送给它的主站的后 10 字节数据。值得注意的是，在 DX 通信组态中，通信伙伴是发送数据的 3 号从站，但是通信

伙伴的地址区是 2 号主站接收 3 号从站发送的数据的输入区 IB90～IB99。相当于 3 号从站向它的主站发送数据时，4 号主站用 IB100～IB109“偷听”其中的部分数据（见图 5-17）。

点击工具栏上的按钮，编译与保存组态信息。最后关闭 HW Config。

7．通信的编程

因为在组态主从方式（MS）和直接数据交换（DX）通信时，数据的一致性均为“全部”，各主站和从站均需要调用 SFC 14 和 SFC 15，对通信数据打包和解包。

下面是 2 号主站的 OB1 中的程序：

程序段 1：解开 IB80～IB99 接收到的数据包，并将数据存放在 MB10～MB29 中

```
CALL   "DPRD_DAT"                    //调用 SFC 14
  LADDR    :=W#16#50                 //接收数据的输入区的起始地址 IB80
  RET_VAL  :=MW2                     //错误代码
  RECORD   :=P#M 10.0 BYTE 20        //存放读取的数据的目的数据区
```

程序段 2：将 MB30～MB39 中的数据打包，通过 QB80～QB89 发送出去

```
CALL   "DPWR_DAT"                    //调用 SFC 15
  LADDR    :=W#16#50                 //发送数据的输出区的起始地址 QB80
  RECORD   :=P#M 30.0 BYTE 10        //存放要发送的用户数据的源数据区
  RET_VAL  :=MW4                     //错误代码
```

各主站和 3 号从站的 OB1 调用 SFC 14 和 SFC 15 的程序基本上相同。

在 2 号主站的 OB100，将发送给从站的地址区的字初始化为 W#16#2222，将接收数据的地址区清零。

在 3 号从站的 OB100，将发送给 2 号主站的 MB30~MB39 初始化为 W#16#3331，将发送给 2 号和 4 号主站的 MB40~MB49 初始化为 W#16#3332，将接收数据的输入区清零。

在 4 号主站的 OB100，将接收数据的地址区的字清零。

具体的程序见随书光盘中的项目 DX_3。

8．通信的验证

用电缆连接 3 块 CPU 的 MPI 通信接口和计算机的 CP 5611 的 MPI 接口，将用户程序和组态信息分别下载到 3 台 PLC。

用电缆连接各 CPU 集成的 DP 接口，将各 CPU 切换到 RUN 模式。用 MPI 或 DP 网络监控系统的运行。在 SIMATIC 管理器中创建 3 个站的变量表，在变量表中生成各个站接收到的数据区的第一个字和最后一个字。

图 5-22～图 5-24 是运行时 3 个站的变量表，可以看出，2 号主站和 3 号从站接收到了对方发送的数据。4 号主站接收到了 3 号从站发送给 2 号主站的数据的后半部分。

VAT_1 -- @DX_...

	地址	显示格式	状态值
1	IW 80	HEX	W#16#3331
2	IW 88	HEX	W#16#3331
3	IW 90	HEX	W#16#3332
4	IW 98	HEX	W#16#3332

图 5-22　2 号站的变量表

VAT_1 -- @DX_3\...

	地址	显示格式	状态值
1	IW 80	HEX	W#16#2222
2	IW 88	HEX	W#16#2222

图 5-23　3 号从站的变量表

VAT_1 -- @DX_3\...

	地址	显示格式	状态值
1	IW 100	HEX	W#16#3332
2	IW 108	HEX	W#16#3332

图 5-24　4 号站的变量表

5.2 PROFIBS-DP 通信的其他应用

5.2.1 智能从站触发主站的硬件中断

与本地的中央机架或扩展机架中的 I/O 一样，分布式 I/O 设备也可以产生硬件中断，或称为过程中断。在 PROFIBUS 网络中，硬件中断可以由支持中断处理的 DP 从站或由 DP 从站设备中的某个模块产生。本节介绍智能从站调用 SFC 7 产生主站的硬件中断的组态和编程的方法。

1．生成 DP 主站

在 STEP 7 中用新建项目向导创建一个项目（见随书光盘中的例程 Intrrupt），CPU 为 CPU 412-2DP。选中 SIMATIC 管理器中的“SIMATIC 400”站对象，双击右边窗口的“硬件”图标，打开 HW Config，在 CPU 412-2DP 的机架中添加电源模块和信号模块。

双击 CPU 中“DP”所在的行，在打开的 DP 属性对话框中，采用默认的参数。新建一个 DP 网络，主站的地址为 2。返回 HW Config 后，双击“DP”所在的行，在“工作模式”选项卡中，采用默认的 DP 主站模式，“DP 模式”采用默认的 DPV1（V1 版的 DP）。

2．生成智能从站

用右键点击 SIMATIC 管理器屏幕左边最上面的项目对象 Intrrupt，在打开的快捷菜单中执行命令“插入新对象”→“SIMATIC 300 站点”。选中生成的新站，双击右边窗口中的“硬件”图标，对该站的硬件组态。首先生成该站的机架，插入 CPU 315-2DP、电源模块和信号模块。CPU 的订货号为 6ES7 315-2AG10-0AB0。

将 CPU 放到机架上时，DP 接口属性对话框的“参数”选项卡被自动打开。设置 PROFIBUS 站地址为 3，不连接到 PROFIBUS（1）子网络。点击“确定”按钮，返回 HW Config。

双击 CPU 中 DP 所在的行，打开 DP 属性对话框。在“工作模式”选项卡中，将该站设置为“DP 从站”（见图 5-25）。

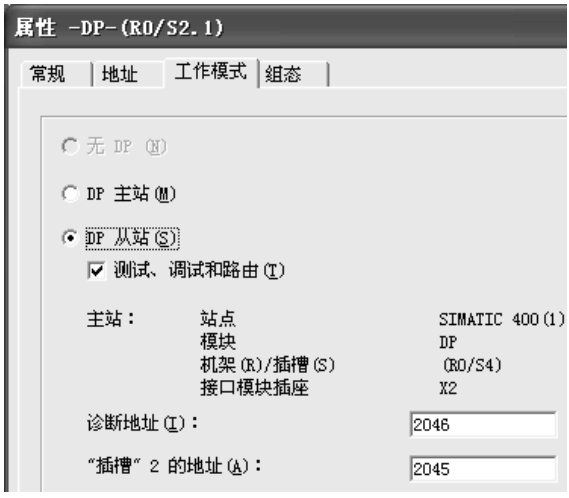


图 5-25 组态 DP 从站的工作模式

此处选中了通过 DP 网络实现“测试、调试和路由”功能。在“DP 从站”模式下，这些

功能会增大总线周期。因此，对时间要求苛刻的应用程序不应启用此选项。

此时为 DPV1 主站自动生成了两个诊断地址（见图 5-25）。“诊断地址”2046 被指定给 DP 从站的虚拟插槽 0，通过这个地址，DP 主站的用户程序可以获取 DP 从站的状态信息。

只有当 DP 主站支持“DPV1”模式，并且设置了该模式时，才能看到虚拟插槽 2 的地址 2045。DP 智能从站的用户程序使用该地址，调用 SFC 7“DP_PRAL”来触发 DP 主站的硬件中断。

点击“确定”按钮，返回 HW Config。点击工具栏上的按钮，保存组态信息。

3．将智能从站连接到 DP 主站系统

选中 SIMATIC 管理器中的 S7-400 站，双击右边窗口的“硬件”图标，在 HW Config 中，打开右边的硬件目录窗口中的“\PROFIBUS DP\Configured Stations”（已组态的站）文件夹，将其中的“CPU 31x”拖放到左边窗口的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡被自动打开，选中从站 CPU 列表中的“CPU 315-2DP”，点击“连接”按钮，该从站被连接到 DP 网络上（见图 5-26）。

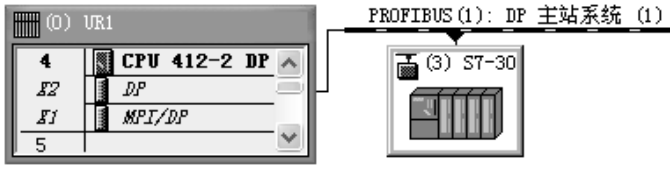


图 5-26　HW Config 中的主站与智能从站

双击图 5-26 中的 DP 从站，在打开的 DP 从站属性对话框的“常规”选项卡中（见图 5-27），自动分配了两个地址：

图 5-27　DP 从站属性对话框

1）诊断地址 4093 分配给 DP 从站的虚拟插槽 0。DP 从站用来读取 DP 主站的故障信息。通过此诊断地址，向主站报告 DP 从站的故障或返回信息。主站 CPU 将会启动 OB86“机架/DP 从站故障”。

主站也可以通过 SFC 13“DPNRM_DG”，用此地址来从 DP 从站获取完整的诊断信息。

2）“插槽”2 的地址 4092 分配给 DP 从站的虚拟插槽 2，只能用于 DPV1 模式。该地址作为从站触发的主站的硬件中断的局部变量 OB40_MDL_ADDR（模块地址），通知主站是哪一个从站触发的中断。

4．为主站和从站之间的数据交换分配 I/O 区

在“组态”选项卡中，为主站和从站之间的数据交换分配 I/O 区（见图 5-28）。

本项目通信双方的 OB1、OB35 和 OB100 中与主从通信有关的程序和变量表，与项目 PB_MS_2 的完全相同，详情请参阅 3.2 节和随书光盘中本项目的文件。

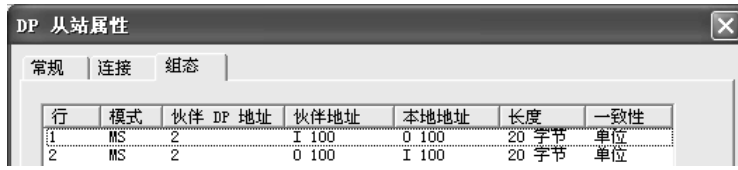


图 5-28　主站从站数据交换的 I/O 区

5．从站触发过程中断的程序设计

在智能从站的 OB1 中调用 SFC 7“DP_PRAL”，在它的输入信号 REQ 的脉冲上升沿，触发 DP 主站的硬件中断，DP 主站调用 OB40 来处理硬件中断，执行过程如图 5-29 所示。

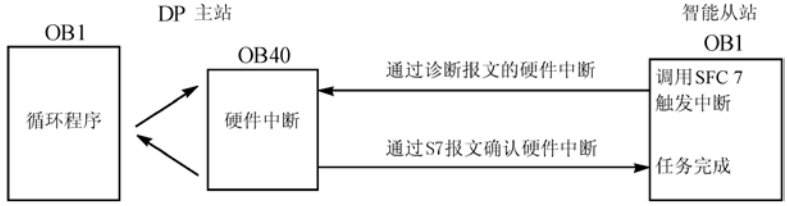


图 5-29　硬件中断执行过程示意图

从站发送两条附加信息给 DP 主站：

1）AL_INFO 为中断标识符，用来说明触发硬件中断的原因。中断标识符被作为硬件中断报文的一部分发送给 DP 主站的 OB40，保存在局部变量 OB40_POINT_ADDR 中。

2）IOID 是模块的地址区标识符，IOID=B#16#54 时为外设输入（PI），IOID=B#16#55 时为外设输出（PQ）。对于既有输入又有输出的混合模块，区域标识符为两个地址中较低的那一个。若两个地址相同，则指定为 B#16#54。

SFC 7 是异步执行的，需要执行多个 SFC 调用周期。下面是从站的 OB1 的程序：

```
L       DW#16#1234ABCD          //设置中断标识符
T       MD      100
CALL    "DP_PRAL"               //调用 SFC 7
  REQ       :=I0.0              //为 1 时启动硬件中断请求
  IOID      :=B#16#54           //模块的地址区域标识符，外设输入（PI）地址
  LADDR     :=W#16#7FD          //组态的虚拟插槽 2 的地址 2045
  AL_INFO   :=MD100             //从站发送给主站的中断信息
  RET_VAL   :=MW2               //返回的故障代码
  BUSY      :=M1.1              //主站未确认中断时为 1
```

如果 SFC 7 未被 DP 主站确认，BUSY 为 1。在主站的 OB40 执行结束，SFC 7 的任务完成时，BUSY 变为 0。SFC 7 的执行过程如果发生错误，返回的故障代码在输出参数 RET_VAL 中。如果 DP 从站是标准从站，只要主站接收到诊断报文，则从站触发的硬件中断完成。

6．S7-400 DP 主站处理硬件中断的程序

由智能从站触发并通过 DP 网络发送的硬件中断被 DP 主站的 CPU 识别后，主站 CPU 的操作系统调用硬件中断组织块 OB40。OB40 的局部数据包含产生中断的模块的逻辑基准地址和中断源的其他信息。对于更复杂的模块，OB40 的局部数据还包含中断标识符和状态信息。在 OB40 执行结束后，DP 主站的 CPU 自动发送一个确认信号给触发此中断的智能从站，使从站的 SFC 7 的输出参数 BUSY 的状态从 1 变为 0。

DP 主站 CPU 412-2DP 的组织块 OB40 中的程序如下所示：

```
        L     #OB40_MDL_ADDR          //保存触发中断的从站的虚拟插槽 2 的地址
        T     MW      10
        L     #OB40_POINT_ADDR        //保存智能从站发送的中断 ID
        T     MD      12
        L     DW#16#1234ABCD          //与产生中断时从站传送的常数进行比较
        ==I
        JC    m001                    //如果相等则跳转
        BEC                           //块结束
m001:   CALL  FC      100             //调用处理 3 号从站中断的FC
```

7．测试 DP 主站对硬件中断的响应

主站和从站的变量表（见图 5-30 和图 5-31）上面 5 行用来监控主从通信的实现情况。主站 CPU 的变量表的下面两行用来监控 MW10 和 MD12，它们保存了触发中断的 DP 从站虚拟插槽 2 的地址和发送的中断 ID。

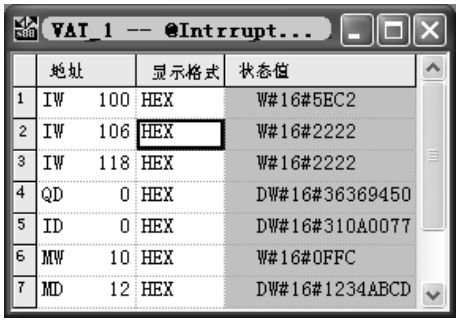


图 5-30　主站的变量表

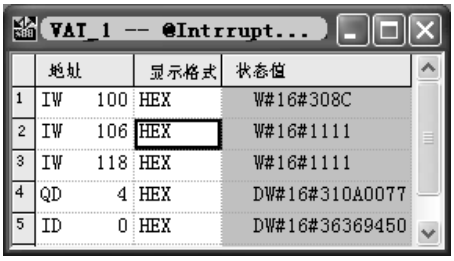


图 5-31　从站的变量表

将组态信息和程序下载到两个 CPU，用 PROFIBUS 电缆连接两个 CPU 的 DP 接口，PLC 切换到 RUN 模式。将变量表切换到监控模式，用从站的 I0.0 触发硬件中断。

2 号主站的变量表中的 MW10 是 OB40 的局部变量 OB40_MDL_ADDR，即触发中断的 3 号从站的虚拟“插槽”2 的地址 16#FFC，对应的十进制数为 4092（见图 5-27）。MD12 是 OB40 的局部变量 OB40_POINT_ADDR（模块的中断信息），它就是在从站的 OB1 中调用的 SFC 7 的参数 AL_INFO，通过它可以验证通过通信产生的硬件中断是否成功。

5.2.2　一组从站的输出同步与输入冻结

1．同步输出与解除同步

通常情况下，DP 主站周期性地将输出数据发送到 DP 从站的输出模块上。DP 主站调用

用 SFC 11 发送 SYNC（同步）控制命令，组态的 DP 从站组中的从站将切换到“同步”模式，DP 主站将当前的输出数据发送给从站，并指示相关 DP 从站冻结各自的输出。DP 从站组将主站随后的输出数据存放在它们的内部缓冲区，将它们送到输出模块，并保持输出状态不变。这样可以同步激活一组 DP 从站上的输出数据。每执行一次 SYNC 控制命令，该组从站将新的输出数据发送到输出模块上。

只有用 SFC 11 发送控制命令 UNSYNC，才能解除 DP 从站组的 SYNC 模式，使该组 DP 从站返回正常的循环数据传送状态，即 DP 主站发送的数据立即被传送到从站的输出点。

2．输入信号的冻结与解除冻结

通常情况下，DP 主站按照 PROFIBUS-DP 的总线周期，周期性地读取 DP 从站的输入数据，供 CPU 使用。如果需要得到一组 DP 从站上同一时刻的输入数据，可以通过 SFC 11 将 FREEZE 控制命令发送到该组 DP 从站来实现。

当 FREEZE 命令被发送到一组 DP 从站时，组内所有的 DP 从站切换到 FREEZE 模式，即它们的输入模块上的信号被冻结，并将它们传送到 CPU 的过程映像输入区，以便 DP 主站来读取这些信号。接收到下一个 FREEZE 命令时，DP 从站更新和重新冻结它们的输入数据。

只有用 SFC 11 发送 UNFREEZE 命令，才能解除 DP 从站的 FREEZE 模式，使 DP 主站重新开始周期性地接收从站当前的输入状态。

在暖启动和热启动后，DP 从站不进入 SYNC 或 FREEZE 模式，只有当它们接收到由 DP 主站发出的第一个 SYNC 或 FREEZE 命令之后，才进入 SYNC 或 FREEZE 模式。

3．生成主站和 DP 网络

打开 STEP 7，用新建项目向导创建一个名为“Syncfrez”的项目（见随书光盘中的同名例程），CPU 为 CPU 313C。选中该站，双击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

将支持 SYNC 和 FREEZE 功能的 CP 342-5 插入机架，其订货号为 6GK7 342-5DA01-0XE0。在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，默认的 DP 站地址为 2。点击“新建”按钮，生成一个新的 PROFIBUS 子网络，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，采用默认的传输速率（1.5Mbit/s）和配置文件（DP），点击“确定”按钮返回 CP 属性对话框，将 CP 连接到 DP 网络上。点击“确定”按钮返回 HW Config，双击 CP 342-5，打开它的属性对话框，在“常规”选项卡中设置它的 MPI 地址为 3。在“工作模式”选项卡中设置它的工作模式为 DP 主站。

4．组态从站

在硬件目录中打开文件夹“\PROFIBUS DP\ET 200B”，将模块“B-16DI/16DO DP”拖放到图 5–32 中的 DP 主站系统网络线上。在自动打开的模块的 PROFIBUS 接口属性对话框的“参数”选项卡中，设置其站地址为 3。用同样的方法将一块“B-24DI/8DO DP”模块和一块“B-16DO DP”模块拖放到 DP 网络线上，它们默认的站地址分别为 4 和 5。

5．组态 SYNC/FREEZE 功能

双击图 5–32 中的“PROFIBUS（1）: DP 主站系统（180）”网络线，出现“属性 –DP 主站系统”对话框。首先指定组的属性，为此打开“组属性”选项卡（见图 5–33），用“属性:”

下面的复选框选择要指定给各组的属性。图中定义组 1 为 FREEZE 组，组 2 为 SYNC 组。实际上只使用了组 1 和组 2。在“注释:”列可以为各组附加注释。

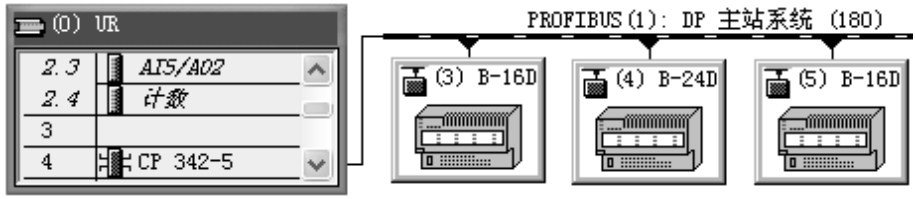


图 5-32　组态 DP 从站

图 5-33　设置 SYNC/FREEZE 组的属性

在“组分配”选项卡（见图 5-34 中），将 DP 从站分配到各组。列表框中的每一行对应一个 DP 从站，最左边是从站的地址和模块的型号，例如 3 号从站对应于“(3) B-16DI/16DO”。列表框的上面给出了每一组的属性，例如，第 1 组下面的“--”表示它不属于“SYNC（同步）”组，“X”表示它属于“FREEZE（冻结）”组。

选中列表框中的第一行（3 号从站 B-16DI/16DO），用鼠标在列表框下面的 1 和 2 前面的复选框中打勾，第一行中第 1 组和第 2 组对应的位置出现字符“X”，表示 3 号从站分别属于第 1 组和第 2 组。

图 5-34　分配 SYNC/FREEZE 组

用同样的方法，使 4 号从站属于第 1 组，5 号从站属于第 2 组。从图 5-34 可以看出，3 号从站和 4 号从站属于冻结组（第 1 组），3 号从站和 5 号从站属于同步组（第 2 组）。设置好后点击“确定”按钮，关闭对话框。点击工具栏上的按钮（编译与保存），在保存组态信

息的同时对硬件组态进行编译。编译成功后将硬件组态信息下载到 CPU 313C。下载结束后关闭“HW Config”。

6．系统功能 SFC 11“DPSYC_FR”

SFC 11“DPSYC_FR”用于将控制命令 SYNC（同步输出）、UNSYNC（解除同步）、FREEZE（冻结输入）和 UNFREEZE（取消冻结）发送给一个或多个 DP 从站。这些命令用来实现一组 DP 从站的同步输出或冻结它们的输入。

DP 主站使用全局控制报文（广播报文），同时发送控制命令 SYNC 和/或 FREEZE 给一组 DP 从站。在用 SFC 11 发送上述控制命令之前，应使用 STEP 7 的硬件组态工具将有关的 DP 从站组合到 SYNC/ FREEZE DP 组中，一个主站系统最多可以建立 8 个组。

SFC 11 用输入参数 MODE 指定的控制命令可能的组合见表 5-1。

表 5-1　SFC 11 的控制命令可能的组合

位　　号	7	6	5	4	3	2	1	0	取　　值
MODE				UNSYNC					B#16#10
				UNSYNC		UNFREEZE			B#16#14
				UNSYNC	FREEZE				B#16#18
			SYNC						B#16#20
			SYNC			UNFREEZE			B#16#24
			SYNC		FREEZE				B#16#28
						UNFREEZE			B#16#04
					FREEZE				B#16#08

参数 LADDR 是 DP 主站的逻辑基准地址。如果已触发的系统功能还未结束执行，则 BUSY 为 1。在块执行过程中发生错误时，RET_VAL 参数中是返回的故障代码。

SFC 11 是异步执行的，需要执行多个 SFC 调用周期。REQ 为 1 时，调用 SFC 11 来执行同步和冻结操作。在同一时间只能初始化一条 SYNC/UNSYNC 命令或一条 FREEZE/UNFREEZE 命令。

若调用了 SFC 15“DPWR_DAT”（写 DP 数据），在发送 SYNC 给有关的输出之前，SFC 15 必须执行完毕。若调用了 SFC 14“DPRD_DAT”（读 DP 数据），在发送 FREEZE 给有关的输入之前，SFC 14 必须执行完毕。

7．SYNC/FREEZE 功能的编程

SFC 11 在程序编辑器左边窗口的文件夹“\Libraries\Standard Library\System Function Blocks”中。

双击 SIMATIC 管理器的“块”文件夹中的“OB1”图标，打开程序编辑器，然后输入下面的语句表程序。在 I0.0 的上升沿调用 FC 11“DPSYC_FR”，发送 FREEZE 命令，在 I0.1 的上升沿发送 SYNC 命令。

```
Network 1：检测 I0.0 的上升沿
      A     I     0.0
```

```
          FP    M     10.1          //在 I0.0 的脉冲上升沿
          =     M     10.2          //M10.2 在一个循环周期为 1 状态
Network 2：发送 FREEZE 命令
m001:     CALL  "DPSYC_FR"          //调用 SFC 11
            REQ     :=M10.2         //在 I0.0 的上升沿发送 FREEZE 命令
            LADDR   :=W#16#2        //DP 主站的逻辑地址
            GROUP   :=B#16#1        //选择第 1 组
            MODE    :=B#16#8        //发送 FREEZE 命令（见表 5-1）
            RET_VAL :=MW12          //故障代码返回值
            BUSY    :=M10.3         //操作未完成时为 1
          A     M     10.3          //如果没有执行完 SFC 11
          JC    m001                //跳转到标号 m001 处继续执行
Network 3：检测 I0.1 的上升沿
          A     I     0.1
          FP    M     10.5          //在 I0.1 的脉冲上升沿
          =     M     10.6          //M10.6 在一个循环周期为 1 状态
Network 4：发送 SYNC 命令
m002:     CALL  "DPSYC_FR"          //调用 SFC 11
            REQ     :=M10.6         //在 I0.1 的上升沿发送同步命令
            LADDR   :=W#16#2        //DP 主站的逻辑地址
            GROUP   :=B#16#2        //选择第 2 组
            MODE    :=B#16#20       //发送 SYNC 命令（见表 5-1）
            RET_VAL :=MW14          //故障代码返回值
            BUSY    :=M10.7         //操作未完成时为 1
          A     M     10.7          //如果没有执行完 SFC 11（M10.7 为 1）
          JC    m002                //跳转到标号 m002 处继续执行
```

用 PROFIBUS 电缆连接 CPU 313C 和 3 个 ET 200B 模块的 DP 接口，并将 CPU 313C 和 CP 的运行模式切换到 RUN。

在 SIMATIC 管理器中生成变量表，在 RUN 模式打开变量表。在变量表中监视“冻结”组中的 3 号从站的 IW0 和 4 号从站的 IB2～IB4，以及“同步”组中 3 号从站的 QW0 和 5 号从站的 QW3。用 I0.0 来触发 FREEZE 组的操作，用 I0.1 来触发 SYNC 组的操作。

系统进入 RUN 模式后，主站与各 DP 从站循环地传送数据。将 I0.0 置为 1 状态，SFC 11 发送 FREEZE 控制命令，使 3 号从站和 4 号从站的输入处于 FREEZE 模式。用输入点外接的小开关改变 3 号从站或 4 号从站的输入信号的状态，因为处于冻结模式，这些变化不会传送给主站的 CPU，在主站的变量表中也不能观察到这些变化。

将 I0.1 置为 1 状态，SFC 11 发送 SYNC 命令，使 3 号从站和 5 号从站的输出处于 SYNC 模式。在变量表中修改 QW0 或 QW3 的值后，不能将它们传送到 3 号从站或 5 号从站的输出模块。

在 I0.0 的下一个上升沿，重新发送 FREEZE 命令，读取 3 号从站和 5 号从站当前的输入数据。在 I0.1 的下一个上升沿，重新发送 SYNC 命令，把设置好的数据传送到 3 号从站和 5 号从站的输出模块。使用 CP 342-5 时，需要调用 FC 1，将数据打包后发送给 ET 200B，调用 FC 2，将来自 ET 200B 的数据存放到指定的地址区。

```
Network 5：发送数据到 ET 200B
      CALL   "DP_SEND"                    //调用 FC 1
        CPLADDR :=W#16#100                //CP 342-5 的起始地址 256
        SEND    :=P#M 20.0 BYTE 5         //S7-300 的发送数据区
        DONE    :=M0.0                    //发送完成产生一个脉冲
        ERROR   :=M0.1                    //错误标志位
        STATUS  :=MW2                     //通信状态字
Network 6：  接收来自 ET 200B 的数据
      CALL   "DP_RECV"                    //调用 FC 2
        CPLADDR :=W#16#100                //CP 342-5 的起始地址 256
        RECV    :=P#M 25.0 BYTE 5         //S7-300 的接收数据区
        NDR     :=M0.2                    //接收完成产生一个脉冲
        ERROR   :=M0.3                    //错误标志位
        STATUS  :=MW4                     //通信状态字
        DPSTATUS :=MB6                    //DP 网络的状态字节
```

5.2.3 用 SFC 12 激活和禁止 DP 从站

1．SFC 12 的功能

使用 SFC 12“D_ACT_DP”，可以禁止已组态，但是在 PLC 启动时并不存在或当前并不需要或有故障的 DP 从站和 PROFINET IO 设备。可以在需要时用 SFC 12 来激活它们，还可以查询它们当前处于激活状态还是处于禁止状态。

某些设备有大量的选件可供使用，但是机器制造厂商交付的具体的设备仅仅是选定组件的组合。例如，机床可以使用大量的加工选件，但是实际上经常用到的仅仅是其中的一小部分。制造商将这些可能的机器选件组态为 DP 从站或 PROFINET IO 设备，以便创建包含所有可能的选件的通用用户程序。这些选件在 STEP 7 项目中作为 DP 从站或 PROFINET IO 设备出现。使用 SFC 12，用户程序可以激活当前需要的选件，禁止那些当前不需要的选件。

如果系统中有已经组态，但是并不存在或不是当前所需的 DP 从站或 PROFINET IO 设备，CPU 仍然会不断地访问这些 DP 从站或 PROFINET IO 设备。如果禁止这些从站或 IO 设备，CPU 将停止访问它们，这样可以缩短 DP 总线周期。如果用 SFC 12 禁止了 IE/PB Link PN IO，所有从属的 PROFINET IO 设备也将停止运行，这一事件将被报告。

SFC 12 不能用于通过 DP/PA 链接器连接到 DP 主站系统的 PROFIBUS-PA 现场设备。SFC 12 以异步方式运行，它通过多次 SFC 调用来执行。输入参数 REQ 为 1 时执行激活或禁止操作。

2．禁止 DP 从站或 PROFINET IO 设备

用 SFC 12 禁止 DP 从站或 PROFINET IO 设备后，其过程输出将被设置为组态的替换值或 0（安全状态）。相应的 DP 主站将不再访问这个 DP 从站。DP 主站、PROFINET IO 控制器或 CPU 上的错误 LED 不会因为禁止操作显示错误信息。被禁止的 DP 从站或 PROFINET IO 设备输入的过程映像将用 0 更新，即将其作为有故障的 DP 从站或 PROFINET IO 设备处理。

如果用程序直接访问被禁止的 DP 从站或 PROFINET IO 设备的用户数据，CPU 将自动调用 I/O 访问错误 OB（OB122），并在诊断缓冲区中输入相应的启动事件。如果用 SFC 59

“RD_REC”访问禁止的DP从站或PROFINET IO设备，与访问不可用的DP从站或PROFINET IO设备相同，SFC的RET_VAL将提供出错信息。

禁止DP从站或PROFINET IO设备时不会启动OB85、OB86，操作系统也不会向诊断缓冲区输入条目。如果用SFC 12禁止DP从站或PROFINET IO设备，则当它们出现故障时，操作系统不对故障进行检测，因而不会启动OB86或诊断缓冲区条目。只有在重新激活该站后，才能检测到站故障，并将故障信息写入返回值RET_VAL。

3．激活DP从站或PROFINET IO设备

使用SFC 12重新激活DP从站或PROFINET IO设备时，DP主站或PROFINET IO控制器将对该组件进行组态和分配参数，就像有故障的DP从站或PROFINET IO设备被重新激活的情况一样。当组件能够传送用户数据时，激活操作即告完成。

即使其输入或输出属于要更新的过程映像，激活DP从站或PROFINET IO设备也不会启动OB85，并且不会向诊断缓冲区输入条目。激活DP从站或PROFINET IO设备不会启动OB86，操作系统也不会向诊断缓冲区输入条目。

如果试图用SFC 12激活一个被禁止并且已经在物理上与DP总线分离的从站，CPU上的BUSF LED将会闪烁1min，SFC将返回错误代码W#16#80A2，从站仍保持禁止状态。如果以后从站重新连接到DP总线，必须用SFC 12重新激活它。

4．CPU启动时对DP从站或PROFINET IO设备的处理

根据启动模式的不同，CPU操作系统对DP从站或PROFINET IO设备按下面的方式处理：

1）在冷启动和暖启动模式，DP从站或PROFINET IO设备被自动激活。

2）在热启动模式，DP从站或PROFINET IO设备的激活状态保持不变。

CPU启动之后，将定期试图联系所有已组态但未禁止的从站/设备，这些从站可能不存在，或者未响应。

5．DP网络组态

在STEP 7中创建一个新的项目（见随书光盘中的例程SFC_12），CPU为CPU 313C-2DP。打开HW Config，将电源模块和I/O模块插入机架。

双击机架中“DP”所在的行，点击出现的DP属性对话框的“属性”按钮，在出现的PROFIBUS接口属性对话框中，采用默认的DP地址2。点击“新建”按钮，生成一条新的PROFIBUS子网络，在出现的“属性 – 新建子网PROFIBUS”对话框的“网络设置”选项卡中，采用默认的传输速率（1.5Mbit/s）和配置文件（DP），点击“确定”按钮返回DP属性对话框，将CPU连接到DP网络上。

返回HW Config后，可以看到新创建的PROFIBUS-DP子网络（见图5-35）。将右边硬件目录窗口的“\PROFIBUS DP\ET 200B”文件夹中的“B-16DO”和“B-16DI”拖放到DP网络上，在自动打开的“属性 –PROFIBUS接口”对话框的“参数”选项卡中，设置从站的地址为4和5，自动分配的4号从站的输出字节地址为0和1，5号从站的输入字节地址为4和5。

将“\PROFIBUS DP\ET 200M”文件夹中的“IM 153-1”拖放到DP网络上，在自动打开的“属性 –PROFIBUS接口”对话框的“参数”选项卡中，设置从站的地址为7。

选中该从站后，在下面的4号槽插入一块8点DO模块，5号槽插入一块16点DI模块。自动分配的输出字节地址为2，输入字节地址为6和7。点击工具栏上的按钮，编译并保存硬件组态信息。

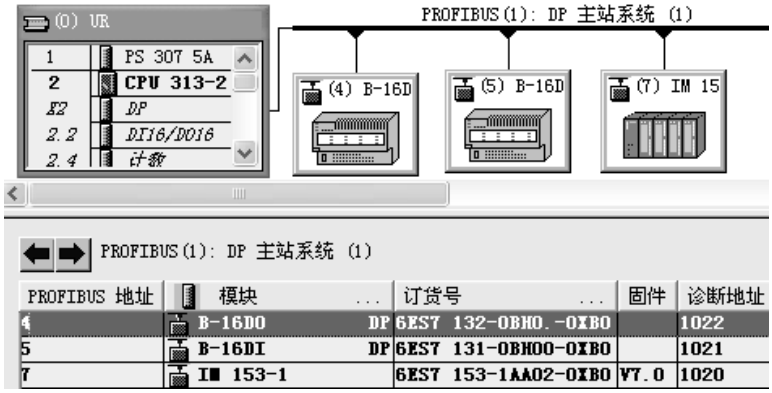


图 5-35　组态 DP 从站

选中 DP 主站系统的网络线，HW Config 下面的窗口中的诊断地址将用于 SFC 12 的输入参数 LADDR。

6．编写激活与禁止 DP 从站的程序

SFC 12 的输入参数 MODE 可能的取值和意义如下：

- MODE=0：查询 DP 从站处于激活还是禁止状态。
- MODE=1：激活 DP 从站。
- MODE=2：禁止 DP 从站。

在 OB1 中编写调用 SFC 12 的程序，图 5-36 中 SFC 12 的输入参数 MODE 为 1，表示激活 DP 从站。输入参数 LADDR 为 W#16#3FC，是 7 号从站的诊断地址 1020。I0.2 为 1 状态时，控制参数 REQ 为 1，开始激活 7 号 DP 从站。激活操作未完成时，输出参数 BUSY（M0.4）为 1，REQ 保持 1 状态。激活操作完成后，M0.4 变为 0 状态。图 5-37 中的 MODE 为 2，I0.2 为 0 状态时，禁止 7 号 DP 从站。

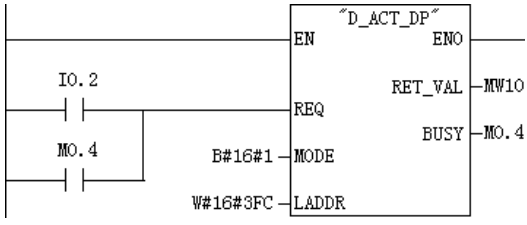


图 5-36　激活 7 号从站

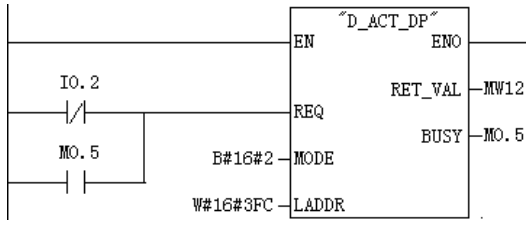


图 5-37　禁止 7 号从站

禁止与激活 4 号和 5 号从站的程序与 7 号从站的程序基本上相同，分别用 I0.0 和 I0.1 禁止和激活 4 号和 5 号从站。

在 SIMATIC 管理器中，打开“块”文件夹，生成组织块 OB82、OB86 和 OB122。下面是 OB82 中的语句表程序，MW20 用来记录调用 OB82 的次数。

```
L     MW    20
+     1
T     MW    20
```

在 OB86 和 OB122 中，分别将 MW22 和 MW24 加 1。

7．实验过程

下面以 7 号从站 ET 200M 为例，介绍用 SFC 12 禁止和激活 DP 从站的实验过程。

在 SIMATIC 管理器中生成变量表（见图 5-38），在变量表中监控 ET 200M 的输出字节 QB2 和输入字 IW6。变量表中的 MW20～MW24 分别用来记录 OB82、OB86 和 OB122 的中断次数。

将程序和组态数据下载到 CPU 后，用 PROFIBUS 电缆连接 CPU 313C-2DP、ET 200B 和 ET 200M 的 DP 接口。接通它们的电源，将 CPU 切换到 RUN 模式。打开变量表，点击工具栏上的按钮，使变量表进入监控状态。I0.0～I0.2 分别用于激活和禁止 3 个从站。

（1）激活 7 号从站

为了验证 DP 主站与从站的通信，在 OB1 的程序段 7 编写下面两条语句，将主站读取的 7 号从站的外设输入字节写入 7 号从站的外设输出字节：

```
L     PIB     6
T     PQB     2
```

令 I0.2 为 1 状态，7 号从站处于激活状态。用接在 7 号从站的输入模块的小开关改变 IB6 的值，可以看到 7 号从站的输出模块的 QB2 的值随之而变，表明主站与 7 号从站之间的通信正常（见图 5-38）。

（2）禁止 7 号从站

令 I0.2 为 0 状态，7 号从站处于禁止状态。ET 200M 的接口模块 IM 153-1 的 BF（总线故障）LED 闪烁，各输出点的 LED 熄灭，它们被禁止输出。变量表中 7 号从站的输入字 IW6 为 0。主站 CPU 313C-2DP 的 LED 没有显示错误信息。

因为用外设输入 PIB6、外设输出 PQB2 直接访问被禁止的 7 号从站，CPU 在每个扫描循环周期调用一次 I/O 访问错误组织块 OB122，用来记录 OB122 中断次数的 MW24 快速增大（见图 5-39），并在 CPU 的诊断缓冲区中输入相应的 OB 启动事件。

图 5-38　激活 7 号从站时的变量表

图 5-39　禁止 7 号从站时的变量表

禁止 DP 从站时，CPU 没有启动其他故障处理组织块，操作系统也没有向诊断缓冲区输入其他条目。用 SFC 12 禁止 3 个从站，断开它们的电源后又接通，操作系统没有启动故障处理组织块 OB86、OB82，诊断缓冲区也没有有关事件的条目。

用 I0.0 和 I0.1 禁止 4 号、5 号从站（ET 200B）时，它们的 BF 和 RUN LED 亮，4 号从站的输出点的 LED 熄灭，它们被禁止输出。主站读取的 5 号从站的输入字 IW4 为 0。

（3）重新激活 7 号从站

令主站的 I0.2 为 1 状态，7 号从站返回激活状态，BF LED 停止闪烁，7 号从站的输入、

输出恢复正常，又可以用 IB6 控制 QB2。

重新激活 DP 从站时，没有启动故障处理 OB，操作系统也没有向诊断缓冲区输入条目。

激活后即使用外设输入 PIB6、外设输出 PQB2 直接访问 7 号从站，CPU 也不会调用 I/O 访问错误组织块 OB 122。

7 号从站被激活后，断开它的电源，3 个从站同时断电。CPU 调用 OB86，CPU 的 SF LED 亮，BF（总线错误）LED 闪烁，诊断缓冲区写入有关的事件信息。

5.2.4 PROFIBUS 子网的恒定总线周期

1．恒定总线周期的基本概念

与主站机架中的集中式 I/O 相比，PROFIBUS-DP 网络上的分布式 I/O 没有确定的过程响应时间。以图 5-40 中的系统为例，如果 CPU 从 ET 200S 读入一个输入信号，执行用户程序后发送给 ET 200M 的一个输出点，整个过程由图中的 T1～T7 这 7 段时间组成：

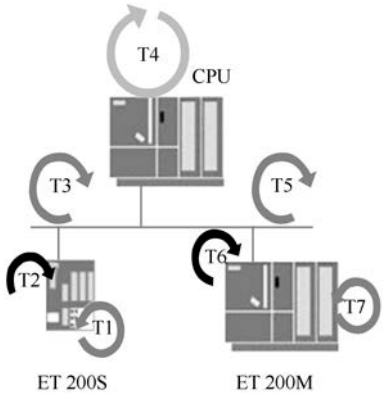


图 5-40　DP 网络的信号传输

T1：读入输入信号的转换时间。

T2 和 T6：从站模块背板总线上的循环时间。

T4：主站 CPU 程序执行时间。

T3 和 T5：PROFIBUS-DP 总线上的轮询时间。

T7：输出信号传送到输出端子的时间。

T1～T7 决定了整个过程的响应时间。上述 7 段时间并非全部是固定不变的，它们之间也没有同步的关系，因此总的处理周期不是恒定的。

某些生产和过程（例如运动控制和闭环控制）要求控制系统具有迅速准确的响应时间。通过 PROFIBUS 的等时模式，可以实现从分布式 I/O 的输入响应、CPU 的处理和输出到分布式 I/O 的端子，有一个确定的和相等的时间，时间响应的确定性甚至可能优于集中式 I/O。

等时模式整个过程的响应时间是由恒定的 DP 总线周期和同步的单个时间段构成的。DP 主站周期性地发送给各从站一个作为时钟脉冲的全局控制帧（GC），用它来同步接收和发送从站数据。总线循环时间的固定和各段时间的同步保证了过程响应时间的恒定。与非等时模式相比，总线循环时间减少了等待读取数据和发送数据的时间，使响应变得更加迅速。

在每个恒定的总线周期，DP 主站首先处理与各从站的循环数据交换（见图 5-41），然后处理中断、总线接收、诊断服务等非循环部分，可能还需要处理与 PG/OP 的通信。DP 主站随后将保持一段等待的时间，直到组态的恒定 DP 总线周期时间到，以便对可能的网络干扰进行补偿，并重新获取可能重发的消息帧。此后，全局控制帧（GC）启动新的 DP 周期。

为确保在新的DP总线周期启动时读取从分布式I/O输入的一致性状态信息，必须在时间T_i内提前执行读取过程（见图 5-41）。T_i包括输入信号的准备和转换时间，以及DP从站背板总线上的传送时间。T_i结束时，所有的输入数据刚好传送到从站的接口模块，此时DP 主站发送全局控制帧（GC），开始轮询DP从站。当所有从站上的数据都已经准备好后，触发同步循环中断OB61，开始执行OB61 中的程序。在OB61 中，调用SFC 126“SYNC_PI”来更新过程映像分区的输入，而在OB61 结束之前，调用SFC 127“SYNC_PO”来更新过程映像分区的输出。在输出时间T_o内，将上一个周期的程序执行结果输出到被控过程，然后开始下一个周期T_i时间内的数据读取过程。

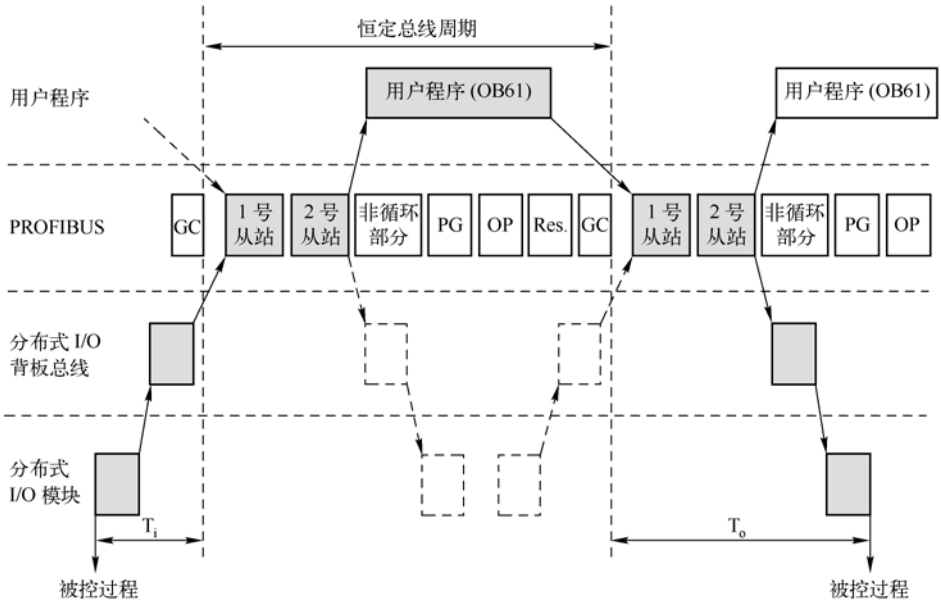


图 5-41　恒定总线周期示意图

时间T_o将确保用户程序的过程响应以相等的时间输出到从站的端子上。T_o包括主站与所有DP从站的数据交换时间、用于电子模块的信号准备和转换时间，以及DP从站背板总线的输出处理时间。可以在组态时设置T_i和T_o。

从图 5-41 可以看出，通信处理、执行用户程序和读写分布式 I/O 是并行（即同时）进行的，这样能提高运行效率，减少所需的总线周期。

循环同步只适用于 ET 200M 和 ET 200S，不能用于集中式 I/O 设备。

2．生成主站和 DP 网络

在 STEP 7 中创建一个名为“Isochron”的项目（见随书光盘中的同名例程），CPU 为 CPU 414-2DP。选中该站点，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，在出现的对话框的“常规”选项卡中点击“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成 PROFIBUS-DP 网络，点击“确定”按钮返回 HW Config。采用默认的参数，CPU 414-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5Mbit/s，配置文件为“DP”。多次点击“确定”按钮，返回硬件组态窗口，在 CPU 414-2DP 的机架右侧出现 PROFIBUS 的网络线。

3．组态模块式 DP 从站 ET 200 M

打开 HW Config 右边的硬件目录窗口中的文件夹“\PROFIBUS-DP\ET 200M”，将其中支持等时线功能的接口模块 IM 153-2 拖放到 PROFIBUS 网络线上，其订货号为 6ES7 153-2BA00-0XB0。在自动打开的模块的 PROFIBUS 接口属性对话框的“参数”选项卡中，设置其站地址为 3。用 IM 153-2 模块上的 DIP 开关设置的站地址应与 STEP 7 组态的站地址相同。

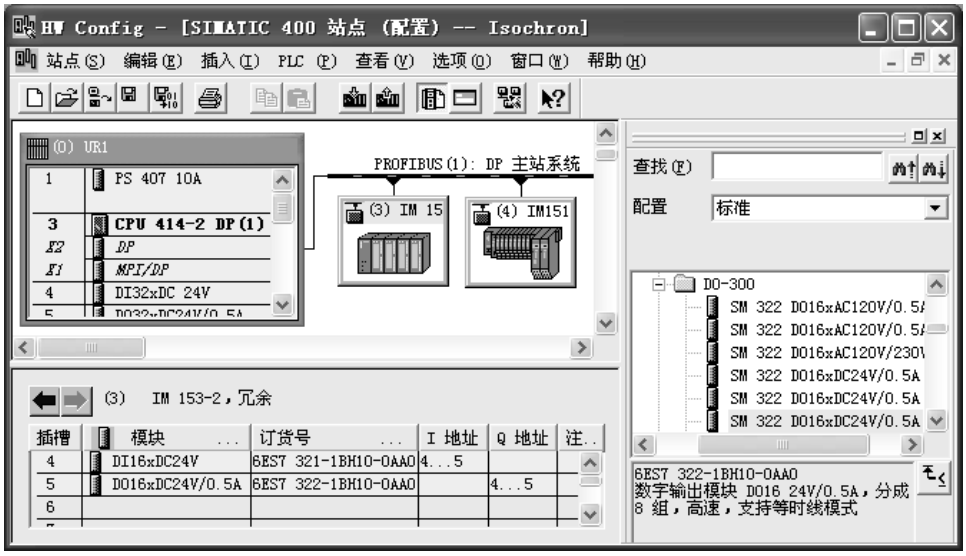


图 5-42　组态 DP 从站

返回 HW Config 后，选中图 5-42 上面窗口中的 3 号从站，打开硬件目录中的“IM 153-2”子文件夹，在下面窗口的 4 号槽插入支持等时线模式的 DI 模块和 DO 模块，它们的订货号见图 5-42。它们的输入、输出字节地址均为 4、5。

4．组态模块式 DP 从站 ET 200S

打开 HW Config 右边的硬件目录窗口中的文件夹“\PROFIBUS DP\ET 200S”，将其中支持等时线模式的高性能接口模块“IM 151 HF”拖放到 PROFIBUS 网络上，其订货号为 6ES7 151-1BA00-0AB0。在自动打开的模块的 PROFIBUS 接口属性对话框的“参数”选项卡中，设置其站地址为 4。

返回HW Config后，选中 4 号从站，在下面窗口的 1 号槽插入电源模块，2 号槽插入支持等时线模式的数字量输入模块 4DI，订货号为 6ES7 131-4BD00-0AB0，3 号槽插入支持等时线模式的 4DO模块，订货号为 6ES7 132-4BD30-0AA0。4DI模块的“输入延迟”时间应设置得尽量短（见图 5-43）。这是因为较短的输入延迟时间可以缩短STEP 7 计算的数据输入时间T_i，从而缩短整个响应时间。

图 5-43　设置数字量输入模块的输入延迟时间

5．设置同步周期中断

双击 HW Config 的机架中 CPU 414-2DP 所在的行，在打开的 CPU 属性对话框的“同步周期中断”选项卡中，设置同步周期中断 OB61 的 DP 主站系统编号为 1（见图 5-44）。如果有多个网段，可以设置和调用 OB61～OB63。“延迟时间”是全局控制帧与启动 OB 61 之间的时间，即 DP 主站用来完成与 DP 从站的循环数据交换的时间。

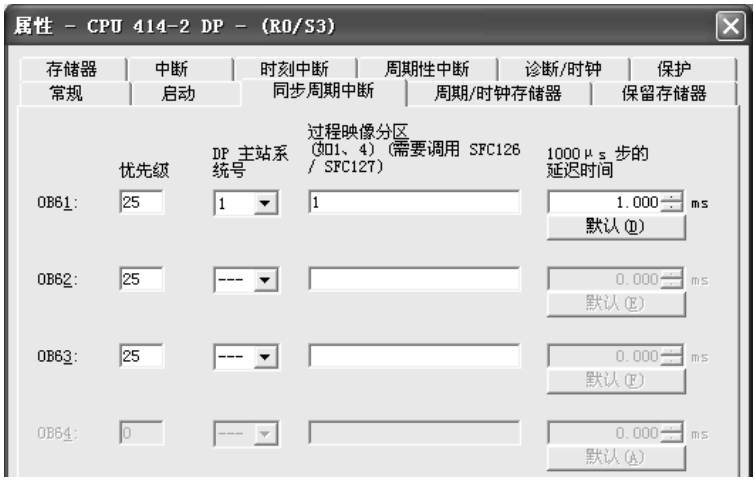


图 5-44　设置同步周期中断

双击 HW Config 中的 DP 网络，点击出现的对话框的“常规”选项卡中的“属性”按钮，再点击出现的“属性 – PROFIBUS”对话框的“网络设置”选项卡中的“选项”按钮，打开出现的“选项”对话框中的“恒定的总线周期”选项卡（见图 5-45），选中复选框“激活恒定的总线周期”，使 PROFIBUS-DP 主站轮询从站的时间是一个固定值，即采用等时模式下的 DP 轮询方式。

激活复选框“对于所有的从站，T_i 和T_o相同”，将使所有从站的 T_i 和T_o保持一致。如果没有选中它，则需要在各DP从站的属性对话框分别设置T_i 和T_o，这样各个从站的T_i 和T_o可能不一致。其他参数采用默认值，最后点击“确定”按钮退出“选项”对话框。

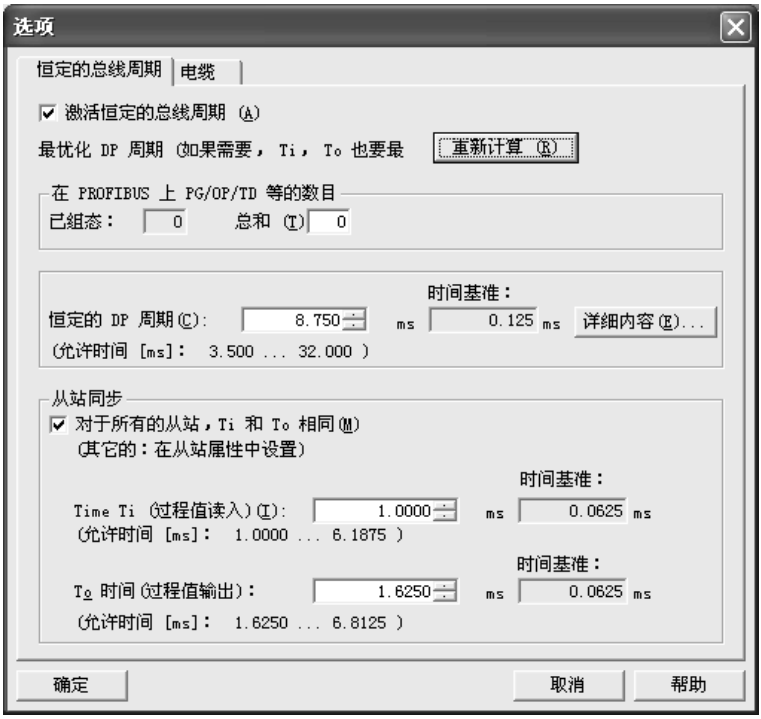


图 5-45　设置恒定的总线周期

如果 PROFIBUS 网络上有 PG/PC（编程器/计算机），在优化恒定的 DP 周期时，它们将会被考虑在内。STEP 7 计算恒定 DP 总线周期的最小值，可以采用比建议的周期更长的时间。

6．设置 DP 从站的等时模式参数

双击 HW Config 上面的硬件组态窗口中的 3 号从站（ET 200M），在打开的“DP 从站属性”对话框的“等时线模式”选项卡（见图 5-46）中，激活复选框“按恒定 DP 总线循环周期时间对 DP 从站进行同步”，选中支持等时模式的 DI 和 DO 模块的“等时曲线操作”，用同样的方法设置 ET 200S 的等时模式参数。

图 5-46　设置从站的等时线模式属性

如果DP主站组态时没有选中复选框“对于所有的从站，T_i 和T_o相同”，则需要单独设置每个从站的T_i 和T_o。

7．设置分布式 I/O 模块的过程映像分区

OB 61 与参与恒定总线周期数据交换的 I/O 模块应采用相同的过程映像分区，这样才能保证快速地更新 I/O 的映像区。以 ET 200M 中的 DI 模块为例，选中硬件组态窗口中的 3 号从站，双击下面窗口中第 4 槽的 DI 模块，在打开的 DI 模块属性对话框的“地址”选项卡中（见图 5-47），设置过程映像分区为 PIP 1（1 号分区）。用同样的方法设置其他参与恒定总线周期数据交换的 I/O 模块所属的过程映像分区均为 PIP1。

图 5-47　设置分布式 I/O 模块的过程映像分区

返回图 5-45 中的“选项”对话框的“恒定的总线周期”选项卡，点击“重新计算”按钮，T_i 和T_o将被重新计算和优化。为了更新全局控制帧与调用同步循环中断OB之间的延迟时间，打开CPU的属性对话框，然后点击“同步周期中断”选项卡的“默认”按钮（见图 5-44），重新计算延迟时间的数值。设置完了全部参数后，点击工具栏上的按钮，编译与保存组态信息。编译成功后在HW Config或SIMATIC管理器中下载组态信息。

8．优化组态

为了优化组态，在 HW Config 中执行菜单命令“编辑”→“等时线模式”，打开“等时线模式”对话框（见图 5-48）。

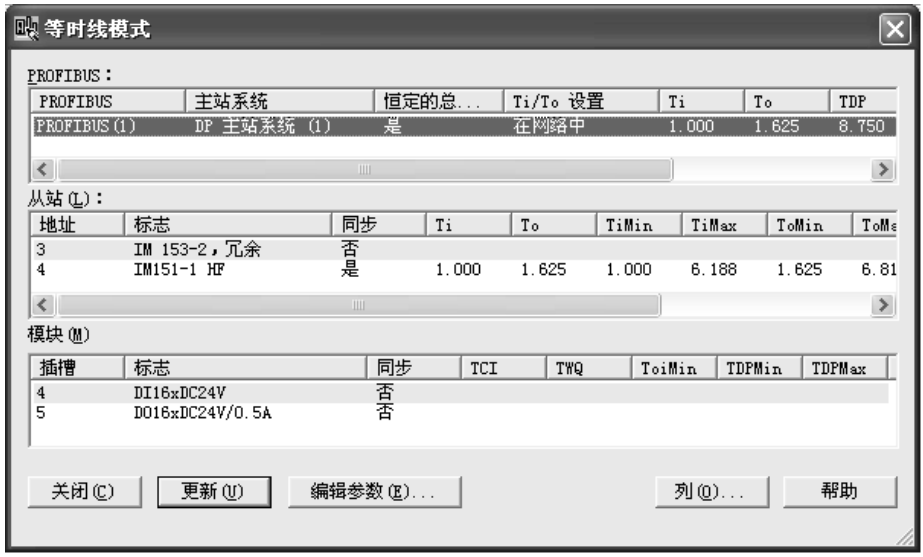


图 5-48　等时线模式对话框

选中“PROFIBUS”列表中的某个主站系统，“从站”列表中将显示该主站系统的从站。选中某个 DP 从站，“模块”列表中将显示该从站中的模块。点击“编辑参数”按钮，将打开选中的对象的属性对话框。修改组态后，单击“更新”按钮，确认在对话框中应用这些更改。

9．等时线模式的编程

在同步循环中断 OB61 中，必须调用 SFC 126“SYNC_PI”来更新过程映像分区的输入，而在 OB 61 结束之前，必须调用 SFC 127“SYNC_PO”来更新过程映像分区的输出。这里所使用的过程映像分区就是在 CPU 属性对话框的“同步循环中断”选项卡中组态的分区。下面是 OB61 中的程序：

程序段 1：在同步循环中更新过程映像分区输入表

```
CALL    "SYNC_PI"                  //调用 SFC 126
  PART       :=B#16#1              //需要在同步循环中更新的过程映像分区编号
  RET_VAL :=MW10                   //错误信息
  FLADDR  :=MW12                   //出现访问错误时引起错误的第一个字节的地址
```

程序段 2：在同步循环中更新过程映像分区输出表

```
CALL    "SYNC_PO"                  //调用 SFC 127
  PART       :=B#16#1              //需要在同步循环中更新的过程映像分区编号
  RET_VAL :=MW14                   //错误信息
  FLADDR  :=MW16                   //出现访问错误时引起错误的第一个字节的地址
```

5.3 练习题

1．组态一个项目，主站为 CPU 412-2DP，3 号从站为 CPU 313C-2DP，4 号从站为 CPU 315-2DP，5 号从站为有两块 16 点 DI 模块的 ET 200M，各从站均有直接数据交换功能。3 号从站与主站之间、4 号从站与主站之间分别用主从通信双向交换 20B 的数据。4 号从站用直接数据交换功能接收 3 号从站发送给主站的 20B 数据。3 号从站用直接数据交换功能接收 5 号从站发送给主站的 4B 输入数据。

2．DP 从站组的输出同步有什么作用，怎样实现输出同步？

3．DP 从站组的输入冻结有什么作用，怎样实现输入冻结？

4．什么情况需要禁止 DP 从站？禁止 DP 从站有什么好处？

5．怎样用 SFC 12 禁止和激活 DP 从站？

6．PROFIBUS 子网的恒定总线周期有什么意义？怎样组态恒定总线周期？

第 6 章　使用 STEP 7 和硬件诊断 PROFIBUS 通信的故障

现代网络控制系统的站点越来越多，网络越来越复杂，对网络控制系统的故障诊断的要求越来越高。出现故障时，应能及时向 CPU 报告，并用预先编制的程序进行处理。同时将故障信息及时地发送给人机界面或上位机，以报警消息的形式显示出来，帮助维修人员及时地查明故障的原因。S7−300/400 提供多种多样的故障诊断和故障显示的方法、诊断用的程序块和用于诊断的专用设备，供用户检查和定位网络控制系统的故障。

本章介绍西门子的 PROFIBUS 网络故障的主要诊断方法，包括用设备上的 LED（发光二极管）和 STEP 7 诊断故障的方法，使用通信程序块的输出参数和组织块的局部变量进行诊断的方法，使用通信处理器进行诊断的方法，以及使用专用硬件诊断网络故障的方法。

用专用的程序块诊断通信故障的方法将在第 7 章介绍，诊断消息的显示方法将在第 8 章介绍，PROFIBUS-PA、PROFINET 和 AS-i 的故障诊断分别在第 9、11、12 章介绍。

6.1　用设备上的 LED 进行诊断

CPU、远程 I/O 站点和模块上的 LED 提供了定位网络故障的基本信息。可以使用 STEP 7 或调用诊断程序块来获取更多的诊断信息，以便准确地确定故障的具体原因和位置。CPU 在线的模块信息中的诊断缓冲区提供了错误的文本信息，例如出错的 DP 站地址、出错的模块的地址和故障。

6.1.1　用 S7-300 CPU 的 LED 进行诊断

1．S7-300 的 LED 功能简介

1）SF（系统错误/故障，红色）：在 CPU 有硬件故障或软件错误时亮。可能的故障包括硬件故障、固件故障、存储卡故障、外部 I/O 故障、上电时电池有故障或没有后备电池、编程错误、参数设置错误、计算错误和时间错误等。

2）BATF（电池故障，红色）：电池电压低或没有电池时亮。

3）DC 5V（+5V 电源指示，绿色）：CPU 和 S7-300 总线的 5V 电源正常时亮。

4）FRCE（强制，黄色）：至少有一个 I/O 点被强制时亮。正常运行时应取消全部强制。

5）RUN（运行模式，绿色）：CPU 处于 RUN 模式时亮；重新启动时以 2Hz 的频率闪亮；HOLD（保持）状态时以 0.5Hz 的频率闪亮。

6）STOP（停止模式，黄色）：CPU 处于 STOP、HOLD 模式或重新启动时常亮；请求存储器复位时以 0.5 Hz 的频率闪动，正在执行存储器复位时以 2 Hz 的频率闪动。

7）BUSF（总线错误，红色）：PROFIBUS−DP 接口硬件或软件故障时亮，集成有 DP 接口的 CPU 才有此 LED。

2．SF LED 亮时可能的软件错误

1）启用和触发了日期时间中断，但是未装载日期时间中断组织块 OB10。

2）用 SFC 32 触发了延时中断，但是未装载延时中断组织块 OB20。

3）启用并触发了过程中断（硬件中断），但是未装载硬件中断组织块 OB40。

4）DPV1 从站因为改变运行模式触发了状态中断，但是未装载状态中断组织块 OB55。

5）通过本地或远程访问，用户更改了 DPV1 从站插槽的参数，因此触发了更新中断，但是未装载更新中断组织块 OB56。

6）DPV1 从站的插槽触发了供应商特定的中断，但是未装载对应的组织块 OB57。

7）更新过程映像时访问了缺失的或有故障的模块。

以上情况下，CPU 将调用 OB85，如果未装载需要的错误处理 OB，CPU 会切换到 STOP 模式。装载 OB85 后，其启动信息包含相关模块的地址。应装载上述对应的组织块，更换有关的模块或排除程序错误。

8）已启用的日期时间中断的启动时间被跳过，例如 PLC 的内部时钟被提前。CPU 将调用 OB80。用 SFC 29 设置实时时钟之前，应禁止使用日期时间中断。

9）因为同时调用了太多的中断 OB，超出了循环时间。CPU 将调用 OB80，如果未装载 OB80，CPU 会切换到 STOP 模式。如果超出了循环时间的两倍，即使装载了 OB80，CPU 仍将切换到 STOP 模式。可以在 CPU 的属性对话框中增大循环时间，或改变程序结构。需要时，可以调用 SFC 43，来重新触发扫描循环时间监视器。

10）编程错误：未加载块、块编号错误、定时器/计数器编号错误、对错误区域进行读/写访问等。CPU 将调用 OB121，应消除编程错误。

11）I/O 访问错误，访问模块数据时出错，CPU 将调用 OB122。

在 HW Config 中检查模块地址，检查模块和 DP 从站是否发生故障。

12）全局数据通信错误，例如，用于全局数据通信的 DB 的长度不够时，CPU 将调用 OB87。

3．SF LED 亮时可能的硬件错误

1）系统处于 RUN 模式时，卸下或插入机架中的模块，CPU 将切换到 STOP 模式。

处理方法：用螺钉拧紧模块并重新启动 CPU。

2）系统处于 RUN 模式时，在 PROFIBUS-DP 上卸下或插入了分布式模块，CPU 将调用 OB86。如果模块通过 GSD 文件集成，将调用 OB82。

3）系统处于 RUN 模式时，在 PROFINET IO 上卸下或插入了分布式模块。CPU 将调用 OB83。系统处于 RUN 模式时，卸下或插入 ET 200S（IO 设备）的模块，将会调用 OB86。

4）具有诊断功能的模块报告诊断中断，CPU 将调用 OB82。应根据模块的组态对诊断事件作出响应。

5）如果在更新过程映像期间试图访问缺失的或有故障的模块，或连接器松动，将调用 OB85，其启动信息包含有关模块的地址。应更换有关模块，紧固插座或排除程序错误。如果程序中用 PI 或 PQ 区地址直接访问有故障的模块，将会调用 OB122。

6）MMC（微存储卡）故障。CPU 将切换为 STOP 模式，并请求存储器复位。

应更换 MMC，复位 CPU 存储器，再次传送程序，然后将 CPU 设置为 RUN 模式。

4．BF LED 常亮的故障与解决的方法

可能出现的问题：

- 总线故障(硬件故障)。
- DP接口故障。
- 多DP主站模式下不同的传输速率。
- DP接口（设置为从站/主站）被激活时总线短路。
- 被动的DP从站接口正在搜索传输速率，即总线上没有其他激活的节点（例如主站）。

以上情况 CPU 将调用 OB86。应检查总线电缆有无短路或断路，查看诊断信息，改正原有的组态。

5．BF LED 闪烁的故障与解决的方法

1）CPU 作 DP 主站可能出现的问题：连接的站有故障、无法访问至少一个已组态的从站、错误的项目组态。CPU 将调用 OB86，应检查总线电缆是否已连接到 CPU，总线是否断开。CPU 启动时如果 LED 不停止闪烁，应检查 DP 从站，或查看 DP 从站的诊断数据。

2）CPU 是活动的 DP 从站，可能的原因：超过了响应监视时间、PROFIBUS-DP 通信中断、错误的 PROFIBUS 地址和错误的项目组态。CPU 将调用 OB86。应检查 CPU、确认总线连接器是否安装正确、检查连接 DP 主站的总线电缆是否断路，检查组态数据和参数。

6．PROFINET 接口故障（BF2/BF3 LED 常亮）

CPU 315-2PN/DP 和 CPU 319-3PN/DP 的 LINK LED（见图 6-1）亮表示 PROFINET 接口的连接处于激活状态，RX/TX LED 亮表示 PROFINET 接口正在接收/发送数据。

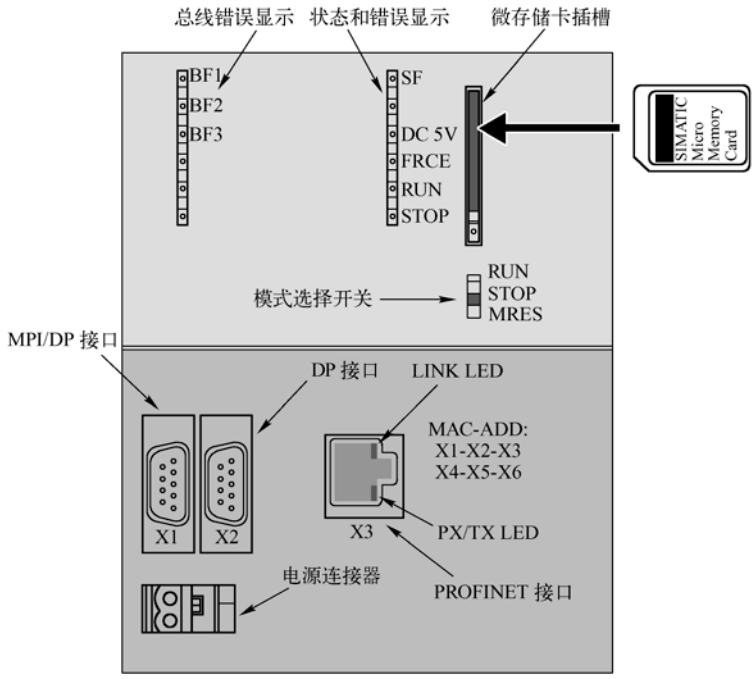


图 6-1　CPU 319-3PN/DP 的面板图

BF2/ BF3 LED 常亮可能出现的问题：PROFINET 接口故障，不能通信，例如，作为 IO 控制器的 CPU 与交换机或子网的连接断开、传输速率错误、未设置全双工模式。

CPU 将调用 OB86，如果未装载 OB86，CPU 将切换到 STOP 模式。

应检查总线电缆有无短路或断路，检查模块是否连接到交换机，而不是集线器。检查数据是否以 100 Mbit/s 的速率在全双工模式下传输。分析诊断数据，改正组态的错误。

7．PROFINET 接口故障（BF2/BF3 LED 闪烁）

可能的问题：连接的 I/O 设备有故障，至少一个已分配的 I/O 设备无法寻址，项目组态错误。CPU 将调用 OB86。

处理方法：应检查以太网电缆是否已连接到模块，或总线是否断开。CPU 启动时，如果 LED 不停止闪烁，应检查 I/O 设备，评估其诊断信息。检验已组态的设备名称是否与实际分配的名称匹配。

8．I/O 设备的 PROFINET 接口故障（BF LED 闪烁）

可能的问题：IP 地址不正确、项目组态错误、参数分配错误。I/O 控制器未找到或已关闭，但是有以太网连接。设备名称错误或不存在、已超过响应监视时间。

处理方法：检查以太网电缆是否正确连接，检查连接到控制器的以太网电缆是否断开，检查组态的数据和参数。将 I/O 设备切换到 I/O 控制器，检查期望的组态与实际的组态是否匹配，检查物理通信连接是否断开。

6.1.2 用 S7-400 CPU 的 LED 进行诊断

1．S7-400 CPU 的 LED 功能简介

表 6-1 列出带 PROFIBUS-DP 接口的 S7-400 CPU 的 LED 的功能。集成有两个 DP 接口的 CPU 有两个对应的 LED（BUS1F 和 BUS2F）。是否有 BUS2F、IFM1F、IFM2F 和 BUS5F 这几个 LED，与 CPU 的型号有关。

表 6-1 带 DP 接口的 S7-400 CPU 的 LED

指示灯	颜色	说　明	指示灯	颜色	说　明
INTF	红色	内部故障，例如用户程序运行超时	BUS1F	红色	1 号通信接口（MPI/ DP）的总线故障
EXTF	红色	外部故障，例如电源或模块故障	BUS2F	红色	2 号通信接口（DP）的总线故障
FRCE	黄色	有输入/输出处于被强制的状态	BUS5F	红色	PROFINET 接口的总线故障，组态了 PROFINET IO 系统但未进行连接
RUN	绿色	运行模式			
STOP	黄色	停止模式	IFM1F	红色	接口子模块 1 故障
CRST	黄色	完全复位（冷启动）	IFM2F	红色	接口子模块 2 故障

CPU 41x-3 有 IFM1F LED，CPU 41x-4 有 IFM1F 和 IFM2F LED，它们用于指示与存储器子模块的第一个和第二个接口有关的故障。CPU 41x-3PN/DP 有 BUS5F LED。

BUS1F、BUS2F、1FM1F、1FM2F 闪烁：CPU 作为主站，DP 接口上的一个或多个从站无响应。CPU 作为从站，不能被 DP 主站访问。

BUS5F 闪烁：PROFINET 接口的一个或多个设备无响应。

CPU 41x-3PN/DP 的 LENK LED 亮表示 PROFINET 接口的连接处于激活状态，RX/TX 以 6Hz 的频率闪烁，表示 PROFINET 接口正在接收或发送数据。

2.INTF LED

INTF LED 亮表示检测到内部错误（编程或参数设置错误），或 CPU 正在执行 CiR（在运行时修改系统），下面是可能的错误原因：

1）超时错误，包括用户程序执行时间（OB1 及所有中断和错误 OB 的执行时间）超过指定的最大循环时间、OB 请求错误、启动信息缓冲区溢出、监控定时器中断、CiR 之后恢复为 RUN 模式。出现超时错误时，将调用时间错误组织块 OB80。

2）CPU 硬件故障，当 CPU 检测到 MPI 网络的接口故障、通信总线的接口故障或分布式 I/O 网卡的接口故障，或故障消失时，将调用 CPU 硬件故障组织块 OB84。

4）通信错误，接收到全局数据时，检测到错误的帧标识符（ID）、帧长度错误、共享数据帧结构错误、全局数据的状态信息数据块不存在或太短，将调用通信错误组织块 OB87。

5）因为同步错误嵌套深度过大、块调用嵌套深度过大、分配本地数据时出错，程序块执行被中止后，将调用处理中止组织块 OB88。

6）有编程错误时，CPU 将调用 OB121。

7）编译的用户程序错误，CPU 切换至 STOP，需要重新启动或复位 CPU 的存储器。

8）如果出现 CPU 硬件错误（检测到存储器错误并且已将它消除），将调用 OB84。如果没有下载 OB84，CPU 保持 RUN 模式。出现其他错误或故障时，如果没有下载相应的组织块，CPU 将会进入 STOP 模式。

3．EXTF LED

EXTF LED 亮表示检测到与 CPU 模块无关的外部错误或故障，下面是可能的错误原因。

1）I/O 访问错误，可能是模块故障，或者访问了一个 CPU 不能识别的 I/O 地址，将调用访问错误组织块 OB122。

2）电源模块故障（非电力网故障）出现或消失，电源模块的备用电池失效或未安装，机架上的 DC 24V 电源故障，将调用电源错误组织块 OB81。

3）诊断中断出现或消失，例如有诊断功能的 I/O 模块的断线故障，模拟量输入模块的电源故障，输入信号超过模拟量模块的测量范围等。出现诊断中断时将调用诊断中断组织块 OB82。

4）插入/拔出模块中断，S7-400 可以在 RUN、STOP 或 STARTUP 模式下带电拔出和插入模块，上述操作将会产生插入/拔出模块中断，CPU 将调用 OB83。

5）优先级错误，产生了一个中断事件，但是对应的 OB 块没有下载到 CPU；访问一个系统功能块的背景数据块时出错；刷新过程映像表时 I/O 访问出错，模块不存在或有故障。出现上述错误时将调用优先级错误组织块 OB85。

6）机架/站故障：扩展机架故障，DP 主站系统故障或分布式 I/O 的故障。故障产生和故障消失时都会产生中断，操作系统将调用机架故障组织块 OB86。

4．与通信有关的 LED

有通信有关的 LED 的意义见表 6-2 和表 6-3。用作 DP 从站的 CPU 41x 的 BUSF LED 闪烁时，表示不能被 DP 主站寻址。

表 6-2　作 DP 主站的 CPU 41x 的 BUSF LED 的含义

BUSF	含　义	措　施
熄灭	组态正确，所有组态的从站均可以寻址	
亮	总线硬件故障，DP 接口故障，多 DP 主站模式的传输速率不一致	检查总线电缆有无短路或中断，查看诊断信息，重新组态或改正组态
闪烁	一个或多个从站故障，至少一个已分配的从站无法寻址	检查总线电缆是否已连接到 CPU 41x，或者总线是否中断。等待 CPU 41x 完成启动，如果 LED 不停止闪烁，则检查 DP 从站或分析 DP 从站的诊断数据

表 6-3　接口子模块故障 LED IFM1F 和 IFM2F

IFM2F 或 IFM2F	含　义	
亮	在接口子模块 1 或接口子模块 2 上检测到错误	
闪烁	DP 主站	插入插槽 1 或插槽 2 的 DP 接口模块上的一个或多个从站无响应
	DP 从站	插入插槽 1 或插槽 2 的 DP 接口模块不能被 DP 主站访问

6.1.3　用 DP 从站的 LED 进行诊断

DP 从站模块也有用于指示 DP 从站的运行状态和故障的 LED。LED 的数量和它们的含义与从站的类型有关。详细的信息请参考各种 DP 从站的用户手册。

ET 200M 的接口模块 IM 153-2 的 LED 的意义如表 6-4 和表 6-5 所示。

表 6-4　IM 153-2 的状态和故障 LED

LED	含　义	LED	含　义
ON(绿色)	供电电压正常	BF(红色)	PROFIBUS 故障
SF(红色)	组错误	ACT(黄色)	冗余模式中的主动模块

表 6-5　IM 153-2 的 LED 的组合意义

SF	BF	ACT	ON	含　义	措　施
熄灭	熄灭	熄灭	熄灭	IM 153-2 没有通电，或模块有硬件故障	接通电源模块或更换 IM 153-2
无关	无关	无关	亮	IM 153-2 通电，运行状态	
亮	熄灭	熄灭	熄灭	模块通电后正在硬件复位	
亮	亮	亮	亮	通电后的硬件测试	
亮	闪烁 0.5Hz	熄灭	熄灭	外部故障，例如使用了不合适的操作系统或微存储卡 MMC	使用合适的用于更新的操作系统，更新期间不要取出 IM 153-2Bx00 的 MMC
亮	闪烁 2Hz	熄灭	熄灭	内部错误，例如在写入更新文件期间的内部错误	重复更新过程。如果 LED 再次指示错误，则内部存储器损坏
无关	闪烁	熄灭	亮	模块未正确组态，DP 主站和模块之间没有数据交换。原因：站地址不正确、总线故障	检查 IM 模块，检查组态和参数，检查 IM 模块和 STEP 7 项目中的站地址。检查电缆长度和终端电阻设置，检查波特率是否匹配
无关	亮	熄灭	亮	与 DP 主站无连接(搜索波特率)。原因：DP 到 IM 153-2 的总线通信已中断	检查是否正确安装了总线连接器、电缆/光缆与 DP 主站的连接是否中断。断开电源模块上的 DC 24V 开关，然后重新接通
亮	闪烁	熄灭	亮	组态的 ET 200M 与 ET 200M 的实际结构不一致	检查 ET 200M 的组态，确定模块是否未插入或有故障，是否有未组态的模块
亮	熄灭	熄灭	亮	无效的 PROFIBUS 地址。如果 SM/FM 的 SF LED 同时点亮，S7-300 模块有故障或诊断事件。否则 IM 153-2 有故障	在 IM 153-2 上设置有效的 PROFIBUS 地址（1～125）。通过诊断检查 SM/FM，更换 S7-300 模块或 IM 153-2
无关	熄灭	亮	亮	IM 153-2 正在与 DP 主站和 ET 200M 的 I/O 模块交换数据。在冗余模式，IM 153-2 是 ET 200M 中的主动的模块	
无关	熄灭	熄灭	亮	电压已供给 IM 153-2。IM 153-2 是冗余模式的被动的模块，它与 I/O 模块没有交换数据	
闪烁 0.5Hz	熄灭	熄灭	亮	在冗余模式， IM 153-2 是被动的模块，未做好无扰动切换的准备	将容错系统切换到冗余状态
闪烁	闪烁	闪烁	闪烁	当前运行模式的 IM 153-2 与冗余 IM 153-2 不兼容	

6.2 使用 STEP 7 进行诊断

6.2.1 故障诊断的步骤

习惯上将 STEP 7 称为编程软件，西门子称之为标准工具。实际上 STEP 7 的功能已经远远超越了编程软件的范畴。STEP 7 用于对整个控制系统（包括 PLC、远程 I/O、HMI、驱动装置和通信网络等）组态、编程和监控。

STEP 7 提供了用于诊断的强大的在线功能，本节将通过实例，详细介绍怎样用这些诊断功能来诊断 PROFIBUS-DP，这些诊断功能也可以用于 PROFINET。

1．故障诊断的步骤

硬件故障和通信故障可以按下面的步骤进行诊断：

1）将项目的组态信息下载到 CPU，执行菜单命令“查看”→“在线”，打开项目的在线视图。

2）打开所有的站点，查看其中组态的可编程模块（例如 CPU 和 CP）的状态。

3）检查哪个 CPU 模块上有诊断符号。选中有诊断符号的模块，按〈F1〉键，可以打开解释诊断符号的帮助页面。

4）选中要检查的站点。

5）执行菜单命令“PLC”→“诊断/设置”→“模块信息”，显示该站 CPU 的模块信息。

6）执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，显示该站 CPU 和有故障的模块的快速视图。

7）选中快速视图中的 CPU、某个有故障的模块或 DP 从站，点击“模块信息”按钮，以获取该模块的信息。双击某个模块或 DP 从站，也可以打开它的“模块信息”对话框。

8）点击快速视图中的“打开在线站点”按钮，打开诊断视图。诊断视图包括站点内按插槽顺序排列的所有模块。

9）双击诊断视图中有故障的模块，打开它的模块信息对话框。

当然不必执行上述的所有步骤；一旦获得所需的诊断信息，即可停止。不同的情况可以采用不同的诊断步骤。例如下载程序后 CPU 不能切换到 RUN 模式，如果估计是编程错误造成的，可以首先查看 CPU 的模块信息对话框中的诊断缓冲区。

2．模块的诊断符号

打开在线视图、快速视图和诊断视图，可以看到模块或 DP 从站上的诊断符号，诊断符号用于判断模块的状态（见图 6-2）。双击快速视图或诊断视图中的诊断符号，可以打开模块或 DP 从站的“模块信息”对话框，显示详细的诊断信息。

图 6-2 诊断符号

下面是图 6-2 中各诊断符号的意义：

1）模块出现故障。可能的原因：诊断中断、I/O 访问错误或检测到错误 LED。

2）预设定（下载的）组态和实际组态不一样，已组态的模块不存在或已断电，通信中断，或插入了与组态的模块不同类型的模块。

3）不能进行诊断，没有在线连接，或 CPU 没有将诊断信息返回模块（例如电源或子模块）。

4）启动。

5）停止。

6）在多值计算操作中，由另一个 CPU 的 STOP 模式触发的停止。

7）运行。

8）该模块有变量被强制，即该模块的用户程序中的变量分配有固定值，这些值不能由程序修改。强制符号可以与其他符号组合出现，此例是强制与运行模式符号的组合。

9）保持。

从 STEP 7 V5.4.1 起，某些 PROFINET 部件可以显示信息，指示是否必须采取预防性维护措施，例如，因为 PROFINET 接口的信号衰减增大，必须更换光纤电缆。下面的维护信息还可以提供维护的紧急程度：

10）维护请求（由一个黄色扳手指示）：必须在可预见的时间段内更换相关的部件。

11）维护要求（由一个橙色扳手指示）：必须马上更换相关部件。

6.2.2 使用可访问节点和在线功能进行诊断

通过网络连接好编程用的计算机和 PLC 后，点击 SIMATIC 管理器工具栏上的按钮，打开“可访问的节点”窗口，将显示网络上可访问的可编程模块（CPU、FM 和 CP 等）。用这种方法可以显示那些没有在 STEP 7 项目中组态的站，或不能用 STEP 7 编程的站，例如编程设备或操作面板。

为了检查 DP 从站的 PROFIBUS 地址设置是否有重叠，或者怀疑网络中的电缆连接有故障，可以使用“可访问节点”功能。

使用在线诊断功能之前，应使 PG/PC（编程器/计算机）接口设置的波特率与网络的波特率一致。启动此功能时，PG/PC 检查通信接口的波特率设置与网络的波特率设置是否相同，总线站的地址是否被重复使用。满足上述条件后，PG/PC 才可以作为一个主动的总线站并被包括在令牌环中。

1．显示 MPI 网络上的可访问站点

执行此功能之前，应使用 PROFIBUS 电缆连接好计算机的通信卡的 DP/MPI 接口和 CPU 的 MPI 接口。或者用 PC/MPI 适配器或 USB/MPI 适配器连接好计算机的通信接口和 CPU 的 MPI 接口。在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PG/PC 接口”，设置 MPI 协议和其他通信参数。

点击 SIMATIC 管理器工具栏上的按钮，打开“可访问的节点”窗口（见图 6-3），显示网络中所有可寻址的可编程模块。该窗口最上面的标题栏出现浅蓝色背景的长条，表示是在线视图。选中某个 PLC 站点的“块”，将在右边窗口显示 CPU 中的系统数据、程序块和数据块。

图 6-3　MPI 网络上的可访问站点

2．显示 PROFIBUS 上的可访问站点

将计算机上的 CP 卡（例如 CP 5611）设置为 PROFIBUS 方式，用 DP 电缆连接通信卡的的 DP/MPI 接口和 CPU 的 DP 接口。或者将 PC/MPI 适配器或 USB/MPI 适配器设置为 PROFIBUS 方式，用适配器连接计算机的通信接口和 CPU 的 DP 接口。

图 6-4 是使用 PC/MPI 适配器读取的可访问站点。适配器必须直接连接到 CPU 的 DP 接口上，不能连接到从站的 DP 接口。

点击工具栏上的“详细资料”按钮，窗口的右边显示站点的详细信息。

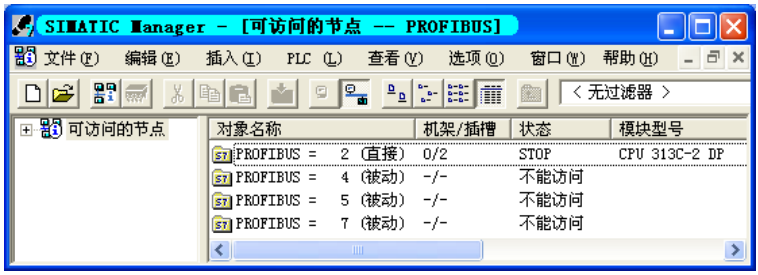


图 6-4　PROFIBUS 网络上的可访问节点

站地址后面的括号中可能的附加信息（见图 6-4）的含义如下：

1）直接：通过 PROFIBUS 电缆或有源总线电缆直接与 PG/PC 连接的总线站。

2）被动：不能通过 PROFIBUS-DP 对该节点进行编程和状态修改。

3）等待：不能与该节点进行通信，因为其组态与网络中其他设置不匹配。

4）模拟：用 PLC 的仿真软件 PLCSIM 模拟的 CPU 站点。

5）附加信息“直接”不支持 PROFINET 节点。

图 6-4 所示系统的 DP 网络上除了有一个作为 DP 主站的 CPU 315-2DP 外，还有 3 个 DP 从站（被动节点）。如果使用 CP 卡，可以显示出从站的模块型号。

3．显示以太网上的可访问站点

首先需要对计算机的以太网网卡（可以是普通的网卡）或 PLC 的以太网通信处理器（CP）组态，设置它们的 MAC 地址或 IP 地址。用电缆和交换机连接好各设备的以太网接口后，在 SIMATIC 管理器点击工具栏上的可访问节点按钮，可以显示出网络上设备的以太网接口的 MAC 地址（见图 6-5）。

图 6-5　以太网上的可访问节点

在基于工业以太网的 PROFINET 网络中，设备用组态时指定的设备名称来识别。图 6-6 给出了 PROFINET 网络上的可访问节点的例子。

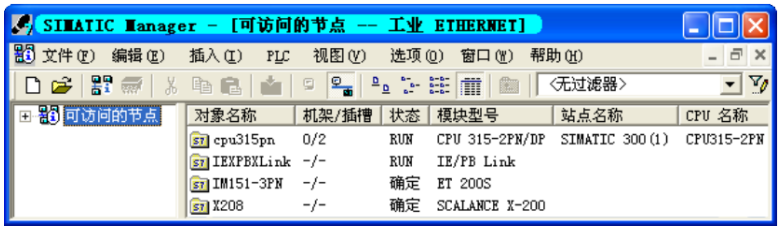


图 6-6　PROFINET 网络上的可访问节点

4．建立在线连接

打开控制系统的 STEP 7 项目后，看到的是离线（OFFLINE）视图，显示的是计算机中的项目信息。通过 MPI、PROFIBUS 或以太网通信，可以打开在线（ONLINE）视图。

用通信硬件（例如 CP 卡）和电缆连接计算机和 PLC，建立起通信连接后，点击 SIMATIC 管理器工具栏上的在线按钮，将打开在线视图。视图最上面的标题栏出现浅蓝色背景的长条，在线视图显示的是通过通信得到的 PLC 中的信息。

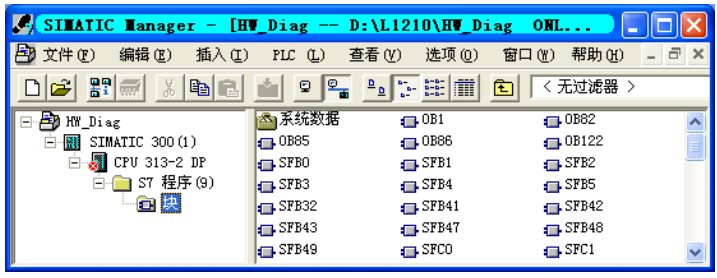


图 6-7　在线视图

图 6-7 的“块”文件夹中的 OB1、OB82 等是下载的用户编写的逻辑块。SFB 0、SFC 0 等是厂家编写并保存在 CPU 中的系统功能块和系统功能。

打开在线视图后，可以用 SIMATIC 管理器工具栏上的按钮和按钮，或者用管理器的“窗口”菜单来切换在线视图和离线视图。可以关闭在线视图。

从模块上的诊断符号可以看出，图 6-7 的在线视图中 CPU 313C-2DP 有故障。

5．用在线视图删除 S7-300 的块

下载到 S7-300 CPU 的块保存在 MMC 卡中，用 CPU 的模式转换开关或 STEP 7 的“PLC”菜单中的命令不能清除下载的块。如果 CPU 中原有的块不能被新下载的块全部覆盖，原有的 OB100 和中断 OB（例如 OB35）可能会在新的项目中继续起作用，程序运行时可能会出现不能预料的结果。选中在线视图中用户下载的程序块，可以用计算机的〈Delete〉键删除它们，但是不能删除固化在 CPU 中的 SFB 和 SFC。

6.2.3　使用快速视图进行诊断

1．创建项目

在 STEP 7 中创建一个名为 HW_Diag 的项目（本章的项目在随书光盘的文件夹

“\Project\PB_Diag”中）。DP 主站为 CPU 313C-2DP（见图 6-8），双击机架中“DP”所在的行，生成 DP 主站系统，将 3 个 DP 从站“拖放”到 DP 网络上。4 号从站为 ET 200B-16DO，5 号从站为 ET 200B-16DI，7 号从站为 ET 200M，它有一块 8DO 模块、一块 16DI 模块和一块 2AO 模块。

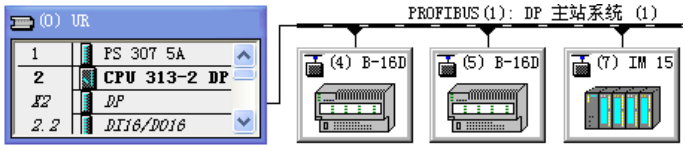


图 6-8　HW Config 中的 DP 网络

2．组态信号模块的诊断功能

信号模块是输入、输出模块的总称，部分信号模块有诊断功能，在模块的属性对话框中设置模块的诊断功能。下面以两通道的 AO（模拟量输出）模块为例，介绍组态和使用信号模块的诊断功能的方法。

选中图 6-8 中的 7 号从站 ET 200M，在下面的窗口的第 6 槽插入一块 2AO 模块，双击该模块，在它的属性对话框的“输出”选项卡中（见图 6-9），设置 0 号通道输出 4～20mA 的电流，1 号通道输出 0～10V 的电压。启用模块的诊断中断功能和两个通道的“组诊断”功能。

图 6-9　2AO 模块的属性对话框

AO 模块的通道被组态为电流输出时，它的输出电阻很大，外部输出回路可以短路，如果开路则出现故障。AO 模块的通道被组态为电压输出时，它的输出电阻很小，外部输出回路可以开路，如果短路则出现故障。

按下计算机的〈F1〉键，在出现的在线帮助中，点击有链接信息的绿色的“诊断”，出现“组诊断”的帮助信息。

由帮助信息可知，组诊断可以检测组态和参数分配错误、电压输出时的接地短路、电流输出时断线和丢失负载电压 L+的故障。出现诊断事件时，CPU 将会调用诊断中断组织块 OB82，同时相应的信息会输入到模块信息的诊断数据区。

3．硬件诊断的设置

在 SIMATIC 管理器中执行菜单命令“选项”→“自定义”，“查看”选项卡中默认的选项是“在硬件诊断期间显示快速视图”，即只显示 CPU 和故障模块。如果未选中该选项，将显示诊断视图（在线的 HW Config），STEP 7 将用较长的时间来计算和显示数据。

4．用硬件诊断的快速视图诊断故障

将程序和组态信息下载到 CPU，用 PROFIBUS 电缆连接主站和从站的 DP 接口，接通它们的电源，将 CPU 切换到 RUN 模式。在 AO 模块的输出端外接两个小开关，来模拟外部输出电路的开路和短路故障。

令 AO 模块 0 号通道的输出电路处于允许的短路状态，1 号通道的输出电路处于允许的开路状态，如果没有其他组态和接线的问题，CPU 和各从站模块的 RUN LED 亮，故障 LED 全部处于熄灭状态。

选中 SIMATIC 管理器左边窗口的 SIMATIC 300 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开“硬件诊断 － 快速查看”对话框，该对话框又称为“快速视图”。因为没有故障模块，所有的 DP 从站运行正常，只能看到处于运行模式的 CPU 模块。

用下面的方法“人为地”制造两个从站的故障：

1）断开 7 号从站 AO 模块 0 号通道输出端外接的小开关，出现电流输出回路开路故障。CPU、IM 153-1 和 AO 模块的 SF LED 亮。

2）断开 4 号从站 ET 200B-16DO 的电源，CPU 的 BF LED 闪烁。

如果没有下载与上述故障有关的组织块 OB82、OB86 和 OB122（见 6.4 节），出现上述故障时 CPU 将进入 STOP 模式。如果下载了上述组织块，CPU 不会进入 STOP 模式。

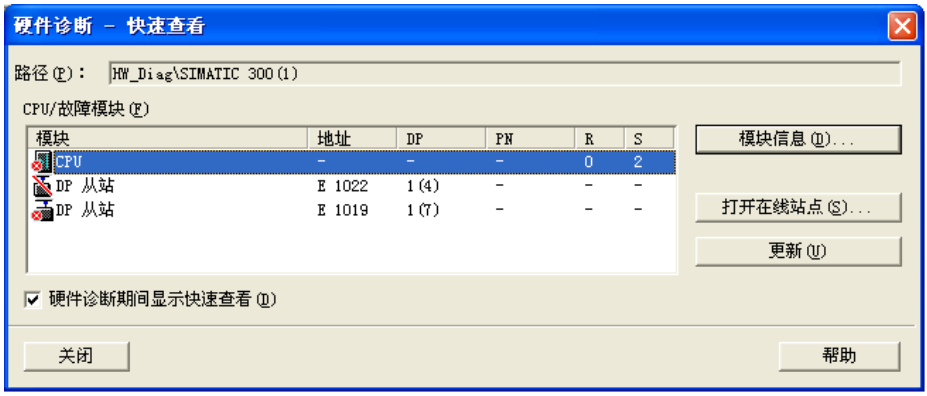


图 6-10　快速视图

打开快速视图（见图 6-10），“CPU/故障模块”列表给出了在线连接的 CPU 和有故障的 DP 从站的诊断符号和参数，例如模块的类型，机架号（R）和插槽号（S）。DP 列中的“1（4）”表示编号为 1 的 PROFIBUS 主站系统中的 4 号从站。PN 是 PROFINET IO 系统的编号和设备号。图中的“E”是德语 Eingang（输入）的缩写。

图 6-10 的 CPU 模块上有一个故障符号（红色的指示灯），4 号从站上的红色斜线表示其“当前组态与实际组态不匹配”，这是因为 4 号从站断电，CPU 不能获取 4 号从站当前的组态信息造成的。

选中快速视图中的某个模块，点击“模块信息”按钮，可以查看该模块的诊断信息。

6.2.4　使用 DP 从站的模块信息进行诊断

1．打开 DP 从站的模块信息对话框

可以用下面的方法打开有故障的 DP 从站的模块信息对话框：

1）双击硬件诊断快速视图中某个有故障的从站（见图 6-10）。

2）打开“诊断视图”（即在线的 HW Config，见图 6-13），双击 DP 网络上的某个从站。

2．查看 4 号从站的模块信息

在 4 号从站断电时，双击快速视图中的 4 号从站（见图 6-10），打开 4 号从站的模块信息对话框（见图 6-11）。在“常规”选项卡，可以看到从站的基本信息。“状态”区显示模块组态的“预设值/实际值不匹配”（插入的模块和组态的模块类型不同），实际上是因为 4 号模块断电，通信被中断造成的。

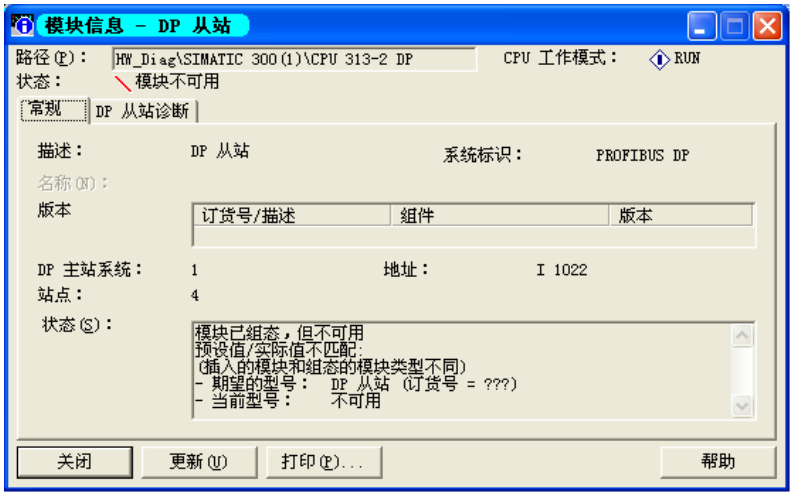


图 6-11　4 号从站的模块信息对话框

“DP 从站诊断”选项卡中的诊断信息为“DP 从站不能通过总线访问”。

3．用模块信息查看从站的模块故障

选中快速视图中的 7 号从站，点击“模块信息”按钮，打开 ET 200M 的接口模块 IM 153-1 的模块信息对话框，“常规”选项卡中的“状态”区的信息为“模块可用且正常，Q256，模块故障（检测到诊断中断）”。在“DP 从站诊断”选项卡（见图 6-12）的“从站的标准诊断”列表中，列出了 DP 从站的常规诊断信息，指出从站的 6 号插槽有故障，响应监视器激活。

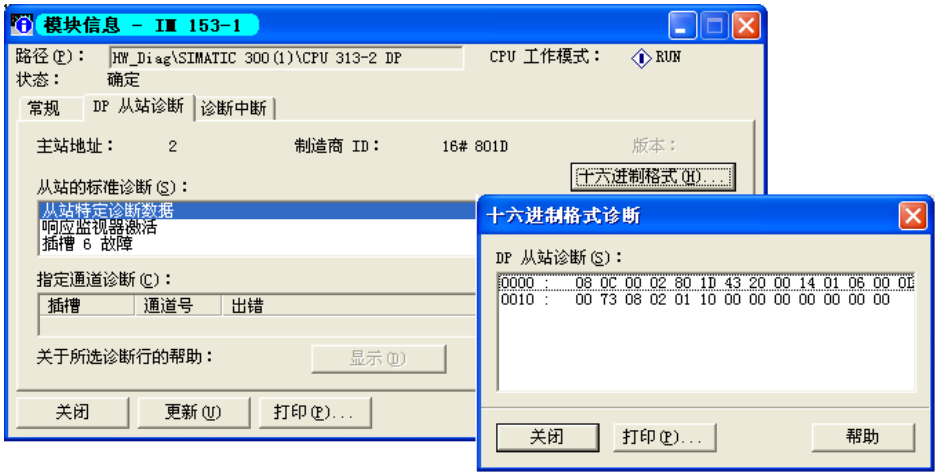


图 6-12　IM 153-1 的模块信息对话框

点击“十六进制格式”按钮，可以查看十六进制格式的完整的诊断报文。

“指定通道诊断”区显示 DP 从站与通道有关的诊断信息，每个诊断信息指出了产生该信息的通道，用模块的插槽号和通道号描述通道。如果选择一个诊断行，然后单击“显示”按钮，将显示所选的标准诊断或通道专用诊断的帮助文本。

每个 DP 从站都有一个响应监视器，使 DP 从站能对 DP 主站的错误或总线上的数据通信中断作出反应。如果在组态的响应监控时间内 DP 从站没有被主站访问，该从站将转入安全状态，所有输出均被置为“0”状态或者输出替换值。

可以在组态从站时关闭响应监视器，但是通信出错时该 DP 从站的输出不会被置为“0”状态，因此强烈建议只在调试期间关闭响应监视器。

4.“诊断中断”选项卡

如果模块支持诊断中断功能，DP 从站的“模块信息”对话框的“诊断中断”选项卡列出了模块的诊断事件信息。只有为模块分配了适当的参数时，才会向 CPU 发送诊断中断。本例程的“诊断中断”选项卡没有诊断信息。

6.2.5 使用诊断视图进行诊断

1．打开诊断视图的方法

诊断视图实际上就是在线的硬件组态窗口。可以用下列 3 种方法打开诊断视图：

1）点击快速视图中的“打开在线站点”按钮（见图 6-10），打开硬件组态的在线诊断视图（见图 6-13），它包含该站机架中所有的模块。

2）点击 SIMATIC 管理器工具栏上的按钮，打开在线视图。选中某个站，双击右边窗口该站的“硬件”图标，可以打开诊断视图。

3）如果已经打开了离线的硬件组态工具 HW Config，点击工具栏上的按钮，也能打开诊断视图。

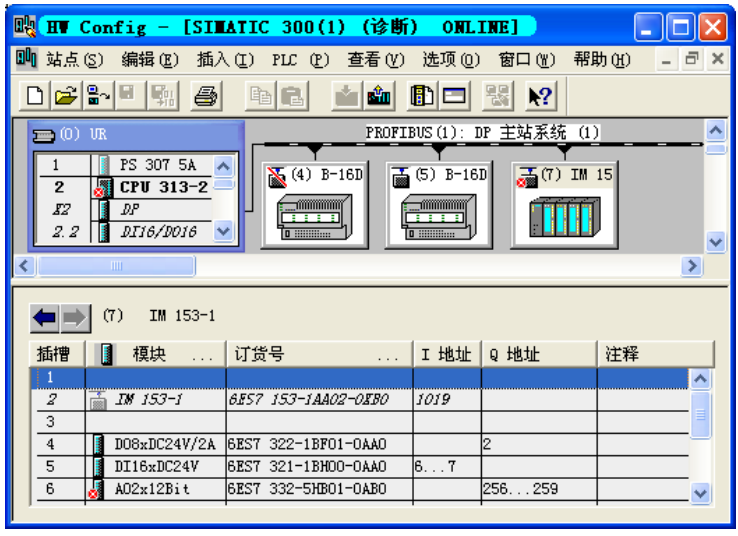


图 6-13 诊断视图

2．诊断视图的功能

与快速视图相比，诊断视图显示整个站在线的组态。包括机架的组态和模块的诊断符号。

可以读取每个模块的在线状态，以及 CPU 模块的操作模式、模块类型、序列号和地址、有关组态的注释。用这种方法可以得到那些没有故障因而没有在快速视图中显示的模块的信息。

双击诊断视图中的某个 DP 从站或从站中的某个模块，打开它的模块信息对话框，可以查看该模块的详细信息。STEP 7 用于 PROFIBUS-DP 组件的诊断方法同样也可供 PROFINET IO 使用，操作步骤是相同的。在线诊断视图还会显示 IE/PB 链接器“下面的”DP 从站。

在诊断视图中执行菜单命令“PLC”→“故障模块”，在打开的对话框中可以看到与故障有关的所有模块（见图 6-14）。选中某个模块后，点击“模块信息”按钮，可以查看该模块的诊断信息。

图 6-14　故障模块对话框

3．诊断视图应用举例

用接在 7 号从站 AO 模块 0 号通道输出端的小开关断开其电流输出电路，用接在 1 号通道输出端的小开关将其电压输出电路短路，将会触发诊断中断，CPU 调用 OB82。CPU、IM 153 和 AO 模块的 SF LED 亮。诊断视图中 CPU、7 号从站和 AO 模块上均有错误符号（见图 6-13）。

选中诊断视图中的 7 号从站，双击下面窗口 6 号槽的 AO 模块，打开 AO 模块的模块信息对话框，“常规”选项卡给出了出现故障的 DP 主站系统编号、从站地址，故障模块的插槽号和输入/输出地址。模块的状态为“模块故障（检测到诊断中断）”。

“诊断中断”选项卡（见图 6-15）给出了模块的标准诊断信息。“指定通道的诊断”列表给出了出现故障的通道编号和具体的错误信息。选中该列表中的通道 0，点击下面的“显示”按钮，出现的帮助信息提示故障的原因是到传感器的连线发生断路。选中列表中的通道 1，其帮助信息提示故障的原因是传感器电源地线发生短路。

图 6-15　AO 模块的模块信息对话框

断开 AO 模块的 DC 24V 电压，AO 模块的模块信息对话框的“诊断中断”选项卡给出的诊断信息为“故障模块，外部出错，无外部辅助电压”。

6.2.6 使用 CPU 的模块信息进行诊断

1．打开 CPU 模块信息对话框的方法

可以用下列 3 种方法打开 CPU 的模块信息对话框：

1）在 SIMATIC 管理器中，选中要检查的站，执行菜单命令“PLC”→“诊断/设置”→“模块信息”。

2）选中快速视图中的 CPU 后，点击“模块信息”按钮（见图 6-10）。

3）双击诊断视图机架中 CPU 所在的行（见图 6-13）。

2．诊断缓冲区

“诊断缓冲区”选项卡提供了最重要的故障和事件的诊断信息（见图 6-16），显示发生的事件总览和选中的事件的详细信息，可以找到使 CPU 进入 STOP 模式的原因。

诊断缓冲区给出了发生的事件一览表，所有的诊断事件和有关的诊断信息都按照它们发生的先后次序存储在诊断缓冲区中。CPU 进入 STOP 模式时，诊断缓冲区的内容仍然保留。使用诊断缓冲区可以对系统的错误进行分析，查找停机原因。甚至可以追溯到很久以前发生的各诊断事件的情况。

图 6-16　CPU 的模块信息对话框

诊断事件包括模块故障、过程写错误、CPU 中的系统错误、CPU 操作模式的切换、用户程序的错误，和用户用系统功能 SFC 52 定义的诊断事件。

断开和接通 4 号从站的电源，打开模块信息对话框。图 6-16 的“事件”列表第二行的“分布式 I/O：站故障”是 4 号从站出现故障（电源丢失）的信息，“事件”列表第一行的“分布式 I/O：站返回”是故障消失的信息。

选中“事件”列表中某一行的事件，下面灰色背景的“关于事件的详细资料”窗口将显示所选事件的详细信息。图 6-17 是“站故障”事件的“关于事件的详细资料”的下半部分。

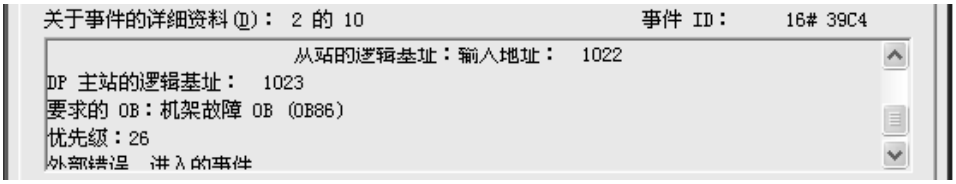


图 6-17　关于事件的详细资料

在诊断缓冲区中，编号为 1、位于最上面的事件是最近发生的事件。选中因编程错误使 CPU 进入 STOP 模式的事件，点击“打开块”按钮，将在程序编辑器中打开与该错误有关的块，显示出错的程序段，光标出现在出错的地方。

选中诊断缓冲区中的某个事件，点击“事件帮助”按钮，将获得有关的帮助信息。

诊断缓冲区是一个环形缓冲区，最大条目数与模块的型号和模块的工作模式有关。

点击“设置”按钮，在出现的对话框中可以设置待读出的条目数，和显示哪些事件等。

3．CPU 的模块信息对话框其他选项卡的功能

模块信息中的“常规”选项卡显示所选模块的标识数据，例如型号、版本、订货号、机架和插槽号等。

“存储器”选项卡给出了所选的 CPU 的工作存储器和装载存储器当前的使用情况，可以检查 CPU 的装载存储器是否有足够的空间来存储新的块。

“扫描循环时间”选项卡用于显示 CPU 的最小循环时间、最大循环时间和当前循环时间。

“时间系统”选项卡显示当前日期、时间、运行的小时数以及时钟同步的信息。

“性能数据”选项卡给出了模块可以使用的 I、Q、M、T、C、OB、SFB 和 SFC 等信息。

“通信”选项卡给出了模块的传输速率、可以建立的连接个数和通信处理占扫描周期的百分比。

模块信息对话框各选项卡的上面显示了附加的信息（见图 6-16），例如模块的在线路径、CPU 的操作模式和状态（例如出错或 OK）。

切换模块信息对话框的选项卡时，从模块中读取数据。但是显示某一选项卡时，其内容不再刷新。点击“更新”按钮，可以在不改变选项卡的情况下从模块读取新的数据。

4．在停机模式下诊断

如果 CPU 在处理用户程序时自动进入 STOP 模式，或下载程序后无法将 CPU 从 STOP 模式切换到 RUN 模式，可以在 STOP 模式建立与 CPU 的在线连接，打开模块信息对话框，根据诊断缓冲区中的信息判断停机的原因。

在 OB1 中输入下面两条语句，第一条语句超出了定时器编号的允许范围：

```
A     T     3000
=     M     100.0
```

下载后将仿真 PLC 切换到 RUN 模式时，PLCSIM 的 CPU 视图对象上的红色 SF LED 亮，不能进入 RUN。

在 SIMATIC 管理器中执行菜单命令“PLC”→“诊断/设置”→“模块信息”，打开 CPU

的模块信息对话框，打开“诊断缓冲区”选项卡（见图 6-18）。

选中第 2 条信息“由编程错误引起的 STOP 模式（OB 没有装载……）”，下面的窗口指出停机原因的详细信息：因为没有下载错误 OB，程序在 OB1 中断。选中第 3 条信息，可以看到发生了定时器编号错误，需要调用 OB121。点击对话框中的“打开块”按钮，将会打开出错的块 OB 1，光标在出错的语句所在的行。

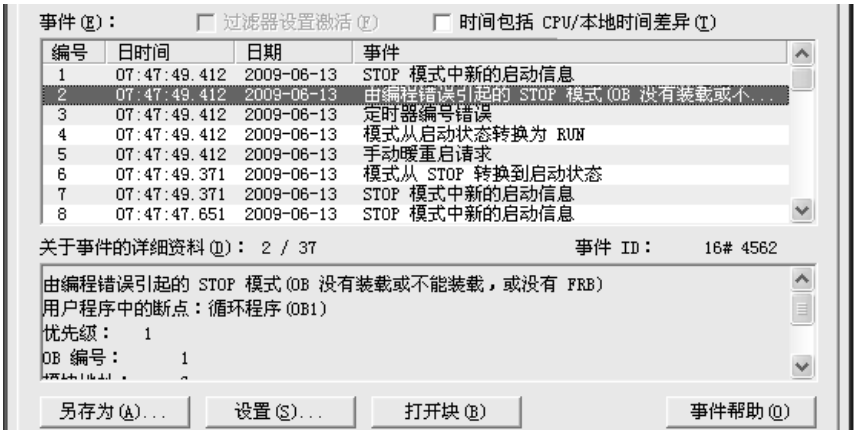


图 6-18　诊断缓冲区

返回 SIMATIC 管理器，生成 OB121（可以是一个空的块），下载后重新运行，可以进入 RUN 模式，但是 SF 灯仍然亮。

5．停机模式下堆栈的内容

“堆栈”选项卡只能在 STOP 模式或 HOLD（保持）模式下调用，可以显示模块的 B 堆栈（块堆栈）、I 堆栈（中断堆栈）、L 堆栈（局部堆栈）以及嵌套深度堆栈。

选中模块信息对话框中的“堆栈”选项卡，通过诊断缓冲区和堆栈中的内容，可以判定用户程序执行过程中引起故障的原因。

例如由于编程错误或停机指令使 CPU 进入停机状态时，可以用“I 堆栈”（中断堆栈）、“L 堆栈”（局部堆栈）和“嵌套堆栈”按钮显示停机时这些堆栈中的内容。堆栈内容将提供哪个块的哪条指令使 CPU 停机的信息。B 堆栈（块堆栈）列出了所有停机前已经被调用但是还没有完全处理的块。

中断堆栈包含中断时的数据或状态，例如累加器和寄存器的内容、打开的数据块及其大小、状态字的内容、优先级（嵌套层次）、被中断的块和中断后程序将继续处理的块。

对于每个在 B 堆栈中列出的块，都可以通过选择某个块并点击“L 堆栈”按钮显示相应的局部数据。它包含中断时用户程序正在处理的块的局部数据。

嵌套堆栈是带左括号的逻辑操作“A(”等使用的存储区域。如果中断时没有处理括号操作，该按钮为灰色（不能操作）。

6.2.7　各种故障诊断方法的比较

本书介绍的故障诊断和故障显示的方法是建立在控制系统的 STEP 7 项目文件的基础上的，它是进行故障诊断的必要条件。必须保证下载到 CPU 的项目文件与运行 STEP 7 的计算机中的项目文件完全相同，才能对控制系统进行监控和故障诊断。

1．使用设备上的 LED 进行诊断

这种诊断方法简单、方便、直观，但是给出的故障信号可能很笼统，需要进一步使用其他诊断方法，例如用 STEP 7 的快速视图、诊断视图和模块信息进行诊断，才能获得具体、准确的诊断信息。如果控制系统的分布范围很宽，查看所有设备的 LED 也很费时费事。

可以在 CPU、CP 和 DP 从站（例如 ET 200）的用户手册中获取用 LED 诊断故障的详细信息。

2．使用 STEP 7 进行诊断

这种诊断方法简便易行，可以迅速地获取准确、详细的诊断信息，CPU 的模块信息的诊断缓冲区提供了错误的文本信息，例如出错的 DP 站地址、出错的模块的地址和故障。应将这种诊断方法作为故障诊断的首选方法。但是需要使用安装了 STEP 7 的计算机，和与 PLC 通信的硬件。此外还要求使用者熟悉 STEP 7，掌握用 STEP 7 进行故障诊断的操作方法。

3．诊断 DP 从站是否与 CPU 正常通信的方法

诊断 DP 网络的故障时，首先需要判断 DP 从站与 CPU 的通信是否正常。可以用下列方法之一进行诊断：

1）使用设备上的故障诊断 LED。

2）使用 STEP 7 的可访问节点功能。

3）使用计算机或 PLC 的通信处理器（例如 CP 5611 和 CP 342-5）的诊断功能。

4）使用 STEP 7 的快速视图、诊断视图和诊断缓冲区。

5）使用 FB 125 或 FC 125。

4．使用 OB86 和 OB82 的局部变量进行诊断

在 OB 块中编写简单的程序（见 6.4 节），用变量表可以监控是否产生中断和产生中断的次数。调用 SFC 20 保存 OB 的 20B 局部变量后，可以获取产生中断的故障信息，查阅 OB 的在线帮助，可以分析局部变量的意义。但是这些信息不一定具体和准确。

5．使用硬件进行诊断

可以用计算机和 PLC 的通信处理器（见 6.5 节）附加的诊断功能诊断故障，但是诊断信息不一定具体和准确。

诊断中继器（见 6.6 节）用于要求较高的控制系统，可以检测故障类型和故障位置（精确到 m），其组态和诊断的准备工作较为麻烦。BT 200 总线测试仪（见 6.6.4 节）多用于系统的安装和调试过程。

6．调用 SFC 13 进行诊断

诊断 DP 网络故障最常用的是 SFC 13（见 7.1 和 7.2 节），考虑到某些从站可能同时出现故障，需要将各从站的故障信息分别保存到不同的存储区，程序较为复杂，要求具有较高的编程水平。诊断数据的长度与 DP 从站的型号、订货号和结构均有关系，需要仔细阅读 DP 从站的用户手册，才能确定 SFC 13 保存诊断数据的地址区的长度。

DP 从站的诊断数据的数据结构和诊断数据各基本单元的具体意义，与从站的型号、订货号、组成从站的模块数量和是否用于冗余系统均有关系。在分析 SFC 13 读取的诊断数据时，必须仔细阅读从站的用户手册，首先了解诊断数据的基本结构，然后搞清每个字、字节、甚至每一位的意义，在大量的数据中找到关键的信息，最后得出故障诊断的结论。因为 DP 从站和从站中的模块的型号很多，分析诊断数据的工作量非常大，并且有一定的难度。

7．调用 FB 125 进行诊断

FB 125（见 7.4 节）是中断驱动的功能块，其背景数据块有一千多个字节，可以用变量表显示其背景数据块中各种状态的从站。可以用手动方式获取某一 DP 从站的详细诊断数据。通过查阅随书光盘中 FB 125 的英语帮助文件《FB125 HELP.chm》，可以较快的得到错误的位置和错误的原因，比人工分析 SFC 13 获取的诊断信息方便一些。分析 FB 125 提供的诊断数据的工作量还是相当大的。

可以在西门子的支持网站下载 FB 125 的英语例程，该例程的库中有用于显示上述 DP 从站故障信息的人机界面的画面。但是同时只能显示一个从站、一个模块和一个通道的详细诊断信息，必须用手动方式切换要诊断的对象。因为使用的是英语，二次开发的工作量很大。

FC 125 是一个较简单的版本，它只提供“哪些站点有故障”的信息，不能显示详细的诊断信息。

8．用报告系统错误功能诊断和显示故障

STEP 7 的“报告系统错误”功能（见 8.2 节）只需要进行简单的组态，几乎可以全部采用默认的参数，就可以自动生成用于诊断故障和发送消息的 OB、FB、SFC 和 DB，以及各机架、从站和模块对应的故障消息，故障的消息文本被自动传送到 HMI 或 WinCC 的项目中。运行时如果出现故障，CPU 将对应的消息编号发送到 HMI 设备或 WinCC，用报警消息的形式显示故障信息。

报警消息是一种比较理想的故障显示方式，可以显示几乎同时出现的多个故障的消息，每条消息包含较丰富的故障信息。

这种诊断方法的组态过程非常简单，诊断和显示用的程序块和程序都是自动生成的，不需要编写故障诊断的程序，生成的消息几乎覆盖了所有的硬件故障和已组态的诊断事件。读取故障信息、分析故障信息和将报警消息发送到 HMI 都是自动完成的。因此这是一种相当理想、极为实用的故障诊断和显示的方法。在有条件的情况下，应作为故障诊断的首选方法。

6.3 使用通信块的输出参数进行诊断

1．通信块用于诊断的输出参数

用于通信的逻辑块有功能（FC）、功能块（FB）、系统功能（SFC）和系统功能块（SFB）。

它们的输出参数 STATUS（状态字）、RET_VAL（返回值）和 ERROR（错误标志位）包含了逻辑块执行后的错误和警告信息。当状态字和错误标志位均为 0 时，没有警告和错误。如果 ERROR 为 0 状态，STATUS 非零，STATUS 中是警告信息。如果 ERROR 为 1，STATUS 提供错误的详细信息。

系统功能 SFC 用返回值 RET_VAL 取代状态字。如果执行 SFC 时出错，其返回值给出错误代码。在块的在线帮助中，可以查阅到各 FB、SFB、FC 的 STATUS 或 SFC 的 RET_VAL 中的故障代码的意义。

2．项目简介

本节使用第 4 章的项目 PB_S7_C，两个 S7-400 站的 CPU 均为 CPU 413-2，在网络组态工具 NetPro 中，建立 MPI 网络上的一个双向 S7 连接，连接 ID 为 1。

通信双方在 OB1 调用 SFB 12“BSEND”，发送 200B 的数据。调用 SFB 13“BRCV”，接收通信伙伴发送的数据。时钟存储器位 M8.0 提供发送请求信号，每 100ms 发送一次数据。

2 号站的 OB100 用于初始化 SFB 的参数，将 BSEND 的发送数据区 DB 1 中的各个字预置为 16#4131，将 BRCV 保存接收数据的地址区 DB 2 清零。3 号站的 OB100 与 2 号站的基本上相同，其区别在于发送数据区的各个字预置的初始值为 16#4132。

3．通信过程的监控

用电缆连接两块 CPU 的 MPI 接口和计算机上的 CP 5613 的 MPI 接口，将程序块和系统数据分别下载到两个 CPU。用电缆连接两块 CPU 的 DP 接口，将 CPU 切换到 RUN 模式。

两台 PLC 进入 RUN 模式后，双方的接收使能位 M0.0 的初始值为 0（false），接收不到对方发送的数据，变量表中除了本机的 ID0 外，其他变量值均为 0（见图 6-19）。

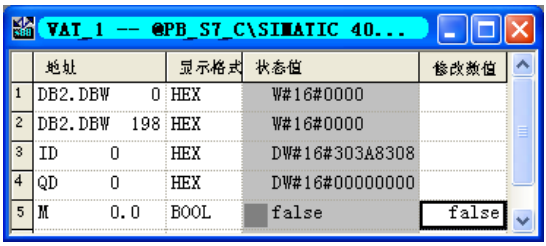


图 6-19　开机后接收使能位为 0 的变量表

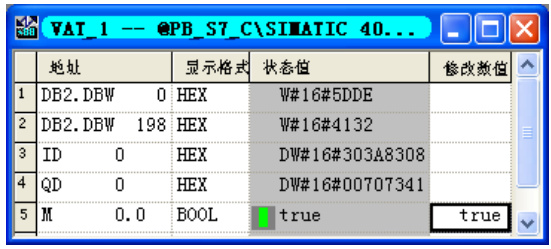


图 6-20　接收使能位为 1 的变量表

在两个站的接收使能位 M0.0 的“修改数值”列输入 1，按回车键后 1 变为 true。点击工具栏上的按钮，“true”被写入 PLC，“状态值”列显示的是 PLC 中变量的值。

在通信双方的 OB35 中，将 DB1.DBW0 加 1，DB 1 中的数据发送给对方的 DB 2。

通信正常时，2 号站的变量表见图 6-20，双方接收到的 DB2.DBW0 的值在不断地变化，用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD0 的状态随之而变。

4. 用 SFB 的输出参数监视通信故障

断开 3 号站的电源，2 号站的 CPU 的 DP EXTF（DP 外部故障）和 DP BUSF（DP 总线故障）LED 亮，通信中止。打开 2 号站的 OB1，点击工具栏上的按钮，在语句表程序的右边出现 SFB 的输入参数（IN）和输出参数（OUT），如图 6-21 和图 6-22 所示。SFB 12“BSEND”和 SFB 13“BRCV”的错误标志位 ERROR 和状态字 STATUS 的值均为 1。

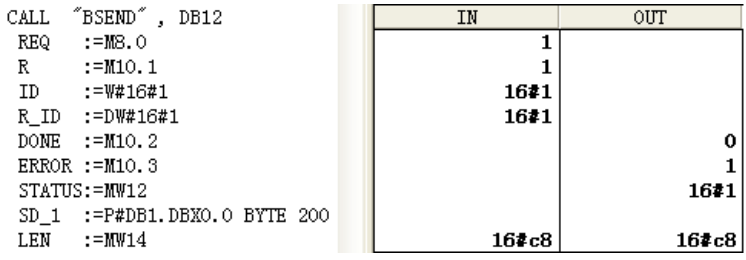


图 6-21　程序状态监控

由在线帮助可知，STATUS 为 1 表示出现了通信故障，例如：

1）本地或远程没有下载组态信息。

2）没有建立到通信伙伴的连接。

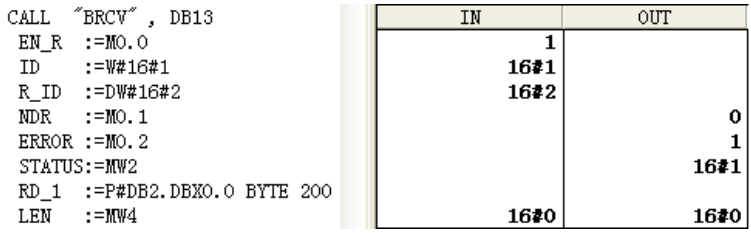


图 6-22　程序状态监控

3）连接被中断，例如电缆断线、CPU 断电，或 CP 处于 STOP 模式。

4）S7-300 超出并行作业/实例的最大数目。

通信正常时拔掉 3 号站一侧的 DP 总线连接器，出现的现象与断开 3 号站的电源时的相同。通信正常时将 3 号站切换到 STOP 模式，或将接收使能位 M0.0 复位为 0，通信中止，但是 DP 故障 LED 不亮，SFB 12 和 SFB 13 的 ERROR 和 STATUS 的值均为 0。

5．用 SFC 的返回值诊断通信错误

3.2.3 节的项目 PB_MS_3 中，CPU 413-2DP 是 DP 主站，CPU 313C-2DP 是智能从站。在 OB1 中调用 SFC 15“DPWR_DAT”，将数据“打包”后发送，调用 SFC 14“DPRD_DAT”，将接收到的数据“解包”。

打开主站的 OB1，在运行时点击工具栏上的按钮，进入程序监控状态。断开智能从站的电源，SFC 14 的返回值 RET_VAL 为 16#80a0（见图 6-23），SFC 15 的返回值为 16#80a1（见图 6-24）。由在线帮助可知，上述返回值表示在访问 I/O 设备时检测到访问错误。

图 6-23　程序状态监控

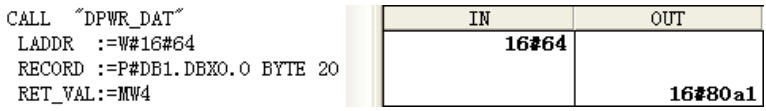


图 6-24　程序状态监控

将 CPU 313C 通电后，由 STOP 模式切换到 RUN 模式，CPU 413-2DP 的 LED 正常。SFC 14 和 SFC 15 的返回值 RET_VAL 为 0，表示故障消失。

6.4　中断组织块在故障诊断中的应用

6.4.1　与 DP 通信有关的中断组织块

CPU 在识别到一个故障或编程错误，例如，DP 从站或者 PROFINET I/O 设备的诊断报警、站的故障等，将会调用对应的中断组织块（OB），应生成这些 OB，通过 OB 中编写的程序对故障进行处理。如果这些组织块没有下载到 CPU，CPU 将会因为无法调用这些块而进入 STOP 状态。下面介绍与通信故障有关的几个主要的中断组织块。

1．DP 从站产生的诊断中断（OB82）

具有诊断功能的分布式 I/O 模块通过产生诊断中断来报告事件，例如部分节点故障、信号模块导线断开、I/O 通道的短路或过载、模拟量模块的电源故障等。产生诊断中断时，CPU 的操作系统将自动调用处理诊断中断的组织块 OB82。OB82 的启动信息提供了产生故障的模块的类型（输入模块或输出模块）、模块的地址和故障的种类。当 DP 主站的 CPU 从 RUN 模式切换到 STOP 模式时，智能从站将调用诊断中断组织块 OB82。故障出现和消失时将分别调用一次 OB82。

通过在 OB82 中调用故障诊断的程序块，可以判断在哪个通道发生了什么样的故障。

2．外设输入/输出区与过程映像输入/输出区

S7-300/400 的外设输入/输出区（PI/PQ 区）用于直接读写 I/O 模块。过程映像输入/输出区（I/Q 区）是输入/输出模块在 CPU 的存储区中的“映像”。在每一扫描循环周期开始时，CPU 将过程映像输出区中的数据成批地传送到输出模块，将输入模块外接的输入电路的状态成批地读入过程映像输入区。

PI/PQ 区与 I/Q 区的关系如下：

1）访问 PI/PQ 区时，直接读写输入/输出模块，而 I/Q 区是 CPU 内的存储区。

2）I/Q 区可以按位、字节、字和双字寻址，PI/PQ 区不能按位寻址。

3）I/Q 区的地址也可以用 PI/PQ 区访问。

3．优先级错误中断（OB85）

以下情况将会触发优先级错误中断：

1）产生了一个中断事件，但是没有将对应的 OB 块下载到 CPU（不包括 OB81）。

2）操作系统访问模块时出错。

3）由于通信或组态的原因，模块不存在或有故障，刷新过程映像表时 I/O 访问出错。出现故障的 DP 从站的输入/输出值装入 S7 CPU 的过程映像表时，就可能出现上述情况。

访问出错的输入字节被复位和保持为“0”，直到故障消失。

双击 HW Config 的机架中的 CPU，打开 CPU 的属性对话框。可以用“周期/时钟存储器”选项卡中的选择框选择调用 OB85 的方式（见图 6-25）。

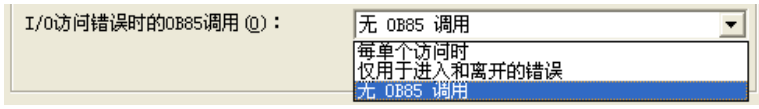


图 6-25　CPU 的属性对话框的周期/时钟存储器选项卡

S7-300 CPU 默认的选项是“无 OB85 调用”，在发生 I/O 访问错误时不调用 OB85，也不会在诊断缓冲区中生成条目，一般采用默认的设置。

S7-400 CPU 默认的选项是“每单个访问时”，在满足条件时，每个扫描循环周期都要调用一次 OB85 和在诊断缓冲区中生成一个条目，这样会增加扫描周期。建议选用“仅用于进入和离开的错误”，该选项只是在错误刚发生和刚消失时分别调用一次 OB85。

在编写 OB85 的程序时，应根据 OB85 的启动信息，判定是哪个模块损坏或没有插入。OB85 的局部变量 OB_85_FLT_ID 的错误代码的意义举例如下：B#16#B1 和 B#16#B2 分别表示更新过程映像输入、输出表时的 I/O 访问错误。可以用 SFC 49“LGC_GADR”查找有关模块所在的机架和槽位，以及模块的用户数据地址区中的偏移量。

4．机架故障或分布式 I/O 的站故障中断（OB86）

如果机架、DP 主站系统或分布式 I/O（DP 从站或 PROFINET IO 设备）由于掉电、总线导线断开、I/O 系统的故障，或者某些其他原因引起的故障，CPU 的操作系统将调用组织块 OB86。故障出现和消失时将分别调用一次 OB86。可以根据 OB86 的启动信息，在 OB86 中编程，确定是哪个机架或分布式设备有故障或通信中断。

5．I/O 访问错误中断（OB122）

CPU 如果访问有故障的 I/O 模块、不存在的或有故障的 DP 从站的 PI/PQ 输入/输出数据，或者访问了一个 CPU 不能识别的 I/O 地址，CPU 的操作系统将在每个扫描周期调用一次 OB122。

6．故障处理中断组织块的作用

DP 从站出现故障时，如果 S7-400 没有生成和下载 OB82、OB85、OB86 和 OB122，S7-300 没有生成和下载 OB82、OB86 和 OB122，CPU 将切换到 STOP 状态。为了防止某个从站的故障造成整个 PROFIBUS 主站系统停机，作为一个常规的措施，至少要生成和下载上述组织块。即使没有在这些 OB 中编写任何程序，在 DP 从站出现上述故障时，CPU 也不会进入 STOP 模式。

需要注意的是，生成上述 OB 后，CPU 虽然不再进入 STOP 模式，但是可能不易察觉这些危险状态，它们会被忽视。为了解决这一问题，在故障 OB 中，应编写记录、处理和显示故障的程序，例如记录中断的次数，保存 OB 的局部变量，调用读取诊断数据的 SFC 13 等。以便在出现故障时，迅速地查明故障的原因和采取相应的措施。

通过中断组织块的局部变量提供的信息，可以获得故障的原因、出现故障的模块地址、模块的类型（输入模块或输出模块）、故障出现或故障消失等信息。CPU 的模块信息对话框中的诊断缓冲区保留着 CPU 曾经调用过的组织块的信息。

最好在出错时通过监控设备产生一条报警信息，以便操作人员安全和正确地操作设备，具体的方法将在第 8 章介绍。

中断组织块的详细信息可以参阅 STEP 7 的在线帮助，或者参考随书光盘中的手册《用于 S7 的系统软件和标准功能参考手册》。

6.4.2 与 DP 通信有关的中断组织块的实验

1．硬件结构

在 SIMATIC 管理器中创建一个名为 OB_Diag1 的项目，CPU 为 CPU 413-2DP。在 HW Config 中生成 DP 主站系统（见图 6-26），ET 200B-16DO、ET 200B-16DI 和 ET 200M 分别是 4 号、5 号和 7 号从站。

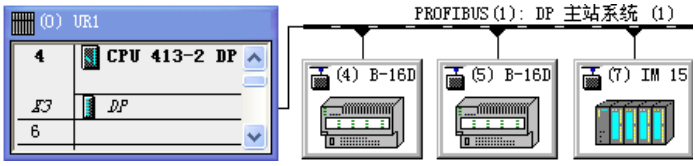


图 6-26　HW Config 中的 DP 网络

将程序块和组态信息下载到 CPU，用 PROFIBUS 电缆连接主站和从站的 DP 接口，接通它们的电源，将 CPU 切换到 RUN 模式。

2．用变量表监视产生中断的次数

在 SIMATIC 管理器中生成 OB82、OB85、OB86 和 OB122。为它们编写简单的程序，每次中断分别将 MW10、MW12、MW14 和 MW16 加 1。下面是 OB82 中的程序：

程序段 1：MW10 加 1

```
L     MW    10
+     1
T     MW    10
```

运行时用变量表监控中断的次数（见图 6-27）。

图 6-27　用变量表监控中断的次数

3．从站电源丢失时调用 OB85

将硬件组态信息下载到 CPU，将它切换到 RUN 模式，系统运行正常，错误指示 LED 全部处于熄灭状态。

断开 4 号从站的电源，CPU 模块上的 EXTF（外部故障）和 DP EXTF LED 亮，DP BUSF（总线故障）LED 闪烁。每个扫描循环周期调用一次 OB85，MW12 被加 1，变量表中 MW12 的值快速增大。即使用户程序没有访问有故障的从站，也会调用 OB85。

在 SIMATIC 管理器中，选中 400 站点，执行菜单命令“PLC”→“诊断/设置”→“模块信息”，打开 CPU 的模块信息对话框。在“诊断缓冲区”选项卡（见图 6-28），选中“事件”列表第 2 行的“在将过程映像传送到输出模块时的 I/O 访问错误”，在下面的“关于事件的详细资料”文本框中，可以看到下列信息：访问 4 号从站的过程映像输出字 QW12，过程映像分区编号为 0，需要调用程序执行错误组织块 OB85，中断的优先级为 26，外部错误，进入的事件（事件发生）。

点击“事件帮助”按钮，可以看到该事件的帮助信息：“在更新过程映像输出表时出错。详细信息包含引起故障的 I/O 地址。硬件配置可能不正确，或模块没有插入，或处于故障状态，或机架不能工作”。S7-400“在更新长度大于 32B 的一致性用户数据的过程映像时，如果时间间隔太短，那么会发生临时错误。在这种情况下，将不执行更新。”

图 6-28　CPU 的诊断缓冲区

如果没有生成和下载 OB85，将 4 号从站断电时，CPU 将进入 STOP 模式。反之不会进入 STOP 模式。S7-300 作主站时，同样的情况不会调用 OB85。

4．从站电源丢失时调用 OB122

在例程 OB_Diag 的 OB1 编写下面两条语句，将 MW4 的值写入 4 号从站（ET 200B-16DO）的外设输出字 PQW12。

```
L     MW  4
T     PQW   12
```

断开 4 号从站的电源，CPU 模块上的 EXTF 和 DP EXTF LED 亮，DP BUSF（DP 总线故障）LED 闪烁。每个扫描循环周期都要调用一次 OB122 和 OB85，变量表中用来计 OB122 中断次数的 MW16 的值快速增大。如果用户程序没有用外设变量（PI/PQ）直接访问断电的从站，不会调用 OB122。在 CPU 的模块信息对话框的“诊断缓冲区”的“事件”列表中，选中信息“在第 n 次（n > 1）写入访问时发生 I/O 访问错误”，“关于事件的详细资料”如图 6-29 所示。

点击“事件帮助”按钮，帮助信息给出的原因是“用户程序不止一次尝试写入到一个不存在的 I/O 地址”。纠正或避免出错的方法为“进行检查，如有必要，更正 I/O 地址；在模块取出的情况下，则插入模块；在机架不能工作的情况下，则使机架开始工作”。

点击“打开块”按钮，自动打开出现 I/O 访问错误的 OB1 中的程序段 1，光标位于访问出错的外设输出地址 PQW12 所在的行。

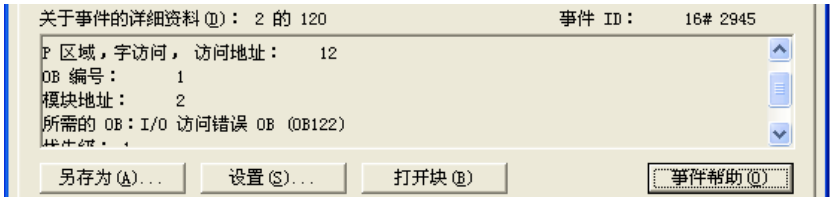


图 6-29　CPU 的诊断缓冲区中事件的详细资料

如果没有生成和下载 OB122，将 4 号从站断电，CPU 将进入 STOP 模式。生成和下载 OB122 后，将某个从站断电，CPU 不会进入 STOP 模式。

5．从站电源丢失时调用 OB86

用变量表监控用来计 OB86 中断次数的 MW16，可以看到，断开 4 号从站的电源，调用一次 OB86，CPU 模块上的 EXTF 和 DP EXTF LED 亮，DP BUSF LED 闪烁。

从站电源断电时，S7-400 CPU 每个循环扫描周期都要调用一次 OB85。如果用 PI/PQ 地址访问了出错的从站地址，每个循环扫描周期还要调用一次 OB122，CPU 的诊断缓冲区全部被调用 OB85 和 OB122 的事件占据，看不到断电时调用 OB86 的事件。

S7-300 的 CPU 如果没有用 PI/PQ 地址访问出错的从站地址，不会调用 OB85 和 OB122，可以看到诊断缓冲区中调用 OB86 的事件。

例程中的 3 个从站共用 7 号从站的电源。同时断开 3 个从站的电源，CPU 调用 3 次 OB86。同时接通 3 个从站的电源，CPU 又调用 3 次 OB86。

如果没有生成和下载 OB86，将某个从站断电时，CPU 进入 STOP 模式。

6．AO 模块出现故障时调用 OB82

硬件组态时选中 7 号从站 ET 200M，在下面的窗口第 6 槽插入一块 2AO 模块，在它的属性对话框的“输出”选项卡中，启用模块的诊断中断功能和 0 号通道的“组诊断”功能（见图 6-9）。将 0 号通道设为输出 4～20mA 的电流。

在 AO 模块 0 号通道的输出端外接一个小开关，将开关断开，模块的输出回路出现开路故障。CPU 的 EXTF 和 DP EXTF LED 亮，IM 153-1 和 AO 模块的 SF LED 亮。

变量表中的 MW10 的值加 1，表明调用了一次 OB82。在 CPU 的模块信息对话框诊断缓冲区的“事件”列表中，可以看到调用 OB82 的信息，触发中断的模块的地址和其他信息。

接通接在 AO 模块输出端的小开关，电流输出电路断开的故障消失，各模块的故障 LED 熄灭。CPU 又调用一次 OB82，MW10 的值加 1。在 CPU 的模块信息对话框中，可以看到有关的信息。

7．S7-300 出现通信故障的实验

例程 PB_MS_7 的 3 个 DP 从站与例程 OB_Diag1 的相同，CPU 为 CPU 313C-2DP，4 号从站 ET 200B-16DO 的地址为 QW0。断开某个从站的电源时，不能访问该从站的过程映像输入/输出区（I/Q 区），如果采用默认的设置，出现 I/O 访问错误时 CPU 不会调用 OB85，S7-300 可以不生成和下载 OB85。

断开 4 号从站的电源，CPU 的 SF LED 亮，BF LED 闪烁，调用一次 OB86。

接通 4 号从站的电源，故障消失，又调用一次 OB86，CPU 上的故障显示 LED 熄灭。选中 CPU 模块信息对话框中的事件信息“分布式 I/O：站返回”， 在下面的“关于事件的详细资料”文本框中，给出了信息“外部错误，离开的事件”（故障消失）。

如果用户程序用外设地址 PQW0 直接访问 4 号从站，断开 4 号从站的电源时，每个循环扫描周期调用一次 OB122。接通 4 号从站的电源时，停止调用 OB122。

与项目 OB_Diag1 相同，断开 7 号从站（ET 200M）的 AO 模块 0 号通道的输出电路，出现电流输出电路开路的故障，调用一次诊断中断组织块 OB82，AO 模块、IM 153-1 和 CPU 的 SF LED 亮。接通 AO 模块 0 号通道的输出电路，故障消失，错误显示 LED 熄灭，又调用一次 OB82。

6.4.3 使用 OB86 和 OB82 的局部变量进行诊断

处理故障的中断组织块的局部变量包含大量的故障信息，本节介绍用 OB86 和 OB82 的局部变量诊断 DP 从站的方法。

1．项目介绍

在 STEP 7 中打开上一节的项目 OB_Diag1，CPU 为 CPU 413-2DP，ET 200B-16 DO、ET 200B-16 DI 和 ET 200M 分别是 4 号、5 号和 7 号从站。

在 HW Config 中，双击机架中 DP 所在的行，打开 CPU 的 DP 接口属性对话框，在地址选项卡可以看到 DP 接口的诊断地址为 2044。选中 PROFIBUS 网络，在下面的窗口可以看到各个从站的诊断地址（见图 6-30）。双击某个从站的图标，在打开的 DP 从站属性对话框的“常规”选项卡中，也可以找到该从站的诊断地址。

在 SIMATIC 管理器中打开“块”文件夹，用鼠标右键点击右边的“块”工作区，执行快捷菜单中的命令“插入新对象”→“组织块”，生成 OB 82、OB85、OB86 和 OB122，分别在

这些组织块中编写将 MW10～MW16 加 1 的程序，用来记录调用 OB 的次数。

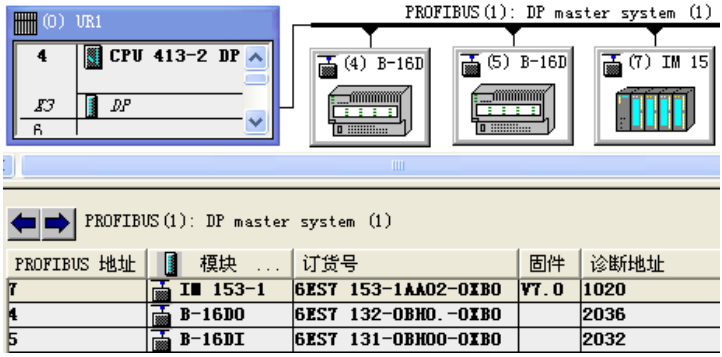


图 6-30　HW Config 中的 DP 网络

2．组织块的变量声明表

组织块（OB）是用户程序与 CPU 的操作系统的接口，OB 不能被用户程序调用。它是在触发 OB 的事件出现时，由操作系统调用的。OB 的变量声明表中只有临时变量。

操作系统为所有的 OB 块声明了一个包含 OB 的启动信息的 20B 的变量声明表（见表 6-6），声明表中变量的具体内容与组织块的类型有关。用户可以通过 OB 的变量声明表获得与启动 OB 的原因有关的信息。

表 6-6　OB 的变量声明表

字 节 地 址	内　　容
0	事件级别与标识符
1	用代码表示与启动 OB 的事件有关的信息
2	OB 的优先级
3	OB 块的编号
4～11	附加信息
12～19	OB 被启动的日期和时间（年、月、日、时、分、秒、毫秒与星期）

3．OB86 的局部变量

如果机架或 DP 从站发生故障，在故障出现和消失时，CPU 都会自动调用 OB86。

1）OB86_EV_CLASS 为 B#16#39 时表示故障刚出现，为 B#16#38 时表示故障刚消失。

2）下面是与 DP 通信有关的故障代码 OB86_FLT_ID 的值：

- B#16#C3：分布式I/O设备的DP主站系统故障。
- B#16#C4：DP站故障。
- B#16#C5：DP站内部的故障。

3）OB86_MDL_ADDR 为 DP 主站的逻辑基地址，它是 HW Config 中主站的 DP 接口的诊断地址，可以在 DP 接口属性对话框的“地址”选项卡找到它。CPU 的操作系统使用该地址来报告该接口的故障。

4）OB86_RACKS_FLTD：其数据类型为 32 个位元素的数组（Array），为了方便编程，可以将它的数据类型更改为 DWORD（双字）。

如果 OB86_FLT_ID 为 B#16#C4，且故障刚结束，OB86_RACKS_FLTD 的第 0～7 位为出错的 DP 站的编号，第 8～15 位为 DP 主站系统的编号（ID），第 16～30 位为 S7 从站的逻辑基地址，即标准 DP 从站的诊断地址，第 31 位为 I/O 标识符。

4．用 OB86 的局部变量诊断 ET 200B 的故障

在 SIMATIC 管理器中生成数据块 DB 1。双击打开它后，生成有 5 个双字元素的数组 ARY。为了用 OB86 的临时局部变量提供的信息来诊断从站的故障，在 OB86 中，调用 SFC 20 将 OB86 的局部变量保存在 DB 1 中。

程序段 2：保存 OB86 的局部变量

```
CALL   "BLKMOV"
 SRCBLK :=P#L 0.0 BYTE 20
 RET_VAL   :=MW54
 DSTBLK    :=DB3.ARY
```

5．OB86 局部变量的意义

在系统运行时断开 4 号从站 ET 200B 的电源，CPU 调用一次 OB86，变量表中用来计 OB86 中断次数的 MW14 加 1（见图 6-31），CPU 413-2DP 的 EXTF 和 DP EXTF LED 亮，BUSF LED 闪烁。

DB 1 中 OB86 的局部变量（见图 6-32）的意义如下：

1）DBB0 的事件等级标识符为 16#39，表示事件发生。

2）DBB1 的错误代码为 16#C4，表示 DP 从站故障。

图 6-31　变量表

地址	名称	类型	初始值	实际值
0.0	ary[0]	DWORD	DW#16#0	DW#16#39C41A56
4.0	ary[1]	DWORD	DW#16#0	DW#16#C05407FC
8.0	ary[2]	DWORD	DW#16#0	DW#16#07F40104
12.0	ary[3]	DWORD	DW#16#0	DW#16#09022214
16.0	ary[4]	DWORD	DW#16#0	DW#16#55177251

图 6-32　DB 1 中 OB86 的局部变量

3）DBB2 的中断优先级为 16#1A（26）。

4）DBB3 的 OB 编号为 16#56（86）。

5）DBW4 保留未用。

6）DBW6 的#07FC（2044）是 DP 主站的诊断地址 OB86_MDL_ADDR。

7）DBD8 为 16#07F40104，表示从站的诊断地址为 16#07F4（2036），DP 主站系统的编号为 1，从站的站地址为 4，与硬件组态中对应的参数相同。

8）DBD12 的 16#09022214 和 DBD16 的 16#55177251 表示事件发生在 2009 年 2 月 22 日 14 点 55 分 17 秒。

接通 4 号从站的电源，CPU 又调用一次 OB86，MW14 加 1，CPU 413-2DP 的 LED 恢复正常。OB86 的局部变量与断开电源时基本上相同，其区别仅在于第一个字节为 16#38，表示事件离开（消失）。

6．保存 OB82 的局部变量

如果具有诊断功能的模块在组态时已经启用了诊断中断，在检测到故障产生和消失时，

它将会向 CPU 发送一个诊断中断请求，操作系统将调用 OB82。

在 SIMATIC 管理器中生成数据块 DB 1，打开它以后生成由 5 个双字组成的名为 ARY 的数组。在 OB82 中调用 SFC 20“BLKMOV”，将 20B 的临时局部数据保存到数组 ARY 中。

7．用 OB82 的局部变量诊断 ET 200M 的故障

用接在 7 号从站的 2AO 模块通道 0 的输出电路的小开关将电流输出电路开路，产生一个诊断中断。CPU 调用一次 OB82，MW10 加 1（见图 6-31）。2AO 模块和 IM 153-1 的 SF LED 亮，CPU 413-2DP 的 EXTF 和 DP EXTF LED 亮。

将 2AO 模块通道 0 的输出电路接通，输出电路开路的故障消失，又产生一个诊断中断，CPU 调用一次 OB82，MW10 加 1，各 LED 的状态恢复正常。图 6-33 是 2AO 模块的输出电路断开时，读取的 OB82 的局部变量的前 12 个字节。

地址	名称	类型	初始值	实际值
0.0	ary[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ary[1]	DWORD	DW#16#0	DW#16#C5550100
8.0	ary[2]	DWORD	DW#16#0	DW#16#0D150000

图 6-33　输出电路开路时 OB82 的局部变量

8．OB82 局部变量的分析

OB82 的局部变量的意义见表 6-7，其中最主要的部分是逻辑基地址和 4 个字节的故障模块的诊断数据。2AO 模块 0 号通道的输出电路断开时，由图 6-33 可以获取下列主要的信息：

DBB0 为 16#39，表示进入事件（事件发生）。

DBB5 为 16#55，故障模块为输出模块。

表 6-7　OB82 的局部变量表

地　址	变　量	数 据 类 型	描　述
0.0	OB82_EV_CLASS	BYTE	事件级别和标识符，B#16#38：离开事件，B#16#39：进入事件
1.0	OB82_FLT_ID	BYTE	错误代码（B#16#42）
2.0	OB82_PRIORITY	BYTE	优先级，可通过 STEP 7 分配（硬件配置）
3.0	OB82_OB_NUMBR	BYTE	OB 编号（82 或 W#16#52）
4.0	OB82_RESERVED_1	BYTE	保留
5.0	OB82_IO_FLAG	BYTE	输入模块：B#16#54，输出模块：B#16#55
6.0	OB82_MDL_ADDR	WORD	发生故障的模块的逻辑基地址
8.0	OB82_MDL_DEFECT	BOOL	模块发生故障
8.1	OB82_INT_FAULT	BOOL	内部故障
8.2	OB82_EXT_FAULT	BOOL	外部故障
8.3	OB82_PNT_INFO	BOOL	通道故障
8.4	OB82_EXT_VOLTAGE	BOOL	外部电压故障
8.5	OB82_FLD_CONNCTR	BOOL	未插入前面板连接器
8.6	OB82_NO_CONFIG	BOOL	未组态模块
8.7	OB82_CONFIG_ERR	BOOL	模块中的参数不正确
9.0	OB82_MDL_TYPE	BYTE	第 0～3 位：模块等级，第 4 位：有通道信息，第 5 位：有用户信息，第 6 位：来自替换者的诊断中断，第 7 位：保留

（续）

地　址	变　量	数 据 类 型	描　述
10.0	OB82_SUB_MDL_ERR	BOOL	子模块丢失或存在错误
10.1	OB82_COMM_FAULT	BOOL	通信故障
10.2	OB82_MDL_STOP	BOOL	操作模式（0：RUN，1：STOP）
10.3	OB82_WTCH_DOG_FLT	BOOL	监控定时器响应
10.4	OB82_INT_PS_FLT	BOOL	内部电源故障
10.5	OB82_PRIM_BATT_FLT	BOOL	电池耗尽
10.6	OB82_BCKUP_BATT_FLT	BOOL	整个备份失败
10.7	OB82_RESERVED_2	BOOL	维护请求
11.0	OB82_RACK_FLT	BOOL	扩展机架故障
11.1	OB82_PROC_FLT	BOOL	处理器故障
11.2	OB82_EPROM_FLT	BOOL	EPROM 故障
11.3	OB82_RAM_FLT	BOOL	RAM 故障
11.4	OB82_ADU_FLT	BOOL	ADC/DAC 错误
11.5	OB82_FUSE_FLT	BOOL	保险丝断开
11.6	OB82_HW_INTR_FLT	BOOL	硬件中断丢失
11.7	OB82_RESERVED_3	BOOL	保留
12.0	OB82_DATE_TIME	DATE_AND_TIME	调用 OB 时的 DATE_AND_TIME

DBW6 为 16#0100（256），是出现故障的 AO 模块的逻辑基地址。

DBB8 为 16#0D 或 2#0000 1101，表示有通道故障、外部故障和模块发生故障。

DBB9 为 16#15 或 2#0001 0101，表示有通道信息，低 4 位为模块种类（模拟量模块）。

输出电路断开时，OB82 的局部变量与输出电路接通时的基本上相同，其区别在于 OB86_EV_CLASS 为 16#38，表示事件消失（离开事件）。此外 DBB8 由 B#16#0D 变为 B#16#00，表示故障消失。

6.5　使用 PROFIBUS 通信处理器进行诊断

6.5.1　使用 PLC 的 PROFIBUS 通信处理器进行诊断

S7-300/400 常用的 PROFIBUS 通信处理器有 CP 443-5 和 CP 342-5、CP 343-5 等，它们扩展了 PLC 的通信接口，其功能比集成的 DP 接口的功能要强得多。它们支持 PG/OP（编程器/操作面板）通信、S7 通信和 S5 兼容的通信（FDL 通信）。此外，这些通信处理器还有非常强的诊断功能，下面以 CP 443-5Ext 为例，介绍用 PROFIBUS 通信处理器诊断故障的方法，其他型号的 CP 诊断故障的方法基本上相同。

1．使用 CP 443-5Ext 的 LED 进行诊断

CP 443-5Ext 的 3 个故障 LED（INTF、EXTF 和 BUSF）均为红色，RUN LED 为绿色，STOP LED 为黄色。表 6-8 给出了 CP 443-5Ext 的 LED 可能的状态和 CP 对应的运行状态。表中的灰色单元表示未使用该 LED。

表 6-8　CP 443-5 Ext 的 LED

INTF	EXTF	BUSF	RUN	STOP	CP 运行状态
			闪烁	亮	启动（STOP→RUN）
			亮	熄灭	RUN
			亮	闪烁	停止（RUN→STOP）
			熄灭	亮	STOP
亮			熄灭	亮	因为内部错误或存储器复位进入 STOP 状态
熄灭	熄灭	熄灭	熄灭	闪烁	等待固件升级（上电后 10s 内）
亮	亮	熄灭	熄灭	闪烁	等待固件升级（CP 当前有不完善的固件版本）
亮			亮	熄灭	运行时下载，或有内部错误的运行，例如错误的组态数据
		亮			PROFIBUS 总线错误
	亮	闪烁	亮	熄灭	运行，但是 DP 从站未传输数据或不能访问
	亮	熄灭	亮	熄灭	运行，但是 DP 从站有出故障的模块
闪烁	闪烁	闪烁	闪烁	闪烁	模块故障或系统错误

2．启动 CP 443-5Ext 的诊断功能

在 SIMATIC 管理器中创建一个名为 443_Diag 的项目（见随书光盘中的同名例程），系统的硬件结构如图 6-34 所示，CP 443-5 Ext 是 DP 主站，站地址为 3。它有 3 个从站，4 号和 5 号从站分别是 16DI 和 16DO 的 ET 200B。7 号从站是 ET 200M，它有一块 8DO 模块、一块 16DI 模块和一块 2AO 模块。

3．DP 从站的故障

在 SIMATIC 管理器生成 OB82、OB85、OB86 和 OB122。用 DP 电缆连接 CP 和各从站的 DP 接口，接通主站和从站的电源，将组态信息和逻辑块下载到 CPU。将 CPU 和 CP 的模式选择开关置于 RUN 位置。用下面的方法“人为地”制造两个从站的故障：

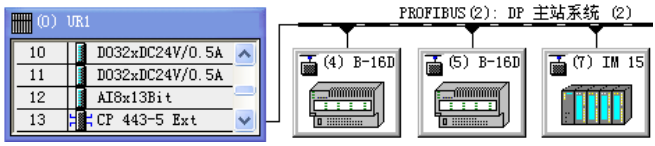


图 6-34　HW Config 中的 DP 网络

1）断开 4 号从站 ET 200B-16DO 的电源。

2）断开 AO 模块 0 号通道输出端外接的小开关，模块出现电流输出回路开路故障。

出现故障后，IM 153-1 和 AO 模块的 SF LED 亮，CPU 的 EXTF 和 DP EXTF LED 亮，CP 443-5 Ext 的 EXTF LED 闪烁，CPU 和 CP 的 RUN LED 亮。选中 SIMATIC 管理器左边窗口的 400 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图，在“CPU/故障模块”列表中可以看到 CPU 和有故障的 4 号、7 号从站。

4．用 CP 443-5 Ext 的在线视图进行诊断

选中 SIMATIC 管理器中的 400 站点，打开 HW Config，双击 CP 443-5 Ext，点击 CP 443-5 Ext 属性对话框的“诊断”选项卡中的“运行”按钮（见图 6-35），打开 CP 的诊断对话框（见图 6-36），点击工具栏上的按钮，进行在线诊断。

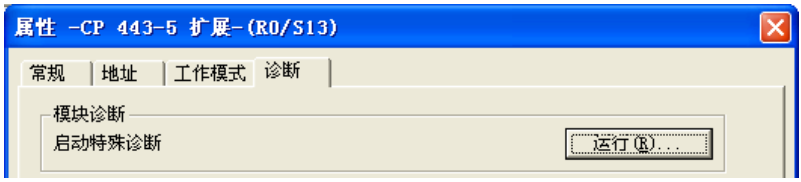


图 6-35　启动 CP 443-5 Ext 的诊断功能

选中左边窗口中的某个对象，右边窗口是该对象有关的信息。选中左边窗口的“模块”，可以看到 CP 模块的主要参数。

图 6-36 的诊断缓冲区给出了按事件顺序排列的事件消息。诊断缓冲区提供所有与 CP 通信服务有关的消息，最新的消息在最上面显示。

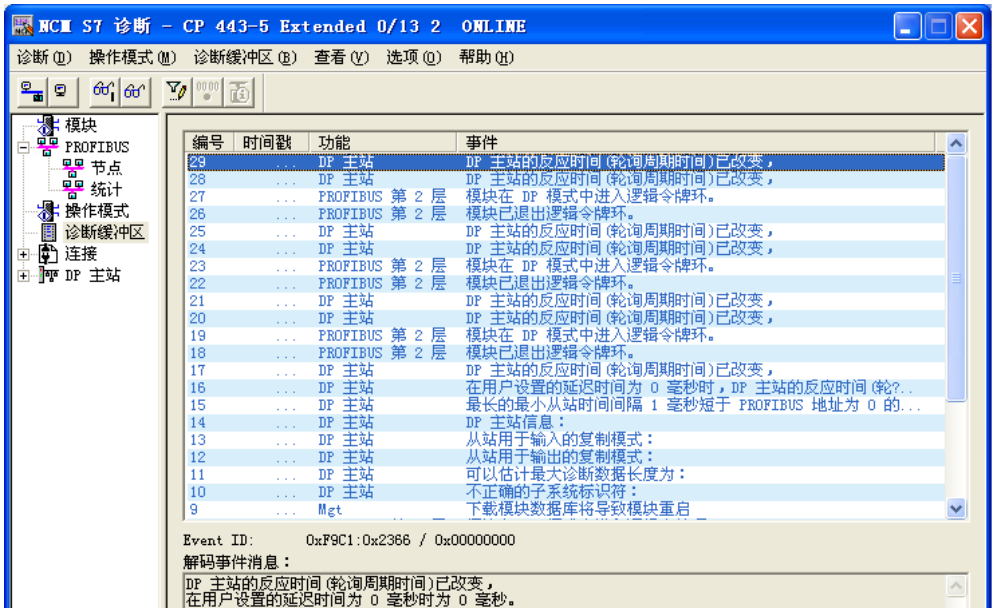


图 6-36　CP 443-5 Ext 的诊断缓冲区

选中诊断缓冲区中某个事件消息，下面的“解码事件消息”区给出该消息有关的信息。双击某个事件消息，将会显示帮助文本，进一步详细解释该消息。

选中左边窗口中的 PROFIBUS，右边窗口是网络的状态和总线参数。

选中左边窗口的“节点”，图 6-37 右边窗口的 3 号站是 DP 主站（CP 443-5 Rxt），其余的节点是 DP 从站。因为 4 号从站断电，没有被检测出来。

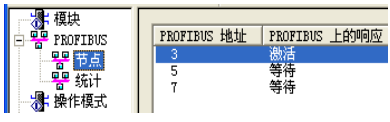


图 6-37　诊断视图中的节点信息

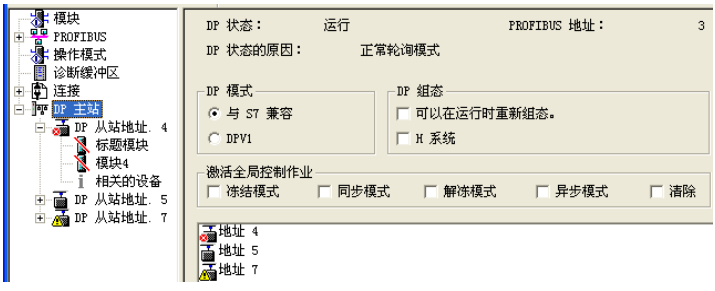


图 6-38　CP 443-5 Ext 诊断视图中 DP 主站的信息

选中图 6-38 左边窗口的“DP 主站”，DP 的状态为“运行”，DP 的模式为“与 S7 兼容”。某些老式的 CP 不能选用 DPV1 模式。4 号从站上的诊断符号（红色的指示灯）表示 DP 主站没有与该从站交换用户数据。可能的原因如下：

1）该 DP 从站的参数分配/组态错误。也可能是因为从站故障，读取不到组态信息。

2）作为 DP 主站的 CP 没有周期性地轮询该 DP 从站。

图 6-39 是选中 4 号从站的“标题模块”时，从站的诊断结果。

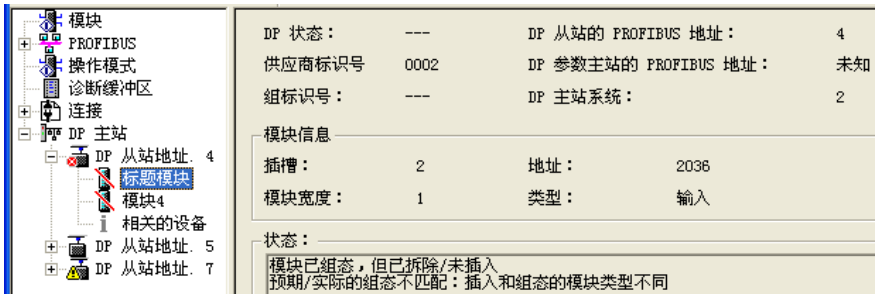


图 6-39　CP 443-5 Ext 诊断视图中 DP 从站的标题模块信息

正常运行的 4 号从站的“标题模块”中的诊断信息为“模块存在并正常”。

图 6-40 是 7 号从站的诊断信息，“从站诊断”区有 3 个打勾的诊断信息：

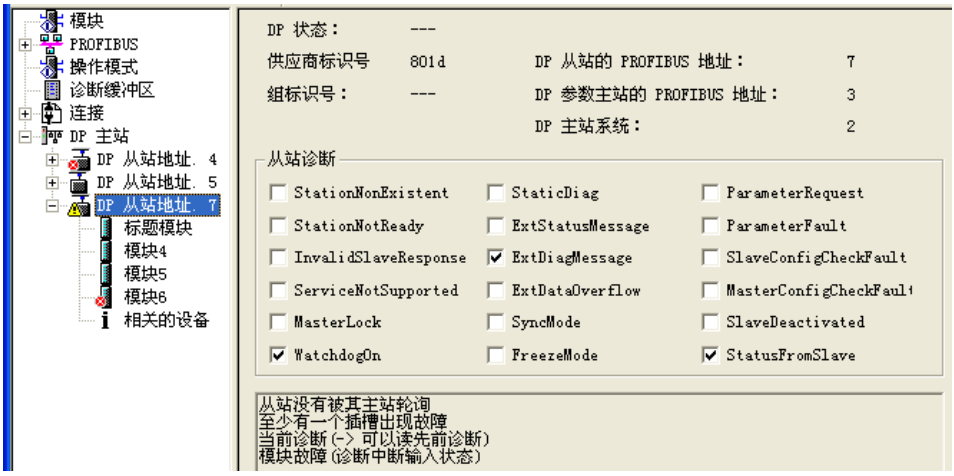


图 6-40　CP 443-5 Ext 诊断视图中 7 号从站的信息

1）ExtDiagMessage：存在与 DP 从站有关的重要的诊断数据。

2）WatchdogOn：从站的看门狗（监控定时器）处于激活状态。

3）StatusFromSlave：显示的诊断数据来自从站。

“从站诊断”区的其他英文信息的含义可用 STEP 7 的在线帮助功能查阅。

选中 7 号从站的“标题模块”，其状态为“模块存在且正常”。选中该从站有故障符号的模块 6（6 号槽的 AO 模块，见图 6-41），右边窗口是它的诊断信息。

选中左边窗口最下面的 7 号从站的“相关的设备”，右边窗口是十六进制的供应商信息（诊断信息）和供应商标识符。

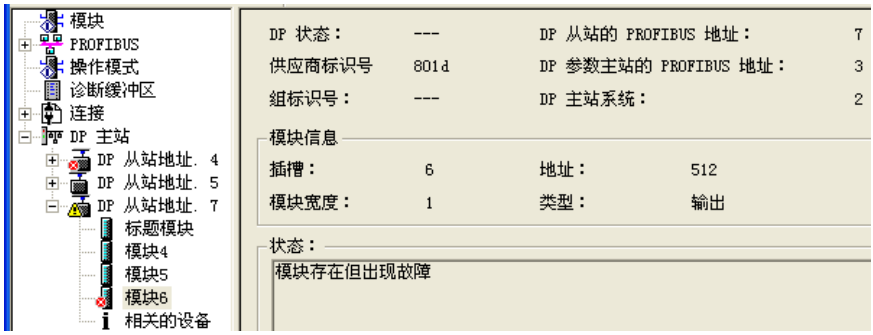


图 6-41　7 号从站的 AO 模块的诊断信息

6.5.2　PROFIBUS 通信处理器的典型故障与可能的原因

1．CP 不能切换到 RUN 模式

如果系统上电后，即使 CP 342-5 的模式开关在 RUN 位置，但是始终处于 STOP 状态，出现的现象和可能的原因如下：

1）为 CP 下载了无效的组态，STOP LED 和红色的故障 LED 持续亮。

2）STOP LED 持续亮，RUN LED 闪烁。PROFIBUS 网络中有一个具有不同的传输速率或不同的总线参数的主站，应改正传输速率。

3）STOP LED 持续亮，绿色 RUN LED 闪烁，NCM 的在线功能超时。S7 CPU 未生成总线标识，PROFIBUS CP 等待 S7 CPU 传送正确的 MPI 参数。

4）物理总线故障，例如总线短路，该站不在逻辑环中。

5）CP 的开关在 STOP 位置。

此外还应检查 CP342-5 连接的 DC 24V 电源线是否正常，M 端是否与 CPU 的 M 端连接。

2．CP 不能切换到 DP 主站模式

可能的原因如下：

1）未组态 DP 主站。

2）调用 FC 1 DP_SEND 时指定了错误的发送长度。

3）CPU 处于 STOP 模式或尚未运行 FC 1“DP_SEND”和 FC 2“DP_RECV”。

4）调用了 FC 4“DP_CTRL”，请求 DP 模式为 STOP 或 OFFLINE（离线，非轮询模式）。应调用 FC 4，请求 RUN DP 模式。DP 模式的定义见 3.5.4 节。

5）从站处于 STOP 模式，例如从站的开关在 STOP 位置。

6）PROFIBUS CP 作为 2 类主站，周期性地读取从站的输入/输出数据。应关闭读取服务，打开正常数据传输，即切换到 1 类主站模式。

3．DP 从站的输出总是为零

用户程序要求输出数据不等于零，但是 DP 从站的输出为零。可能的原因：

1）组态了对错误的反应为 AUTOCLEAR（自动清除），且至少一个已组态的 DP 从站没有处于数据传输状态。应改正组态，检查 RUN 开关的位置和与总线的物理连接。

2）调用了 FC 4，请求 DP 模式为 CLEAR。应调用 FC 4，请求 RUN DP 模式。

4．DP 主站的其他问题

1）输入数据没有到达 CPU 中的所需区域，输出了错误的输出数据。可能为 DP_SEND 或 DP_RECV 指定了错误的 ANY 指针区域。

2）虽然触发了周期性全局控制作业 SYNC（同步）和 FREEZE（冻结），但是只处理最后一个作业。可能发送了两个独立的全局控制作业，应使用一个全局控制作业，发送全局控制命令 SYNC 和 FREEZE。

5．DP 从站模式的典型问题

从站 CP 没有接收到来自 DP 主站的数据，或 DP 主站没有接收到来自从站 CP 的数据。下面是可能的原因：

1）DP 主站尚未处于数据传输状态。

2）没有将 PROFIBUS CP 组态为 DP 从站模式。

3）作为 DP 从站的 PROFIBUS CP 没有运行 FC 2“DP_RECV”或 FC 1“DP_SEND”。

4）调用 DP_SEND 或 DP_RECV 时，在从站中指定的 I/O 长度与主站组态的 I/O 长度不匹配。

5）DP 主站处于 CLEAR 状态，或 DP 主站组态了对错误的反应为 AUTOCLEAR，且在 DP 主站上组态的至少一个 DP 从站不在数据传输状态。应将 DP 主站切换到 RUN 模式，消除 CLEAR 模式。

6）因为超过了监控定时器（看门狗）的定时时间，DP 主站不再轮询作为 DP 从站的 PROFIBUS CP。

7）DP 主站已经为其他主站释放了 PROFIBUS CP，例如改变为 OFFLINE 模式。

应使 DP 主站返回 RUN 模式，或用另一个主站启动数据传输。

8）CPU 处于 STOP 模式，未运行 DP_SEND 和 DP_RECV。

9）作为 DP 从站的 PROFIBUS CP 的模式开关位于 STOP 位置。

6．FDL 连接的典型故障

（1）FDL 连接没有数据传输，或只在一个方向上有数据传输，可能的原因如下：

1）没有在用户程序中调用 FC 5“AG_SEND”和 FC 6“AG_RECV”，或接收、发送缓冲区太小或设置错误。如有必要，改正定义数据区的 ANY 指针。

2）LSAP（连接服务访问点）分配错误，可根据诊断缓冲区的条目改变 LSAP。

3）不能到达 PROFIBUS 目标地址，应输入正确的目标地址。

4）FC 5“AG_SEND”作业报头有错误。

（2）数据传输太慢

可延迟发送触发器，检查目标站，并优化接收。

（3）没有在 FDL 连接上发送完整的数据区，可能的原因：

1）FC 5“AG_SEND”的参数 LEN 被设置成错误的数值。对于带作业报文头的作业，参数 LEN 必须包括作业报文头和用户数据的字节数。

2）用 ANY 指针指定的发送缓冲区太小。

6.5.3 使用计算机的通信处理器进行诊断

CP 5611、CP 5621、CP 5613 和 CP 5614 是用于台式计算机的 PCI 总线通信卡，CP 5511

和 CP 5512 是用于笔记本电脑的 PCMCIA 卡。可以用它们来将计算机连接到 MPI 或 PROFIBUS 网络，通过网络实现计算机与 PLC 的通信。

也可以使用计算机的工业以太网通信卡 CP 1613、CP 1616，通过工业以太网实现计算机与 PLC 的通信。

与 PC/MPI 适配器、USB/MPI 适配器相比，用于计算机的 CP 卡的通信速率高，还有很强的网络故障诊断功能。本节以 CP 5613 为例，介绍用计算机的通信处理器诊断网络的方法。

1．用 CP5613 诊断 MPI 网络

在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PG/PC 接口”，打开“设置 PG/PC 接口”对话框。可以看出，已经激活了 MPI 协议（见图 6-42）。用 PROFIBUS 电缆连接 CP 5613 和两台 PLC 的 MPI 接口。点击“诊断”按钮，打开诊断对话框（见图 6-42 的右图）。点击“Test”按钮，如果网络正常，在按钮右边出现“OK”，在“Bus parameters”（总线参数）区将会出现检测的结果。如果 CP 5613 不能正常使用，则将会出现错误信息。

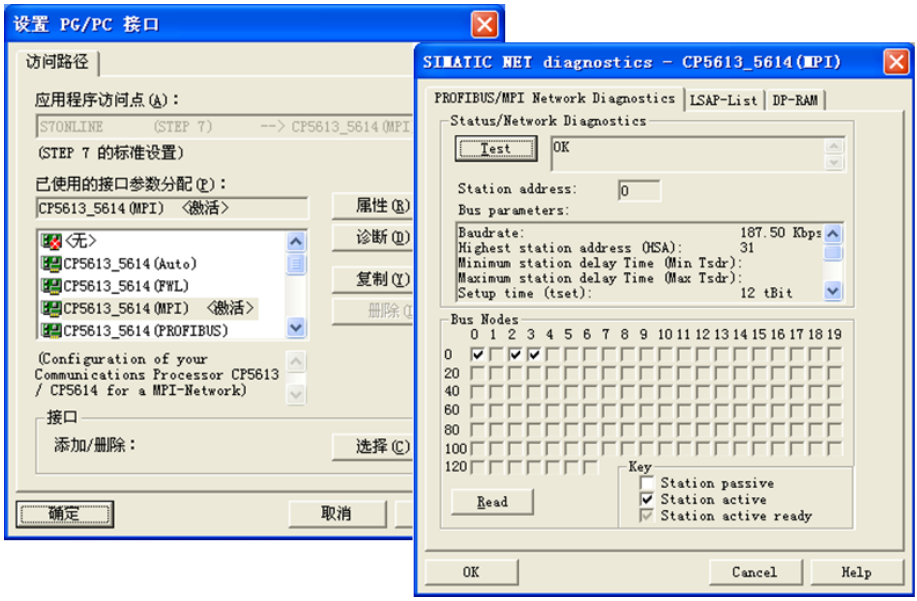


图 6-42　用 CP 5613 诊断 MPI 网络

点击“Read”按钮，在“Bus Nodes”（总线节点）区，显示 MPI 网络上检测到的节点（用打勾表示），0 号节点是计算机，2 号和 3 号节点是两块 CPU。用这个方法可以准确地判断出网络上站点的连接情况。

2．切换 CP 5613 的通信协议

MPI 是 CPU 最基本的通信网络，首先应使用 MPI 通信接口，将 PLC 的 PROFIBUS 或以太网通信接口和通信网络的组态信息下载到 CPU，为 STEP 7 使用 PROFIBUS 或以太网来下载和监控 PLC 做好准备工作。

项目 OB_Diag1（见随书光盘中的同名例程）中的 CPU 413-2DP 是 DP 主站，它有 3 个 ET 200 从站。首先用 MPI 网络将组态信息和逻辑块下载到 CPU。

选中图 6-42 左图的列表框中的“CP5613_5614（PROFIBUS）”后，点击“确定”按钮，

出现警告对话框（见图 6-43）。

点击“确定”按钮，对话框消失。再次打开“设置 PG/PC 接口”对话框，可以看到 PROFIBUS 已被激活。可以用同样的方法，切换 PC/MPI 适配器和 USB/MPI 适配器使用的通信协议。

图 6-43　警告对话框

3．用 CP 5613 诊断 PROFIBUS-DP 网络

将 CP 5613 切换到 PROFIBUS 模式后，用电缆连接 CP 5613、作为主站的 PLC（3 号站）集成的 DP 接口，和 3 个 DP 从站（4、5、7 号从站）的 DP 接口。

点击图 6-42 中的“诊断”按钮，打开诊断对话框（见图 6-44 中的左图）。点击“Test”按钮，按钮右边出现“OK”，“Bus parameters”（总线参数）区出现检测的结果。点击“Read”按钮，“Bus Nodes”（总线节点）区显示 PROFIBUS 网络上检测到的节点，0 号节点是计算机，3 号节点是作为主站的 CPU。总线节点表中打勾的站是被激活的站，没有打勾，背景为白色的 4、5、7 号从站是被动的站（即从站）。

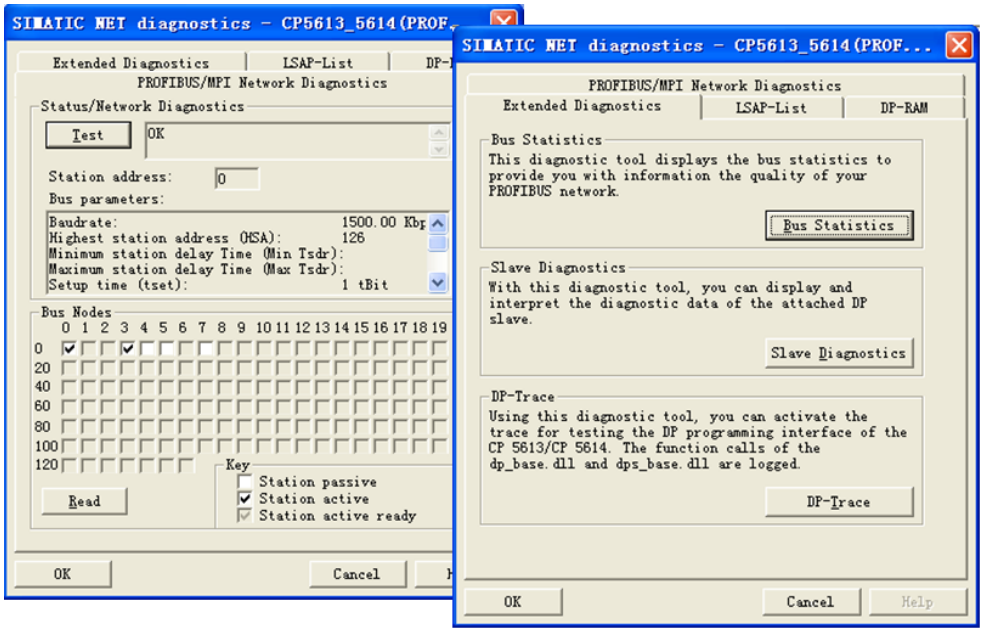


图 6-44　用 CP 5613 诊断 PROFIBUS 网络

用上述的方法，可以迅速地查明网络上各站点的连接情况。

在诊断对话框的“Extended Diagnostics”（扩展的诊断）选项卡中，可以查看总线的统计数据（Bus Statistics）。点击“Slave Diagnostics”，可以对 DP 从站进行诊断。点击“DP-Trace”按钮，可以激活 CP 5613 的编程接口的追溯功能。

6.6 使用专用硬件进行测试与诊断

6.6.1 诊断中继器

1．诊断中继器简介

诊断中继器是有诊断功能的 RS-485 中继器，用于在系统正常工作时进行线路诊断。它以 DP 从站模式运行，作为一个 RS-485 中继器集成在 PROFIBUS-DP 网络中，传输速率为 9.6 kbit/s～12 Mbit/s。它有铜质总线电缆的物理在线监控功能，在运行中发生故障时，自动检测故障类型和故障位置，发送诊断报文到 DP 主站，可以提高对故障模块或 DP 电缆的中断位置的定位能力，减少系统的停机时间。可以用 HMI（人机界面）显示错误位置及原因。

诊断中继器可以诊断通信线断线、短路、没有诊断电阻、网段中节点过多等故障。

诊断中继器的外形和面板见图 6-45 和图 6-46，它可以安装在 S7-300 组合导轨或 DIN 导轨上，使用外部 DC 24 V 电源，用 DIP 开关设置站地址。面板上的元件的作用见表 6-9。

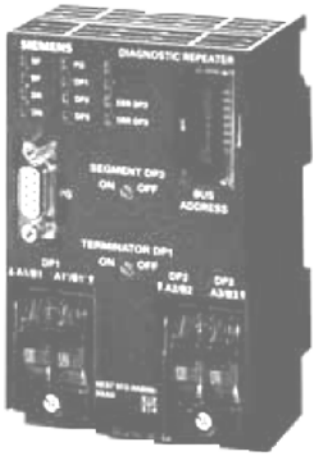


图 6-45　诊断中继器

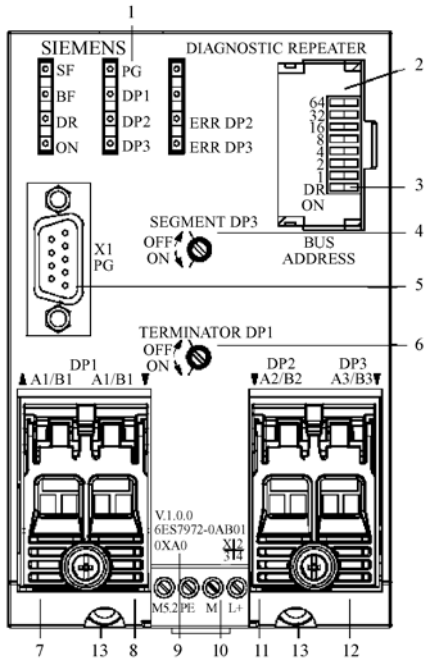


图 6-46　诊断中继器的面板

表 6-9　诊断中继器面板上的元件

序　号	功　能	序　号	功　能
1	指示故障信息的 LED 指示灯	8	网段 DP1 的电缆输出接口（A1'/B1'）
2	设置中继器站地址的 DIP 开关	9	中继器产品订货号和版本号
3	DR 开关用于激活中继器的诊断功能	10	电源接口
4	网段 DP3 的转换开关	11	网段 DP2 的电缆输入接口（A2/B2）
5	集成有终端电阻的 PG（编程器）接口	12	网段 DP3 的电缆输出接口（A3/B3）
6	网段 DP1 的终端电阻开关	13	机架固定螺母
7	网段 DP1 的电缆输入接口（A1/B1）		

9 针 D 形连接器用于连接编程计算机，诊断中继器作为 DP 标准从站集成在总线系统中，可连接 3 个 DP 总线段，对其中的两个网段进行监控。每个网段最大长度 100m，最多 31 个站，可以串联连接最多 9 个诊断中继器。诊断中继器可以安装在 S7-300 的导轨上，也可以安装在标准导轨上。

2．诊断中继器的 LED 功能

表 6-10　诊断中继器 LED 的功能

名　称	颜　色	功　能
SF	红	组错误，内部故障
BF	红	总线故障；闪烁表示站点组态错误，站地址设置错误、组态与实际不符合
DR	绿	中继器功能正常；闪烁表示正在搜索波特率
ON	绿	电源接通，电压正常
PG	黄	编程设备接口总线为激活状态，闪烁表示编程设备接口总线未激活
DP1	黄	DP1 网段总线被激活，闪烁表示诊断中继器自动断开 DP1 网段，因为检测不到正确的报文帧
DP2	黄	DP2 网段总线被激活，闪烁表示诊断中继器自动断开 DP2 网段，因为检测不到正确的报文帧
DP3	黄	DP3 网段总线被激活，闪烁表示诊断中继器自动断开 DP3 网段，因为检测不到正确的报文帧
ERR DP2	红	DP2 网段线路故障，闪烁表示诊断中继器正在激活网段 DP2 的线路
ERR DP3	红	DP3 网段线路故障，闪烁表示诊断中继器正在激活网段 DP3 的线路

6.6.2　硬件组态与诊断的准备工作

1．组态主站和 PROFIBUS 网络

在 STEP 7 中用新建项目向导创建一个项目（见随书光盘中的例程 Repeater），CPU 为 CPU 315-2DP。在 SIMATIC 管理器选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具（见图 6-47），将电源模块和信号模块插入机架。

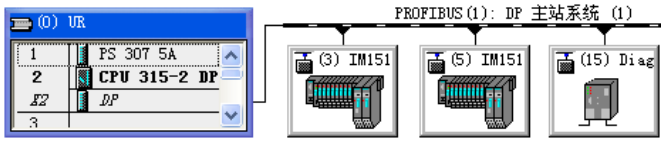


图 6-47　HW Config 中的 DP 网络

双击 CPU 模块中“DP”所在的行，点击打开的对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中点击“新建”按钮，生成一条 PROFIBUS-DP 网络，点击“确定”按钮返回 HW Config。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。

2．组态从站和诊断中继器

将 IM 151-1 拖放到 DP 网络，生成两个 ET 200S 从站，站地址分别为 3 和 5。将电源模块和 I/O 模块插入插槽。

将诊断中继器作为一个标准的 DP 从站，组态到 DP 网络。诊断中继器在 HW Cofig 的硬件目录的“\PROFIBUS-DP\Network Components\Diagnostic Repeater”文件夹中，订货号为 6ES7 972-0AB01-0XA0，组态的方法与其他标准从站相同。组态好后点击工具栏上的按钮，编译并保存组态信息。

3．硬件接线与设置

本例程只使用网段 DP2 进行诊断，没有使用网段 DP3。网段 DP3 的选择开关在 OFF 位置。DP 主站接在网段 DP1 的进线接口（A1/B1）上，DP1 的出线接口（A1'/B1'）没有连接电缆，网段 DP1 被中断。网段 DP1 的选择开关在 ON 位置，即在网段 DP1 上连接终端电阻，DP1 右边的连接器无效。

用面板右上角的 DIP 开关将诊断中继器的 DP 站地址设置为 15。开关的左边按下为 ON，站地址是所有设置为 ON 的开关左侧的数字之和。将 DIP 开关最下面一位 DR 设置为 ON，启动诊断功能。用 CPU 的 MPI 接口下载程序和硬件组态。然后在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PC/PG 接口”，将 CP 5611 使用的通信协议由 MPI 改为 PROFIBUS。用 DP 电缆连接 CP 5611 的接口和中继器的 PG 接口，DP 电缆中继器一端的 DP 连接器的终端电阻应设为 OFF。将主站 CPU 集成的 DP 接口连接到网段 DP1 的进线连接器(A1/B1）上，两个 ET 200S 从站连接到网段 DP2 的连接器上（见图 6-49 中的网络拓扑）。

4．线路诊断的准备工作

要在运行时查找到出现异常的位置，诊断中继器首先必须检测连接的 PROFIBUS 子网的拓扑结构和有关的参数。

测量网络拓扑的条件如下：诊断中继器已经组态、安装和连接；诊断中继器上 DIP 开关的 DR 位被设置为 ON（出厂设置）；已设置各站的 PROFIBUS 地址；DP 主站的电源开关已接通；计算机与诊断中继器的 DP 接口已经连接好。测量网络拓扑的操作步骤如下：

1）首先选中管理器左边窗口的项目 Repeater，然后选中右边窗口的 PROFIBUS（1）。

2）执行菜单命令“PLC”→“PROFIBUS”→“准备线路诊断”。

3）在打开的准备线路诊断对话框中（见图 6-48），显示出 DP 网络上搜索到的各个站点的地址、标识符和所属的主站系统。点击“重启动”按钮，开始测量线路。

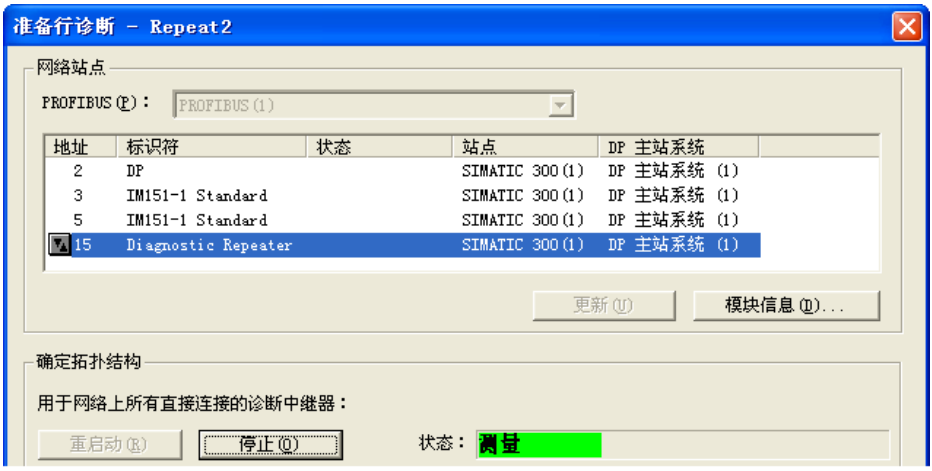


图 6-48　准备线路诊断对话框

诊断中继器搜索各个从站的 PROFIBUS 地址；使用反射测量方式，检测到各个从站的距离，将它们存储在拓扑表中。诊断中继器还会记忆检测到的通信伙伴所在的网段。运行时用拓扑表的条目来确定哪些伙伴之间有需要处理的段故障。

在线路检测过程中，对话框下面的“状态：”右边动态变化的绿色的进度条显示检测的

过程。测量完成后，“状态：”右边显示“测量完成”。

为了得到网络的拓扑结构，计算机连接到诊断中继器的 PG 接口的 CP 5611 的通信接口必须设置为 PROFIBUS，不能设置为 MPI。计算机与诊断中继器的通信不能使用 PC 适配器。

在总线正常工作时，如果出现故障，诊断中继器检查电缆，确定到故障点的距离和产生故障的原因，并发出诊断报警。

5．显示网络拓扑

成功地完成了上述操作后，执行菜单命令“PLC”→“PROFIBUS”→“显示网络拓扑”，打开拓扑显示视图（见图 6-49）。

网络拓扑图不但可以显示 PROFIBUS-DP 网络的拓扑结构，还可以用从站上的符号来显示有故障的站点的状态，以及各个站点所在的网段名称，站地址和线路长度（见图 6-49）。距离测量的误差为±1 m。

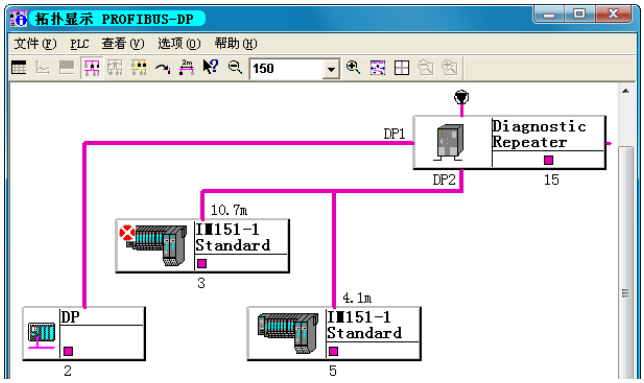


图 6-49　网络拓扑与电缆长度显示

如果有不能进行分配的节点，将在对话框的上半部分显示它们。每次修改硬件结构之后，必须重新执行“准备线路诊断”命令，为显示网络拓扑和诊断做好准备。执行菜单命令“查看”→“拓扑表”，可以用表格而不是图形方式来显示网络拓扑。

系统正常运行时，选中拓扑显示图中的诊断中继器，执行菜单命令“PLC”→“模块状态”，打开中继器的模块信息对话框。“DP 从站诊断”选项卡见图 6-50，“常规”选项卡给出的模块状态为“模块可用且正常”。在选项卡“DP1”、“DP2”和“PG”中，可以看到对应的网段完好的信息。图 6-51 是“DP3”选项卡的信息。

图 6-50　诊断中继器的模块信息

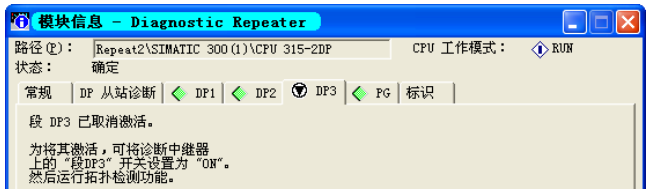


图 6-51　诊断中继器的模块信息

6.6.3　用拓扑显示视图诊断网络故障

1．用拓扑显示视图诊断有故障的站

正常运行时中继器上的 DP1、DP2、DR、ON LED 亮。

在运行时拔出 3 号 DP 从站 6 号槽有诊断功能的 DI 模块，CPU 和 IM 151-1 的 SF LED 亮，BF LED 闪烁，其余的 LED 状态不变。

在拓扑显示图中，可以看到 3 号从站上的红色故障符号（见图 6-49）。双击 3 号从站，打开 IM 151-1 的模块信息对话框（见图 6-52），可以看到 6 号槽模块丢失的故障信息（kein 是德语单词，表示“没有”）。点击“十六进制格式”按钮，显示出十六进制的诊断信息。

图 6-52　IM 151-1 的模块信息

2．断开 3 号从站 DP 连接器上的终端电阻

如果线路中断或有故障，没有终端电阻或终端电阻过多，将会出现反射错误。在运行期间，断开 3 号从站 DP 连接器上的终端电阻，CPU 的 SF LED、中继器的 SF LED 和 ERR DP2 LED 亮，其余的 LED 正常。在拓扑显示图中，诊断中继器上出现红色的故障符号。

双击诊断中继器，打开它的模块信息对话框（见图 6-53）。对话框各选项卡的标签上的符号、和分别表示没有网段错误、有网段错误和网段被关闭。“DP 从站诊断”选项卡的诊断信息为“反射错误率为 100%，3 号从站 A/B 线没有电阻”。点击“十六进制格式”按钮，显示出十六进制格式的诊断信息。

图 6-53　诊断中继器的模块信息

点击有故障符号的 DP2 的标签，打开“DP2”选项卡（见图 6-54）。可以看到网段 DP2 出错的故障点位于 3 号从站，以及错误信息和解决方法。点击“详细资料”按钮，可以获得详细的信息（见图 6-55）。

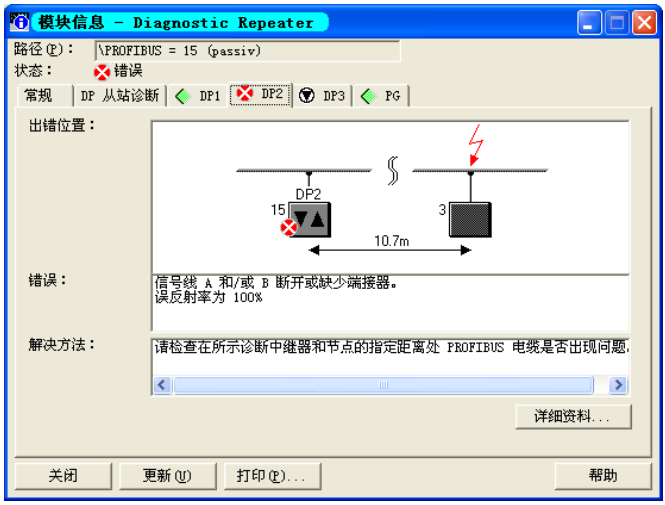


图 6-54　诊断中继器的模块信息

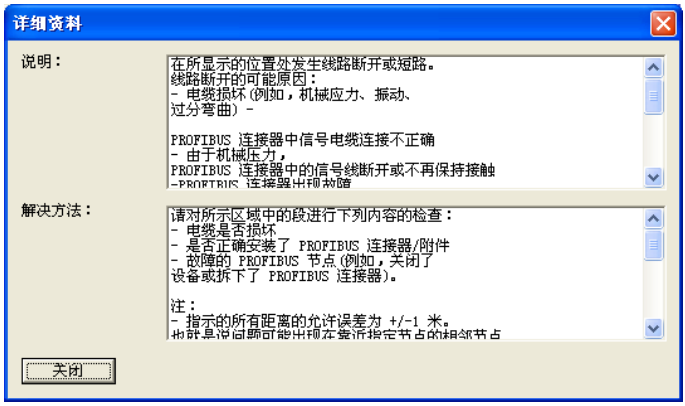


图 6-55　DP2 网段的详细信息

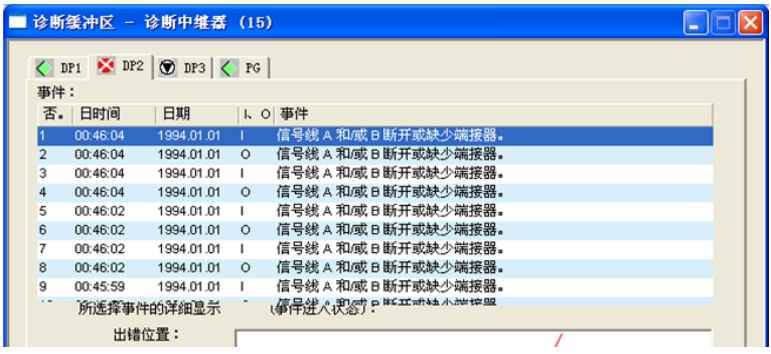


图 6-56　诊断中继器网段 DP2 的诊断缓冲区

3．诊断中继器的诊断缓冲区

诊断中继器为网段 DP1、DP2、DP3 和 PG 接口分别分配一个诊断缓冲区，用来记录和显示 PROFIBUS 的历史错误事件，各缓冲区保存最后的 10 个事件。缓冲区包含了事件的时间、日期和简短的描述。

打开拓扑显示对话框，选中诊断中继器。执行菜单命令“PLC”→“诊断缓冲区”，打开诊断中继器的诊断缓冲区对话框（见图 6-56），显示诊断中继器检测到的错误事件的列表，事件按发生时间排序。选中某一事件，对话框的下半部分显示的信息和图形与图 6-54 的基本上相同。也可以点击“详细资料”按钮来查看详细的信息。

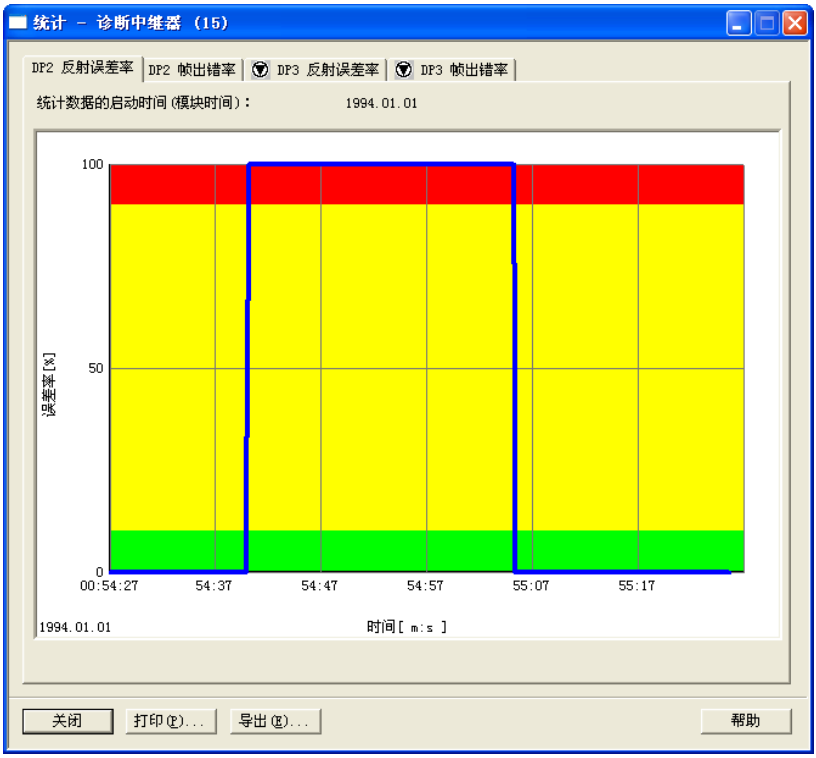


图 6-57　诊断中继器网段 DP2 反射误差率的统计图

4．诊断缓冲区的统计功能

诊断中继器为网段 DP2 和 DP3 各分配了一个统计缓冲区，其中包含反射错误率和消息帧错误率的信息。

1）反射错误：如果线路中断或有故障，没有终端电阻或终端电阻过多，将会出现反射错误。

2）消息帧错误：有缺陷的节点可能导致奇偶校验错误。该值持续显示的时间为 60s，从对话框被打开的时刻开始计时。在显示期间，缓冲区内可以继续保存更多的数值。

选中拓扑显示视图中的诊断中继器，执行菜单命令“PLC”→“统计”，打开诊断中继器的统计对话框（见图 6-57）。系统正常运行时，反射误差率为 0%，终端电阻开路时为 100%。横坐标为时间轴。

6.6.4 BT 200 总线测试仪的应用

1．BT 200 总线测试仪简介

BT-200 总线测试仪用于测试 RS-485 接口、PROFIUBS 总线电缆和可以访问的 DP 从站。BT 200 的外观和按钮功能见图 6-58。

测试仪的结构紧凑，容易操作，可供安装人员、调试工程师和维修人员使用。

在安装阶段，可以用 BT 200 测试 PROFIBUS 电缆，检查站的接线，能迅速发现安装错误。运行时 BT 200 可用于查找网络的故障，以减少设备停机的时间。

在系统调试之前，就可以测试 PROFIBUS-DP 站的 RS-485 接口。即使 PROFIBUS-DP 上没有主站，也能列出连接到总线上的所有从站。每次测试的结果都可以保存在 BT 200 中，可以将它传送到 PC。计算机安装了用于生成测试结果报告的 BT 200 软件，可以打印出测试结果。

图 6-58　BT 200

2．普通模式的线路测试

BT 200 的运行分为普通模式和专家模式。普通模式只能测试网络的接线状态。

总线段的线路测试在 BT 200 和测试连接器之间进行（见图 6-59）。在设备安装阶段，测试连接器安装在总线段的一端，对总线段一段一段地依次进行测试。总线段的两端需要配置终端电阻，中间不能有终端电阻。

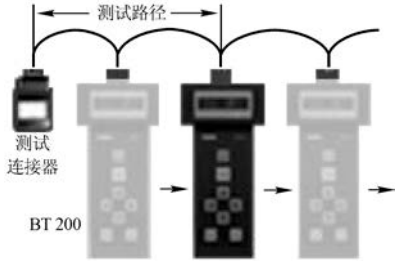


图 6-59　用 BT 200 测试线路

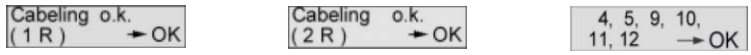
按下 BT 200 的电源按钮，直到出现图 6-60a 所示的待命画面，大约 2s 后出现图 6-60b 所示的电池容量画面，再过 2s 后进入普通模式，显示图 6-60c 所示的线路检测启动画面。

a）　　b）　　c）

图 6-60　BT 200 的显示

如果 3min 内没有按键和进行测试，将进入节能模式，BT 200 自动关闭。

测试时总线上可以连接或不连接站点，按下 TEST（测试）键，开始测试。图 6-61a 和图 6-61b 显示的信息表示测试成功，电缆正常，1R 和 2R 分别表示有 1 个或两个终端电阻。如果按 OK 键或 ESC 键，测量结束，返回图 6-60c，又可以开始新的测量。



a）　　b）　　c）

图 6-61　BT 200 的显示

下面是线路测试中可能显示的一些信息：

1）“Turn off all stations”：请检查所有站点和电源设备的电源是否关闭。

2）“Change A-B”：需要交换 RS-485 的 A、B 芯线。

3）“Fix short circuit A-B”：A、B 线之间短路。

4）“Fix short circ. A-Shield”、“Fix short circ. B-Shield”：A 线或 B 线与屏蔽层之间短路。

5）“Fix all wire”：多根芯线或芯线和屏蔽层断开、测试连接器未连接好。

6）“Fix broken wire A”、“Fix broken wire B”、“Fix broken wire shield”：A 线、B 线或屏蔽层开路。为了检测后者，屏蔽层不能接地。

7）“switch on termination …>”：没有终端电阻或终端电阻不止两个，按▶键显示“<...R on both ends –> TEST”。按◀键返回“switch on termination …>”。断开线路中间多余的终端电阻，或接入线路两端缺少的终端电阻后，按〈TEST〉键重新测试。

3．专家模式的测试

同时按 ESC 键和 OK 按钮，从普通模式切换到专家模式。下面是专家模式的菜单：

1）Cabeling：普通模式的线路测试。

2）Station test：站点测试，检查从站或主站的 RS-485 接口。

3）Bus-scan：支路（Branch）测试。

4）Cablelength：电缆长度测量。

5）Reflections：信号反射测试。

6）Service：服务菜单。

用▲、▼键上下移动菜单项。屏幕第一行最左边有一个闪动的矩形光标，按 OK 键执行第一行的命令。可以用 ESC 键来取消当前的功能，或返回更高一级的菜单。

（1）测试主站的 RS-485

站点测试用于测试单个主站或从站的 RS-485 接口。可以测试出 RS-485 接口正常、没有连续接收到信号、没有接收到任何信号、总线的 5V 电平的大小、是否有 RTS 信号。

1）首先建立站点的 RS-485 接口和 BT 200 之间的点对点连接。接通站点的电源，主站的 CPU 应处于 RUN 模式。用测试电缆将 BT 200 连接到站点的 RS-485 接口上。

2）同时按 OK 和 ESC 键，进入专家模式。按▼键，将菜单项“Station test”移到第一行。

3）按 OK 键，然后按 TEST 键，显示“Searching slaves…”（搜索从站）。搜索到主站时显示“Master-address xxx –>OK”。“xxx”是主站的地址。

4）按 OK 键显示网络的波特率，例如“Baudrate 1.5M Baud –>OK”。

5）按 OK 键显示“Please wait…”，等待若干秒后，显示“RS485 o. k. 5V：4, 96 V …>”。冒号后面是检测到的实际信号电平值，小数点用逗号表示。如果实际信号电平与 5V 相差太远，通信可能不可靠。

6）按▶键显示是否有 RTS 信号，“<… RTS Yes”表示有 RTS 信号。

（2）测试从站的 RS-485

第 1、2 步与测试主站的 RS-485 相同。第 3 步搜索到从站时显示“Slave-address xxx –>OK”，xxx 是从站地址。按 OK 键显示 5V 电压实际值。按►键显示是否有 RTS 信号，再按 OK 键返回专家模式菜单。可能的检测结果：RS485 defective（没有接收到连续的信号）、“No response”（没有接收到任何东西）。

（3）支路测试

可以跨越中继器或光纤，对整个网络上的某个站点或所有的站点进行测试。

1）测试时断开总线上所有的主站，包括 PG、OP、CP，连接所有的从站，测试时各从站通电，将 BT 200 连接到总线上。

2）同时按 OK 和 ESC 键，进入专家模式。按▼键，将菜单项“Bus-scan”移到第一行。

3）按 OK 键，然后按 TEST 键，显示“Slave-address 000 –>OK”。地址 000 表示测试所有的从站。

4）按 OK 键后显示“Please wait…”，等待若干秒后，显示检测到的总线上的从站的地址列表（见图 6-61c）。如果两个站的地址重叠，检测不到它们。

5）按 OK 键返回专家模式菜单。

如果要检测 4 号从站的状态，在第 3 步用►、◄键键将光标移动到从站地址右边的“000”的个位，用▲、▼键将个位设置为 4。如果 4 号从站工作正常，按 OK 键后显示“Slave 4 o.k.”。如果没有找到 4 号从站，显示“No response”（没有响应）。

（4）距离测量

距离测量只能检测大于 15m 没有中继器的线路。

1）断开所有总线站点的电源，或者断开连接在线路上的站点。测试连接器和 BT 200 分别连接到线路的两端，断开 BT 200 的终端电阻。

2）同时按 OK 和 ESC 键，进入专家模式。按▼键，将菜单项“Cablelength”移到第一行。

3）按 OK 键，然后按 TEST 键，显示“All stations power off? –> OK”（是否断开了所有站点的电源？）。

4）按 OK 键，显示“Testplug mounted? –> OK”（是否安装了测试连接器？）。

5）按 OK 键，显示“Nearby terminator removed? –> OK”（是否断开了附近的终端电阻？）。

6）按 OK 键显示“Resistance 100Ω/km –> OK”。此时可以用光标键修改线路每千米的电阻值，默认值为 110Ω/km。

7）按 OK 键，显示“12MBd Plugs 00 pcs –> OK”。此时可以输入带纵向电感的 12Mbit/s 的连接器或设备的个数。

8）按 OK 键，显示“R per plug? 1.10□ –> OK”。此时可以输入每个连接器或设备的电阻值，默认值为 0.32 Ω。

9）按 OK 键，显示测量结果“Cablelength x m –> OK”，其中的 x 为测量值。如果显示 0m，表示没有获得合理的测量长度。线路长度小于 15m 是可能的原因之一。

在操作过程中可能显示出故障信息，例如“Fix broken wire shield”、“Fix all wire”和“No terminator connected”。按 ESC 键返回专家模式菜单。可能的原因：没有终端电阻，或终端电阻不止一个，被测试网段有分支电路等。

（5）反射测试 reflection

下列原因可能引起反射：有分支电路，没有终端电阻或终端电阻过多，线路使用了不正确的电缆型号，未正确安装电缆。反射测试可以用于故障（例如短路）定位，和对距离测量进行确认。

1）断开线路上的主站，并保证总线上没有电源，总线上没有通信信号。BT 200 连接到电缆的一端，另一端不要连接测试连接器。

2）同时按 OK 和 ESC 键，进入专家模式。按▼键，将菜单项“Reflections”移到第一行。

3）按 OK 键，然后按 TEST 键，如果电缆的另一端有终端电阻，没有检测到反射，显示“No reflections”。如果电缆的另一端没有终端电阻，检测到反射，将显示以 m 为单位的测量点到故障点的距离。如果距离测量失败，显示“Reflections in 65535 m”。

如果该距离与以前的距离测量一致，则距离测量被确认，证明所测量的总线网段的布线中不存在故障。按 OK 键返回菜单。

6.7 练习题

1. 使用可访问节点和在线功能进行诊断分别需要什么条件？
2. 怎样打开快速视图和诊断视图？
3. 怎样打开 CPU、从站和模块的模块信息对话框？
4. 怎样获取程序块的在线帮助信息？
5. 为了防止出现故障时 CPU 进入 STOP 模式，S7-300 和 S7-400 分别需要生成和下载哪些组织块？它们各有什么作用？
6. 怎样启动 CP 342-5 的诊断功能？
7. 试分析图 6-62 中 OB82 的局部变量的意义。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55407FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#05230000

图 6-62　OB 82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05407FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#07FE0103

图 6-63　OB 86 的局部变量

8. 试分析图 6-63 中 OB86 的局部变量的意义。

第 7 章　PROFIBUS 通信故障诊断的编程与实验

本章将介绍用下列诊断用的程序块来诊断 PROFIBUS 网络故障的方法：

1）用 SFC 13“DPNRM_DG”读取 DP 从站的标准诊断数据。

2）用 FB 125/FC 125 对 DP 从站进行诊断。

4）用 SFC 51“RDSYSST”读取系统状态表（SSL）。

5）用 FC 3 诊断 CP 342-5 的 DP 从站。

7.1　使用 SFC 13 诊断 ET 200M 和 ET 200B

7.1.1　SFC 13 简介

1．项目的硬件结构

在 SIMATIC 管理器中创建一个名为 SFC_13 的项目（本章的项目在随书光盘的文件夹“\Project\Diag”中），其硬件结构与组态的方法与 5.2.3 节中的项目 SFC_12 的相同（见图 5-35），其 DP 主站为 CPU 313C-2DP，4 号从站为 ET 200B-16DO，5 号从站为 ET 200B-16DI，7 号从站为 ET 200M，它有一块 8DO 模块、一块 16DI 模块和一块 2AO 模块。2AO 模块的 0 号通道被组态为 4～20mA 的电流输出，1 号通道被组态为 0～10V 的电压输出，有自诊断功能（见图 6-9），0 号通信的输出电路断开时，将会产生诊断中断。

2．诊断数据的长度

通过调用系统功能 SFC 13“DPNRM_DG”，可以查看遵循 EN 50 170 标准的 DP 从站的诊断信息。SFC 13 可以读取的最大报文长度为 240B。

SFC 13 的参数 RECORD 用来定义存放诊断数据的目标地址区，只能使用 BYTE 数据类型，其字节数（6～240）与从站的型号有关。

由随书光盘中的文件《ET 200B Manual_e》可知，ET 200B 数字量模块的诊断数据有 13B（13 个字节）。由随书光盘中的文件《ET 200M 操作说明》可知，ET 200M 的诊断数据与 IM 153 的型号有关，实验所用的 IM 153-1 的订货号为 6ES7 153-1AA02-0XB0，其诊断数据的长度为 29B。由 IM 153-1 的模块信息对话框的“十六进制格式诊断”信息可知（见图 7-5），实验所用的 IM 153-1 的诊断数据的长度为 29B。

如果为 RECORD 指定的存放诊断数据的地址区长度，小于从站手册提供的诊断数据字节数，不能读取完整的诊断数据，甚至可能导致读不出诊断数据，此时 RET_VAL 提供相应的错误代码。反之，目标区域将接收诊断数据，RET_VAL 中是接收的实际字节数。

3．从站的诊断地址

OB86_RACKS_FLTD 的高位字（LW8）是从站的诊断地址，用它作为 SFC 13 的输入参数 LADDR 的实参。根据它来判断是哪一个从站产生的中断。

4 号、5 号和 7 号从站的诊断地址分别为 1022、1021 和 1020，对应的十六进制数为 16#3FE～3FC。主站（2 号站）的诊断地址为 1023，对应的十六进制数为 16#3FF。

4．诊断数据的读取过程

SFC 13 是异步执行的，这就是说，从控制参数 REQ 的上升沿触发读请求，到读完 DP 从站的诊断数据的过程中，需要重复调用 SFC 13。可以根据 SFC 13 的输出参数 BUSY 判断是否读取完了 DP 从站的诊断数据，如果没有读完（BUSY = 1），则返回去重新调用 SFC 13，只有这样才能确保诊断数据被完全读取。

7.1.2 在 OB86 中调用 SFC 13

在 SIMATIC 管理器中生成 OB82、OB86 和 OB122。可以在 OB1、OB82 和 OB86 中调用 SFC 13。

OB86_EV_CLASS 为 B#16#39 时，故障出现；为 B#16#38 时，故障消失。每个从站的故障刚出现和刚消失时，分别调用 SFC 13，读取的从站诊断信息保存在不同的数据区。

数据块 DB 4、DB 5 和 DB 7 分别用来保存 4 号、5 号和 7 号从站的诊断数据。它们的大小用数组来定义，数组元素为双字。在线时监控数据块中的数组元素比较方便。

在 7 号、4 号、5 号从站刚发生故障时，从站故障标志 M20.0、M30.0 和 M40.0 分别被置位为 1，故障刚消失时被复位为 0。在运行时用变量表可以看到它们与从站故障之间的关系。通过它们，可以用人机界面或 WinCC 画面上的指示灯来显示 3 个从站是否有故障，它们在 8.1.5 节中用来触发报警消息。下面是 OB86 调用 SFC 13 读取从站的诊断数据的程序：

```
程序段 1：记录执行 OB86 的次数
          L     MW    84
          +     1
          T     MW    84
程序段 2：
          L     LW    8                    //局部变量中的从站地址
          L     W#16#3FC
          ==I
          JC    m001                       //7 号从站中断则跳转
          L     LW    8
          L     W#16#3FE
          ==I
          JC    m002                       //4 号从站中断则跳转
          L     LW    8
          L     W#16#3FD
          ==I
          JC    m003                       //5 号从站中断则跳转
          BEU                              //块无条件结束
程序段 3：7 号从站中断处理
m001:     L     #OB86_EV_CLASS             //从站故障刚发生时为B#16#39
          L     B#16#39
          ==I
          JC    m005                       //从站故障刚发生则跳转
m004:     CALL  "DPNRM_DG"                 //从站故障刚结束时调用SFC 13
            REQ       :=TRUE               //为 1（TURE）时请求读取DP从站的诊断数据
            LADDR     :=LW8                //从站诊断地址
            RET_VAL   :=MW22               //故障代码，没有故障则存放实际传送的数据字节数
```

```
          RECORD   :=P#DB7.DBX32.0 BYTE 29    //存放诊断数据的地址区
          BUSY     :=M20.1                    //为 1 表示读取过程未完成
          A    M      20.1
          JC   m004                          //读取过程未完成则跳转
          R    M      20.0                   //复位 7 号从站故障标志
          BEU                                //块无条件结束
m005:     CALL  "DPNRM_DG"                   //从站故障刚发生时调用SFC 13
          REQ      :=TRUE                    //为 1 时请求读取 DP 从站的诊断数据
          LADDR    :=LW8                     //从站诊断地址
          RET_VAL  :=MW24                    //故障代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB7.DBX0.0 BYTE 29    //存放诊断数据的地址区
          BUSY     :=M20.2                   //为 1 表示读取过程未完成
          A    M      20.2
          JC   m005                          //读取过程未完成则跳转
          S    M      20.0                   //置位 7 号从站故障标志
          BEU                                //块无条件结束
程序段 4：4 号从站中断处理
m002:     L    #OB86_EV_CLASS                //从站故障刚发生时为B#16#39
          L    B#16#39
          ==I
          JC   m007                          //从站故障刚发生则跳转
m006:     CALL  "DPNRM_DG"                   //从站故障刚结束时调用 SFC 13
          REQ      :=TRUE                    //为 1 时请求读取 DP 从站的诊断数据
          LADDR    :=LW8                     //从站诊断地址
          RET_VAL  :=MW32                    //故障代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB4.DBX20.0 BYTE 13   //存放诊断数据的地址
          BUSY     :=M30.1                   //为 1 表示读取过程未完成
          A    M      30.1
          JC   m006                          //读取过程未完成则跳转
          R    M      30.0                   //复位 4 号从站故障标志
          BEU                                //块无条件结束
m007:     CALL  "DPNRM_DG"                   //从站故障刚发生时调用 SFC 13
          REQ      :=TRUE                    //为 1 时请求读取 DP 从站的诊断数据
          LADDR    :=LW8                     //从站诊断地址
          RET_VAL  :=MW34                    //故障代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB4.DBX0.0 BYTE 13    //存放诊断数据的地址区
          BUSY     :=M30.2                   //为 1 表示读取过程未完成
          A    M      30.2
          JC   m007                          //读取过程未完成则跳转
          S    M      30.0                   //置位 4 号从站故障标志
          BEU                                //块无条件结束
程序段 5：5 号从站中断处理
m003:     L    #OB86_EV_CLASS                //从站故障刚发生时为B#16#39
          L    B#16#39
          ==I
          JC   m009                          //从站故障刚发生则跳转
```

```
m008:   CALL   "DPNRM_DG"                    //从站故障刚结束时调用SFC 13
          REQ      :=TRUE                    //为 1 时请求读取DP从站的诊断数据
          LADDR    :=LW8                     //从站诊断地址
          RET_VAL  :=MW42                    //故障代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB5.DBX20.0 BYTE 13   //存放诊断数据的地址
          BUSY     :=M40.1                   //为 1 表示读取过程未完成
        A      M      40.1
        JC     m008                         //读取过程未完成则跳转
        R      M      40.0                  //复位 5 号从站故障标志
        BEU                                 //块无条件结束
m009:   CALL   "DPNRM_DG"                    //从站故障刚发生时调用SFC 13
          REQ      :=TRUE                    //为 1 时请求读取DP从站的诊断数据
          LADDR    :=LW8                     //从站诊断地址
          RET_VAL  :=MW44                    //故障代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB5.DBX0.0 BYTE 13    //存放诊断数据的地址区
          BUSY     :=M40.2                   //为 1 表示读取过程未完成
        A      M      40.2
        JC     m009                         //读取过程未完成则跳转
        S      M      40.0                  //置位 5 号从站故障标志
```

7.1.3 在 OB82 中调用 SFC 13

7 号从站 ET 200M 的 2AO 模块出现电流输出电路断开等故障时，产生诊断中断，CPU 调用一次 OB82，在 OB82 中调用 SFC 13 来读取诊断信息。输出电路接通时，故障消失，又调用一次 OB82 和 SFC 13。每次调用 OB82 时 MW80 加 1。

程序段 1：记录调用 OB82 的次数

```
        L      MW     80
        +      1
        T      MW     80
```

程序段 2：用 SFC 13 读取 7 号从站的诊断数据

```
        L      B#16#39
        L      #OB82_EV_CLASS                //从站故障刚发生时为 B#16#39
        ==I
        JC     m002                          //从站故障刚发生则跳转
m001:   CALL   "DPNRM_DG"                    //中断事件刚结束时调用SFC 13
          REQ      :=TRUE                    //为 1 时请求读取DP从站的诊断数据
          LADDR    :=W#16#3FC                //7 号从站的诊断地址
          RET_VAL  :=MW52                    //错误代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB82.DBX32.0 BYTE 29  //存放诊断数据的地址区
          BUSY     :=M50.1                   //为 1 表示读取过程未完成
        A      M      50.1
        JC     m001                          //读取过程未完成则跳转
        BEU
m002:   CALL   "DPNRM_DG"                    //中断事件刚发生时调用SFC 13
          REQ      :=TRUE                    //为 1 时请求读取DP从站的诊断数据
```

```
LADDR     :=W#16#3FC                  //7 号从站的诊断地址
RET_VAL   :=MW54                      //错误代码，没有故障则存放实际传送的数据字节数
RECORD    :=P#DB82.DBX0.0 BYTE 29     //存放诊断数据的地址区
BUSY       :=M50.2                     //为 1 表示读取过程未完成
A      M      50.2
JC     m002                           //读取过程未完成则跳转
```

7.1.4 在 OB1 中调用 SFC 13

1．在 OB1 中调用 SFC 13 的原因

编程的一条重要的原则是尽量减少中断程序的执行时间。SFC 13 读取诊断数据的操作采用异步执行方式，需要多次调用 SFC 13 才能完成。在编程时通过判断 SFC 13 的 BUSY 信号的状态来确定是否继续调用 SFC 13。如果在中断组织块（例如 OB86）中调用 SFC 13 来读取从站故障的诊断数据，中断组织块执行的时间可能较长。

为了解决这一问题，可以在 OB1 中调用 SFC 13。以 7 号从站为例，它出现故障时，在 OB86 中将 M20.0 置位为 1；故障消失时，在 OB86 中将 M20.0 复位为 0。OB1 的程序检测到 M20.0 的上升沿或下降沿时调用 SFC 13，将读取的诊断数据分别保存到 DB 7 中不同的区域。对其他从站的故障诊断可以按同样的方法处理。4 号和 5 号从站的故障刚发生和刚结束时，分别将 M30.0 和 M40.0 置位和复位。在运行时用变量表监控 M20.0、M30.0 和 M40.0，可以看到它们与各从站故障出现和消失之间的关系。

2．项目简介

在前面介绍的项目 SFC_13 的基础上，创建一个名为 OB1SFC13 的项目（见随书光盘中的同名例程），两个项目的硬件结构和程序结构相同，OB82 中的程序完全相同。

3．OB86 的程序设计

将项目 SFC_13 的 OB86 中调用 SFC 13 的程序删除，保留下列程序：MW84 加 1、保存 OB 的局部变量、根据局部变量字 LW8 判断是哪个从站触发的中断、对从站故障标志 M20.0、M30.0 和 M40.0 置位和复位。下面是对 7 号从站故障标志 M20.0 置位和复位的程序：

程序段 4：7 号从站故障标志处理

```
M001:   L    #OB86_EV_CLASS
        L    B#16#39              //中断事件刚发生时为 B#16#39
        ==I
        JC   M005                 //中断事件刚发生则跳转
        R    M      20.0          //中断事件刚结束时复位 7 号从站故障标志
        BEU                       //块无条件结束
M005:   S    M      20.0          //中断事件刚发生时置位 7 号从站故障标志
        BEU
```

4．OB1 的程序设计

在 M20.0、M30.0 和 M40.0 的上升沿和下降沿调用 SFC 13，分别读取 7 号、4 号和 5 号从站的诊断数据。FP 和 FN 分别是上升沿检测和下降沿检测指令。

SFC 13 的请求信号 REQ 在 M20.0、M30.0 和 M40.0 的上升沿和下降沿时有效，持续的时间为一个扫描循环周期。用跳转指令 JC 实现的 SFC 13 的循环调用是在 CPU 的一个扫描循环周期内完成的。

下面是读取 7 号从站诊断数据的程序。在程序段 1，如果不是 7 号从站故障刚出现时产生的诊断中断，SFC 13 的使能位 M20.2 为 0，不会触发 SFC 13 来读取诊断数据。BUSY 信号 M21.3 为 0 状态时（读取诊断数据的操作结束），跳转指令的条件不满足，将进入程序段 2。

```
程序段 1：读取 7 号从站故障刚出现时的诊断数据
          A        M    20.0
          FP       M    20.1                    //上升沿检测
          =        M    20.2                    //7 号从站故障刚出现的一个扫描循环周期为 1
m002:     CALL     "DPNRM_DG"                   //调用 SFC 13
           REQ          :=M20.2                 //7 号从站故障刚出现时请求读取 DP 从站的诊断数据
           LADDR        :=W#16#3FC              //7 号从站的诊断地址
           RET_VAL      :=MW28                  //错误代码，没有故障则存放实际传送的数据字节数
           RECORD       :=P#DB7.DBX0.0 BYTE 29  //存放诊断数据的地址区
           BUSY         :=M21.3                 //为 1 表示读取过程未完成
          A        M    21.3
          JC       m002                         //读取过程未完成则跳转
程序段 2：读取 7 号从站故障刚消失时的诊断数据
          A        M    20.0
          FN       M    20.3                    //下降沿检测
          =        M    20.4                    //7 号从站故障刚消失的一个扫描循环周期为 1
m001:     CALL     "DPNRM_DG"                   //调用SFC 13
           REQ          :=M20.4                 //7 号从站故障刚消失时请求读取 DP 从站的诊断数据
           LADDR        :=W#16#3FC              //7 号从站的诊断地址
           RET_VAL      :=MW26                  //错误代码，没有故障则存放实际传送的数据字节数
           RECORD       :=P#DB7.DBX32.0 BYTE 29 //存放诊断数据的地址区
           BUSY         :=M21.2                 //为 1 表示读取过程未完成
          A        M    21.2
          JC       m001                         //读取过程未完成则跳转
```

实验时发现在 M20.0、M30.0 和 M40.0 的上升沿、下降沿读取的同一从站的诊断数据基本上相同，可以将代表同一方波信号的上升沿、下降沿的变量（例如上面程序中的 M20.2 和 M20.4）作“或”运算后，作为 SFC 13 的请求信号 REQ，这样可以少调用一次 SFC 13。

7.1.5 ET 200B 的诊断数据结构与诊断结果分析

1．ET 200 的诊断数据结构

在 SIMATIC 产品的用户手册中，可以查阅到产品诊断数据的结构，不同产品的诊断数据的前 6 个字节基本上相同（见表 7-1），详细的信息请参阅 DP 从站的用户手册。在本例中，主站地址为 2。4 号从站 ET 200B-16DO、5 号从站 ET 200B-16DI、7 号从站 ET 200M 的接口模块 IM 153-1 的制造商 ID 分别为 16#0002、16#0001 和 16#801D。

表 7-1　DP 从站的诊断数据结构

字　节	意　义	字　节	意　义
0～2	站状态 1～3	4，5	制造商 ID（标识符）
3	DP 主站的 PROFIBUS 地址	6～	从站特定的其他诊断数据

各系列 ET 200 的站状态字节 1~3 基本上相同（见表 7-2 和表 7-3）。

表 7-2　站状态 1 各位为 1 的意义

位	意　义
0	DP 主站不能访问 DP 从站
1	DP 从站尚未准备好进行数据交换
2	DP 主站发送到 DP 从站的组态数据与 DP 从站的实际组态不匹配
3	有外部诊断信息
4	DP 从站不支持请求的功能
5	DP 主站不能解释 DP 从站的响应
6	DP 从站类型与软件组态不相符
7	其它 DP 主站（不是当前访问 DP 从站的 DP 主站）已组态 DP 从站

站状态 1 的第 0 位为 1 时，DP 主站不能访问 DP 从站。应检查 DP 从站设置的 PROFIBUS 地址是否正确，总线连接器与电缆或光纤是否已连接好，RS-485 中继器的设置是否正确，DP 从站的电源电压是否正常。

有外部诊断信息时（第 3 位为 1），应根据 CPU 和从站的模块信息、模块状态、与站和通道有关的诊断信息，改正所有的错误。错误消失后，在新的诊断信息中，该位将会变为 0。

表 7-3　站状态 2 各位为 1 的意义

位	意　义
0	必须重新组态 DP 从站
1	有诊断消息，在解决问题之前，DP 从站不会运行（静态诊断消息）
2	DP 从站中该位始终为 1
3	已启用该 DP 从站的响应监视器
4	DP 从站已接收到“FREEZE”（冻结）控制命令
5	DP 从站已接收到“SYNC”（同步）控制命令
6	该位始终为 0
7	DP 从站被禁用（取消激活），即已将它从当前处理中隔离出来

站状态 3（DBB2）的值一般为 16#00，第 7 位（最高位）为 1 时，表示与通道有关的诊断消息太多，超过了诊断帧可以表示的消息量。

2．4 号从站电源断电的诊断

用电缆连接 CPU 和各从站的 DP 接口，将系统数据和程序下载到 CPU 后，将 CPU 切换到 RUN 模式。用变量表（见图 7-1）监控 MW80 和 MW84，观察 CPU 调用 OB82 和 OB86 的次数。

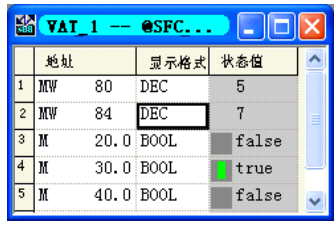


图 7-1　变量表

断开 ET 200B-16DO（4 号从站）模块的电源，CPU 313C-2DP 的 SF LED 亮，BF LED 闪烁后常亮。CPU 调用一次 OB86，MW84 加 1，M30.0 变为 1 状态（True）。

接通 4 号从站的电源，故障消失，又调用一次 OB86，MW84 加 1。M30.0 被复位，CPU 的上述故障 LED 均熄灭。OB86 读取的诊断信息保存在 DB 4 中。

双击打开 SIMATIC 管理器中的 DB 4，点击工具栏上的按钮，启动监控功能。图 7-2 和图 7-3 是 DB 4 中诊断信息的前 8 个字节，后面的字节均为零，未在图中列出。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#010C0002
4.0	ARY[1]	DWORD	DW#16#0	DW#16#00020700

图 7-2　断电时 DB 4 中的诊断数据

20.0	ARY[5]	DWORD	DW#16#0	DW#16#000C0002
24.0	ARY[6]	DWORD	DW#16#0	DW#16#00020700

图 7-3　电源恢复时 DB 4 中的诊断数据

离线的数据块中各数组元素的初始值均为 0，在离线时将数据块下载到 CPU，可以将 CPU 模块中该数据块的数组元素的值全部清零。

ET 200B 的诊断数据结构见随书光盘中的文件《ET 200B Manual_e》的 5.3 节。

断电时各诊断数据的意义如下：

- DBB0（站状态 1）的 16#01 表示主站不能访问该从站，
- DBB1（站状态 2）的 16#0C表示已启用该DP从站的响应监视器。
- DBB3 中的 16#02 是DP主站的地址。
- DBW4 的 16#0002 是ET 200B-16DO的制造商ID。
- DBB6 的 16#07 表示与设备有关（device-related）的诊断数据的长度为 7B。

4 号从站的电源恢复正常后，诊断数据的 DBB0 为 0，表示 DP 主站可以访问该 DP 从站。5 号从站电源断电和恢复时读取的诊断数据与 4 号从站的基本上相同，其区别仅在于制造商 ID 不同。

3．3 个从站的电源同时断电的诊断

3 个从站共用 ET 200M 的电源，由实验可知，同时断开 3 个从站的电源时，变量表中 MW84 的值加 3，说明调用了 3 次 OB86。M20.0、M30.0 和 M40.0 同时变为 1 状态，CPU 的 LED 的状态变化与断开 4 号从站电源时相同。

同时接通 3 个从站的电源时，又调用 3 次 OB86，MW84 的值加 3，M20.0、M30.0 和 M40.0 同时变为 0 状态。CPU 的 LED 的状态恢复正常。

打开 DB 4、DB 5 和 DB 7，点击工具栏上的按钮，启动监控功能。表 7-4 是在 OB86 中读取的诊断数据。

表 7-4　3 个从站电源同时断电时 OB86 中读取的诊断数据

从站状态	ET 200M	ET 200B-16DO	ET 200B-16DI
断开电源	018C 0002 801D 4300	098C 0002 0002 0700	018C 0002 0001 0700
接通电源	000C 0002 801D 4300	000C 0002 0002 0700	000C 0002 0001 0700

3 个从站的诊断数据基本上相同，其区别在于制造商 ID 不同。7 号从站的 IM 153-1 的制造商 ID 为 16#801D。DBB6 为 16#43，其低 6 位表示与标识符有关的诊断数据的字节数为 3。

7.1.6 ET 200M 的诊断数据结构与诊断结果分析

用接在 2AO 模块 0 号通道输出端的小开关断开电流输出电路，CPU、AO 模块和 IM 153 的 SF LED 亮。AO 模块触发诊断中断，CPU 调用 OB82。

1．用 STEP 7 诊断从站和 AO 模块

选中 SIMATIC 管理器左边窗口的 300 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图（见图 7-4）。CPU 与 7 号从站的图标上均有故障符号。

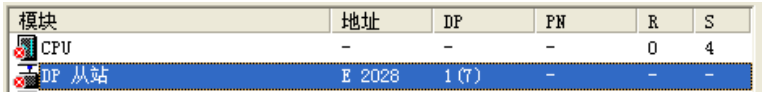


图 7-4　快速视图中的 CPU/故障模块列表

双击 7 号从站所在的行，在打开的 IM 153-1 的模块信息对话框中（见图 7-5），可以看到 6 号插槽的 2AO 模块有故障。点击“十六进制格式”按钮，在打开的对话框中看到的从站的 29B 诊断信息与调用 SFC 13 得到的诊断信息相同。

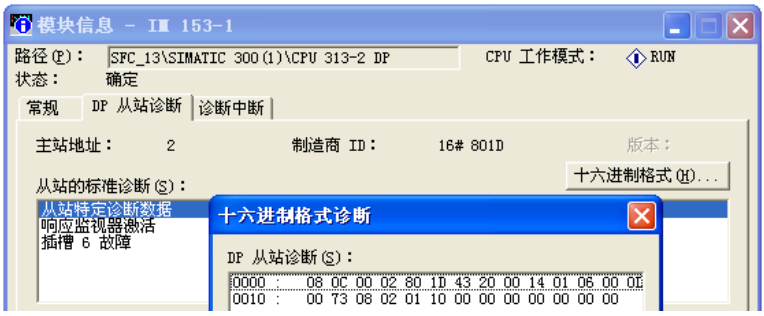


图 7-5　IM 153-1 的模块信息对话框

打开诊断视图（即在线的 HW Config）后，选中 7 号从站，双击下面窗口中的 AO 模块，打开它的模块信息对话框，在“诊断中断”选项卡，可以看到 AO 模块的通道 0 的模拟量输出断线的故障。

2．ET 200M 诊断数据的结构

ET 200M 诊断数据的结构与接口模块 IM 153 的型号有关，表 7-5 是本例使用的 IM 153-1AA02 及以下版本的诊断数据结构（见随书光盘中的《ET 200M 操作说明》）。

表 7-5　ET 200M 诊断数据的结构

字　节　号	意　　义	字　节　号	意　　义
0～2	站状态 1～3	6～8	与标识符有关的诊断数据
3	PROFIBUS 主站地址	9～28	中断（站诊断），最多 20B
4、5	制造商 ID	x～x+7	H 状态（仅用于 S7-400H）

站状态字节 1～3 各位的意义与 ET 200B 基本上相同（见表 7-2 和表 7-3）。IM 153-1 的制造商 ID 为 16#801D。

3．AO 模块电流输出电路断开的诊断信息

开始调试时用 SFC 13 的参数 RECORD 设置的数据区长度为 13B，下载后程序不能正常运行，OB82 中 SFC 13 的返回值为 16#80B1（数据长度错误），改为 29B 后程序正常运行。

用接在 2AO 模块 0 号通道输出端的小开关断开电流输出电路时，CPU 调用一次 OB82，SFC 13 将读取的诊断数据保存在 DB82.DBB0 开始的地址区（见图 7-6）。输出电路接通时故障消失，又调用一次 OB82，SFC 13 将读取的诊断数据保存在 DB82.DBB32 开始的区域（见图 7-7）。

地址	名称	类型	初始值	实际值
0.0	ARY[1]	DWORD	DW#16#0	DW#16#080C0002
4.0	ARY[2]	DWORD	DW#16#0	DW#16#801D4320
8.0	ARY[3]	DWORD	DW#16#0	DW#16#00140106
12.0	ARY[4]	DWORD	DW#16#0	DW#16#000D1500
16.0	ARY[5]	DWORD	DW#16#0	DW#16#00730802
20.0	ARY[6]	DWORD	DW#16#0	DW#16#01100000

图 7-6　输出电路断开时 DB 82 中的诊断数据

32.0	ARY[9]	DWORD	DW#16#0	DW#16#000C0002
36.0	ARY[10]	DWORD	DW#16#0	DW#16#801D4300
40.0	ARY[11]	DWORD	DW#16#0	DW#16#00140106
44.0	ARY[12]	DWORD	DW#16#0	DW#16#00001500
48.0	ARY[13]	DWORD	DW#16#0	DW#16#00730802
52.0	ARY[14]	DWORD	DW#16#0	DW#16#00000000

图 7-7　输出电路接通时 DB 82 中的诊断数据

4．电流输出电路断开时诊断数据的解读

（1）站状态字节

DBB0 中的站状态 1 为 16#08，第 0 位为 0，表示主站可以与 7 号从站通信，IM 153-1 的 BF LED 未亮，通信正常。站状态 1 的第 3 位为 1，表示有外部诊断信息。

DBB1 为 16#0C，表示已启用该 DP 从站的响应监视。

（2）主站地址与制造商 ID

DBB3 的 16#02 是主站的站地址，DBW4 的 16#801D 是 IM 153-1 的制造商 ID。

（3）与标识符有关的诊断数据

DBB6～8 是与标识符有关的诊断数据，用于指明 ET 200M 是否发生故障。

DBB6 为 16#43，其低 6 位为 3，表示与标识符有关的诊断数据的字节数为 3。

模块出现故障时，DBB7 和 DBB8 对应的位被置位。DBB7 为 16#20，表示 6 号 槽的 AO 模块有诊断中断。

（4）诊断数据的中断部分

从站诊断数据的中断部分提供有关中断类型和触发该中断的原因的信息，最多 29B。如果出现模块的通道/通道组 0 的诊断事件，除了通道错误外，可能还有模块错误。

中断信息的内容和长度取决于中断类型，本例程的中断是诊断中断。

图 7-6 的 DBB9 为 16#14，表示中断部分的字节数为 20B，诊断数据一共 29B。

DBB10 为 16#01 表示诊断中断。

DBB11 为 16#06，是产生中断的模块的插槽号。如果为 2，是 IM 153 产生的中断。

DBB12 为 0，表示是过程中断。

从 DBB13 开始，是附加的中断信息：

DBB13 的 16#0D 表示模块通道错误，外部错误和模块故障。

DBB14 的 16#15 表示通道信息可用，模块的类型为模拟量模块。

DBB15 和 DBB16 为 0，没有对应的错误。

DBB17 的 16#73 表示产生中断的模块是模拟量输出模块。

DBB18 的 16#08 是特定通道诊断的长度（8B）。

DBB19 的 16#02 是每个模块的通道数。

DBB20 的 16#01 表示是通道 0 的诊断事件。

DBB21 为 16#10，表示通道断线。

（5）故障消失时诊断数据的变化

0 号通道的电流输出电路接通时，故障消失。

DBB32（站状态 1）变为 0（见图 7-7），表示外部故障消失。

DBB39（对应于图 7-6 的 DBB7）变为 0，表示 6 号槽的故障消失。

DBB45（对应于图 7-6 的 DBB13）变为 0，表示模块故障消失。

DBW52（对应于图 7-6 的 DBW20）为 0，表示通道 0 的断线故障消失。

5．1 号通道电压输出电路短路的诊断数据

用接在 7 号从站 1 号通道输出端的小开关将其输出电路短路，CPU、IM 153 和 AO 模块的 SF LED 亮。AO 模块触发了诊断中断，CPU 调用 OB82，SFC 13 读取的诊断数据见图 7-8。与 0 号通道输出电路开路故障的诊断数据相比，其区别在于 DBB20 为 16#02，表示有故障的是 1 号通道。DBB22 为 16#08，表示输出电路对 M 点短路。

6．两个输出电路同时出现故障时的诊断数据

用接在 7 号从站 AO 模块 0 号通道输出端的小开关断开其电流输出电路，用接在 1 号通道输出端的小开关将其电压输出电路短路，触发了诊断中断，CPU 调用 OB82。CPU、IM 153 和 AO 模块的 SF LED 亮。

OB82 调用 SFC 13 读取的诊断数据，与上述的一个通道出现故障时的诊断数据基本上相同，其区别在于最后 3 个字节。DBB20 为 16#03，表示 0 号和 1 号通道有故障。DBB21 为 16#10，表示 0 号通道的输出电路开路，DBB22 为 16#08，表示 1 号通道的输出电路相对于 M 点短路。

7．DC 24V 电源丢失的诊断数据

消除通道故障后，断开 AO 模块的 DC 24V 电源，触发了诊断中断，CPU 调用 OB82。CPU、IM 153 和 AO 模块的 SF LED 亮。

图 7-9 中 OB82 调用 SFC 13 读取的诊断数据，与上述的通道出现故障时的诊断数据基本上相同，其区别在于 DBB20 开始的最后 3 个字节均为 0，表示没有通道故障。DBB13 为 16#15，第 4 位为 1，表示没有外部辅助电源，负载电源缺失，或电压过低。第 2 位为 1，表示外部错误，第 0 位为 1，表示模块故障。

地址	名称	类型	初始值	实际值
0.0	ARY[1]	DWORD	DW#16#0	DW#16#080C0002
4.0	ARY[2]	DWORD	DW#16#0	DW#16#801D4320
8.0	ARY[3]	DWORD	DW#16#0	DW#16#00140106
12.0	ARY[4]	DWORD	DW#16#0	DW#16#000D1500
16.0	ARY[5]	DWORD	DW#16#0	DW#16#00730802
20.0	ARY[6]	DWORD	DW#16#0	DW#16#02000800

图 7-8　输出电路短路时 DB 82 中的诊断数据

地址	名称	类型	初始值	实际值
0.0	ARY[1]	DWORD	DW#16#0	DW#16#080C0002
4.0	ARY[2]	DWORD	DW#16#0	DW#16#801D4320
8.0	ARY[3]	DWORD	DW#16#0	DW#16#00140106
12.0	ARY[4]	DWORD	DW#16#0	DW#16#00151500
16.0	ARY[5]	DWORD	DW#16#0	DW#16#00730802
20.0	ARY[6]	DWORD	DW#16#0	DW#16#00000000

图 7-9　24V 电源消失时 DB 82 中的诊断数据

7.2　使用 SFC 13 诊断 ET 200S

7.2.1　项目组态与编程

1．项目组态

在 SIMATIC 管理器中创建一个名为 SFC13_S 的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架（见图 7-10）。

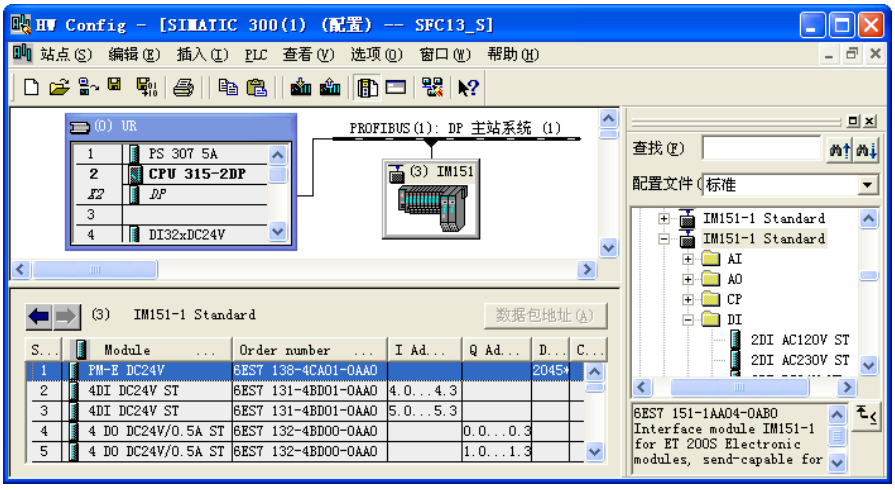


图 7-10　组态 ET 200S 从站

双击机架中 CPU 模块内标有 DP 的行，点击出现的对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成一条 PROFIBUS-DP 网络。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮，返回 HW Config。

打开图 7-10 右边硬件目录窗口的文件夹“\PROFIBUS DP\ET 200S”，将其中的“IM 151-1 Standard”拖放到左边窗口的 PROFIBUS 网络线上。在自动打开的“属性 －PROFIBUS 接口”对话框中，设置该 DP 从站的站地址为 3，点击“确定”按钮，返回 HW Config。

用 IM 151-1 的 DIP 开关设置的从站地址，应与 HW Config 组态的站地址相同。

选中图 7-10 中的 DP 从站，下面的窗口是它的机架中的槽位。打开右边硬件目录窗口中的“\IM 151-1 Standard\PM”子文件夹，将其中的电源模块 PM-E DC24V 拖放到下面的 1 号槽，将 4DI 模块拖放到 2、3 号槽，4DO 模块拖放到 4、5 号槽。图中的模块地址是自动分配的，CPU 用这些地址直接访问从站。

ET 200S 的 I/O 模块的点数很少（例如每块 2 点或 4 点），STEP 7 给每块模块分配一个字节的地址，造成了很多未用的地址“碎片”，给编程和监控带来了不便。可以很方便地将相邻模块的地址合并，方法如下：选中图 7-11 中的 2、3 号槽的 4DI 模块，点击图中的“数据包地址”按钮，2、3 号槽的地址由原来占 4、5 号字节变为只占 4 号字节（见图 7-12）。用同样的方法，可以将物理上相邻的 4 块、每块 2 点的同类模块的地址合并到一个字节。

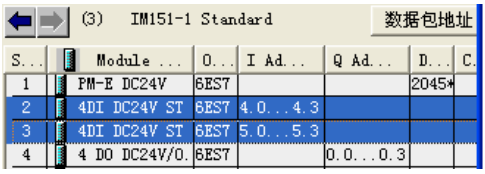


图 7-11　合并前 ET 200S 的输入地址

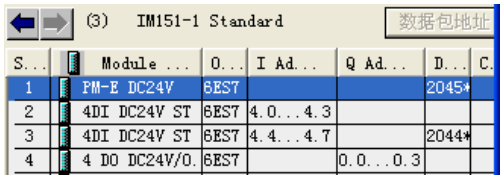


图 7-12　合并后 ET 200S 的输入地址

双击图 7-10 中的 3 号从站，打开从站属性对话框，“General”选项卡中的诊断地址为 2046（16#7FE），调用 SFC 13 时将会用到它。

在“Operating Parameters”（运行参数）选项卡，设置“DP interrupts mode”（DP 中断模式）为 DPV1（见图 7-13）。此外还用复选框设置了一些与中断和诊断有关的选项。

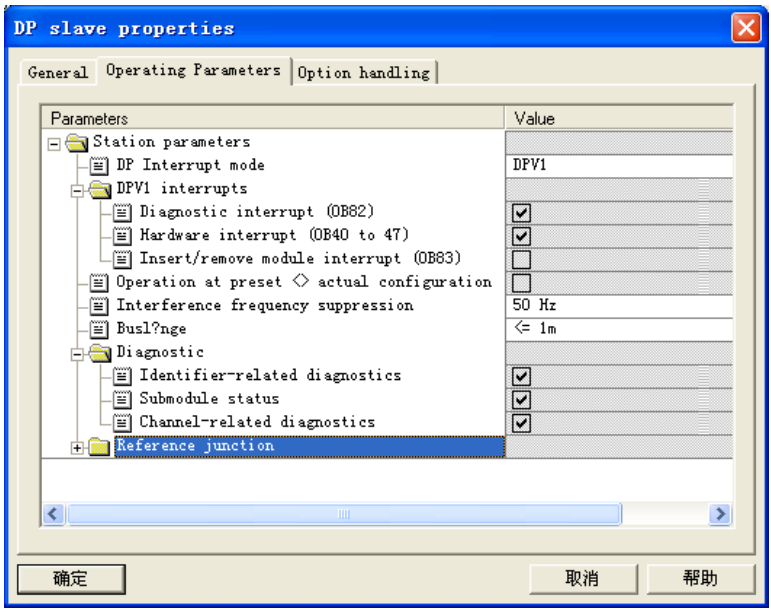


图 7-13　ET 200S 从站属性对话框

点击“确定”按钮，返回 HW Config。选中 3 号从站，双击下面的电源模块，在打开的模块属性对话框的“参数”选项卡中（见图 7-14），选中诊断功能。

双击打开 6 号槽的 2DI 模块，激活它的诊断和硬件中断功能（见图 7-15）。

组态完成后，点击工具栏上的按钮，编译并保存组态信息。

图 7-14　组态 ET 200S 的电源模块

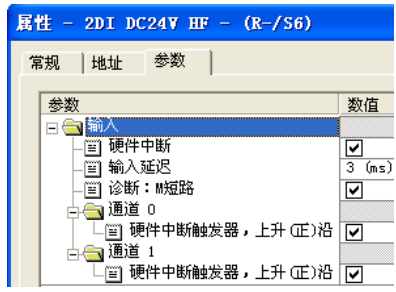


图 7-15　组态 2DI 模块

2．程序设计简介

在 SIMATIC 管理器中生成 OB40（硬件中断组织块）、OB82、OB83、OB86 和 OB122。ET 200S 有模块带电插、拔的功能。运行时拔出、插入模块，CPU 将调用 OB83。

在各中断组织块中，分别将 MW50～MW58 加 1。在 OB40、OB82、OB83、OB86 中调用 SFC 20，分别用 DB 1～DB 4 保存它们的局部变量。

在 OB82 中调用 SFC 13，将诊断数据保存到 DB 82。在 OB86 中调用 SFC 13，在进入事件和离开事件时，分别将诊断数据保存在 DB 86 不同的数据区。

由随书光盘中的文件《ET 200S 操作说明》可知，ET 200S 的诊断数据的长度与 IM 151-1 的型号和从站的模块的块数有关，实验所用的 IM 151-1Standard 的订货号为 6ES7 151-1AA04-0XB0，其诊断数据的最大长度为 110B。考虑到从站的模块很少，调用 SFC 13 时设置其参数 RECORD 的地址区长度为 64B，参数 LADDR 为 W#16#7FE（IM 151-1 的诊断地址 2046）。OB86 调用 SFC 13 的程序与 OB82 的基本上相同。

具体的程序请参阅随书光盘中的项目 SFC13_S。

7.2.2 诊断实验与诊断数据分析

1．拔出 ET 200S 的电源模块

用 DP 电缆连接 CPU 和从站的 DP 接口。将系统数据和程序下载到 CPU 后，将 CPU 切换到 RUN 模式。用变量表（见图 7-16）监控 MW50 和 MW56，观察 CPU 调用 OB82 和 OB86 的次数。

图 7-16　变量表

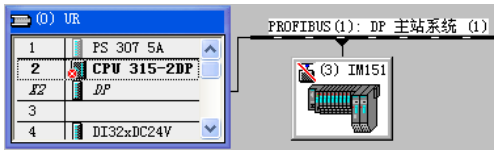


图 7-17　诊断视图

在运行时拔掉 ET 200S 的电源模块，CPU 315-2DP 和 IM 151-1 的 SF LED 亮，BF LED 闪烁，CPU 的 RUN LED 亮，IM 151-1 的 ON LED 亮。CPU 调用一次 OB86，MW56 加 1。在 OB86 中调用 SFC 20“BLKMOV”，将 OB86 的局部变量保存到 DB 4。同时调用 SFC 13，将从站的诊断数据保存到 DB 86。

2．用 STEP 7 诊断故障

选中 SIMATIC 管理器左边窗口的 300 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图，可以看到带故障符号的 CPU 和有故障的 3 号从站。从站上面的诊断符号表示当前组态与实际组态不符合。点击“打开在线站点”按钮，打开诊断视图（见图 7-17）。双击诊断视图中的 3 号从站，打开 DP 从站的模块信息对话框，在“常规”选项卡的“状态”区，可以看到“插入的模块和组态的模块类型”不同的信息。在“DP 从站诊断”选项卡（见图 7-18），诊断信息“Slot 1：no submodule”是 1 号槽没有子模块。点击“十六进制格式诊断”按钮，可以看到从站的十六进制格式的诊断数据。

图 7-18　DP 从站的模块信息

双击诊断视图中的 CPU，打开 CPU 的模块信息对话框。在“诊断缓冲区”选项卡，选中“事件”列表中的“分布式 I/O：站故障”，在下面的“关于事件的详细资料”区，可以看到该事件的详细信息，给出了从站编号、DP 主站系统编号、从站的诊断地址、DP 主站的逻辑基地址，要求调用 OB86，OB 的优先级和信息“外部错误，进入的事件”。

3．诊断数据分析

双击打开 SIMATIC 管理器中的数据块，点击工具栏上的按钮，启动监控功能。图 7-19 和图 7-20 分别是 OB82 和 OB86 的局部变量的前 12 个字节。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55407FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#05230000

图 7-19　OB82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05407FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#07FE0103

图 7-20　OB86 的局部变量

ET 200S 的诊断数据结构与型号有关。作者使用的 IM 151-1 STANDARD 的诊断结构见表 7-6。有关诊断数据的详细信息见随书光盘中的文件《ET 200S 操作说明》。每个从站诊断消息帧只有一个中断的数据。

表 7-6　ET 200S 的诊断数据结构

字　节　号	意　　义	字　节　号	意　　义
0～2	站状态 1～3	15～34	模块状态
3	PROFIBUS 主站地址	35～61	通道特定的诊断，每个通道 3B
4、5	制造商 ID	62～69	H 状态（仅用于 S7-400H）
6～14	ID 特定的诊断	最多 109/127	中断

OB82 和 OB86 读取的从站的诊断数据基本上相同，没有什么本质上的区别。图 7-21 是 OB86 读取的保存在 DB 86 中的诊断数据。下面对诊断数据进行分析。

（1）基本信息

由图 7-21 可知，站状态 1、2（DBW0）为 16#288C，表示 DP 主站不能解释 DP 从站的响应，有外部诊断信息，DP 从站被禁用，已启用该 DP 从站的响应监视器。

DBB3 的 16#02 是主站的站地址。DBW4 的 16#806A 是 IM 151-1 的制造商 ID。

（2）与标识符有关的诊断数据

DBB6～14 是与标识符有关的诊断数据，用于指明 ET 200S 的哪些模块发生故障。

DBB6 为 16#49，低 6 位是与标识符有关的诊断数据的字节数（9B）。用 DBB7～14 的各位来表示模块是否有故障。DBB7 为 1，表示 1 号模块（电源模块）有故障。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#288C0002
4.0	ARY[1]	DWORD	DW#16#0	DW#16#806A4901
8.0	ARY[2]	DWORD	DW#16#0	DW#16#00000000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#00000014
16.0	ARY[4]	DWORD	DW#16#0	DW#16#82000000
20.0	ARY[5]	DWORD	DW#16#0	DW#16#00000000
24.0	ARY[6]	DWORD	DW#16#0	DW#16#00000000
28.0	ARY[7]	DWORD	DW#16#0	DW#16#00000000
32.0	ARY[8]	DWORD	DW#16#0	DW#16#00000080
36.0	ARY[9]	DWORD	DW#16#0	DW#16#00111001
40.0	ARY[10	DWORD	DW#16#0	DW#16#01090D1D
44.0	ARY[11	DWORD	DW#16#0	DW#16#00007D20
48.0	ARY[12	DWORD	DW#16#0	DW#16#01010000
52.0	ARY[13	DWORD	DW#16#0	DW#16#02000000

图 7-21　DB86 的诊断数据

（3）模块状态

DBB15～34（共 20B）是模块状态：

1）DBB15 为 16#14，低 6 位是模块状态的字节数（20B）。

2）DBB16 的 16#82 是模块状态的标志。

3）DBB17、18 一直为 0。

4）DBB19～34（16B）是各模块的状态代码。每个模块的状态占两位，2#00、2#01、2#10 和 2#11 分别表示模块正常、模块有故障、不正确的模块、没有模块（或模块故障）。

DBB19 最低两位是 1 号模块的状态，最高两位是 4 号模块的状态，依此类推。

（4）通道特定的诊断

DBB35 开始是通道特定的诊断数据，提供模块通道错误的详细信息。每个诊断信息占用 3B，本例只有一个诊断信息。第一个字节（16#80）的低 6 位为通道特定的诊断的模块插槽号。DBB37 的错误代码 16#11（17）表示电源模块没有传感器电压或负载电压。

（5）中断的诊断数据

中断数据的起始地址与通道特定的诊断数据有关。本例的中断数据从 DBB38 开始。

DBB38 的 16#10 是中断部分的字节数（16B）。

DBB39 的 16#01 表示诊断中断。

DBB40 的 16#01 是故障模块的插槽号。

DBB41 的 16#09 表示中断序号为 1，至少有一个错误。

DBB42 的 16#0D 表示进入诊断（故障出现），为 0 表示离开诊断。

DBB43 的 16#1D 表示通道信息可用，模块类型为电源模块。

DBW44 为通道特定的诊断的长度和每个模块的通道数，电源模块没有通道，其值为 0。

DBB46 的 16#7D 是通道类型（包括电源模块的几种模块）。

DBB47 的 16#20 是通道特定诊断的长度（32B）。

DBB48 的 16#01 是每个模块的通道数。

DBB49 为 16#01，表示通道 0 的诊断事件。

模块每个通道的故障类型用 4 个字节（DBB50~53）表示，可以同时显示同一个通道的多个故障。DBB52 的 16#02 表示没有传感器电压或负载电压。

（6）故障消失后的诊断信息

重新插入 1 号槽的电源模块，故障消失，调用一次 OB86，MW84 加 1。CPU 和 IM 151-1 的故障 LED 均熄灭。DP 从站模块信息对话框的“DP 从站诊断”选项卡中的故障信息消失，仅保留了监控定时器（Watchdog）被激活的信息。

插入电源模块后，OB86 的局部变量的前 12 个字节见图 7-22。

OB86 调用 SFC 13 读取的诊断信息见图 7-23。

DBB64（站状态 1）由 16#08 变为 0，表示没有外部诊断信息。

DBB71（对应于图 7-21 中的 DBB7）由 16#01 变为 0，表示电源模块的故障消失。

从 DBB81（对应于图 7-21 中的 DBB17）开始的模块状态、通道特定的诊断和中断的诊断数据均变为 0，表示故障消失。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#38C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05407FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#07FE0103

图 7-22　OB86 的局部变量

64.0	ARY[16	DWORD	DW#16#0	DW#16#000C0002
68.0	ARY[17	DWORD	DW#16#0	DW#16#806A4900
72.0	ARY[18	DWORD	DW#16#0	DW#16#00000000
76.0	ARY[19	DWORD	DW#16#0	DW#16#00000014
80.0	ARY[20	DWORD	DW#16#0	DW#16#82000000

图 7-23　DB 86 中的诊断数据

4．拔出和插入 I/O 模块的故障诊断

拔出插槽 2 的 DO 模块，CPU 315-2DP 和 IM 151-1 的 SF LED 亮，BF LED 闪烁。CPU 调用一次 OB86，MW84 加 1。

打开 DP 从站的模块信息对话框（见图 7-24），由“DP 从站诊断”选项卡可知，2 号插槽有故障，没有模块。图 7-24 同时给出了从站的十六进制格式的诊断数据。

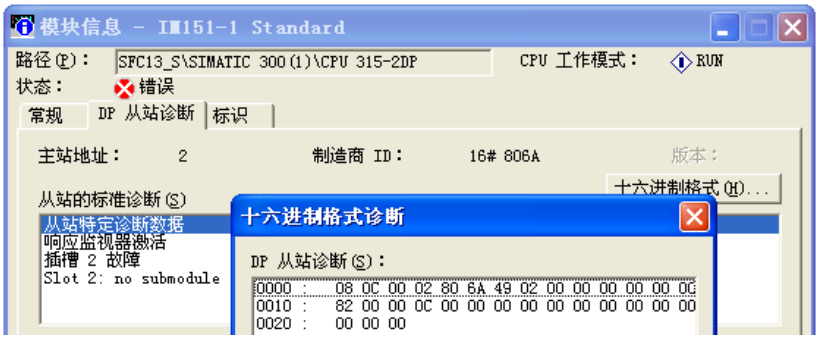


图 7-24　IM 151-1 模块信息中的诊断信息

图 7-25 是 OB86 读取的诊断数据，站状态 1、2 为 16#0A85，表示有外部诊断信息，DP 从站尚未准备好进行数据交换，DP 从站被禁用，必须重新组态 DP 从站。

DBB7 为 16#02，表示 2 号槽的模块有故障。

DBB19 为 16#0C，表示 2 号槽没有模块。

从 DBB20 开始，诊断数据与图 7-21 中的相同。

插入 2 号槽的 DO 模块后，DP 从站模块信息对话框的“DP 从站诊断”选项卡的故障信

息消失，仅保留了监控定时器被激活的信息。

图 7-26 是插入 2 号槽的模块后，OB86 读取的诊断数据，后面的部分均为 0。站状态 1、2 为 16#080C，表示有外部诊断信息，已启用该 DP 从站的响应监视，DP 主站发送到 DP 从站的组态数据与 DP 从站的实际组态匹配，DP 从站被激活，不需重新组态 DP 从站。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#0A8500FF
4.0	ARY[1]	DWORD	DW#16#0	DW#16#806A4902
8.0	ARY[2]	DWORD	DW#16#0	DW#16#00000000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#00000014
16.0	ARY[4]	DWORD	DW#16#0	DW#16#8200000C
20.0	ARY[5]	DWORD	DW#16#0	DW#16#00000000
24.0	ARY[6]	DWORD	DW#16#0	DW#16#00000000
28.0	ARY[7]	DWORD	DW#16#0	DW#16#00000000
32.0	ARY[8]	DWORD	DW#16#0	DW#16#00000080
36.0	ARY[9]	DWORD	DW#16#0	DW#16#00111001
40.0	ARY[10	DWORD	DW#16#0	DW#16#01090D1D
44.0	ARY[11	DWORD	DW#16#0	DW#16#00007D20
48.0	ARY[12	DWORD	DW#16#0	DW#16#01010000
52.0	ARY[13	DWORD	DW#16#0	DW#16#02000000

图 7-25　拔出 2 号槽的模块的诊断数据

64.0	ARY[16	DWORD	DW#16#0	DW#16#080C0002
68.0	ARY[17	DWORD	DW#16#0	DW#16#806A4902
72.0	ARY[18	DWORD	DW#16#0	DW#16#00000000
76.0	ARY[19	DWORD	DW#16#0	DW#16#00000014
80.0	ARY[20	DWORD	DW#16#0	DW#16#8200000C

图 7-26　插入 2 号槽的模块的诊断数据

7.3　DP 主站与智能从站的相互诊断

7.3.1　项目组态与编程

1．组态 DP 主站和 DP 网络

在 STEP 7 中创建一个名为“MS_Diag”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。选中该站，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，点击出现的 DP 接口属性对话框的“常规”选项卡中的“属性”按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成一条 PROFIBUS-DP 网络。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。返回 DP 接口属性对话框，在“地址”选项卡可以看到 DP 主站的诊断地址为 2047（16#7FF）。

返回 HW Config，双击机架中 CPU 所在的行，将 CPU 的 MPI 地址设置为 2。

2．组态智能从站

用鼠标右键点击 SIMATIC 管理器屏幕左边最上面的项目对象“MS_Diag”，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”，插入新的站。选中生成的新站后，双击右边窗口的“硬件”图标，对该站的硬件组态。首先生成该站的机架，然后插入 CPU 313C-2DP、电源模块和信号模块。

将 CPU 放到机架上时，将会自动打开 DP 接口属性对话框的“参数”选项卡。设置 PROFIBUS 站地址为 3，不连接到 PROFIBUS 网络。

返回 HW Config 后，双击 CPU 313C-2DP 所在的行，将它的 MPI 地址设置为 3。双击 DP

所在的行，打开 DP 接口对话框（见图 7-27）。在“工作模式”选项卡将该站设置为 DP 从站。

最后点击按钮，保存对 S7-300 站的组态。

组态时 STEP 7 自动地为智能从站分配了两个诊断地址（见图 7-27），其中的“诊断地址”分配给虚拟插槽 0，另一个分配给虚拟插槽 2。这两个地址的功能如下：

智能 DP 从站用插槽 0 的诊断地址 1022 来接收 DP 主站断电或总线中断的信息。

只有在 DPV1 模式才能看到“插槽”2 的地址（图中为 1021），智能 DP 从站通过该地址检测 DP 主站的操作状态（RUN/STOP）的切换，和触发 DP 主站的诊断中断（OB82）。该地址还用来触发 DP 主站的硬件中断（见 5.2.1 节）。

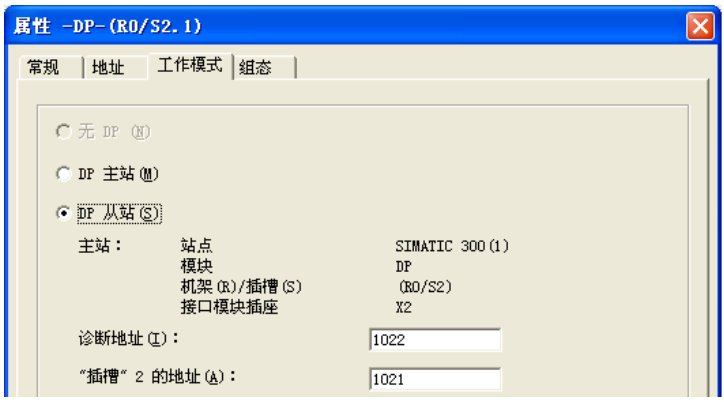


图 7-27　DP 从站属性对话框

3．将智能从站连接到 DP 网络上

返回 DP 主站的硬件组态视图，打开右边的硬件目录窗口中的“\PROFIBUS-DP \Configured Stations”（已组态的站）文件夹，将图标“CPU 31x”拖放到左边窗口中的 PROFIBUS 网络线上。“DP 从站属性”对话框的“连接”选项卡被自动打开，选中列表框中的 CPU 313-2DP，点击“连接”按钮，该站被连接到 DP 网络上，图 7-28 是组态好后的 DP 网络。

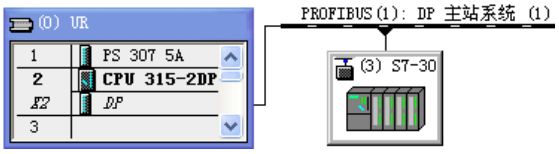


图 7-28　HW Config 中的 DP 主站和智能从站

双击 DP 网络上的智能从站，在打开的“DP 从站属性”对话框的“常规”选项卡中，可以看到两个诊断地址（见图 7-29）。

DP 从站通过诊断地址 2046，向主站报告从站的故障或返回信息，触发主站的 OB86“机架/DP 从站故障”。主站用此地址来调用 SFC 13，可以获取 DP 从站完整的诊断信息。DPV1 从站将此地址分配给虚拟插槽 0。

只有在 DPV1 模式才能看到“插槽”2 地址（2045），它通过主站的诊断中断组织块 OB82，来报告智能从站的工作状态（RUN/STOP）的切换。

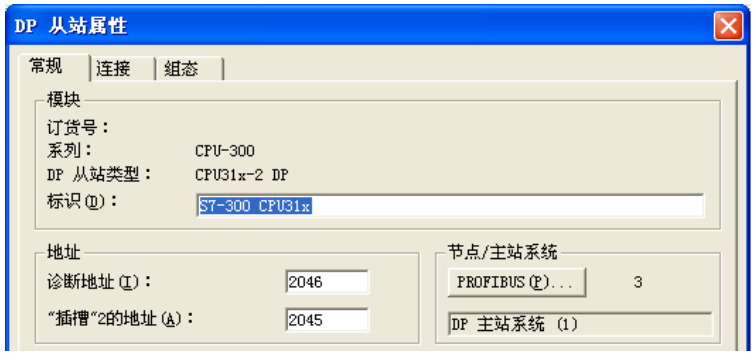


图 7-29　DP 从站属性对话框

4．组态用于主从通信的 I/O 地址区

DP 主站与 DP 从站用于通信的输入/输出地址区如图 7-30 所示。

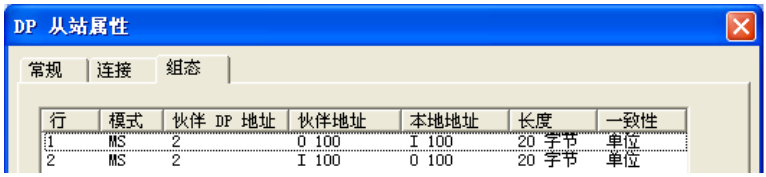


图 7-30　组态用于通信的输入/输出地址区

5．程序设计

在通信双方的初始化组织块 OB100 中，将组态时指定的主站和从站的数据发送区 QB100～QB119 分别预置为 16#2222 和 16#3333，将组态时指定的通信双方的数据接收区 IB100～IB119 清零。在循环中断组织块 OB35 中，每 100ms（OB35 中断的时间间隔的默认值）将数据发送区中的 QW100 加 1。在双方的 OB1 中，将 ID0 传送到发送区的 QD102，用 ID102 接收到的对方的 ID0 的数据来控制本站的 QD4。具体的程序请参阅随书光盘中的项目 MS_Diag。

在 SIMATIC 管理器中为主站和从站分别生成 OB82、OB86 和 OB122。

下面是主站的 OB82 中的程序，ARY 是有 5 个双字的数组。

```
程序段 1：记录中断次数
          L     MW    50
          +     1
          T     MW    50
程序段 2：将 OB82 的局部变量保存到 DB 1 的数组 ARY 中
          CALL   "BLKMOV"
            SRCBLK   :=P#L 0.0 BYTE 20
            RET_VAL  :=LW20
            DSTBLK   :=DB1.ARY
程序段 3：用 SFC 13 读取从站的诊断数据
          L     B#16#39                    //从站故障刚发生时为 B#16#39
          L     #OB82_EV_CLASS
          ==I
          JC    m002                       //从站故障刚发生则跳转
m001:     CALL   "DPNRM_DG"                //中断事件刚结束时调用 SFC 13
```

```
          REQ      :=TRUE                          //为 1 时请求读取DP从站的诊断数据
          LADDR    :=W#16#7FD                      //从站的诊断地址（2045）
          RET_VAL  :=MW42                          //错误代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB82.DBX40.0 BYTE 40        //存放诊断数据的地址区
          BUSY     :=M40.1                         //为 1 表示读取过程未完成
        A      M    40.1
        JC     m001                                //读取过程未完成则跳转
        BEU
m002:   CALL   "DPNRM_DG"                          //中断事件刚发生时调用 SFC 13
          REQ      :=TRUE                          //为 1 时请求读取DP从站的诊断数据
          LADDR    :=W#16#7FD                      //从站的诊断地址
          RET_VAL  :=MW44                          //错误代码，没有故障则存放实际传送的数据字节数
          RECORD   :=P#DB82.DBX0.0 BYTE 40         //存放诊断数据的地址区
          BUSY     :=M40.2                         //为 1 表示读取过程未完成
        A      M     40.2
        JC     m002                                //读取过程未完成则跳转
```

出现 DP 从站故障时，主站调用 OB86，将 MW54 加 1，将 20B 局部变量保存在 DB 2 的数组 ARY 中。用 SFC 13 读取的诊断数据保存在 DB 86 中，SFC 13 的参数 LADDR 仍然是 W#16#7FD。

因为从站不能调用 SFC 13 来读取主站的诊断数据，它的 OB82 和 OB86 没有调用 SFC 13 的程序，程序的其余部分与主站的基本上相同。

7.3.2 DP 主站诊断智能从站的实验

1．正常运行时对数据通信的监控

将系统数据和程序下载到两台 CPU，用电缆连接 CPU 和从站的 DP 接口，将它们切换到 RUN 模式。图 7-31 和图 7-32 是系统运行时主站和从站的变量表。通信双方的 IW100 和 IW118 是接收数据的输入区的第一个字和最后一个字，正常通信时 IW100 不断地增大。用输入端外接的小开关改变 ID0 的值，对方的 QD4 的值随之而变。

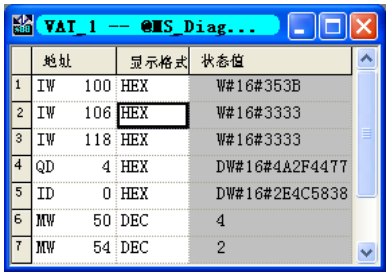


图 7-31　主站的变量表

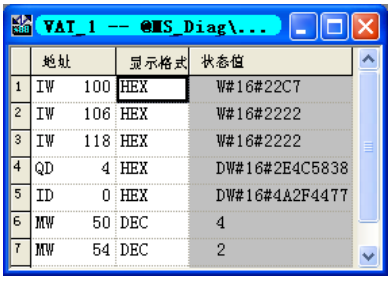


图 7-32　从站的变量表

2．用 STEP 7 诊断从站运行模式的切换

用主站和从站的变量表监控 MW50 和 MW54，观察 CPU 调用 OB82 和 OB86 的次数。

用模式选择开关将 DP 从站从 RUN 模式切换到 STOP 模式，通信中断，变量表中的 IW100 停止变化。主站和从站 QD4 所有的输出点变为 0 状态，模块上各输出点对应的 LED 熄灭。

DP 主站的 RUN 和 SF LED 亮。主站的 MW50 加 1，调用一次 OB82。

选中主站，在 SIMATIC 管理器执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，在打开的快速视图中（见图 7-33），主站 CPU 和从站上均有故障符号，从站的诊断地址为 E2046，E 是“输入”的德语缩写。

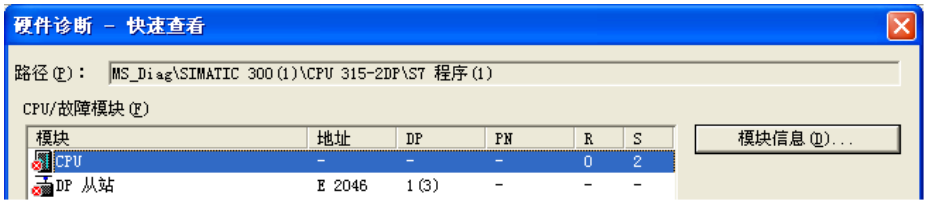


图 7-33　快速视图

选中“CPU/故障模块”列表中的 DP 从站，点击“模块信息”按钮，打开 DP 从站模块信息对话框，“常规”选项卡提供的模块状态为“模块可用且正常，外部出错”。

在“DP 从站诊断”选项卡，“从站的标准诊断”给出的信息是“从站特定诊断信息，响应监视器激活”。点击“十六进制格式”按钮，出现从站的十六进制格式的诊断数据对话框。诊断数据一共 34B，在调用 SFC 13 时，保存诊断数据的参数 LADDR 的地址区的长度应大于等于 34B。

选中快速视图中的 CPU，点击“模块信息”按钮，打开 CPU 的模块信息对话框，选中“诊断缓冲区”选项卡的事件列表第 1 行的“模块故障存在”，在下面的“关于事件的详细资料”区，可以看到下列信息：模块类型为分布式 I/O 从站，输入地址为 2045，工作模式为 STOP，请求调用诊断中断 OB82，优先级为 26；外部错误，进入的事件。

3．OB82 的局部变量分析

在 SIMATIC 管理器中双击打开 DB 1，点击工具栏上的按钮，启动监控功能。图 7-34 是用 DB 1 保存的 OB82 的局部变量的前 12 个字节，后 8 个字节为日期时间值，未在图中给出。

DBB0（OB82_EV_CLASS）为 16#39，表示是进入事件。

DBW6（OB82_MDL_ADDR）为 16#7FD（2045），是从站的虚拟插槽 2 的地址。

DBB8 为 16#01，表示有内部故障。

DBB10 为 16#04，表示 DP 从站的运行模式为 STOP。

4．诊断数据分析

双击打开 SIMATIC 管理器中的 DB 82，点击工具栏上的按钮，启动监控功能。图 7-35 是从站 CPU 切换到 STOP 时，OB82 调用 SFC 13 读取的诊断数据。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55407FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#010B0400

图 7-34　切换到 STOP 时 OB82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#080C0002
4.0	ARY[1]	DWORD	DW#16#0	DW#16#80D04202
8.0	ARY[2]	DWORD	DW#16#0	DW#16#06820000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#04001401
16.0	ARY[4]	DWORD	DW#16#0	DW#16#0201010B
20.0	ARY[5]	DWORD	DW#16#0	DW#16#04000000

图 7-35　切换到 STOP 的诊断数据

S7-300 CPU 的诊断数据结构见表 7-7。诊断数据的详细信息见随书光盘中的文件《CPU 31xC 和 CPU 31x 安装操作说明》。

表 7-7　S7-300 CPU 的诊断数据结构

字　节　号	意　　义	字　节　号	意　　义
0～2	站状态 1～3	6～x-1	ID（标识符）特定的诊断
3	PROFIBUS 主站地址	x～y-1	模块状态
4、5	制造商 ID	y～z	中断状态（设备特定的状态）

ID 特定的诊断和模块状态的长度取决于已组态的地址区的数量，“中断状态”部分的长度取决于中断的类型。

（1）基本信息

由图 7-35 的诊断数据可知，站状态 1、2 为 16#080C，表示有诊断中断，DP 从站的监控定时器被启用。DBB3 的 16#02 是主站的站地址。DBW4 的 16#80D0 是 CPU 313C-2DP 的制造商 ID。

（2）与标识符有关的诊断数据

DBB6～14 是与标识符有关的诊断数据。

DBB6 为 16#42，低 6 位是与标识符有关的诊断数据的字节数（2B）。

DBB7 为 16#02，表示 DP 从站处于 STOP 模式。

（3）模块状态

1）DBB8 为 16#06，低 6 位是模块状态的字节数（6B）。

2）DBB9 的 16#82 是模块状态的标识符。

3）DBW10 一直为 0。

4）DBW12 是各模块的状态代码，16#0400 表示从站的 CPU 有故障。

（4）中断状态

DBB14 开始是中断状态，它提供有关 DP 从站的详细信息，最长 20B。

DBB14 的 16#14 是包括 DBB14 在内的中断状态的字节数（20B）。

DBB15 的 16#01 表示诊断中断。

DBB16 的 16#02 是插槽号，对应于 CPU 所在的 2 号槽。

DBB17 的 16#01 表示进入中断，至少有一个错误未解决。

DBB18 开始的 16B 是与 OB82 有关的中断数据。

DBB18 的 16#01 表示模块有故障。

DBB19 的 16#0B 是传送存储器的地址区（常数）的标志。

DBB20 的 16#04 表示 CPU 处于 STOP 模式。

从 DBB21 开始，后面的字节均为 0。

5．将 CPU 切换到 RUN 模式的 OB82 局部变量分析

将从站的模式选择开关切换到 RUN 位置，故障消失，又调用一次 OB82，MW50 加 1。主站 CPU 的 SF LED 熄灭。DB 1 中的 OB82 的局部变量的前 12B 如图 7-36 所示。

DBB0 为 16#38，表示是离开事件。DBB8 为 0，表示内部故障消失，DBB10 为 0，表示

DP 从站的运行模式为 RUN。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#38421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55407FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#000B0000

图 7-36　切换到 RUN 时 OB82 的局部变量

40.0	ARY[10]	DWORD	DW#16#0	DW#16#000C0002
44.0	ARY[11]	DWORD	DW#16#0	DW#16#80D04200
48.0	ARY[12]	DWORD	DW#16#0	DW#16#06820000
52.0	ARY[13]	DWORD	DW#16#0	DW#16#00001401
56.0	ARY[14]	DWORD	DW#16#0	DW#16#0202000B
60.0	ARY[15]	DWORD	DW#16#0	DW#16#00000000

图 7-37　从站切换到 RUN 的诊断数据

6．将 CPU 切换到 RUN 模式的诊断数据分析

DB82 中的诊断数据如图 7-37 所示。

DBB40（站状态 1）为 0，表示没有外部诊断信息。

DBB47 和 DBB60(对应于图 7-35 中的 DBB7 和 DBB20)为 16#0，表示从站为 RUN 模式。

DBB52（对应于图 7-35 中的 DBB12)为 0，表示 CPU 模块正常。

DBB57（对应于图 7-35 中的 DBB17)变为 16#02，表示离开中断。

DBB58（对应于图 7-35 中的 DBB18)为 0，表示模块没有故障。

7．断开智能从站电源的诊断实验

断开智能从站的电源，通信中断，主站变量表中的 IW100 停止变化，主站 QD0 所有的输出点变为 0 状态，模块上各输出点对应的 LED 熄灭。DP 主站的 RUN 和 SF LED 亮，BF（总线故障）LED 闪烁。

选中 SIMATIC 管理器中的主站，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，在快速视图中（见图 7-38），CPU 和从站均有故障符号。DP 从站上的诊断符号表示当前组态与实际组态不匹配，这是因为通信中断造成的。

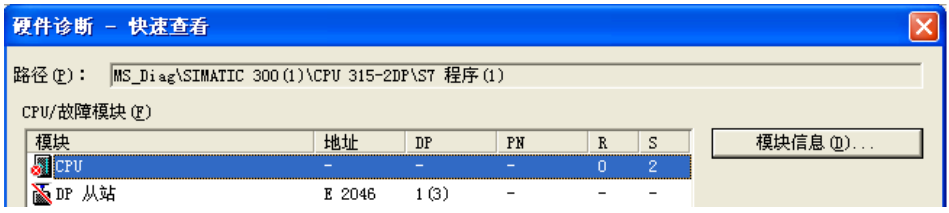


图 7-38　DP 从站电源断开的快速视图

选中“CPU/故障模块”列表中的 DP 从站，点击“模块信息”按钮，打开 DP 从站的模块信息对话框。“常规”选项卡中的从站状态为“模块已组态，但不可用，插入的模块和组态的模块类型不同”，“DP 从站诊断”选项卡的标准诊断信息为“DP 从站不能通过总线访问”。

选中“CPU/故障模块”列表中的主站 CPU，点击“模块信息”按钮，打开主站的模块信息对话框。选中“诊断缓冲区”选项卡事件列表第 1 行的“分布式 I/O：站故障”，在“关于事件的详细资料”区，可以看到下列信息：受影响的 DP 从站的站地址为 3，从站的逻辑基地址为 2046，DP 主站的逻辑基地址为 2047，请求调用机架故障组织块（OB86），优先级为 26；外部错误，进入的事件。点击“事件帮助”按钮，在出现的帮助信息视图中，可以看到 DP 从站发生故障可能的原因是关闭或接通 DP 从站的电源，或连接 DP 从站的电缆断开或接通。

8．诊断数据分析

双击打开 SIMATIC 管理器中的 DB 2，点击工具栏上的按钮，启动监控功能。图 7-39 是 DB 2 中保存的 OB86 的局部变量的前 12 个字节。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05407FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#07FE0103

图 7-39　从站断电时 OB86 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#018C0002
4.0	ARY[1]	DWORD	DW#16#0	DW#16#80D04200
8.0	ARY[2]	DWORD	DW#16#0	DW#16#06820000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#00001401
16.0	ARY[4]	DWORD	DW#16#0	DW#16#0202000B
20.0	ARY[5]	DWORD	DW#16#0	DW#16#00000000

图 7-40　从站断电时 DB 86 中的诊断数据

局部变量中的 LB0 为 16#39，表示是进入事件。主站地址为 16#7FF（2047），出现故障的模块的起始地址为 16#7FE（2046）。DP 主站系统编号为 1，出现故障的从站地址为 3。

双击打开 SIMATIC 管理器中的 DB 86，点击工具栏上的按钮，启动动监控功能。图 7-40 是从站断电时，OB86 调用 SFC 13 读取的诊断数据。

DBB0 的站状态 1 为 16#01，表示主站不能访问智能从站。

DBB1 的站状态 2 为 16#8C，表示有外部诊断信息，从站的监控定时器被启用。

DBB3 是主站的站地址，DBW4 是 CPU 313C-DP 的制造商 ID。

DBW6 是与标识符有关的诊断数据（2B），因为不能与从站通信，没有具体的故障信息。

DBB8 为 16#06，表示模块状态的字节数为 6B。DBB9 的 16#82 是模块状态标识符。

DBB10～13 均为 0，没有模块状态信息。

DBB14 开始是中断状态，它提供有关 DP 从站的详细信息。

DBB14 的 16#14 是中断状态的字节数（20B）。

DBB15 的 16#01 表示诊断中断。

DBB16 的 16#02 是插槽号，对应于 CPU 所在的 2 号槽。

DBB17 的 16#02 表示离开诊断。

DBB19 的 16#0B 是传送存储器的地址区（常数）的标识符。

9．DP 从站电源接通的诊断数据

从站电源接通时，局部变量中的 LB0 为 16#38，表示是离开事件。其他局部变量与从站断电时的相同（见图 7-39）。

7.3.3　智能从站诊断 DP 主站的实验

1．DP 主站运行模式切换的诊断

用模式选择开关将 DP 主站从 RUN 模式切换到 STOP 模式，主站所有的输出点变为 0 状态。通信中断，从站的 SF LED 亮。

选中 SIMATIC 管理器中的从站，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图（见图 7-41），CPU 和 DP 从站上均有故障符号。这里的从站是 CPU 313C-2DP 的通信伙伴，实际上是 DP 主站。图中的 E 1022 是图 7-27 中虚拟插槽 0 的诊断地址。

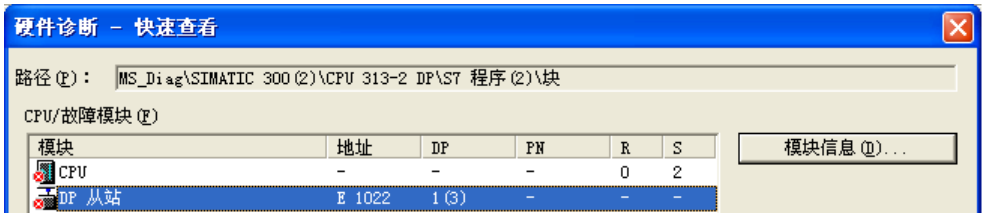


图 7-41　快速视图

选中 DP 从站，点击“模块信息”按钮，打开其模块信息对话框，可以看到故障模块的诊断地址为 I 1022，模块的状态为“模块故障（检测到诊断中断）”。

选中图 7-41 中的 CPU，点击“模块信息”按钮，选中“诊断缓冲区”选项卡事件列表第 1 行的“模块故障存在”，在下面的“关于事件的详细资料”区，可以看到下列信息：模块类型为分布式 I/O 从站，输入地址为 1021，工作模式为 STOP，请求调用诊断中断 OB82，优先级为 26；外部错误，进入的事件。

DP 主站出现故障时，将触发智能从站的中断组织块 OB82 和 OB86，但是不会向 DP 从站提供诊断数据，因此智能从站不能用 SFC 13 读取 DP 主站的诊断数据。

图 7-42 和图 7-43 分别是 DP 主站切换到 STOP 模式和切换到 RUN 模式时，DB 1 保存的 OB82 的局部变量的前 12B，后 8B 为日期时间值，未在图中给出。

图 7-42 中的 DBB0 为 16#39，表示是进入事件。

DBB6 为 16#3FD（1021），是出现故障的模块的虚拟插槽 2 的诊断地址（见图 7-27）。

DBB8 为 16#01，表示有内部故障。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55403FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#010B0400

图 7-42　主站切换到 STOP 时 OB82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#38421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55403FD
8.0	ARY[2]	DWORD	DW#16#0	DW#16#000B0000

图 7-43　主站切换到 RUN 时 OB82 的局部变量

DBB10 为 16#04，表示 DP 主站的运行模式为 STOP。

图 7-43 中的 DBB0 为 16#38，表示是离开事件。

DBB8 为 0，表示没有内部故障。

DBB10 为 0，表示 DP 主站的运行模式为 RUN。

2．主站断电的诊断

断开 DP 主站的电源，通信中断，从站 QD0 所有的输出点变为 0 状态。智能从站的 RUN 和 SF LED 亮，BF LED 闪烁后长亮。

选中 SIMATIC 管理器中的从站，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图，CPU 和 DP 从站均有故障符号（见图 7-44）。DP 从站上的诊断符号表示当前组态与实际组态不匹配，这是因为通信中断造成的。这里的从站是 CPU 313C-2DP 的通信伙伴，实际上是 DP 主站。

选中 DP 从站，点击“模块信息”按钮，打开的模块信息对话框显示“模块已组态，但不可用。预设值/实际值不匹配（插入的模块和组态的模块类型不同）”。

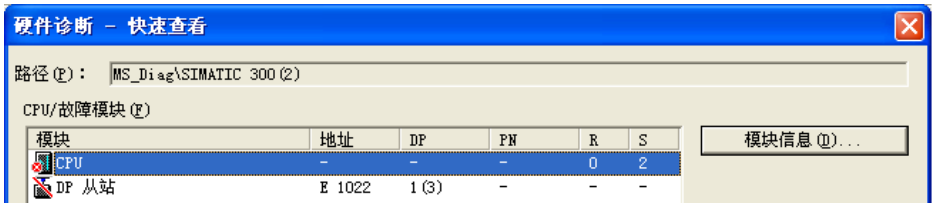


图 7-44　快速视图

选中快速视图中的 CPU，点击“模块信息”按钮，打开 CPU 模块信息对话框。在“诊断缓冲区”选项卡中可以看到下列信息：模块类型为分布式 I/O，站故障，受影响的从站地址为 3。从站的逻辑基地址为 1022，DP 主站的逻辑基地址为 1023，请求调用机架故障组织块（OB86），优先级为 26；外部错误，进入的事件。

图 7-45 和图 7-46 分别是 DP 主站断电和电源恢复时，DB 2 保存的 OB86 的局部变量的前 12B。图中的 16#03FE（1022）是智能从站虚拟插槽 0 的诊断地址（见图 7-27）。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05403FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#03FE0103

图 7-45　主站断电时 OB86 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#38C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05403FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#03FE0103

图 7-46　主站通电时 OB86 的局部变量

7.4　使用 FB 125 或 FC 125 诊断 DP 从站

7.4.1　FB 125 和 FC 125 简介

1．FB 125 和 FC 125

FB 125 是中断驱动的功能块，用于诊断 DP 从站的故障，它可以提供故障从站的详细诊断信息，例如插槽号或模块号、模块状态、通道号和通道故障等。如果 DP/ASI 链接器作为 DP 从站，FB 125 还提供下一层 AS-i 总线系统的信息。如果网络中有诊断中继器，FB 125 将中继器的诊断帧解码，提供错误位置和错误原因的信息。FC 125 是一个较简单的版本，它只提供“哪些站点有故障”的信息，不能显示详细的诊断信息。随书光盘的文件夹“\资料与手册\诊断”中的《FB125 HELP.chm》是 FB 125 和 FC 125 的英文帮助文件。如果不能打开它，将它复制到一个全英文名称的文件夹后再试试，或者在互联网上搜索解决的方法。

随书光盘中的例程 FB_125 在 OB1 中调用 FB 125 和 FC 125，在 OB82 和 OB86 中调用 FB 125。FB 125 内部调用了 SFC 5、SFC 6、SFC 13、SFC 41、SFC 42、SFC 49 和 SFC 51。FC 125 内部调用了 SFC 51。执行 FB 125 时，根据 DP 主站系统的编号和 DP 接口的类型，初始化和启动对 DP 网络的诊断。整个 DP 主站系统被识别，所有被组态的、现存的、有故障或失效的 DP 从站被一个初始化子程序检测。

FC 125 和 FB 125 不能用于 CP 342-5，用于诊断 CP 342-5 的 DP 从站的 FC 3 将在 7.6 节介绍。

2．硬件组态

在 SIMATIC 管理器中创建一个名为 FB_125 的项目，CPU 为 CPU 313C-2DP。在 HW

Config 中生成 DP 主站系统（见图 7-47），实际存在的 4 号、5 号和 7 号从站分别是 ET 200B-16DO、ET 200B-16DI 和 ET 200M，此外还组态了一个并不存在的 8 号从站 ET 200B-16DI/16DO。

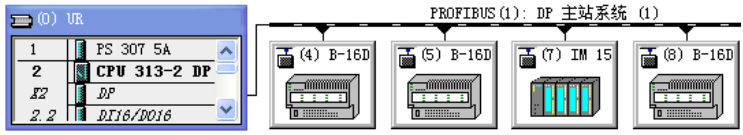


图 7-47　HW Config 中组态的 DP 从站

7.4.2　FB 125 的参数说明

1．调用 FB 125 的程序

如果有几个 DP 主站系统，例如 CPU 集成的 DP 接口和 CP/IM 的 DP 接口分别作 DP 主站，每一个 DP 主站系统应分别调用一个诊断块。

在调用 FB 时，因为块的实参保存在背景数据块中，可以只指定部分形参的实参。在 OB1、OB82 和 OB86 中调用 FB 125 时，使用同一个背景数据块 DB 125 和相同的实参。下面是调用 FB 125 的语句表程序。

```
CALL   FB    125 , DB125                    //调用 DP 诊断功能块
  DP_MASTERSYSTEM          :=1             //DP主站系统编号
  EXTERNAL_DP_INTERFACE:=FALSE             //诊断CPU集成的DP接口的主站系统
  MANUAL_MODE              :=I0.0          //手动模式，诊断单个DP从站
  SINGLE_STEP_SLAVE          :=I0.1          //切换到下一个出错的DP从站
  SINGLE_STEP_ERROR        :=I0.2          //切换到显示的DP从站的下一个错误
  RESET                      :=I0.3          //复位对DP的评估，初始化系统
  SINGLE_DIAG              :=I0.4          //为 1 时在手动模式诊断单个DP从站
  SINGLE_DIAG_ADR          :=MB125         //手动模式单独诊断的DP从站的站号
  ALL_DP_SLAVES_OK         :=M100.0        //为 1 时所有的DP从站运行正常
  SUM_SLAVES_DIAG          :=MB102         //PLC起动时检测到的出错的DP从站的个数
  SLAVE_ADR                :=MB103         //出错的DP从站的站号（1～126）
  SLAVE_STATE              :=MB104         //出错的从站的状态
  SLAVE_IDENT_NO             :=MW106         //显示的从站的PROFIBUS标识号
  ERROR_NO                 :=MB108         //显示的从站的当前错误编号，用于区分不同的错误
  ERROR_TYPE               :=MB109         //当前从站的错误类型编号（见表 7-8）
  MODULE_NO                :=MB110         //当前从站有故障的模块编号（插槽号或模块号）
  MODULE_STATE               :=MB111         //模块状态(0～3)
  CHANNEL_NO               :=MB112         //当前从站的故障模块出错的通道编号
  CHANNEL_TYPE               :=MB113         //出错的通道的类型，见光盘上FB 125 的帮助文件
  CHANNEL_ERROR_CODE       :=MW114         //出错的通道的错误代码，见光盘上FB 125 的帮助文件
  CHANNEL_ERROR_INFO_1     :=MD116         //出错的通道的错误信息，见光盘上FB 125 的帮助文件
  CHANNEL_ERROR_INFO_2     :=MD120         //S7 诊断的附加错误信息，见光盘上FB 125 的帮助文件
  DIAG_COUNTER             :=MB124         //显示的DP从站诊断的总数
  DIAG_OVERFLOW            :=M100.1        //诊断溢出
  BUSY                       :=M100.2        //FB 125 正在诊断DP系统
```

2．FB 125 参数的进一步说明

FB 125 的前 8 个参数是输入变量，其余的是输出变量。位变量为 1 时，该变量的意义如程序中的参数注释所示，为 0 时意义相反。

下面对某些参数作进一步的说明：

1）EXTERNAL_DP_INTERFACE（Bool）：为 1 时为外部 DP 接口（CP/IM），为 0 时为 CPU 集成的 DP 接口。

2）MANUAL_MODE（Bool）：为 1 时为手动模式；为 0 时为自动模式，不支持对单独从站的诊断。

3）SINGLE_STEP_SLAVE（Bool）：在手动模式（MANUAL_MODE=1），按站号增大的方向，显示下一个出错的从站的地址。

4）SINGLE_DIAG（Bool）：只在手动模式下有效，为1 时读 DP 从站的诊断，用 SINGLE_DIAG_ADR 设置该从站的站号。

5）SINGLE_DIAG_ADR（Byte）：单独诊断的 DP 从站的地址，与 SINGLE_DIAG 配合使用，只在手动模式有效。可以用参数 SINGLE_STEP_ ERROR 逐一查看该从站的多个错误。

6）SLAVE_STATE（Byte）：显示的出错的从站的状态，0～3 分别表示正常、连接从站失败、有故障、未组态或无法诊断。

7）ERROR_NO（Byte）：与当前 SLAVE_ADR 参数指示的从站相对应的错误编号，每个编号都有一个对应的错误信息。

8）ERROR_TYPE（Byte）：当前从站的错误类型，见表 7-8。

表 7-8　从站的错误类型

编　　号	意　　义
1	插槽诊断，有故障的插槽/模块的一般属性
2	模块状态，包括插槽或模块的诊断状态
3	符合 DP 标准的通道诊断，定位当前诊断的模块号、通道号和错误类型
4	S7 诊断，定位模块号、通道号和错误类型，仅用于西门子的 S7 从站或 S7 模块
5	单元诊断，从站的诊断数据在背景数据块的第 932～1175 字节
6	电缆诊断，通过诊断中继器检测错误位置和错误原因
7	解码设备指定的诊断

9）MODULE_STATE（Byte）：模块状态，为 0～3 时分别为模块正常、有故障、不正常和没有模块。

10）CHANNEL_ERROR_INFO_1（DWORD，通道错误信息 1）：与当前 SLAVE_ADR 参数指示的从站对应，故障通道的位错误信息。参数 RROR_TYPE 为 3 和 4 时，每一位对应什么错误，可以查阅随书光盘 FB 125 的帮助文件中不同的表格。

ERROR_TYPE 为 4（S7 诊断）时，CHANNEL_ERROR_INFO_1 包含 S7 从站或 S7 模块的诊断数据记录 DS1，如果 AS-i 网络连接到 DP 从站（CHANNEL_TYPE =16#60，即 DP/ASI 链接器），有故障的 AS-i 从站（地址为 0～31）对应的位为 1。

11）CHANNEL_ERROR_INFO_2（DWORD，通道错误信息 2）：与当前 SLAVE_ADR 参数指示的从站对应，ERROR_TYPE 为 4（S7 诊断）时，提供满足 S7 模块数据记录 DS0 的

特殊错误信息，详细信息见随书光盘中 FB 125 的帮助文件中的表格。

12）DIAG_OVERFLOW（Bool）：用 CP、IM 作 DP 主站时，如果同时接收到的诊断个数大于 32，超过 FB 125 可处理的上限，需要用输入变量 RESET 来复位。

3．FB 125 的背景数据块

在 STEP 7 中打开 DB 125，点击工具栏上的按钮，进入在线监控状态，可以通过读取表 7-9 中的背景数据块中的数据，获取诊断信息。

表 7-9　FB 125 的背景数据块中的诊断信息

字　节　段	功　　能
932～1175	当前显示的 DP 从站的标准诊断数据
1176～1191	用位显示的已组态的从站
1192～1207	用位显示的检测到的从站
1208～1223	用位显示的检测不到的从站
1224～1239	用位显示的有故障的从站
1240～1255	用位显示的受影响的从站
1256～1271	用位显示存储的受影响的从站，一直保存到 CPU 启动或 FB 被复位时
1272～1397	每个从站的诊断位，每个从站占一位（见图 7-48），有故障的从站对应的位为 1

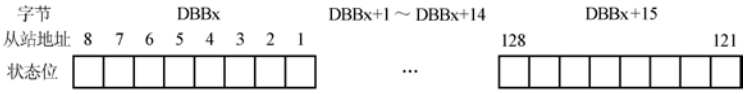


图 7-48　数据块中的从站状态位

7.4.3　使用 FB 125 诊断 DP 从站

项目 SFB_125 有两个变量表，因为例程中的从站地址均小于等于 8，变量表只需监控每一种从站状态的第一个字节。图 7-49 是变量表 VAT_FB125，DB125.DBB1176 是已组态的从站，DBB1192 是检测到的从站，DBB1208 是检测不到的从站，DBB1224 是有故障的从站，DBB1240 是受影响的从站，DBB1256 是存储的受影响的从站。

1．诊断不存在的从站

用 PROFIBUS 电缆连接主站和从站的 DP 接口，接通它们的电源，将程序和组态数据下载到 CPU 后，将 CPU 切换到 RUN 模式。因为组态的从站与实际的从站不一致，CPU 模块上的 SF LED 亮，BUSF LED 闪烁。

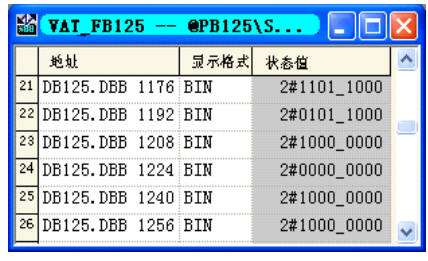


图 7-49　刚进入 RUN 模式的变量表

图 7-49 是刚进入 RUN 模式时 DB 125 中的简要诊断结果，8 号从站对应于变量表中各字节的最高位。因为在组态时“故意”多组态了一个 8 号从站，已组态的从站（DBB1176）的最高位（对应于 8 号从站）为 1，检测不到的从站（DBB1208）的最高位为 1，检测到的从站（DBB1192）的最高位为 0，可知没有检测到 8 号从站。

2．3 个从站同时断电的诊断

断开 ET 200M 的电源，3 个从站的电源消失，组态的 4 个从站都检测不到，DBB1192（检测到的从站）变为全零（见图 7-50）；DBB1208（检测不到的从站）、DBB1240（受影响的从站）和 DBB1256（存储的受影响的从站）的第 3、4、6、7 位均为 1。

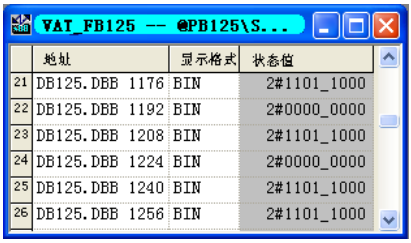


图 7-50　3 个从站断电时的变量表

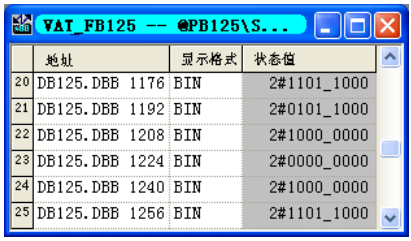


图 7-51　3 个从站电源恢复时的变量表

接通 3 个从站的电源，除了 DBB1256（存储的受影响的从站）之外，诊断的结果恢复原状（见图 7-51）。

3．对同时出现故障的两个从站的诊断

7 号从站 ET 200M 的 6 号槽是一块 2AO 模块，其 0 号通道被组态为电流输出，1 号通道为电压输出，均有组诊断功能（见 6.2 节的图 6-9）。在两个通道的输出端外接两个小开关，在系统运行时用它们来产生模块的输出回路的开路、短路故障。

用下面的方法“人为地”制造两个从站的故障：

1）断开 4 号从站 ET 200B 的电源；

2）断开 AO 模块 0 号通道输出端外接的小开关，出现电流输出回路开路故障。

CPU、IM 153-1 和 AO 模块的 SF LED 亮。7 号从站的 BF LED 未亮，用变量表改写 7 号从站的 DO 模块 QB2 的值，该模块的 LED 的状态随之而变，都说明 7 号从站与主站的通信正常。

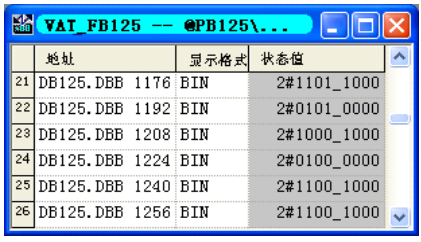


图 7-52　4 号、7 号从站故障时的变量表

变量表 VAT_FB125（见图 7-52）的 DBB1192（检测到的从站）的第 3 位（对应于 4 号从站）变为 0，DBB1208（检测不到的从站）的第 3 位变为 1，都说明检测不到 4 号从站。DBB1240（受影响的从站）和 DBB1256（存储的受影响的从站）的第 3 位和第 6 位变为 1，说明 4 号从站和 7 号从站有故障。

DB125.DBW1256（存储的受影响的从站）用于保存曾经出过问题的 4 个从站。从站故障消失后，用 I0.3 将 FB 125 的 RESET 信号置为 1，可以复位 DBW1240 中曾经出现问题、当前已恢复正常的从站对应的位。同时 MB124 中 DP 从站诊断的总数被清零（见程序中的注释），DBD932 开始的从站的诊断数据保持不变。

4．7 号从站电流输出电路开路的诊断

用接在输入端的小开关将 I0.0 置为 1 状态，FB 125 进入诊断单个 DP 从站的手动模式（见程序中的注释）。在 7 号从站的 2AO 模块的故障仍然存在的情况下，用变量表将 MB125（要诊断的从站的地址）修改为 7，用 I0.4（在手动模式诊断单个 DP 从站）外接的小开关产生一个脉冲，启动对 7 号从站的诊断。

从变量表 VAT_FB125 得到下列诊断信息（见图 7-53）：MB103 中出错的从站的站号为 7（见调用 FB 125 的程序中的注释），MB104 中出错的从站的状态为 2（有故障），MW106 中是从站的制造商 ID，MB108 中从站的当前错误编号为 1，MB109 中当前从站的错误类型编号为 4（S7 诊断，见表 7-8），MB110 中是出错的模块的插槽号，MB111 中模块的状态为 0，表示模块正常。MB112 为出错的通道号（0 号通道）。

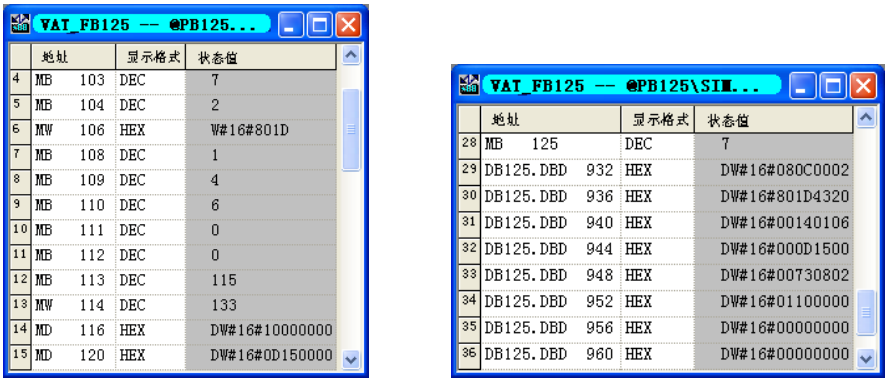


图 7-53　7 号从站电流输出电路开路的诊断数据

MB113 中的 115（16#73）是出错的通道的类型，打开随书光盘中的文件《FB125 HELP.chm》，在文件夹“\Diagnostic block\Integration in STEP 7\Parameters\FB 125”的 CHANNEL_TYPE 下面的表格可以查到，16#73 对应的类型为 Analog-Output（模拟量输出）。

MW114 中出错的通道的错误代码为 133，查阅帮助文件上述文件夹中的 CHANNEL_ERROR_CODE 下面的表格，133 对应的错误为“Wire break”（断线）。

MD116 中是出错的通道的通道错误信息 1。MD116 第 1 个字节的 16#10 的第 4 位为 1。首先找到 FB 125 的帮助文件上述文件夹中的 CHANNEL_ERROR_INFO_1 下面的 S7 signal modules（S7 信号模块），在下面的表格的 Analog Output Module（模拟量输出模块）列中，查到第 1 个字节第 4 位（0.4）对应的故障为 Wire break（断线）。

MD120 中是 S7 诊断的附加错误信息。可以在上述文件夹的 CHANNEL_ERROR_INFO_2 下面的表格中查到错误信息的意义。MB120 的 16#0D 表示模块通道错误、外部错误和 S7 模块故障。MB121 的 16#15 表示通道信息存在，模块的类型为 S7 模拟量模块。

图 7-53 右图的 MB125 中是要诊断的从站的编号，从 DB125.DBD932 开始，是 7 号从站的详细诊断数据，它与用 SFC 13 读取的 DP 从站的诊断数据（见图 7-6）完全相同。

5．7 号从站电压输出电路短路故障的诊断

用输出端外接的小开关接通 7 号从站的 AO 模块 0 号通道的电流输出电路，故障消失。用小开关接通 1 号通道的电压输出电路，产生一个短路故障。AO 模块和 IM 153-1 的 SF LED 亮。用与前面相同的方法启动对 7 号从站的诊断。FB 125 的诊断结果与图 7-53 的区别如下：

1）MB112 中的通道号为 1。

2）MW114 中的通道的错误代码为 132，由帮助文件可知对应的故障为 M short-circuit（相对于 M 点短路）。

3）MD116 中的通道错误信息 1 的第一个字节为 16#08，由帮助文件可知故障为 M short-circuit。

从 DB125.DBD932 开始，是 7 号从站的详细诊断数据，它与图 7-8 中用 SFC 13 读取的诊断数据完全相同。

6．AO 模块电源电压消失的诊断

用输出端外接的小开关断开 7 号从站的 AO 模块 1 号通道的电压输出电路，短路故障消失。断开模块的 DC 24V 电源，AO 模块和 IM 153-1 的 SF LED 亮。

用与前面相同的方法启动对 7 号从站的诊断，FB 125 的诊断结果与图 7-53 的区别如下：

1）MW114 中的通道错误代码为 517，由帮助文件可知为 No external auxiliary voltage（无外部辅助电压）。

2）MD116 中的通道错误信息 1 为 0，没有通道错误。

3）MD120 中的 S7 诊断的附加错误信息（CHANNEL_ERROR_INFO_2）的第一个字节为 16#15，表示没有外部辅助电压、外部错误、S7 模块故障。

从 DB125.DBD932 开始，是 7 号从站的详细诊断数据，它与图 7-9 中用 SFC 13 读取的诊断数据相同。

7．4 号从站断电故障的诊断

消除 7 号从站的 AO 模块的故障，断开 4 号从站的电源。令 I0.0 为 1 状态，FB 125 处于诊断单个 DP 从站的手动模式。用变量表 VAT_FB125 将 MB125 中要诊断的 DP 从站的站号修改为 4，用 I0.4 外接的小开关产生一个脉冲，启动对 4 号从站的诊断。

变量表 VAT_FB125 中的 MB103 为出错的从站的站号（见图 7-54），MB104 中出错的从站的状态为 1，表示连接从站失败。MW106 为从站的制造商 ID，MB109 中当前从站的错误类型为 5，表示单元诊断。

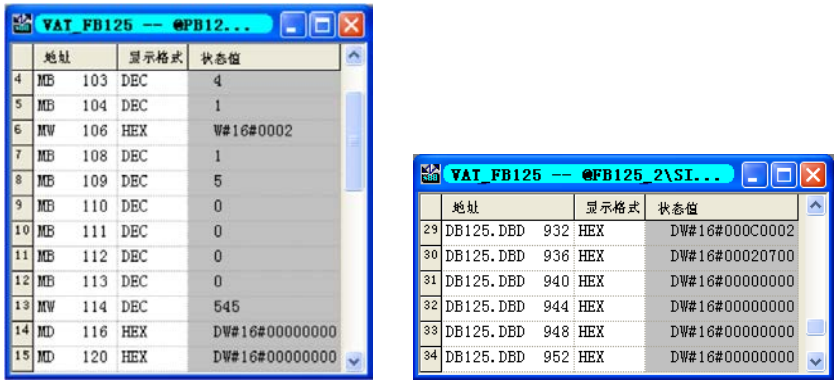


图 7-54　4 号从站断电的诊断数据

MW114 中的通道的错误代码为 545，由帮助文件可知为单元诊断，从站的诊断数据在背景数据块的第 932～944 个字节。诊断数据与图 7-2 中用 SFC 13 读取的基本上相同。

7.4.4　使用 FC 125 诊断 DP 从站

1．调用 FC 125

可以在 OB1、OB82、OB86 和 OB100 中分别调用 FB 125 或 FC 125。项目 FB_125 在 OB82、OB86 和 OB100 中分别将 M200.0～M200.2 置位，M200.0～M200.2 中任何一个为 1 时，在 OB1 中调用 FB 125 或 FC 125。下面是 OB82 将 M200.0 置位的指令：

```
SET
S       M200.0                                          //将调用 FC 125 的标志位置位
```

下面是 OB1 中调用 FC 125 的语句表程序：

```
A       M       200.0                                   //调用 OB82 时 M200.0 被置位
O       M       200.1                                   //调用 OB86 时 M200.1 被置位
O       M       200.2                                   //调用 OB100 时 M200.2 被置位
O       DB1.DBX86.0                                     //FC 125 正在读取诊断数据
S       M       200.3                                   //将 FC 125 的启动信号置位
CALL    FC      125
  CHECK_ACTIV                 := M200.3                 //为 1 时激活诊断
  EXTERNAL_DP_INTERFACE:=FALSE                          //CPU集成的DP接口
  DP_MASTERSYSTEM             :=1                       //DP主站系统编号
  DATA_FIELD                  :=P#DB1.DBX0.0 BYTE 50    //大于等于 50B的内部块处理数据区
  SUM_SLAVES_DIAG             :=DB1.DBW50               //检测到的出错的DP从站的个数
  LIST_SLAVES_NOT_PRESENT:=P#DB1.DBX52.0 BYTE 16        //失效的DP站列表
  LIST_SLAVES_ERROR           :=P#DB1.DBX68.0 BYTE 16   //有故障的DP站列表
  RETVAL                      :=DB1.DBW84               //错误代码返回值
  BUSY                        :=DB1.DBX86.0             //FC 125 正在读取诊断数据
AN      DB1.DBX86.0                                     //FC 125 已执行完
R       M       200.0                                   //复位 OB82 中置位的标志位
R       M       200.1                                   //复位 OB86 中置位的标志位
R       M       200.2                                   //复位 OB100 中置位的标志位
R       M       200.3                                   //复位 FC 125 的启动位
```

FC 125 的前 4 个参数是输入变量，其余的是输出变量。

参数 LIST_SLAVES_NOT_PRESENT（失效的 DP 站列表）和 LIST_SLAVES_ERROR（有故障的 DP 站列表）中，从站编号与数据区中各位的关系见前面的图 7-48。

2．实验结果

FC 125 的诊断结果用名为 VAT_FC125 的变量表来监控（见图 7-55），其中的 DB1.DBW50 是 FC 125 检测出来的出错的 DP 从站的个数，从 DB1.DBB52 开始的 16 个字节是失效的（检测不到的）DP 从站列表，DB1.DBW68 开始的 16 个字节是出错的 DP 从站列表。因为各从站的地址均在失效的从站列表和有故障的从站列表的第一个字节，变量表只监控 DB1.DBB52 和 DB1.DBB68。

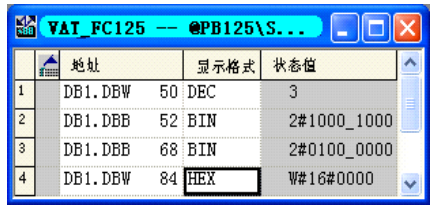


图 7-55　变量表中的诊断数据

同时断开 4 号从站的电源和 7 号从站的 2AO 模块通道 0 的电流输出回路，变量表如图 7-55 所示。4 号从站因为电源断电，主站检测不到它，属于出错的从站，因此 DBB52 的第 3 位为 1。

DBB52 的第 7 位为 1，表示已组态和下载但是并不存在的 8 号从站是出错的从站。

出错的 DP 从站 DBB68 的第 6 位为 1，对应于其 2AO 模块出现故障的 7 号从站 ET 200M。

断开或接通 4 号从站的电源，断开或接通 7 号从站的 2AO 模块的电流输出回路，可以看到 DBW50（出错的从站个数）的值和 DBB52、DBB68 中的对应位随之而变。

7.5 使用 SFC 51 诊断 DP 从站

7.5.1 系统状态表 SSL

所有的诊断事件按它们出现的先后次序登录在诊断缓冲区中。可以通过模块信息和诊断缓冲区等查看系统状态表的信息，也可以在用户程序中用系统功能 SFC 51“RDSYSST”读取系统状态表的诊断信息。

1．系统状态表

系统状态表（System Status List，SSL）用来描述 PLC 的当前状态。

SSL 只能用 SFC 51“RDSYSST”读取，但是不能改写它。与 DP 有关的局部系统状态表是虚拟表，即仅仅在用户程序有请求时，才由 PLC 的操作系统生成它们。

SSL 包含以下的信息：

1）系统数据：包括 CPU 固定的和可以调整的属性数据，用来描述 CPU 的硬件配置、优先权等级和通信的状态等。

2）CPU 内的诊断状态数据：描述系统诊断功能监视的所有部件的当前状态。

3）模块的诊断数据：除了 CPU 之外，其他有诊断功能的模块生成和存储的模块诊断信息和诊断数据。

2．局部系统状态表的结构

局部系统状态表（局部 SSL）由报头（Header）和数据记录组成。报头由局部 SSL 的标识符 SSL-ID、索引（INDEX）、包含在局部 SSL 中的数据记录的字节长度和数据记录的个数组成。局部 SSL 的标识符 SSL-ID 占一个字（见图 7-56），它由局部 SSL 的编号（低字节）、局部 SSL 的摘录号（第 8～11 位）和模块类型（第 12～15 位）组成。

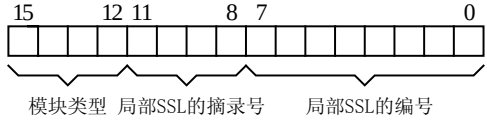


图 7-56　局部系统状态表的标识符

SSL-ID 中的模块类型占用高 4 位，指定要读取的局部 SSL 或它的摘录的模块类型。CPU、IM、FM 和 CP 的模块类型分别为二进制数 0000、0100、1000 和 1100。

某些局部 SSL 或它的摘录（Extract）需要一个对象类型标识符或一个对象号，此时必须使用索引 INDEX。通过局部 SSL 的摘录号，可以指定要读取局部列表的哪一个子集。

SFC 51 的参数 SSL-ID 和 INDEX 决定将要读哪一个局部 SSL 或哪一个局部 SSL 的摘录。

表 7-10 给出了部分局部 SSL 和它们的 SSL-ID 号。参考文献[8]的第 33 章给出了系统状态表详细的信息。

表 7-10　部分局部系统状态表

SSL-ID	局 部 列 表	SSL-ID	局 部 列 表
W#16#xy 11	模块标识	W#16#xy 75	在冗余系统中切换的 DP 从站
W#16#xy 12	CPU 特性	W#16#xy 90	DP 主站的系统信息
W#16#xy 13	用户存储区	W#16#xy 91	模块的状态信息
W#16#xy 14	系统存储区	W#16#xy 92	机架/站的状态信息
W#16#xy 15	块的类型	W#16#xy 94	机架/站的状态信息
W#16#xy 19	模块 LED 的状态	W#16#xy 95	扩展的 DP 主站系统信息
W#16#xy 1C	组件标识	W#16#xy 96	PROFINET IO 和 PROFIBUS-DP 模块状态信息
W#16#xy 22	中断状态	W#16#xyA0	CPU 的诊断缓冲区
W#16#xy 25	过程映像区与 OB 块之间的参数设置	W#16#00B1	模块的诊断信息（数据记录 0）
W#16#xy 32	通信状态数据	W#16#00B2	对应物理地址的模块诊断数据记录 1
W#16#xy 71	冗余 CPU 组态信息	W#16#00B3	带逻辑基地址的模块诊断数据
W#16#xy 74	模块 LED 的状态	W#16#00B4	DP 从站的诊断数据

7.5.2　使用 SFC 51 读取局部系统状态表

1．SFC 51 简介

模块式 DP 从站 ET 200M 使用 S7-300 的全系列模块，有的模块可以发送诊断中断给主站的 CPU，使主站 CPU 调用诊断中断组织块 OB82。可以在 OB82 中调用系统功能 SFC 51“RDSYSST”（Read System Status），来读取一个局部 SSL 或其摘录中的诊断数据。

为了获得 PROFINET IO 系统模块的状态信息，可以调用 SFC 51，读取 SSL W#16#0y91（见参考文献[8]的 33.24 节）。

SFC 51 的输入参数 SZL-ID 是要读取的系统状态列表或部分列表的标识符（即 SSL-ID）。

可以在诊断中断 OB 中调用 SFC 51“RDSYSST”，访问启动该诊断中断的模块，立即读取系统状态列表。

SZL-ID 为 W#16#00B1 时，读取具有诊断功能的模块的前 4 个诊断字节（数据记录 0）。诊断字节各位的意义见参考文献[8]的 33.30 节。

SZL-ID 为 W#16#00B2 时，读取集中式机架中的模块的全部诊断数据记录（≤220B，数据记录 1），INDEX 的格式为 W#16#xxyy，xx 和 yy 分别是模块所在的机架号和插槽号。但是不能用于读取 DP 从站或从站的模块的诊断数据记录。数据记录的长度与模块的型号有关。

SZL-ID 为 W#16#00B3 时，读取 DP 从站或模块的诊断数据（≤220B，数据记录 1），根据模块的逻辑基地址来选择模块。输入参数 INDEX 的第 0～14 位为模块的逻辑基地址，输出模块时最高位（第 15 位）为 1，输入模块时为 0。数据记录的长度与模块的型号有关。

SZL-ID 为 W#16#00B4 时，读取按照 PROFIBUS 标准 EN 50 170 Volume 2 构建的 DP 从站的诊断数据。可以使用组态的诊断地址选择模块。此时的报头结构如下所示：

1）SSL-ID 为 W#16#00B4。

2）INDEX 为 DP 从站已组态的诊断地址。

3）LENTHDR 为数据记录的长度。最大长度为 240B，如果诊断数据的长度大于 240B（最

多 244B）的标准从站，只读取前 240B，并置位溢出位。

4）数据记录的编号 N_DR 为 1。

数据记录的前 3 个字节为站状态，第 4 个字节为主站号，第 5、6 个字节为供应商 ID。从第 7 个字节开始为特定从站的专用诊断数据。

2．例程中系统的硬件组态

打开随书光盘中的项目 SFC_51（见随书光盘中的同名项目），其 DP 主站为 CPU 313C-2DP，4 号从站为 ET 200B-16DO，5 号从站为 ET 200B-16DI，7 号从站为 ET 200M，它有一块 8DO 模块、一块 16DI 模块和一块 2AO 模块。在 HW Config 中组态 DP 主站系统，将 ET 200B 和 IM 153 连接到 DP 网络上，在 ET 200M 的插槽插入信号模块。

OB82_MDL_ADDR 是有故障的从站或模块的地址，在 OB82 中调用 SFC 51 来读取故障信息时，可以用它来作为 SFC 51 的参数 INDEX。如果可能有多个模块分别触发 OB82 诊断中断，不用对每个 DP 从站的每个模块分别调用 SFC 51。

3．程序设计

SFC 51 的参数 SZL_HEADER 的数据类型为 STRUCT（结构），在 OB82 的局部变量表最下面的空白行生成临时变量 SSL_HEADER，数据类型为 STRUCT。双击打开它后，输入该结构的元素 LENTHDR 和 N_DR（见图 7-57）。LENTHDR 是读取的 SSL 或局部 SSL 的数据记录的长度，N_DR 是读取的数据记录的编号。

图 7-57　局部变量表中的结构

在 SIMATIC 管理器中，打开 CPU 的“块”文件夹，点击鼠标右键，执行快捷菜单中的命令“插入新对象”→“数据块”，生成 DB 1。用同样的方法生成组织块 OB82。双击打开 DB 1，生成一个 32B 的数组。

OB82 的局部变量 OB82_MDL_ADDR 是要读取的有故障模块的地址，局部变量 OB82_IO_FLAG（模块类型）为 B#16#54 时，为输入模块；为 B#16#55 时，为输出模块。

根据 OB82_IO_FLAG 的值，判断出故障模块的类型，如果是输出模块，则将地址 OB82_MDL_ADDR 的最高位置 1。

下面是 OB82 调用 SFC 51 来读取数据记录 1 的语句表程序：

程序段 1：中断次数计数器加 1

```
L     MW    20
+     1
T     MW    20
```

程序段 2：调用 SFC 51 的准备工作

```
L     B#16#54                    //B#16#54 为输入模块
L     #OB82_IO_FLAG              //模块输入/输出类型
==I
JC    _001                       //输入模块则跳转，模块地址最高位不变
L     #OB82_MDL_ADDR
L     W#16#8000
```

```
        OW                                          //将输出模块起始地址的最高位置 1
        T     #OB82_MDL_ADDR
_001:   T     MW     22                             //保存产生诊断中断的模块地址
        L     B#16#39
        L     #OB82_EV_CLASS                        //事件类型标识符
        ==I
        JC    _002                                  //中断事件刚发生则跳转
_003:   CALL  "RDSYSST"                             //中断事件刚结束时调用SFC 51
          REQ          :=TRUE                       //为 1（TURE）时请求读
          SZL_ID       :=W#16#B3                    //要读取的SSL或局部SSL的标识符（ID）
          INDEX        :=#OB82_MDL_ADDR             //局部SSL中模块的类型和地址
          RET_VAL      :=MW24                       //SFC的返回值，执行时出错则返回故障代码
          BUSY         :=M28.2                      //读过程未结束时为 1
          SZL_HEADER   :=#SSL_HEADER                //读取的数据记录的长度与编号
          DR           :=P#DB1.DBX20.0 BYTE 16      //存放读取的SSL或局部SSL的目标区域
        A     M      28.2
        JC    _003                                  //读取未完成则跳转
        JU    _004                                  //无条件跳转
```

程序段 3：

```
_002:   CALL  "RDSYSST"                             //中断事件刚发生时调用SFC 51
          REQ          :=TRUE                       //为 1 时请求读
          SZL_ID       :=W#16#B3                    //要读取的SSL或局部SSL的标识符（ID）
          INDEX        :=#OB82_MDL_ADDR             //局部SSL中模块的类型和地址
          RET_VAL      :=MW26                       //SFC的返回值，执行时出错则返回故障代码
          BUSY         :=M28.3                      //读过程未结束时为 1
          SZL_HEADER   :=#SSL_HEADER                //读取的数据记录的长度与编号
          DR           :=P#DB1.DBX0.0 BYTE 20       //存放读取的SSL或局部SSL的目标区域
        A     M      28.3
        JC    _002                                  //读取未完成则跳转
_004:   L     #SSL_HEADER.LENTHDR
        T     MW     30                             //保存读取的数据记录的长度
        L     #SSL_HEADER.N_DR
        T     MW     32                             //保存读取的数据记录的编号
```

调试时发现，如果只调用一次 SFC 51，中断事件刚发生时读取不到数据记录，将程序改为反复调用 SFC 51，BUSY 信号变为 0 状态时停止调用，解决了这一问题。

用户程序可以对 SFC 51 提供的诊断信息作进一步的分析，并采取相应的措施。

图 7-58 变量表

4．实验结果

用 PROFIBUS 电缆连接 CPU 和 ET 200M 模块的 DP 接口。将程序块和系统数据下载到 CPU，将 CPU 的运行模式开关切换到 RUN。

断开 ET 200M 的 2AO 输出模块的电流输出电路，产生一个输出电路开路故障，CPU、IM 153-1 和 AO 模块的 SF LED 亮。

CPU 每调用一次 OB82，变量表中的 MW20 加 1（见图 7-58）。诊断中断的模块地址 MW22

为 16#8100，最高位（第 15 位）为 1，表示是输出模块（AO）产生的诊断中断，该模块的起始地址为 16#100，即十进制数 256。MW30 中保存的数据记录的长度为 16B，MW32 中是读取的数据记录的编号。SFC 51 读取的数据记录保存在 DB 1 中，事件发生时读取的诊断数据见图 7-59，与 7.1.5 节中用 SFC 13 读取的诊断数据中从 DBB13 开始附加的中断信息相同（见图 7-6）。

地址	名称	类型	初始值	实际值
0.0	DR1[0]	DWORD	DW#16#0	DW#16#0D150000
4.0	DR1[1]	DWORD	DW#16#0	DW#16#73080201
8.0	DR1[2]	DWORD	DW#16#0	DW#16#10000000

图 7-59　事件发生时 DB 1 中的诊断数据

20.0	DR1[5]	DWORD	DW#16#0	DW#16#00150000
24.0	DR1[6]	DWORD	DW#16#0	DW#16#73080200
28.0	DR1[7]	DWORD	DW#16#0	DW#16#00000000

图 7-60　事件结束时 DB 1 中的诊断数据

接通 2AO 模块的输出电路，输出电路开路故障消失，CPU、IM 153-1 和 AO 的 SF LED 熄灭。CPU 又调用一次 OB82，变量表中的 MW20 加 1。SFC 51 读取的数据记录见图 7-60。

7.6　使用 FC 3 诊断 CP 342-5 的 DP 从站

7.6.1　使用 FC 3 诊断的顺序

CPU 集成的 DP 接口和 CP 443-5 的 DP 从站可以用 SFC 13 来诊断故障，CP-342-5 的 DP 从站不能采用这种诊断方法。只能用 CP 的诊断功能和调用 FC 3“DP_DIAG”来诊断它的从站。FC 3 在程序编辑器左边窗口的“\SIMATIC_NET_CP\CP 300”文件夹中。

1．诊断的顺序

通过 CP 342-5 读、写 DP 从站和诊断从站故障的程序均在 OB1 中编写。

1）首先调用 FC 2“DP_RECV”，读取 DP 从站的输入点，查询 DP_RECV 返回的 DP 状态字节 DPSTATUS 中的状态位。

2）如果有 DP 从站没有进行正常的数据传输，DPSTATUS 的第 1 位为 1，此时调用 FC 3，读取站列表，了解哪些从站工作不正常。

3）如果 DP 从站有新的诊断数据，DPSTATUS 的第 2 位为 1，此时调用 FC 3，读取诊断列表，判别哪些从站有新的诊断数据。

4）调用 FC 3，读取诊断列表中为 1 的位对应的从站的诊断数据。读取结束后将诊断列表中的对应位复位。

5）调用 FC 1“DP_SEND”，将发送给从站的数据写入 CP。

2．硬件组态

在 STEP 7 中创建一个名为 FC3_Diag 的项目。在 HW Config 中，将电源模块、CPU 和信号模块插入机架，CPU 模块为 CPU 315-2DP（见图 7-61）。CPU 的 MPI 接口和集成的 DP 接口的地址均为 2，未使用集成的 DP 接口。

将 CP 324-5 插入机架，点击自动打开的“属性– PROFIBUS 接口”对话框的参数选项卡中的“新建”按钮，采用默认的参数，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。

用“拖放”的方法，将一个 ET 200M 从站、一块 ET 200B-16DI 和一块 ET 200B-16DO“挂”到 DP 网络上（见图 7-62）。

图 7-61　SIMATIC 管理器

ET 200B-16DO 的 DP 站地址为 4，分配的输出地址为 QW0。ET 200B-16DI 的 DP 站地址为 5，分配的输出地址为 IW0。ET 200M 的 DP 站地址为 7，各模块的地址见图 7-62。DP 主站系统共有连续的 7B 输出和 4B 输入。

组态结束后，点击工具栏上的按钮，编译并保存组态信息。

将组态信息下载到 CPU 后，打开诊断视图，3 个从站内左上角的小图标为灰色，选中 7 号从站，下面窗口中各模块的图标也是灰色的。不能双击打开模块信息对话框对它们进行诊断，但是可以用 CP 342-5 来诊断它们。

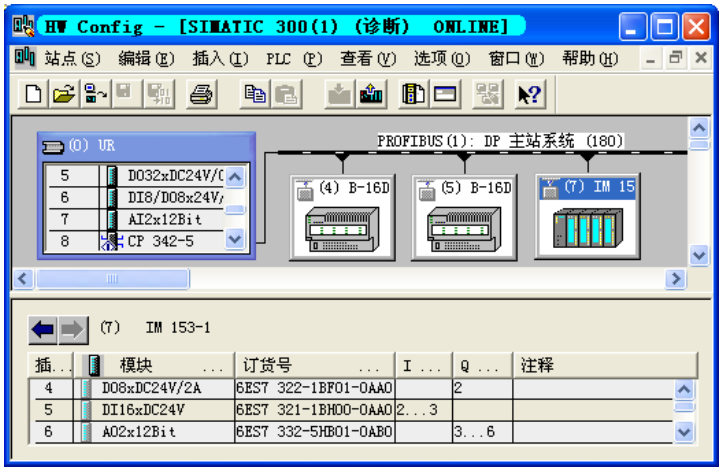


图 7-62　诊断视图

7.6.2　程序设计

1．调用 FC 2

在主程序 OB1 的程序段 1，首先调用 FC 2，读取 DP 从站的输入值和 DP 状态信息。

程序段 1：读取从站的输入

```
CALL   "DP_RECV"                          //调用 FC 2
  CPLADDR    :=W#16#140                   //CP342-5 的模块起始地址 320
  RECV       :=P#DB1.DBX0.0 BYTE 4          //存放接收的数据的地址区指针
  NDR        :=M0.0                       //每次接收完成产生一个脉冲
  ERROR      :=M0.1                       //错误标志位
  STATUS     :=MW2                        //状态字
  DPSTATUS   :=MB4                        //DP状态字节
```

FC 2 的输入参数 CPLADDR 是 CP 模块的起始地址，可以在 CP 的属性对话框的“地址”选项卡中找到它，其十进制数为 320。

2．读取站列表

FC 2“DP_RECV”的 DP 状态字节 DPSTATUS 的第 1 位 M4.1 为 0 时，表示所有的 DP 从站都处于正常的数据传送状态；M4.1 为 1 时，至少一个已组态的 DP 从站没有处于正常的数据传送状态。为了识别这些站，必须用 DP_DIAG 读取和评估 DP 站列表。FC 3 的输入参数 DTYPE（诊断类型）的意义见表 7-11。

表 7-11　诊断类型 DTYPE 的意义

DTYPE	功　能	DTYPE	功　能
0	读取 DP 站列表	5	读取 CPU STOP 模式的 DP 状态
1	读取 DP 诊断列表	6	读取 CP STOP 模式的 DP 状态
2	读取单个站的当前诊断信息	7	读取输入数据(非周期)
3	读取单个站较早的诊断信息	8	读取输出数据(非周期)
4	读取 DP 的状态	10	读取 DP 从站的当前状态

下面是读取 DP 从站列表的程序：

程序段 2：读取站列表

```
      A     M     4.1
      JCN   m001                          //所有从站正常则跳转
      CALL  "DP_DIAG"                     //调用 FC 3
        CPLADDR   :=W#16#140              //CP模块起始地址
        DTYPE     :=B#16#0                //读取DP站列表，见表 7-11
        STATION   :=B#16#3                //DP主站的站地址
        DIAG      :=P#DB1.DBX10.0 BYTE 16 //存放DP站列表的地址指针
        NDR       :=M0.2                  //接收了新数据为 1
        ERROR     :=M0.3                  //错误标志位
        STATUS      :=MW6                   //状态字
        DIAGLNG   :=MB8                   //CP获得的数据的实际长度
m001:   NOP   0
```

组态期间分配给 DP 主站的所有 DP 从站的状态和可用的信息都在 DP 站列表中给出。站列表保存在 PROFIBUS CP 中，并在 DP 轮询周期内持续更新。读入的站列表与通过 FC 2 读取的最新输入数据匹配。

DP 站列表的地址区用 FC 3 的输入参数 DIAG 设置，长度为 16B（128bit），每一位对应于一个 PROFIBUS 地址，即对应于一个 DP 从站（见图 7-63）。

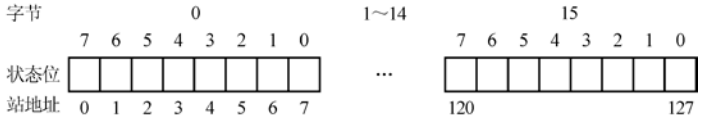


图 7-63　DP 站列表中的从站状态位

状态位代码为 0 可能的含义：

1）组态的从站处于周期性数据传输状态。

2）该从站的输入/输出数据长度被组态为 0，即 DP 主站没有周期性地处理该从站。

3）没有使用该站地址。

状态位代码为 1 表示对应的从站没有处于周期性数据传输状态，可能的原因有：

1）组态的从站不在总线上，或在总线上但是没有响应。

2）从站的组态不正确。

3）组态的从站没有准备好与 DP 主站进行数据传输，仍然处于启动阶段。

每次成功地调用 DP_RECV 后，无论其状态字节怎样，都可以读取 DP 站列表。站列表被读入 FC 3 指定的地址区。

因为只有 4 号、5 号和 7 号从站，它们对应的站列表中的状态位均在 DB1.DBB10 中，可以用变量表监控 DB1.DBB10。

3．读取诊断列表

DPSTATUS 的第 2 位 M4.2 为 0 时，没有新的诊断数据；M4.2 为 1 时，表示至少一个站有新的诊断数据，必须用 FC 3 读取该站的诊断数据。

单个站的历史数据存储在 PROFIBUS CP 中，并根据环形缓冲区的“后进先出”的原则读取。如果 DP 从站的诊断数据频繁地改变，可以在 DP 主站的 CPU 程序中使用该功能获取和评估 DP 从站的诊断数据。

DTYPE 为 4 时，读取的 DP 的状态包括 RUN、CLEAR、STOP 和 OFFLINE，DP 各状态的意义见 3.5.4 节。

程序段 3：读取诊断列表

```
      A      M      4.2                    //DP 状态字节的第 2 位
      JCN    m002                          //没有新的诊断数据时跳转
      CALL   "DP_DIAG"                     //调用 FC 3
        CPLADDR :=W#16#140                 //CP模块起始地址
        DTYPE   :=B#16#1                   //诊断类型为读取诊断列表，见表 7-11
        STATION :=B#16#3                   //DP主站的站地址
        DIAG    :=P#DB1.DBX30.0 BYTE 16    //存放诊断列表的地址指针
        NDR     :=M10.0                    //接收了新数据为 1
        ERROR   :=M10.1                    //错误标志位
        STATUS     :=MW12                    //状态字
        DIAGLNG :=MB14                     //CP获得的数据的实际长度
m002:   NOP    0
```

DP 诊断列表提供哪些 DP 从站已修改了诊断数据，必须通过单个 DP 站诊断功能获取诊断数据。诊断列表保存在 PROFIBUS CP 中，并在 DP 轮询周期内持续更新。一旦其中一个从站改变了诊断信息，就通过来自 DP 从站的高优先级消息实现更新。每次用户程序读出诊断列表后，诊断列表被禁止。只有出现至少一个新条目时，才启用诊断列表，可以随时读取单个 DP 站的诊断信息。

DP 诊断列表的地址区用 FC 3 的输入参数 DIAG 设置，长度为 16B（128bit），每一位对应于一个 PROFIBUS 地址，即对应于一个 DP 从站。诊断位与 DP 站地址的关系与 DP 站列表的相同（见图 7-64）。

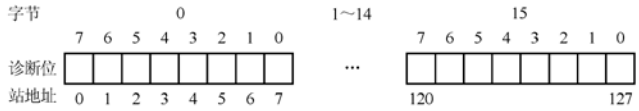


图 7-64　DP 诊断列表中的从站诊断位

DP 诊断列表的位代码为 0 可能具有下列含义（只能是一个含义）：

1）组态的 DP 从站没有新的诊断数据。

2）该从站的输入/输出数据长度被组态为 0，即 DP 主站没有周期性地处理该从站。

3）没有使用该站地址。

DP 诊断列表的某位代码为 1，表示组态的 DP 从站具有新的诊断数据，这些数据可以通过单个 DP 站诊断功能获取。至少一个站具有新的诊断数据时，用 FC 3 读取 DP 诊断列表，并保存在调用 FC 3 时指定的地址区。

在主站的初始化阶段（参数分配与组态），忽略诊断列表中的诊断消息，即用 0 初始化诊断位。如果在初始化 DP 从站期间发生错误，则将该站的诊断位设置为 1。

4 号、5 号和 7 号从站对应的诊断列表的状态位均在 DB1.DBB30 中。

读取诊断列表后，DP_RECV 的输出参数 DPSTATUS 中的“诊断列表有效”状态位（第 2 位，见表 3-6）被复位。读出某个从站的诊断信息之后，用户程序将诊断列表中该站对应的位复位。

4．单个 DP 站诊断

4 号、5 号和 7 号从站对应的诊断列表中的位为 DB 1 中的 DBX30.3、DBX30.2 和 DBX30.0（见图 7-64），某一位为 1 时，调用 FC 3 读取该从站详细的诊断信息。

DP 从站的诊断信息的第 1～3 个字节是站状态字节（见随书光盘中的文档《PROFIBUS CP 组态与调试》的 5.4.3 节），第 4 个字节是已将参数分配给 DP 从站的 DP 主站的 PROFIBUS 地址，16#FF 表示没有参数，16#FE 表示不能通过 PROFIBUS 获得。第 5、第 6 个字节是制造商标识号。

从第 7 个字节开始，是与设备、标识号和通道有关的诊断数据。诊断数据的字节数和各字节的意义与 DP 从站的型号有关，可以查阅从站的用户手册。

ET 200B 数字量模块的诊断数据长度为 13B，实验使用的 ET 200M 的诊断数据的最大长度为 29B。

程序段 4：读取 4 号从站的诊断信息

```
A       DB1.DBX   30.3
JCN     m003                                  //4 号从站没有新的诊断数据时跳转
CALL    "DP_DIAG"                             //调用 FC 3
  CPLADDR     :=W#16#140                      //CP模块起始地址
  DTYPE       :=B#16#2                        //读取单个站的诊断信息（见表 7-11）
  STATION     :=B#16#4                        //DP从站的站地址
  DIAG        :=P#DB4.DBX0.0 BYTE 13          //存放诊断信息的地址指针
  NDR         :=M10.2                         //接收了新数据为 1
  ERROR       :=M10.3                         //错误标志位
  STATUS        :=MW16                          //状态字
```

```
      DIAGLNG    :=MB18                    //CP获得的数据的实际长度
      R      DB1.DBX    30.3              //复位诊断列表中的标志
      S      Q      4.0                   //置位诊断信息指示灯
m003: NOP    0
```

程序段 5：读取 5 号从站的诊断信息

```
      A      DB1.DBX    30.2
      JCN    m004                         //5 号从站没有新的诊断数据时跳转
      CALL   "DP_DIAG"                    //调用 FC 3
        CPLADDR    :=W#16#140             //CP模块起始地址
        DTYPE      :=B#16#2               //读取单个站的诊断信息
        STATION    :=B#16#5               //DP从站的站地址
        DIAG       :=P#DB5.DBX0.0 BYTE 13 //存放诊断信息的地址指针
        NDR        :=M20.0                //接收了新数据为 1
        ERROR      :=M20.1                //错误标志位
        STATUS        :=MW22                 //状态字
        DIAGLNG    :=MB24                 //CP获得的数据的实际长度
      R      DB1.DBX    30.2              //复位诊断列表中的标志
      S      Q      4.1                   //置位诊断信息指示灯
m004: NOP    0
```

程序段 6：读取 7 号从站的诊断信息

```
      A      DB1.DBX    30.0
      JCN    m005                         //7 号从站没有新的诊断数据时跳转
      CALL   "DP_DIAG"                    //调用 FC 3
        CPLADDR    :=W#16#140             //CP模块起始地址
        DTYPE      :=B#16#2               //读取单个站的诊断信息
        STATION    :=B#16#7               //DP从站的站地址
        DIAG       :=P#DB7.DBX0.0 BYTE 29 //存放诊断信息的地址指针
        NDR        :=M20.2                //接收了新数据为 1
        ERROR      :=M20.3                //错误标志位
        STATUS        :=MW26                 //状态字
        DIAGLNG    :=MB28                 //CP获得的数据的实际长度
      R      DB1.DBX    30.0              //复位诊断列表中的标志
      S      Q      4.2                   //置位诊断信息指示灯
m005: NOP    0
```

程序段 7：发送输出数据到从站

```
      CALL   "DP_SEND"                    //调用 FC 1
        CPLADDR    :=W#16#140             //CP342-5 模块的起始地址 320
        SEND       :=P#DB1.DBX4.0 BYTE 7  //存放要发送的数据的地址区
        DONE          :=M30.0                 //每次发送完成产生一个脉冲
        ERROR         :=M30.1                 //错误标志位
        STATUS     :=MW32                 //状态字
```

程序段 8：用 I0.0 复位诊断数据指示灯

```
      A      I      0.0
      R      Q      4.0
      R      Q      4.1
      R      Q      4.2
```

7.6.3 程序运行与监控

1．FC 1 参数错误的诊断

7 号从站 ET 200M 实际上有一块 2AO 模块，最初的程序在调用 FC 1 时未考虑它，FC 1 “DP_SEND”的参数 SEND（要发送的数据的地址区）的长度被设置为 3B。下载组态信息和程序后，将 CPU 和 CP 模块上的开关切换到 RUN 位置，CP 的 SF LED 闪烁，3 个从站的 SF（系统故障）LED 亮，CPU 和 CP 均处于运行模式。

双击 HW Config 中的 CP 342-5，打开其属性对话框，点击“诊断”选项卡中的“运行”按钮，打开 NCM S7 诊断视图（见图 7-65）。“诊断缓冲区”右边窗口第一条信息是“输出数据长度：DP_SEND 块的输出数据长度 3 字节太短，组态的输出数据的长度为 7 字节”。

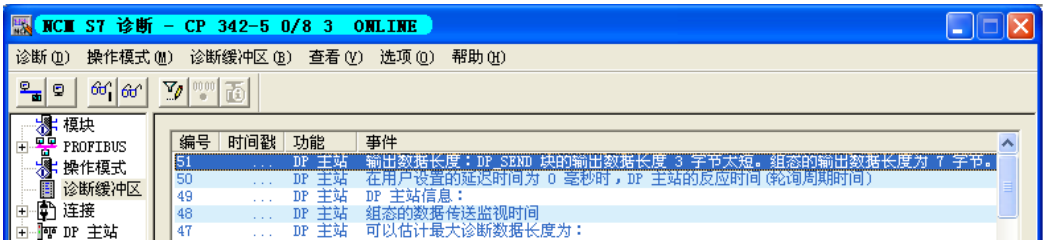


图 7-65　CP 342-5 的诊断视图

诊断缓冲区下面的“解码事件消息”区给出了选中的事件的详细资料。

打开 OB1，点击工具栏上的按钮，启动程序状态监控功能，FC 2“DP_RCV”的状态字 STATUS 为 16#8180（见图 7-66）。由在线帮助可知，16#8180 表示 DP 服务已启动，但是不能接收数据。

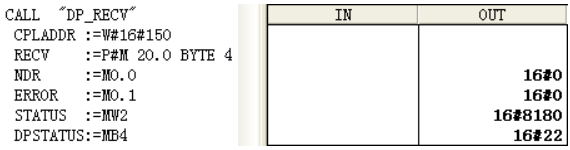


图 7-66　程序状态监控

DP 状态字节 DPSTATUS 为 16#22（2#0010 0010）。由在线帮助和表 3-6 可知，第 4 位和第 5 位为 2#10，表示从站的 DP 状态为 STOP；第 1 位为 1，表示站列表有效。

将 FC 1“DP_SEND”的参数 SEND 的地址区长度改为 7B，重新下载 OB1 后，所有故障 LED 熄灭，系统运行正常。

2．从站正常运行时的监控

下载了程序和系统数据后，接通主站和从站的电源，将 CPU 和 CP 的模式开关扳到 RUN 位置，运行正常，指示通信故障的 LED 未亮。

图 7-67 变量表中的 MB4 是 FC 2“DP_RECV”输出的 DP 状态字节 DPSTATUS（见 OB1 的程序段 1）。DB1.DBB10 是站列表的第 1 个字节，图 7-67 中 MB4 和 DB1.DBB10 均为 0，表示 DP 和各从站运行正常。

用变量表改写 DB1.DBW4 的值后，OB1 的程序段 7 调用 FC 1，将新的输出值发送到 4 号从站。DB1.DBW2 是读取的 7 号从站的数字量输入值，可以用接在模块输入端的小开关改变 DB1.DBW2 的值。

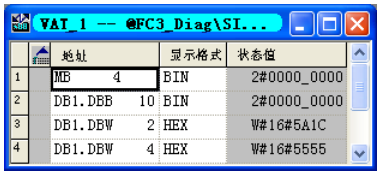


图 7-67　从站正常运行时的变量表

图 7-68　4 号从站断电时的变量表

3．4 号从站断电的故障诊断

断开 4 号从站的电源，观察到下列现象：

1）CP 的 SF LED 闪烁，CPU 和 CP 的 RUN LED 仍然亮。

2）MB4 的第 1 位为 1（见图 7-68），由 FC 2 的在线帮助可知，站列表有效。

3）站列表的第 1 个字节 DB1.DBB10 的值为 2#0000_1000，表示 4 号从站没有与 DP 主站进行数据传输。

4）诊断列表的第 1 个字节 DB1.DBB30 的值为 2#0000_1000，表示 4 号从站有新的诊断数据。OB1 的程序段 4 调用 FC 3 读取 4 号从站的诊断数据后，将 4 号从站对应的 DB1.DBX30.3 复位，因此在变量表中看不到该位的状态变化过程。

双击打开 DB 4。点击工具栏上的  按钮，可以看到诊断数据。图 7-69 和图 7-70 给出了 4 号从站的诊断数据的前 8 个字节，其余部分均为 0。

除了 DBW2 为 16#1103（DP 主站地址为 3），图 7-69 和图 7-70 中的诊断数据与图 7-2 和图 7-3 中用 SFC 13 读取的诊断数据相同。诊断数据各字节的意义可以参考 7.1 节。

地址	名称	类型	初始值	实际值
0.0	Ary[0]	DWORD	DW#16#0	DW#16#01001103
4.0	Ary[1]	DWORD	DW#16#0	DW#16#00020700

图 7-69　4 号从站断电的诊断数据

地址	名称	类型	初始值	实际值
0.0	Ary[0]	DWORD	DW#16#0	DW#16#000C1103
4.0	Ary[1]	DWORD	DW#16#0	DW#16#00020700

图 7-70　4 号从站通电的诊断数据

4 号从站断电时，打开 CP 的诊断视图（见图 7-71），选中左边窗口的 4 号从站，“从站诊断”区中打勾的诊断信息“StationNonExistant”表示该站不存在。

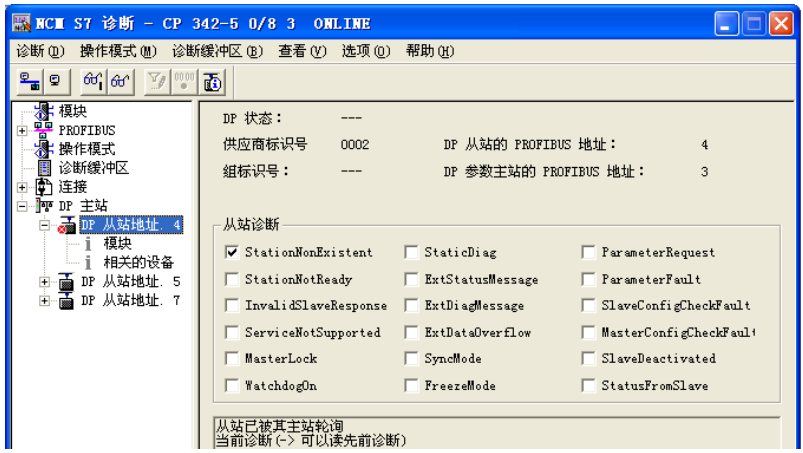


图 7-71　CP 342-5 的诊断视图

选中左边窗口 4 号从站下面的“模块”和“相关的设备”，显示“没有可用的信息”。

4．AO 模块输出电路开路故障的诊断

用接在 7 号从站的 AO 模块 0 号通道输出端的小开关断开电流输出电路，CPU、AO 模块

和 IM 153 的 SF LED 亮，CP 的 SF LED 闪烁。

输出电路断开和接通时 FC 3 读取的诊断数据，与出现同样的故障时用 SFC 13 读取的诊断数据基本上相同（见图 7-6 和图 7-7，仅 DBB3 的主站地址不同）。

选中 CP 342-5 的诊断视图左边窗口的 7 号从站（见图 7-71），右边窗口中的诊断信息与图 6-40 中的基本上相同。

选中左边窗口“7 号从站”中的“模块”，右边窗口的信息提示 6 号槽的 AO 模块有故障（见图 7-72）。

选中左边窗口“7 号从站”中的“相关的设备”，给出的供应商信息（见图 7-73）与用 FC 3 读取的诊断信息的后半部分相同。

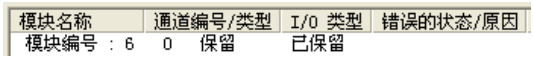


图 7-72　7 号从站的“模块”信息

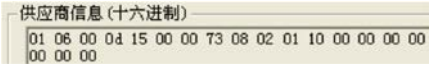


图 7-73　7 号从站“相关的设备”信息

0 号通道的电流输出电路接通后，故障消失，各故障 LED 熄灭。选中左边窗口的 7 号从站，下面的信息是“模块正常（诊断中断离开状态）”。选中“7 号从站”中的“模块”，显示“无可用信息！”。选中“7 号从站”中的“相关的设备”，供应商信息为“01 06 00 00 15 00 00 73 08 02”，后面的字节均为 0。

5．7 号从站 AO 模块电压输出电路短路故障的诊断

用接在 7 号从站 1 号通道输出端的小开关将其输出电路短路，CPU、AO 模块和 IM 153 的 SF LED 亮，CP 的 SF LED 闪烁。1 号通道电压输出电路短路故障消失后，各故障 LED 熄灭。除了 DBW2 为 16#1103 外，故障出现时 FC 3 读取的数据与出现同样的故障时，图 7-8 中用 SFC 13 读取的数据相同。

6．7 号从站 AO 模块 DC 24V 电压丢失的诊断

消除通道故障后，断开 AO 模块的 DC 24V 电源。除了 DBW2 为 16#1103 外，故障出现时 FC 3 读取的数据与出现同样的故障时，图 7-9 中用 SFC 13 读取的数据相同。

7.7　练习题

1. 可以在哪些程序块中调用 SFC 13，怎样获取触发中断的从站地址？
2. 怎样分析 SFC 13 读取的诊断数据？
3. 怎样确定 SFC 13 保存诊断数据的地址区长度？
4. 怎样将 ET 200S 同类相邻模块的地址合并到一个字节？
5. DP 主站怎样诊断智能从站的 RUN/STOP 模式的切换，和从站电源断电的故障？
6. 智能从站怎样诊断 DP 主站的 RUN/STOP 模式的切换，和主站电源断电的故障？
7. 组态一个项目，CPU 315-2DP 为 DP 主站，ET 200S 为 3 号从站，ET 200M 为 4 号从站，ET 200B-16DI 为 5 号从站，ET 200B-16DI 为 6 号从站。编写用 FC 125 诊断从站故障的程序。
8. 用什么 SFC 来诊断 CPU 集成的 DP 接口和 CP 443-5 的 DP 从站的故障？用什么 FC 来诊断 CP 342-5 的 DP 从站？
9. 应按什么顺序编写诊断 CP 342-5 的 DP 从站故障的程序？

第 8 章　故障诊断消息的显示

8.1　与块有关的消息的组态与显示

控制系统的故障诊断信息可以用人机界面(HMI)和上位计算机的组态软件(例如 WinCC)来显示，例如用 SFC 125 来诊断 DP 从站，用画面上的指示灯来显示有故障的 DP 从站。比较理想的显示方式是在报警事件出现时，用 HMI 的报警视图、报警窗口或 WinCC 的报警控件来显示报警消息（简称为消息）。出现故障时，CPU 自动地将消息发送到 HMI 和上位计算机。这种显示方式的编程和组态的工作量少，故障显示及时、准确、简单、方便。本章主要介绍这种消息显示的组态和编程的方法。

8.1.1　消息的分类与生成消息的块

消息（Message）用来向操作人员报告系统当前的运行状态、故障的位置及原因，以供快速检测、定位和排除故障，减少设备的停机时间。

组态时用 STEP 7 创建、编辑和编译消息，运行时 PLC 将消息自动地发送到 WinCC 和 HMI，将故障信息或系统事件显示出来。

消息的传送分为位消息传送和用消息号传送这两种方法。

1．位消息传送

PLC 和操作面板没有共享的公用数据库时采用位消息传送，用 HMI 的组态软件 WinCC flexible 对位消息传送组态，用 PLC 中的位信号作为触发信号。运行时操作面板不断地查询位信号的状态，位信号为 1 时，表示有故障，用 HMI 的报警视图或报警窗口显示报警文本。位信号为 0 时，故障消除，报警文本显示故障消除的信息，消息使用 HMI 的时间标记。因为 HMI 需要扫描 PLC 中故障位的状态，所以通信的负担较重。

2．用消息号传送

使用消息号传送消息时，用编程设备和 HMI 的公用数据库管理消息。组态和编程时将消息分配给用户程序中的某个位。PLC 中该位的状态变化时，只将对应的消息号传送到操作面板，显示出组态时保存在 HMI 或 WinCC 中的消息文本，消息使用 PLC 的时间标记。这种方式的通信负担较轻。

（1）面向 CPU 的消息号分配

消息的属性和文本与使用的 HMI 单元无关，程序可以复制到项目的其他位置或复制到其他项目。如果只复制了单个块，程序必须重新编译。如果改变了消息的文本和属性，这些改变将自动应用于实例，文本可以写在多行中。

组态消息时可能会出现“选择消息号分配”对话框，一般选择面向 CPU 的消息号分配。

（2）面向项目的消息号分配

某些消息属性和文本与使用的 HMI 单元有关，必须面向特定的显示器进行组态。程序复

制后必须重新编译。改变消息的文本和属性后，必须修改实例，文本只能写在一行中。

3．基于消息号的消息传送方式的分类

基于消息号的传送方式分为下面 3 类：

（1）与块有关的消息

与块有关的消息（block-specific messages）用于报告与程序同步的事件，用 WinCC 或 HMI 显示，可用于 S7-300/400，用系统功能（SFC）编程。

本章介绍的消息的传送和显示方法使用了 TIA（全集成自动化）功能。通过在 STEP 7 的项目中生成 PC 站或 HMI 站，将 STEP 7 和 WinCC，或者 STEP 7 和 WinCC flexible 的项目集成在一起。它们使用同一个 SQL 数据库，能轻松地实现消息的自动传送和显示。

（2）与符号有关的消息

与符号有关的消息用于报告与程序无关的事件，用 WinCC 显示，只能用于 S7-400。它用符号表组态，通过系统数据块（SDB）下载到 PLC，通过 AS-OS 链接传送到 WinCC。

（3）用户自定义的诊断消息

这种诊断消息是用户用 SFC 52“WR_USMSG”自定义的，它与程序同步，在编程设备的诊断缓冲区中显示，不能传送到操作面板。这种消息可用于 S7-300/400。

4．用于生成与块有关的消息的系统功能

下面的 SFC（系统功能）用于生成与块有关的消息：

- SFC 17 (ALARM_SQ)。
- SFC 18 (ALARM_S)。
- SFC 107 (ALARM_DQ)。
- SFC 108 (ALARM_D)。

消息的触发信号（SFC 的输入参数 SIG）的状态变化时生成一条消息。SFC 17 和 SFC 107 生成的消息需要确认，SFC 18 和 SFC 108 生成的消息不需要确认。

下面首先介绍 SFC 17 和 SFC 18 的使用方法，SFC 107 和 SFC 108 的使用方法将在后面介绍。某些低档的 HMI 和 CPU 不支持用上述 SFC 组态和自动传送消息。

8.1.2 硬件组态与程序设计

1．创建 STEP 7 项目和硬件组态

在 STEP 7 中创建一个名为“Alarm_S”的项目（本章的项目在随书光盘的文件夹“\Project \Alarm Display”中），CPU 为 CPU 315-2DP。项目和它所在的文件夹的名称不能使用汉字。

选中该站点，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

2．生成 HMI 站点

WinCC flexible是西门子人机界面的组态软件，作者编写的《西门子人机界面（触摸屏）组态与应用技术（第二版）》的随书光盘有该软件的中文 2007 版，不需要许可证就可以使用。

在安装软件时，应先安装STEP 7，然后安装WinCC，最后安装WinCC flexible。创建HMI站实际上就是创建集成在STEP 7 中的WinCC flexible项目，后者保存在STEP 7 项目的HmiEs文件夹中。

点击管理器左侧窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→

“SIMATIC HMI Station”命令，在出现的对话框中设置HMI的型号为TP 177B 6" color PN/DP，在STEP 7的项目中生成SIMATIC HMI站对象（见图8-1）。

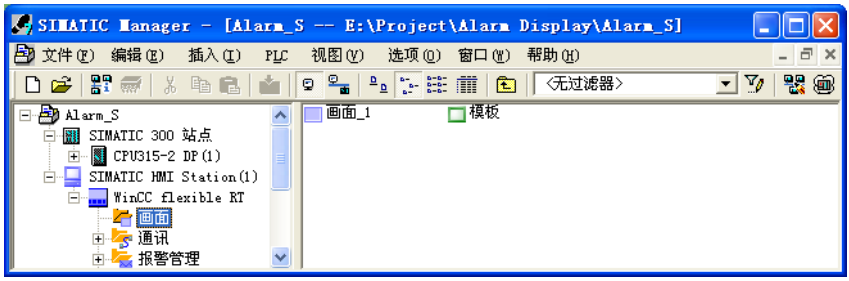


图8-1　SIMATIC管理器

3．在NetPro中建立连接

为了实现PLC和HMI设备之间的自动数据交换，需要建立它们之间的连接。点击STEP 7工具栏上的按钮，打开网络组态工具NetPro，显示出STEP 7项目中尚未与MPI网络连接的S7-300站和MPI站。用鼠标左键将两个站点中代表MPI接口的红色的小方框“拖放”到MPI网络上，即可将两个站连接起来（见图8-2）。点击工具栏上的按钮，编译和保存组态信息。

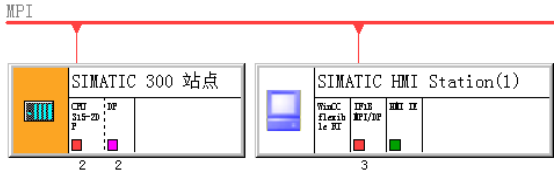


图8-2　在NetPro中建立连接

4．组态功能块的局部变量

在SIMATIC管理器中生成一个功能块FB 1，双击打开它。在FB 1的局部变量表中创建一个双字型（Dword）的输入（IN）参数EV_ID1（见图8-3）。用右键点击它，执行出现的快捷菜单中的命令“对象属性”。在打开的“变量属性”对话框的“属性”选项卡中，按图8-3的要求设置变量的属性。用同样的方法生成输入变量EV_ID2，它的属性与EV_ID1相同。

SFC 17、SFC18生成的消息可以带3个被称为相关值的变量。在FB 1的变量声明区中，创建3个双字型的静态变量（STAT）SD1～SD3，用来保存消息的相关值。在FB 1中输入这些静态变量的值，它们将作为消息的组成部分发送到操作面板。实际的程序可以用它们来保存需要随消息显示的动态变化的变量的值。

图8-3　FB 1的局部变量声明表

5．FB 1 的程序设计

下面是项目“Alarm_S”的 FB 1 中调用 SFC17、SFC18 生成消息的程序。用小数形式输入浮点数 80.2 后，STEP 7 将它自动地转换为指数形式 8.020000e+001。程序中的 DB 1 是 FB 1 的背景数据块。

程序段 1：

```
L       8.020000e+001
T       #SD1                            //预置浮点数
L       3.925000e+002
T       #SD2                            //预置浮点数
L       MW     50                       //在 OB35 中将 MW50 加 1
T       #SD3                            //预置十进制整数
CALL    "ALARM_SQ"                      //调用 SFC 17
  SIG           :=M10.0                 //消息的触发信号
  ID            :=W#16#EEEE             //消息的数据通道（常数）
  EV_ID         :=#EV_ID1               //消息号（双字）
  SD            :=P#DB1.DBX8.0 BYTE 12  //相关值的地址区，最大 12B
  RET_VAL       :=MW14                  //错误信息
```

程序段 2：

```
L       2354
T       #SD1                            //预置十进制整数
L       7.536000e+001
T       #SD2                            //预置浮点数
L       MW52                            //在 OB35 中将 MW52 加 2
T       #SD3                            //预置十进制整数
CALL    "ALARM_S"                       //调用 SFC 18
  SIG           :=M10.1                 //消息的触发信号
  ID            :=W#16#EEEE             //消息的数据通道
  EV_ID         :=#EV_ID2               //消息号（双字）
  SD            :=P#DB1.DBX8.0 BYTE 12  //相关值的地址区，最大 12B
  RET_VAL       :=MW16                  //错误信息
```

6．创建消息文本

用右键点击 SIMATIC 管理器左边窗口中的“块”，执行出现的快捷菜单中的“插入新对象”命令，创建 FB 1 的背景数据块 DB 1。

用右键点击 SIMATIC 管理器中的 FB 1，执行快捷菜单中的命令“特殊的对象属性”→“消息”。在打开的“消息组态”对话框中（见图 8-4），创建带有参数值的消息文本。点击“更多>>”按钮，该按钮上的字符变为“<<更少”，同时出现该按钮下面的“消息文本”和“信息文本”文本框。点击按钮“<<更少”，该按钮下面的文本框消失。

人机界面运行时点击报警视图左边的“信息文本”按钮（见图 8-12），将显示图 8-4 中组态的信息文本。

7．组态相关值

按下面的格式组态消息文本附带的变量值。

（1）相关值显示方式的设置

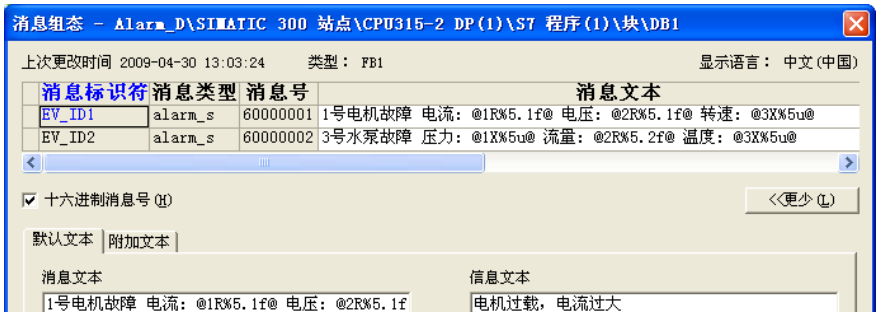


图 8-4　消息组态对话框

SFC 用参数 SD 指定的地址区来传递相关值。用下面的字符串来定义相关值在面板上的显示方式：

@<相关值的编号><元素类型><格式代码> @

其中的“@”是格式字符串的起始标志和结束标志。

（2）元素类型

B、Y、C、W、I、X、D 和 R 分别用来表示 BOOL（位）、BYTE（字节）、String（字符串）、WORD（字）、INT（整数）、DWORD（双字）、DINT（双整数）和 REAL（浮点数）。

（3）格式代码

格式代码见表 8-1，格式中的方括号只是用来作分隔符用，实际的格式代码没有方括号。

表 8-1　消息文本中的格式代码

格 式 代 码	描　　述	格 式 代 码	描　　述
%[i]X	i 位十六进制数	%[i].[y]f	小数点后 y 位，共 i 位的定点数
%[i]u	i 位无符号十进制数	%[i]S	i 位 ANSI 字符串
%[i]d	i 位有符号十进制数	%t#<文本库名称>	访问文本库
%[i]b	i 位二进制数		

例如，“@1R%5.1f@”表示 1 号浮点数相关值，最多显示 5 位，小数点后 1 位；“@3X%5u@”表示 3 号相关值，最多显示 5 位无符号双字（DWORD）十进制数。

组态信息时必须使用英文的标点符号，如果使用中文的标点符号，将会出错。组态结束后点击“确定”按钮关闭对话框。组态时各相关值与程序中对应的变量的数据类型应相同。

8．在 OB1 中调用 FB 1

消息用 CPU 内或项目内唯一的编号来识别。在创建背景数据块时，STEP 7 自动分配消息号给 FB 的形参。通过这个消息号，操作面板建立与消息文本的连接。可以选择给项目（面向项目）或给 CPU（面向 CPU）分配消息号。给 CPU 分配消息号的优点在于允许复制一个程序而无需修改消息号，但是修改了编号后需要重新编译。

在 OB1 中调用 FB 1 时，FB 1 的形参（EV_ID1 和 EV_ID2）和实参（消息号）是自动生成的。下面是 OB1 调用 FB 1 的程序：

```
CALL   FB      1 , DB1
  EV_ID1 :=DW#16#60000001
  EV_ID2 :=DW#16#60000002
```

9．SFC 107 与 SFC 108

与 SFC 17 和 18 相比，SFC 107“ALARM_DQ”和 SFC 108“Alarm_D”增加了下列功能：

1）可以用 SFC 105“READ_SI”来获取与占用的系统资源有关的信息。

2）可以用 SFC 106“DEL_SI”释放占用的系统资源。如果在程序更改过程中删除了包含 SFC 107 或 SFC 108 调用的 FB，当前占用的系统资源将保持占用状态，直到下次暖启动。更改程序并重新加载包含 SFC 107 或 SFC 108 调用的 FB 时，可能会发生 SFC 107 和 SFC108 不再生成消息的情况。

与 SFC 17 和 SFC 18 相比，SFC 107 和 SFC 108 多了一个输入参数 CMP_ID。这个参数用来确定消息所属的子系统的 ID（标识符）。在 FB 中调用 SFC 107 和 SFC 108 时，需要将 FB 对应的背景数据块的编号分配给 CMP_ID。

10．项目 Alarm_D

项目 Alarm_D 在 FB 1 中调用 SFC 108 和 SFC 107 来生成消息，程序的其他部分和组态方法与项目“Alarm_S”完全相同。下面是项目“Alarm_D”的 FB 1 调用 SFC 108 的程序：

程序段 1：

```
...
CALL   "ALARM_DQ"                    //调用 SFC 108
  SIG          :=M10.0               //消息触发信号
  ID           :=W#16#EEEE           //消息的数据通道
  EV_ID        :=#EV_ID1             //消息号（双字）
  CMP_ID       :=DW#16#1             //组件标识符（双字），不能为 0
  SD             :=P#DB1.DBX8.0 BYTE 12     //相关值，最大 12B
  RET_VAL      :=MW14                //错误信息
```

11．读取消息的确认状态

可以调用 SFC 19 来读取消息触发信号的状态和 SFC 17/107 的确认状态。 下面是在项目 ALARM_D 的 FB 1 中调用 SFC 19 的程序：

程序段 3：

```
CALL   "ALARM_SC"                    //调用SFC 19
  EV_ID        :=DW#16#60000001      //上次调用SFC 17/107的消息号
  RET_VAL      :=MW12                //出错信息
  STATE        :=M10.4               //触发消息的信号状态
  Q_STATE      :=M10.5               //为 1 时已确认
```

在系统运行时用M10.0 来触发电机故障消息，SFC 19 返回的触发信息（M10.4）的状态与M10.0 的状态相同。点击TP 177B报警视图右边  （确认）按钮 确认消息后，SFC 19 的输出参数Q_STATE（M10.5）变为 1 状态。触发信号M10.0 变为 0 状态后，M10.4 和M10.5 同时变为 0 状态。

8.1.3 用 HMI 显示消息的仿真实验

1．打开 WinCC flexible 的项目

打开图 8-1 中SIMATIC管理器左边窗口的HMI站点，选中“画面”，双击右边窗口的“画面_1”，打开 WinCC flexible（见图 8-5）。用画面下面的属性视图将画面_1 的背景色由灰色

改为白色。

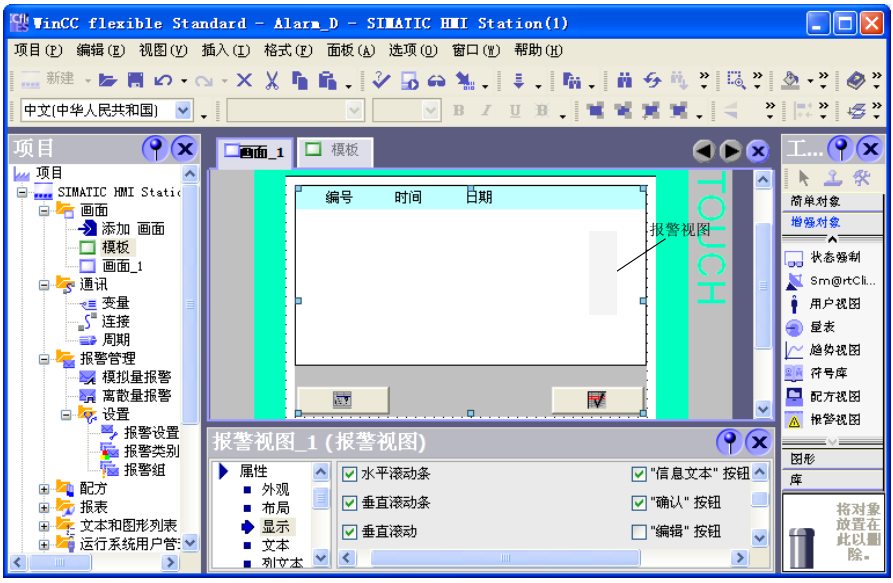


图 8-5　WinCC flexible

2．激活连接

双击图 8-5 左边窗口“通讯”文件夹中的“连接”图标，打开连接表（见图 8-6）。点击“激活的”列右边隐藏的按钮，使该列的参数由“关”变为“开”，即打开HMI与PLC的通信连接。

图 8-6　激活连接

3．设置报警

双击图 8-5 左边窗口“\报警管理\设置”文件夹中的“报警设置”图标，在“报警设置”视图中（见图 8-7），激活“S7 诊断报警”。

在“报警程序”表格的第一行，点击“ALARM_S”列右边的按钮，选中出现的对话框中“所有显示的类”复选框，点击（确认）按钮，关闭对话框。在“ALARM_S”列出现“0-16”。

4．组态报警视图

双击图 8-5 右边窗口中的“画面_1”，打开它以后，点击左边的工具箱中的“增强对象”（见图 8-5），将其中的“报警视图”拖放到画面上，用鼠标调节它的位置和大小。选中报警视图，下面是它的属性视图。

图 8-7　设置报警

在属性视图的“常规”选项卡（见图 8-8）中，用单选框选中“报警事件”。在“报警类别”列表中，选中“S7 报警”。必须按上述步骤进行操作，在报警视图中才能看到“S7 报警”复选框。

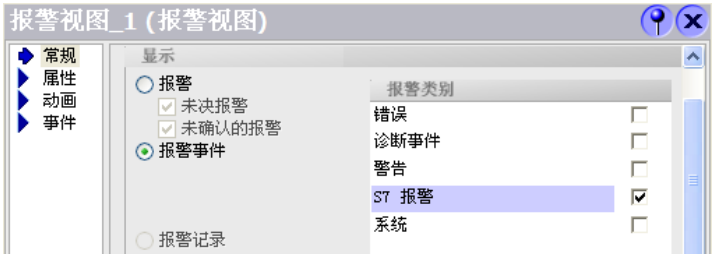


图 8-8　组态报警视图

选中图 8-5 报警视图的属性视图的“属性”文件夹中的“显示”，用复选框选中“确定”按钮和“信息文本”按钮。“确定”按钮在报警视图的右边。

选中属性视图的“属性”中的“外观”（见图 8-5），设置报警视图和表头的背景色。

选中属性视图的“属性”中的“文本”，设置报警视图中文字的大小为 9 个像素点。

双击图 8-5 左边窗口的“模板”，打开模板画面。点击右边“工具箱”中的“增强对象”，将其中的“报警窗口”和“报警指示器”拖放到模板画面中。有报警消息时，不管当前正在显示的是什么画面，“模板”中组态的报警窗口和报警指示器都会在当前画面出现（见图 8-11）。

报警窗口的组态方法与报警视图差不多。在“常规”选项卡中选中“报警”，而不是“报警事件”，报警类别为“S7 报警”（见图 8-8），在消息的触发信号（M10.0 和M10.1）为 1 时显示消息。消息被确认，且触发信号消失后，报警窗口和其中的消息亦随之消失，即报警窗口不能保留和显示历史消息。

5．启动 PLCSIM

S7-PLCSIM 是 S7-300/400 功能强大、使用方便的仿真软件。可以用它代替 PLC 硬件来调试用户程序。安装好 STEP 7 后，需要安装 S7-PLCSIM，后者自动嵌入 STEP 7。

点击 STEP 7 的 SIMATIC 管理器工具栏上的按钮，出现“Open Project”对话框（见图 8-9）。点击“OK”按钮，双击出现的对话框中 CPU 的 MPI 地址（MPI（1）adr: 2），打开 S7-PLCSIM（见图 8-10）。与此同时，自动建立了 STEP 7 与仿真 CPU 的 MPI 连接。

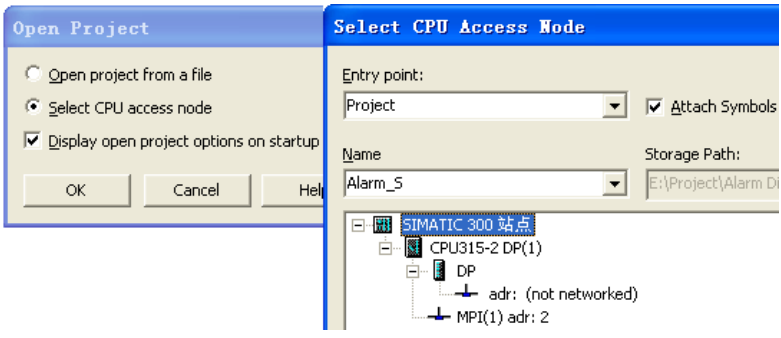


图 8-9　打开 PLCSIM

可以用鼠标调节 S7-PLCSIM 窗口的位置和大小。点击 CPU 视图对象中的“STOP”、

“RUN”或“RUN-P”小方框，可以令仿真 PLC 处于相应的运行模式。点击“MRES”按钮，可以清除仿真 PLC 中已下载的程序。

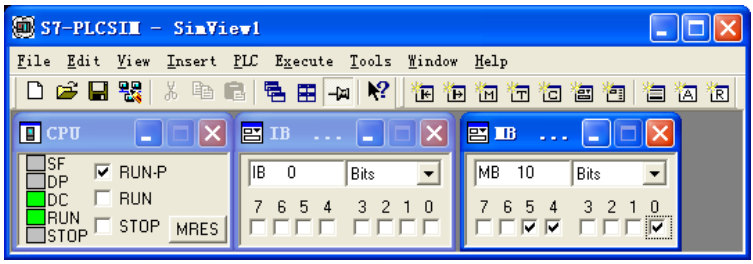


图 8-10　PLCSIM

点击 S7-PLCSIM 工具栏上的、、按钮，生成 IB0、QB0 和 MB0 视图对象。将视图对象中的 MB0 改为 MB10，按〈Enter〉键确认。

选中 SIMATIC 管理器左边窗口中的“块”对象，点击工具栏上的（下载）按钮，将所有的块和系统数据下载到仿真 PLC。点击 CPU 视图对象中的小方框，将 CPU 切换到 RUN 或 RUN-P 模式。在 RUN-P 模式可以下载修改后的程序块和系统数据。

6．启动 WinCC flexible 的运行系统

点击WinCC flexible工具栏上的按钮，启动WinCC flexible的运行系统，出现触摸屏的模拟运行画面 （见图 8-12），此时报警视图中还没有报警信息 。

用鼠标点击 PLCSIM 的 MB10 视图对象最低位对应的小方框(M10.0)，方框内出现“√”（见图 8-10），CPU 的 M10.0 变为 1 状态。它触发 FB 1 调用的 SFC 17，OB1 中编号为 DW#16#60000001 的消息号被发送到 TP 177B，模拟运行画面出现报警窗口和闪动的报警指示器（见图 8-11），报警窗口中是 1 号电机故障的消息。M10.4（SFC 19 的输出变量 STATE，触发消息的信号 M10.0 的状态）同时变为为 1 状态。

图 8-11　报警窗口和报警指示器

点击报警视图右边的（确认）按钮，消息被确认，报警指示器停止闪动。M10.5（SFC 19 的输出变量 Q_STATE，确认信号）变为 1 状态。

点击 PLCSIM 中 M10.0 对应的小方框，其中的“√”消失，M10.0 变为 0 状态，M10.4 和 M10.5 同时变为 0 状态。报警窗口消失，此时图 8-12 的报警视图只有下面 3 条消息，最先出现的消息在最下面。3 条消息分别表示消息“1 号电机故障”出现、被确认和故障消失。

在每 100ms 调用一次的 OB35 中，将 MW50（转速值）加 1，MW52（温度值）加 2。从图 8-12 可以看到这两个变量动态变化的情况。

用 PLCSIM 使 M10.1 变为 1 状态，OB1 中的消息号 DW#16#60000002 被发送到 TP 177B，出现图 8-12 所示从上往下的第 2 条消息的报警窗口。该消息不需要确认，因此报警指示器没有闪动。用 PLCSIM 将 M10.1 变为 0 状态，报警窗口和报警指示器消失，画面上的报警视图

出现图 8-12 中上面两条消息。

选中报警视图中水泵的故障消息，点击左边的“信息文本”按钮，出现信息文本对话框（见图 8-12 中的小图）。信息文本是在组态消息时输入的（见图 8-4）。

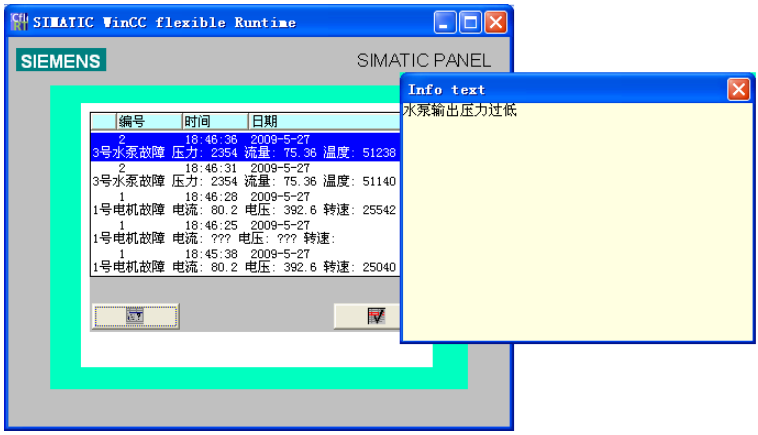


图 8-12　WinCC flexible 的模拟运行画面

7．用实际的触摸屏显示消息的操作

可以用 CP 5611 等 CP 卡、S7-300/400 的 PC/MPI 适配器、USB/MPI 适配器，和 S7-200 的 PC/PPI 适配器，将 WinCC flexible 的组态信息下载到 HMI。有的 HMI 设备有以太网接口和 USB 接口，可以用于下载。

下面以 USB/MPI 适配器为例，介绍操作的步骤：

1）安装适配器的驱动程序。

2）打开“设置 PC/PG 接口”对话框，设置适配器使用 MPI，波特率为 187.5kbit/s。

3）用 HMI 的控制面板设置 HMI 的站地址、通信协议和波特率，具体的操作见 HMI 的操作手册。

4）HMI 上电后，点击出现的 Loader 对话框中的 Transfer（传送）按钮，显示“Connecting to host…”（正在与上位机连接），HMI 处于等待传送的状态。

5）点击 WinCC flexible 工具栏上的按钮，在打开的对话框中设置传送的参数。点击“传送”按钮，开始下载。

8．HMI 与 PLC 的通信

HMI 一般用 RS-485/422 接口与 CPU 通信，MP 系列人机界面可以通过 RS-485 和 PROFINET（以太网）连接 PLC。根据面板背面的图形给出的 DIP 开关位置，来设置使用 RS-485 还是 RS-422 接口。

根据 WinCC flexible 的连接表中的参数，用 HMI 的“控制面板”设置它的站地址、通信协议和波特率。具体的方法请查阅人机界面的操作手册。

8.1.4　用户自定义的诊断消息

1．创建 STEP 7 项目和组态硬件

在 STEP 7 中创建一个名为“SFC_52”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config，将电源模块和信号模块插入机架。

双击机架中“DP”所在的行，点击出现的 DP 属性对话框的“属性”按钮，在出现的 PROFIBUS 接口属性对话框中，点击“新建”按钮，生成一条新的 PROFIBUS 子网络，将 CPU 连接到 DP 网络上。

将右边硬件目录窗口的“\PROFIBUS DP\ET 200B”文件夹中的“B-16DO”拖放到 DP 网络上，在自动打开的“属性 –PROFIBUS 接口”对话框的“参数”选项卡中，设置从站的地址为 4。点击工具栏上的按钮，编译和保存组态信息。

2．组态用户自定义的诊断消息

选中 SIMATIC 管理器左边窗口的“S7 程序”，执行菜单命令“编辑”→“特殊对象属性”→“消息”，打开“消息组态”对话框（见图 8-13）。

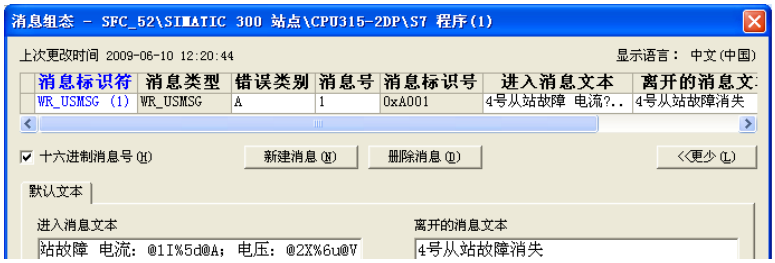


图 8-13　消息组态对话框

点击“新建消息”按钮，生成一条新的消息。消息从左边开始的前 5 项属性是自动生成的。可以直接在表格中输入消息文本。如果要输入字数较多的文本，点击“更多>>”按钮，在下面出现的文本框内输入消息文本。“更多>>”按钮上的字符变为“<<更少”，点击它将关闭下面的文本框。

可以将 SFC 52 的参数 INFO1 和 INFO2 指定的附加信息作为消息文本中的关联值，附加信息的显示格式见 8.1.2 节。图 8-13 中的“@1I%5d@”是对应于 INFO1 的 5 位十进制整数附加信息，“@2X%6u@”是对应于 INFO2 的 6 位无符号十进制双字附加信息。

“错误类别”可以选择 A 或 B，它只是用来将消息分组。每个错误类别可以生成 255 个消息（1～255 号消息）。

下面的 OB1 中的程序调用 SFC 52“WR_USMSG”来创建用户自定义的诊断消息。分别在 M10.0 的上升沿和下降沿发送图 8-13 中的“进入消息文本”和“离开的消息文本”。

```
        A     M     10.0
        FP    M     11.0
        JCN   m001                  //不是 M0.0 的上升沿时跳转
        CALL  "WR_USMSG"            //调用 SFC 52
          SEND     :=TRUE           //为 1 时发送自定义诊断消息
          EVENTN   :=W#16#A101      //事件标识符，进入事件
          INFO1    :=MW12           //1 个字长的附加信息
          INFO2    :=MD14           //两个字长的附加信息
          RET_VAL  :=MW18           //错误信息
m001:   NOP   0
        A     M     10.0
        FN    M     11.1
        JCN   m002                  //不是 M0.0 的下降沿时跳转
```

```
        CALL    "WR_USMSG"          //调用 SFC 52
          SEND      :=TRUE          //为 1 时发送自定义诊断消息
          EVENTN    :=W#16#A001     //事件标识符，离开事件
          INFO1       :=MW22          //一个字长的附加信息
          INFO2     :=MD24          //两个字长的附加信息
          RET_VAL   :=MW28          //错误信息
m002:     NOP     0
```

参数 EVENTN 的格式为 W#16#Axxx，首位可以取 A 或 B（与组态有关），第 2 位用 1 和 0 来表示进入和离开状态的事件，最低两位是消息的编号。可以用图 8-13 左边的复选框选择用十进制或十六进制格式显示消息号。

3．仿真实验

点击 STEP 7 的 SIMATIC 管理器工具栏上的按钮，打开 PLCSIM，生成 MB10 视图对象。选中 SIMATIC 管理器左边窗口中的“块”对象，点击工具栏上的（下载）按钮，将所有的块和系统数据下载到仿真 PLC。点击 CPU 视图对象中的小方框，将 CPU 切换到 RUN 或 RUN-P 模式。

用变量表将常数写入存放消息的附加信息的 MW12 和 MD14（见图 8-14）。

用鼠标点击 PLCSIM 中的 M10.0 对应的小方框，方框内出现“√”，M10.0 变为 1 状态。在 M10.0 的上升沿调用 SFC 52，发送图 8-13 中定义的带有两个附加信息的进入事件消息文本“4 号从站故障”。

再次点击 PLCSIM 中 M10.0 对应的小方框，其中的“√”消失，M10.0 变为 0 状态，在 M10.0 的下降沿调用 SFC 52，发送自定义的离开事件消息文本“4 号从站故障消失”。

执行菜单命令“PLC”→“模块信息”，在打开的“模块信息”对话框的“诊断缓冲区”选项卡中，可以看到两条自定义的诊断消息（见图 8-14）。

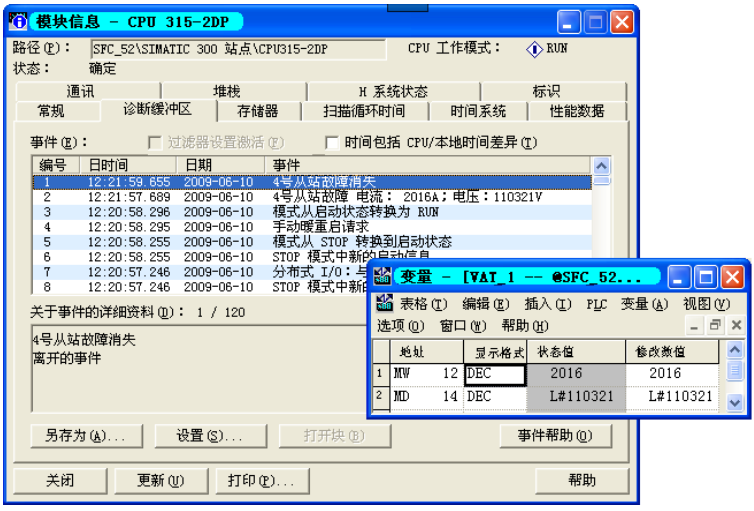


图 8-14　CPU 模块信息的诊断缓冲区

需要点击“更新”按钮，才能看到打开模块信息对话框以后触发的自定义诊断消息。

8.1.5　用软件 S7-PDIAG 组态过程诊断

软件 S7-PDIAG 在内部调用系统功能 SFC 17/18、SFC107/108 来实现消息的传送。与直

接调用 SFC 来产生消息相比，S7-PDIAG 的功能更强，组态和编程的工作量要小一些。

1．S7-PDIAG 的监控功能

S7-PDIAG 有地址监控、全局监控和运动监控 3 种监控功能。

（1）地址监控

地址监控功能监控位信号的状态变化，可以设置为边沿信号或电平信号（见图 8-15）。

（2）全局监控

全局监控（Global monitoring）又称为常规监控（General monitoring），它监控一个表达式，如果表达式的运算结果为 1，则触发消息。

可以定义事件触发的条件，触发条件必须用 S7-PDIAG 语言编写，S7-PDIAG 语言包含下列基本指令：AND（与）、OR（或）、NOT（取反）、XOR（异或）、EP（存入上升沿结果）、EN（存入下降沿结果）、ONDT（ON 延时）、SRT（置位复位延时）和括号。各指令之间有规定的优先级，详细的情况见 S7-PDIAG 的在线帮助。

（3）运动监控

运动监控对用户编写的定位过程进行监控，运动监控分为 4 种监控方式：

1）执行监控（Action Monitoring）：执行启动命令后，在设定的时间内没有到达定位的位置，将触发报警消息。

2）启动监控（Startup Monitoring）：启动命令发出后，在设定的时间内没有动作，将触发报警消息。

3）反应监控（Reaction Monitoring）：定位任务完成，但是没有到达要求的位置，在设定的延时时间后触发报警消息。

4）互锁监控（Interlock Monitoring）：检测在控制命令触发后，互锁条件是否满足，如果不满足则触发报警消息。

2．创建 STEP 7 项目和组态硬件

随书光盘中的项目 S7PDIAG 是在第 7 章的项目 SFC_13 的基础上创建的。将后者另存为项目 S7PDIAG，CPU 为 CPU 313-2DP，4 号从站和 5 号从站分别是 ET 200B-16DO 和 ET 200B-16DI，7 号从站是 ET 200M。在生成该项目之前，首先应安装随书光盘中的软件 S7-PDiag。

点击管理器左边窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→“SIMATIC HMI Station”命令，在出现的对话框中设置 HMI 的型号为 TP 177B 6"color PN/DP，在 STEP 7 的项目中生成 HMI 站对象。

点击 STEP 7 工具栏上的按钮，打开网络组态工具 NetPro。用鼠标左键将两个站点中代表 MPI 接口的红色小方框“拖放”到 MPI 网络上，将两个站连接到网络上。

点击工具栏上的按钮，编译和保存组态信息。

3．组态地址监控

7 号、4 号和 5 号从站有故障时，OB86 将 M20.0、M30.0 和 M40.0 分别置位为 1（见 7.1.2 节中的程序）。从站故障消失时，OB86 分别将它们复位为 0。

用鼠标右键点击 OB86 程序中的 M20.0，执行快捷菜单中的命令“特殊对象属性”→“监视”，选中打开的“过程监控”对话框的“模板”列表中的“地址监控”（见图 8-15）。点击“新建”按钮，在“地址监控”对话框的“常规”选项卡中，将监控的名称修改为“M20.0 上升沿地址监控”。在“定义”选项卡的“监控定义”区，用单选框选中“上升沿”，即在 M20.0

的上升沿发送消息。

如果激活了“延迟”功能，需要设置延迟时间，默认的时间单位为 ms。如果监控信号变化，将在设置的延迟时间后触发消息。

图 8-15　过程监控对话框

可以在图 8-15 左图的“消息”区直接设置信号触发的消息的文本和消息的优先级。点击“消息”区中的“组态”按钮，在打开的对话框中（见图 8-16），可以组态消息更多的属性。

图 8-16　消息组态对话框

两次点击“确定”按钮，返回“过程监控”对话框。再次点击“新建”按钮，生成名为“M20.0 下降沿地址监控”的地址监控。其消息文本为 7 号从站故障消失，信息文本为“ET 200M”。

用同样的方法生成下面 4 条消息，信息文本给出了 ET 200 的型号。

1）在 M30.0 的上升沿发送消息文本“4 号站故障”。

2）在 M30.0 的下降沿发送消息文本“4 号站故障消失”。

3）在 M40.0 的上升沿发送消息文本“5 号站故障”。

4）在 M40.0 的下降沿发送消息文本“5 号站故障消失”。

4．组态过程诊断

选中 SIMATIC 管理器左边窗口中的“块”，执行菜单命令“选项”→“组态过程诊断”，打开 S7-PDIAG 视图（见图 8-17）。

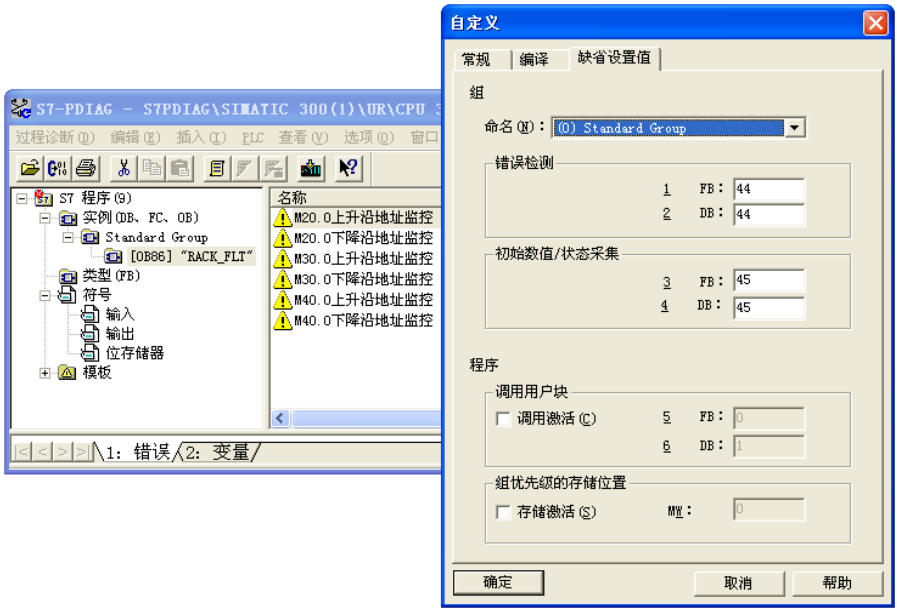


图 8-17　S7-PDIAG

点击（编译）按钮，出现“请检查编译设置”对话框，点击“确定”按钮，出现“自定义”对话框。采用默认的设置，点击“确定”按钮，开始编译。编译成功完成后，图 8-17 右边窗口的“S7 程序”图标上面的红色指示灯符号消失。

关闭 S7-PDIAG 视图，在 SIMATIC 管理器的“块”文件夹中可以看到自动生成的 FB、DB 和 SFC。

在 OB1 中编写下面调用 FB 44 的语句：

```
CALL   FB     44 , DB44
  PDIAGZyklus  :=#OB1_SCAN_1
```

5．WinCC flexible 的组态

打开 SIMATIC 管理器左边窗口的 HMI 站点（见图 8-18），选中“画面”图标，双击右边窗口的“画面_1”，打开 WinCC flexible。

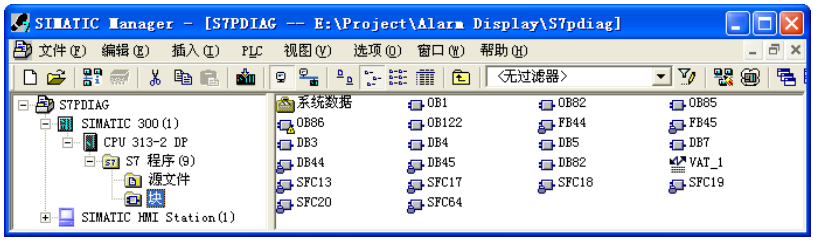


图 8-18　SIMATIC 管理器

按下面的步骤进行操作，详细的操作方法见 8.1.3 节。

1）在连接表中打开自动生成的 HMI 与 PLC 的通信连接（见图 8-6）。

2）在“报警设置”视图中激活“S7 诊断报警”（见图 8-7），在“报警程序”表的“ALARM_S”列选中所有显示的类。

3）组态报警视图，选中“报警类别”列表中的“S7 报警”（见图 8-8）。

6．对过程诊断的仿真实验

点击 SIMATIC 管理器工具栏上的按钮，打开 S7-PLCSIM（见图 8-10）。

选中 SIMATIC 管理器左边窗口的“块”对象，点击工具栏上的（下载）按钮，将所有的块和系统数据下载到仿真 PLC。然后将 CPU 切换到 RUN 或 RUN-P 模式。

点击 WinCC flexible 工具栏上的按钮（见图 8-5），启动 WinCC flexible 的运行系统，出现模拟的 HMI 画面（见图 8-19）。

在 PLCSIM 中生成 MB20、MB30 和 MB40 视图对象。点击 M30.0，将它置为 1 状态，出现消息“4 号站故障”。再次点击 M20.0，将它置为 0 状态，出现消息“4 号站故障消失”。用同样的方法，通过对 M20.0 和 M40.0 的操作，可以触发 7 号从站和 5 号从站的故障消息。

7．用硬件 PLC 触发消息的实验

在 WinCC flexible 中做好上面的准备工作后，用 DP 电缆连接好 CPU、3 个 DP 从站和 CP 5613 的 DP 接口。PLC 和从站的电源通电后，将程序和组态信息下载到 CPU，将 CPU 切换到 RUN 模式。

点击 WinCC flexible 工具栏上的按钮，启动 WinCC flexible 的运行系统，出现模拟运行画面。这种仿真采用真实的 PLC，只是对 HMI 仿真，称为在线仿真。

断开 4 号从站的电源，与 PLCSIM 仿真相同，出现图 8-19 中的 1 号消息“4 号站故障”。接通 4 号从站的电源，出现 2 号消息“4 号站故障消失”。

图 8-19　HMI 显示的消息

8.2　用报告系统错误功能组态消息

8.2.1　组态报告系统错误功能

STEP 7 的“报告系统错误”功能只需作简单的组态，就可以自动生成用于诊断和发送消息的 OB、FB、DB 和 SFC，以及各机架、从站和模块对应的故障消息，故障的文本被自动传送到 HMI 或 WinCC 的项目中。将生成的块下载到 CPU，运行时如果出现故障，CPU 将触发对应的消息，用 HMI 设备或 WinCC 显示出故障信息。

1．创建 STEP 7 项目和组态硬件

在 STEP 7 中创建一个名为“ReportEr”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config，将电源模块和信号模块插入机架。

双击机架中“DP”所在的行，点击出现的 DP 属性对话框的“属性”按钮，在出现的 PROFIBUS 接口属性对话框中，点击“新建”按钮，生成一条新的 PROFIBUS 子网络。点击“确定”按钮，返回 DP 属性对话框，将 CPU 连接到 DP 网络上。

将右边硬件目录窗口的“\PROFIBUS DP\ET 200B”文件夹中的“B-16DO”和“B-16DI”拖放到 DP 网络上，在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，设置从站的地址为 4 和 5。

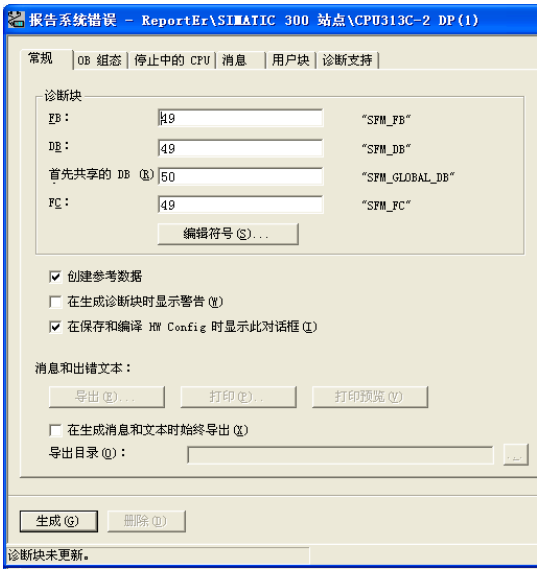


图 8-20　报告系统错误对话框

将“\PROFIBUS DP\ET 200M”文件夹中的“IM 153-1”拖放到 DP 网络上，设置从站的地址为 7。将 DI、DO 和 AO 模块拖放到 HW Config 下面窗口的“插槽”内，组态 AO 模块的诊断功能。

点击管理器左侧窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→“SIMATIC HMI Station”命令，在出现的对话框中设置 HMI 的型号为 TP 177B 6" color PN/DP，在 STEP 7 的项目中生成 HMI 站对象。

点击 STEP 7 工具栏上的按钮，打开网络组态工具 NetPro。用鼠标左键将 CPU 和 HMI 站点中代表 MPI 接口的红色的小方框“拖放”到 MPI 网络，两个站被连接到网络上。

点击工具栏上的按钮，编译和保存组态信息。

2．组态报告系统错误功能

选中 HW Config 中的 CPU，执行菜单命令“选项”→“报告系统错误”。在打开的对话框中，“常规”选项卡给出了要生成的诊断块和数据块（见图 8-20）。

图 8-21 将 3 张选项卡的图叠放在一起。在“OB 组态”选项卡中组态是否要生成 S7 程序中尚不存在的错误处理 OB，以及在哪些 OB 中调用报告系统错误的 FB 49。

通常情况下，S7-300 访问 DP 从站出错时不会调用 OB85，但是在组态报告系统错误功能时，如果没有生成和下载 OB85，在运行时出现触发 OB85 的故障，CPU 将会停机。

如果激活了“停止中的 CPU”选项卡的某个选项，在出现故障时 CPU 将进入 STOP 模式。默认的设置是不激活此选项卡的所有选项。

在“消息”选项卡，可以用按钮增添消息文本的内容，还可以设置与消息有关的其他参数。详细的说明可以参阅对话框的在线帮助。

在“用户块”选项卡中，可以指定诊断块是否调用独立的用户块，可以设置用户块的接口参数。本例程在组态报告系统错误时基本上采用默认的设置。

点击对话框中的“生成”按钮（见图 8-20），出现的对话框报告了生成或修改哪些块。

在 SIMATIC 管理器中，可以看到生成的块（见图 8-22）。FB 49 调用 SFC 17 来生成消息。打开 OB1 和“OB 组态”选项卡中指定的 OB，可以看到自动生成的调用 FB 49 的程序。双击

管理器中的 DB 49，将会出现图 8-20 中的“报告系统错误”对话框。

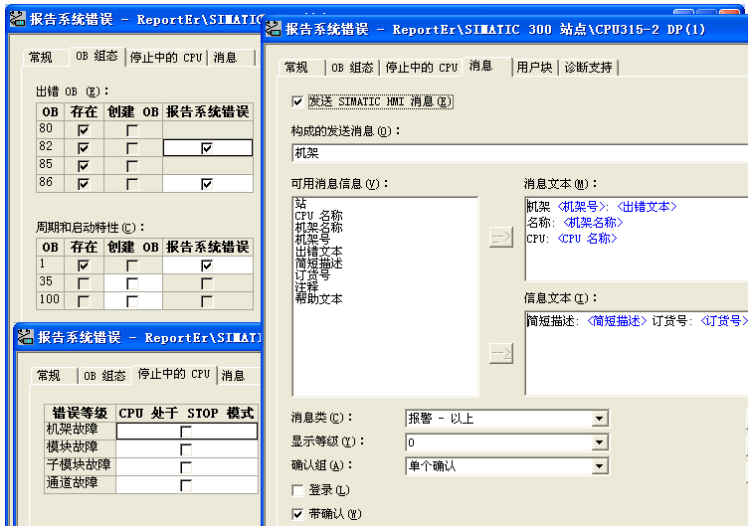


图 8-21　报告系统错误对话框

如果组态了分布式 I/O 设备，为了判断分布式 I/O 的错误，生成的 FB 49 将自动调用 SFC 13 来读取 DP 从站的诊断数据。应在 OB1 中，或者在间隔较短的时间中断 OB 中（例如 OB35），或者在 OB100 中调用生成的 FB 49。

图 8-22　SIMATIC 管理器

用鼠标右键点击 FB 49，执行快捷菜单命令“特殊的对象属性”→“消息”，在“消息组态”对话框（见图 8-23）中，可以看到 STEP 7 自动生成的大量的消息。出现硬件故障时，CPU 将会把对应的消息发送给 HMI 或 WinCC。

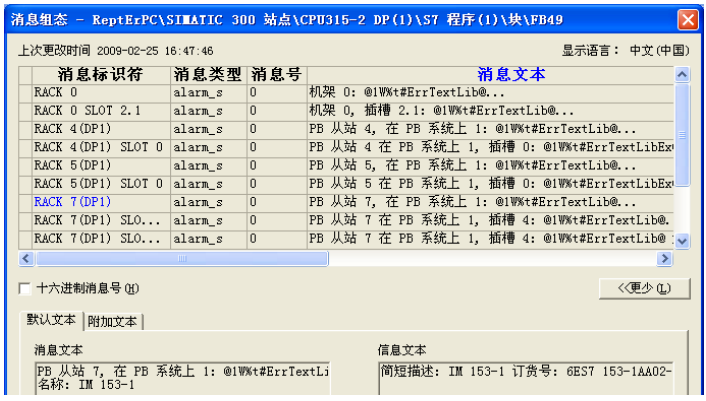


图 8-23　消息组态对话框

8.2.2 用 HMI 显示消息的实验

1．在 WinCC flexible 中组态 HMI

打开 SIMATIC 管理器左边窗口的 HMI 站点（见图 8-22），选中“画面”图标，双击右边窗口的“画面_1”，打开 WinCC flexible。

按下面的步骤进行操作，详细的操作方法见 8.1.3 节。

1）在连接表中激活自动生成的 HMI 与 PLC 的通信连接（见图 8-6）。

2）在“报警设置”视图中激活“S7 诊断报警”（见图 8-7），点击“ALARM_S”列右边的按钮，选中出现的对话框中的“所有显示的类”复选框。

3）组态报警视图，选中“报警类别”列表中的“S7 报警”（见图 8-8）。

2．仿真的准备工作

点击 STEP 7 的 SIMATIC 管理器工具栏上的按钮，打开 S7-PLCSIM（见图 8-24）。

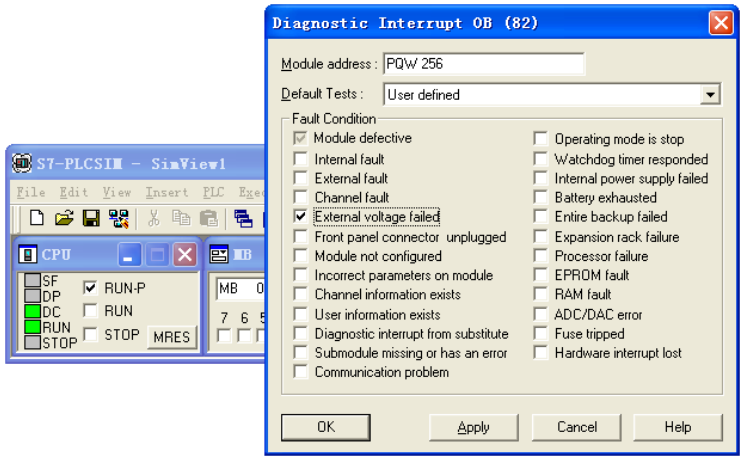


图 8-24　对 OB82 仿真的对话框

选中 SIMATIC 管理器左边窗口的“块”对象，点击工具栏上的（下载）按钮，将所有的块和系统数据下载到仿真 PLC。然后将 CPU 切换到 RUN 或 RUN-P 模式。

点击 WinCC flexible 工具栏上的按钮，启动 WinCC flexible 的运行系统，出现模拟的 HMI 画面（见图 8-25）。

3．对诊断中断仿真

PLCSIM 有较强的对硬件故障的仿真功能。执行 PLCSIM 的菜单命令“Execute”（执行）→“Trigger Error OB”（触发错误 OB）→“Diagnostic Interrupt（OB82）”（诊断中断 OB82），打开 OB82 的仿真对话框（见图 8-24）。

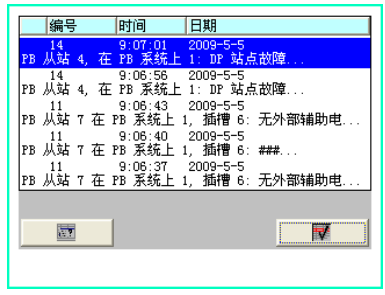


图 8-25　HMI 显示的消息

在“Module address”（模块地址）文本框中输入 7 号从站的 2AO 模块的模拟量输出通道 0 的地址 PQW256。如果输入的地址对应的通道没有组态诊断功能，将会出现一个对话框，显示“Invalid module address”（无效的地址），要求输入正确的值。

OB82 的仿真对话框的“Default Tests”（故障测试）选择框默认的选项是“User defined”

（用户定义），选中它后可以用下面的复选框设置各种故障。

用复选框选中“External voltage failed”（外部电压故障），点击“Apply”（应用）按钮，模拟运行画面出现第一条消息，即图 8-25 最下面的消息。点击“OK”按钮的作用与点击“Apply”按钮的作用相同，但是同时会关闭 OB82 诊断中断对话框。激活面板上的某条消息，将会在消息的下面看到消息中的信息文本。

点击面板右边的（确认）按钮，出现以“###...”结束的确认消息。有时可能需要点击两次，第一次激活模拟的面板窗口，第二次点击才真正起作用。

再次点击图 8-24 中的“External voltage failed”，复选框中的“√”消失，表示故障消失。点击“Apply”按钮，面板中又出现一次“无外部辅助电压”消息。打开图 8-24 中 OB82 的仿真对话框后按〈F1〉键，在出现的在线帮助中，可以看到 OB82 的局部变量（见表 6-7）与图中各种故障之间的关系。

用 PLCSIM 对故障仿真时，除了用仿真面板或 WinCC 来显示消息外，还可以用 PLCSIM 的 CPU 视图对象上的 LED、STEP 7 的快速视图、诊断视图、CPU，以及其他模块的模块信息、变量表等工具来诊断系统（见 6.2 节）。上述诊断功能得到的信息与实际的硬件系统提供的信息基本上相同。低档的 CPU 没有 DP 从站故障仿真功能。

点击“Cancel”（取消）按钮，关闭图 8-24 中的 OB82 诊断中断对话框。

4．对 DP 从站故障的仿真

执行 PLCSIM 的菜单命令“Execute”→“Trigger Error OB”→“Rack failure（OB86）”（机架与 DP 从站故障 OB86），打开 OB86 的仿真对话框（见图 8-26）。

“Expansion Rack Failure”（扩展机架故障）选项卡用来对扩展机架的故障仿真。在 DP Failure（DP 故障）选项卡，可以看到用绿色显示的已组态的 4、5、7 号从站。

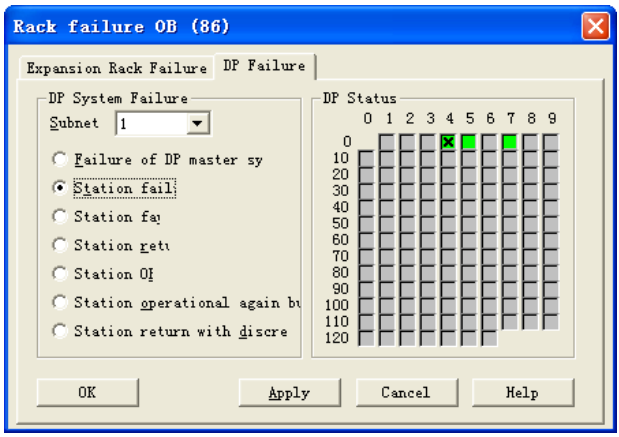


图 8-26　对 OB86 仿真的对话框

选中 4 号从站对应的小方框，它的中间出现×（见图 8-26）。用单选框选中“Station failure”（站故障），点击“Apply”按钮，面板上出现 4 号从站站点故障的消息。4 号从站对应的小方框中的“×”消失，该方框变为红色，表示有故障。

点击面板右边的（确认）按钮，出现以“###...”结束的确认消息。

用单选框选中“Station OK”（站正常），点击“Apply”按钮，故障消失，面板上又出现一次 4 号从站站点故障的消息。4 号从站对应的小方框变为绿色。

5．用 PLC 的硬件触发消息

CPU 通过 MPI 或 DP 网络与计算机通信，用 PLC 的硬件取代 PLCSIM 对 CPU 和 DP 从站的仿真。与上述的例子相同，HMI 仍然用 WinCC flexible 的运行系统来模拟。

将程序和组态信息下载到 CPU，CPU 和 DP 从站切换到运行模式。分别断开各个从站的电源，然后又接通它，模拟的 HMI 画面上出现对应的消息。断开 7 号从站（ET 200M）6 号槽的 2AO 模块 0 号通道的电流输出电路，然后又接通它。在仿真面板上出现“模拟输出断线”的消息（见图 8-27）。

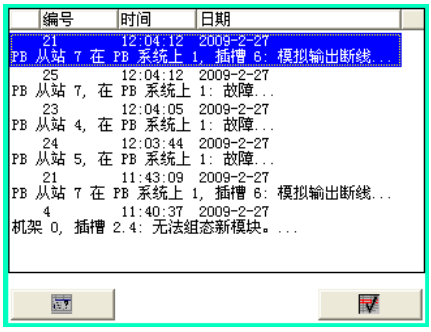


图 8-27　硬件 PLC 的诊断消息

图 8-28　硬件 PLC 的诊断消息

点击某条消息，该消息由两行变为多行（见图 8-28），可以看到详细的信息。点击右边的“确认”按钮，显示的消息缩为两行，可以看到条数更多的消息。

如果在故障出现后已确认了消息，故障消失时面板上对应的消息同时消失。如果在故障消失后确认对应的消息，该消息也会在面板上消失。

这种方法的组态过程非常简单，程序块和程序都是自动生成的，几乎没有什么编程的工作量，生成的消息几乎覆盖了所有的硬件故障和组态的诊断事件。用 SFC 13 读取故障信息、分析故障信息，用 SFC 17 生成消息、发送消息和将消息发送到 HMI 都是自动完成的。因此这是一种理想、实用的故障诊断和显示的方法。

8.2.3　故障诊断的必要条件

本书第 6～8 章介绍了实现故障诊断和故障显示的方法，这些方法是建立在控制系统的 STEP 7 项目文件的基础上的，它是进行故障诊断的必要条件。如果有 STEP 7 的项目文件，最简单直观的故障诊断方法是用 STEP 7 进行在线故障诊断。在 STEP 7 项目的基础上，可以用“报告系统错误”功能来诊断和显示故障，这种诊断方法的功能强，实现简单方便。

如果下载到 CPU 的 STEP 7 项目文件没有加密，可以按下面的步骤上传组态信息和用户程序：

1）在 STEP 7 中新建一个空的项目。

2)用计算机的 CP 卡或 PC/MPI、USB/MPI 适配器和通信电缆连接好计算机和各站点 CPU 的 MPI 接口。

3）在 SIMATIC 管理器执行菜单命令“选项”→“设置 PG/PC 接口”，设置计算机的通信接口的参数。

4）在 SIMATIC 管理器中执行菜单命令“PLC”→“将站点上传到 PG”，点击出现的“选择节点地址”对话框（见图 8-29）中的“显示”按钮。几秒钟后“可访问的节点”列表出现 MPI 网络上的节点，“显示”按钮上的字符变为“更新”。选中“可访问的节点”列表的某个 CPU，被选中的 CPU 出现在上面的表格中。

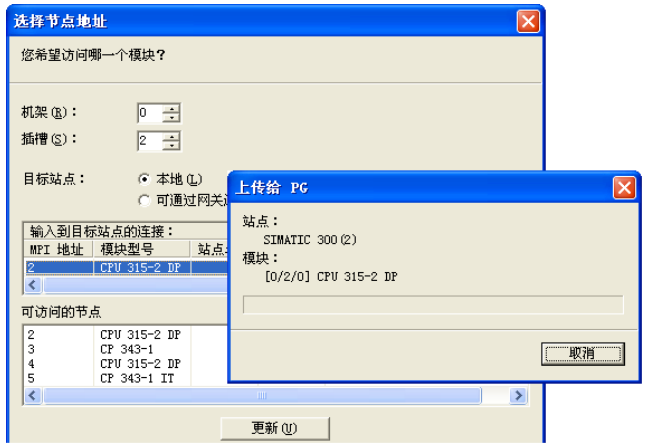


图 8-29　上传站点给 PG

5）点击“确定”按钮，开始上传组态信息和用户程序。上传结束后，在 SIMATIC 管理器左边窗口可以看到上传的站点中的程序块和系统数据。选中上传的站点，双击右边窗口中的“硬件”，打开 HW Config，可以看到该站点的硬件结构、网络和网络上的 DP 从站或 PROFINET IO 设备。

6）重复第 4 步和第 5 步，上传系统中所有站点的信息。

上传结束后，在 NetPro 中可以看到上传的网络结构和连接表中的连接。

实验时发现如果使用以太网 CP 的以太网接口来上传，图 8-29 中的“可访问的节点”列表只有 CP 的 IP 地址或 MAC 地址。如果选中某个 CP 后启动上传，因为必须通过 CPU 上传，所以不能用以太网 CP 来将站点上传到 PG。

下载用户程序时没有下载符号表和程序中的注释，所以上传的用户程序没有符号信息和注释，读懂这样的程序是很困难的。但是在上传的项目的基础上，可以用 STEP 7 来进行故障诊断，还可以用“报告系统错误”功能来诊断和显示故障。

如果下载的项目被设计者加密，设置了读、写保护，这样的项目是不能上传的。在对设备和生产线订货时，电气技术人员应提醒主管部门注意 STEP 7 项目文件在故障诊断和维护中的重要意义，要求供货商提供控制系统的项目文件。如果 STEP 7 项目文件涉及到知识产权的保护问题，可以建议设备生产厂家只对某些关键的程序块加密，不对整个项目加密，这样不会影响设备投入运行后的故障诊断和使用“报告系统错误”功能来诊断和显示故障。

8.3　用 WinCC 显示消息

8.3.1　用 WinCC 和 PLCSIM 显示消息的仿真实验

本节的实验使用的是 STEP 7 V5.4.3 中文版和 V6.2 版的 WinCC。安装软件时应先安装

STEP 7。

1．创建 STEP 7 项目和组态硬件

在 STEP 7 中创建一个名为“ReptErPC”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config，将电源模块和信号模块插入机架。

双击机架中“DP”所在的行，点击出现的 DP 属性对话框的“属性”按钮，在出现的 PROFIBUS 接口属性对话框中，点击“新建”按钮，生成一条新的 PROFIBUS 子网络，将 CPU 连接到 DP 网络上。

将右边硬件目录窗口的“\PROFIBUS DP\ET 200B”文件夹中的“B-16DO”和“B-16DI”拖放到 DP 网络上，在自动打开的“属性 – PROFIBUS 接口”对话框的“参数”选项卡中，设置从站的地址为 4 和 5。

将“\PROFIBUS DP\ET 200M”文件夹中的“IM 153-1”拖放到 DP 网络上，设置从站的地址为 7。将 DI、DO 和 AO 模块拖放到 HW Config 下面窗口的“插槽”内，组态 AO 模块具有诊断功能。

2．插入 PC 站

点击管理器左侧窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→“SIMATIC PC 站点”命令，在 STEP 7 的项目中生成 PC 站点（见图 8-30）。

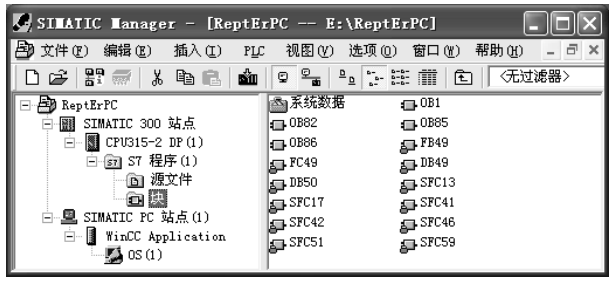


图 8-30　SIMATIC 管理器

选中生成的 PC 站，双击右边窗口中的“硬件”图标，打开 HW Config。将右边硬件目录窗口的文件夹“\SIMAIC PC Station\HMI”中的“WinCC Application”拖放到 1 号槽（见图 8-31），将文件夹“\SIMAIC PC Station\CP PROFIBUS\CP 5613”中的“SW 6.0 SP5”拖放到 2 号槽。

索引	模块 ...	订货号 ...	固件	MPI 地址	I 地址
1	WinCC Application	----			
2	CP 5613	6GK1 561-3AA00	V6.0.5	1	

图 8-31　HW Config 中的 PC 站点

用鼠标双击 CP 5613，在打开的 CP 属性对话框中，将接口类型由默认的 PROFIBUS 改为 MPI，站地址设为 1，将它连接到 MPI 网络上。用同样的方法将 CPU 连接到 MPI。

最后点击工具栏上的按钮，编译和保存组态信息。

安装了 WinCC 后，才能看到图 8-30 中的 OS 和图 8-31 中的 WinCC Application。

3．组态报告系统错误功能

选中 HW Config 中的 CPU，执行菜单命令“选项”→“报告系统错误”。在打开的对话

框中，采用默认的设置。点击“生成”按钮，生成对话框中设置的 OB、FB、FC 和 DB。详细的组态方法见 8.2.1 节。

4．编译 OS

在 SIMATIC 管理器中，用鼠标右键点击 PC 站点中的“OS”（见图 8-30），执行出现的快捷菜单中的命令“编译”，在依次打开的对话框中点击“Next”按钮，最后点击“Profile”（编译）按钮。STEP 7 用对话框显示编译是否成功的信息。

5．打开 WinCC

编译成功后用右键点击图 8-30 中的“OS”，执行快捷菜单命令“打开对象”，打开 WinCC。图 8-32 的“变量管理”中的 SIMATIC S7 Protocol Suite 是 S7 PLC 通信的驱动程序。它和右边窗口中的各通道单元是编译时自动生成的。

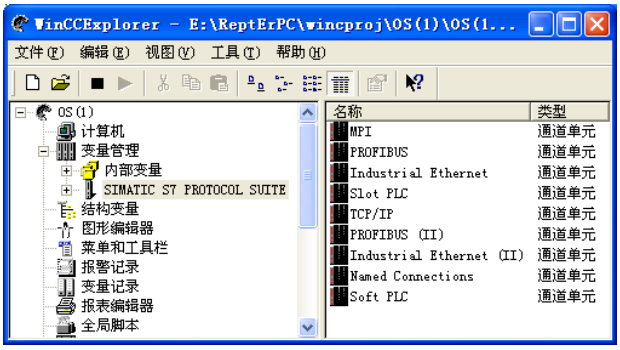


图 8-32　WinCC 管理器

6．设置 WinCC 的启动属性

用鼠标右键点击 WinCC 管理器左边窗口最上面的“计算机”，执行快捷菜单中的命令“属性”。点击打开的“计算机列表属性”对话框中的“属性”按钮，在“计算机属性”对话框的“启动”选项卡中（见图 8-33），选中“报警记录运行系统”，最后点击“确定”按钮，退出对话框。

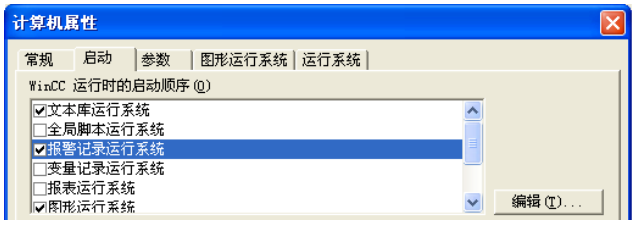


图 8-33　计算机属性对话框

打开随书光盘中的 WinCC 项目后，应选中图 8-32 左边窗口中的“计算机”，双击右边窗口中的服务器名称，在打开的“计算机属性”对话框的常规选项卡中修改计算机的名称。

用右键点击计算机桌面上的“我的电脑”图标，执行快捷菜单中的命令“属性”，在出现的“系统属性”对话框的“计算机名”选项卡中，可以找到和复制计算机的名称。

7．组态报警控件的显示内容和属性

WinCC 用报警控件显示消息，用“报警记录”视图设置报警控件显示的内容和属性。双

击 WinCC 管理器左边窗口的“报警记录”（见图 8-32），打开“报警记录”视图（见图 8-34）。视图的下面是在 STEP 7 中生成的大量的消息。

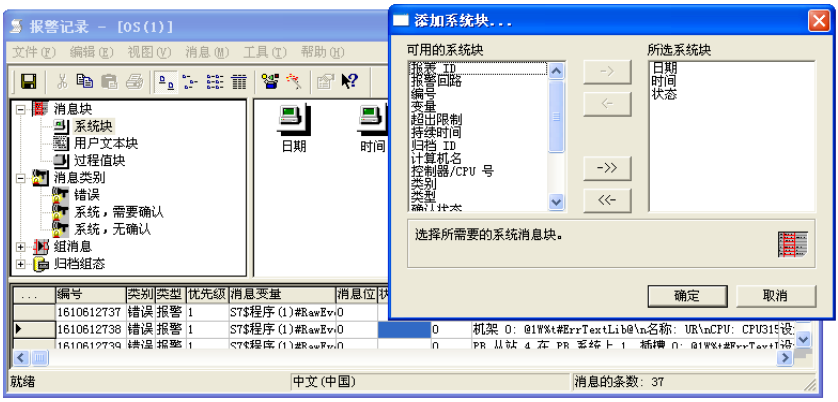


图 8-34　报警记录视图

用鼠标右键点击左边窗口的“系统块”，执行菜单命令“增加/删除”，选中打开的“添加系统块”对话框的“可用的系统块”列表中的“状态”，点击“->”按钮，将它添加到左边窗口，报警控件中将会增加“状态”列。选中右边窗口中的“编号”，点击中间的“<-”按钮，将它移到左边的窗口。

用鼠标右键点击系统块中的“状态”图标，执行快捷菜单中的“属性”命令，在打开的“消息块”对话框（见图 8-35），将报警控件中该列的宽度修改为 13 个字符。用同样的方法调节报警控件各列的宽度。

用鼠标右键点击报警记录视图左边窗口的“用户文本块”，执行菜单命令“增加/删除”，选中打开的对话框左边窗口的“块：3”，点击“->”按钮，将它移到右边窗口。选中右边窗口的“错误点”和“消息文本”，点击“<-”按钮，将它们移到左边窗口。

用鼠标右键点击“块：3”，在出现的“消息块”对话框中，将它的名称改为与 STEP 7 中相同的“消息文本”，宽度改为 95 个字符。

8．修改消息中的状态文本

用右键点击图 8-34 左边窗口“消息类别”中的“错误”，执行快捷菜单中的命令“属性”，在打开的对话框的“状态文本”选项卡（见图 8-36）中，将原来表示状态的符号改为文字。

图 8-35　消息块对话框

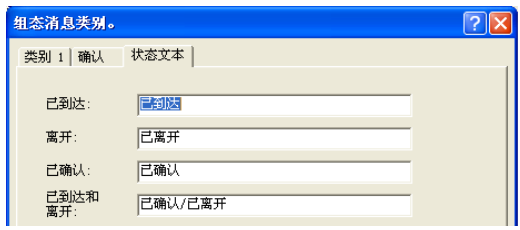


图 8-36　组态消息类别对话框

选中图 8-34 左边窗口中的“错误”，双击右边窗口中的“报警”，在打开的“类型”对话框中，将“进入”、“离开”和“已确认”的背景色改为白色。

9．编辑消息

用鼠标右键点击报警记录视图下面的某条消息，执行快捷菜单中的“属性”命令，在出现的对话框（见图 8-37）中，可以查看和修改消息的属性。

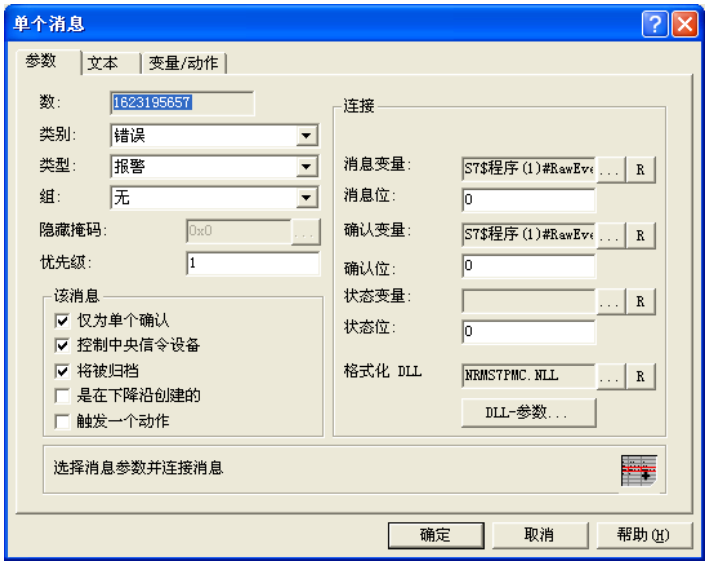


图 8-37　消息属性对话框

在“文本”选项卡（见图 8-38）中，可以查看和修改在 STEP 7 中生成的消息的信息文本和消息文本（见图 8-23）。在“变量/动作”选项卡中，可以设置消息中过程值的变量，还可以设置用报警回路来打开一个画面。

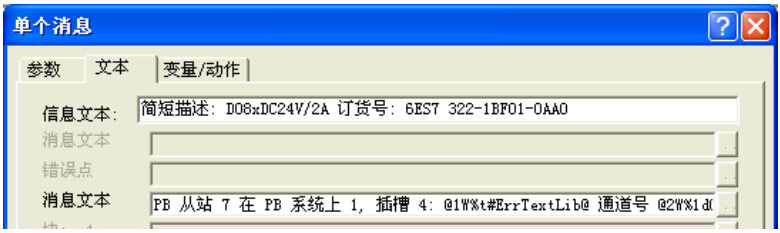


图 8-38　消息属性对话框

10．生成系统消息

执行报警记录视图中的菜单命令“工具”→“WinCC 系统信息”，点击打开的对话框中的“创建”按钮，将会生成大量的系统消息。可以在“报警记录”视图下面的列表中看到它们。

最后点击工具栏上的按钮，保存报警记录的组态信息。

11．组态画面

用右键点击 WinCC 管理器左边窗口的“图形编辑器”，执行菜单命令“新建画面”。双击右边窗口新建的画面“NewPd10”，打开图形编辑器（见图 8-39）。

选中右边窗口的“对象选项板”的“控件”选项卡中的“WinCC Alarm Control”，用鼠标在画面中生成报警控件。

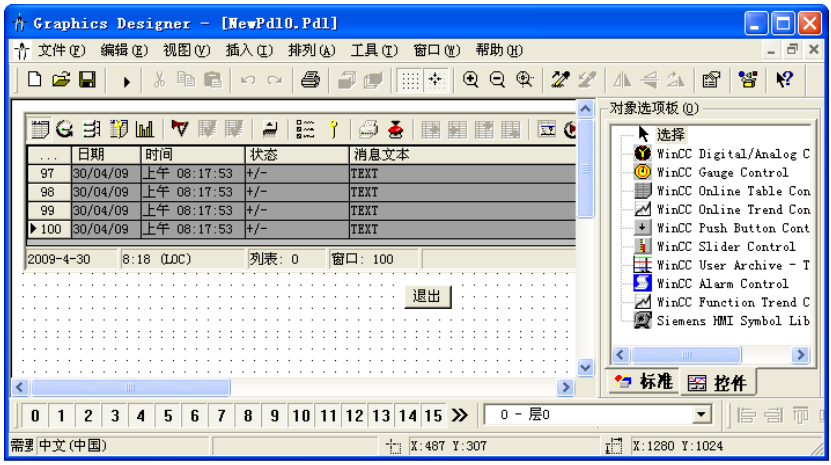


图 8-39　WinCC 的图形编辑器

双击打开它的属性对话框（见图 8-40），选中“消息列表”选项卡中的“状态”和“消息文本”，点击 → 按钮，将它们传送到右边窗口。选中右边窗口的某个消息块后，可以用上、下键移动它的排序位置，以改变它在报警控件各列中的左右位置。

在画面上还生成了一个用来退出 WinCC 运行系统的按钮（见图 8-39）。

图 8-40　报警控件属性对话框

12．显示消息的仿真实验

（1）启动仿真

点击 SIMATIC 管理器工具栏上的按钮，打开 S7-PLCSIM。选中 SIMATIC 管理器左边窗口中的“块”对象，点击工具栏上的（下载）按钮，将所有的块和系统数据下载到仿真 PLC，然后将 CPU 切换到 RUN 或 RUN-P 模式。

点击 WinCC 图形编辑器工具栏上的按钮，保存对画面的修改。点击工具栏上的按钮，打开 WinCC 运行系统。

（2）对诊断中断（OB82）故障的仿真

执行 PLCSIM 的菜单命令“Execute”（执行）→“Trigger Error OB”（触发错误 OB）→“Diagnostic Intrrupt（OB82）”（诊断中断 OB82），打开 OB82 的仿真对话框（见图 8-24）。

在“Module addrress”（模块地址）文本框中输入 7 号从站的 2AO 模块模拟量输出通道 0 的地址 PQW256。用 OB82 的仿真对话框的选择框选中“User defined”（用户定义），用复选框选中“External voltage failed”（外部电压故障），点击“Apply”（应用）按钮，WinCC 运行系统中的报警控件（见图 8-41）出现第一条消息“无外部辅助电压”。消息的状态为“已到达”。

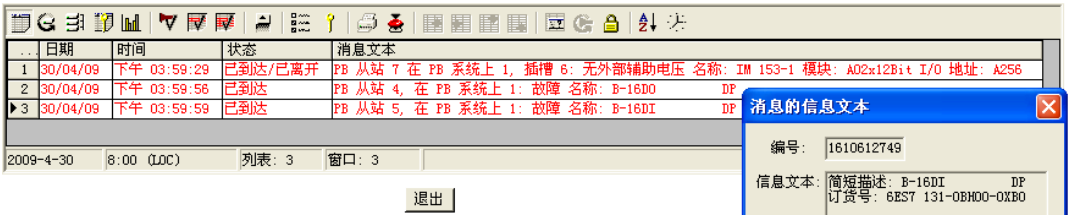


图 8-41　报警控件显示的消息

点击 OB82 的仿真对话框中的“External voltage failed”复选框，“√”消失。点击“Apply”按钮，故障消失，消息的状态变为“已到达/已离开”。

（3）对 DP 从站故障（OB86）的仿真

执行 PLCSIM 的菜单命令“Execute”→“Trigger Error OB”→“Rack failure（OB86）”（机架与 DP 故障 OB86），打开 OB86 的仿真对话框（见图 8-26）。

在 DP Failure（DP 故障）选项卡中，可以看到组态的 3 个从站。选中 4 号从站对应的小方框，它的中间出现一个“×”。用单选框选中“Station failure”（站故障），点击“Apply”按钮，报警控件出现 4 号从站故障的消息。4 号从站对应的小方框中的“×”消失，方框变为红色，表示有故障。

用 OB86 的仿真对话框中的单选框选中“Station OK”（站正常），点击“Apply”按钮，故障消失，消息的状态变为“已到达/已离开”。4 号从站对应的小方框变为绿色。用同样的方法模拟 5 号从站的故障。

点击报警控件工具栏上的按钮，打开信息文本对话框（见图 8-41 右边的小图），可以看到在 STEP 7 组态的消息的信息文本。

（4）消息的确认

点击报警控件中某一行的消息，该行最左边出现符号“▶”。点击报警视图工具栏上的（确认）按钮，该行的消息被确认。如果该行消息的状态为“已到达/已离开”，消息将会消失。如果该行的状态为“已到达”，消息被确认后，在故障消失时消息同时消失。

和按钮分别用于显示短期消息归档列表和长期消息归档列表，用它们可以显示过去的消息。点击按钮可以激活报警控件或取消激活功能。激活后，可以调节各列的宽度。

8.3.2　用 WinCC 显示硬件控制系统的消息

1．创建 STEP 7 项目和组态硬件

在 STEP 7 中创建一个名为“ReptPC1”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config，将电源模块和信号模块插入机架。

双击机架中“DP”所在的行，在打开的对话框中创建一个 PROFIBUS 子网络，将 CPU 连接到 DP 网络。在 DP 网络上创建 3 个从站，4 号从站为 ET 200B-16DO，5 号从站为 ET 200B-16DI，7 号从站为 ET 200M，其 6 号槽的 2AO 模块组态了诊断功能。

2．组态报告系统错误功能

选中 HW Config 中的 CPU，执行菜单命令“选项”→“报告系统错误”。在打开的对话框中，采用默认的设置。点击“生成”按钮，生成对话框中设置的 OB、FB、FC 和 DB。详细的情况见 8.2.1 节。

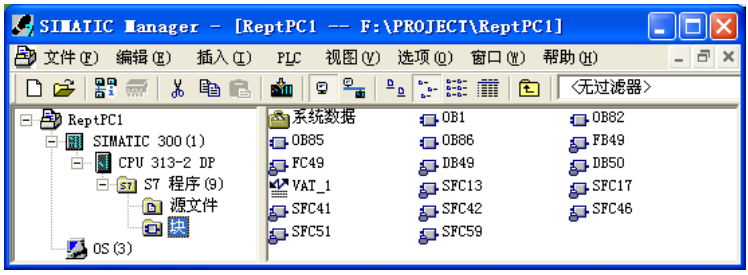


图 8-42　SIMATIC 管理器

3．插入和编译 WinCC 项目

点击 SIMATIC 管理器左边窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→“OS”命令，在 STEP 7 的项目中生成 OS（即 WinCC 的项目）。

用鼠标右键点击图 8-42 中的“OS”，执行快捷菜单命令“编译”，在依次打开的对话框中点击“Next”按钮，最后点击“Profile”（编译）按钮。

4．打开 WinCC

编译成功后用鼠标右键点击“OS”，执行快捷菜单命令“打开对象”，打开 WinCC（见图 8-43）。用鼠标右键点击 WinCC 管理器左边窗口最上面的“计算机”，执行快捷菜单中的命令“属性”。点击打开的“计算机列表属性”对话框中的“属性”按钮，在“计算机属性”对话框的“启动”选项卡中，激活“报警记录运行系统”（见图 8-33）。

图 8-43　WinCC 管理器

本项目用硬件 PLC 和 ET 200 取代了 PLCSIM 的仿真 PLC，CPU 和 DP 从站通过 CP 5613 和 PROFIBUS 网络与运行 WinCC 的计算机通信，因此需要在 WinCC 中组态计算机与 PLC 通信的硬件接口和通信协议。

用右键点击图 8-43 左边窗口的“PROFIBUS”，执行快捷菜单命令“系统参数”，用打开的对话框设置“逻辑设备名称”为 CP 5613_5614（PROFIBUS）。

此外还应在 STEP 7 的 SIMATIC 管理器中执行菜单命令“选项”→“设置 PC/PG 接口”，将计算机的通信接口设置为 CP 5613_5614（PROFIBUS）。

5．组态报警控件的显示内容和属性

双击 WinCC 管理器左边窗口的“报警记录”，打开“报警记录”视图（见图 8-34）。按

8.3.1 节的要求，设置报警控件显示的列为“日期”、“时间”、“状态”和“消息文本”，设置各列的宽度。

用右键点击报警记录视图左边窗口“消息类别”中的“错误”（见图 8-34），执行快捷菜单中的命令“属性”，在打开的对话框的“状态文本”选项卡（见图 8-36）中，将原来表示状态的符号改为文字。

选中报警视图左边窗口中的“错误”，双击右边窗口中的“报警”，在打开的“类型”对话框中，将“进入”、“离开”和“已确认”的背景色改为白色。

最后点击工具栏上的按钮，保存报警记录的组态信息。

6．画面组态

用右键点击 WinCC 管理器左边窗口的“图形编辑器”，执行菜单命令“新建画面”。双击右边窗口生成的画面“NewPd10”，打开图形编辑器（见图 8-39）。

选中右边窗口的“对象选项板”的“控件”选项卡中的“WinCC Alarm Control”，用鼠标在画面中生成报警控件。双击打开它的属性对话框（见图 8-40），点击“消息列表”选项卡中的“>>”按钮，将左边窗口的消息块传送到右边窗口。在画面上还组态了一个用来退出 WinCC 运行系统的按钮。最后点击工具栏上的按钮，保存对画面的组态。

7．显示消息的实验

做好上面的准备工作后，用 DP 电缆连接好 CPU、3 个 DP 从站和 CP 5613 的 DP 接口。PLC 和从站的电源通电后，将程序和组态信息下载到 CPU，将 CPU 切换到 RUN 模式。

点击 WinCC 工具栏上的按钮，启动 WinCC 的运行系统。

断开 5 号从站的电源，出现图 8-44 中的第 2 条消息，其状态为“已到达”。点击第 2 行，符号“▶”移到第 2 行最左边，表示选中了第 2 行。点击报警视图工具栏上的（确认）按钮，该行的消息被确认，其状态为“已到达/已确认”。接通 5 号从站的电源，故障消失，第 2 行的消息随之消失。

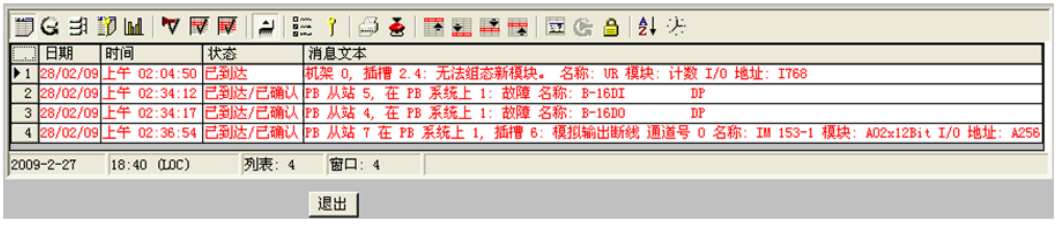


图 8-44　报警控件显示的消息

如果 5 号从站的电源断开期间没有确认消息，电源恢复后消息的状态变为“已到达/已离开”。 选中该消息后点击报警视图工具栏上的（确认）按钮，该消息被确认后消失。

断开 7 号从站（ET 200M）6 号槽的 2AO 模块 0 号通道的电流输出电路，报警视图出现第 4 条信息“插槽 6：模拟输出断线”，并给出了从站编号、模块的插槽号、通道号，模块的型号和 I/O 地址。消息中的故障信息非常详细，与 STEP 7 中故障模块的“模块信息”对话框看到的信息相同（见图 6-15）。点击报警视图工具栏上的按钮，打开信息文本对话框，可以看到模块的简要描述和订货号。

与用 PLCSIM 仿真相比，硬件故障触发的消息的文本更为丰富、详细和准确。

8.3.3 组态 PC 站点实现 WinCC 和 PLC 的通信

8.3.2 节用 OS 创建 WinCC 项目的方法简单易行，通常用于只有一台上位计算机的小型控制系统。如果系统中有多台上位计算机和多台 CPU，用这种方法组态网络很不方便。大型复杂的控制系统常用软件 SIMATIC NET 来组态网络。作者使用的是随书光盘中的 SIMATIC NET 2007。在组态时用 PC 站点来代替上一个项目的 OS，PC 站点对应于一台运行 WinCC 的计算机，可以在 NetPro 中方便地组态 PLC 和 PC 站点之间的连接。

本节的例程需要安装 STEP 7、WinCC、WinCC flexible 和 SIMATIC NET 才能正常运行。

1．创建 STEP 7 项目和组态硬件

在 STEP 7 中创建一个名为“ReptPC2”的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config，将电源模块和信号模块插入机架。

双击机架中“DP”所在的行，在打开的对话框中创建一个 PROFIBUS 子网络，将 CPU 连接到 DP 网络上。在 DP 网络上创建 3 个从站，4 号从站为 ET 200B-16DO，5 号从站为 ET 200B-16DI，7 号从站为 ET 200M，其 6 号槽的 2AO 模块具有诊断功能（见图 8-45）。

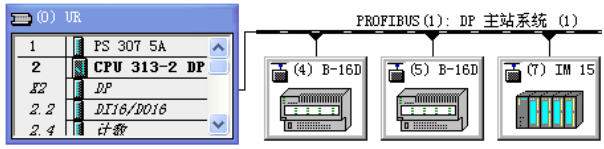


图 8-45　组态 DP 从站

2．组态报告系统错误功能

选中 HW Config 中的 CPU，执行菜单命令“选项”→“报告系统错误”。在打开的对话框中，采用默认的设置。点击“生成”按钮，生成对话框中设置的 OB、FB、FC 和 DB。详细的组态方法见 8.2.1 节。

3．组态 HMI 站点

在 SIMATIC 管理器组态一个 HMI 站点（见图 8-46），将它连接到 DP 网络上，在画面上生成一个报警视图。具体的组态方法见 8.1.3 节。

4．插入 PC 站点

点击管理器左侧窗口最上面的项目图标，执行弹出的快捷菜单中的“插入新对象”→“SIMATIC PC 站点”命令，在 STEP 7 的项目中生成 PC 站点（见图 8-46）。

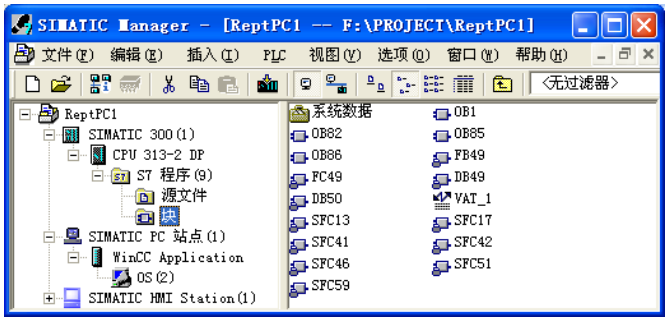


图 8-46　SIMATIC 管理器

选中生成的 PC 站点，双击右边窗口的“组态”图标，打开 HW Config。将右边硬件目录窗口的文件夹“\SIMAIC PC Station\HMI”中的“WinCC Application”拖放到 1 号槽（见图 8-47），将文件夹“\SIMAIC PC Station\CP PROFIBUS\CP 5613”中的“SW 6.0 SP5”拖放到 2 号槽。

索引	模块	订货号	固件	MPI 地址	I 地址
1	WinCC Application	----			
2	CP 5613	6GK1 561-3AA00	V6.0.5	1	

图 8-47　HW Config 中的 PC 站点

双击 2 号槽的 CP 5613，在打开的 CP 属性对话框中，将站地址设为 0，将 CP 连接到 DP 网络上。用同样的方法将 CPU 连接到 DP 网络上。最后点击工具栏上的按钮，编译和保存组态信息。CP 5613 使用 PROFIBUS 协议，通过 PROFIBUS 接口连接 CPU、DP 从站和 HMI（见图 8-48）。

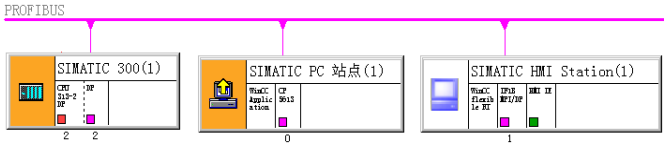


图 8-48　NetPro 中的站点

5．在 SIMATIC NET 中组态 PC 站点

安装 SIMATIC NET 2007 后，双击 Windows 任务栏中的图标，打开 Station Configuration Editor。在 1 号槽插入 WinCC Application，在 2 号槽插入 CP 5613。PC 站的名称和 CP 5613 的 DP 地址应与 STEP 7 中组态的相同。

详细的组态方法可以参考 13.2.1 节，图 8-49 是组态和下载成功后的站组态编辑器。

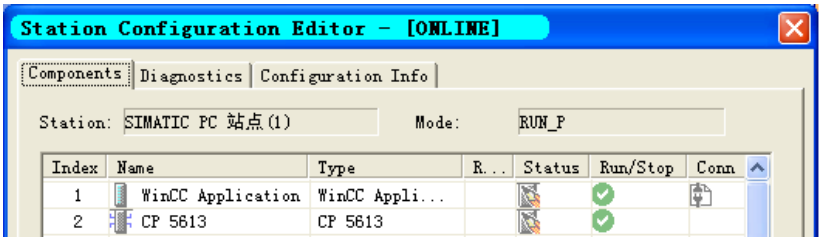


图 8-49　组态 CP 站点

6．组态 CP 5613

完成 PC 站的硬件组态后，点击 Windows 下面的工具栏左边的“开始”按钮，执行菜单命令“开始”→“所有程序”→“SIMATIC”→“SIMATIC NET”→“Configuration console”，打开组态控制台。选中左边窗口的文件夹“\Modules\CP5613/5614”中的“General”图标（见图 13-7），CP 5613 的模式应为 Configuration mode（组态模式），插槽号（Index）自动指向 2。

选中左边窗口的“Access Points”（访问点，见图 13-8），双击右边窗口的“S7ONLINE”，在出现的“S7ONLINE 属性”对话框中，选中“PC internal（local）”后点击“确定”按钮。

在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PC/PG 接口”，可以看到 PC 的通信接口被自动设置为“PC internal”，在这种模式下 CP 5613 可以作为下载和监控的通信接口使用。

7．下载组态数据

完成上述组态任务后，用 DP 电缆连接 CP 5613 和 CPU 313C 的 DP 接口。

打开 SIMATIC 管理器，选中 PC 站点，点击工具栏上的  （下载）按钮，将组态数据下载到 PC 站点。选中管理器左边窗口的“块”文件夹，将程序块和组态信息下载到 CPU。

下载成功后，打开 Station Configuration Editor 视图（见图 8-49），检查组件状态。1 号插槽的 Conn 列出现连接图标，表明连接被激活。

8．编译和打开 WinCC 的项目

在 SIMATIC 管理器中，用鼠标右键点击 PC 站点中的“OS”（见图 8-46），执行快捷菜单命令“编译”，在依次打开的对话框中点击“Next”按钮，最后点击“Profile”（编译）按钮。编译成功后用右键点击“OS”，执行快捷菜单命令“打开对象”，打开 WinCC。

9．WinCC 中的组态和实验

在 WinCC 管理器中，用右键点击左边窗口“变量管理”文件夹中的“PROFIBUS”（见图 8-43），执行快捷菜单命令“系统参数”，用打开的对话框设置逻辑设备名称为 CP 5613_5614。

用鼠标右键点击左边窗口的“计算机”，用复选框设置启动时打开“报警记录运行系统”。打开“报警记录”视图，按项目 ReptPC1 的要求组态与报警控件有关的属性。打开图形编辑器，生成和组态一个报警控件。

用 DP 电缆连接好 CPU、3 个 DP 从站和 CP 5613 的 DP 接口，通电后将 CPU 切换到 RUN 模式。点击 WinCC 工具栏上的  按钮，启动 WinCC 的运行系统。

用硬件故障触发和显示消息的操作步骤和实验结果与前一个项目 ReptPC1 的相同。

8.4 练习题

1. 控制系统的故障信息主要用什么方式显示？这种方式有什么优点？

2. 有哪两种传送消息的方法？各有什么特点？

3. 怎样组态与块有关的消息？

4. 生成一个项目，自行确定系统结构、CPU、信号模块和 DP 从站的型号，组态某些模块的诊断功能。组态“报告系统错误”功能，生成有关的消息和程序块。生成一个 HMI 站点，参照 8.2.2 节，对 WinCC flexible 的项目组态。用 PLCSIM 的仿真功能模拟产生从站的故障，观察是否能实现故障消息的传送和显示。

第 9 章　PROFIBS-PA

9.1　PROFIBS-PA 网络的组态

9.1.1　PROFIBUS-PA 概述

1．PROFIBUS-PA

PA 是 Process Automation（过程自动化）的缩写。PROFIBUS-PA 用于 PLC 与过程自动化的现场传感器和执行器的低速数据传输，特别适合于在需要防爆的化学工业和过程控制中使用。PROFIBUS-PA 功能集成在起动执行器、电磁阀和测量变送器等现场设备中。

PROFIBUS-PA 基于 MBP（Manchester Coding and Bus Powering，曼彻斯特码编码与总线供电）传输技术，由于采用 IEC 1158-2 标准，确保了本质安全和通过屏蔽双绞线电缆进行数据传输和供电，适用于防爆区域的传感器和执行器与中央控制系统的通信。

PROFIBUS-PA 采用 PROFIBUS-DP 的基本功能来传送测量值和状态。并用扩展的 PROFIBUS-DP 功能来制订现场设备的参数和进行设备操作。PROFIBUS-PA 规范保证了不同厂商生产的现场设备的互换性和互操作性，它是 PROFIBUS-PA 的组成部分。

PA 规范已对所有通用的测量变送器和其他一些设备类型作了具体规定，这些设备包括压力、液位、温度和流量变送器，数字量输入/输出，模拟量输入/输出，阀门和定位器等。

可以用 DP/PA 耦合器或 DP/PA 链接器将 PROFIBUS-PA 设备集成到 PROFIBUS-DP 网络中。在危险区域每个 DP/PA 链路可以连接 10 个现场设备，在非危险区域每个 DP/PA 链路可以连接 30 个现场设备。传输速率为 31.25 kbit/s，可以采用总线型或树形结构。即使在本质安全区增加和去除总线站点，也不会影响到其他站。

PA 通信包括循环访问和非循环访问，用 PLC 循环访问输入/输出。非循环访问的典型例子是用工程工具软件（例如过程设备管理器 PDM）设置设备的运行参数。

与 PROFIBUS-DP 设备一样，PROFIBUS-PA 设备也是用制造商的 GSD 文件来描述的，在 STEP 7 的硬件组态工具中安装 GSD 文件后，新增的 PA 设备将在设备目录中出现。

2．PROFIBUS-PA 的 IEC 1158-2 传输

PROFIBUS-PA 采用符合 IEC 1158-2 标准的传输技术，即曼彻斯特码编码与总线供电传输技术。这种技术确保本质安全，并通过总线直接给现场设备供电，能满足石油化学工业的要求。用曼彻斯特编码传输数据时，从 0 (−9mA)到 1(+9mA)的上升沿发送二进制数“0”（见图 9−1），从 1 到 0 的下降沿发送二进制数“1”。每一位的前半位电平对应于传送的二进制数（高电平为 1，低电平为 0），后半位与前半位的电平相反。

传输速率为 31.25 kbit/s。传输媒体为屏蔽或非屏蔽的双绞线，允许使用线性、树形和星形网络。

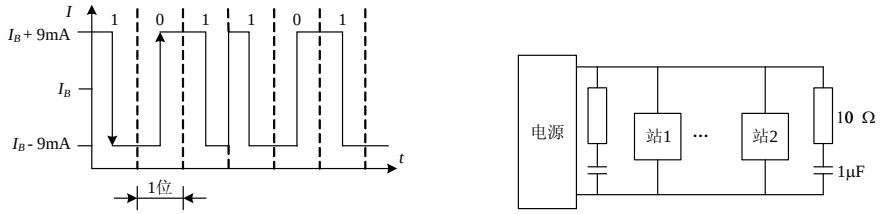


图 9–1　PROFIBUS-PA 的数据传输

总线段的两端用一个无源的 RC 总线终端器来终止，在一个 PA 总线段上最多可以连接 32 个站，站的总数最多为 126 个，最多可以扩展 4 台中继器。最大的总线段长度取决于供电装置、导线类型和所连接的站的电流消耗。

为了增加系统的可靠性，可以用冗余总线段作总线段的备份。DP/PA 链接器用于 PA 总线段与 DP 总线段的连接（见图 1-4 和图 9–3）。

3．DP/PA 耦合器

DP/PA 耦合器（Coupler）用于将 PA 现场设备连接到 PROFIBUS-DP 网络。DP/PA 耦合器可完成下列任务：

1）数据从 11 bit/字符的异步编码转换为 8 bit/字符的同步编码，将来自 PROFIBUS-DP 的报文简单地转换为 PROFIBUS-PA 报文。

2）将 DP 网络的传输速率（45.45 kbit/s）转换为固定的 31.25 kbit/s。

3）通过传送数据的电缆对现场设备供电。

使用多个DP/PA耦合器，一个DP主站系统最多可以连接 125 个PA从站。

使用 DP/PA 耦合器后，DP 网络的通信速率最高为 45.45kbit/s。系统规模较小（最多十几台设备）和对通信速率要求不高时，可以使用 DP/PA 耦合器来连接 DP 网络和 PA 现场设备（见图 9–2）。PA 从站被映射为 DP 从站，就像组态 DP 从站一样组态 PA 设备，需要为 PA 设备设置 DP 地址。

PA从站的诊断方法与DP从站相同，例如可以用SFC 13 或SFC 51 诊断每个 PA从站。

DP/PA 耦合器有两种类型（见图 9–2）：non-Ex 型（非本质安全型）和 Ex 型（本质安全型）。在网络中，DP/PA 耦合器是“透明”的，它没有地址，不用对它组态。

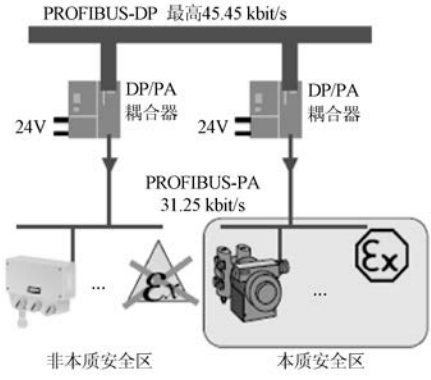


图 9–2　DP/PA 耦合器

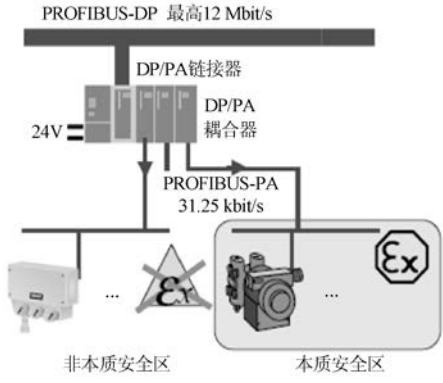


图 9–3　DP/PA 链接器

一个DP/PA耦合器的Non-Ex区（非本质安全区）最多可连接 30 个现场设备，Ex区（本质安全区）最多可连接 10 个现场设备。DP/PA耦合器在Non-Ex区的网络允许最大长度为 1.9km，Ex区为 1km。

DP/PA耦合器后面的PA从站的总线地址不能与DP主站系统的其他站点地址重叠，因为PA现场设备是被视为“直接”连接在DP总线上的。

4．DP/PA 链接器

DP/PA 链接器（Link）通过 DP/PA 耦合器与 PA 从站交换信息（见图 9–3），一个 DP/PA 链接器最多可以连接 5 个 DP/PA 耦合器，最多 64 个 PA 从站。

DP/PA链接器是连接PROFIBUS-DP和PROFIBUS-PA的网关。PLC通过DP/PA 链接器访问现场设备，就像访问一个由PA设备作为模块构成的模块化设备。

DP/PA 链接器是 DP 网络上的一个从站，链接器和它连接的 PA 现场设备占用一个 DP 地址。使用 DP/PA 链接器时，DP 网络的传输速率最高可达 12Mbit/s。DP/PA 链接器又是 PA 网络中的主站，一个链接器可以驱动的 PA 从站的数目与它的版本有关，订货号为 6ES7 157-0AA82-0XA0 的 IM 157 最多可连接 64 个 PA 从站，比它低档的 IM 157 为 32 个。

PA现场设备独立于DP网络单独编址，其地址称为PA地址（3～124），PA主站默认的PA地址为 2。只能为每个DP/PA链接器分配 244B的组态数据和参数数据。在某些情况下，这将导致减少每个DP/PA链接器上连接的PA现场设备的个数。

通过 GSD 文件组态 DP/PA 链接器时，一个 PA 从站可能占用多个插槽。一个 DP/PA 链接器的最大插槽数目为 236。

可以用SFC 13 或SFC 51 实现IM 157 的诊断。DP/PA链接器下面的每个 PA从站的诊断数据可以通过IM 157 以数据记录的方式检索出来。DPV0 主站和DPV1 主站可以分别用SFC 59“RD_REC”和SFB 52“RDREC”读取数据记录，实现对PROFIBUS-PA的诊断。

系统规模较大（例如超过 20 台现场设备）和对时间要求较苛刻时，建议采用 DP/PA 链接器加 DP/PA 耦合器的方案。Non-Ex 区（非本质安全区）最多可连接 30 个现场设备，Ex 区（本质安全区）最多可连接 3×10 个现场设备。

用 DP/PA 耦合器和 DP/PA 链接器组态 PA 系统时，应保证它们连接的所有现场设备的功耗不超过允许的值，具体的参数可以查阅有关的产品用户手册。

随书光盘中的文件《PROFIBUS-PA 应用技术手册》和《如何配置 PA 总线仪表方案》给出了不同应用场合下 PA 网络的配置方案，包括设备选型和订货号。

9.1.2 仅使用 DP/PA 耦合器的 PROFIBUS–PA 网络组态

在 STEP 7 中创建一个名为 PA_1 的项目（见随书光盘文件夹\Project\PA 中的同名例程），CPU 为 CPU 315-2DP。选中该站点，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，在出现的对话框的“常规”选项卡中点击“属性”按钮，在出现的对话框的“参数”选项卡中，采用默认的站地址 2。点击“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”对话框的“网络设置”选项卡中，设置网络的传输速率为 45.45 kbit/s，配置文件为默认的“DP”。

多次点击“确定”按钮，返回硬件组态窗口。此时只能看到 S7-300 的机架和新生成的

PROFIBUS（1）网络线。图 9-5 是已经组态好的 PROFIBUS 网络。

安装 STEP 7 以后，HW Config 右边的硬件目录窗口的“PROFIBUS-PA”文件夹中没有 PA 设备，需要在互联网下载和安装 PA 现场设备的 GSD 文件，才能对 PA 设备组态。

可以在西门子公司的支持网站搜索“PA GSD”，下载西门子的 PA 现场设备的 GSD 文件。本例程安装了 3 种 PA 现场设备的 GSD 文件：

1）SIPART PS 2：智能电气阀门定位器，可以用于直行程和角行程的阀门。

2）SITRANS FM：用于测量导电液体的体积流量的电磁流量计。

3）SITRANS P DSIII：有 HART 或 PROFIBUS-PA 通信功能的高性能压力变送器。

执行 HW Config 的菜单命令“选项”→“安装 GSD 文件”，在出现的“安装 GSD 文件”对话框（见图 9-4）中，点击“浏览”按钮，用出现的“浏览文件夹”对话框选中随书光盘中的文件夹“\Project\PA\PA_GSD”，点击“确定”按钮后，该文件夹中的 GSD 文件出现在列表框中。选中需要安装的 GSD 文件，点击“安装”按钮，开始安装。

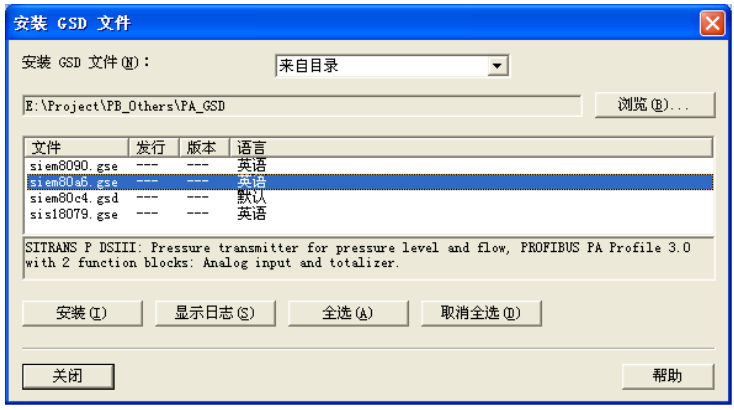


图 9-4　安装 PA 设备的 GSD

安装结束后，在 HW Config 右边的硬件目录窗口的“\PROFIBUS-PA”文件夹中，可以找到新安装的 PA 现场设备（见图 9-5）。

图 9-5　组态 PA 设备

将该文件夹中的“SIPART PS2”图标拖放到 DP 网络线上，在自动打开的 PROFIBUS 接口属性对话框中，设置其 DP 地址为 3。点击“确定”按钮，返回 HW Config，可以看到 DP 网络上的 SIPART PS2。注意不需要组态 PA 耦合器。

用右键点击某个 PA 从站的图标，执行出现的快捷菜单中的“对象属性”对话框，可以对从站组态，例如修改从站的站地址。安装 PDM 软件包后，才能对某些 PA 设备作进一步的设置。选中 PA 网络线上的某个 PA 从站，在下面的窗口可以看到该从站的详细信息，例如该从站的地址信息（见图 9-5）。双击插槽 1 所在的行，在打开的 DP 从站对话框中，可以看到默认的数据一致性属性为“总长度”（不可更改），本例程中其他 PA 从站的输入/输出地址区的一致性均为“总长度”，因此必须在 OB1 中调用 SFC 14，将接收到的数据解包；调用 SFC 15，将需要发送的数据打包后发送到 PA 从站。

组态好硬件后，点击工具栏上的按钮，编译并保存组态信息。

下面是 OB1 读写各 PA 从站数据的程序：

程序段 1：将接收到的 3 号 PA 从站的数据解包后保存

```
CALL   "DPRD_DAT"                  //调用 SFC 14
  LADDR    :=W#16#100              //PA从站的起始地址
  RET_VAL  :=MW10                  //状态字
  RECORD   :=P#M 20.0 BYTE 10      //保存读取的数据的地址区
```

程序段 2：将数据打包后发送到 3 号 PA 从站

```
CALL   "DPWR_DAT"                  //调用 SFC 15
  LADDR    :=W#16#100              //PA从站的起始地址
  RECORD   :=P#M 45.0 BYTE 5       //存放要发送的数据的地址区
  RET_VAL  :=MW12                  //状态字
```

程序段 3：将接收到的 4 号 PA 从站的数据解包后保存

```
CALL   "DPRD_DAT"                  //调用 SFC 14
  LADDR    :=W#16#10A              //PA从站的起始地址
  RET_VAL  :=MW10                  //状态字
  RECORD   :=P#M 30.0 BYTE 10      //保存读取的数据的地址区
```

程序段 4：将接收到的 5 号 PA 从站的数据解包后保存

```
CALL   "DPRD_DAT"                  //调用 SFC 14
  LADDR    :=W#16#114              //PA从站的起始地址
  RET_VAL  :=MW10                  //状态字
  RECORD   :=P#M 40.0 BYTE 5       //保存读取的数据的地址区
```

9.1.3 使用 DP/PA 链接器的 PROFIBUS-PA 网络组态

在 STEP 7 中创建一个名为 PA_2 的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。选中该站点，点击右边窗口的“硬件”图标，打开硬件组态工具 HW Config，将电源模块和信号模块插入机架。

双击机架中 CPU 模块内标有 DP 的行，新建一个 DP 网络 PROFIBUS（1），网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。图 9-6 是已经组态好的 PROFIBUS 网络。

将文件夹“\PROFIBUS DP\DP/PA Link”中的 IM 157 拖放到 DP 网络线上，在自动打开的 IM 157 的 PROFIBUS 接口属性对话框中，PROFIBUS（2）的传输速率为 45.45 kbit/s，设置 IM 157 的 DP 地址为 3。点击“确定”按钮，返回 HW Config，可以看到 DP 网络上的 IM 157 和自动生成的 PA 主站系统网络线。此时还没有图中的 PA 从站。

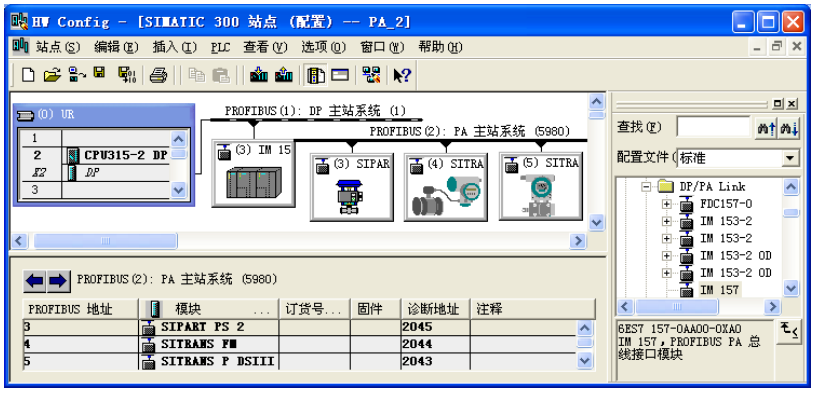


图 9-6　组态 DP、PA 网络

选中左边窗口的 IM 157，在下面的组态信息窗口中，可以看到 IM 157 是 PA-Master（PA 主站）模块，其 PROFIBUS 地址（即 PA 网络中的站地址）为 2。选中 PA 网络线，在下面的窗口中可以看到 PA 从站列表（见图 9-6）。其中的“PROFIBUS 地址”是 PA 地址。双击 PA 网络线，点击打开的“PA 主站系统属性”对话框中的“属性”按钮（见图 9-7），在打开的“PROFIBUS 属性”对话框的“网络设置”选项卡中，可以看到 PA 的传输速率为 45.45 kbit/s。

图 9-7　PA 主站系统属性对话框

将硬件目录窗口的“\PROFIBUS-PA”文件夹中的“SIPART PS2”图标拖放到 PA 主站系统网络线上。在自动打开的 SIPART PS2 接口属性对话框中，设置 PA 站地址为 3。图 9-6 中有两个 3 号从站，IM 157 是 DP 网络中的 3 号从站和 PA 网络中的主站（2 号站），SIPART PS2 是 PA 网络中的 3 号从站。

将另外两个 PA 设备的图标拖放到 PA 网络线上，自动分配的 PA 站地址分别为 4 和 5。

与项目 PA_1 相同，PA 从站的输入/输出地址区的一致性均为“总长度”，因此应在 OB1 中调用 SFC 14，将接收到的数据解包；调用 SFC 15，将要发送的数据打包后发送到 PA 从站，其程序与项目 PA_1 相同。

9.1.4　使用 PDM 组态 PROFIBUS-PA 设备

1．SIMATIC PDM

SIMATIC PDM（Process Device Manager，过程设备管理器）是用于组态、参数分配、调试、诊断以及维护智能过程设备和自动化部件的制造商通用工具软件。借助于 SIMATIC PDM，

可以用一个用户接口，组态大量不同制造商生产的现场设备。可以简单地设置、修改过程数据，并检查数据是否可靠。此外，还可以在线监控选定的过程值、设备状态信号和报警信息。SIMATIC PDM 还有仿真、诊断、管理和调试功能。

SIMATIC PDM 支持 PROFIBUS DP/PA、HART、Modbus 等通信协议。通过 LifeList，无需组态知识就可以诊断参数分配的错误和现场设备的故障。SIMATIC PDM 全面支持与现场设备有关的功能，例如仿真、测试测量电路、定义特征曲线、标定功能和文档等。不用停止 PROFIBUS 主站的运行，就可以在线修改所有现场设备的参数。

SIMATIC PDM 支持符合 PROFIBUS 用户组织（PUO）制定的 PROFIBUS-PA 配置文件描述的现场设备，以及符合电子设备描述（EDD）或 HART 设备描述（HART DD）的设备。通过 GSD 文件，可以把 PA 现场设备作为 DP 标准从站来组态。

2．SITRANS T3K PA 温度变送器

SITRANS T3K PA 是一体化温度变送器，它将来自热电阻、电阻变送器、热电偶和电压变送器的信号转换成数字信号。测量值通过微处理器，转化为温度输出值和其他状态参数，通过 PROFIBUS PA 传送到上位机。它的结构紧凑，可以安装在带盖的 B 型接线盒中。SITRANS T3K PA 必须用软件 SIMATIC PDM 来组态。

3．安装 SIMATIC PDM

在安装随书光盘中的 SIMATIC PDM 的过程中，将会出现图 9-8 所示的视图，要求安装设备库。点击“Browse”（浏览）按钮，在出现的对话框中打开 PDM 软件所带的 PDM 库，在“Souse”（源）文本框中出现 PDM 库所在的文件夹。在“Device type”（设备类型）列表中，选中需要安装 GSD 文件的现场设备的制造商和设备的类型。点击“OK”按钮，出现安装进度对话框。

安装好 SIMATIC PDM 后，在 STEP 7 的 HW Config 右边的设备窗口的文件夹\PROFIBUS-PA 中，可以看到安装 PDM 时选中和安装的设备库中的设备。

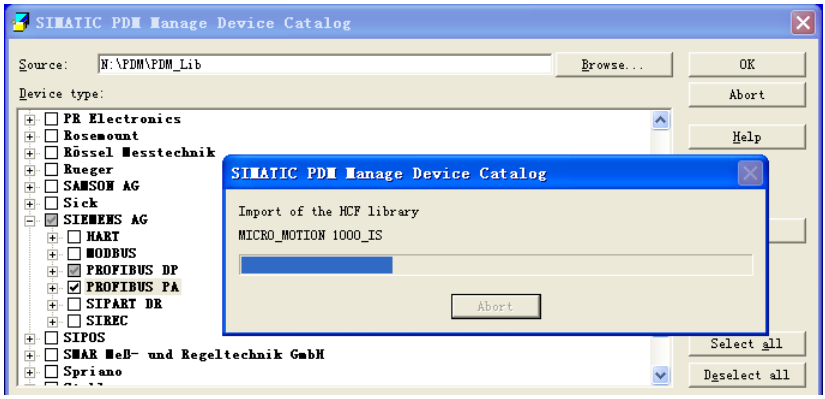


图 9-8　安装 PDM 的设备库

4．项目组态

在 STEP 7 中创建一个名为 PA 的项目（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。打开 HW Config（见图 9-9），将电源模块和信号模块插入机架。双击机架中 CPU 模块内标有 DP 的行，在出现的对话框的“常规”选项卡中点击“属性”按钮，在出现的对话框的“参数”选项卡中，采用默认的站地址 2。点击“新建”按钮，在出现的“属性 – 新建子网 PROFIBUS”

对话框的“网络设置”选项卡中，设置网络的传输速率为 45.45 kbit/s，配置文件为默认的 DP。

多次点击“确定”按钮，返回硬件组态窗口。此时只能看到 S7-300 的机架和新生成的 PROFIBUS（1）网络线。图 9–9 是已经组态好的 PROFIBUS 网络。

图 9–9　组态 PA 从站

将右边的设备目录窗口的文件夹“\PROFIBUS-PA\Sensors\Temperature\SIEMENS AG”中的“SITRANS T3K”拖放到右边窗口的 DP 网络线上。在自动打开的 PA 从站的 PROFIBUS 接口属性对话框中，设置 PA 从站的站地址为 18。只有安装了 PDM 和有关的设备库，才能看到图 9–9 右边窗口的文件夹和 T3K。

实际硬件需要通过 DP/PA 耦合器来连接 DP 网络和 T3K，耦合器不需要组态。

点击“确定”按钮，返回 HW Config。用右键点击新生成的 PA 从站，执行弹出的快捷菜单中的“对象属性”按钮，在出现的“DP 从站属性”对话框中，可以看到从站的诊断地址为 2046。选中 PA 从站，双击下面窗口中的第一行，在出现的 DP 从站属性对话框中，将该从站输入区的起始地址修改为过程映像输入区中的 IB60。

点击工具栏上的按钮，编译和保存组态信息。点击按钮，用 MPI 接口将组态信息下载到 CPU。

5．用 PDM 组态 T3K 的参数

在 HW Config 中双击 PA 从站，自动打开 SIMATIC PDM，在出现的“User”（用户）对话框（见图 9–10）中，用单选框选中“Specialist”（专家），允许用户修改 T3K 的参数值。点击“OK”按钮确认。

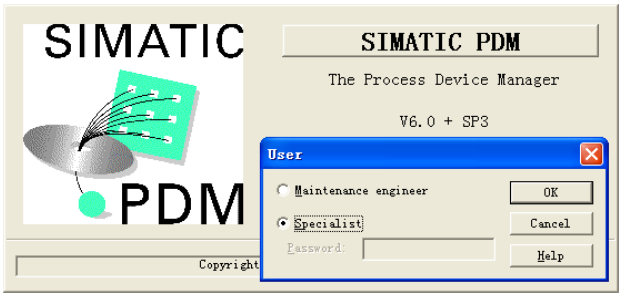
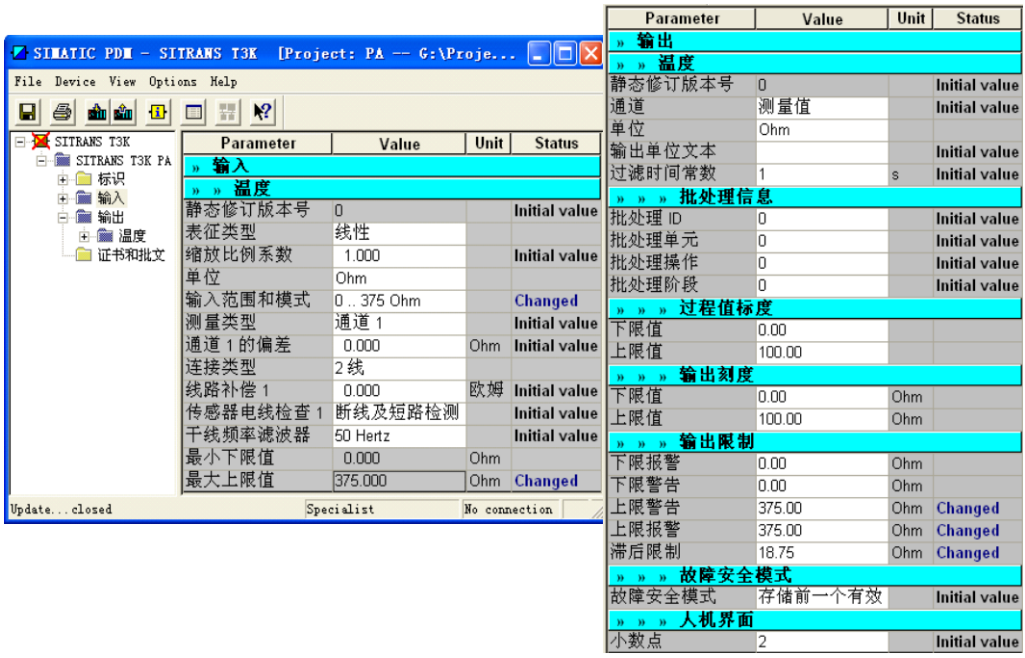


图 9–10　设置 PDM 的用户模式

在打开的 SIMATIC PDM 视图中，如果选中左边窗口的“标识”，则可看到右边窗口中的产品基本信息。图 9-11 给出了 T3K 的输入/输出参数。因为做实验时用电位器来作传感器，在输入区设置“表征类型”为“线性”，输入范围为 0～375Ω，其他参数均未修改。参数的 Status（状态）列中的“Initial value”为自动设置的初始值，“Changed”表示该参数被修改。



Parameter	Value	Unit	Status
» 输入			
» » 温度			
静态修订版本号	0		Initial value
表征类型	线性		
缩放比例系数	1.000		Initial value
单位	Ohm		
输入范围和模式	0 .. 375 Ohm		Changed
测量类型	通道 1		Initial value
通道 1 的偏差	0.000	Ohm	Initial value
连接类型	2 线		
线路补偿 1	0.000	欧姆	Initial value
传感器电线检查 1	断线及短路检测		Initial value
干线频率滤波器	50 Hertz		Initial value
最小下限值	0.000	Ohm	
最大上限值	375.000	Ohm	Changed

Parameter	Value	Unit	Status
» 输出			
» » 温度			
静态修订版本号	0		Initial value
通道	测量值		Initial value
单位	Ohm		
输出单位文本			Initial value
过滤时间常数	1	s	Initial value
» » » 批处理信息			
批处理 ID	0		Initial value
批处理单元	0		Initial value
批处理操作	0		Initial value
批处理阶段	0		Initial value
» » » 过程值标度			
下限值	0.00		
上限值	100.00		
» » » 输出刻度			
下限值	0.00	Ohm	
上限值	100.00	Ohm	
» » » 输出限制			
下限报警	0.00	Ohm	
下限警告	0.00	Ohm	
上限警告	375.00	Ohm	Changed
上限报警	375.00	Ohm	Changed
滞后限制	18.75	Ohm	Changed
» » » 故障安全模式			
故障安全模式	存储前一个有效		Initial value
» » » 人机界面			
小数点	2		Initial value

图 9-11　T3K 的输入输出参数设置

执行菜单命令“Option”→“settings”，可以在打开的对话框的“Communication”选项卡中设置刷新测量值的周期（Cycle），默认值为 1000ms。

6．T3K 的接线

用 DP 电缆连接 CPU 和 DP/PA 耦合器的 DP 接口，用 PA 电缆连接耦合器和 T3K 的 PA 接口，按图 9-12 连接 T3K 和阻值为 500Ω 的电位器。

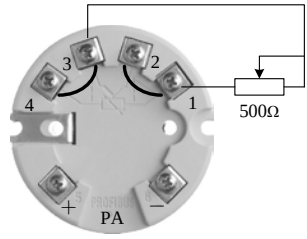


图 9-12　T3K 的接线

为了实现计算机与 DP 和 PA 的通信，设置 PG/PC 接口为 PROFIBUS，网络的传输速率设置为 45.45kbit/s，配置文件为 DP。用鼠标右键点击 PA 从站，执行出现的快捷菜单中的“SIMATIC PDM”→“下载到设备”命令，下载组态信息。

9.2　用 PDM 和 SFC 13 诊断 PROFIBUS-PA 设备的故障

1．诊断程序

用 STEP 7 打开随书光盘的文件夹“\Project\PA”中的项目“PA”。实验装置用 T3K 来测量电阻的阻值，外接的电位器的阻值过大和过小时，都会出现故障信息。实验时发现，故障

出现和故障消失时，都会调用 OB82 和 OB86。

在 OB82 和 OB86 中，用程序记录中断的次数，调用 SFC 20 保存组织块的局部数据，调用 SFC 13 读取诊断数据。下面是 OB82 中的程序，OB86 中的程序基本上相同：

```
程序段 1：记录执行 OB82 的次数
          L     MW    52
          +     1
          T     MW    52
程序段 2：调用 SFC 20 保存 OB 的局部变量
          CALL  "BLKMOV"
            SRCBLK  :=P#L 0.0 BYTE 20
            RET_VAL :=LW20
            DSTBLK  :=DB2.ARY                    //将局部变量保存在数据块的数组中
程序段 2：调用 SFC 13 读取从站的诊断数据
          L     B#16#39
          L     #OB82_EV_CLASS                   //从站故障刚发生时为 B#16#39
          ==I
          JC    m002                             //从站故障刚发生则跳转
m001:     CALL  "DPNRM_DG"                       //从站故障刚结束时调用 SFC 13
            REQ       :=TRUE                     //为 1 时请求读取DP从站的诊断数据
            LADDR     :=W#16#7FE                 //从站的诊断地址
            RET_VAL   :=MW52                     //错误代码，没有故障则存放实际传送的数据字节数
            RECORD    :=P#DB82.DBX64.0 BYTE 64   //存放诊断数据的地址区
            BUSY      :=M50.1                    //为 1 表示读取过程未完成
            A     M     50.1
            JC    m001                           //读取过程未完成则跳转
            BEU                                  //无条件结束块调用
m002:     CALL  "DPNRM_DG"                       //从站故障刚发生时调用 SFC 13
            REQ       :=TRUE                     //为 1 时请求读取DP从站的诊断数据
            LADDR     :=W#16#7FE                 //从站的诊断地址
            RET_VAL   :=MW54                     //错误代码，没有故障则存放实际传送的数据字节数
            RECORD    :=P#DB82.DBX0.0 BYTE 64    //存放诊断数据的地址区
            BUSY      :=M50.2                    //为 1 表示读取过程未完成
          A     M     50.2
          JC    m002                             //读取过程未完成则跳转
```

2．用 Lifelist 监控 T3K

用 DP 电缆连接 CPU 和 DP/PA 耦合器的 DP 接口，用 PA 电缆连接耦合器和 T3K 的 PA 接口，按图 9-12 连接 T3K 和阻值为 500Ω 的电位器。接通 PLC 和 DP/PA 耦合器的电源。

安装 PDM 后，桌面上出现 Lifelist 的图标  ，双击该图标，在出现的对话框（见图 9-13）的 Communication（通信）区中，用单选框选中“PROFIBUS”，计算机默认的站地址为 0。用复选框选中“Scan immediately after Start”（启动后立即扫描）、“Scan cyclically（周期性扫描）”和“With Diagnostics（使用诊断）”。其他均为默认的设置。

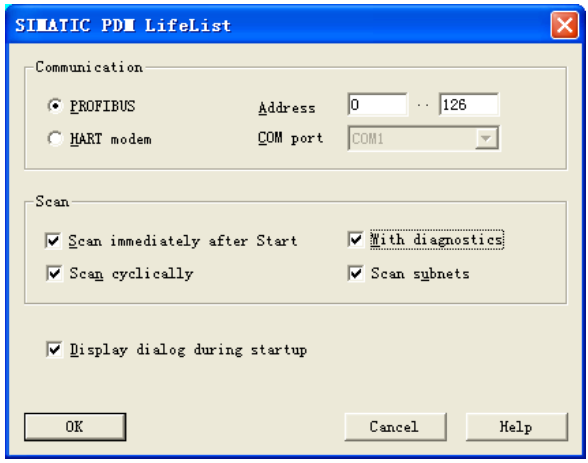


图 9-13　设置 PDM 的参数

点击“OK”按钮，打开 PDM LifeList 视图，开始自动扫描 DP 和 PA 网络上的站点。图 9-14 给出了扫描的结果。其中的 0 号站是编程用的计算机，2 号站是 DP 主站 CPU 315-2DP，18 号站是 PA 从站。系统正常运行时，DP/PA 链接器上的 DP 和 ON LED 亮，PA LED 闪烁。

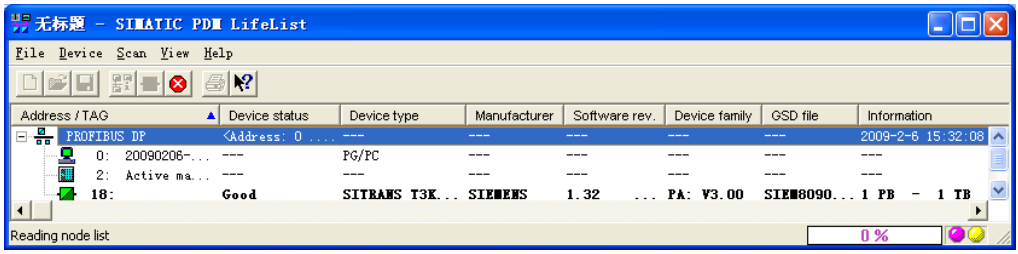


图 9-14　PDM Lifelist 扫描到的 DP、PA 站点

选中 PA 从站，执行菜单命令“Device”→“Download to devices”，出现下载对话框，将组态参数下载到该从站。

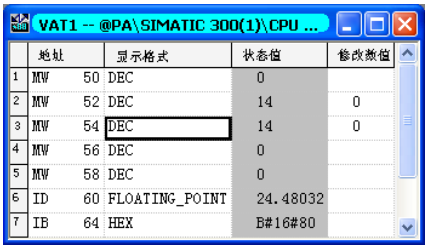
T3K 的输出数据（即 CPU 的输入）在 HW config 中被组态为 IB60～IB64（见图 9-9）。前 4 个字节是浮点数格式的测量值，第 5 个字节是被称为质量指示器（quality indicator）的状态字节。随书光盘中的用户手册《SITRANS T3K PA》的 66～68 页给出了各种情况的状态代码。

3．正常范围内的测量值和状态字节

在变量表中分别用 MW 52 和 MW54 记录 CPU 调用 OB82 和 OB86 的次数（见图 9-15）。用浮点数格式的 ID60 显示电阻的测量值，单位为Ω。IB64 是 T3K 的状态字节，16#80 表示测量值在正常范围，没有故障。

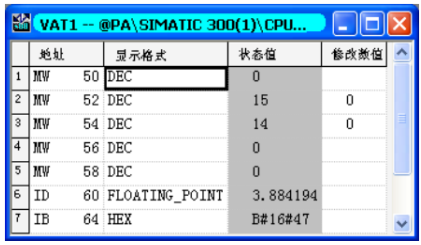
4．测量值超下限的故障信息

调节电位器，减小它的阻值。电阻测量值小于某个值时，CPU 的 SF LED 亮。变量表中的状态字节 IB64 的值为 16#47（见图 9-16）。查手册可知，16#47 表示新的测量值是“坏”的值，变送器提供的是最后的有效值。



	地址		显示格式	状态值	修改数值
1	MW	50	DEC	0	
2	MW	52	DEC	14	0
3	MW	54	DEC	14	0
4	MW	56	DEC	0	
5	MW	58	DEC	0	
6	ID	60	FLOATING_POINT	24.48032	
7	IB	64	HEX	B#16#80	

图 9-15　测量值在正常范围的变量表



	地址		显示格式	状态值	修改数值
1	MW	50	DEC	0	
2	MW	52	DEC	15	0
3	MW	54	DEC	14	0
4	MW	56	DEC	0	
5	MW	58	DEC	0	
6	ID	60	FLOATING_POINT	3.884194	
7	IB	64	HEX	B#16#47	

图 9-16　测量值超下限的变量表

与此同时，PDM LifeList 中 18 号站最左边的图标变为有故障的符号（红色的扳手，见图 9-17）。将光标放在 18 号站所在的行，出现黄色背景的故障信息窗口。其中的信息的意义如下：“维护报警：通道 1 的传感器或传感器连接的故障为短路，错误可能发生在传感器、接线或变送器”。

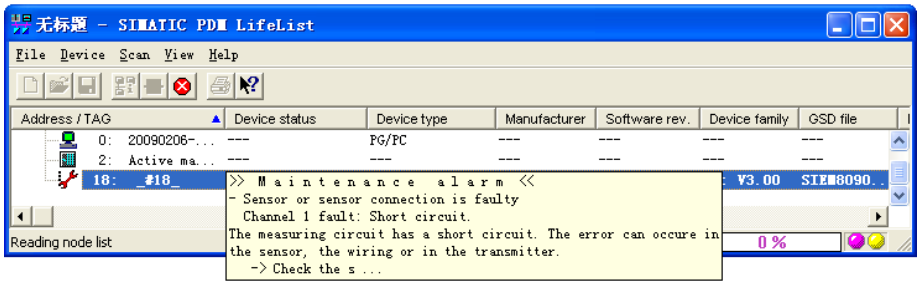


图 9-17　短路故障时的 PDM Lifelist

5．测量值超下限时 OB 的局部变量

图 9-18 是 OB82 的局部变量的前 12B。主要变量的意义如下：

- DBB0 的 16#39 表示进入事件（事件发生）。
- DBB5 的 16#54 表示输入，DBW6 的 16#07FE（2046）是从站的诊断地址。
- DBB8 的 16#05 表示模块出现外部故障。
- DBB9 的 16#63 表示来自替换值的诊断中断，有用户信息。

图 9-19 是故障消失时 OB86 的局部变量的前 12B。主要变量的意义如下：

- DBB0 的 16#38 表示离开事件（事件消失）。
- DBW6 的 16#07FF（2047）是主站的诊断地址，DBW8 的 16#07FE是从站的诊断地址。
- DBB10 的 16#01 是DP主站系统的编号，DBB11 的 16#12（18）是从站的站地址。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C55407FE
8.0	ARY[2]	DWORD	DW#16#0	DW#16#05630000

图 9-18　OB82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#38C41A56
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C05407FF
8.0	ARY[2]	DWORD	DW#16#0	DW#16#07FE0112

图 9-19　OB86 的局部变量

6．测量值超下限时 SFC 13 读取的诊断信息

在诊断事件出现和消失时，CPU 各调用一次 OB82 和 OB86。在 OB82 和 OB86 中，调用 SFC 13 读取 T3K 的诊断信息。OB82 和 OB86 读取的诊断数据基本上相同，OB82 读取的诊断数据见图 9-20 和图 9-21。SITRANS T3K PA 的用户手册的 1.3.4 节给出了诊断消息的结构

和意义。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#080C0002
4.0	ARY[1]	DWORD	DW#16#0	DW#16#809008FE
8.0	ARY[2]	DWORD	DW#16#0	DW#16#01012000

图 9-20　故障出现时 DB 82 中的故障信息

64.0	ARY[16]	DWORD	DW#16#0	DW#16#000C0002
68.0	ARY[17]	DWORD	DW#16#0	DW#16#809008FE
72.0	ARY[18]	DWORD	DW#16#0	DW#16#01020000

图 9-21　故障消失时 DB 82 中的故障信息

诊断数据的前 3 个字节是站状态 1～3，站状态 3（DBB2）为 0。主要变量的意义如下：

- DBB0（站状态 1）为 16#08，表示有外部诊断信息。
- DBB1（站状态 2）为 16#0C，表示已启用该DP从站的响应监视器。
- DBB3 为 16#02，是DP主站地址。
- DBW4 为 16#8090，是制造商标识符。
- DBW6 为 16#08FE，分别是头部（Header）和状态类型。
- DBB8 的 16#01 是插槽号。
- DBB9 为 1 和为 2 分别表示事件出现和事件消失。
- DBB10 的 16#20 表示存储器错误。

7．测量值超上限的故障信息

调节电位器，增大它的阻值。电阻测量值超过组态的 375Ω 时，CPU 的 SF LED 亮。变量表中的状态字节 IB64 的值为 16#52（见图 9-22），表示测量值超过上限。

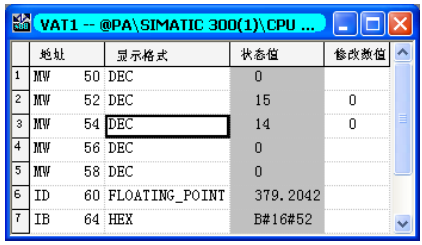


图 9-22　测量值超上限的变量表

与此同时，PDM LifeList 中 18 号从站最左边的图标变为有故障的符号。将光标放在 18 号从站所在的行，出现黄色背景的故障信息窗口（见图 9-23）。其中的信息的意义如下：“维护报警：传感器或传感器连接故障，通道 1 超出范围，测量值高于合理值，传感器可能损坏，也可能是与变送器的连接故障”。在本例中，故障的主要原因是电位器的额定值设置得过小。

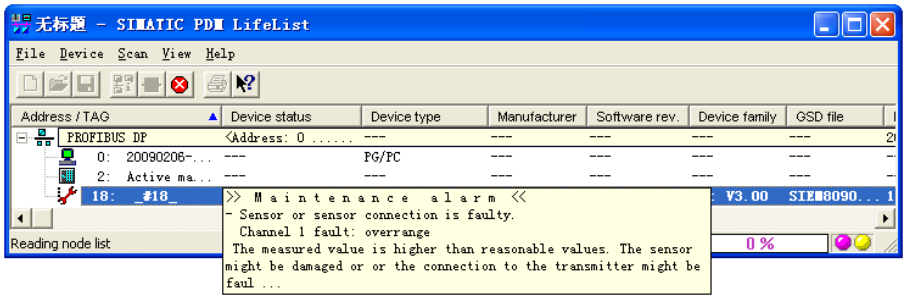


图 9-23　PDM Lifelist 的故障信息

选中 SIMATIC 管理器左边窗口的 300 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开“硬件诊断 - 快速查看”对话框。CPU 和 18 号 DP 从站上均有故障符号。

选中 DP 从站，点击“在线站点”按钮，在出现的诊断视图（见图 9-24）中的 CPU 和 18 号从站上，也有模块故障符号。

双击 PA 从站，打开 T3K 的模块信息对话框，“常规”选项卡中的模块状态为“模块故障（检测到诊断中断），外部出错”。

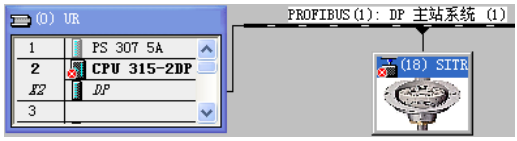


图 9-24　诊断视图

图 9-25 中的“从站的标准诊断”区中的英语部分的意义为“状态出现，测量出错”。点击“十六进制格式”按钮，出现的十六进制格式的诊断信息，与图 9-20 中用 SFC 13 读出的相同。在诊断事件出现和消失时，OB82 和 OB86 调用 SFC 13 读取的诊断信息，与测量值超下限时的基本上相同。

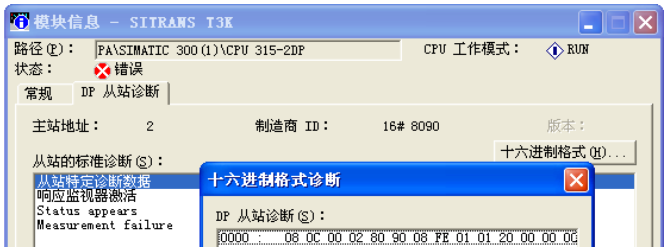


图 9-25　T3K 的模块信息对话框

9.3　练习题

1．PROFIBUS-PA 主要用于什么场合？

2．PROFIBUS-PA 采用什么传输技术？有什么特点？

3．PROFIBUS-PA 的耦合器有什么作用？是否要对它组态？耦合器直接连接到 DP 网络后，DP 网络的传输速率应设置为多大？

4．PROFIBUS-PA 的链接器有什么作用？使用链接器后，DP 网络的传输速率的上限是多少？PA 网络的传输速率是多少？

5. 在西门子网站搜索和下载某个 PA 设备的 GSD 文件，在 HW Config 中安装 GSD 文件。创建一个项目，将 PA 从站连接到 DP 网络。

6．PDM 软件有什么功能？

7．怎样用 PDM 来诊断 PA 从站的故障？

8．怎样用 SFC 13 来诊断 PA 从站的故障？

第 10 章　工业以太网

10.1　工业以太网

10.1.1　工业以太网概述

1．现场总线存在的问题

现有的现场总线标准过多，不同的现场总线不能兼容，不能实现透明的信息互访和信息的无缝集成。现场总线（不包括基于以太网的现场总线）是专用的网络，不像以太网那样得到了极其广泛的应用。因此硬件费用较高，传输速率较低，支持的应用有限，不能与互联网集成。

2．以太网的优点

以太网的市场占有率高达 80%，它具有以下优点：

1）工业以太网采用 TCP/IP，可以通过以太网将自动化系统连接到企业内部互联网（Intranet）、外部互联网（Extranet）和因特网（Internet）。不需要额外的硬件设备，就可以实现管理网络与控制网络的数据共享，即实现“管控一体化”。不需要专用的软件，可以用 IE 浏览器访问控制终端的数据。通过交换技术可以提供实际上没有限制的通信性能。

2）以太网的灵活性好，现有的设备可以不受影响地扩展。可以采用冗余的网络拓扑结构，可靠性高。通过有线电话网和无线电话网，可以用以太网实现远程数据交换。

近年来，为了满足实时性应用的需要，各大公司和标准化组织纷纷提出了各种提升工业以太网实时性的解决方案，从而产生了实时以太网。2007 年出版的 IEC 61158 第 4 版的 20 种现场总线中，实时以太网占了 10 种（见 1.1.3 节），可见，实时以太网已成为现场总线发展的主要方向。

3．传统以太网存在的问题

以太网采用载波侦听多路访问/冲突检测（CSMA/CD）的机制，两个工作站发生冲突时，必须延迟一定的时间后重发报文。发生堵塞时，有的报文可能长时间发送不出去，造成了通信时间的不确定性。所以传统的以太网一般不能用于工业自动化控制，但是可以用于实时性要求不高的场合。商用以太网一般用于办公室环境，不能用于恶劣的工业现场环境。

4．SIMATIC 工业以太网的特点

工业以太网通信用于管理层和车间层的控制器之间或控制器与 PC 之间的通信，一般数据量较大，传输距离较远，传输速率快，可以适应环境恶劣和抗干扰要求高的工业场合。工业以太网有以下特点：

1）与 IEEE802.3 及 IEEE802.3u 兼容，产品的设计制造充分考虑到工业应用的需要。

2）抗干扰能力强，可以用于严酷的工业现场环境。

3）采用标准导轨安装，容易安装和更换。能方便地组成各种网络拓扑结构。

4）10Mbit/s/100 Mbit/s 自适应传输速率，最多 1024 个网络节点，网络最大范围为 150km。

5）可用 DC 24V 冗余供电，实现冗余连接。

6）故障自动恢复，网络出现故障时（例如断线或交换机故障）最大网络重构时间小于 0.3s。

7）快速的网络故障定位与诊断。

8）支持虚拟局域网技术，将网络划分为几个虚拟的子网，可以有效地减轻网络负荷。

10.1.2 工业以太网的通信介质与网络部件

1．TP 电缆与 RJ-45 连接器

西门子的工业以太网可以采用双绞线、光纤和无线方式进行通信。

TP Cord 电缆是 8 芯的屏蔽双绞线，连接到 RJ-45 连接器有两种线序（见表 10-1）。RJ-45 电缆有交叉连接和直接连接这两种连接方式。

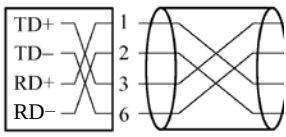
表 10-1 TP Cord 电缆与 RJ-45 连接器的线序

连接方式	RJ 45 的引脚号与导线的颜色							
	1	2	3	4	5	6	7	8
T568A	绿白	绿	橙白	蓝	蓝白	橙	棕白	棕
T568B	橙白	橙	绿白	蓝	蓝白	绿	棕白	棕

电缆两端的 RJ-45 连接器采用不同的线序为交叉连接（见图 10-1），交叉连接电缆用于两台设备（例如 PC 和 PLC）的以太网接口之间的直接连接。

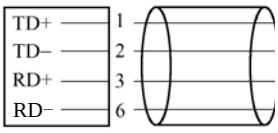
电缆两端的 RJ-45 采用相同的线序（一般是 T568B）为直通连接（见图 10-2），直通连接电缆用于 PC、PLC 等设备与交换机（或集线器）之间的连接。

西门子交换机采用自适应技术，可以自动检测线序，连接西门子交换机时可以采用上述的任意一种连接方式。



RD-

图 10-1 交叉连接



RD-

图 10-2 直通连接

2．快速连接双绞线

快速连接双绞线（Fast Connection Twist Pair）FC TP 是一种 4 芯电缆。使用 FC 剥线工具，一次就可以剥去电缆外包层和编织的屏蔽层，连接长度可达 100m。

通过导线和 IE FC RJ-45 接头上的透明接点盖的彩色标记，可以避免接线错误（见图 10-3）。将双绞线按接头上标记的颜色插入连接孔中，可以快速、方便地将数据终端设备（DTE）连接到以太网上。使用 4 芯（2×2）电缆，传输速率最大 100 Mbit/s。主干网使用 8 芯（4×2）电缆，传输速率可达 1000 Mbit/s。

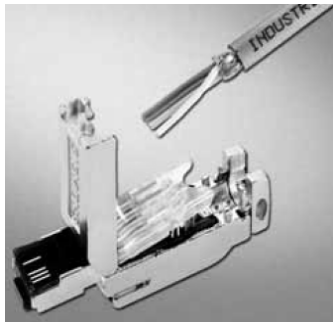


图 10-3 快速连接电缆与 TP RJ45 接头

3．工业双绞线 ITP

使用预装有 9 针或 15 针 Sub-D 插头的工业双绞线（Industry Twist Pair，ITP）标准电缆，可以连接某些通信处理器（CP）的 ITP 接口，进行牢固的连接，

适合于现场恶劣的环境，最长传输距离为 100m。ITP 电缆已逐渐被 FC TP 电缆代替。

4．光纤

光纤（FOC）通过光学频率范围内的电磁波，沿光缆无辐射传输数据，不受外部电磁场的干扰，没有接地问题，重量轻、容易安装。光缆的传导芯是一种低衰减的光学透明材料。即使线缆被弯曲，光束也能在芯体和周围材料之间通过完全反射传输，光缆包裹在一层保护套内。有两种不同类型的光缆，标准玻璃光缆可以在室内和室外使用；拖曳式玻璃光缆用于室内或室外以及需要移动的应用场合。

5．中继器和集线器

中继器又称为转发器，用来增加网络的长度。中继器仅工作在物理层，对同一协议的相同或不相同的传输介质之间的信号进行中继放大和整形。共享式集线器（Hub）是多端口的中继器，仅工作在物理层（见图 10-4）。它们将接收到的信号进行整形和中继，不加区别地广播输出，传送给所有连接到中继器或集线器的站点。它们不对接收到的报文进行过滤和负载隔离，不会干扰通信，不能解决以太网的冲突问题。它们只能在一个冲突域内使用。

6．交换机和网桥

交换机是工作在物理层和数据链路层的智能设备（见图 10-5）。工业以太网采用星形网络拓扑结构，用交换机将网络划分为若干个网段，交换机之间通过主干网络进行连接。互连不同体系的局域网的交换机称为网桥，例如 PROFIBUS 的 DP/PA 耦合器就是一种网桥。

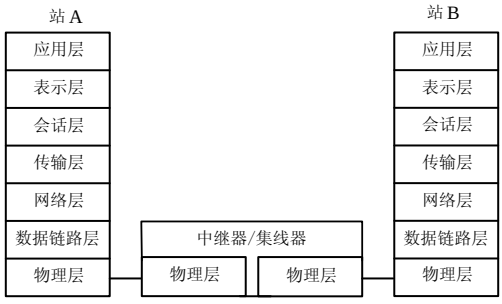


图 10-4　集线器

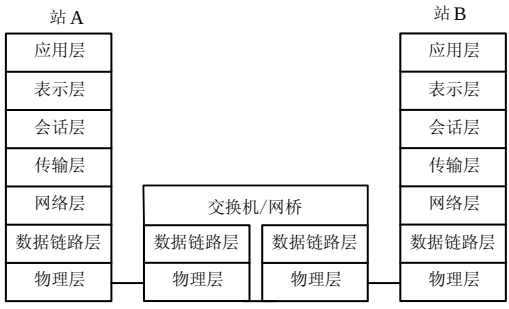


图 10-5　交换机

交换机和网桥可以对报文进行存储、过滤和转发，有一定的自学习能力，自动记录端口所有设备的 MAC 地址。在一个完整的交换网络中，整个网络只有交换机和通信节点，没有集线器。节点之间的数据通过交换机转发。单个站出现故障时，仍然可以进行数据交换。

7．路由器

路由器是在网络层存储转发数据帧的设备（见图 10-6），支持不同的网络协议，在进行异构网络的互连时，实现网络协议的转换。它适用于大规模网络和复杂的拓扑结构，路由器最重要的功能是实现路由选择，可以实现负载共享和最优路径，提高安全性，隔离不需要的通信量，节约局域网的频宽，减少主机的处理工作量。

8．网关

网关是不同协议之间的转换设备（见图 10-7），网关设备需要理解双方协议的全部含义，并对报文进行转换。网关可以提供防火墙和代理服务器等安全机制。网关包含用于连接不同网络的硬件和软件，可以对 OSI 参考模型所有 7 层进行完全的协议转换。

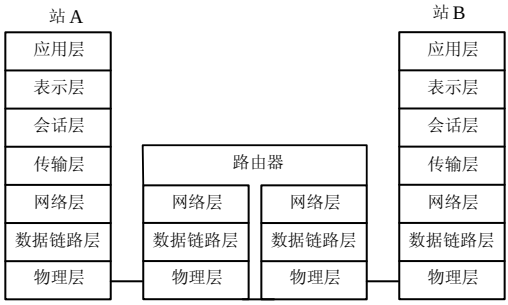


图 10-6　路由器

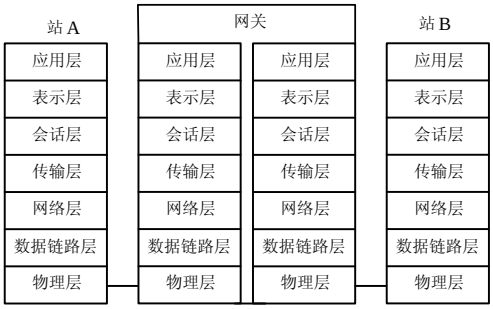


图 10-7　网关

9．工业以太网的构成

典型的工业以太网由以下 4 类网络器件组成：

1）连接部件包括 FC 快速连接插座，电气链接模块（ELM），电气交换模块（ESM），光纤交换模块（OSM）和光纤电气转换模块（MC TP11）。

2）通信媒体可以采用普通双绞线、快速连接双绞线、工业屏蔽双绞线和光纤。

3）CPU 集成的 PN 接口和工业以太网通信处理器用于将 PLC 连接到工业以太网。

4）PG/PC 的工业以太网通信处理器用于将 PG/PC 连接到工业以太网。

工业以太网的网络结构、组网方法、网络元件的参数和选型见随书光盘文件夹“\资料与手册\产品样本”中的有关样本。

10.1.3　工业以太网的交换技术

1．交换技术

实时以太网通过采用交换机和全双工通信，解决了 CSMA/CD 机制带来的冲突问题。

在共享局域网（LAN）中，所有站点共享网络性能和数据传输带宽，所有的数据包都经过所有的网段，同一时刻只能传送一个报文。

交换机可以同时在其端口之间建立多个连接。这些连接是根据数据流量的需要，动态和暂时建立的。

在交换式局域网中，用交换模块将一个网络分为若干个网段，在每个网段中可以分别同时传输一个报文，每个网段都能达到网络的整体性能和数据传输速率。本地数据通信在本网段进行，只有指定的数据包可以超出本地网段的范围。利用交换技术易于扩展网络的规模，并且可以限制子网内的错误在整个网络上的传输。

交换机首先对接收的输入信号进行缓存，再进行地址比较，根据网络的占用状况适时发送到正确的端口。每个网段内节点间数据的传输只限于在本地网段内进行，不需经过主干网，使本地数据传输不占其他网段的带宽，从而降低了各网段和主干网的网络负荷。交换机对网络上传输的数据包进行检测，不再传送错误帧和错误报告帧。交换机可以同时处理不同网段之间的多个数据包，在通信设备之间建立多个动态连接，实现并行数据交换。

PROFINET 使用点对点连接的交换式以太网，其中每个设备直接与一个（且只有一个）其他设备相连。

2．全双工模式

全双工交换式以太网的一对线用来发送数据，另一对线用来接收数据，这种以太网消除了冲

突的可能，使以太网的通信确定性和实时性大大提高，减少了 CSMA/CD 机制造成的冲突和大量的无关的通信量，将局域网的范围扩展到 150km，每个网络节点甚至可以独享整个介质带宽。

使用具有全双工功能的交换机，在两个节点之间可以同时发送和接收数据，全双工快速以太网的数据传输速率增加到 200 Mbit/s。

并不需要在网络的所有地方都使用全双工，只需要在某些特定的节点之间建立少量的全双工连接，例如，服务器和集线器/交换机之间的连接，或者交换机之间的连接。

3．电气交换模块与光纤交换模块

电气交换模块（ESM）与光纤交换模块（OSM）用来构建 10 Mbit/s/100 Mbit/s 交换网络，能低成本、高效率地在现场建成具有交换功能的线性结构或星形结构的工业以太网。

可以将网络划分为若干个部分或网段，并将各网段连接到 ESM 或 OSM 上，这样可以分散网络的负担，实现负载解耦，改善网络的性能。

利用 ESM 或 OSM 的网络冗余管理器，可以构建环形冗余工业以太网。

通过 ESM 可以方便地构建适用于车间的网络拓朴结构，包括线性结构和星形结构。级联深度和网络规模仅受信号传输时间的限制，使用 ESM 可以使网络总体规模达 5km，使用 OSM 时网络长度可达 150 km。

4．自适应与自协商功能

具有自适应功能的网络站点（终端设备和网络部件）能自动检测出信号传输速率（10 Mbit/s 或 100 Mbit/s），自适应功能可以实现所有以太网部件之间的无缝互操作性。

自协商是高速以太网的配置协议，该协议使有关站点在数据传输开始之前就能协商，以确定它们之间的数据传输速率和工作方式，例如是全双工或半双工。

5．冗余网络

冗余软件包 S7-REDCONNECT 用来将 PC 链接到高可靠性的 SIMATIC S7-H 冗余系统，可以避免设备停机。万一出现子系统故障或断线，系统交换模块会切换到双总线，或者切换到冗余环的后备系统或后备网络，以保证网络的正常通信。

6．SIMATIC NET 的快速重新配置

网络发生故障后，应尽快对网络进行重构。重新配置的时间对工业应用是至关重要的，否则网络上连接的终端设备将会断开连接，从而引起工厂生产过程的失控或紧急停机。SIMATIC NET 采用了专门为此开发的冗余控制程序，对于有 50 个交换模块（OSM/ESM）的 100 Mbit/s 环形网络，重新配置的时间不超过 0.3s。

10.1.4 工业以太网的通信处理器与带 PN 接口的 CPU

1．用于 PC 的工业以太网通信处理器

工业以太网通信处理器支持 ISO、TCP/IP、UDP、S7、PG/OP 通信、OPC 通信和 SNMP 诊断。

1）CP 1616 是 PCI 插槽的以太网通信处理器，可作 PROFINET 控制器或 PROFINET IO 设备。它集成有 4 端口交换机，有 PROFINET 的 IRT（同步实时）功能，可以用于运动控制领域对时间要求严格的同步闭环控制。

2）CP 1613 A2 是 PCI 插槽的以太网通信处理器，通信速率为 10Mbit/s 或 100 Mbit/s，有一个 15 针 ITP 接口和一个 RJ-45 接口。该 CP 可以用于冗余通信，可以实现时钟的网络同步。

3）CP 1623 是 PCI 插槽的以太网通信处理器，它集成了 RJ-45 接口的二端口交换机，其他功能与 CP 1613 A2 相同，主要用于要求很高的场合。

2．S7-300/400 的工业以太网通信处理器

1）CP 343-1 是用于 S7-300 的全双工以太网通信处理器，通信速率为 10Mbit/s 或 100Mbit/s，有一个 RJ-45 插座。具有 Keep Alive（保持激活）连接控制功能，即可以为活动的或被动的通信伙伴所有 TCP 传输连接组态一个可调节的时间（见图 10-19）。通信服务包括 TCP/IP、UDP、S7 通信、S5 兼容通信。通过 S7 路由，可以在多个网络之间进行 PG/OP 通信和远程编程。

2）CP 343-1 IT 通信处理器除了具有 CP 343-1 的功能外，还可以实现信息技术（Information technology，IT）通信，包括发送电子邮件，HTTP 通信支持通过 Web 浏览器访问过程数据；支持 FTP（文件传输协议）客户机通信，可以通过 FTP 服务器访问数据块；有基于 IP 地址的访问保护，有时钟同步功能。

3）CP 343-1 PN 有 15 针的 ITP/AUI 接口和 RJ-45 接口。除了具有 CP 343-1 的功能外，还有 PROFINET 控制器和 PROFINET CBA（基于组件的自动化）功能。

4）CP 343-1 Advanced 同时具有 CP 343-1 IT 和 CP 343-1 PN 的功能。

5）CP 443-1 的功能与 CP 343-1 基本上相同，此外有 15 针的 ITP/AUI 接口和 RJ 45 接口。可以实现冗余通信，有时钟同步功能。

6）CP 443-1 Advanced 的功能与 CP 343-1 Advanced 基本上相同，此外集成了 4 端口的 RJ-45 交换机。

7）CP 444 将 S7-400 连接到工业以太网，根据 MAP 3.0（制造自动化协议）标准提供 MMS（制造业信息规范）服务，包括环境管理（启动、停止和紧急退出）、VMD（设备监控）和变量访问服务。

3．带 PROFINET 接口的 CPU

1）CPU 315/317-2PN/DP 集成有一个 MPI/PROFIBUS-DP 接口和一个 PROFINET 接口，可以作 PROFINET I/O 控制器，在 PROFINET 上实现基于组件的自动化（CBA）。可以作 CBA 的 PROFIBUS-DP 智能设备的 PROFINET 代理服务器。

SIPLUS CPU 315/SIPLUS 317-2PN/DP 是宽温型（环境温度-25～+60℃）。

2）CPU 319-3PN/DP 是具有智能技术/运动控制功能的 CPU，是 S7-300 系列性能最高的 CPU。它集成了 1 个 MPI/PROFIBUS-DP 接口，1 个 PROFIBUS-DP 接口和 1 个 PROFINET 接口。它提供 PROFIBUS 接口的时钟同步，可以连接 256 个 I/O 设备。

3）CPU 414-3PN/ CPU 416-3PN 的 3 个通信接口与 CPU 319-3PN/DP 的相同。PROFINET 接口带两个端口，可以作交换机。可以用 IF 964-DP 接口子模块连接到 PROFIBUS-DP 主站系统。SIPLUS CPU 416-3PN 是宽温型，CPU 416F-3PN 用于故障安全自动化系统。

10.1.5 工业以太网的交换机

如果在系统的快速性和冗余控制方面没有什么要求，现场环境较好，工业以太网可以使用普通的交换机和普通的网卡，反之则应选用西门子公司的交换机和网卡。

1．SCALANCE X 系列交换机

1）SCALANCE X005 是非网络管理型交换机，有 5 个 RJ-45 接口，价格低廉，可以构建

具有交换功能的小型星形结构或线性结构。

2）SCALANCE X-100 系列是非网络管理型交换机，带有冗余电源和信号触点，适合在设备附近使用。各种规格有不同点数的电气接口和光学接口，最多的有 24 个电气接口。

3）SCALANCE X-200 系列是网络管理型交换机，应用广泛。作为冗余管理器，与 SCALANCE X-400/X-200IRT 或 OSM/ESM 组合，可以实现环形冗余网。各种规格有不同点数的电气接口和光学接口。

4）SCALANCE X-200IRT 网络管理型交换机可以用于具有严格的实时要求（实时及同步实时）的网络，可以满足 PROFINET 的实时要求。

5）SCALANCE X-200 IRT PRO 除了具有 SCALANCE X-200 IRT 的功能外，防护等级达 IP65/67，可以安装在控制柜外面。

SCALANCE X-200 系列交换机可以通过 LED 进行设备诊断，可以使用信号触点、PROFINET、SNMP 和 Web 浏览器等方式对交换机进行远程管理和诊断。

6）SCALANCE X-300 是网络管理增强型千兆交换机，结合了 SCALANCE X-400（第 3 层没有路由功能）的固件功能与 SCALANCE X-200 的紧凑型设计。

7）SCALANCE X-400 是高性能模块化的千兆交换机，适用于高速的光学/电气线性、环形和星形拓扑结构（传输速率 10/100/1000 Mbit/s）。可以根据需要将介质模块和扩展模块插入交换机。使用扩展模块最多可以增加 8 个电气接口和 8 个光学接口。

2．冗余系统

冗余系统用于减少停机时间，系统中重要的自动化组件（例如 CPU、网络和 CP 等）都有备份。在双绞线电缆连接成的冗余环中（见图 10–8），SCALANCE X400 交换机作为冗余管理器。如果网络中的某个组件（例如电缆）发生故障，冗余管理器将使备用（冗余）的连接路径自动接管通信任务，连接不会中断。

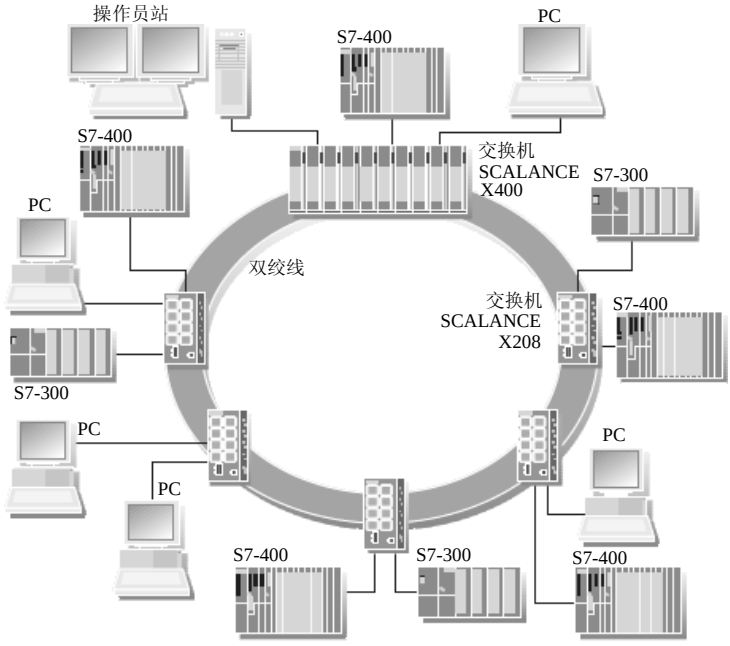


图 10–8　工业以太网中的电气环型冗余

10.1.6 以太网的地址

1．MAC 地址

在 OSI（开放系统互连）7 层网络协议参考模型中，第 2 层（数据链路层）由 MAC（Media Access Control，媒体访问控制）子层和 LLC（逻辑链路控制）子层组成。

MAC 地址也叫物理地址、硬件地址或链路地址。MAC 地址是识别 LAN（局域网）节点的标识，即以太网接口设备的物理地址。它通常由设备生产厂家写入 EEPROM 或闪存芯片，在传输数据时，用 MAC 地址标识发送和接收数据的主机的地址。在网络底层的物理传输过程中，通过 MAC 地址来识别主机。MAC 地址是 48 位二进制数，通常分为 6 段（6B），一般用十六进制数表示，例如 00-05-BA-CE-07-0C。其中的前 6 位十六进制数是网络硬件制造商的编号，它由 IEEE（电气与电子工程师协会）分配，后 6 位十六进制数代表该制造商制造的某个网络产品（例如网卡）的系列号。形象地说，MAC 地址就像我们的身份证号码，具有全球唯一性。

在 Windows XP 中，执行菜单命令“开始”→“运行”，在出现的“运行”对话框中输入“CMD”后按〈Enter〉键，在出现的 DOS 窗口中输入命令行“ipconfig /all”后按〈Enter〉键，将显示出计算机网卡的物理地址（即 MAC 地址）、IP 地址和子网掩码等。

MAC 地址是以太网包头的组成部分，以太网交换机根据以太网包头中的 MAC 源地址和 MAC 目的地址实现包的交换和传递。如果使用 ISO 协议，必须输入模块的 MAC 地址。

可以通过下载组态信息，修改 SIMATIC 以太网 CP 模块的 MAC 地址。

2．IP 地址

为了使信息能在以太网上准确快捷地传送到目的地，连接到以太网的每台计算机必须拥有一个唯一的地址。为每台计算机指定的地址称为 IP 地址。

IP 地址由 32 位二进制数（4B）组成，是 Internet（网际）协议地址，每个 Internet 包必须有 IP 地址，每个 Internet 服务提供商（ISP）必须向有关组织申请一组 IP 地址，一般是动态分配给其用户，用户也可以根据接入方式向 ISP 申请一个 IP 地址。

IP 地址通常用十进制数表示，用小数点分隔，例如 192.168.0.117。

同一个 IP 地址可以使用具有不同 MAC 地址的网卡，更换网卡后可以使用原来的 IP 地址。

3．子网掩码

子网掩码（Subnet mask）是一个 32 位地址，用于将网络划分为一些小的子网。

IP 地址由子网地址和子网内的节点地址组成，子网掩码用于将这两个地址分开。由子网掩码确定的两个 IP 地址段分别用于寻址子网 IP 和节点 IP。二进制的子网掩码的高位应是连续的 1，低位应是连续的 0。以子网掩码 255.255.255.0 为例，其高 24 位二进制数为 1，表示 IP 地址中的网络标识（类似于长途电话的地区号）为 24 位；低 8 位二进制数为 0，表示子网内节点的标识（类似于长途电话的电话号）为 8 位。IP 地址和子网掩码进行“与”逻辑运算，得到子网地址。IP 地址和子网掩码取反后得到的 0.0.0.255 进行“与”逻辑运算，得到节点地址。

10.1.7 工业控制网络的信息安全

现代的工业控制网络已经越来越多地被连接到办公网络和企业内部互联网（Intranet），无线局域网的使用也日益增多，加上远程维护等新技术的采用，工业通信网络与信息技术（IT）环境的交互越来越多。由于 PLC 等现场控制设备通过 Web 服务器和电子邮件等与互联网上的设备交换信息，办公和 IT 环境中常见的威胁，例如黑客程序、病毒、蠕虫和木马程序等，也会威胁到控制系统的安全。

用于办公环境的数据安全解决方案不能简单地照搬到工业应用场合，西门子提供了适用于工业自动化工程的安全解决方案，以防止敏感系统和生产网络被恶意操纵和破坏。

以下是在工业环境中防止操纵和丢失数据的最重要的预防措施：

1）通过虚拟专用网（VPN）来过滤和检查数据通信。虚拟专用网用于在办公网络（例如互联网）中交换私有数据。最常用的 VPN 技术是 IPsec。IPsec 是一个协议集，用于保证在网络层使用 IP 的信息的安全性。

2）在受保护的自动化单元中进行分段，用安全模块来保护网络节点，一组受保护的设备构成一个受保护的自动化单元。只有同一类型的安全模块或它们保护的设备之间才能互相交换数据。

3）通过对节点进行身份验证，安全模块可以通过安全（加密）通道相互识别。未经授权不能访问受保护的网段。

4）通过对数据通信加密来确保数据的机密性。为每个安全模块提供一个包含密钥的 VPN 证书。

5）在 PROFINET 和控制设备之间设置 Scalance S 安全模块（见图 10–9）。安全模块的防火墙用于保护 PLC 免受未经授权的访问。在验证通信伙伴身份的可靠性和发送数据的加密性方面，防火墙可以作 VPN 的替代或补充。

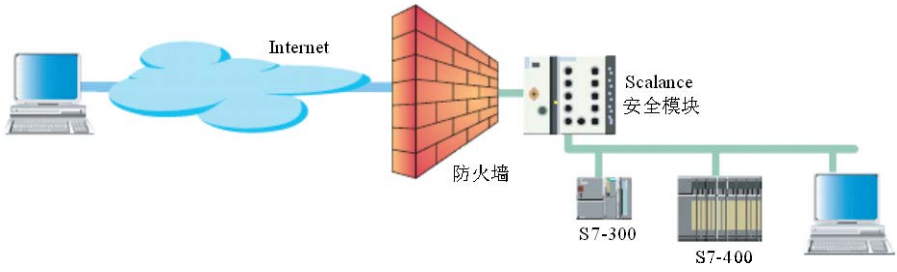


图 10–9　防火墙与安全模块

从逻辑上看，防火墙是分离器、限制器和分析器。从物理上看，防火墙由路由器、计算机和软件组成。防火墙可以过滤数据包，根据过滤表来禁用或启用通信连接。进入和离开通信、IP 地址、MAC 地址和通信协议（端口）都可以被过滤。

Scalance S 可以作 SOFTNET 的安全客户机，用于 PC 安全地访问受 Scalance S 保护的 PLC。即使没有专业的 IT 知识，也可以很简单地进行组态。只需创建和组态需要相互之间进行安全通信的安全模块或 Softnet 安全客户机。如果发生故障，无需编程设备就可以快速

更换安全模块。

10.1.8 IT 通信服务

SIMATIC 通信网络通过工业以太网将 IT 功能集成到控制系统。在办公环境中，电子邮件和 Web 浏览器得到了广泛应用。除了电话线和互联网外，以太网被用作主要的通信路径。通过 TCP/IP，这些通信媒体和路径也可以供 SIMATIC 控制设备使用。SIMATIC 设备支持下述 IT 服务。

1．FTP 服务

FTP（File Transfer Protocol，文件传输协议）通信用于不同操作系统的计算机之间程序控制的数据交换。IT-CP/Adv-CP (CP 443-1 Advanced 和 CP 343-1 Advanced)的 FTP 服务功能提供一种与下列 S7 设备交换文件的高效方式：

1）编程设备或 PC 与 S7-300/400 之间。

2）S7-200/300/400 设备之间。

3）S7 PLC 与过程控制计算机或 MES（制造执行系统）之间。

IT-CP/Adv-CP 既可以作 FTP 服务器，也可以作 FTP 客户机。为了用 FTP 传输数据，需要在 S7 站点的 CPU 中创建数据块（文件 DB）。

（1）IT-CP/Adv-CP 作 FTP 服务器

FTP 客户机（编程设备或 PC）可以通过 IT-CP/Adv-CP，使用 FTP 命令以文件形式与 S7 站的数据块交换数据。作为 FTP 服务器的 IT-CP/Adv-CP 将使用文件分配表来确定如何将 S7 站中用于文件传输的数据块映射到文件。通过文件分配表的信息，可以访问 S7 站的一个或多个 CPU 的数据块。

（2）IT-CP/Adv-CP 作 FTP 客户机

用户程序用专用的 FC（功能）发出 FTP 请求，要求与 FTP 服务器交换数据，然后由作为 FTP 客户机的 IT-CP/Adv-CP 执行这些请求。FTP 请求包含 FTP 服务器的 IP 地址、文件的存储位置、文件名以及访问信息等目标参数。

FTP 传输通过 FTP 连接进行。FTP 连接是在 STEP 7 的 NetPro 中组态的专用 TCP 连接。

为了管理作为 FTP 客户机和 FTP 服务器的 S7 站之间的 FTP 请求序列，IT-CP/Adv-CP 必须与该 S7 站的 CPU 建立连接。可以使用 STEP 7 中的连接组态，或在用户程序中调用 FB IP_CONFIG 来建立这种连接。

2．电子邮件服务

自动化系统可以使用 IT-CP/Adv-CP 的电子邮件功能，通过 SMTP（Simple Mail Transfer Protocol，简单邮件传输协议），在工业以太网上发送包含过程信息的电子邮件，发送邮件时可以带附件。IT-CP/Adv-CP 作电子邮件客户机，通过 SMTP 服务发送电子邮件，但是不能接收电子邮件。要发送 S7-300/400 CPU 用户程序中的电子邮件，需要调用 FC 5 AG_SEND 或 FC 50 AG_LSEND 轮询 SEND/RECEIVE 接口。

为了发送电子邮件，必须建立电子邮件连接，该连接定义了用 IT-CP/Adv-CP 发送电子邮

件的邮件服务器。完整的电子邮件在随机数据块中生成。

3．SNMP 服务

SNMP（简单网络管理协议）是以太网的一种开放的标准化网络管理协议。网络管理包括监视、控制和组态网络节点的所有功能。网络管理（例如错误记录）可以防止有 SNMP 功能的网络节点组成的网络发生故障，以确保网络的高质、高效。

网络管理产品（例如西门子公司的 SINEMA E 和 SNMP-OPC 服务器）可以在工业环境中对网络进行规划、控制和监视。SNMP 使用无线 UDP 传输协议。SNMP 管理器监视网络节点，而 SNMP 代理收集各网络节点中各种特定的网络信息，并以结构化形式将其置于管理信息库（Management Information Base，MIB）中。

管理站周期性轮询 MIB 信息，节点也可以通过所谓的陷阱向网络管理站报告某些状态。通过 SNMP，不但可以监视节点，而且可以监视用于控制设备的操作和指令，例如网络组件上端口的激活或取消激活。所有以太网设备都可以通过其 IP 地址和/或 SNMP 被检测到，并且可以传输到组态中。

对于有 IT 功能的 CP，用户可以用它提供的 HTML（超文本标记语言）页面，通过超文本传输协议（Hyper Text Transfer Protocol，HTTP）和 Web 浏览器，查询重要的系统数据。HTML 过程控制可以用于 PC 站和 S7-300/400 之间的通信。

10.2 用普通网卡实现计算机与 S7-300 的通信

10.2.1 使用 ISO 协议进行通信

普通网卡可以用 ISO 协议或 TCP/IP 与 PLC 的以太网 CP（通信处理器）通信。使用 ISO 协议的优点是不需要用其他通信接口（一般是 MPI 接口）对 CP 的以太网接口初始化。即使 CPU 中原来没有以太网 CP 的组态信息，或者更换了以太网 CP 模块，不用下载 CP 的组态信息，就可以实现 ISO 通信。只要 PLC 有以太网 CP，现场工程师就可以用计算机的普通网卡进行下载和监控操作。因此可以省掉计算机的 MPI 通信的 CP 卡、PC/MPI 适配器或 USB/MPI 适配器。某些低档的 CPU 没有这一功能。

1．硬件连接

作者做实验使用的是笔记本电脑，它有一个有线网卡和一个无线网卡。用一条交叉连接的 RJ-45 电缆（见图 10-1）连接 PLC 的以太网 CP 和计算机的普通网卡。也可以用两条直通连接的 RJ-45 电缆（见图 10-2）和交换机连接它们。

2．设置 PG/PC 接口

在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PG/PC 接口”，用出现的对话框（见图 10-10）中间的选择框，选中使用 ISO 协议的计算机集成的有线网卡。点击“确定”按钮，出现“访问路径已更改”的警告信息。点击“确定”按钮，退出“设置 PG/PC 接口”对话框后，ISO 协议才会生效。

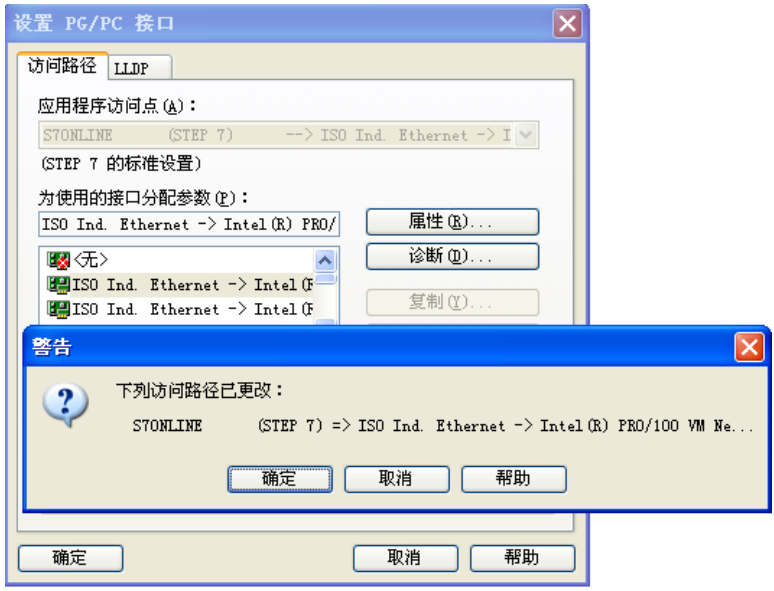


图 10-10　设置 PG/PC 接口

3．组态 CP 343-1

在 HW Config 中，双击 CP 343-1 所在的插槽，点击打开的 CP 属性对话框的“常规”选项卡（见图 10-11）中的“属性”按钮，选中出现的以太网接口属性对话框中的复选框“设置 MAC 地址/使用 ISO 协议”，在“MAC 地址”输入框输入 CP 模块上标出的 MAC 地址。

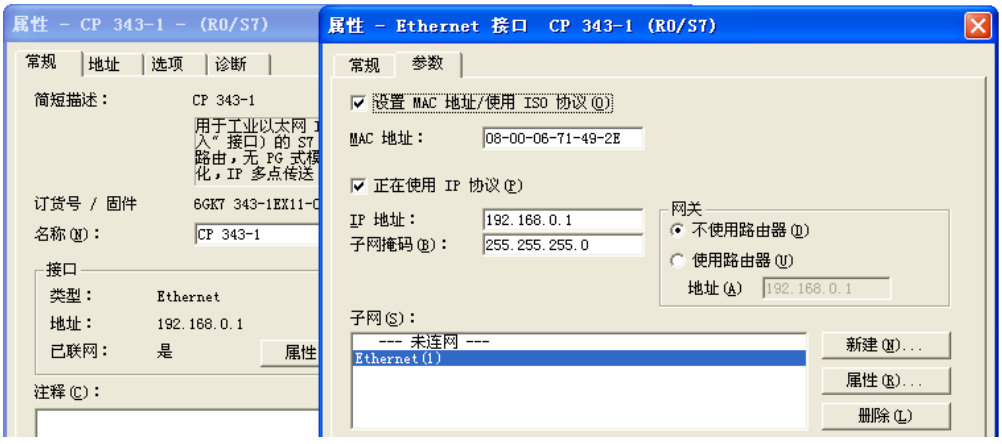


图 10-11　设置 CP 的以太网接口

4．用 ISO 协议下载硬件信息

点击 HW Config 工具栏中的  （下载）按钮，在出现的“选择目标模块”对话框（见图 10-12），选择通过 CP 343-1 下载。点击“确定”按钮，出现“选择节点地址”对话框。此时只能看到组态时设置的 MAC 地址。点击“显示”按钮，等待几秒钟后，在“可访问的节点”列表中出现读取的 CP 模块原有的 MAC 地址和模块的型号，“显示”按钮上的字符变为“更新”。点击“可访问的节点”中的 MAC 地址，它将出现在上面的表格中。点击“确定”

按钮，开始下载硬件组态信息。

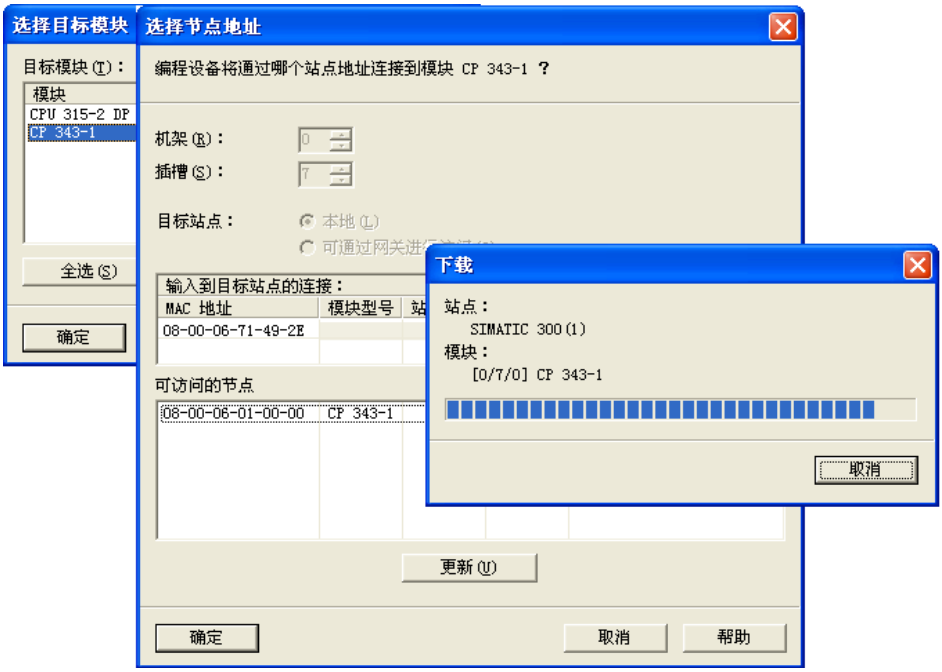


图 10-12　下载硬件组态信息

用 ISO 协议下载了 IP 地址后，就可以使用 TCP/IP 协议来通信。使用 CPU 和 CP 的路由器功能，可以用计算机访问其他网络上的站点（见 15.2.1 节）。

10.2.2　使用 TCP/IP 进行通信

1．设置计算机网卡的 IP 地址

计算机的网卡与 CP 的以太网接口的 IP 地址应在同一个网段内，它们应使用相同的子网掩码。一般采用默认的子网网段地址 192.168.0，和默认的子网掩码 255.255.255.0（见图 10-11）。

打开计算机的控制面板，双击其中的“网络连接”图标。在“网络连接”对话框（见图 10-13）中，用鼠标右键点击“本地连接”图标，执行出现的快捷菜单中的“属性”命令，打开“本地连接属性”对话框。选中“此连接使用下列项目”列表框中的“Internet 协议（TCP/IP）”，点击“属性”按钮，打开“Internet 协议（TCP/IP）属性”对话框。用单选框选中“使用下面的 IP 地址”，然后按上述的原则设置网卡的 IP 地址和子网掩码。计算机的 IP 地址的最后一个字节只要不与其他站点冲突就可以了。设置结束后，点击各级对话框中的“确定”按钮，最后关闭“网络连接”对话框。

2．设置 PG/PC 接口

在 SIMATIC 管理器中，执行菜单命令“选项”→“设置 PG/PC 接口”，用出现的对话框（见图 10-10）中间的选择框选中使用 TCP/IP（Auto）的计算机网卡。点击“确定”按钮，出现显示“访问路径已更改”的对话框。点击“确定”按钮，退出“设置 PG/PC 接口”对话框后，TCP/IP 会生效。

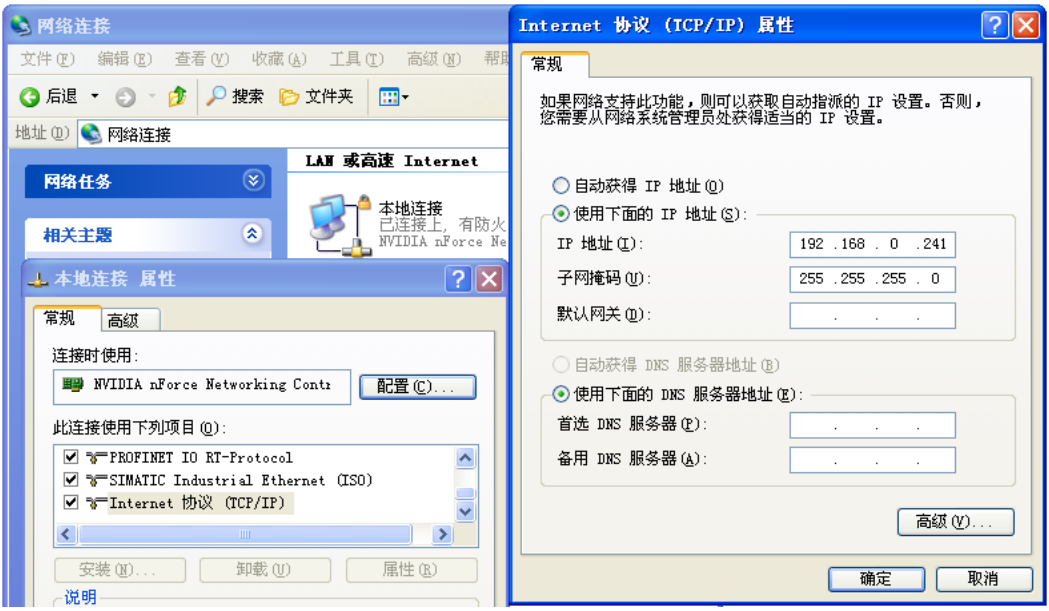


图 10-13　设置计算机网卡的 IP 地址

3．验证 TCP/IP 通信

用 MPI 接口或使用 ISO 协议的普通网卡将 IP 地址下载到 CPU 模块后，就可以进行 TCP/IP 通信了。点击 HW Config 工具栏上的（下载）按钮，在出现的“选择目标模块”对话框中，点击“确定”按钮，出现“选择节点地址”对话框（见图 10-14），列出了组态的目标站点的 IP 地址和 MAC 地址。点击“确定”按钮，开始下载硬件组态信息。

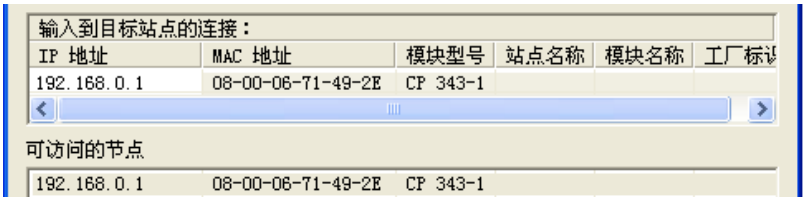


图 10-14　选择节点地址对话框

单击对话框中的“显示”按钮，经过几秒钟后，在“可访问的节点”列表中将会出现 CP 模块的 IP 地址、MAC 地址和模块的型号。如果已经下载了 CP 的 IP 地址，不用执行这一操作。

选中 SIMATIC 管理器中的 300 站点，执行菜单命令“编辑”→“对象属性”，在“接口”选项卡（见图 10-15）中，可以看到该站点所有通信接口的信息。

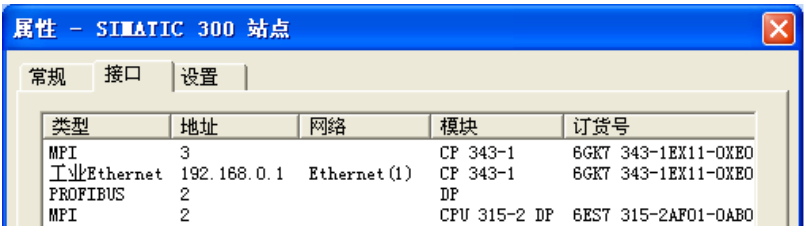


图 10-15　300 站点的通信接口

10.3 基于以太网的 S5 兼容通信

10.3.1 S5 兼容的通信服务

1．S5 兼容的通信概述

S5 兼容的通信服务包括 PROFIBUS 的 FDL，和以太网的 TCP/IP、ISO 传输、ISO-on-TCP 和 UDP，它们的组态和编程的方法基本上相同。TCP/IP、ISO 传输和 ISO-on-TCP 可以发送和接收 8KB（8192B）数据，UDP 可以发送和接收 2KB（2048B）数据。

需要在 STEP 7 中为 S5 兼容的通信组态静态连接。在站点启动时，连接被立即建立。

在 SIMATIC S7 中，调用功能 FC 5 AG_SEND 和 FC 6 AG_RECV 来实现 S5 兼容的通信。

2．网络协议

网络协议是网络上所有的设备（网络服务器、计算机、交换机、路由器、防火墙等）之间通信规则的集合，它定义了通信时信息必须采用的格式和这些格式的意义。大多数网络都采用分层的体系结构，每一层都建立在它的下一层之上，向它的上一层提供一定的服务，而把如何实现这一服务的细节对上一层加以屏蔽。一台设备的第 *n* 层与另一台设备的第 *n* 层进行通信的规则就是第 *n* 层协议。

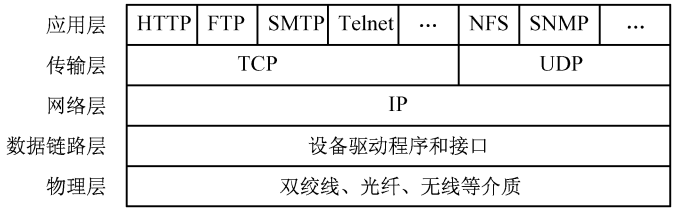


图 10-16　OSI 模型中的网络协议

在 OSI 参考模型各层中有许多协议，接收方和发送方同一层的协议必须一致，否则无法识别通信伙伴发出的信息。网络协议使网络上各种设备能相互交换信息。图 10-16 是 7 层 OSI 参考模型中的网络协议，IP 位于第 3 层，TCP 和 UDP 位于第 4 层，应用层是 OSI 模型中的第 7 层。应用层负责处理特定的应用程序细节，常用的应用协议有：

- HHTP：超文本传输协议。
- FTP：文件传输协议。
- SMTP：简单邮件传输协议。
- Telnet：远程登陆协议。
- NFS：网络文件系统。
- SNMP：简单网络管理协议。

3．TCP/IP 服务

TCP/IP 是“Transmission Control Protocol/Internet Protocol”的简写，中文译名为“传输控制协议/网际协议”。TCP/IP 规范了网络上所有的通信设备，尤其是一个主机与另一个主机之间的数据交换的格式，以及传送方式。TCP/IP 是互联网的基础协议，也是一种数据打包和寻址的标准方法。通过 TCP 连接和 SEND/RECEIVE 接口，以太网卡可以提供和几乎所有的终端（PC 或其他系统）相连的接口。

TCP/IP 服务可以用于 S7-300/400 与 PC 或非西门子公司系统的通信，将最多 8KB 的连续数据块从一个以太网节点传送到另一个以太网节点，数据的接收由通信伙伴确认。

（1）IP

IP（网际协议）是 OSI 参考模型第 3 层的协议，主要用于在整个网络中寻址，通过 32 位 IP 地址寻址主机。数据包被分割为多个小的单元发送，并且在目的主机中重新组合，数据包最大为 65535B。传输时间监视可以防止传输过程中的死循环。通过生成校验和来验证数据报头的传输是否正确。IP 并不检查数据传输的正确性，也不提供确认和纠错机制。

（2）TCP

TCP（传输控制协议）是 OSI 参考模型第 4 层的协议，用于在两个站点之间建立逻辑的（虚拟的）全双工连接。TCP 是面向连接的端到端协议，通过使用 TCP 端口号，提供多路复用技术功能。节点之间的数据通信是面向连接的，连接到端点上的每一个站原则上在任何时刻都有权利发送数据。S7-300/400 通过工业以太网和通信块 AG_SEND/AG_RECV 提供 TCP、ISO、ISO-on-TCP 和 UDP 通信服务。

（3）TCP/IP 的可靠性措施

TCP/IP 通过序列编号、校验和、流量控制、定时监视、错误检测和纠正传输错误（数据的丢失、重复和损坏）的机制，确保了数据传输的最优化和准确性。在传输出错时重复发送数据。接收器通过循环冗余校验算法（CRC）检查接收到的数据的完整性，并确认数据的接收，发送端将在 SEND/RECEIVE 接口上收到一个返回值。

（4）TCP/IP 的数据传输过程

TCP 是基于连接的协议，在正式收发数据之前，必须和对方建立可靠的连接。下面对通信过程作简单的介绍：

1）主机 A 向主机 B 发送连接请求数据包。

2）主机 B 向主机 A 发送同意连接和要求同步的数据包，同步是指两台主机一个在发送，一个在接收，协调工作。

3）主机 A 再发送一个数据包确认主机 B 要求的同步。

经过上述“对话”之后，主机 A 才向主机 B 正式发送数据。

TCP 能为应用程序提供可靠的通信连接，使一台计算机发送的字节流无差错地发往网络上的其他计算机，对可靠性要求高的数据通信系统应使用 TCP 传输数据。

4．ISO 传输服务

ISO 协议符合 ISO 8073 标准，作为一种开放的协议，通过组态的连接，实现数据的安全通信。ISO 传输用于在组态的连接上进行可靠的数据传送，它将数据分段，可以传送大量的数据。S7 PLC 可以与具有以太网 CP 的 S7/S5 PLC、PC/PG 和使用 ISO 传输协议的系统通信，ISO 传输连接上的数据传输是双向的。ISO 传输对应于 ISO 参考模型的第 4 层（传输层），仅适用于工业以太网。ISO 传输服务保证数据传输及数据的完整性的方法与 TCP/IP 的相同。ISO 传输协议的通信过程如下（见图 10-17）：

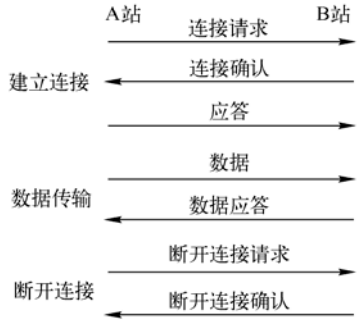


图 10-17　ISO 传输协议的传输过程

1）建立连接：A 站发出通信请求，B 站返回连接确认，A 站发出应答。

2）数据传输：A 站发送数据，B 站收到后返回确认应答。

3）断开连接：A 站发送断开连接请求，B 站返回确认报文，连接被断开。

5．ISO-on-TCP 服务

ISO-on-TCP 主要用于可靠的网际数据传输，符合 TCP/IP 标准，并根据 ISO 参考模型的第 4 层，增加了 RFC 1006 协议，可以改变长度的数据传输是通过 RFC 1006 实现的。RFC 1006 将 ISO 第 4 层的服务映射到 TCP。RFC 是 Requests for Comments（请求说明）的缩写，是数据通信领域事实上的工业标准。由于自动重发和附加的块校验机制（CRC 校验），ISO-on-TCP 的传输可靠性极高。通信伙伴将确认数据的接收，发送端将收到一个返回值。

通过以太网和 ISO-on-TCP 连接，SIMATIC S7 PLC 可以与支持 ISO-on-TCP 连接的 S7/S5 PLC、PC/PG 和非西门子公司的系统通信，最多可以发送 8KB 数据。可以使用 SEND/RECEIVE 和 FETCH/WRITE 等功能来传送数据。

在符合 RFC 1006 标准的 SIMATIC 设备之间，推荐使用这种通信连接。

6．UDP 服务

UDP 是 User Datagram Protocol（用户数据报协议）的简称，UDP 提供简单的跨网络数据传输通信服务（数据报服务），UDP 没有数据确认报文，不检测数据传输的正确性，属于 OSI 模型的第 4 层。必须的可靠性措施由应用层提供，可以将最大 2 KB 的连续数据块从一个以太网节点传送到另一个以太网节点。由于不发送对接收到的数据的确认报文，所以 UDP 通信并不可靠。UDP 适用于一次只传送少量数据、对可靠性要求不高的应用环境。

UDP 服务可以用于工业以太网、电话网或互联网，与支持 UDP 通信的 PC 或非西门子公司的系统的通信伙伴通信，UDP 服务也需要建立连接。

由于报文头短、没有传输应答和超时监控，UDP 比 TCP 更适合于对传输时间要求较高的应用。通过 UDP 连接，可以实现广播（向网络中所有站点发送消息）和多点传送（向网络中的多个站点发送消息）。空闲（free）的 UDP 连接用发送的数据的前 6 个字节来定义接收站的端口地址和 IP 地址。

在 7 层 OSI 参考模型中，TCP 和 UDP 在 IP 之上，因此使用 TCP、ISO-on-TCP 和 UDP 的 S5 兼容的通信必须设置 IP 地址，可以不设置 MAC 地址。ISO 传输必须设置 MAC 地址。

10.3.2 TCP 连接的组态与编程

下面以 S7-300 之间通过 CP 343-1 IT 和 CP 343-1 建立的 TCP 连接为例，介绍 S5 兼容通信的组态和编程的方法。

1．硬件组态

在 SIMATIC 管理器中，用新建项目向导创建一个新的项目，项目名称为“IE_TCP”，CPU 315-2DP 的 DP 和 MPI 地址均为 2。本章的例程在随书光盘的文件夹“\Project\Ethernet”中。建议在组态站点之前创建子网。步骤如下：

1）选中 SIMATIC 管理器中的项目。

2）执行菜单命令“插入”→“子网”→“工业 Ethernet”。

以后这个项目中创建的所有 SIMATIC 站点都可以连接到这个子网上。

在 HW Config 中，将电源模块、信号模块和 CP 343-1 IT 插入机架。插入 CP 343-1 IT 时，在自动打开的 CP 属性对话框的 “参数”选项卡中，可以看到默认的 CP 的 IP 地址 192.168.0.1

和子网掩码 255.255.255.0（见图 10-18 的左图），默认的网关设置是不使用路由器，TCP/IP 通信不需要设置 MAC 地址。选中子网列表中的 Ethernet(1)，点击“确定”按钮，返回 HW Config。

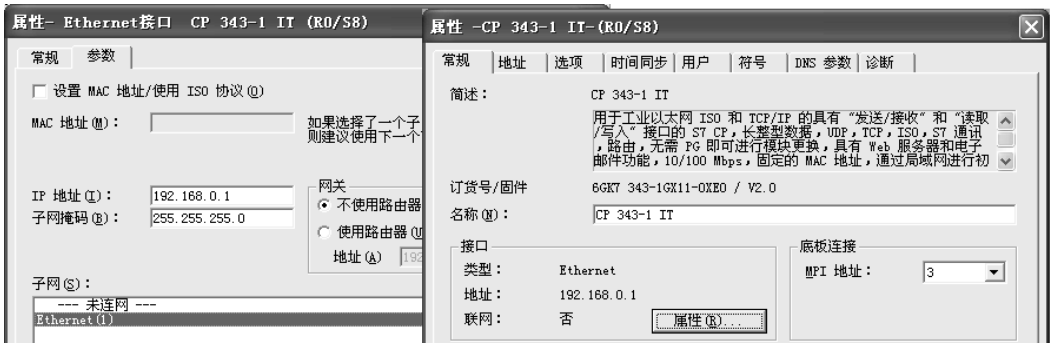


图 10-18　以太网接口属性对话框

双击机架中的 CP 343-1 IT，在打开的 CP 属性对话框的“常规”选项卡（见图 10-18 的右图）中，设置 CP 的 MPI 地址为 3。

在 SIMATIC 管理器中生成另一个 300 站点，在 HW Config 中，将电源模块、CPU 315-2DP、信号模块和 CP 343-1 插入机架。设置 CPU 的 DP 地址为 3，MPI 地址为 4。

双击 CP 343-1，在出现的 CP 属性对话框中，点击“属性”按钮。将它连接到前面生成的以太网上。采用默认的 IP 地址 192.168.0.2 和子网掩码 255.255.255.0。设置 CP 的 MPI 地址为 5。通信双方的 IP 地址必须在同一个网段内，即 IP 地址的前 3 个字节应为 192.168.0。如果用以太网下载和监控 PLC，PC 与 CPU 的 IP 地址也应在同一个网段内。

2．以太网 CP 的选项

功能 FC 5 AG_SEND 和 FC 6 AG_RECV 默认的最大数据传输量为 240B，TCP 通信一次最多可以发送 8KB 数据。

新版本的 CP 支持大于 240B 的数据传输。如果需要传输的数据量大于 240B，应在以太网 CP 的属性对话框的“选项”选项卡选中复选框“数据长度>240 字节”（见图 10-19）。在 S7-300 上进行此配置时，将使用 S7-300 CPU 的一个连接资源（用于 S7 功能的空闲连接）。如果要发送的数据长度不大于 240B，不要选中此选项。如果以太网 CP 不支持此功能，该复选框为灰色。

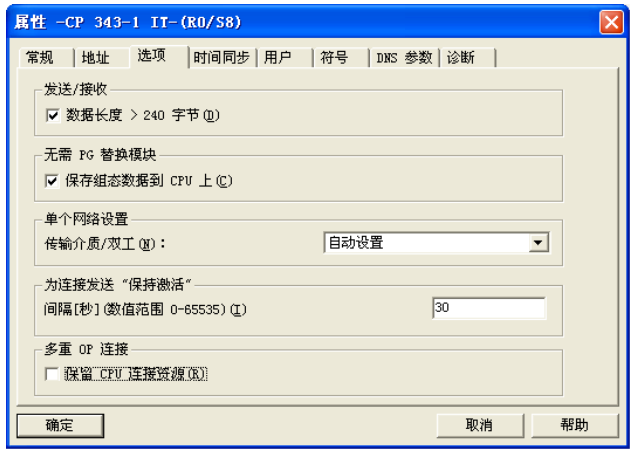


图 10-19　CP 属性对话框

如果选中复选框“保存组态数据到 CPU 上”，更换 CP 后，不需要重新下载组态数据。“传输介质/双工”一般采用默认的“自动设置”。

在“为连接发送‘保持激活’（keepalives）”区，可以设置通信伙伴出现故障时，向连接的通信伙伴发送“保持激活”的时间间隔（0～65535s），默认值为 30s。这样可以确保经过设置的延时时间后终止连接，以释放连接占用的资源。为连接发送“保持激活”只适用于 TCP 和 ISO-on-TCP 连接。

选中下面的复选框“保留 CPU 连接资源”，可以通过单个 CPU 连接资源，与最多 16 个 HMI（人机界面）设备进行通信（多路复用），来优化 S7-300 CPU 的连接资源。如果没有选中此选项，可以使用的 HMI 设备的个数取决于 CPU 可用的连接资源数。默认的设置是取消激活此选项。也就是说，只有在需要时才使用连接资源的多路复用。

PG（编程计算机）连接不使用多路复用连接；如果使用了编程计算机，它将单独占用一个连接资源。组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

3．组态连接

组态好两个 S7-300 站后，点击工具栏上的按钮，打开 NetPro 窗口，看到连接到以太网上的两个站（见图 10-20）。选中“SIMATIC 300（1）”站点的 CPU 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。在弹出的“插入新连接”对话框中，将“连接伙伴”设为与本站通信的 CPU 315-2DP，设置连接类型为“TCP 连接”。

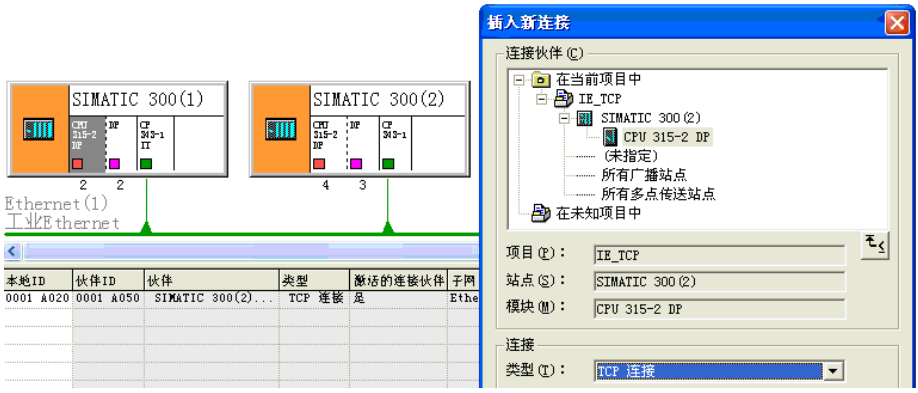


图 10-20　组态 TCP 连接

点击“确定”按钮，出现 TCP 连接属性对话框（见图 10-21）。在编程时要用到“块参数”区中的“标识”（ID）和 LADDR（CP 的起始地址）。

图 10-21　TCP 连接属性对话框

参数 LADDR（CP 的起始地址）与 CP 所在的插槽有关。该地址也是 S7-300 分配给模拟量模块的起始地址。本项目的 CP 343-1 IT 在 8 号槽，LADDR 的值为 320（W#16#140）。另一台 PLC 的 CP 343-1 在 9 号槽，LADDR 的值为 336（W#16#150）。

选中 NetPro 中“SIMATIC 300（2）”站点的 CPU 所在的小方框，因为是双向连接，下面的窗口出现自动生成的该站点一侧的连接表（见图 10-22），双击连接表中的“TCP 连接”，出现该站点一侧的连接属性对话框（见图 10-23）。

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
0001 A050	0001 A020	SIMATIC 300(1)...	TCP 连接	否	Ethernet(1) [IE]

图 10-22　SIMATIC 300（2）一侧的 TCP 连接

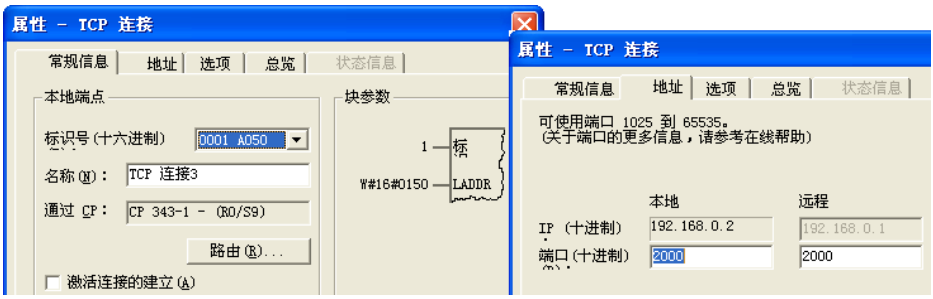


图 10-23　TCP 连接属性对话框

4．设计验证通信的程序

S5 兼容通信的双方通过调用 AG_SEND/AG_RECV（FC 5/FC 6）实现数据的发送和接收。AG_SEND/AG_RECV 在程序编辑器左边窗口的文件夹“\库\SIMATIC_NET_CP\CP 300”中。S7-400 的 CP 的通信块在文件夹“\库\SIMATIC_NET_CP\CP 400”中。

在 SIMATIC 管理器中生成数据块 DB 1 和 DB 2，用数组定义数据块的大小。

（1）OB35 中的发送程序

下面是 CPU 315-2DP 的 OB35 中的程序，ACT 是 FC 5 “AG_SEND”的发送使能位，ACT 为 1 状态时发送数据。为了实现周期性的数据发送，令 ACT 一直为 1 状态（true），如果在 OB1 中调用 FC 5，每个扫描循环周期都要发送一次，发送将过于频繁。因此将发送程序放在中断循环周期为 100ms（默认值）的 OB35 中。如果设置 ACT 的实参为一个位地址（例如 M10.0），可以用它来控制是否发送数据。

```
程序段 1：DB1.DBW0 加 1
        L     DB1.DBW     0
        +     1
        T     DB1.DBW     0
程序段 2：发送程序
        L     ID      0
        T     DB1.DBD     2          //用本站的 ID0 控制通信伙伴的 QD4
        CALL  "AG_SEND"              //调用 FC 5
          ACT     :=TRUE             //发送使能位，为 1 时发送数据
          ID      :=1                //组态时指定的连接ID
          LADDR   :=W#16#140         //组态时指定的CP起始地址
```

```
SEND    :=P#DB1.DBX 0.0 BYTE 240    //存放要发送的数据的地址区
LEN     :=240                       //发送数据字节数
DONE    :=M10.1                     //每次发送成功产生一个脉冲
ERROR   :=M10.2                     //错误标志位
STATUS  :=MW14                      //错误状态字
```

（2）OB1 中的接收程序

下面是 OB1 中的接收程序：

程序段 1：接收程序

```
CALL    "AG_RECV"                   //调用 FC 6
  ID        :=1                     //组态时指定的连接ID
  LADDR     :=W#16#140              //组态时指定的CP起始地址
  RECV      :=P#DB2.DBX 0.0 BYTE 240  //存放接收的数据的地址区
  NDR       :=M0.1                  //每次接收新数据产生一个脉冲
  ERROR     :=M0.2                  //错误标志位
  STATUS    :=MW2                   //错误状态字
  LEN       :=MW4                   //实际接收的数据长度
L       DB2.DBD     2
T       QD      4                   //用通信伙伴的 ID0 控制本站的 QD4
```

另一台 CPU 315-2DP 的发送程序和接收程序基本上相同，其区别仅在于 LADDR 的值为 W#16#150。

（3）初始化程序

在初始化程序 OB100 中，用 SFC 21 预置 DB 1 的数据发送区各个字的初值为 16#1111，将 DB 2 的数据接收区各个字清零。另一台 PLC 的 OB100 中的程序基本上相同，二者的区别在于发送区的数据字被初始化为 W#16#2222。

5．通信过程的监控

用 PROFIBUS 电缆将两块 CPU 315-2DP 和 CP 5613 的 MPI 接口连接到一起，将组态信息和程序分别下载到两台 PLC，运行时可以用 MPI 或以太网对通信过程进行监控。将以太网 CP 模块和计算机的以太网接口连接到交换机，将 CPU 和 CP 模块的模式选择开关切换到 RUN 位置，CPU 和 CP 上的 RUN 指示灯亮。CP 上的 LINK LED 亮，表示已建立起连接；RX/TX LED 闪烁，表示 CP 正在发送或接收数据。

同时打开通信双方的变量表（见图 10-24 和图 10-25），选中某个站的变量表后，点击工具栏上的  按钮，变量表进入监控状态，“状态值”列显示的是 PLC 中变量的值。通信双方在 OB35 中将 DB1.DBW0 加 1，然后发送到对方的 DB2.DBW0。在变量表中可以看到双方接收到的 DB2.DBW0 在不断变化。DB2.DBW238 是数据接收区的最后一个字。

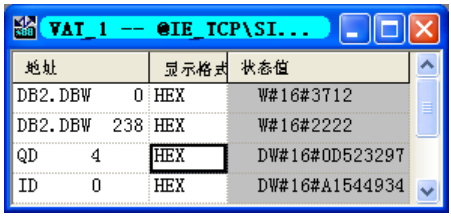


图 10-24　SIMTTC300（1）的变量表

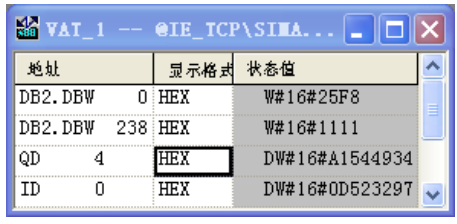


图 10-25　SIMTTC300（2）的变量表

在通信程序中，双方的 ID0 通过 DB1.DBD2 发送给通信伙伴的 DB2.DBD2，用来控制通信伙伴的 QD4。在运行时用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD4 的状态随之而变。

10.3.3 ISO 连接的组态与编程

在 SIMATIC 管理器中，用新建项目向导创建一个新的项目，项目名称为“IE_ISO”（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。

使用 ISO 传输协议时，必须设置 CP 的 MAC 地址。在 HW Config 中，将电源模块、信号模块和 CP 343-1 IT 插入机架。双击 CP 343-1 IT，在出现的 CP 属性对话框的“常规”选项卡中，点击“属性”按钮。在打开的以太网接口属性对话框的“参数”选项卡（见图 10-26 的左图）中，选中复选框“设置 MAC 地址/使用 ISO 协议”，在“MAC 地址”输入框中，输入 CP 模块的 MAC 地址，后者可以在模块上找到。点击“新建”按钮，生成一条名为“Ethernet(1)”的以太网，CP 343-1 IT 被连接到以太网上。点击“确定”按钮，返回 HW Config。

在 SIMATIC 管理器中生成另一个 300 站点，在 HW Config 中，将 CPU 315-2DP、电源模块、信号模块和 CP 343-1 插入机架。用上述的方法，设置 CP 模块的 MAC 地址（见图 10-26 的右图）。组态好硬件后，点击工具栏上的  按钮，编译并保存硬件组态信息。

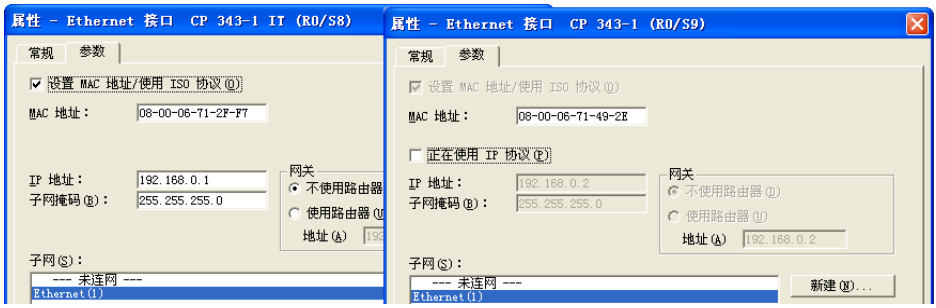


图 10-26　CP 的以太网接口属性对话框

组态好两个 S7-300 站后，点击工具栏上的按钮  ，打开 NetPro 窗口，看到连接到以太网上的两个站（见图 10-27）。选中“SIMATIC 300（1）”站点的 CPU 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。在弹出的“插入新连接”对话框中，将“连接伙伴”设为与本站通信的 CPU 315-2DP，连接类型为“ISO 传输连接”。

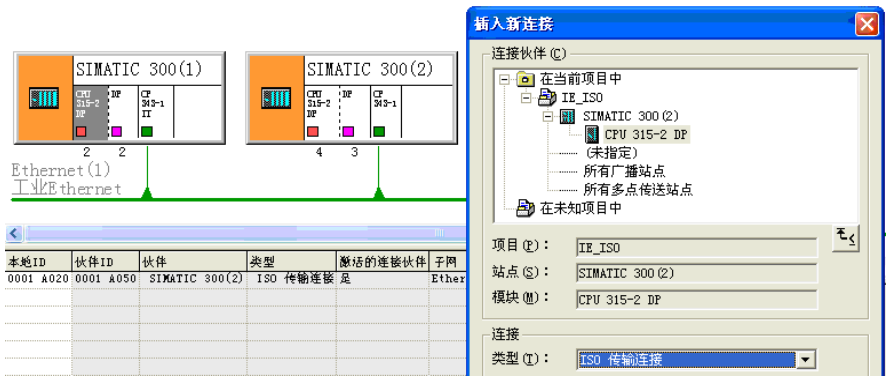


图 10-27　组态 ISO 传输连接

点击“确定”按钮，出现 ISO 传输连接属性对话框（见图 10-28）。选中 NetPro 中“SIMATIC 300（2）”站点的 CPU 所在的小方框，因为是双向连接，在下面的窗口出现自动生成的该站点一侧的连接表，双击连接表中的“ISO 传输连接”，可以看到该站点一侧的连接属性对话框。

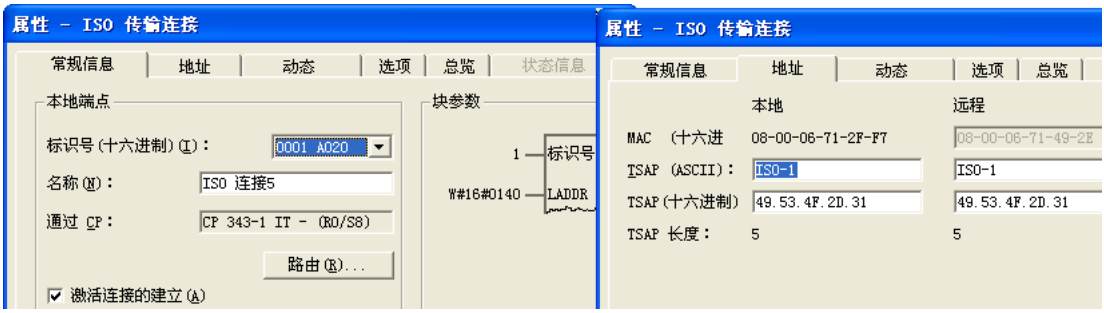


图 10-28　ISO 传输连接属性对话框

组态好通信连接后，点击工具栏上的按钮，编译并保存组态信息。

项目 IE_ISO 的通信程序与项目 IE_TCP 的基本上相同（见随书光盘中的例程），只是通信双方发送和接收的数据量为 500B。变量表、通信过程的监控方法也基本上相同。

10.3.4　ISO-on-TCP 连接的组态与编程

在 SIMATIC 管理器中，用新建项目向导创建一个新项目，项目名称为“ISOonTCP”（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。

在 HW Config 中，将电源模块、信号模块和 CP 343-1 IT 插入机架。双击 CP 343-1 IT，在出现的 CP 属性对话框中，点击“常规”选项卡中的“属性”按钮，在打开的以太网接口属性对话框的“参数”选项卡中，采用 CP 默认的 IP 地址 192.168.0.1 和子网掩码 255.255.255.0，不使用路由器，ISO-on-TCP 通信不需要设置 MAC 地址。

点击“新建”按钮，生成一条名为“Ethernet（1）”的以太网，CP 343-1 IT 被连接到以太网上。点击“确定”按钮，返回 HW Config。

在 SIMATIC 管理器中生成另一个 300 站点，在 HW Config 中，将 CPU 315-2DP、电源模块、信号模块和 CP 343-1 插入机架。在 CP 343-1 的属性对话框中，将它连接到前面生成的以太网上，采用默认的 IP 地址 192.168.0.2 和子网掩码 255.255.255.0。

组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

组态好两个 S7-300 站后，点击工具栏上的按钮，打开 NetPro 窗口，看到连接到以太网上的两个站（见图 10-29）。选中“SIMATIC 300（1）”站点的 CPU 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。在弹出的“插入新连接”对话框中，将“连接伙伴”设为与本站通信的 CPU 315-2DP，连接类型为“ISO-on-TCP 连接”。

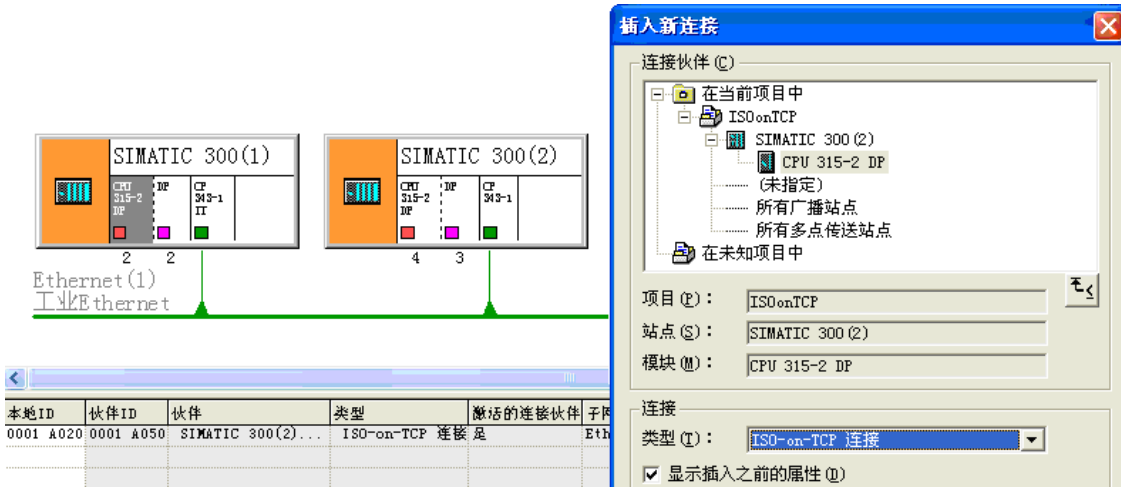


图 10-29　组态 ISO-on-TCP 连接

点击“确定”按钮，出现 ISO-on-TCP 连接属性对话框（见图 10-30）。

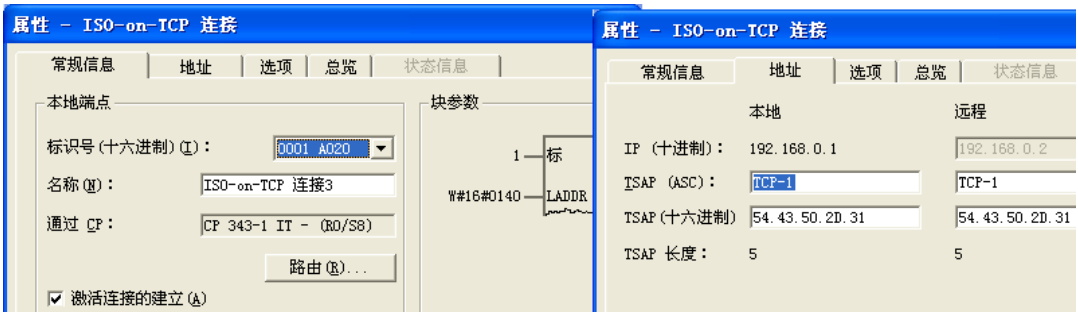


图 10-30　ISO-on-TCP 连接属性对话框

选中 NetPro 中“SIMATIC 300（2）”站点的 CPU 所在的小方框，因为是双向连接，在下面的窗口出现自动生成的该站点一侧的连接表，双击连接表中的“ISO-on-TCP 连接”，可以看到该站点一侧的连接属性对话框。

组态好通信连接后，点击工具栏上的按钮，编译并保存组态信息。

项目 ISOonTCP 的通信程序和变量表与项目 IE_TCP 的完全一样（见随书光盘中的例程），通信过程的监控方法也完全相同。

10.3.5　指定通信伙伴的 UDP 连接的组态与编程

在 SIMATIC 管理器中，创建一个新的项目，项目名称为“IE_UDP”（见随书光盘中的同名例程），CPU 为 CPU 315-2DP。

在 HW Config 中，将电源模块、信号模块和 CP 343-1 插入机架。双击 CP 343-1，在出现的 CP 属性对话框中，点击“常规”选项卡中的“属性”按钮。在打开的以太网接口属性对话框的“参数”选项卡中，设置 IP 地址为 192.168.0.3，子网掩码为默认的 255.255.255.0，不使用路由器，UDP 通信不需要设置 MAC 地址。

点击“新建”按钮，生成一条名为“Ethernet（1）”的以太网，CP 343-1 IT 被连接到以太网上。点击“确定”按钮，返回 HW Config。

在 SIMATIC 管理器中生成另外两个 300 站点，在 HW Config 中，将 CPU 315-2DP、电源模块、信号模块和 CP 343-1 IT 插入机架。在 CP 343-1 IT 的属性对话框中，将它连接到前面生成的以太网上，设置 IP 地址分别为 192.168.0.4 和 192.168.0.5，子网掩码为默认的 255.255.255.0。组态好各站的硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

3 个站点的以太网 CP 均在第 8 槽，其起始地址为 320（W#16#140）。本节使用 UDP 协议的 3 个项目均使用上述的硬件实验装置。

点击工具栏上的按钮，打开 NetPro 窗口，看到连接到以太网上的 3 个站（见图 10-31）。选中站点“SIMATIC 300（1）”的 CPU 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。在弹出的“插入新连接”对话框中，将“连接伙伴”设为站点“SIMATIC 300（2）”，连接类型为“UDP 连接”。点击“确定”按钮，出现 UDP 连接属性对话框（见图 10-32）。

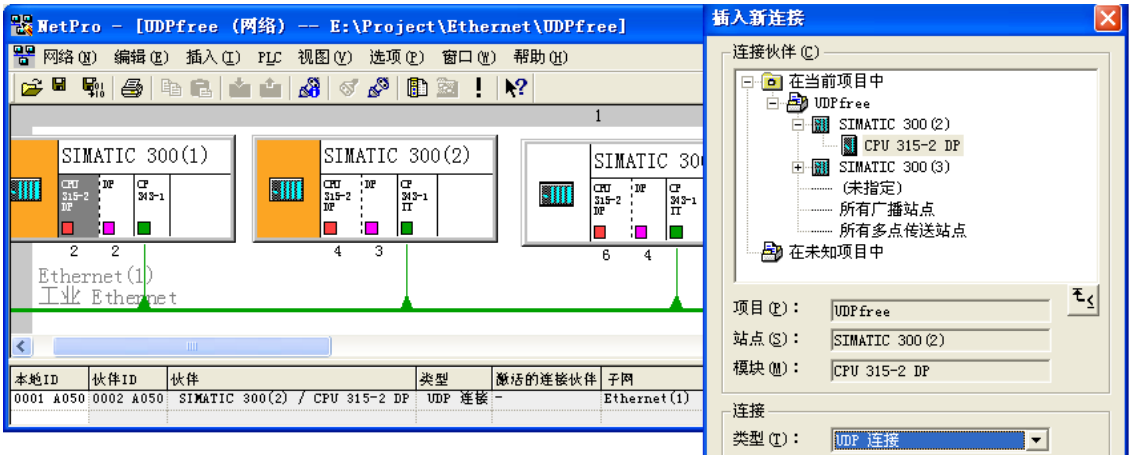


图 10-31　组态 UDP 连接

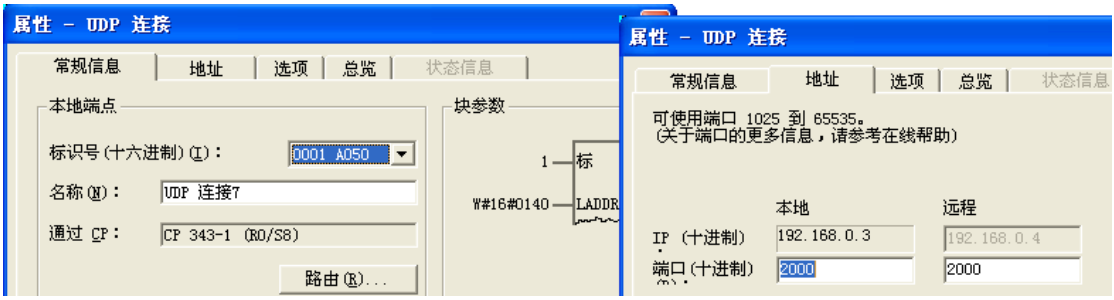


图 10-32　UDP 连接属性对话框

选中 NetPro 中站点“SIMATIC 300（2）”的 CPU 所在的小方框，因为是双向连接，在下面的窗口出现自动生成的该站点一侧的连接表，双击连接表中的“UDP 连接”，可以看到该站点一侧的连接属性对话框。

组态好通信连接后，点击工具栏上的按钮，编译并保存组态信息。项目 IE_UDP 的通信程序与项目 IE_TCP 的相同（见随书光盘中的例程），变量表和通信过程的监控方法也完全相同。

10.3.6 未指定通信伙伴的 UDP 连接的组态与编程

1．UDP 连接组态

在 SIMATIC 管理器中，用新建项目向导建立一个新的项目，项目名称为“UDPfree”（见随书光盘中的同名例程），3 个 S7-300 站点的硬件结构和组态方法与项目 IE_UDP 相同。

点击工具栏上的按钮，打开 NetPro 窗口。选中“SIMATIC 300（1）”站点的 CPU 所在的小方框，在下面的窗口出现连接表。双击连接表第一行的空白处，建立一个新连接。在出现的“插入新连接”对话框中，将“连接伙伴”设为“（未指定）”（见图 10-31），连接类型为“UDP 连接”。点击“确定”按钮，出现 UDP 连接属性对话框（见图 10-33）。

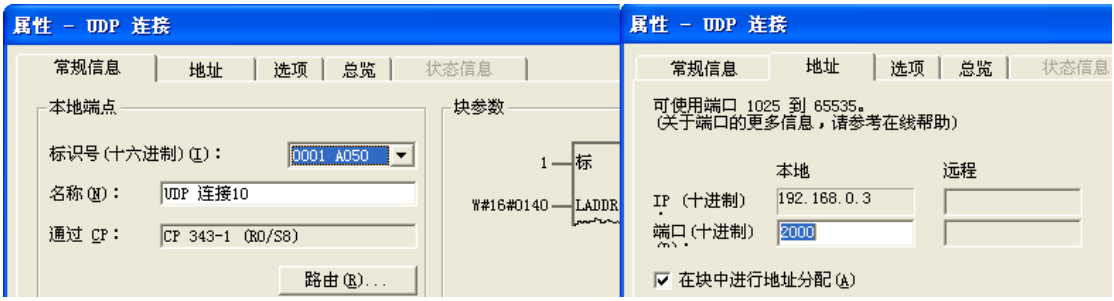


图 10-33　UDP 连接属性对话框

选中“地址”选项卡中的复选框“在块中进行地址分配”，这种连接称为“空闲（free）的 UDP 连接”，由用户程序指定远程通信伙伴的 IP 地址和端口地址，不能在组态时指定它们。目标站可以在 STEP 7 项目之内，也可以在 STEP 7 项目之外，可以用程序改变远程通信伙伴。

端口地址定义了站内用户程序的访问点，空闲的 UDP 连接的端口地址应大于等于 2000。

其他两个站的空闲的 UDP 连接的组态方法和 UDP 连接属性对话框，与站点 SIMATIC 300（1）的基本上相同。

2．发送数据的程序

各站调用 FC 1 和 FC 2，将数据发送给通信伙伴（见图 10-34）。发送的数据的前 6 个字节是接收数据的站点的端口地址和 IP 地址。接收站接收到的数据的前 6 个字节是发送站的端口地址和 IP 地址。

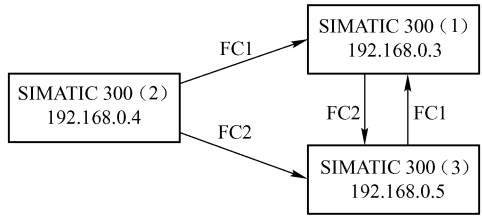


图 10-34　数据传送关系图

下面是站点 SIMATIC 300（2）的 FC 1 中的程序。

程序段 1：预置通信伙伴的端口号和 IP 地址

```
L     2000                          //SIMATIC 300（1）的端口号
T     DB1.DBW    0
L     192                           //SIMATIC 300（1）的 IP 地址第 1 字节
T     DB1.DBB    2
L     168                           //SIMATIC 300（1）的 IP 地址第 2 字节
T     DB1.DBB    3
L     0                             //SIMATIC 300（1）的 IP 地址第 3 字节
T     DB1.DBB    4
L     3                             //SIMATIC 300（1）的 IP 地址第 4 字节
T     DB1.DBB    5
```

程序段 2：发送数据

```
CALL   "AG_SEND"                    //调用 FC 5
  ACT      :=I0.0                   //发送使能位，为 1 时发送数据
  ID       :=1                      //组态时指定的连接ID
  LADDR    :=W#16#140               //组态时指定的CP起始地址
  SEND     :=P#DB1.DBX 0.0 BYTE 240 //存放要发送的数据的地址区
  LEN      :=240                    //发送的数据的字节数
  DONE     :=M10.1                  //每次发送成功产生一个脉冲
  ERROR    :=M10.2                  //错误标志位
  STATUS   :=MW14                   //错误状态字
```

站点 SIMATIC 300（2）的 FC 2 的程序段 1 将 SIMATIC 300(3)的端口地址 2000 和 IP 地址 192.168.0.5 传送给 DB 2 的前 6 个字节。下面是 FC 2 的程序段 2：

程序段 2：

```
CALL   "AG_SEND"                    //调用 FC 5
  ACT      :=I0.1                   //发送使能位，为 1 时发送数据
  ID       :=1                      //组态时指定的连接ID
  LADDR    :=W#16#140               //组态时指定的CP起始地址
  SEND     :=P#DB2.DBX 0.0 BYTE 240 //存放要发送的数据的地址区
  LEN      :=240                    //发送的数据的字节数
  DONE     :=M10.3                  //每次发送成功产生一个脉冲
  ERROR    :=M10.4                  //错误标志位
  STATUS   :=MW18                   //错误状态字
```

为了观察数据传输的动态过程，在各站点的 OB35 中，将需要发送的 DB 1、DB 2 的 DBW6 加 1。

在站点 SIMATIC 300（2）的 OB35 中调用 FC 1 和 FC 2，FC 5“AG_SEND”的发送使能位 I0.0 或 I0.1 为 1 时，分别将数据发送给站点 SIMATIC 300（1）和 SIMATIC 300（3）。

站点 SIMATIC 300（1）与 SIMATIC 300（2）的 FC 2 相同。在站点 SIMATIC 300（1）的 OB35 中调用 FC 2，FC 5 的发送使能位 I0.1 为 1 时，将数据发送给站点 SIMATIC 300（3）。

站点 SIMATIC 300（3）与 SIMATIC 300（2）的 FC 1 相同。在站点 SIMATIC 300（3）的 OB35 中调用 FC 1，FC 5 的发送使能位 I0.0 为 1 时，将数据发送给站点 SIMATIC 300（1）。

3．初始化程序

站点 SIMATIC 300（2）的 OB100 调用 SFC 21，将 DB 1 和 DB 2 的数据发送区的各个字预置为 16#2222。

站点 SIMATIC 300（1）的 OB100 调用 SFC 21，将 DB 2 数据发送区的各个字预置为 16#1111，将 DB 3 和 DB 4 的数据接收区中的字清零。

站点 SIMATIC 300（3）的 OB100 调用 SFC 21，将 DB 1 数据发送区的各个字预置为 16#3333，将 DB 3 和 DB 4 的数据接收区中的字清零。

4．接收和保存数据的程序

接收站接收到的数据的前 6 个字节是发送站的端口地址和 IP 地址。

站点 SIMATIC 300（1）的 OB1 调用 FC 6 接收到数据后保存在 DB 5，根据 DB5.DBB5（发送站 IP 地址第 4 个字节）判断是哪个站发送的，然后保存到不同的数据块。

程序段 1：接收程序

```
      CALL    "AG_RECV"                    //调用 FC 6
        ID     :=1                         //连接ID
        LADDR  :=W#16#140                  //CP 343-1 的起始地址
        RECV   :=P#DB5.DBX 0.0 BYTE 240    //存放接收的数据的地址区
        NDR    :=M0.1                      //每次接收到新数据产生一个脉冲
        ERROR  :=M0.2                      //错误标志位
        STATUS :=MW2                       //状态字
        LEN    :=MW4                       //接收的数据字节数
```

程序段 2：保存数据

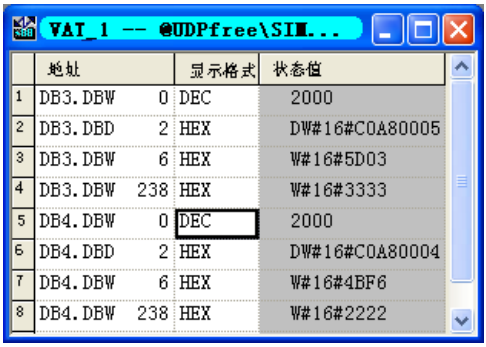
```
      A     M       0.1
      JCN   m002                           //未接收到数据时跳转
      L     5
      L     DB5.DBB     5                  //发送站 IP 地址第 4 字节
      ==I
      JCN   m001                           //不是站点 SIMATIC 300（3）发送的数据则跳转
      CALL "BLKMOV"                        //调用 SFC 20
        SRCBLK  :=P#DB5.DBX0.0 BYTE 240    //源数据区
        RET_VAL :=MW6
        DSTBLK  :=P#DB3.DBX0.0 BYTE 240    //目的数据区
m001: L     4
      L     DB5.DBB     5                  //发送站 IP 地址第 4 字节
      ==I
      JCN   m002                           //不是站点 SIMATIC 300（2）发送的数据则跳转
      CALL "BLKMOV"                        //调用 SFC 20
        SRCBLK  :=P#DB5.DBX0.0 BYTE 240    //源数据区
        RET_VAL      :=MW8
        DSTBLK  :=P#DB4.DBX0.0 BYTE 240    //目的数据区
m002: NOP   0
```

站点 SIMATIC 300（3）和 SIMATIC 300（1）的 OB1 的程序基本上相同。其区别仅在于前者的程序段 2 两个发送站的 IP 地址第 4 字节的值分别为 3 和 4。

5．通信过程的监控

将组态信息和程序块分别下载到 3 台 PLC，将以太网 CP 模块和计算机的以太网接口连接到交换机，将 CPU 和 CP 模块的模式选择开关切换到 RUN 位置，CPU 和 CP 上的 RUN 指示灯亮。用以太网对通信过程进行监控。

打开站点 SIMATIC 300（1）与 SIMATIC 300（3）的变量表（见图 10-35 和图 10-36），选中某个站的变量表后，点击工具栏上的按钮，变量表进入监控状态，“状态值”列显示的是 PLC 中变量的值。发送方的 OB35 将 DB 1 或 DB 2 的 DBW6 加 1，然后发送给对方。各站 FC 5 的发送使能位（I0.0 或 I0.1）为 1 时，将数据发送给通信伙伴。在变量表中可以看到接收到的 DBW6 在动态变化。I0.0 或 I0.1 为 0 时，对应的数据块的 DBW6 停止变化。DBW238 是数据接收区的最后一个字。



	地址		显示格式	状态值
1	DB3.DBW	0	DEC	2000
2	DB3.DBD	2	HEX	DW#16#C0A80005
3	DB3.DBW	6	HEX	W#16#5D03
4	DB3.DBW	238	HEX	W#16#3333
5	DB4.DBW	0	DEC	2000
6	DB4.DBD	2	HEX	DW#16#C0A80004
7	DB4.DBW	6	HEX	W#16#4BF6
8	DB4.DBW	238	HEX	W#16#2222

图 10-35　站点 SIMATIC 300（1）的变量表

	地址		显示格式	状态值
1	DB3.DBW	0	DEC	2000
2	DB3.DBD	2	HEX	DW#16#C0A80003
3	DB3.DBW	6	HEX	W#16#94A8
4	DB3.DBW	238	HEX	W#16#1111
5	DB4.DBW	0	DEC	2000
6	DB4.DBD	2	HEX	DW#16#C0A80004
7	DB4.DBW	6	HEX	W#16#4970
8	DB4.DBW	238	HEX	W#16#2222

图 10-36　站点 SIMATIC 300(3)的变量表

以站点 SIMATIC 300（2）发送数据给 SIMATIC 300（1）为例，当前者的 I0.0 为 1 状态，I0.1 为 0 状态时，将 DB 1 中的数据发送给站点 SIMATIC 300（1）。后者接收到后，根据 DB5.DBB5（发送方 IP 地址的最后一个字节）的值为 4，判断出是站点 SIMATIC 300（2）发送的，调用 SFC 20 将 DB 5 中的数据传送给 DB 4。图 10-35 中的 DB4.DBW0 为 2000，是发送站的端口地址。DB4.DBD2 为 W#16#C0A80004，即发送站 SIMATIC 300（2）的 IP 地址 192.168.0.4。DB4.DBW6 的值不断增大。用外接的小开关使站点 SIMATIC 300（2）的 I0.0 为 0，DB4.DBW6 停止变化。

实验时发现站点 SIMATIC 300(2)不能用 I0.0 和 I0.1 同时调用 FC 1 和 FC 2 来发送数据，只能单独调用 FC 1 或 FC 2 来发送数据。3 个站的端口地址可以相同，也可以不同。

在 HW Config 和 NetPro 中编译和保存组态信息后，如果仅在 SIMATIC 管理器下载系统数据，实验时发现某些站点没有接收到完整的网络组态信息，导致不能实现全部通信功能。遇到这种情况时，在 NetPro 中分别下载网络组态信息给各 CPU，可以保证系统的正常运行。

10.3.7　多点传送方式的 UDP 连接的组态与编程

在工业以太网中，只有组态了 UDP 连接的 CP 才支持多点传送。组态连接时选择“所有多点传送站点”作为连接伙伴，通信伙伴是多点传送组中的所有多点传送节点。本地设备将接收它所在的多点传送组中的多点传送帧。

1．硬件与网络组态

在 SIMATIC 管理器中，创建一个新项目，项目名称为“UDP_MUL”（见随书光盘中的

同名例程），3 个 S7-300 站点的硬件结构和组态方法与项目 IE_UDP 相同。

点击工具栏上的按钮，打开 NetPro 窗口（见图 10-37）。选中“SIMATIC 300（1）”站点的 CPU 所在的小方框，在下面的窗口出现连接表。双击连接表第一行的空白处，建立一个新连接。在弹出的“插入新连接”对话框中，将“连接伙伴”设为“所有多点传送站点”，连接类型为“UDP 连接”。点击“确定”按钮，出现 UDP 连接属性对话框（见图 10-38）。

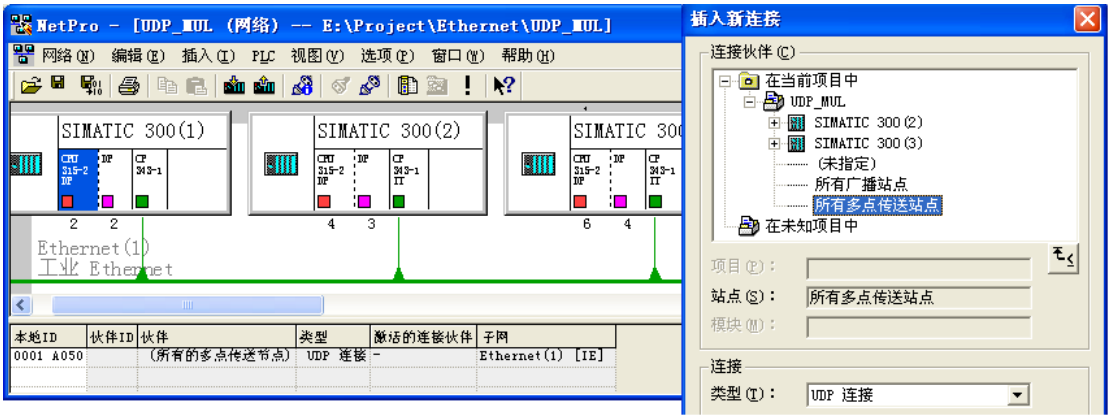


图 10-37　组态 UDP 连接

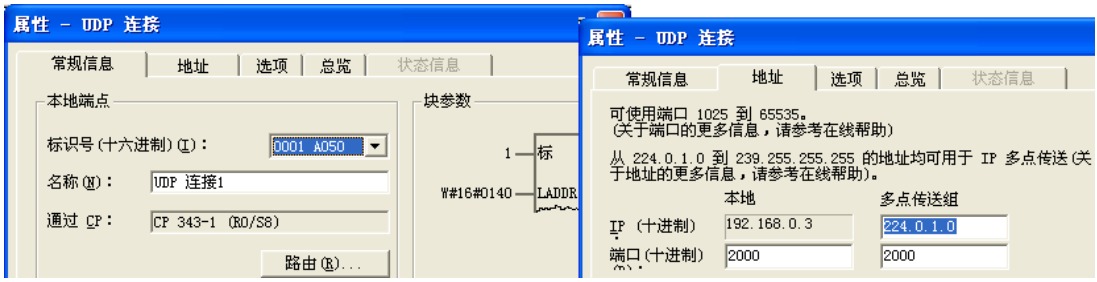


图 10-38　UDP 连接属性对话框

除了本地 IP 地址不同外，其他两个站的“所有多点传送站点”UDP 连接的组态方法和 UDP 连接属性对话框，与站点 SIMATIC 300（1）的基本上相同。

多点传送组用它的 IP 地址和端口地址来定义。UDP 连接属性对话框的“地址”选项卡给出了建议的多点传送组的 IP 地址和端口地址（见图 10-38 中的右图），供同一组的通信伙伴使用。应为多点传送组内的本地站和伙伴站分配完全相同的端口地址。

多点传送组可以使用的 IP 地址从 224.0.1.0 到 239.255.255.255，未使用 IP 地址的第 1 个字节，和第 2 个字节的最高位（共 9 位）。

通过创建 IP 地址相同，但是端口地址不同的几个多点传送 UDP 连接，可以用一个 IP 地址对多个多点传送组寻址。

2．初始化程序

站点 SIMATIC 300（1）～SIMATIC 300（3）的 OB100 将 DB 1 的数据发送区的各个字分别预置为 16#1111、16#2222 和 16#3333，将接收数据的 DB 3 和 DB 4 清零。

与空闲的 UDP 连接不同，多点传送的 UDP 连接不需要用发送的数据的前 6 个字节来指

定接收站的端口地址和 IP 地址。为了便于接收方判别是哪个站点发送的数据，在 OB100 中，将本站 IP 地址的最后一个字节传送给保存发送的数据的 DB 1 的 DBB2。

3．发送数据的程序

下面是各站点 OB35 中的发送程序：

```
程序段 1：将 DB1.DBW0 加 1
        L       DB1.DBW      0
        +       1
        T       DB1.DBW      0
程序段 1：  发送程序
        CALL    "AG_SEND"                          //调用 FC 5
          ACT       :=TRUE                         //发送使能位，为 1 时发送数据
          ID        :=1                            //组态时指定的连接ID
          LADDR     :=W#16#140                     //组态时指定的CP起始地址
          SEND      :=P#DB1.DBX 0.0 BYTE 240       //存放要发送的数据的地址区
          LEN       :=240                          //发送数据字节数
          DONE      :=M10.1                        //每次发送成功产生一个脉冲
          ERROR     :=M10.2                        //错误标志位
          STATUS    :=MW14                         //错误状态字
```

4．接收程序

各站点 OB1 的程序基本上相同，首先调用 FC 6，将接收到的多点传送组内的通信伙伴发送的数据保存到 DB 2，根据 DB2.DBB2（IP 地址的最后一个字节）判断是哪个站点发送的，然后分别保存到不同的数据块中。

下面是站点 SIMATIC 300（1）的 OB1 中的程序：

```
程序段 1：接收程序
        CALL    "AG_RECV"                          //调用 FC 6
          ID        :=1                            //连接ID
          LADDR     :=W#16#140                     //CP的起始地址
          RECV      :=P#DB2.DBX 0.0 BYTE 240       //存放接收的数据的地址区
          NDR       :=M0.1                         //每次接收到新数据产生一个脉冲
          ERROR     :=M0.2                         //错误标志位
          STATUS    :=MW2                          //状态字
          LEN       :=MW4                          //接收的数据字节数
程序段 2：如果是站点 SIMATIC 300（2）发送的数据，则保存到 DB 3
        A       M       0.1
        JCN     m002                               //未接收到数据时跳转
        L       4
        L       DB2.DBB      2                     //发送方 IP 地址最低字节的值
        ==I
        JCN     m001                               //不是站点 SIMATIC 300（2）发送的数据则跳转
        CALL    "BLKMOV"                           //调用 SFC 20 将接收的数据保存到 DB 3
          SRCBLK    :=P#DB2.DBX0.0 BYTE 240        //源数据区
          RET_VAL   :=MW6
          DSTBLK    :=P#DB3.DBX0.0 BYTE 240        //目的数据区
```

```
          JU      m002
```
程序段 3：如果是 SIMATIC 300(3)发送的数据，则保存到 DB 4

```
m001:     L       5
          L       DB2.DBB     2                 //发送方 IP 地址最低字节的值
          ==I
          JCN     m002                          //不是站点 SIMATIC 300(3)发送的数据则跳转
          CALL    "BLKMOV"                      //调用 SFC 20 将接收的数据保存到 DB 4
            SRCBLK  :=P#DB2.DBX0.0 BYTE 240     //源数据区
            RET_VAL       :=MW6
            DSTBLK  :=P#DB4.DBX0.0 BYTE 240     //目的数据区
m002:     NOP     0
```

各站点 OB1 的区别仅在于发送站的 IP 地址的最低字节不同。

5．通信过程的监控

将组态信息和程序分别下载到 3 台 PLC，将以太网 CP 模块和计算机的以太网接口连接到交换机，CPU 和 CP 模块的模式选择开关切换到 RUN 位置，CPU 和 CP 上的 RUN 指示灯亮。打开各站点的变量表（见图 10-39～图 10-41），选中某个站的变量表后，点击工具栏上的按钮，变量表进入监控状态。

程序正常运行时，图 10-39～图 10-41 中各个站接收到的 DB3.DBW0 和 DB4.DBW0 同时不断增大，说明多点传送组中的每个站都可以同时接收到其他站发送的数据。图中的 DBB2 是发送站 IP 地址的最后一个字节，DBW 238 是接收到的最后一个字。

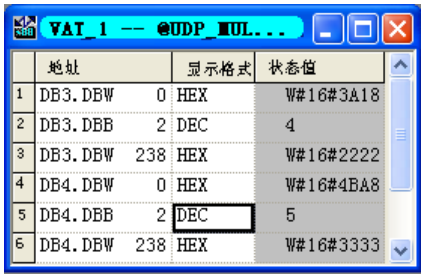


图 10-39　站点 SIMATIC 300（1）的变量表

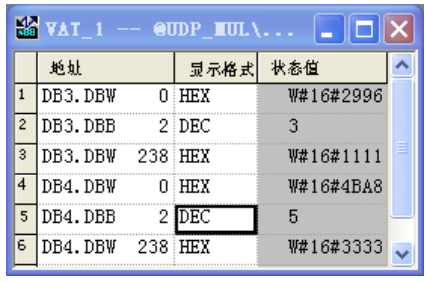


图 10-40　站点 SIMATIC 300（2）的变量表

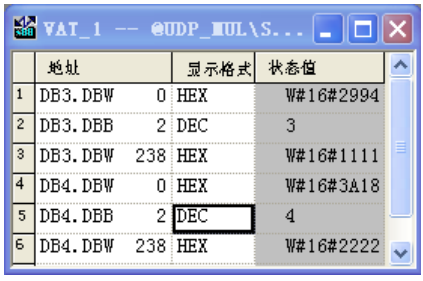


图 10-41　站点 SIMATIC 300(3)的变量表

6．广播方式的 UDP 通信

组态 UDP 连接时，选择通信伙伴为“所有广播站点”，可以把 UDP 帧发送给所有的广播站点。发送广播帧（ARP 请求）来查找已知 IP 地址的某个站点的 MAC 地址，是广播方式的

典型例子。在广播方式，只能用 CP 发送数据，CP 不能接收以传送用户数据为目的的广播帧。如果需要将数据同时传送给多个通信伙伴，可以使用多点传送的 UDP 连接。

10.4　基于以太网的 S7 通信

S7 通信是专为 SIMATIC S7 和 C7 优化设计的通信协议，提供简明、强有力的通信服务。

S7 通信主要用于 S7-300/400 CPU 之间的主-主通信、CPU 与功能模块（FM）之间、CPU 与西门子公司的人机界面 TP/OP（触摸屏/操作员面板）和组态软件 WinCC 之间的通信。S7 通信可以用于 PROFINET（工业以太网）、PROFIBUS 或 MPI 网络。

10.4.1　使用 PUT/GET 的单向 S7 通信

1．硬件组态

在 SIMATIC 管理器中，用新建项目向导创建一个新的项目（见图 10-42），项目名称为 IE_S7_1（见随书光盘中的同名例程），CPU 313C-2DP 的 DP 和 MPI 地址均为 2。

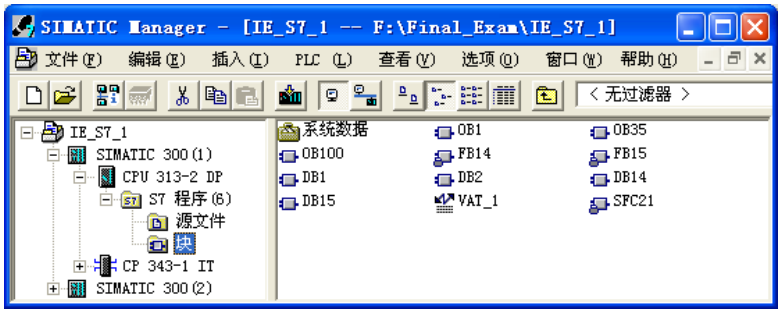


图 10-42　SIMATIC 管理器

在 HW Config 中，将电源模块、信号模块和 CP 343-1 IT 插入机架。双击 CP 343-1 IT，在出现的 CP 属性对话框的“常规”选项卡中，设置 CP 的 MPI 地址为 3。点击“属性”按钮，在自动打开的以太网接口属性对话框的“参数”选项卡中，采用默认的 CP 的 IP 地址 192.168.0.1 和子网掩码 255.255.255.0（见图 10-43）。设置了 CP 的 IP 地址后，可以不设置 MAC 地址。默认的网关设置是不使用路由器。

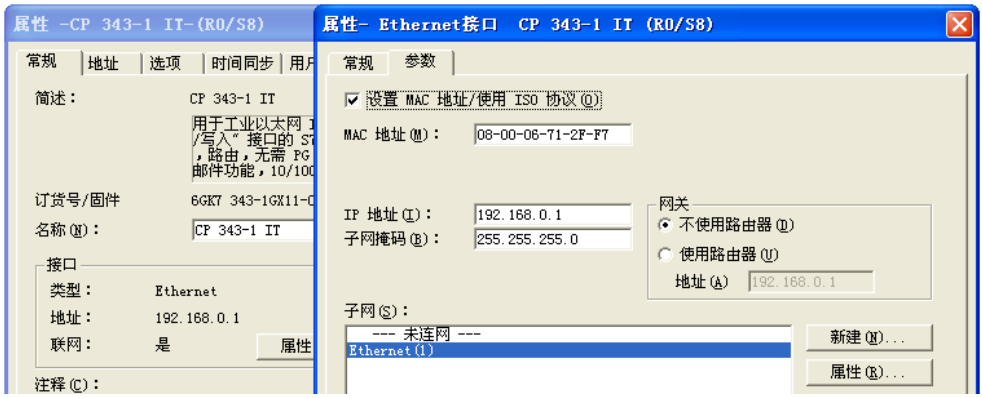


图 10-43　CP 的以太网接口属性对话框

点击“新建”按钮，在出现的新建工业以太网属性对话框中，点击“确定”按钮，确认自动生成的子网 ID。返回以太网接口属性对话框，可以看到生成的名为“Ethernet（1）”的以太网，CP 343-1 IT 被连接到以太网上。两次点击“确定”按钮，返回 HW Config。

在 SIMATIC 管理器中生成另一个 300 站点，在 HW Config 中，将电源模块、CPU 315、信号模块和 CP 343-1 插入机架。设置 CPU 的 MPI 地址为 4。

双击 CP 343-1，在出现的 CP 属性对话框中，设置 CP 的 MPI 地址为 5。点击“属性”按钮，在出现的以太网接口属性对话框的“参数”选项卡中，将它连接到前面生成的以太网上。采用默认的 IP 地址 192.168.0.2 和子网掩码 255.255.255.0，设置 CP 的 MAC 地址。

通信双方的 IP 地址必须在同一个网段内，即 IP 地址的前 3 个字节应为 192.168.0。如果用以太网下载和监控 PLC，PC 与 CPU 的 IP 地址也应在同一个网段内。组态和编程完成后的项目见图 10-42。组态好硬件后，点击工具栏上的按钮，编译并保存硬件组态信息。

2．S7 连接的组态

组态好两个 S7-300 站后，点击工具栏上的按钮，打开 NetPro 窗口，看到连接到以太网上的两个站（见图 10-44）。选中“SIMATIC 300（1）”站点的 CPU 313-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。

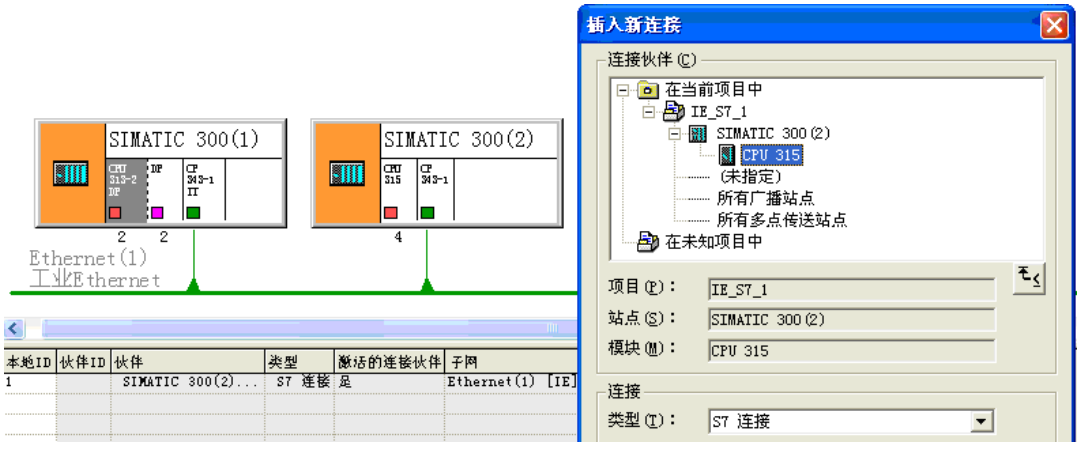


图 10-44　组态 S7 连接

在出现的“插入新连接”对话框中（见图 10-44 的右图），系统默认的通信伙伴为站点 SIMATCI 300（2）的 CPU 315，默认的连接类型为 S7 连接。

点击“确定”按钮，出现 S7 连接属性对话框（见图 10-45）。在编程时要用到“块参数”区中的“ID”（标识）的值。本项目使用的硬件可以建立双向的 S7 连接，有的 CPU 和 CP 只能建立单向的 S7 连接。点击 S7 连接属性对话框中的“单向”复选框，建立一个单向的 S7 连接。因为是单向连接，连接表中没有通信伙伴的 ID，选中 SIMATIC 300（2）站点的 CPU 315 所在的小方框，连接表中没有连接信息。

在 S7 单向连接中，CPU 313C-2DP 和 CPU 315 分别作为客户机（Client）和服务器（Server），客户机调用单向通信功能块 GET 和 PUT，通过以太网和 S7 通信，读、写服务器的存储区。服务器是通信中的被动方，不需要调用通信功能块。

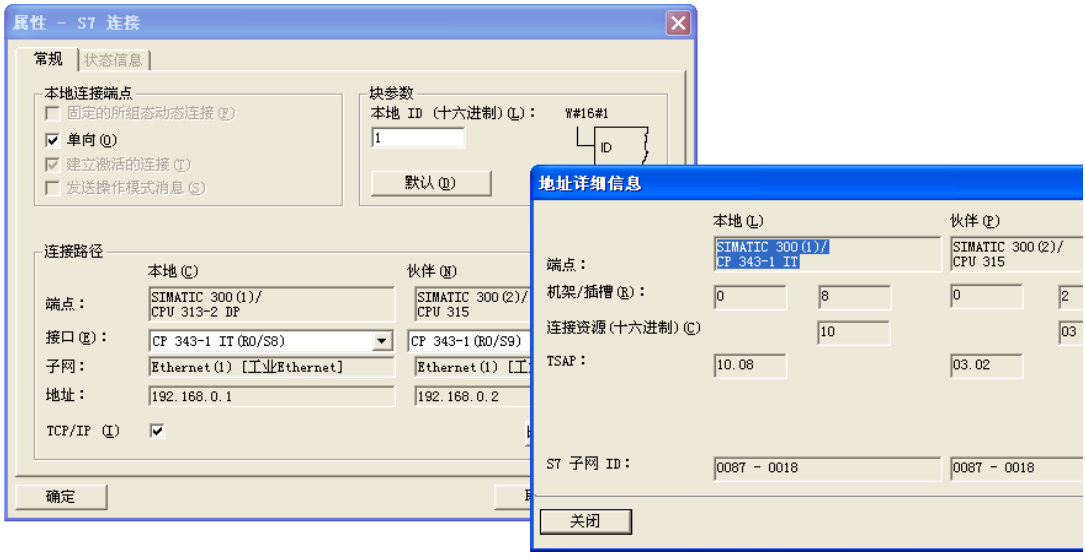


图 10-45　S7 连接属性对话框

选中下面的“TCP/IP”复选框，使用 IP 地址。它上面的“地址”行显示的是 CP 的以太网接口属性对话框中设置的 CP 的 IP 地址（见图 10-45）。如果未选中该复选框，“地址”行将显示 CP 的以太网接口属性对话框中设置的 CP 的 MAC 地址。

组态好连接后，点击工具栏上的按钮，编译并保存网络组态信息。

3．S7 通信的编程

S7-400 使用的 S7 通信系统功能块（SFB）在程序编辑器左边窗口的文件夹“\库\Standard Library\System Function Blocks”中，S7-300 使用的 S7 通信功能块（FB）在文件夹“\库\SIMATIC_NET_CP\CP 300 中”。

S7 通信功能块在通信请求信号 REQ 的上升沿时激活数据传输，为了实现周期性的数据传输，在 CPU 的属性对话框的“周期/时钟存储器”选项卡中（见图 4-5），定义 MB8 为时钟存储器字节，用时钟周期为 200ms 的 M8.1 作 REQ 信号。

OB1 的程序段 1 中的两条语句使 M10.0 和 M8.1 的相位相反，用它们来作功能块 GET 和 PUT 的 REQ（通信请求）信号，它们的上升沿相差 100ms。下面是 OB1 中的程序：

```
程序段 1：  时钟脉冲信号反相
        AN    M          8.1
        =     M          10.0
程序段 2：  读取通信伙伴的数据
        CALL  "GET", DB14                    //调用 FB 14
          REQ     :=M8.1                     //通信请求，上升沿时激活数据传输，每 200ms读取一次
          ID      :=W#16#1                   //S7 连接号
          NDR     :=M0.1                     //每次读取完成产生一个脉冲
          ERROR   :=M0.2                     //错误标志，出错时为 1
          STATUS  :=MW2                      //状态字，为 0 时表示没有警告和错误
          ADDR_1  :=P#DB1.DBX0.0 BYTE 160    //要读取的通信伙伴的地址区
          RD_1    :=P#DB2.DBX0.0 BYTE 160    //本站存放读取的数据的地址区
        L     DB2.DBD    2
```

```
T       QD          4                    //用对方的 ID0 控制本站的 QD4
```

程序段 3：向通信伙伴的数据区写入数据

```
L       ID          0
T       DB1.DBD     2                    //用本站的 ID0 控制通信伙伴的 QD4
CALL    "PUT" , DB15                     //调用 FB 15
  REQ       :=M10.0                      //通信请求，上升沿时激活数据交换，每 200ms写一次
  ID        :=W#16#1                     //S7 连接号
  DONE      :=M10.1                      //每次写完成产生一个脉冲
  ERROR     :=M10.2                      //错误标志，出错时为 1
  STATUS    :=MW12                       //状态字，为 0 时表示没有警告和错误
  ADDR_1    :=P#DB2.DBX0.0 BYTE 160      //通信伙伴要写入数据的地址区
  SD_1      :=P#DB1.DBX0.0 BYTE 160      //存放本站要发送的数据的地址区
```

下面是 CPU 315 的 OB1 中的程序：

```
L       ID      0
T       DB1.DBD 2                        //用本站的 ID0 控制通信伙伴的 QD4
L       DB2.DBD 2
T       QD      4                        //用通信伙伴的 ID0 控制本站的 QD4
```

通过 CPU 313-2DP 读、写 CPU 315 的数据区，实现了用两个站的 ID0 分别控制对方的 QD4。在通信双方的 OB35 中，每 100ms 将 DB1.DBW0 加 1。

4．初始化程序

在 CPU 313C-2DP 的初始化程序 OB100 中，用 SFC 21 预置数据发送区 DB 1 各个字的初值为 16#1111，将 DB 2 中的数据接收区清零。

CPU 315 的 OB100 中的程序基本上相同，只是将数据发送区中的字初始化为 W#16#2222。

5．通信的监控

用 PROFIBUS 电缆将两块 CPU 和 CP 5613 的 MPI 接口连接到一起，将组态信息和程序分别下载到两台 PLC，运行时可以用 MPI 或以太网对通信过程进行监控。

将以太网 CP 模块和计算机的以太网接口连接到交换机，将 CPU 和 CP 模块的模式选择开关切换到 RUN 位置，CPU 和 CP 上的 RUN 指示灯亮。

同时打开通信双方的变量表，将它们调节到适当的大小。点击工具栏上的  按钮，变量表进入监控状态，“状态值”列显示的是 PLC 中变量的值。

通信双方在 OB35 中将 DB1.DBW0 加 1，然后发送到对方的 DB2.DBW0。在变量表中可以看到双方接收到的 DB2.DBW0 在不断地变化。图 10-46 和图 10-47 是在运行时复制的变量表。在通信程序中，双方的 ID0 通过 DB1.DBD2 发送给对方的 DB2.DBD2，用来控制对方的 QD4。在运行时用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD4 的状态随之而变。

图 10-46　CPU 313C-2DP 的变量表

图 10-47　CPU 315 的变量表

10.4.2 使用 USEND/URCV 的双向 S7 通信

使用 SFB/FB USEND/URCV，可以进行快速、不可靠的数据传送，例如，可以用于事件消息和报警消息的传送。BSEND/BRCV 和 USEND/URCV 属于双向通信块，通信的双方都必须调用通信功能块。如果使用集成的 MPI 接口或集成的 DP 接口，它们只能用于两台 S7-400 之间的 S7 通信。本项目使用的硬件允许建立双向的 S7 连接。

1．硬件组态

在 STEP 7 中创建一个项目（见随书光盘中的例程 IE_S7_2），生成两个站，CPU 模块分别为 CPU 313C-2DP 和 CPU 315，两个站分别使用以太网模块 CP 343-1 IT 和 CP 343-1。系统的硬件组成与项目 IE_S7_1 相同，硬件组态的过程、以太网接口的地址与项目 IE_S7_1 也完全相同。

2．S7 连接组态

组态好两个 S7-300 站后，点击工具栏上的按钮，打开 NetPro，看到连接到以太网上的两个站（见图 10-48）。选中“SIMATIC 300（1）”站点的 CPU 313-2DP 所在的小方框，在下面的窗口出现连接表，双击连接表第一行的空白处，建立一个新连接。

在出现的“插入新连接”对话框中（见图 10-48 的右图），系统默认的通信伙伴为站点 SIMATCI 300（2）的 CPU 315，默认的连接类型为 S7 连接。

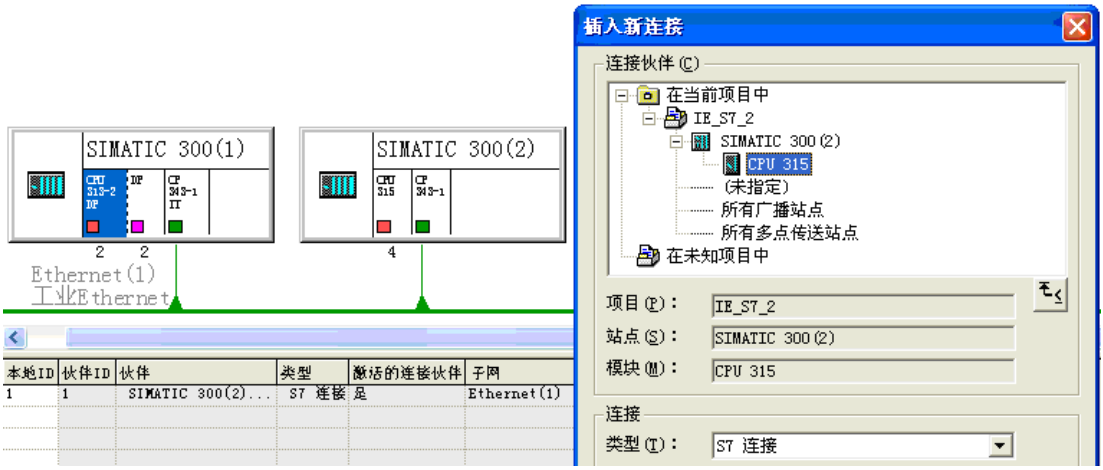


图 10-48　组态 S7 连接

点击“确定”按钮，出现 S7 连接属性对话框（见图 10-49）。在“本地连接端点”区，没有选中复选框“单向”，因此连接是双向的，在连接表中生成了“本地 ID”和“伙伴 ID”。因为两个站是通信伙伴，它们在连接表中的 ID 相同。点击“地址详细信息”按钮，可以查看地址的详细信息。

复选框“建立激活的连接”是默认的设置。选中该复选框时，连接表的“激活的连接伙伴”列显示“是”（见图 10-48），在运行时由本地节点建立连接，反之由通信伙伴建立连接。

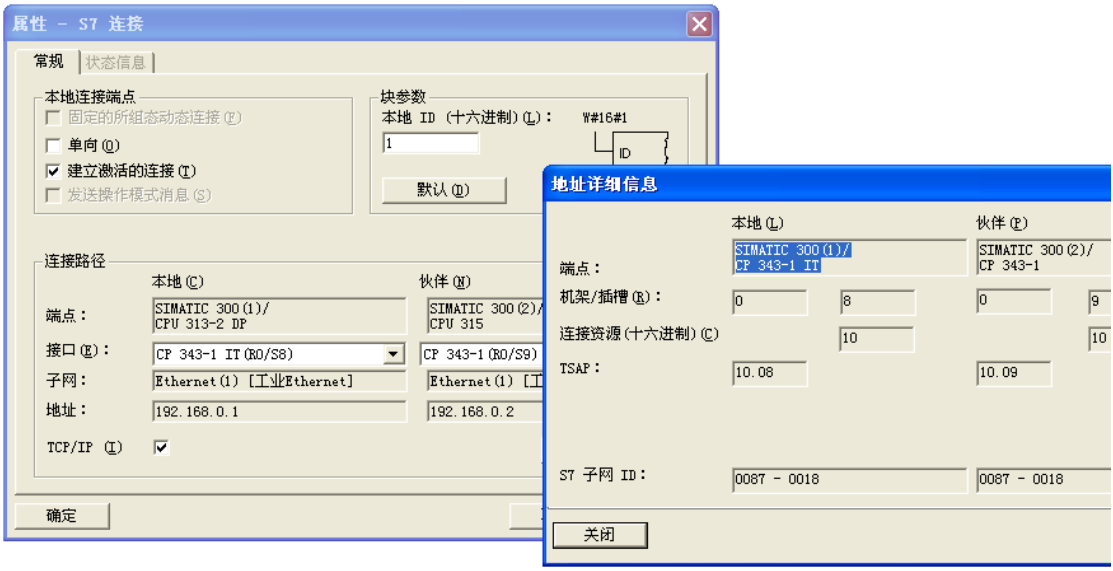


图 10-49　S7 连接属性对话框

选中 NetPro 中“SIMATIC 300（2）”站点的 CPU 315 所在的小方框，下面的窗口是自动生成的该站点一侧的连接表（见图 10-50），双击连接表中的“S7 连接”，将出现该站点一侧的连接属性对话框。

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
1	1	SIMATIC 300(1)...	S7 连接	否	Ethernet(1) [IE]

图 10-50　SIMATIC 300（2）一侧的 S7 连接

组态好连接后，点击工具栏上的  按钮，编译并保存网络组态信息。

因为是双向通信，应将通信双方的连接信息分别下载给两块 CPU。

S7-300 站点是否能作 S7 通信的客户机或建立 S7 双向连接，与 CPU 和以太网 CP 的订货号和固件版本号都有关系。本例中 CPU 315 的订货号为 6ES7 315-1AF03-0AB0，固件版本为 V1.2。CP 的订货号为 6GK7 343-1EX11-0XE0。如果将 CP 的订货号改为 6GK7 343-1EX10-0XE0，或者将 CPU 的固件版本号改为 V1.0，S7-300 站点都不能作 S7 通信的客户机或建立 S7 双向连接。

3．通信程序

通信双方的发送程序和接收程序在 OB1 中。编程时应使用组态时生成的 S7 连接的 ID 号。FB 8 和 FB 9 中的参数 R_ID 用于区分同一连接中不同的 SFB/FB 调用，发送方与接收方的 R_ID 应相同。为了区分两个方向的通信，令 CPU 313C-2DP 发送的数据包（即 CPU 315 接收的数据包）的 R_ID 为 1，CPU 315 发送的数据包的 R_ID 为 2。

提供发送请求信号的 M8.0 是周期为 100ms 的时钟存储器位。接收请求信号 EN_R 一直为 1 状态（TRUE）。如果 EN_R 的实参为一个位地址（例如 M10.0），可以用它来控制是否接收数据。S7-400 使用的 SFB 8/SFB 9 可以发送和接收 4 个数据区的数据，S7-300 使用的 FB 8/FB 9 只能发送和接收一个数据区的数据。

程序段 1： 发送数据

```
      L     ID      0
      T     DB1.DBD     2                  //用本站的 ID0 控制对方的 QD4
      CALL  "USEND" , DB8                  //调用 FB 8
        REQ     :=M8.0                     //发送请求，上升沿有效，周期为 100ms
        ID      :=W#16#1                   //S7 连接号
        R_ID    :=DW#16#1                  //发送与接收请求号
        DONE    :=M10.1                    //任务被正确执行时为 1
        ERROR   :=M10.2                    //错误标志位，为 1 时出错
        STATUS  :=MW12                     //状态字
        SD_1    :=P#DB1.DBX0.0 BYTE 100    //存放本站要发送的数据的地址区
```

程序段 2： 接收数据

```
      CALL  "URCV" , DB9                   //调用 FB 9
        EN_R    :=TRUE                     //接收请求，为 1 时接收
        ID      :=W#16#1                   //S7 连接号
        R_ID    :=DW#16#2                  //发送与接收请求号
        NDR     :=M0.1                     //任务被正确执行时为 1
        ERROR   :=M0.2                     //发送错误标志位，通信出错时为 1
        STATUS  :=MW2                      //状态字
        RD_1    :=P#DB2.DBX0.0 BYTE 100    //本站存放读取的数据的地址区
      L     DB2.DBD     2
      T     QD      4                    //用对方的 ID0 控制本站的 QD4
```

在 OB35 中，DB1.DBW0 每隔 100ms 被加 1。在初始化程序 OB100 中，用 SFC 21 预置数据发送区 DB 1 各个字的初值为 16#1111，将数据接收区 DB 2 各字节清零。

CPU 315 的程序与 CPU 313C-2DP 的基本上相同，需要注意的是，在前者的 OB1 中，USEND 的输入参数 R_ID 为 2，URCV 的 R_ID 为 1，它们的 ID（连接号）均为 1。在 OB100 中，发送区的数据字被初始化为 W#16#2222。

4．通信过程的监控

图 10-51 和图 10-52 是在运行时复制的通信双方的变量表,可以看到双方接收到的 DB2.DBW0 在不断地变化。用外接的小开关改变 ID0 的状态，通信伙伴 QD4 的状态随之而变。

图 10-51 CPU 313C-2DP 的变量表

图 10-52 CPU 315 的变量表

10.4.3 使用 BSEND/BRCV 的双向 S7 通信

1．硬件组态

使用 SFB BSEND/BRCV，可以进行快速、可靠的数据传送。在 STEP 7 中创建一个项目

（见随书光盘中的例程 IE_S7_3），生成两个站，CPU 模块分别为 CPU 313C-2DP 和 CPU 315，两个站分别使用以太网模块 CP 343-1 IT 和 CP 343-1。系统的硬件组成与项目 IE_S7_2 相同，硬件组态和连接组态的过程、以太网接口的地址与项目 IE_S7_2 也完全相同。

2．通信程序简介

通信双方的发送程序和接收程序在 OB1 中。FB 12“BSEND”和 FB 13“BRCV”的输入参数 R_ID 用于区分同一连接中不同的 SFB/FB 调用，发送方与接收方的 R_ID 应相同。为了区分两个方向的通信，令 CPU 313C-2DP 发送的数据包的 R_ID 为 1，CPU 315 发送的数据包的 R_ID 为 2。参数 ID（连接号）均为 1。

在 OB1 调用库文件夹“\SIMATIC_NET_CP\CP 300”中的 FB 12 和 FB 13，用周期为 100ms 的时钟存储器位 M8.0 作发送请求信号，每 100ms 发送一次数据。接收请求信号 EN_R 为常数 1（TRUE）。具体的程序见随书光盘中的项目 IE_S7_3，本项目的程序与 4.3.2 节的例程 PB_S7_C 中的程序基本上相同。

BSEND 的输入参数 LEN 是要发送的数据的字节数，数据类型为 WORD（无符号的字）。因为不能使用常数，其实参为 MW14，在初始化程序 OB100 中用下面两条语句预置它的初值为 250：

```
L     250
T     MW    14                    //预置要发送的字节数
```

在 OB35 中，DB1.DBW0 每隔 100ms 被加 1。在 OB100 中，用 SFC 21 预置数据发送区 DB 1 各个字的初值为 16#1111，将数据接收区 DB 2 各个字清零。

CPU 315 的程序与 CPU 313C-2DP 的基本上相同，在 OB100 中，发送区的数据被初始化为 W#16#2222。

3．通信过程的监控

图 10-53 和图 10-54 是在运行时复制的通信双方的变量表。在运行时可以看到双方接收到的 DB2.DBW0 在不断地变化，数据接收区的最后一个字 DBW248 的值与发送方预置的相同。

在运行时用外接的小开关改变 ID0 的状态，可以看到通信伙伴的 QD4 的状态随之而变。

图 10-53　CPU 313C-2DP 的变量表

图 10-54　CPU 315 的变量表

10.5　练习题

1．将工业以太网用于现场总线有什么好处？

2．实时以太网采用什么方法解决了 CSMA/CD 机制带来的冲突问题？
3．什么情况需要使用 SCALANCE 交换机？
4．MAC 地址和 IP 地址各有什么特点?
5．子网掩码有什么作用？
6．用普通网卡与 S7 CPU 通信需要作哪些准备工作？
7．基于以太网的 S5 兼容的通信服务各有什么特点？
8．哪些基于以太网的 S5 兼容的通信服务必须设置 MAC 地址？
9．哪些基于以太网的 S5 兼容的通信服务必须设置 IP 地址？
10．未指定通信伙伴的 UDP 连接怎样设置接收方的端口地址和 IP 地址？

第 11 章　PROFINET

11.1　PROFINET 通信的组态与编程

11.1.1　PROFINET 概述

1．PROFINET 简介

为了快速应对最新的市场需求，缩短产品面市的时间，需要提供从生产现场到工厂控制层和公司管理层的连续信息流，用于战略决策。现代生产对工厂纵向集成的要求越来越高，工业通信在自动化领域中的地位越来越重要。

PROFINET 是 PROFIBUS 国际组织（PI）推出的基于工业以太网的开放的现场总线标准（IEC 61158 中的类型 10）。使用 PROFINET，可以将分布式 I/O 设备直接连接到工业以太网。PROFINET 可以用于对实时性要求更高的自动化解决方案，例如运动控制。

PROFINET 吸纳了多年积累的 PROFIBUS 和工业以太网的技术诀窍，采用开放的 IT 标准，与以太网的 TCP/IP 标准兼容，并提供了实时功能，能满足所有自动化的需求。PROFINET 能与现有的现场总线系统（例如 PROFIBUS）有机地集成（见图 11-1），无需改动现有设备的组态和编程。PROFINET 通过工业以太网，连接从现场层到管理层的设备，可以实现从公司管理层到现场层的直接、透明的访问，PROFINET 融合了自动化世界和 IT 世界。

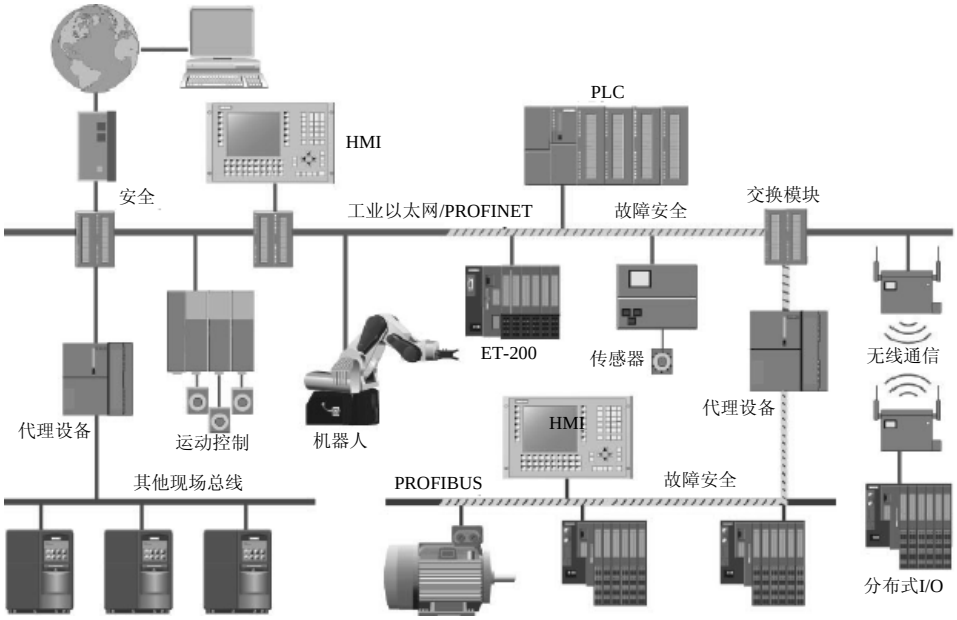


图 11-1　基于工业以太网的 PROFINET

PROFINET 技术的开放性给用户带来更多的选择余地。PROFINET 的设计和推广重点考虑了用户的易用性，用户使用 PROFINET 不需要太深的技术准备和成本。

可以很容易地从现有的 PROFIBUS 方案过渡到 PROFINET 解决方案，很好地整合已有的系统。通过代理服务器（Proxy），PROFINET 可以透明地集成现有的 PROFIBUS 设备，保护对现有系统的投资，实现现场总线系统的无缝集成。

使用 PROFINET IO，使现场设备可以直接连接到以太网，与 PLC 进行高速数据交换。其配置与组态使用控制工程师非常熟悉的 STEP 7。PROFIBUS 各种丰富的设备诊断功能同样也适用于 PROFINET。

使用故障安全通信的标准行规 PROFIsafe，PROFINET 用一个网络可以同时满足标准应用和故障安全方面的应用。PROFINET 支持驱动器配置行规 PROFIdrive，后者为电气驱动装置定义了设备特性和访问驱动器数据的方法，用来实现 PROFINET 上的多驱动器运动控制通信。

PROFINET 已经在诸如汽车工业、食品、饮料以及烟草工业和物流工业等各种行业领域得到了广泛的应用。在相当长的时间内，PROFIBUS 和 PROFINET 将会并存，并不是用 PROFINET 完全替代 PROFIBUS，因为并不是所有的工业场合都需要 PROFINET 这样先进的技术，它更多地用于基础性工业和需要复杂应用的工业控制场合。

可以对 PROFINET IO 使用标准的 SIMATIC 软件组态工具，例如用于现场级工程和诊断的 STEP 7 和用于组态运动控制的 SIMOTION Scout。

PROFINET 为分布式自动化结构开辟了新的前景，基于组件的自动化（Component Based Automation，CBA）可以实现全厂范围项目的彻底模块化。由机械组件、电气/电子组件和应用软件组成智能模块，该模块可以预先测试，交付工厂后可以立即使用。SIMATIC iMap 用图形化方式配置独立的、可以重复使用的模块之间的 PROFINET 和 PROFIBUS 数据交换。界面友好，不需要另外编程。

2．PROFINET 在实时控制中的应用

PROFINET 使用以太网和 TCP/UDP/IP 协议作为通信基础，TCP/UDP/IP 是 IT 领域通信协议事实上的标准。TCP/UDP/IP 提供了以太网设备通过本地和分布式网络的透明通道中进行数据交换的基础。对快速性没有严格要求的数据使用 TCP/IP 协议，响应时间在 100ms 数量级，可以满足工厂控制级的应用。

PROFINET 的实时（Real-Time，RT）通信功能适用于对信号传输时间有严格要求的场合，例如用于传感器和执行器的数据传输。通过 PROFINET，分布式现场设备可以直接连接到工业以太网，与 PLC 等设备通信。其响应时间比 PROFIBUS-DP 等现场总线相同或更短，典型的更新循环时间为 1～10 ms，完全能满足现场级的要求。PROFINET 的实时性可以用标准组件来实现。

PROFINET 的同步实时（Isochronous Real-Time，为 IRT）功能用于高性能的同步运动控制。IRT 提供了等时执行周期，以确保信息始终以相等的时间间隔进行传输。IRT 的响应时间为 0.25~1ms，波动小于 1μs。IRT 通信需要特殊的交换机（例如 SCALANCE X-200 IRT）的支持。等时同步数据传输的实现基于硬件。

PROFINET 的通信循环被分成两个部分，即时间确定性部分和开放性部分，循环的实时报文在时间确定性通道中传输，而 TCP/IP 报文则在开放性通道中传输。PROFINET 这种处理方法可以与高速公路媲美，最左边的车道总是为时间要求最紧迫的车辆（实时通信）保留的，以此防止其它车道上的用户（TCP/IP 通信）占用。甚至在右边车道交通堵塞的情况下，也绝不能影响时间要求紧迫的车辆的交通。

PROFINET 能同时用一条工业以太网电缆满足三个自动化领域的需求，包括 IT 集成化领域、实时（RT）自动化领域和同步实时（IRT）运动控制领域，它们不会相互影响。

使用铜质电缆最多 126 个节点，网络最长 5km。使用光纤多于 1000 个节点，网络最长 150km。无线网络最多 8 个节点，每个网段最长 1000m。

3．PROFINET 中的术语

表 11-1 给出了 PROFINET IO 与 PROFIBUS-DP 术语的比较。

表 11-1　PROFINET IO 与 PROFIBUS-DP 术语的比较

特　　性	PROFINET IO	PROFIBUS-DP
子网名称	以太网	PROFIBUS
子系统名称	I/O 系统	DP 主站系统
主站设备名称	I/O 控制器	DP 主站
从站设备名称	I/O 设备	DP 从站
硬件目录	PROFINET IO	PROFIBUS DP
编号	设备编号	PROFIBUS 地址（与站编号对应）
操作参数与诊断地址	在插槽 0 中接口模块的对象属性中列出	在站的对象属性中列出

4．PROFINET IO 系统

PROFINET 是实现模块化、分布式应用的通信标准。PROFINET IO 具有标准的接口，可以将分布式现场 I/O 设备直接连接到工业以太网。

PROFINET IO 系统由 I/O 控制器和 I/O 设备组成。I/O 控制器是 PROFINET 上的主动节点，它与 I/O 设备进行循环数据交换。I/O 设备是 PROFINET 上的被动站点。

PROFINET IO 与 PROFIBUS 提供的设备模型相同，使用相同的工程系统（例如 STEP 7）对它们组态，它们的属性都用 GSD 文件描述。组态时将现场 I/O 设备分配给一个 IO 控制器。可以使用有代理功能的 PROFINET 设备（例如 IE/PB 链接器），将现有的 PROFIBUS 系统无缝地集成到 PROFINET 中（见图 11-2），以保护现有系统的投资。

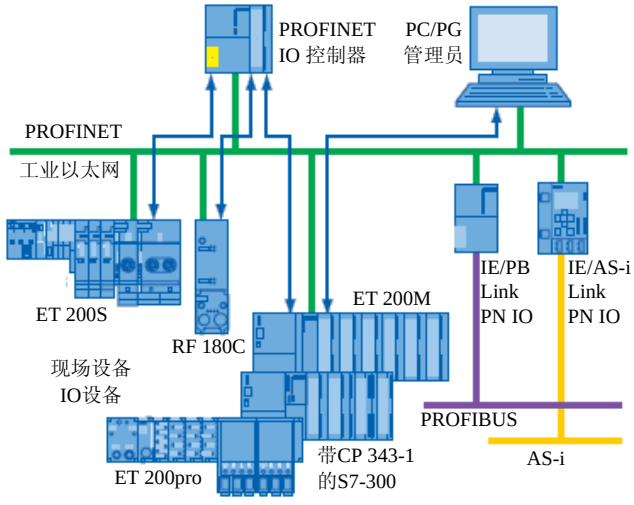


图 11-2　PROFINET

5．PROFINET IO 控制器

1）CPU 315-2DP/PN、CPU 317-2DP/PN 和 CPU 319-3DP/PN：用于处理过程信号和直接将现场设备连接到工业以太网。

2）CP 343-1/CP 343-1 Advanced 和 CP 443-1 Advanced：用于将 S7-300 和 S7-400 连接到 PROFINET。CP 443-1 Advanced 带有集成的 Web 服务器和集成的交换机。

3）IE/PB LINK PN IO：将现有的 PROFIBUS 设备透明地连接到 PROFINET 的代理设备。

4）IWLAN/PB LINK PN IO：通过无线方式将 PROFIBUS 设备透明地连接到 PROFINET 的代理设备。I/O 控制器可以通过代理设备来访问 DP 从站，就像访问 I/O 设备一样。

5）IE/AS-i Link：将 AS-i 设备连接到 PROFINET 的代理设备。

6）CP 1616：用于将 PC 连接到 PROFINET，是带有集成的 4 端口交换机的通信处理器。支持同步实时模式，可以用于运动控制领域对时间要求严格的同步闭环控制。

7）SOFT PN IO：作为 IO PLC，在编程器或 PC 上运行的通信软件。

6．PROFINET IO 设备

1）接口模块为 IM 151-3 PN 的 ET 200S。

2）接口模块为 IM 153-4 PN 的 ET 200M。

3）接口模块为 IM 154-4 PN 的 ET 200pro。

4）ET 200eco PN。

5）SIMATIC HMI。

11.1.2 基于 CPU 集成的 PN 接口的 PROFINET 通信

就像集成了 DP 接口的 S7 CPU 可以直接访问标准 DP 从站一样，带 PROFINET 通信接口的 S7 CPU 可以直接访问 PROFINET IO 设备。与使用 PROFINET CP 模块的方案相比，使用带 PROFINET 通信接口的 CPU 作 IO 控制器的硬件成本低、通信的编程工作量极少，应作为 PROFINET 控制器的首选。本节介绍基于 CPU 315-2PN/DP 的 PROFINET 通信的组态与编程方法。

1．网络结构

网络结构如图 11-3 所示，CPU 315-2PN/DP 是 PROFINET 控制器，以太网上有两个 PROFINET IO 设备：

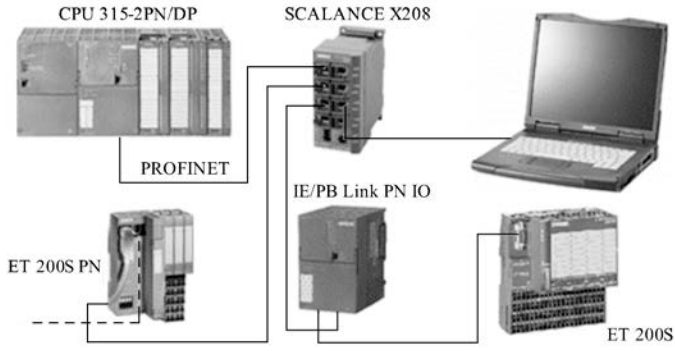


图 11-3 系统硬件与接线

1）一台 ET 200S PN，其接口模块为 IM 151-3PN。IM 151-3PN 需要配 MMC 卡，没有

DIP 开关，用 STEP 7 设置和下载站地址。

2）一块连接以太网和 PROFIBUS 的 IE/PB Link（IE/PB 链接器），它作为 PROFINET IO 代理服务器（Proxy），同时又是 DP 主站。DP 网络上的 DP 从站为 ET 200S，接口模块的订货号为 IM 151-1AA04-0AB0，电源模块和 I/O 模块的详细信息见图 11-4。IM 151-1 不需要配 MMC 卡，用模块上的 DIP 开关设置从站地址。

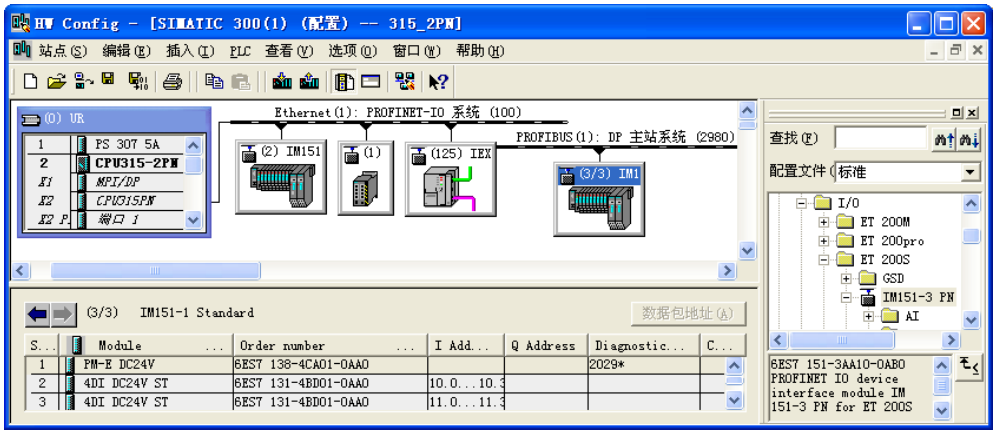


图 11-4　硬件与网络组态

CPU 315-2PN/DP 的集成 PN 口、ET 200S PN、IE/PB Link 和笔记本电脑通过 RJ 45 连接器和以太网线，连接到工业网络管理型交换机 SCALANCE X208 上（见图 11-3）。

2．组态 PROFINET IO 控制器

在 STEP 7 管理器中创建一个名为“315_2PN”的项目（见随书光盘中的同名例程），生成一个 S7-300 站点。打开硬件组态工具 HW Config（见图 11-4），将硬件目录中的导轨拖放到左边的组态窗口，将订货号为 315-2EH13-0AB0 的 CPU 315-2PN/DP 拖放到 2 号槽。

在自动打开的以太网接口属性对话框的“参数”选项卡（见图 11-5）中，点击“新建”按钮，生成一条名为“Ethernet（1）”的以太网，点击“确定”按钮，返回“参数”选项卡，将 CPU 连接到该网上。设置 IP 地址为 192.168.0.2，采用默认的子网掩码 255.255.255.0，点击“确定”按钮，返回 HW Config。可以看到生成的 Ethernet（1）PROFINET-IO 系统（100）。将电源模块和信号模块插入主机架。

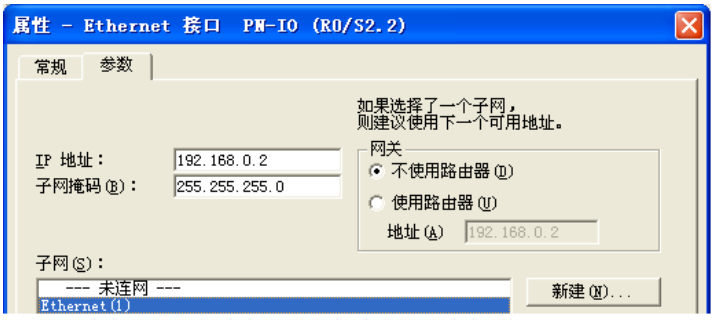


图 11-5　CPU 的以太网接口属性对话框

3．组态 ET 200S PN

打开 HW Config 右边的硬件目录窗口中的文件夹“\PROFINET IO\I/O\ET 200S”，将其中

订货号为 IM 151-3AA10-0AB0 的接口模块 IM 151-3PN 拖放到以太网上（见图 11-4）。

双击刚生成的站，点击打开的 IM151-3PN 属性对话框中的“Ethernet”按钮（见图 11-6 的左图），在出现的 Ethernet 接口属性对话框中（见图 11-6 的右图），设置 IP 地址为 192.168.0.4。点击“确定”按钮返回 IM151-3PN 属性对话框。可以修改图中的 Device Name（设备名称），在分配设备名称时将会用到它。

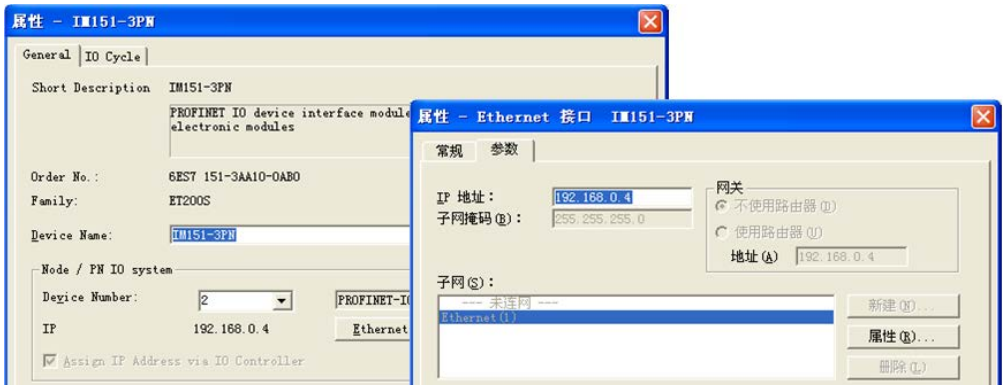


图 11-6　IM151-3PN 属性对话框

STEP 7 按照组态的先后次序，自动分配以太网上各个 IO 设备的编号（Device Number），但是它的重要性不像 DP 网络中的站地址那样大，因为各个 IO 设备除了编号外，还有 MAC 地址、IP 地址，和后面要分配的 IO 设备名称。

返回 HW Config 后，打开硬件目录中的子文件夹“IM 151-3PN”，将其中的子文件夹 PM 中的电源模块、子文件夹 DI 和 DO 中的数字量输入、数字量输出模块拖放到下面的插槽中。在组态时可以看出，主机架、以太网和 DP 网络中各个从站的输入、输出模块的地址是自动统一分配的，没有重叠区，CPU 可以用 I/O 地址直接访问各个站的 I/O 模块。因为使用的 IM 151-3 PN 的版本较低，需要设置的参数很少。

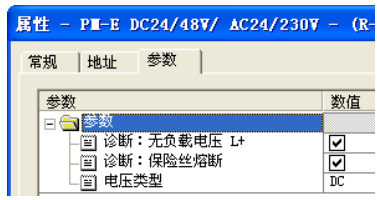


图 11-7　电源模块组态

双击下面窗口中 ET 200S PN 的电源模块，打开它的属性对话框，在参数选项卡设置它的参数（见图 11-7）。用同样的方法设置 DI 模块和 DO 模块的参数（见图 11-8 和图 11-9）。

图 11-8　ET 200S DI 模块组态

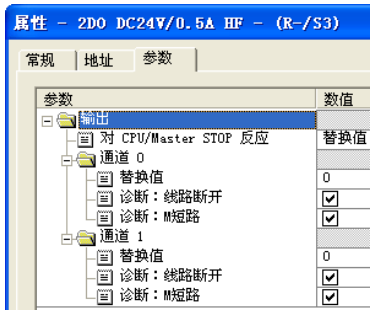


图 11-9　ET 200S DO 模块组态

4．组态 IE/PB Link

IE/PB Link PN I/O 作为 PROFINET 代理服务器，用于将现有的 PROFIBUS 设备作为 I/O

设备透明地接入 PROFINET。

打开 HW Config 右边的硬件目录窗口中的文件夹“\PROFINET IO\Gateway(网关)\IE/PB Link PN IO”,将其中订货号为 6GK1 411-5AB00 的 IE/PB Link 拖放到以太网上(见图 11-4)。点击自动打开的 PROFIBUS 接口属性对话框中的“新建”按钮，采用默认的参数，新建一条 PROFIBUS 网络。IE/PB Link 的 DP 站地址为 2，网络传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮，返回 HW Config。可以看到自动生成了一个 DP 主站系统（见图 11-4)。

双击打开该站的属性对话框，设备名称采用默认的“IEXPBXLink”。点击“Ethernet”按钮，在 Ethernet 接口属性对话框中，设置 IP 地址为 192.168.0.6。

5．组态 DP 网络中的 ET 200S

打开 HW Config 右边的硬件目录窗口中的文件夹“\PROFIBUS DP”，将其中订货号为 6ES7 151-1AA04-0AB0 的 IM 151-1 Standard 拖放到 DP 网络上。在自动打开的 PROFIBUS 接口属性对话框中，将 DP 站地址设置为 3。

返回 HW Config 后，双击 IM 151-1，在它的属性对话框的“Operating Parameters”（操作参数）选项卡中，设置 DP 中断模式为 DPV1，以及与诊断有关的参数（见图 11-10)。

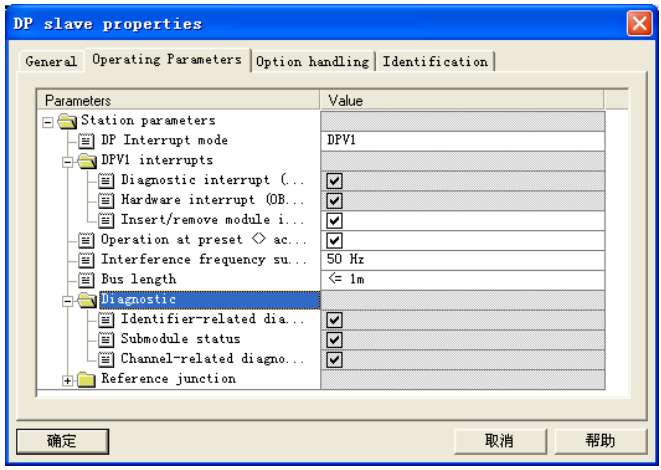


图 11-10　IM 151-1 属性对话框

返回 HW Config 后,选中 IM 151-1,打开硬件目录窗口中的子文件夹“IM 151-1 Standard”,将其中的电源模块（PM)、数字量输入模块（DI）和数字量输出模块（DO）拖放到下面窗口的插槽中。双击电源模块，激活它的“无负载电压 L+”的诊断功能。

2～5 号槽的 DI、DO 模块没有诊断功能，不需要设置什么参数。6 号槽是高性能 DI 模块，其参数设置见图 11-8。

6．组态交换机 SCALANCE X208

SCALANCE X208 是网络管理型工业以太网交换机，用来连接网络中的各个站。

打开 HW Config 右边窗口的硬件目录中的文件夹“\PROFINET IO\Network Components \SCALANCE X-200\SCALANCE X208”，将其中的“V2.0”拖放到左边窗口的 Ethernet 网络线上。在打开的属性对话框中，设置 IP 地址为 192.168.0.5，设备名称为“X208”。

组态结束后，点击工具栏上的按钮，编译与保存组态信息。

7．下载硬件组态信息

可以用计算机的普通以太网卡来下载组态信息和进行监控操作。为此首先用 MPI 接口将组态信息（例如 CPU 的 IP 地址和 MAC 地址）下载到 CPU，为下一步用以太网下载和监控控制系统打好基础。作者做实验使用的是笔记本电脑，用 USB/MPI 适配器连接笔记本电脑的 USB 接口和 CPU 的 MPI 接口。下载之前，应安装好适配器的 USB 驱动程序。

执行 SIMATIC 管理器的菜单命令“选项”→“设置 PG/PC 接口”，设置 USB/MPI 适配器的接口参数为“PC Adapter（MPI）”。点击（下载）按钮，下载硬件组态信息。

8．组态笔记本电脑的以太网接口

（1）设置 PG/PC 接口

用 MPI 成功下载组态信息后，在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PG/PC 接口”，选中出现的对话框（可参考图 10-10）中的“TCP/IP（Auto）”，使用计算机集成的以太网卡。点击“确定”按钮，出现“下列访问路径已更改”的警告信息。点击“确定”按钮，退出“设置 PG/PC 接口”对话框后，TCP/IP 协议生效。

（2）设置计算机网卡的 IP 地址

计算机的网卡与 CPU 的以太网接口的 IP 地址应在同一个网段内，它们应使用相同的子网掩码。子网的网段地址一般采用 192.168.0，子网掩码为 255.255.255.0。

打开计算机的控制面板，双击其中的“网络连接”图标。在“网络连接”对话框中，双击“本地连接”图标，打开“本地连接属性”对话框（参见图 10-13）。用“此连接使用下列项目”选择框，选中“Internet 协议（TCP/IP）”，点击“属性”按钮，打开“Internet 协议（TCP/IP）属性”对话框。用单选框选中“使用下面的 IP 地址”，然后按上述的原则设置网卡的 IP 地址和子网掩码。设置结束后，点击各级对话框中的“确定”按钮，最后关闭“网络连接”对话框。

9．编辑以太网节点

在 HW Config 中执行菜单命令“PLC”→“Ethernet”→“编辑 Ethernet 节点”，打开编辑以太网节点对话框（见图 11-11）。点击“浏览”按钮，出现“浏览网络”对话框。等待几秒钟后，可以看到搜索到的以太网上的节点（不包括计算机本身）的信息。

图 11-11　编辑以太网节点对话框

选中其中的某个节点，点击“确定”按钮，返回到“编辑以太网节点”对话框，可以看到选中的节点的 MAC 地址和 IP 地址、子网掩码和设备名称出现在对话框中。可以在对话框中修改上述的设备参数。点击“分配 IP 组态”和“分配名称”按钮，可以作相应的操作。操作成功后，将会显示“参数已成功传送”的信息。

10．分配 IO 设备的名称

在 PROFINET 通信中，各 IO 设备是用设备名称来识别的，因此在组态时应为每个设备分配好设备名称，并将它们下载到 CPU。

首先应在 SIMATIC 管理器执行菜单命令“选项”→“设置 PG/PC 接口”，将 PG/PC 接口使用的协议设置为 TCP/IP。设置好后，才能执行下面的菜单命令。

在 HW Config 中执行菜单命令“PLC”→“Ethernet”（以太网）→“分配设备名称”，打开“分配设备名称”对话框（见图 11-12）。

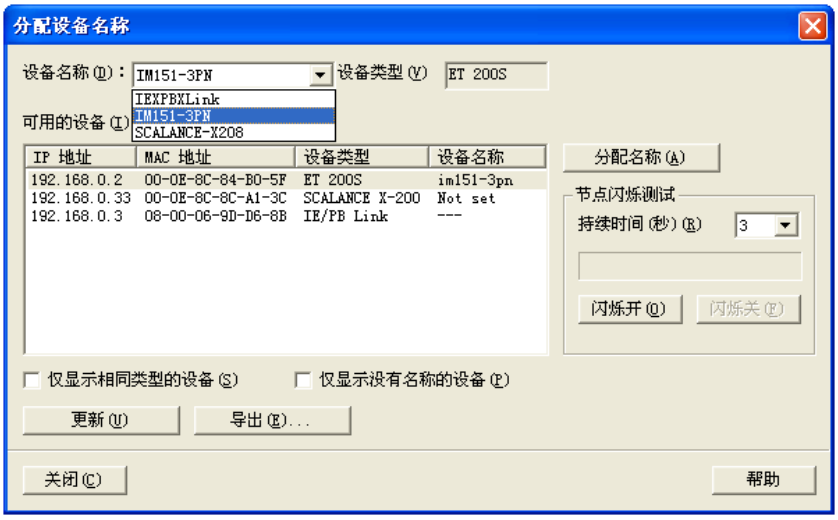


图 11-12　分配设备名称对话框

对话框上面的“设备名称”选择框给出了 STEP 7 已组态和编译的设备名称。在“可用的设备”列表中，列出了 STEP 7 搜索到的以太网子网上所有可用的 IO 设备，包括在线获得的各设备的 MAC 地址和设备类型、原有的 IP 地址和设备名称（如果有的话）。

下面是分配设备名称的操作步骤：

1）用“设备名称”选择框选中设备名称。

2）选中“可用的设备”列表中需要分配名称的 IO 设备；

3）点击“分配名称”按钮，“设备名称”选择框指定的设备名称被分配给“可用的设备”列表选中的 IO 设备。新分配的设备名称显示在“可用的设备”列表的“设备名称”列中。

如果不能确认“可用的设备”列表中的 MAC 地址对应的硬件 I/O 设备，选中该列表中某个设备，例如 ET 200S PN，点击“闪烁开”按钮，IM 151-3PN 上绿色的 Link LED 闪烁，可以将闪烁持续的时间设置在 3～60s 之间。闪烁时“持续时间”下面的进度条显示闪烁的时间。点击“闪烁关”按钮，将会提前停止闪烁。

11．验证设备名称

分配完设备名称后，执行菜单命令“PLC”→“Ethernet”→“验证设备名称”，在出现的对话框中（见图 11-13），分配的设备名称如果与组态的名称符合，状态列显示绿色的“√”，如果不符合，显示红色的“×”。点击“分配名称”按钮，将打开“分配设备名称”对话框。

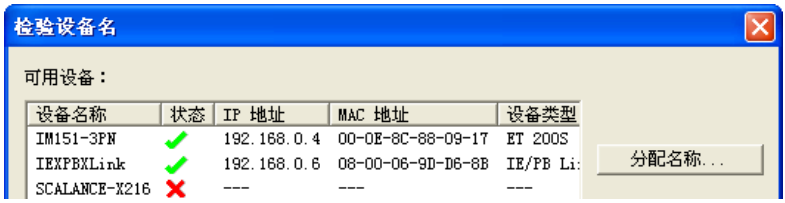


图 11-13　验证设备名称对话框

12．程序设计

与 PROFIBUS-DP 通信相同，为了保证网络系统的正常运行，需要为 S7-300 生成 OB82、OB86 和 OB122，S7-400 还需要生成 OB85。如果在硬件组态时组态了硬件中断，需要生成 OB40。如果 CPU 或 IO 设备有带电插入/拔出模块的功能，还需要生成插入/拔出模块中断组织块 OB83。本例程生成了 OB40、OB82、OB83、OB86 和 OB122。

该项目中的故障诊断程序的设计方法见 11.2.1 节。

13．验证 PROFINET 通信

上述操作全部成功完成后，将程序和组态信息下载到 CPU。

系统断电后再上电时，IM 151-3PN 的 BF LED 闪烁，PM、DI、DO 模块的 SF LED 亮。IE/PB Link 的 SF LED 亮，BF PN 和 BF DP LED 闪烁。过几秒后 CPU 的 BF2 LED 闪烁，SF LED 亮。最后所有设备的红色故障 LED 熄灭，绿色的 RUN LED 亮。

可以用变量表来监视 CPU 与 PROFINET IO 设备的通信，也可以用下面的方法来验证通信是否成功。用右键点击离线的 HW Config 中 ET 200S PN 的 2DO 模块，执行出现的快捷菜单中的“监视/修改”命令，点击出现的“监视/修改”对话框中的“状态值”按钮（见图 11-14），表格中的“状态值”列显示出各输出点的状态。

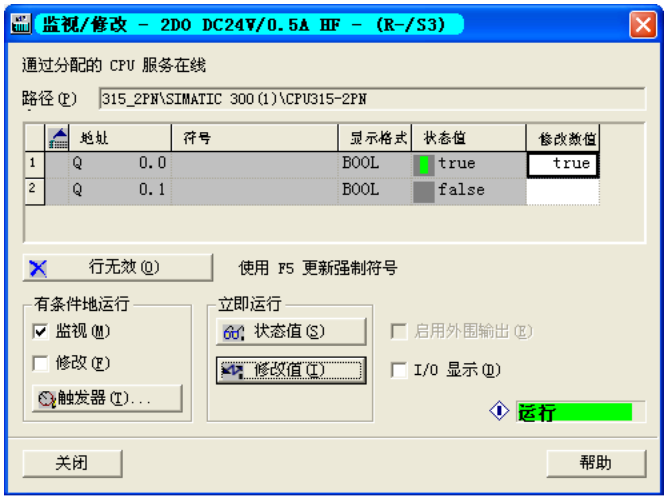


图 11-14　监视/修改对话框

在 Q0.0 的“修改数值”列中输入 1，用鼠标点击对话框后变为 true。点击“修改值”按钮，状态值列出现绿色矩形指示灯符号，状态值由 false（0）变为 true（1）。如果 ET 200S PN 的 2DO 模块上 Q0.0 对应的 LED 亮，表示 CPU 与 IO 设备的通信正常。

用同样的方法监控 ET 200S PN 的 2DI 模块（见图 11-15）。点击“状态值”按钮，“状态值”列显示各输入点的状态。用模块输入端外接的小开关改变 I2.0 的输入电路的通、断状态，对话框中 I2.0 的状态值应随之而变。以上操作可以验证 PROFNET IO 设备与 CPU 的通信是否正常。

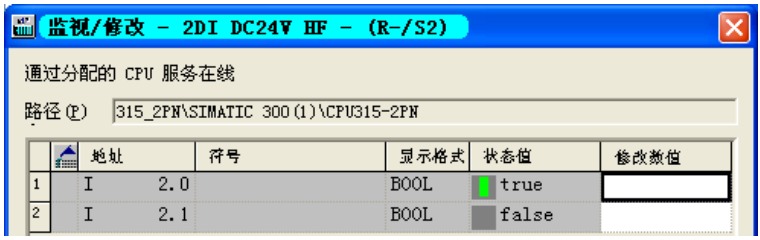


图 11-15　监视 ET 200S PN 的 2DI 模块

14．SCALANCE X204-2 的模块信息对话框

不用在 HW Config 中组态，就可以使用 SCALANCE 系列交换机。但是如果对它组态，则可以查看网络的运行情况。

下面以另一个项目的 SCALANCE X204-2 为例，介绍用交换机诊断网络的方法。X204-2 有 4 个 RJ-45 电气接口，端口号为 P1～P4。还有两个光纤接口，端口号为 P5 和 P6。

拔出接在交换机的 P3 接口上的 IM 151-3PN 的 RJ-45 接头，CPU 的 BF2 LED 闪烁，SF LED 亮。IE/DP Link 的 SF 和 BF PN LED 亮，BF DP LED 闪烁。

点击 HW Config 工具栏上的按钮，显示在线诊断视图，双击其中的交换机，打开它的模块信息对话框（见图 11-16）。可以看到端口 3 断开的信息。选中端口 1，“端口详细情况”列表的“接口 MAC 地址”是 X204-2 的地址，“相邻设备”是指该端口连接的 IM 151-3PN。

图 11-16　X204-2 的模块信息对话框

15．用 IE 浏览器查看 SCALANCE X204-2 的状态

离线打开 X204-2 的属性对话框，点击“管理”选项卡中的“基于 web 的管理”按钮（见图 11-17），出现登陆对话框（见图 11-18），自动生成的用户名为 admin，初始密码也是 admin。输入密码后，点击“Log On”（登录）按钮，自动打开 IE，显示交换机的网页，其地址为组态的 192.168.0.4（见图 11-19）。

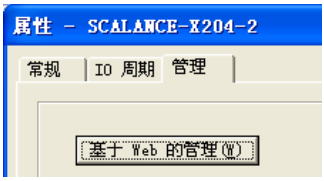


图 11-17　X204-2 属性对话框

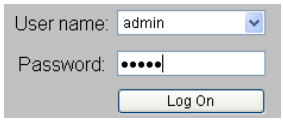


图 11-18　登录对话框

断开连接 IE/PB Link 的 X204-2 的 P3 接口上的电缆。选中左边窗口的“Event Log”（事件记录），右边窗口的事件列表中的第一条信息是端口 3 的链接断开（见图 11-19）。选中左边窗口中的“Ports”（端口），右边窗口第 3 行 Port3 的 Link 列为“down”（见图 11-20），表示端口 3 没有接线。

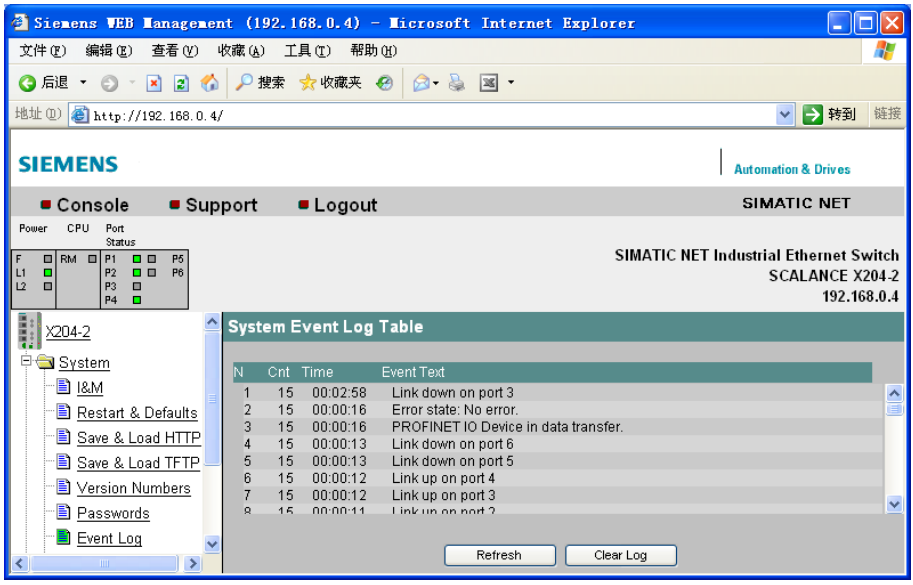


图 11-19　X204-2 的事件记录表

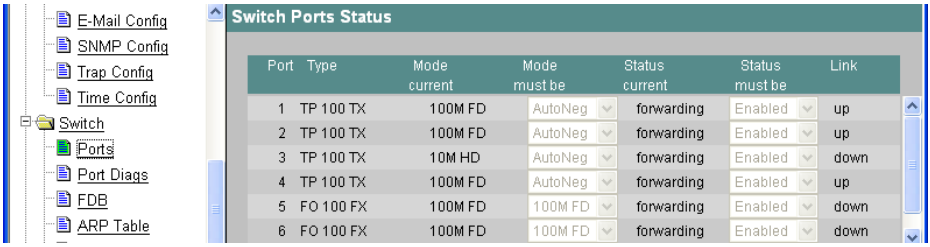


图 11-20　204-2 的接口状态

选中左边窗口中的 Port Diags（端口诊断），右边窗口是有故障的 Port 3 的端口诊断画面。

11.1.3 基于 CP 343-1 的 PROFINET 通信

1．硬件与网络组态

在 STEP 7 管理器中创建一个名为“CP343_1”的项目（见随书光盘中的同名例程），生成一个 S7-300 站点。在 HW Config 中，将 CPU 315-2DP 拖放到 2 号槽，将电源模块和信号模块插入机架。

基于 CP 343-1 的 PROFINET 通信的网络结构如图 11-21 所示，CP 343-1 是 PROFINET 控制器。与项目 315_2PN 相同，以太网上的 PROFINET IO 设备为 ET 200S PN 和 IE/PB Link。DP 网络上的 ET 200S 站点的配置见图 11-21。

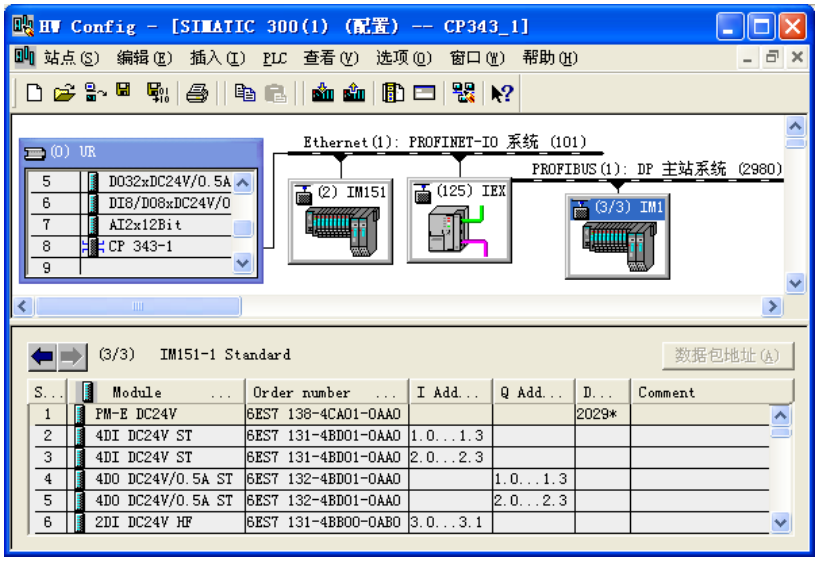


图 11-21　硬件与网络组态

CP 343-1、ET 200S PN、IE/PB Link 和笔记本电脑分别通过 RJ 45 连接器和网线连接到交换机 SCALANCE X208 上，没有对后者组态。CP 343-1、IM 151-3PN、IE/PB Link 的 IP 地址为 192.168.0.1～192.168.0.3，计算机的 IP 地址为 192 .168.0.20。IE/PB Link 的 DP 地址为 2，ET 200S 的 DP 地址为 3。

组态时自动分配的 ET 200S PN 的 2DI、2DO 的地址在 0 号字节，DP 网络上的 ET 200S 的 DI 模块的地址在 1～3 号字节，DO 模块的地址在 1、2 号字节。以太网与编号为 2980 的 DP 主站系统的 IO 设备的输入、输出地址（IB0～IB3 和 QB0～QB2），与 CPU 主机架上的 DI、DO 模块的地址 ID0、QD0 重叠。以太网上 I/O 模块的地址是独立于 CPU 的地址系统的，需要调用 FC 11 和 FC 12 来访问它们。PROFINET 和 PROFIBUS 网络上的 I/O 设备的地址是组态时 STEP 7 自动统一分配的。

组态结束后，点击工具栏上的按钮，编译与保存组态信息。

执行 SIMATIC 管理器的菜单命令“选项”→“设置 PG/PC 接口”，设置 USB/MPI 适配器的接口参数为“PC Adapter（MPI）”。用 USB/MPI 适配器连接 CPU 和笔记本电脑，点击

（下载）按钮，将组态信息下载到 CPU。下载成功后，在 SIMATIC 管理器中执行菜单命令“选项”→“设置 PG/PC 接口”，选中使用 TCP/IP 协议的计算机集成的网卡。点击“确定”按钮，退出该对话框后，TCP/IP 协议才会生效。

用以太网电缆将 CP 343-1、计算机和 PROFINET IO 设备连接到交换机上。在 HW Config 执行菜单命令“PLC”→“Ethernet”→“分配设备名称”，打开分配设备名称对话框。用前面介绍的方法分配各 PROFINET IO 设备的设备名称，图 11-22 是分配成功后的对话框。

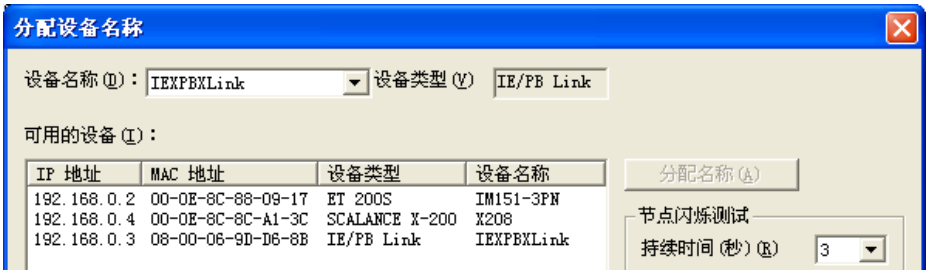


图 11-22　分配设备名称对话框

在 HW Config 执行菜单命令“PLC”→“Ethernet”→“验证设备名称”，因为没有组态 SCALANCE X208 交换机，“检验设备名”对话框中它是不可见的（见图 11-23）。

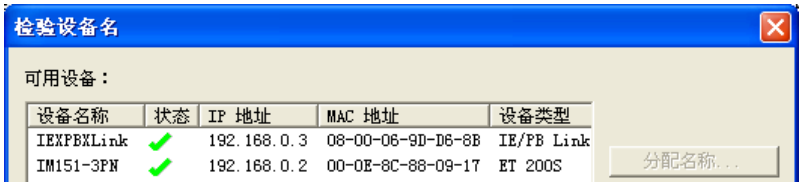


图 11-23　验证设备名称对话框

2．程序设计

本例程的程序结构与项目 315_2PN 的相同，需要生成 OB40、OB82、OB83、OB86 和 OB122。以 CP 343-1 为控制器的 PROFINET IO 系统的地址是独立于 CPU 的地址系统的，需要调用程序编辑器左边窗口中的“\库\SIMATIC_NET_CP\CP300”中的 FC 11 和 FC 12 来访问它们。这种访问方式类似于调用 FC 1 和 FC 2 来访问 CP 342-5 的 DP 从站。在 OB1 中编写下面的程序：

```
CALL    "PNIO_RECV"                  //调用 FC 12
  CPLADDR      :=W#16#140            //CP模块的起始地址
  LEN          :=4                   //实际接收的数据字节数
  IOPS         :=P#M 30.0 BYTE 4     //IO消费者接收的IO提供者每个输入字节对应的位状态
  NDR          :=M1.0                //通信正确完成时为 1
  ERROR        :=M1.1                //错误标志位
  STATUS       :=MW34                //状态字
  CHECK_IOPS :=M1.2                  //为 1 时需要对IOCS状态作进一步分析
  ADD_INFO     :=MW36                //附加诊断信息，当前FC版本为 0，为以后扩展用
  RECV         :=P#M38.0 BYTE 4      //保存接收到的数据的地址区
CALL    "PNIO_SEND"                  //调用 FC 11
```

```
CPLADDR    :=W#16#140          //CP模块的起始地址
LEN        :=3                 //实际发送的数据字节数
IOCS       :=P#M 42.0 BYTE 3   //对方IO消费者提供给IO提供者输出字节对应的位状态
DONE       :=M0.0              //通信正确完成时为 1
ERROR      :=M0.1              //错误标志位
STATUS     :=MW45              //状态字
CHECK_IOCS :=M0.2              //为 1 时需要对IOCS状态作进一步分析
SEND       :=P#M 47.0 BYTE 3   //存放要发送的数据的地址区
```

3．验证 PROFINET 通信

用变量表监视 IO 设备各 I/O 模块上的输入、输出点（见图 11-24）。用模块输入端外接的小开关改变 ET 200S PN 的 I0.0 的输入电路的通、断状态，变量表中输入点 I0.0 对应的 M38.0 的状态值随之而变。在输出点对应的 M47.0 等的“修改数值”列输入 1，用鼠标点击变量表后变为 true，点击工具栏上的按钮，修改值被下载到 CPU，“状态值”列的 false 变为 true，对应的输出模块上的 LED 亮。以上操作说明 PROFNET IO 设备与 CPU 的通信正常。

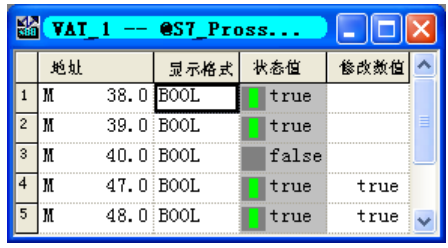


图 11-24　用变量表监视 PROFINET IO

11.1.4　基于 CP 443-1 的 PROFINET 通信

1．硬件与网络组态

创建一个名为“CP443_1”的项目（见随书光盘中的同名例程），网络结构如图 11-25 所示，CP 443-1 是 PROFINET 控制器。与项目 315_2PN 相同，以太网上的 PROFINET IO 设备为 ET 200S PN 和 IE/PB Link。CP 443-1 有 4 个 RJ-45 接口，相当于自带一个有 4 个端口的交换机。ET 200S PN、IE/PB Link 和笔记本电脑直接连接到 CP 443-1 的 RJ-45 连接器上。

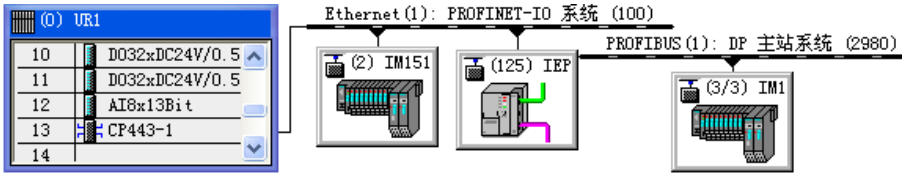


图 11-25　硬件与网络组态

CP 443-1、ET 200S PN 和 IE/PB Link 的 IP 地址分别为 192.168.0.1～192.168.0.3，计算机的 IP 地址为 192 .168.0.20。IE/PB Link 的 DP 地址为 2，ET 200S 的 DP 地址为 3。

组态时可以看出，主机架、以太网和 DP 网络上各从站的 I/O 模块的地址是自动统一分配的，没有重叠区。就像 CPU 集成的 PROFINET 接口一样，CPU 通过 CP 443-1，用 I/O 地址直接访问各远程站的 I/O 模块。组态结束后，点击工具栏上的按钮，编译与保存组态信息。

执行 SIMATIC 管理器的菜单命令“选项”→“设置 PG/PC 接口”，设置 USB/MPI 适配器的接口参数为“PC Adapter（MPI）”。用 USB/MPI 适配器连接 CPU 和笔记本电脑，点击（下载）按钮，将组态信息下载到 CPU。下载成功后，在 SIMATIC 管理器中执行菜单命令“选

项”→“设置 PG/PC 接口”，选中使用 TCP/IP 协议的计算机集成的网卡。点击“确定”按钮，退出该对话框后，TCP/IP 协议才会生效。

在 HW Config 中执行菜单命令“PLC”→“Ethernet”→“分配设备名称”，打开分配设备名称对话框。用前面介绍的方法分配各 PROFINET IO 设备的设备名称，图 11-26 是分配成功后的对话框。

图 11-26　分配设备名称对话框

2．PROFINET 通信的编程与验证

本例程的程序结构与项目 315_2PN 的基本上相同，需要生成 OB40、OB82、OB83、OB85、OB86 和 OB122。在各 OB 中，分别将 MW50～MW60 加 1。

可以用 11.1.2 节介绍的方法，来验证是否实现了 CP 443-1 与 PROFINET IO 设备的通信。

11.2　PROFINET 的故障诊断

11.2.1　PROFINET 通信故障诊断的编程

本章的 3 个 PROFINET 例程的通信故障诊断程序基本上相同，下面介绍项目 315_2PN 的诊断程序。

1．中断组织块

在 OB40、OB82、OB83、OB85、OB86 和 OB122 中，分别将 MW50～MW60 加 1。系统运行时用它们来监控是否产生中断，和中断产生的次数。

在 OB40、OB82、OB83、和 OB86 中调用 SFC 20“BLKMOV”，将 OB 的局部变量分别保存到 DB 1～DB 4，以供进一步分析时使用。下面是 OB86 中的程序：

```
程序段 1：计中断次数
        L     MW    58
        +     1
        T     MW    58
程序段 2：保存 OB86 的局部变量
        CALL   "BLKMOV"
          SRCBLK  :=P#L 0.0 BYTE 20
          RET_VAL  :=MW50
          DSTBLK  :=DB4.ARY
```

DB 4 中的 ARY 是有 5 个双字的数组，用来保存 20B 局部变量。

2．在 OB1 中调用 SFB 52 读取数据记录

在 PROFIBUS-DP 的故障诊断中，用 SFC 13 来读取 DP 从站的诊断数据。PROFINET 通信的故障诊断不能使用 SFC 13，需要在 OB1 中调用 SFB 52，来读取用于诊断的数据记录。在 OB82 中调用 SFB 54，读取 OB 的启动信息和中断源（PROFINET IO 设备）的信息。

调用 SFB 52“RDREC”（读取数据记录），可以从 DP 从站或 PROFINET IO 设备组件（模块或子模块）中，读取用参数 INDEX 指定编号的数据记录。SFB 52 以异步方式工作，处理过程需要多次调用 SFB 52。输入参数 REQ 为 1 时传送数据记录。对于输出模块，应将输入参数 ID（逻辑地址）的第 15 位置 1。对于输入/输出组合模块，应采用两个地址中较小的地址。

SFB 52 的输入参数 INDEX 用于指定数据记录号，编号为 16#800A 的数据记录是故障模块（或子模块）的通道诊断数据。下面的程序用来读取 ET 200S PN 的 DO 模块的诊断信息，该模块的地址为 QB0，因为是输出模块，参数 ID 的第 15 位为 1，所以 ID 为 DW#16#8000。

SFB 52 的输入参数 MLEN 用于指定要读取的数据记录的最大字节数，地址区 RECORD 的长度至少应等于 MLEN 字节的长度。如果输出参数 VALID 为 TRUE，表示已将数据记录成功地传送到了目标区域 RECORD。此时，输出参数 LEN 是读取到的数据记录的字节数。

输出参数 ERROR 用来指示数据记录传输是否出错。如果 ERROR 为 1，输出参数 STATUS 中是错误信息，其 2、3 字节返回请求状态。输出参数 BUSY 为 0 表示数据记录传送已完成。下面是 OB1 中的程序：

程序段 1：

```
AN    M      10.5                    //如果读取操作已完成（BUSY 标志为 0）
AN    M      10.6                    //如果未启动读取数据记录（REQ 为 0）
S     M      10.6                    //令 REQ 为 1，传输数据记录
L     W#16#800A                      //数据记录号为 W#16#800A
T     MW     6                       //将记录号传送到输入变量 INDEX
CALL  "RDREC" , DB52                 //调用 SFB 52
  REQ     :=M10.6                    //为 1 时读取数据记录
  ID      :=DW#16#8000               //ET 200S PN的DO模块地址为QB0，第 15 位为 1
  INDEX   :=MW6                      //数据记录号W#16#800A
  MLEN    :=100                      //要读取的数据记录的最大长度为 100B
  VALID   :=M10.4                    //为 1 时新数据记录已被接收且有效
  BUSY    :=M10.5                    //为 1 时读取过程尚未结束
  ERROR   :=M10.7                    //错误标志位
  STATUS  :=MD12                     //调用ID（第 2、3 字节）或错误代码
  LEN     :=MW16                     //读取的数据记录信息的长度
  RECORD  :=P#DB5.DBX0.0 BYTE 100    //存放读取的数据记录的目标地址区
A     M      10.6
R     M      10.6                    //复位读请求信号 REQ
```

PROFINET IO 系统的诊断数据记录的详细信息见随书光盘中的文件《从 PROFIBUS DP 到 PROFINET IO 编程手册》的第 5 章。

3．在 OB82 中调用 SFB 54 进行诊断

有的 DP 从站、PROFINET IO 设备或从站中的模块具有中断功能，中断组织块（OB）的局部数据提供了中断时产生的部分诊断数据。在中断 OB 中调用 SFB 54“RALRM”，可以读

取与事件相关的（例如由错误 OB 触发的）完整的诊断信息，和中断的附加信息。

SFB 54 从支持诊断的所有模块读取中断数据，不管这些模块在中央机架，还是在 DP 从站或 PROFINET IO 设备。SFB 54 的输出参数中的信息包含调用它的 OB 的启动信息，以及中断源的信息。由于要检查外部设备中断，因此最好在由 CPU 操作系统启动的中断 OB 中调用 SFB 54。如果不是在中断 OB 中调用 SFB 54，SFB 54 的输出变量提供的信息会减少。

SFB 54 的附加中断信息仅报告 PROFINET IO 触发中断的站点的故障通道的状态，和 DP 网络触发中断的站点所有通道的状态。

在不同的 OB 中调用 SFB 54 时，应使用不同的背景数据块。如果要在有关的中断 OB 之外使用 SFB 54 的输出数据，对每个 OB 启动事件应使用单独的背景数据块。诊断信息写入 SFB 54 的输出参数 STATUS、ID、LEN、TINFO 和 AINFO。TINFO 目标区中是 OB 的启动和管理信息，AINFO 目标区中是标题信息和附加的中断信息。如果 TINFO 和 AINFO 设置的数据区的长度不够，不能输入完整的信息。

可以用 3 种模式调用 SFB 54“RALRM”：

1）模式 0：输出参数 ID 提供触发中断的 DP 从站或从站中的模块的逻辑起始地址，并将输出参数 NEW 设置为 TRUE，不改写其他输出参数。

2）模式 1：不论是谁产生的中断，用有关的诊断数据改写 SFB 54 所有的输出参数。

3）模式 2：检查是否是输入参数 F_ID 指定的模块触发了中断。如果不是，则输出参数 NEW 为 FALSE；如果是，输出参数 NEW 为 1 状态，有关的诊断数据将改写 SFB 54 所有的输出参数。

如果有诊断功能的模块检测到故障，在中断事件刚产生和刚结束（进入事件或离开事件）时，将向 CPU 发出诊断中断请求，操作系统调用 OB82 来响应诊断请求。OB82 的局部变量包含了产生中断的逻辑基地址和与故障模块有关的 4 个字节的诊断数据。如果未生成和下载 OB82，CPU 将进入 STOP 模式。下面是 OB82 调用 SFB 54“RALRM”的程序。

程序段 3：

```
L       #OB82_MDL_ADDR              //触发 OB82 中断的模块的逻辑起始地址
ITD                                 //将整数转换为双整数
T       #TEMP                       //保存到临时局部变量 LD20
CALL    "RALRM" , DB54              //调用 SFB 54
  MODE   :=1                        //模式 1，用诊断数据改写SFB所有的输出参数
  F_ID   :=#TEMP                    //触发OB82 中断的模块的逻辑起始地址
  MLEN   :=1500                     //要读取的中断信息的最大字节长度
  NEW    :=M18.0                    //为 1 表示已接收新的中断
  STATUS :=MD20                     //SFB或DP主站的错误代码
  ID     :=MD24                     //接收到的产生中断的模块的逻辑起始地址
  LEN    :=MW28                     //已接收的中断信息的长度
  TINFO  :=P#DB6.DBX0.0 BYTE 100    //存放OB启动和管理信息的目标区域
  AINFO  :=P#DB7.DBX0.0 BYTE 100    //存放标题信息和附加中断信息的目标区域
```

输出参数 ID 的第 15 位为 1，表示产生中断的是输出模块；如果为 0，则是输入模块。

11.2.2 ET 200S PN 的 DO 模块负载断线的诊断

项目 315_2PN 用 CPU 集成的 PN 接口作 PROFINET 控制器，ET 200S PN 的 3 号插槽的

DO 模块组态了断线诊断功能（见图 11-9）。运行时断开该模块已通电的外部负载的接线，或者向外部负载已经断线的输出点 Q0.0 或 Q0.1 写入二进制数 1，将会触发诊断中断，CPU 调用 OB82 和 OB86。CPU、IM 151-3PN 和 DO 模块的 SF LED 亮。

1．用 STEP 7 诊断故障

打开快速视图，可以看到 CPU 和 IM 151-3PN 模块上的故障符号。

打开诊断视图（即在线的 HW Config），可以看到 CPU、IM 151-3PN 和 2DO 模块上的故障符号。双击 IM 151-3PN，打开它的模块信息对话框，“常规”选项卡给出的模块状态为“模块故障（检测到诊断中断），外部出错”。

在“IO 设备诊断”选项卡的“指定通道诊断”列表中，可以看到故障信息“3 号插槽通道 0 的引线断开”。点击“十六进制格式”按钮，显示出十六进制的诊断信息。

双击打开诊断视图中的 2DO 的模块信息对话框，得到的模块信息和十六进制格式诊断信息和 IM 151-3PN 的基本上相同。

2．PROFINET IO 的设备模型和寻址级别

PROFINET IO 设备具有与 PROFIBUS-DP 从站类似的模块化结构。可以将一个插槽分为多个子插槽，模块插入插槽中，而子模块插入子插槽中。模块/子模块有用于读取或输出过程信号的通道，有的模块没有子模块。

PROFINET IO 设备的诊断分为 4 级，1～4 级分别用于设备诊断、模块诊断、子模块诊断和通道诊断。通过下列寻址级别评估诊断数据和组态数据：

- AR（应用关联），数据记录编号为W#16#Exxx。
- API（应用程序进程标识符），数据记录编号为W#16#Fxxx。
- 插槽（Slot），数据记录编号为W#16#Cxxx。
- 子插槽（Subslot），数据记录编号为W#16#8xxx。

每个寻址级别都有一组可用的诊断记录和组态记录。通过记录编号的首字母来区别各记录组。每个 IO 设备（寻址级别为 AR）、模块（寻址级别为插槽）或子模块（寻址级别为子插槽）的信息传送到各个诊断记录或组态记录中。根据寻址级别，记录将返回一个或多个子插槽、插槽和 API 的诊断数据或组态数据。

3．用户结构标识符

用户结构标识符（USI）用于识别诊断数据的类型。诊断数据分为通道诊断、扩展的通道诊断和供应商特定的通道诊断。诊断记录编号的最后两个数字与诊断记录的类型有关。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C5550000
8.0	ARY[2]	DWORD	DW#16#0	DW#16#0D330000

图 11-27　DB 2 中 OB82 的局部变量

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#00100012
4.0	ARY[1]	DWORD	DW#16#0	DW#16#01000003
8.0	ARY[2]	DWORD	DW#16#0	DW#16#00018000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#08008000
16.0	ARY[4]	DWORD	DW#16#0	DW#16#00004801
20.0	ARY[5]	DWORD	DW#16#0	DW#16#00060000

图 11-28　DB 5 中的诊断数据

4．OB82 的局部变量

图 11-27 是 3 号槽的 DO 模块负载断线时，DB 2 保存的 OB82 的局部变量的前 12 个字节的数据。

11.2.3 诊断数据的分析

所有设备厂商的 PROFINET IO 的诊断信息数据记录都具有一致的结构，系统状态列表（SSL）、SFB 54 和 SFB 52 都进行了扩展，以便使 PROFINET IO 系统的状态和诊断信息可以用于 S7 用户程序。要了解为 PROFINET IO 定义了哪些 SSL 和诊断记录、诊断数据记录的结构等信息，请查阅手册《从 PROFIBUS DP 到 PROFINET IO 编程手册》。

1．SFB 52 读取的诊断数据的解读

诊断数据记录的结构和各部分的含义见手册《从 PROFIBUS DP 到 PROFINET IO 编程手册》的图 5-7 和表 6-1。图 11-28 是 DB 5 中用 SFB 52 读取的诊断记录，各部分的意义如下：

- DBW0=16#0010，块的类型，诊断记录。
- DBW2=16#0012，块的长度（不包括前 4 个字节），18B连续的数据。
- DBW4=16#0100，块的版本号。
- DBW6=16#0003，触发中断的模块插槽号（3 号槽）。
- DBW8=16#0001，子模块插槽号。
- DBW10=16#8000，通道编码，16#8000 为子模块级别上的诊断。
- DBW12=16#0800，通道属性。没有通道错误组信号，诊断事件，未决的诊断，特定于制造商。
- DBW14=16#8000，USI（用户结构标识符），16#8000 为通道诊断数据记录。
- DBW16 开始是通道的诊断数据，USI为 16#8000 时，每个通道的诊断数据占 6B。
- DBW16=16#0000，触发中断的模块的通道号。
- DBW18=16#4801，通道属性。数据类型为位（bit），没有通道错误组消息，诊断事件，未决的诊断，输出通道。
- DBW20=16#0006，通道错误类型为线路断开。

通道错误类型的含义见《从 PROFIBUS DP 到 PROFINET IO 编程手册》的表 5-13。

2．TINFO 中的启动和管理信息

TINFO 和 AINFO 的详细信息请参阅随书光盘中的文件《用于 S7 的系统软件和标准功能参考手册》的第 8.3 节，或参阅 SFB 54 的在线帮助。

下面是目标区域 TINFO 的数据结构：

- 字节 0～11：当前调用SFB 54 的OB82 的启动信息。
- 字节 12～19：产生中断请求的日期和时间。
- 字节 20～21：产生中断的从站或模块的地址。
- 字节 22～31：管理信息。

图 11-29 是 OB82 调用 SFB 54 后，保存在 DB 6 中的启动和管理信息。

DB 6 的前 20 个字节与 OB82 的局部变量（即 OB82 的启动信息，见图 11-27）相同。

DBW20=16#8002，是 PROFINET IO 的地址信息，表示 I/O 系统的编号为 100（第 11～14 位加 100），中断源的站编号（第 0～10 位）为 2（见《从 PROFIBUS DP 到 PROFINET IO 编程手册》的表 8-11 和 SFB 54 的在线帮助）。DBB22 开始为管理信息。

DBB22=16#08，分布式设备的类型为 PROFINET IO。

DBB23=16#00，中断信息类型，中断由已组态的分布式模块生成。

DBB24=16#00，PROFINET IO 控制器接口标志，来自 CPU 集成的接口电路的中断。

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#39421A52
4.0	ARY[1]	DWORD	DW#16#0	DW#16#C5550000
8.0	ARY[2]	DWORD	DW#16#0	DW#16#0D330000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#09020516
16.0	ARY[4]	DWORD	DW#16#0	DW#16#09008585
20.0	ARY[5]	DWORD	DW#16#0	DW#16#80020800
24.0	ARY[6]	DWORD	DW#16#0	DW#16#00010301
28.0	ARY[7]	DWORD	DW#16#0	DW#16#002A0001

图 11-29　DB 6 中的启动和管理信息

地址	名称	类型	初始值	实际值
0.0	ARY[0]	DWORD	DW#16#0	DW#16#0002001E
4.0	ARY[1]	DWORD	DW#16#0	DW#16#01000001
8.0	ARY[2]	DWORD	DW#16#0	DW#16#00000000
12.0	ARY[3]	DWORD	DW#16#0	DW#16#00030001
16.0	ARY[4]	DWORD	DW#16#0	DW#16#000088A1
20.0	ARY[5]	DWORD	DW#16#0	DW#16#00000000
24.0	ARY[6]	DWORD	DW#16#0	DW#16#A8008000
28.0	ARY[7]	DWORD	DW#16#0	DW#16#00004801
32.0	ARY[8]	DWORD	DW#16#0	DW#16#00060000

图 11-30　DB 7 中的标题信息和附加中断信息

DBB25=16#01，外部诊断标志，IO 设备故障。

DBW26=16#0301，PROFINET IO 设备的标识符编号。

DBW28=16#002A，供应商标识符。

DBW30=16#0001，实例（instance）标识符编号。

3．AINFO 中的标题信息和附加中断信息

图 11-30 是 OB82 调用 SFB 54 后，保存在 DB 7（目标区域 AINFO）中的标题信息和附加中断信息：

字节 0～3 是标题信息，块类型、中断类型、中断信息的字节长度和插槽号等。

字节 4～199：来自 PROFINET、DP 或集中式 IO 设备的附加的中断信息。

无维护请求的 AINFO 区的数据的意义如下（见《从 PROFIBUS DP 到 PROFINET IO 编程手册》的表 8-14 和 SFB 54 的在线帮助）：

- DBW0=16#0002，块的类型，中断传输通道 2。
- DBW2=16#001E，块的长度（不包括前 4B），30B连续的数据。
- DBW4=16#0100，版本号。
- DBW6=16#0001，中断类型，进入诊断中断。
- DBD8=16#00000000，API（应用程序进程标识符），0 为无配置文件。
- DBW12=16#0003，触发中断的模块的插槽号。
- DBW14=16#0001，触发中断的模块的子模块插槽号。
- DBD16=16#000088A1，模块的标识符。
- DBD20=16#00000000，子模块的标识符。
- DBW24=16#A800，中断的分类符和诊断状态。通道诊断可用，至少有一个通道诊断记录可用。在这个AR中，至少有一个组态的模块报告诊断。
- DBW26=16#8000，格式标识符，DBW28 开始是通道诊断数据记录。
- DBW28=16#0000，触发中断的通道号。
- DBW30=16#4801，信息与数据格式。没有通道错误组消息，诊断，未决的诊断，输出。
- DBW32=16#0006，通道错误类型为线路断开。

4．断线故障消失后的诊断信息

接通断开的负载，或者向断线的输出点写入二进制数 0，故障消失，CPU、IM 151-3PN 和 DO 模块的故障 LED 熄灭。CPU 又调用一次 OB86，DB 5 中的诊断数据不变。

与故障出现时相比，DB 6 中的诊断数据的变化如下：

- DBB0=16#38，离开的事件。
- DBW8=16#0003，错误消失，没有通道信息和错误信息可用。

- DBW24=16#0000，IO设备故障消失。

与故障出现时相比，DB 7 中的诊断数据的变化如下：

- DBW2=16#0016，块的长度为 22B连续的数据。
- DBW6=16#000C，中断类型，离开诊断中断。
- DBW24=16#0001，没有通道诊断记录可用。

11.2.4 其他故障的诊断

在运行时拔掉 ET 200S DP 的电源模块，CPU 调用 OB82 和 OB83，CPU、IE/PB Link、IM 151-1 和 6 号槽有诊断功能的 DI 模块的 SF LED 亮。

选中 SIMATIC 管理器左边窗口的 300 站点，执行菜单命令“PLC”→“诊断/设置”→“硬件诊断”，打开快速视图，可以看到带故障符号的 CPU、PROFINET IO 系统中的 125 号设备（IE/PB Link）和 3 号设备（ET 200S DP）。

点击“打开在线站点”按钮，打开诊断视图（在线的 HW Config，见图 11-31），可以看到 CPU、IE/PB Link 和 ET 200S 上的故障符号。

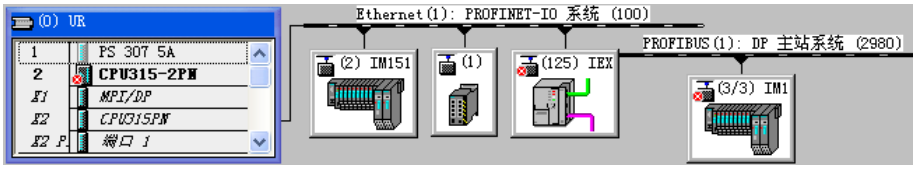


图 11-31　诊断视图

双击 IE/PB Link，打开它的模块信息对话框（见图 11-32），其“常规”选项卡中模块的状态为“模块可用且正常，外部错误”，“IO 设备诊断”选项卡没有诊断信息。诊断缓冲区第一条事件是模块被拔出，“关于事件的详细资料”给出了出故障的站地址为 3，模块的诊断地址为 2029。

图 11-32　IE/PB Link 的模块信息对话框

点击“关闭”按钮，返回诊断视图（见图 11-31）。双击 ET 200S DP，打开 IM 151-1 的模块信息对话框，其“常规”选项卡给出的模块状态是“模块可用且正常，外部出错”。“DP 从站诊断”选项卡给出的信息为“插槽 1 故障，Slot 1：no submodule”（插槽 1 没有模块）。

OB82 调用 SFB 54，读取的诊断数据保存在 DB 6 和 DB 7 中。

11.2.5 IE/PB Link 的诊断功能

拔掉 DP 网络上 ET 200S 的电源模块，在 HW Config 中离线打开 IE/PB Link，点击“诊断”选项卡中的“运行”按钮，打开模块的 MCN 诊断窗口，在诊断缓冲区（见图 11-33）中可以看到历史事件和诊断信息。

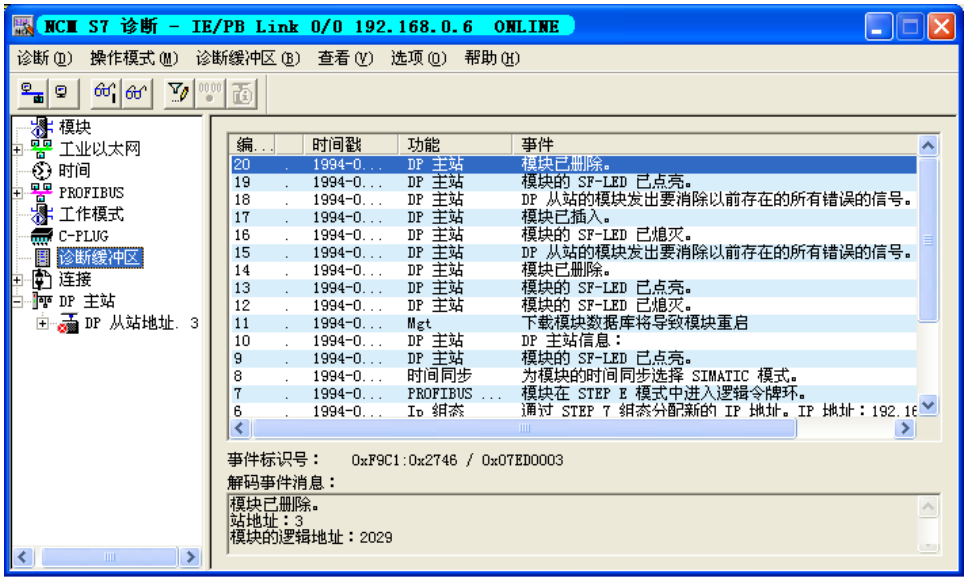


图 11-33 IE/PB Link 的诊断缓冲区

打开左边窗口的“\DP 主站\DP 从站地址：3”文件夹（见图 11-34），选中其中的模块 1（1 号槽的电源模块），右边窗口是该模块的信息。

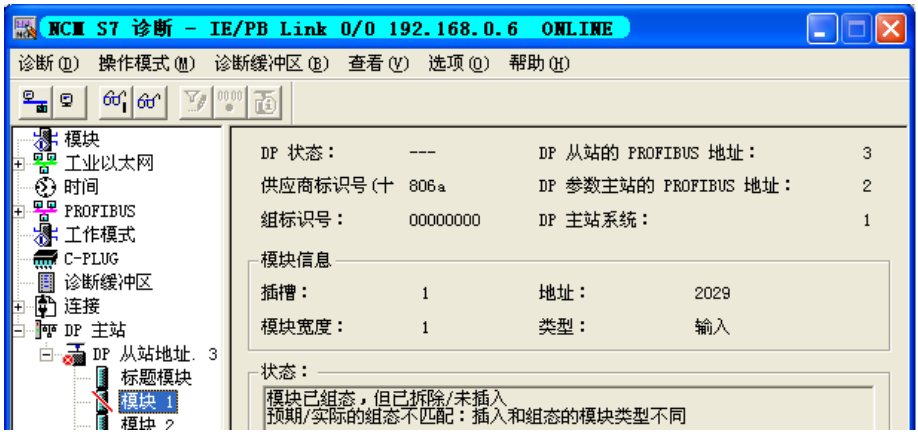


图 11-34 IE/PB Link 的 DP 从站信息

插入电源模块后，DB 2 中 OB82 的局部变量数据不变，DB 3 中 OB83 的局部变量的第一个字节变为 16#38，表示插入的 PROFINET IO 模块与组态的模块匹配。

11.2.6 基于通信处理器的 PROFINET 故障诊断

1．基于 CP 443-1 的 PROFINET 通信的故障诊断

项目 CP443_1 与项目 315_2PN 的诊断程序基本上相同。

（1）DO 模块负载断线的诊断

运行时断开 ET 200S PN 的 DO 模块外部负载的接线，因为 DO 模块组态了断线诊断功能（见图 11-9），触发了诊断中断，CPU 调用 OB82。CPU 和 CP 443-1 的 EXTF LED 亮，IM 151-3PN 和 DO 的 SF LED 亮。

打开诊断视图（即在线的 HW Config），选中 ET 200S PN，可以看到 CPU、IM 151-3PN 和 DO 模块上的故障符号。双击 ET 200S PN 的 DO 模块，打开它的模块信息对话框，在“IO 设备诊断”选项卡的“指定通道诊断”列表中，可以看到 3 号插槽的通道 0 和通道 1 的引线断开的故障。

OB82 调用 SFB 54 后，保存在 DB 6 和 DB 7 中的诊断信息与项目 315_PN 同样的故障读取的诊断信息基本上相同。

（2）拔出有诊断功能的 DI 模块

在运行时拔出 200S DP 的 6 号槽组态了诊断功能的 DI 模块，CPU 和 CP 443-1 的 EXTF LED 亮，IM 151-3 的 SF LED 亮。从变量表可以看出，CPU 分别调用了一次 OB83 和 OB86，模块拔出期间，每个扫描循环周期调用一次 OB85。

选中 SIMATIC 管理器左边窗口的 400 站点，执行菜单命令“PLC”→“诊断/设置”→“模块信息”，打开 CPU 的模块信息对话框。因为在 CPU 的每个扫描循环周期都要调用一次 OB85，诊断缓冲区中的事件都是调用 OB85 的信息，“关于事件的详细资料”区（见图 11-35）给出了出现故障的模块的地址为 IB11（即 6 号槽的 DI 模块的地址）。

双击诊断视图中的 IE/PB Link，打开它的模块信息对话框，在“关于事件的详细资料”区（见图 11-36），可以看到 3 号站地址为 11 的模块被拔出的信息。

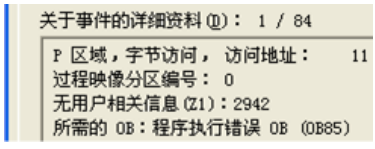


图 11-35 CPU 的模块信息对话框

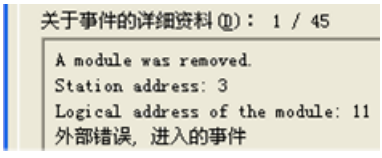


图 11-36 IE/PB Link 的模块信息对话框

插入 6 号槽的 DI 模块，CPU 又调用一次 OB83 和 OB86。

2．基于 CP 343-1 的 PROFINET 通信的故障诊断

（1）拔出 ET 200S DP 电源模块的诊断

运行时拔出 ET 200S DP 的电源模块，CPU 的 LED 状态不变，CP 343-1、IE/PB Link、IM 151-1 和有诊断功能的 6 号槽 DO 模块的 SF LED 亮，未产生中断。

打开诊断视图（即在线的 HW Config），其中的 CP 343-1、IE/PB Link 和 ET 200S DP 上有故障符号。双击 CP 343-1，打开它的模块信息对话框，在“诊断缓冲区”选项卡中可以看

到信息“外部错误，进入的事件”。

双击诊断视图中的 IE/PB Link，打开它的模块信息对话框，在“常规”选项卡中，可以看到“模块可用且正常，外部出错”。在“诊断缓冲区”选项卡，可以看到模块被拔出的信息、该模块所在的站地址和模块的诊断地址，以及信息“外部错误，进入的事件”。

（2）拔出 ET 200S PN 电源模块的诊断

运行时拔出 ET 200S PN 插槽 1 的电源模块，CPU 的 LED 状态不变，CP 343-1、IM 151-3PN 和有诊断功能的 DI、DO 模块的 SF LED 亮，未产生中断。

在诊断视图中，CP 343-1 和 ET 200S PN 有故障符号。

CP 343-1 的诊断缓冲区的诊断信息为“外部错误，进入的事件”。

IM 151-3PN 的“模块信息”对话框的“常规”选项卡中的信息为“模块可用且正常，外部出错”，“IO 设备诊断”选项卡的诊断信息为“插槽 1 中的模块丢失”。

（3）硬件中断

CP 343-1 作 PROFINET 控制器时，ET 200S PN 的 DI 模块属性视图的“参数”选项卡中的硬件中断复选框为灰色，不能组态硬件中断。

CPU 集成的 PN 接口和 CP 443-1 作 PROFINET 控制器时，ET 200S PN 的 DI 模块可以组态和产生硬件中断。使用本章的 3 种 PROFINET 控制器时，ET 200S DP 的 DI 模块都可以组态和产生硬件中断。

11.3 基于组件的自动化

11.3.1 PROFINET CBA

1．PROFINET CBA 的基本概念

CBA 是基于组件的自动化（Component Based Automation）的简称。PROFINET CBA 将自动控制系统组织为独立的组件。这些组件可以是子网络、PLC 或现场设备。组件包括所有的硬件组态数据、模块参数和有关的用户程序。

CBA 采用 Microsoft 的组件模型 COM/DCOM，这是在 PC 领域中应用最广的数据与通信模型，它确定了不同设备软件部件之间数据交换的协议。

可以用 PROFINET IO 将现场设备集成在 PROFINET CBA 组件中。通过使用代理设备，还可以用 CBA 使所有现有的子网与 PLC 或现场设备（例如 PROFIBUS-DP 设备）互连，形成更大的自动化系统。

可以通过统一定义的接口访问这些 PROFINET 组件。这些组件可以用任意方式互连，以实现过程组态。开放的工程接口允许用不同的制造商提供的 PROFINET 组件实现图形组态。

在 STEP 7 中，将有关的机械部件、电气/电子部件和应用软件等具有独立工作能力的工艺模块抽象为一个封装好的组件，并为组件定义标准的接口，以实现组件之间的标准通信。

可以将生产线的单台机器定义为生产线或过程中的一个“标准模块”。各组件之间用 PROFINET 连接。由于组件使用了标准的接口，使得各组件之间的连接变得极为简单。不同的组件可以像模块一样组合，完全独立于其内部程序。在 CBA 的组态工具 iMap 中，组件是一种软件模块，可以像搭积木一样组合组件。各组件之间的 PROFINET 和 PROFIBUS 的通信

用 iMap 进行图形化组态，不需要编程。通过模块化这一成功理念，可以显著降低机器和工厂建设中的组态与在线调试的时间。iMap 还可以为系统组态简单的诊断。

对于设备与工厂的设计者，工艺模块化能更好地对用户的设备和系统进行标准化，组件可以重复利用。因此可以对不同的客户要求作出更快、更灵活的反应。可以对各台设备和工段提前进行预先测试，交付工厂后可以立即使用，因此可以缩短系统上线调试阶段的时间。作为系统操作者和管理者，从现场设备到管理层，用户都可以从 IT 标准的通用通信中获取信息。对现有系统进行扩展也很容易。

2．SIMATIC iMap

在 STEP 7 中，用菜单命令来创建 PROFINET 组件，生成的组件在外观上有可以互连可诊断的接口。PROFINET 组件的技术接口是用 XML（Extensible markup language，可扩展符号化语言）来定义的。PROFINET 组件集成了硬件组态和用户程序的信息。

iMap 是用于图形化组态技术功能模块之间的数据通信的软件。它用于对整个系统组态，还可以对技术功能模块作简单的诊断。iMap 用图形化方式连接组件的接口，以实现组件之间的数据交换。作者使用的是随书光盘中的 V3.0 版的 iMap。

安装好 STEP 7 后，需要单独安装 iMap STEP 7 AddOn，才能在 STEP 7 中定义组件接口和创建 PROFINET 组件。STEP 7 和 iMap 都附有 iMap STEP 7 AddOn。

3．PN CBA 组态和调试的步骤

1）在 STEP 7 中进行硬件组态和下载，创建和编辑组件。

2）在 iMap 中导入组件和将组件互连。

3）下载组件，测试组件和系统功能。

4．PROFINET 组件设备

以下的 SIMATIC 产品可以用于 PROFINET 分布式智能设备通信，可以组态为一个 CBA 的组件，与其它的组件通信。

1）CPU 315/317-2DP/PN 用于处理过程信号，直接将现场设备连接到工业以太网。

2）CP 343-1 是 S7-300 连接现场 PROFINET 设备的通信处理器。

3）CP 443-1 Advanced 是 S7-400 连接现场 PROFINET 设备的通信处理器。带有集成的 WEB 服务器和集成的交换机。

4）通信处理器 CP 1616 用于将 PC 连接到 PROFINET 现场设备，它带有集成的交换机。

5）PN CBA OPC Server 是 PC 的应用程序接口，用于 PC 与 CBA 组件通信。

6）PROFIBUS 设备可以组态为一个 CBA 组件。通过 CPU 315-PN/DP 等代理设备，可以与其他组件通信。

11.3.2 在 STEP 7 中创建组件

1．新建一个项目

在 STEP 7 中创建一个名为 PN_CBA1 的项目（见随书光盘中的同名例程），生成一个 S7-300 站点。打开硬件组态工具 HW Config，将订货号为 315-2EH13-0AB0 的 CPU 315-2PN/DP 拖放到 2 号槽。在自动打开的以太网接口属性对话框的“参数”选项卡中，点击“新建”按钮，生成一条名为“Ethernet（1）”的以太网，点击“确定”按钮，返回“参数”选项卡，将 CPU 连接到该网上。采用默认的 IP 地址 192.168.0.1 和子网掩码 255.255.255.0，点击“确定”

按钮，返回 HW Config。将电源模块和信号模块插入主机架。

双击机架中 CPU 内“PN-IO”所在的行，在打开的接口属性对话框的“PROFINET”选项卡中（见图 11-37），选中复选框“使用本模块进行 PROFINET CBA 通信”。

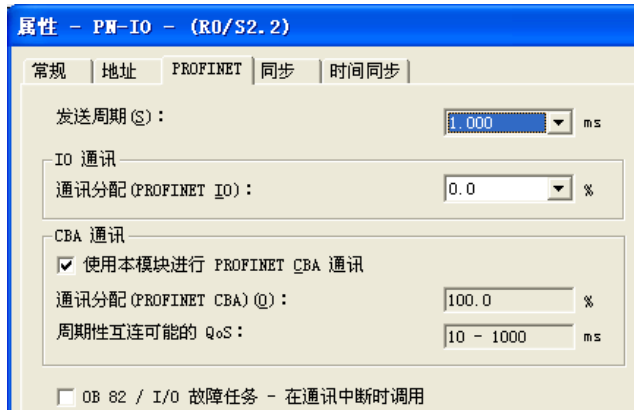


图 11-37　PN IO 属性对话框

2．设置 PG/PC 接口

执行 SIMATIC 管理器的菜单命令“选项”→“设置 PG/PC 接口”，设置计算机的普通网卡的 IP 地址，具体的操作见 10.2.2 节。在 HW Config 中，将组态信息下载到 CPU。

3．创建 S7-300 组件接口

用鼠标右键点击“SIMATIC 300(1)”图标，执行出现的快捷菜单中的命令“Create PROFInet Interface”（生成 PROFINET 接口）。点击出现的 PROFINET 接口编辑器左边窗口中的“Add PN block”（添加 PN 块，见图 11-38），在“Available block”（可用的块）列表中生成一个用于组件的共享数据块 DB 1。点击带有向上箭头的按钮，将 DB 1 传送到上面的“Assigned PN block”（指定的 PN 块）区。

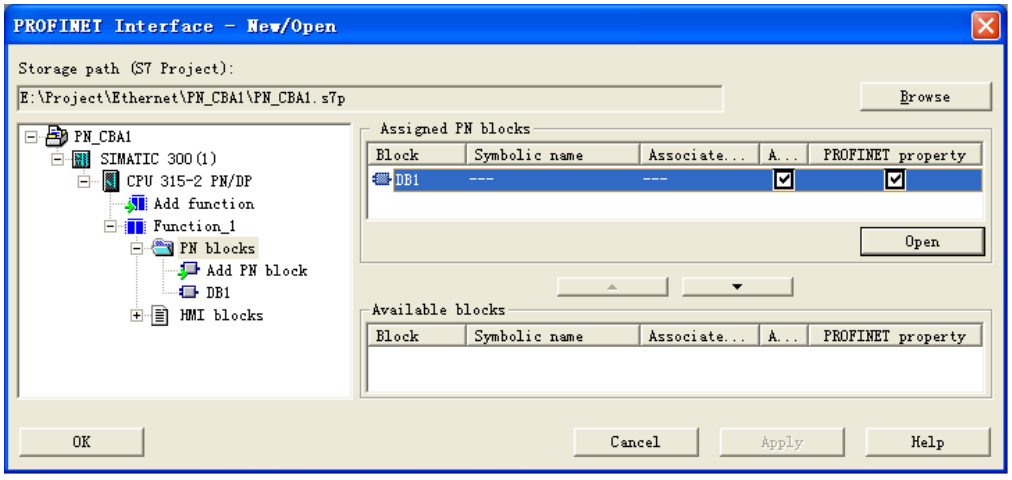


图 11-38　PROFINET 接口属性对话框

点击“Open”按钮或双击 DB 1，打开 PROFINET 接口编辑器和选中的 DB 1（见图 11-39）。

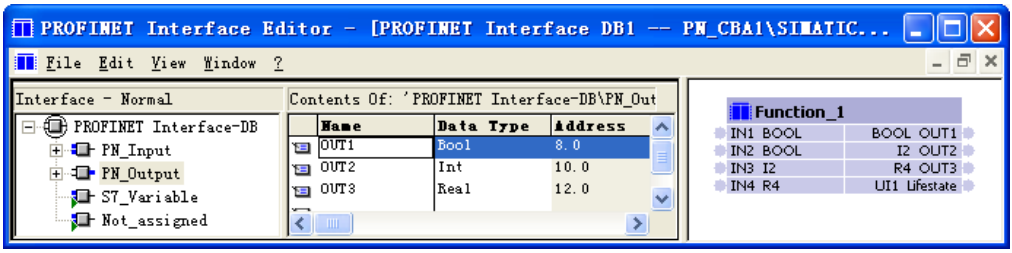


图 11-39　PROFINET 接口编辑器

选中 PROFINET 接口（DB）编辑器左边窗口的“PN_Input”，在中间窗口设置的组件的输入参数，将出现在右边窗口的组件图形的左边。参数中的 I2 表示 2B 的整型数据，R4 表示 4B 的浮点数。组件图形左边的输入变量和右边的输出变量是组件的接口变量，它们用于与其他组件互连。S7 变量（S7 Variable）不用于组件互连，可以通过 OPC 方式用 HMI 访问它们。最后保存组态结果，返回 SIMATIC 管理器。

4．创建 PROFINET 组件

用鼠标右键点击 SIMATIC 管理器中的“SIMATIC 300（1）”图标，执行出现的快捷菜单中的命令“Create PROFInet Component”，生成一个名为“SIMATIC 300（1）”的 PROFINET 组件。

在创建 PROFINET 组件对话框的“General”选项卡中，组件被创建时默认为 New 方式。如果已经创建了该组件，选用 Retain（保持）方式，将产生一个新版本的组件。

在“Component Type”选项卡中，选择创建的组件的类型为 Standard Component（标准组件），创建一个简单的组件。可以选择是否有代理功能。在“Storage Areas”（存储区）选项卡，可以设置保存组件的文件夹。点击“OK”按钮，组件被创建。

5．创建另一个 STEP 7 项目和组件

在 STEP 7 中创建一个名为 PN_CBA2 的项目（见随书光盘中的同名例程），生成一个 S7-300 站点。CPU 为 CPU 317-2PN/DP，IP 地址为 192.168.0.2，子网掩码为 255.255.255.0。

用上述的方法创建一个 PROFINET 组件（见图 11-40），在创建 PROFINET 组件对话框的“General”选项卡，将组件的名称改为“SIMATIC 300（2）”。

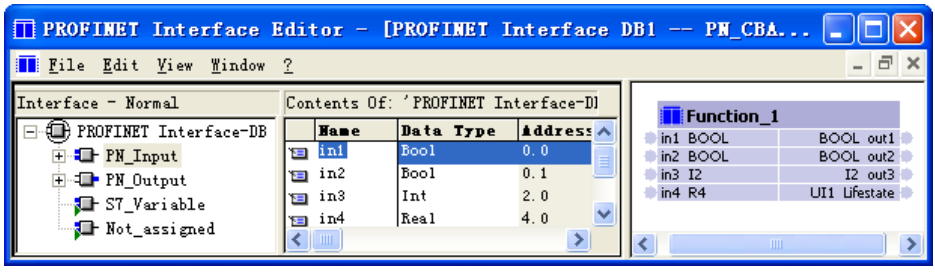


图 11-40　PROFINET 接口编辑器

11.3.3　用 iMap 连接和下载组件

1．导入组件

打开 iMap（见图 11-41），用鼠标右键点击右上角的“Project Library”（项目库）区，执

行弹出的快捷菜单中的命令“Import Components”，导入在 STEP 7 中生成的组件。

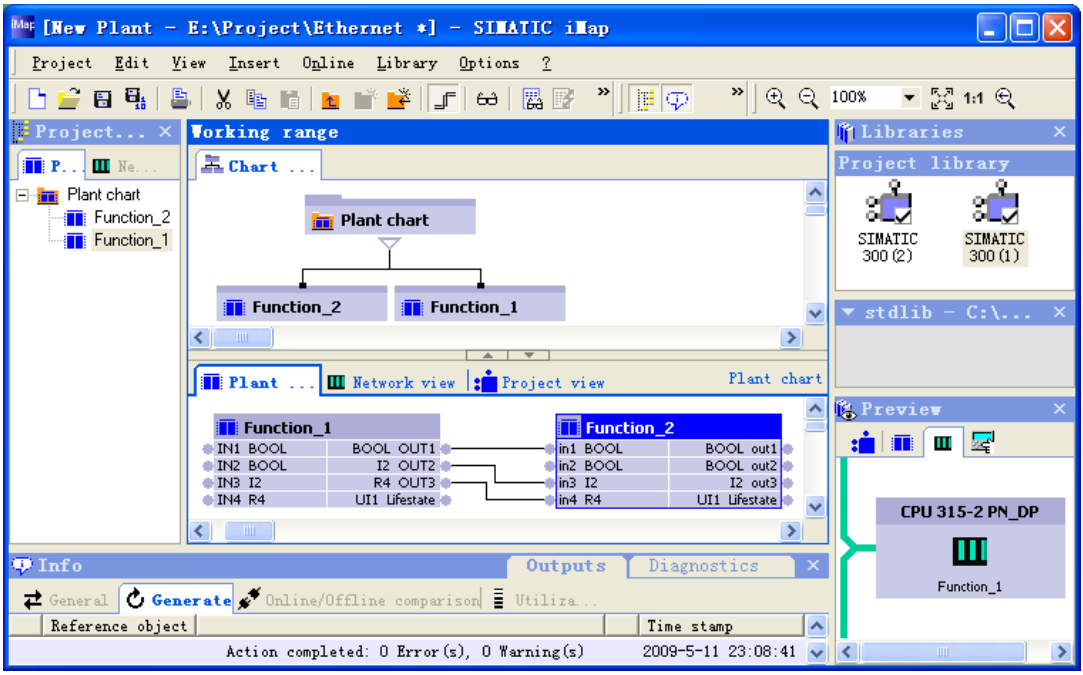


图 11-41　在 iMap 中连接组件

在出现的“Import Components”对话框中，打开组件所在的文件夹，双击其中的 XML 文件，组件被导入到右边的 Project Library 区。在下面的 Preview（预览）区，可以看到选中的组件的图形，用预览区的选项卡可以改变组件显示的方式。

2．用 iMap 互连组件

用鼠标左键将 Project Library 区导入的组件拖放到左边的 Project tree（项目树）窗口。图 11-41 中间的 Chart 区和 Plant 区的图形是自动打开的，但是此时 Plant 区还没有组件之间的连线。

用下面的方法将两个组件相同数据类型的输出变量和输入变量连接到一起：用鼠标点击组件 Function_1（CPU 315-2PN/DP）的输出接口变量 OUT1，出现图标，再点击组件 Function_2（CPU 317-2PN/DP）的输入接口变量 in1。这样两个变量就连接在一起了。可以用工具栏上的按钮选择用编号或联线显示组件之间的连接关系。

本例的组件为 Standard 组件，需要手动设置 IP 地址。打开图 11-41 中间窗口的 Network view 选项卡，用鼠标右键点击图 11-42 中的组件 CPU 317-2PN_DP，执行弹出的快捷菜单中的 Properties（属性）命令。在出现的 Properties 对话框的 Addresses 选项卡中，输入在 STEP 7 的 HW Config 中设置的 IP 地址。用同样的方法设置 CPU 315-2PN_DP 的 IP 地址。

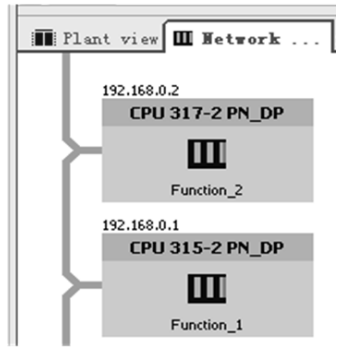


图 11-42　iMap 中的网络显示

执行 iMap 的菜单命令“Project”→“Properties”，在出现

的 iMap 的属性对话框的“Interconnections”（互连）选项卡中，根据需要设置循环时间和非循环时间的大小。可以使用默认的设置。最后点击 iMap 软件工具栏上的  按钮，保存和编译名为“New Plant”的 iMap 项目（见随书光盘中的同名例程）。用鼠标右键点击左边 Project tree 窗口中的某个组件，执行出现的快捷菜单中的“Download selected instances”→“All”命令，下载该组件的全部程序和组件之间的互连信息。用同样的方法下载所有的组件。

3．组件的监控

使用工具栏上的  按钮，可以在线监视 PN 设备。在 iMap 下面的诊断信息区（Info 区）的 Functions 选项卡中，可以看到有关的信息。可以用 Info 区的 Variable table（变量表）选项卡来生成需要监控的变量。点击工具栏上的  按钮，启动或停止变量表监控。

也可以用鼠标右键单击左边的 Plant Chart 窗口中的组件，在弹出的菜单中执行“Display online values”来查看组件变量的在线值。可以在线修改变量的值。

11.4　练习题

1．PROFINET 的实时通信功能和同步实时功能各有什么特点？

2．哪些设备可以作 PROFINET IO 控制器？

3．哪些设备可以作 PROFINET IO 设备？

4．什么设备可以作 PROFINET 代理服务器？

5．组态一个项目，CPU 414-2PN/DP 作 PROFINET IO 控制器，ET 200pro 和 ET 200S 作 PROFINET IO 设备，IE/PB Link 作代理服务器，DP 网络上有一个 ET 200M 从站。给各 PROFINET IO 设备和 DP 从站配置适当的信号模块，给有诊断功能的模块组态诊断功能。生成必须的组织块。

6．怎样诊断 PROFINET IO 系统的故障？

第 12 章　AS-i 网络通信

12.1　AS-i 网络概述

AS-i 是执行器传感器接口（Actuator Sensor Interface）的缩写，它是用于现场自动化设备的双向数据通信网络，位于工厂自动化网络的最底层，是自动化技术中一种最简单、成本最低的解决方案。AS-i 已被列入 IEC 62026 和国家标准。AS-i 特别适合于连接需要传送开关量信号的传感器和执行器，例如读取接近开关、光电开关、压力开关、温度开关、物料位置开关的状态，控制阀门、声光报警器、继电器和接触器等，AS-i 也可以传送模拟量数据。

12.1.1　AS-i 的数据传输方式与网络结构

1．数据传输方式

AS-i 是单主站主从式网络，每个网段只能有一个 AS-i 主站（见图 12-1）。AS-i 通信处理器（CP）作为主站控制现场的通信过程。主站是网络通信的中心，负责网络的初始化，以及设置从站的地址和参数等。

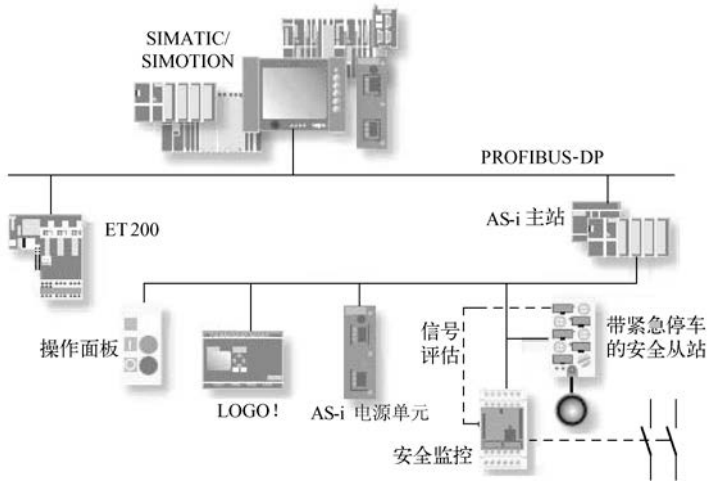


图 12-1　AS-i 网络

AS-i 所有分支电路的最大总长度为 100m，可以用中继器延长，有中继器时最大 500m。传输媒体可以是屏蔽的或非屏蔽的两芯电缆，支持总线供电，即两根电缆同时作信号线和电源线。网络的树形结构允许电缆中的任意点作为新的分支的起点。

AS-i 总线采用轮循方式传送数据，AS-i 主站严格按照精确的时间间隔轮流询问每一个从站，询问后等待从站的响应。主站循环读取输入数据并向从站发送输出数据，控制程序访问分布式 I/O 设备的方式与访问集中式 I/O 设备的方式相同。AS-i 通过自动重复发送数据和采

用附加校验的方法，来提高数据的完整性和准确性。

AS-i 使用电流调制的传输技术，以保证通信的高可靠性。主站如果检测到传输错误或从站的故障，将会发送报文给 PLC，提醒用户进行处理。在正常运行时增加或减少从站，不会影响其他从站的通信。

可以非循环地交换参数和诊断数据，如果从站支持，主站还可以更改从站的地址。

AS-i 网络的 CP 用 STEP 7 或 STEP 7-Micro/WIN 组态，可以自动读取从站的组态信息，也可以在 CP 的属性对话框中组态 AS-i 从站，网络启动时主站自动识别从站。

2．网络结构

AS-i 网络允许使用总线型、树形和星形拓扑。AS-i 网络包括以下网络组件：

1）铜质电缆（通常是外观呈黄色的扁平电缆）。AS-i 子网使用电气总线，不支持光纤和无线网络。

2）用于延长 AS-i 总线的中继器和扩展器。中继器还可以实现两个网段之间的电气隔离。

3）AS-i 从站是 AS-i 系统的输入通道和输出通道，它们仅在被 AS-i 主站访问时才被激活。接到命令时，它们触发动作或者将现场信息传送给主站。AS-i 从站可以是集成有 AS-i 接口的传感器/执行器或 AS-i 模块。每个 AS-i 模块最多可以连接 8 个数字量传感器/执行器。没有 AS-i 接口的传感器和执行器可以通过 AS-i 模块连接至总线。

模拟量从站是特殊的标准 AS-i 从站，它们可以通过特殊配置文件与主站交换模拟量数据。

4）为传感器和执行器供电的 AS-i 供电装置。AS-i 电源模块的额定电压为 DC 24V，最大输出电流为 2A。

3．ASIsafe

ASIsafe 是 AS-i 接口与安全有关的版本。在此版本中，标准数据和与安全有关的数据在同一条总线上传输。ASIsafe 允许将面向安全的组件（例如急停开关、防护门开关和安全光幕等）直接集成到 AS-i 网络中。这些面向安全的组件与符合 EN 62026-2 标准的 AS-i 接口组件完全兼容。可以使用 ASIsafe 执行与安全有关的断电任务，并保持简单经济的接线方式。

ASIsafe 需要添加安全从站和安全监视器才能将 AS-i 接口升级为安全总线，通过安全 AS-i 从站可以获得与安全有关的输入。由于是对安全逻辑进行编程而不是进行接线，因而增加了灵活性。使用简单的图形工具便可以快速浏览系统的安全功能，集成的诊断功能使维护和停机的时间降到最少。安全监视器可以执行以下任务：监视安全输入，通过可组态逻辑连接安全输入，通过内置的安全继电器安全断电。

12.1.2 AS-i 主站模块

1．CP 243-2

CP 243-2 是 S7-200 CPU 22x 的 AS-i 主站，S7-200 同时可以处理最多两个 CP 243-2。

CP 243-2 支持扩展 AS-i 特性的所有特殊功能，可以访问模拟量从站。通过双重地址(A-B)赋值，每个 CP 243-2 最多可以处理 62 个 AS-i 从站，最多连接 248 点数字量输入和 186 点数字量输出。

2．CP 343-2 与 CP 343-2P

CP 343-2 和 CP 343-2P 是用于 S7-300 PLC 和 ET 200M 的 AS-i 主站，其数据处理能力与 CP 243-2 相同。CP 343-2P 可以上载从站的信息，对从站的参数组态。

3．CP 142-2

AS-i 主站 CP 142-2 用于 ET 200X 分布式 I/O 系统。CP 142-2 最多可以寻址 31 个从站，最多 124 点数字量输入和 124 点数字量输出。

4．DP/AS-i Link 模块

DP/AS-i Link 模块是用来连接 PROFIBUS-DP 和 AS-i 网络的路由器。包括 DP/AS-i Link Advanced、DP/AS-i Link 20E 和 DP/AS-i F-Link，后者用于故障安全系统。它们是 DP 从站，又是 AS-i 的主站，可以连接 62 个 AS-i 从站。通过它们，可以从 DP 网络访问 AS-i 从站。

12.1.3 AS-i 从站

1．AS-i 从站的功能

AS-i 从站所有的功能都集成在一片专用的集成电路芯片中，AS-i 连接器可以直接集成在执行器和传感器中。从站的 AS-i 集成电路包含 4 个可组态的输入和输出，以及 4 个参数输出。在 EEPROM 存储器中存储运行参数、指定 I/O 的组态数据、标识码和从站地址等。

使用 AS-i 从站的参数输出，AS-i 主站可以传送参数值，它们用于控制和切换传感器或执行器的内部操作模式，例如在不同的运行阶段修改标度值。4 位输入/输出组态(I/O 组态)用来指定从站的哪根数据线作输入、输出或双向输出，从站的类型用标识码来描述。

2．AS-i 从站模块

AS-i 从站模块最多可以连接 4 个传感器和 4 个执行器。带有集成的 AS-i 接口的传感器和执行器可以直接连接到 AS-i 网络上。

AS-i 从站模块的接线采用绝缘穿刺技术，AS-i 电缆夹在安装板和从站模块之间（见图 12–2），从站通过插针接通电缆中的导线。如果需要移走从站，抽出插针后，橡胶电缆的自封闭特性使电缆的绝缘恢复，可以达到 IP67 的防护等级。SlimLine 模块的防护等级为 IP20，可以像其他低压设备一样安装在 DIN 导轨上，或用螺钉固定在控制柜的背板上。IP65/67 防护等级的 AS-i 从站模块可以直接安装在环境恶劣的工业现场。

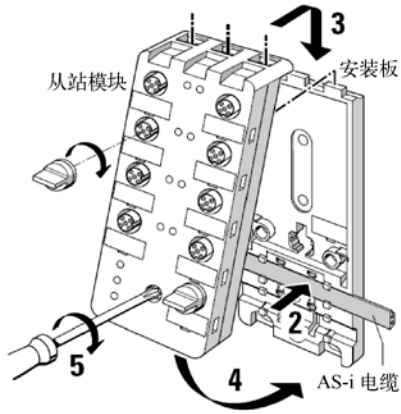


图 12–2　AS-i 从站模块的安装

3．紧凑型 AS-i 模块

这是一种具有较高保护等级的新一代紧凑型 AS-i 模块，包括数字、模拟、气动和 DC 24V 电动机起动器模块。模块有两种尺寸，可以满足各种安装要求，其保护等级为 IP67。每个模拟量模块有两个或 4 个通道。有电流型、电压型、热电阻型传感器输入模块，和电流型、电压型执行器输出模块。通过一个集成的编址插孔可以对模块编址。所有的模块都可以通过与 S7 系列 PLC 的通信实现参数设置。

4．“LOGO!”微型控制器

LOGO！是一种低成本的微型 PLC，它具有数字量或模拟量输入和输出、逻辑处理和实时钟功能。通过内置的 AS-i 模块，LOGO！可以作为 AS-i 网络中的智能型从站使用。使用 LOGO！面板上的按键和显示器，可以对它编程和进行参数设置，也可以用编程软件编程。LOGO！适合于简单的分布式自动化任务，例如门控系统，可以通过 AS-i 网络将它纳入高端自动化系统中。在高端控制系统出现故障时，可以继续进行控制。

5．气动控制模块

西门子提供带两个集成的 3/2 路阀门和带两个集成的 4/2 路阀门的气动紧凑型模块。模块有单稳和双稳两种类型，集成了作为气动单元执行器的阀门，接收来自气缸的位置信号。

6．电动机起动器

电动机起动器是 AS-i 的标准从站。有不同的防护等级和多种电压等级的产品，容量最大的可控制 15 kW 的电动机。

7．接近开关

BERO 接近开关可以直接连接到 AS-i 或接口模块上。特殊的感应式、光学和声纳 BERO 接近开关适合直接连接到 AS-i 上。它们集成有 AS-i 芯片，除了开关量输出之外，还提供其他信息，例如开关范围和线圈故障。通过 AS-i 电缆可以设置这些智能 BERO 的参数。

8．按钮和 LED

3SF5 是具有 AS-i 接口的完整的操作员通信系统人机界面。带 LED 的指令按钮通过 AS-i 电缆供电。

12.1.4　AS-i 的寻址模式与编址单元

1．标准寻址模式

AS-i 的节点（即从站）地址为 5 位二进制数，每个标准从站占一个 AS-i 地址，最多可以连接 31 个从站，地址 0 仅供产品出厂时使用，在网络中应改用别的地址。

AS-i 的数据帧的结构和长度都是固定的。在每个周期内，4 个输入位和 4 个输出位用于从站与主站的数据交换。每个标准 AS-i 从站可以接收 4 位数据或发送 4 位数据，所以一个 AS-i 总线网段最多可以连接 124 个输入点和 124 个输出点，对 31 个标准从站的典型轮询时间为 5ms，因此 AS-i 适用于工业过程数字量高速输入输出的场合。

2．扩展的寻址模式

在扩展的寻址模式中，两个从站使用相同的地址，分别作为 A 从站和 B 从站，这样最多可以寻址 62 个从站。由于地址的扩展，使用扩展的寻址模式后，从站的二进制输出减少到 3 个，每个从站最多 4 点输入和 3 点输出。一个扩展的 AS-i 主站可以操作 186 个输出点和 248 个输入点。使用扩展的寻址模式时对从站的最大轮询时间为 10ms。用于 S7-200 的通信处理器 CP 243-2 和用于 S7-300、ET 200M 的通信处理器 CP 343-2 可以作标准 AS-i 主站或扩展的 AS-i 主站。

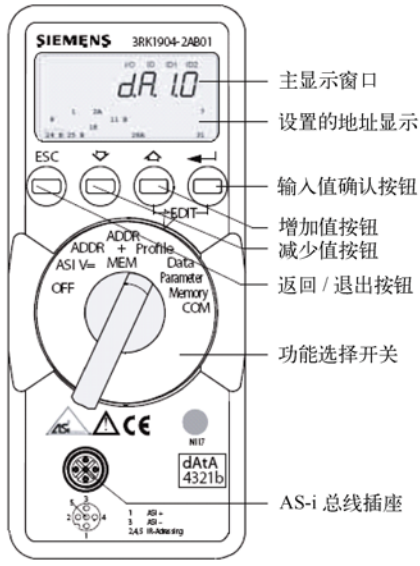


图 12–3　AS-i 编址与诊断单元

3．AS-i 编址与诊断单元

在建立 AS-i 网络之前，应为所有的从站分配地址。可以用编址单元或在线通过 AS-i 主站来完成这一任务。

从站出厂时的地址为 0，网络运行时实际使用的地址为 1～31，扩展寻址的地址为 1A～31A 和 1B～31B。不必按顺序分配地址，例如第 1 个从站可以设置为 1 之外的地址。设置从站地址时，需要通过 AS-i 插座，用 AS-i 电缆连接编址单元（见图 12–3）

和一个待设置的从站。编址与诊断单元有下列功能：

- 读取和设置从站地址。
- 读取从站的I/O代码和ID代码。
- 读取符合AS-i规范V2.1 的代码ID1 和ID2。
- 读取和改变从站的数据显示模式。
- 测试从站功能，读取数字量或模拟量从站的输入和输出。
- 测试AS-i总线电压（0～35V）和消耗的电流（0～100 mA）。
- 保存完整的设备配置（符合AS-i技术规范V2.1 和扩展的编址模式）。
- 检测整个系统的配置。

12.2 基于 CP 243-2 的 AS-i 网络的组态与编程

12.2.1 CP 243-2 简介

1．CP 243-2 的状态指示灯

CP 243-2 是用于 S7-200 的 AS-i 主站模块（见图 12-4），面板上面的 6 个 LED 指示灯用于指示各种状态（见表 12-1）。

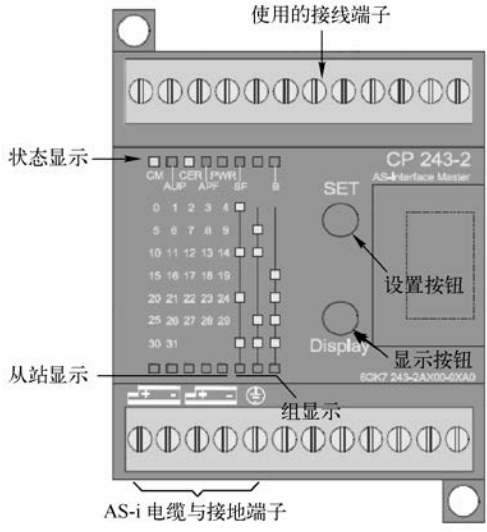


图 12-4 CP243-2

表 12-1 CP 243-2 的状态指示灯

LED	状 态	意 义
CM（黄）	组态模式	组态模式时亮，保护模式时熄灭
AUP（绿）	自动地址编程	保护模式时亮，表示可以为更换的从站自动编址
CER（黄）	组态错误	从站组态与实际的组态不一致时亮，例如组态的从站不存在或失效，从站的参数与组态的不一致，CP 243-2 处于离线状态
APF（红）	AS-i 电源故障	AS-i 电源故障或电压过低
PWR（绿）	电源	电源正常时亮
SF（红）	系统错误	CP 243-2 内部故障或按钮按下时不能切换到要求的模式时亮

2．CP 243-2 的从站状态指示灯

在 CP 243-2 的面板上，32 个从站地址分为 7 组。可以用 Display 按钮切换显示的从站组，每按一次 Display 按钮，显示下一组从站的状态。面板下面右边的 3 个“组显示”LED 对应于 3 位二进制数（最右边的是最高位），用来指示当前显示的是哪一组从站。例如左边的两个“组显示”LED 亮时，对应的二进制数为 2#011（十进制数 3），此时下面左边的 5 个“从站显示”LED 显示的是第 3 组（10～14 号）从站的状态。

如果最左边的两个“从站显示”LED 亮，且最上面一行标有“B”的绿色 LED 熄灭，表示 10 号和 11 号从站工作正常。如果此时标有“B”的绿色 LED 亮，表示 10B 和 11B 号从站工作正常。如果 CP 243-2 处于组态模式，显示所有被检测到的从站，如果处于保护模式，显示激活的从站。有故障或未组态的从站、站地址相同的从站对应的 LED 闪烁。

3．AS-i 主站的操作模式

CP 243-2 有两种操作模式：组态模式和保护模式，SET 按钮用于改变操作模式。

（1）组态模式

进入组态模式之前，应设置好各从站的地址，CPU 应处于 STOP 模式，CP 243-2 和所有的 AS-i 从站应连接到 AS-i 网络上，并且用 AS-i 电源供电。

在组态模式，CP 243-2 与 AS-i 电缆上连接的每个 AS-i 从站交换数据（不包括地址为 0 的从站），被主站检测到的新加入的从站被激活，并参与数据交换循环。主站读取设置好地址的 AS-i 从站的信息，包括从站的地址、ID 代码（包括扩展代码 ID1 和 ID2）、I/O 组态和当前的从站参数，并将它们永久性地储存在 CP 243-2 中。组态的过程如下：

1）按一下“Display”按钮，使 CP 243-2 进入“状态显示”模式（初始状态）。

2）按一下“SET”按钮，使 CP 243-2 进入组态模式，“CM”（组态模式）LED 亮。

3）用“Display”按钮切换各组从站的显示，检查 AS-i 网络上所有的从站是否存在。

4）再按一下“SET”按钮，CP 243-2 组态完成，同时主站切换到保护模式，“CM”LED 熄灭。因为储存在 CP 243-2 的组态与 AS-i 网络上的实际组态一致，“CER”（组态错误）LED 也会熄灭。

从组态模式切换到保护模式时，如果有地址为 0 的从站，“SF”（系统错误）LED 会亮。

（2）保护模式

在保护模式，CP 243-2 仅仅与已组态的从站交换数据。“已组态”是指存储在 CP 243-2 中的从站的地址和组态数据与实际的从站的一致。主站根据存储的从站信息轮询各从站，并与实际的从站比较，进行诊断。

12.2.2 用 AS-i 向导组态 AS-i 网络

CP 243-2 是具有扩展寻址功能的 AS-i 主站。通过 S7-200 的编程软件 STEP 7-Micro/WIN 的 ASI 向导，可以对已组态地址的 AS-i 从站分配通信接口区，在联机（在线）模式可以修改从站地址、读出从站的信息。作者使用的是 STEP 7-Micro/WIN 的 V4.0 SP6，可以在网站 www.f108.com 下载该软件。

打开 STEP 7-Micro/WIN，将项目名称修改为“ASI”，CPU 为 CPU 224。本章的例程在随书光盘的文件夹“\Project\ASI”中。

执行菜单命令“工具”→“AS-i 向导”，打开 AS-i 向导。

1．指定需要编辑的 AS-i 配置

在向导的第 1 页（见图 12-5，只给出了对话框关键的部分），如果选中“改变 AS-i 从站地址”，可以在线设置每个 AS-i 从站的地址。为此首先应在 STEP 7-Micro/WIN 和 S7-200 CPU 之间、CP 243-2 和 AS-i 从站网络之间建立起通信连接。如果选中“映射 AS-i 从站”，AS-i 向导将建立在程序和 AS-i 从站之间传输数据所需的项目代码，并自动生成映射到 AS-i 的 I/O 地址的符号。上述操作可以脱机（离线）进行（未连接 S7-200 CPU、CP 243-2 和从站网络）。如果联机使用向导，向导可以提供 CP 243-2 模块的信息，并比较配置的和实际的 AS-i 网络。例程“ASI”选择的是“映射 AS-i 从站”，脱机组态从站。

每一页的操作完成后，点击“下一步 >”按钮，开始 AS-i 向导下一步的操作。

2．设置 AS-i 模块的位置

在向导的第 2 页（见图 12-6）设置模块的位置。紧靠 CPU 模块的是 0 号扩展模块，可以人工输入模块的编号，也可以点击“读取模块”按钮，自动读取已安装好的 AS-i 模块的位置。如果建立了计算机与 CPU 的通信连接，向导会列出与 PLC 连接的所有 AS-i 模块。可以选中其中一个“联机”的模块，或者输入一个未列入模块列表中的“脱机”模块的位置。

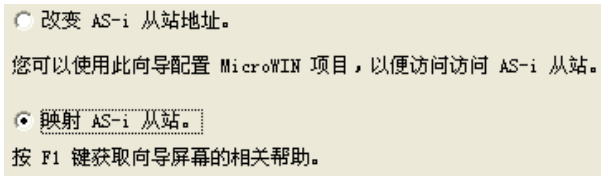


图 12-5　选择指定从站模块地址的方法

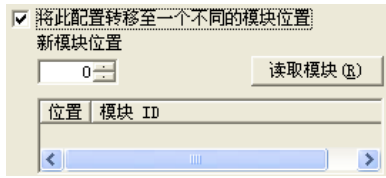


图 12-6　设置 AS-i 模块的位置

3．设置模块的 I/O 地址

在向导的第 3 页（见图 12-7），设置模块的 I/O 地址。

在 S7-200 的映像区中，CP 243-2 占用 1 个数字量输入字节作为状态字节，1 个数字量输出字节作为控制字节。8 个模拟量输入字和 8 个模拟量输出字用于存放 AS-i 从站的数字量/模拟量输入/输出数据、AS-i 的诊断信息、AS-i 命令与响应数据等。

对于 S7-22x CPU 来说，CP 243-2 相当于两个扩展模块，即一个 8DI/8DO 的数字量模块和一个 8AI/8AO 的模拟量模块。CP 243-2 的地址实际上是 AS-i 从站与 CPU 通信用的缓存区。

例程“ASI”的 CPU 模块为 CPU 224，它本身分别占用了两个输入、输出过程映像字节（IW0 和 QW0），因为例程中 AS-i 模块紧靠 CPU 模块，AS-i 模块的数字量输入、输出字节的起始字节地址均为 2，模拟量输入、输出字节的起始字节地址均为 0。

4．设置 AS-i 从站设备的类型

在向导的第 4 页（见图 12-8），设置要组态的 AS-i 从站设备的类型。如果是联机配置从站模块，CP 243-2 将会自动确定网络上的从站类型，所以不需要组态。如果是脱机配置模块，则必须指定网络上的从站类型。

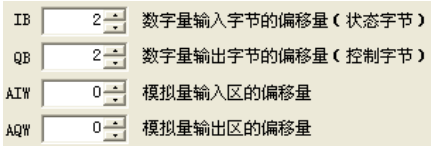


图 12-7　设置 AS-i 主站模块的地址

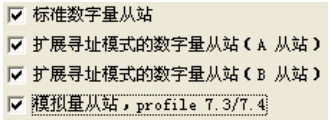


图 12-8　设置 AS-i 从站的类型

数字量（开关量）从站分为标准型和 AB 型。如果只选中“标准数字量从站”，在下一页只能配置 31 个标准从站设备。如果选中“A 从站”，可以使用扩展编址模式中的 A 从站的地址 1A～31A。如果选中“B 从站”，可以使用扩展编址模式中的 B 从站的地址 1B～31B。

如果需要配置模拟量从站，必须选中复选框“模拟量从站，profile 7.3/7.4”，才能在向导中显示模拟量从站配置对话框。例程“ASI”同时选中了 4 种类型的从站（见图 12-8）。

5．组态数字量从站

第 5 页（见图 12-9）组态数字量从站。在联机模式，用图中的表格显示网络中各 AS-i 数字量从站的配置，无须编辑表格中的从站配置。可以增加从站，或修改网络中每个数字 I/O 点预定义的符号名。在脱机模式，表格中不会显示数字量从站的信息。必须指定网络中的数字量从站的类型，并定义每台从站的 I/O 点的符号。

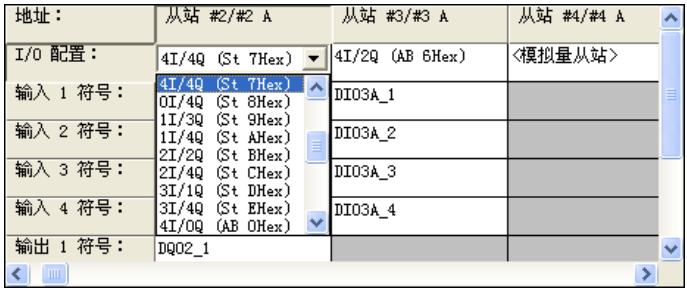


图 12-9　组态数字量从站

点击“I/O 配置”行某一单元右边隐藏的▼按钮，用出现的下拉式列表选中某一从站类型，将会自动生成该从站的 I/O 点的符号，用户可以修改这些符号。符号中的“DI”和“DQ”分别表示数字量输入和数字量输出。从站类型中的“St”为标准从站，“AB”为 A、B 型从站，“Hex”为十六进制数的符号，各种型号的模块均有十六进制数表示的 ID 代码。

例程“ASI”在脱机模式下组态从站地址，地址 1A 和 2A 被标准从站占用，因此地址 1B 和 2B 不能供其他从站使用。地址 3A 组态的是 AB 型从站，地址 3B 组态了另一个 AB 型从站。地址 4 和 5 被模拟量从站占用，因此地址 4A、4B 和 5A、5B 不能用于数字量从站。

各从站各 I/O 点均有自动指定的符号，用户也可以修改它们。用鼠标左键左右拖动图 12-9 下面的水平方向的滚动条，可以看到所有已组态的从站。

6．组态模拟量从站

第 6 页是“指定模拟量从站”（见图 12-10），如果在联机模式进入向导，在网络中检测到模拟量从站，或者在指定从站类型时选中了模拟量从站（见图 12-8），将会出现模拟量从站表。组态模拟量从站与组态数字量从站的方法基本上相同。4 号模拟量从站的第 2 个输入通道默认的符号为 AI04_2。

7．为组态分配 V 存储区

第 7 页是“为配置分配存储区”对话框（见图 12-11），点击“建议地址”按钮，推荐的地址区递增，也可以直接输入希望的地址区的首地址。需要的 V 存储区的大小与组态的从站的类型和个数有关。

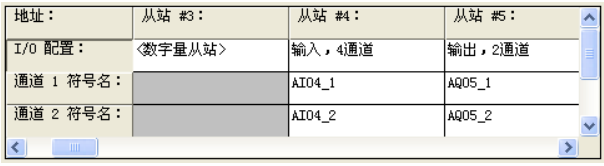


图 12-10　组态模拟量从站

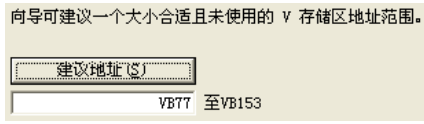


图 12-11　为组态分配 V 存储区

8．生成项目部件

向导的最后一页（见图 12-12）给出了自动生成的子程序、全局符号表和存放组态信息的 V 存储区，以及 AS-i 配置的名称。

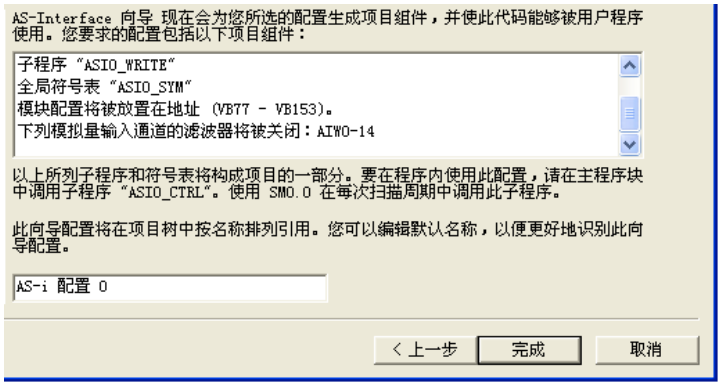


图 12-12　生成的项目文件

点击“完成”按钮，完成从站组态，在左边窗口的指令树的文件夹“\符号表\向导”中，可以看到自动生成的符号表“ASI0_SYM”（见图 12-13），双击打开它后，可以看到为 AS-i 网络中的各 I/O 点自动分配的 V 存储区的地址和符号。用户程序用这些地址来访问 AS-i 从站的 I/O 点。重复的符号名用绿色波浪线标出，带有无效符号的名称用红色标出，可以用汉字作 I/O 点的符号名。应在使用之前将 AS-i 模块的组态下载到 S7-200 CPU。

图 12-13　AS-i 的符号表

12.2.3　AS-i 通信的编程

1．调用 ASIx_CTRL 子程序

将图 12-13 左边窗口的文件夹“\程序块\向导”中的子程序“ASIx_CTRL”拖放到 OB1

中（见图 12-14），其中的 x 是 AS-i 模块所在的槽位号。SM0.0 的常开触点一直接通，每次扫描时都执行 ASIx_CTRL，才能保证正常的 AS-i 网络通信。必须为每个 CP 243-2 模块编写一条 ASIx_CTRL 指令。ASIx_CTRL 指令根据 ASIx_SYM 符号表中的定义，在 AS-i 模块和 CPU 的 V 存储区之间复制从站 I/O 数据。ASIx_CTRL 只有一个输出变量 Error（错误信息），可以在 ASI 向导的在线帮助中查阅错误代码的意义。

CP 243-2 占用的 CPU 的数字量输入、输出字节（本例程为 IB2 和 QB2）分别是 AS-i 的状态（错误）字节和命令字节。它们各位的意义见表 12-2 和表 12-3。

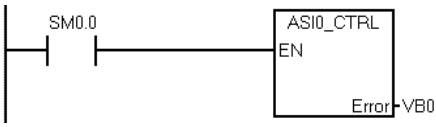


图 12-14　调用 ASI0_CTRL 子程序

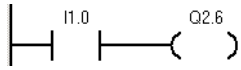


图 12-15　控制 ASI 通信位

表 12-2　状态字节

位	名　称	意　义
0	ASI_MODE	0 为保护模式，1 为组态模式
1	CP_READY	1 表示 CP 243-2 可用
6	ASI_RESP	AS-i 命令接口的响应位

表 12-3　控制字节

位	名　称	意　义
0～5	BS0～BS5	Bank 选择位，取值范围为 0～63
6	ASI_COM	AS-i 命令接口的工作位
7	PLC_RUN	0：CPU 为 STOP 模式，1：CPU 为 RUN 模式

主站发送数据之前，应将控制字节中的“ASI_COM”（Q2.6）置位为 1（见图 12-15）。下载程序后，可以通过 AS-i 专用的符号表（见图 12-13）中的符号地址或绝对地址访问 AS-i 从站的输入、输出变量。

2．CP 243-2 的 Bank

分配给 CP 243-2 的 8 个模拟量输入字和 8 个模拟量输出字（本例从 AIW0 和 AQW0 开始），可以被切换为称为 Bank（库）的 64 个输入区和 64 个输出区，每个区的长度为 8 个字。用控制字的第 0～5 位选择 Bank。Bank0 用于标准从站或 A 从站的数字量输入，Bank1 用于 AS-i 诊断，Bank2～Bank15 用于命令调用的响应数据，Bank31 用于 B 从站的数字量输入。Bank32～Bank47 是从站 1～31 的模拟量输入。

3．用 ASI 命令实现主站的控制和诊断功能

CP 243-2 用各种命令来实现主站的控制功能，AS-i 命令的详细资料见随书光盘中的文件《CP 243-2 AS-i Master Manual》的第 5 章。

子程序 ASIx_READ 从指定的库读取数据（见图 12-16），并将该数据存放在由指针 DB_Ptr 指定的 16B 的 V 存储区。子程序 ASIx_WRITE 将数据写入 CP 243-2 中指定的库（见图 12-17），

用指针 DB_Ptr 指定 16 个字节的 V 存储区的源地址。

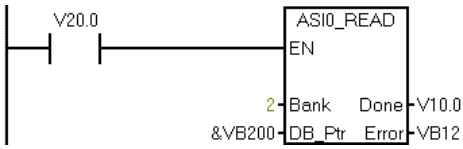


图 12-16　调用 ASI0_READ

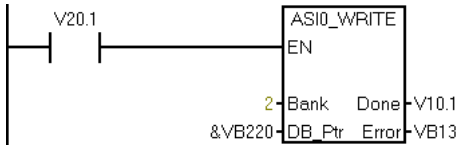


图 12-17　调用 ASI0_WRITE

以执行命令 Change_AS-i_Slave_Address（改变 AS-i 从站地址）为例，由 CP 243-2 的用户手册可知，该命令的发送缓冲区的首字节为命令代码 16#0D，第 2 和第 3 个字节分别是从站老的地址和新的地址。可以用指令或用 STEP 7-Micro/WIN 的状态表（相当于 STEP 7 的变量表）将上述信息写入 VB220～VB222，然后调用图 12-17 中的子程序 ASIx_WRITE，将 VB220～VB222 中的信息发送到从站。用子程序 ASI0_READ 接收从站发送的响应信息（见图 12-16），存放在 VB200 开始的 V 存储区。由 CP 243-2 的用户手册可知，该命令的返回信息有两个字节，第 1 个字节为命令代码 16#0D，第 2 个字节为命令的状态字节。

4．用 CP 243-2 的命令诊断 AS-i 从站

通过 CP 243-2 的命令代码 16#30，读取激活的 AS-i 从站列表 LAS、检测到的 AS-i 从站列表 LDS、永久性的 AS-i 从站列表 LPS 和 AS-i 规范的标志。程序与图 12-15～图 12-17 基本上相同，ASIx_WRITE 的 Bank 值为 2，仅需要发送一个字节的命令代码 16#30。用 ASIx_READ 读取的数据存放在 DB_Ptr 指定的 V 存储区，前 18 个字节在 Bank2，后 10 个字节在 Bank3。Bank2 的第 0 号字节为 16#30，1 号字节为命令状态字节，2～9 号字节为 LAS，10～17 号字节为 LDS。Bank3 的 0～7 号字节为 LPS。每个从站占二进制的一位，0 号从站在第 1 个字节的最高位，7 号从站在第 1 个字节的最低位。

Bank3 的 8 号和 9 号字节为标志字节，标志的详细信息请查阅 CP 243-2 的用户手册。

12.3　CP 343-2P 作主站的 AS-i 网络的组态与编程

12.3.1　组态 AS-i 从站

1．CP 343-2 与 CP 343-2P 主站模块

CP 343-2、CP 343-2P 是用于 S7-300 的 AS-i 主站模块，它们可以安装在中央机架上，也可以安装在 ET 200M 的机架上。CP 343-2P 可以上载从站的信息，对从站的参数组态。

CP 343-2 前面板（见图 12-18）的 LED 指示灯 SF、APF、CER、AUP 和 CM 的意义与 CP 243-2 的相同（见表 12-1），RUN LED 亮表示模块运行正常。图 12-19 是 AS-i 网络示意图。

用面板上 SET 按钮下面的 10 个数字 LED 可以同时显示 10 个从站的状态，LED“20+”和“10+”亮表示从站地址应加 20 或加 10。如果标有数字 2、5、7 和“20+”的 LED 亮，则表示 22、25 和 27 号从站是活动的从站。

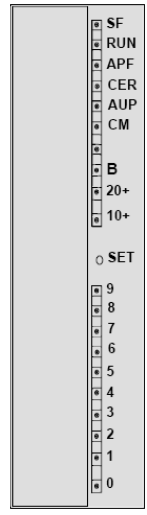


图 12-18　CP343-2

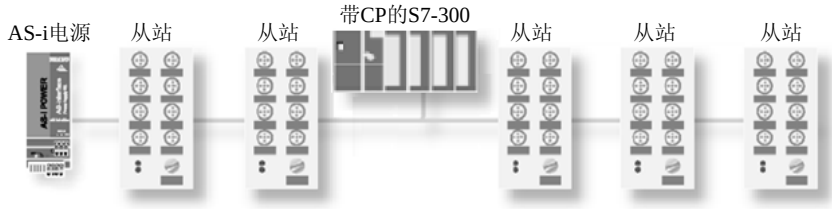


图 12-19　AS-i 网络

2．组态 AS-i 从站的方法

（1）在线读取 AS-i 从站的组态信息

首先用手持式编址器（见图 12-3）设置各 AS-i 从站的地址，然后将所有从站和电源模块连接到 CP 343-2P。在组态硬件时，将 CP 343-2P 插入 S7-300 站点的机架，不组态 AS-i 从站。STEP 7 和 PLC 建立起通信连接后，将上述基本组态下载给 S7-300 站点。

在 HW Config 中双击 CP 343-2P，打开它的属性对话框，点击“AS-i 从站选项”选项卡的按钮“上载给 PG”（见图 12-20）。上载成功后，可以在“从站组态”选项卡看到上载的从站信息。

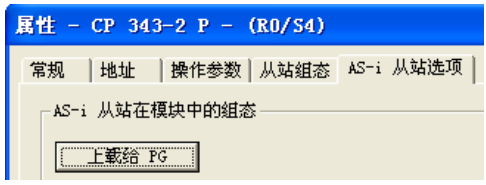


图 12-20　上载 AS-i 从站的组态信息

（2）使用 CP343-2P 面板上的按钮读取 AS-i 从站的组态信息

首先用手持编址器设置各 AS-i 从站的地址，然后将所有从站和电源模块连接到 CP 343-2P 的 AS-i 接口上。PLC 上电后，切换到 STOP 模式，观察 CM（组态）LED 是否点亮，如不亮则按 SET 按钮，使 CM LED 点亮。然后观察连接到 CP 343-2P 上的 AS-i 从站的显示是否正常，如果正常，则再次按 SET 按钮，使 CM LED 熄灭，这时实际的组态被正确读取和保存，同时 CP343-2P 进入保护模式。

（3）用 STEP 7 组态 AS-i 从站

首先按预先的规划，用手持编址器设置各 AS-i 从站的地址。

在 STEP 7 中创建一个项目（见随书光盘中的例程 ASI343_2），在 HW Config 中，将电源模块、CPU 和信号模块插入机架，CPU 模块为 CPU 315-2DP。CPU 的 MPI 接口和集成 DP 接口的默认地址均为 2，未使用集成的 DP 接口。

将 CP 343-2P 插入 4 号槽，双击打开它的属性对话框，在“地址”选项卡，将输入、输出的起始字节地址由 256 改为 60。CP 343-2 占用 16 个输入字节和 16 个输出字节，通过它们来读写从站的输入数据和设置从站的输出数据。

双击“从站组态”选项卡（见图 12-21）的第一行，用出现的对话框（见图 12-22）的“组态”选项卡中的“模块”选择框选择需要的从站。点击“选择内容”按钮，可以用出现的“从站选择对话框”选择从站。

图 12-21　组态 AS-i 从站

CP 343-2 和 CP 343-2P 可以连接 31 个标准从站，或 31 个 A 类从站加 31 个 B 类从站，也可以组合使用。如果设置 x 号从站为标准从站，x 号地址就不能再用于 B 类从站。组态了模拟量从站的地址不能再用于数字量从站。可以通过 16B 输入或 16B 输出地址来访问 31 个数字量标准从站或数字量 A 类从站。本例的从站组态表中的 1 号从站是 2AO 模拟量输出从站（见图 12-22），2 号从站是 4AI 模拟量输入从站，3A 和 3B 号从站是 A/B 型 DI/DO 从站。4 号从站是 DI/DO 数字量标准从站。

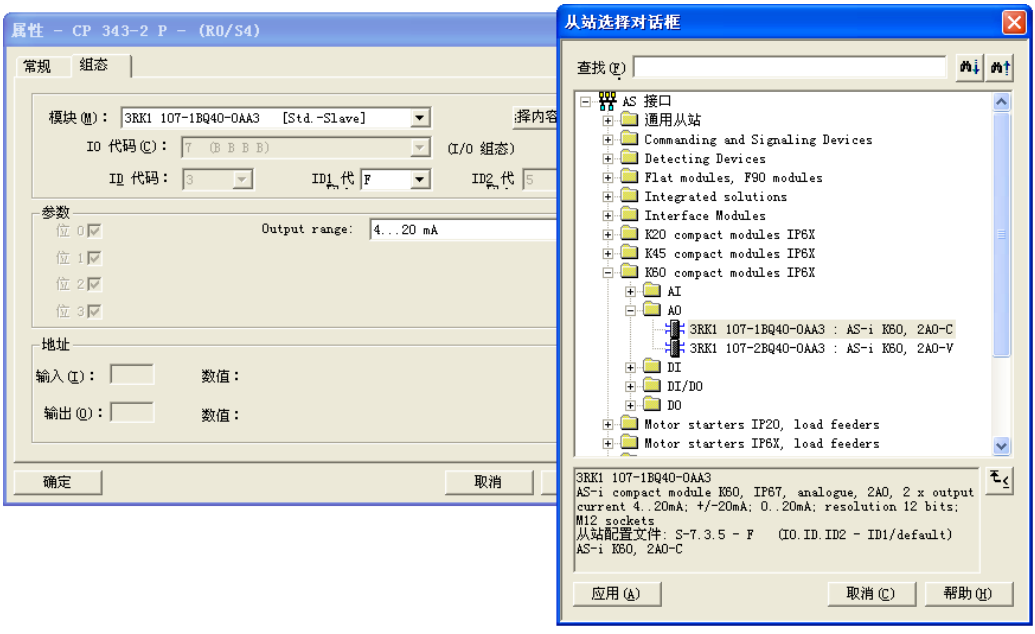


图 12-22　选择 AS-i 从站

3．设置自动寻址编程功能

在运行过程中，如果 AS-i 从站出现故障，为了方便现场更换，可以设置 CP343-2P 的自动寻址功能。双击 HW Config 中的 CP 343-2P，打开它的属性对话框（见图 12-21），用复选框选中“操作参数”选项卡中的“自动寻址编程”。

在从站损坏时，只需要更换一个同样型号的从站，其站地址为出厂时设定的 0，启动运行时它的地址将会自动地分配为被替换的有故障的从站的地址。如果为 CP 343-2P 组态时没有选中“自动寻址编程”，不能自动编址，需要用手持式编址器给新的从站编址后，再更换出现故障的从站。

4．设置诊断功能

如果选中了 CP 的属性对话框的“操作参数”选项卡中的“诊断中断”复选框，出现故障时将调用 OB82。可以通过 OB82 的局部变量进行诊断，或在 OB82 中调用 SFC 51 读取数据记录中的诊断数据。如果选中了该选项，但是没有创建和下载 OB82，中断事件发生时 CPU 将切换到 STOP 模式。

12.3.2 AS-i 通信的编程

1．CPU 访问数字量标准从站或 A 类从站的数据

如果数字量标准从站或 A 类从站的地址在过程映像输入/输出区内，可以直接用 I/Q 地址区按位读写从站的数据区。如果从站的地址在过程映像输入/输出区之外，应使用 PI/PQ 地址区访问 AS-i 从站，但是不能按位进行访问。下面是在 OB1 中访问 3A 号和 4 号从站的例子：

程序段 1：访问标准从站或 A 类从站的数据

```
A       I61.2          //读取 3 号 A 类从站的输入位（见图 12-21）
AN      I62.4          //读取 4 号从站的输入位
=       Q62.7          //改写 4 号从站的输出位
```

2．访问 B 类从站的数据

对于 B 类从站，需要调用 SFC 58/59，来读写 CP343-2P 的 150 号数据记录区。每两个 B 类从站的 I/O 点占一个字节，从站 3B 是 2DI/2DO 模块，它的输入/输出数据在 SFC 59/58 读写的第 2 个字节（DBB1 和 DBB17）的低 4 位。双击图 12-21 中的从站 3B，从打开的“从站选择对话框”下面的从站信息（见图 12-23）的最后两行可知，它的第 0 位和第 1 位为输出点，第 2 位和第 3 位为输入点。

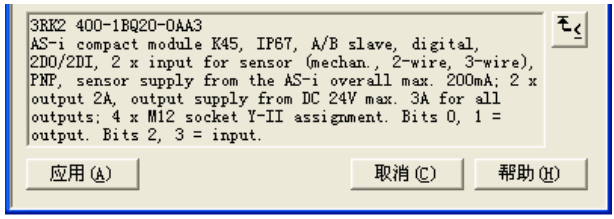


图 12-23　2DI/2DO AS-i 从站模块的信息

程序段 2：读写 B 类从站

```
CALL   "RD_REC"                          //调用 SFC 59 读取 B 类从站的数据
  REQ          :=TRUE                    //读请求为 1
  IOID         :=B#16#54                 //输入/输出模块时为B#16#54
  LADDR        :=W#16#3C                 //CP模块地址（十进制数 60）
  RECNUM       :=B#16#96                 //B类从站的数据记录号 150
  RET_VAL      :=MW2                     //错误代码
  BUSY         :=M0.0                    //为 1 表示读取过程未完成
  RECORD       :=P#DB1.DBX0.0 BYTE 16    //存放读取的数据的地址区
A        DB1.DBX    1.2                  //从站 3B 的第 1 个输入点
S        DB1.DBX    17.0                 //从站 3B 的第 1 个输出点
```

```
A     DB1.DBX    1.3                          //从站 3B 的第 2 个输入点
R     DB1.DBX    17.0                         //从站 3B 的第 1 个输出点
CALL   "WR_REC"                               //调用 SFC 58 将数据写入 B 类从站
  REQ            :=M0.1                       //写请求信号
  IOID           :=B#16#54                    //输入/输出模块时为B#16#54
  LADDR          :=W#16#3C                    //CP模块地址 (十进制数 60)
  RECNUM         :=B#16#96                    //B类从站的数据记录号 150
  RECORD         :=P#DB1.DBX16.0 BYTE 16      //存放要写入的数据的地址区
  RET_VAL        :=MW4                        //错误代码
  BUSY           :=M0.2                       //为 1 表示写过程未完成
```

3．CPU 访问模拟量从站的数据

需要调用 SFC 59/SFC 58 读/写数据记录 DS140～DS147 来访问模拟量从站的数据。每个数据记录区为 128B，每个模拟量通道占用 2B，每个从站占用 8B。数据记录编号与从站地址之间的关系见表 12-4。详细的情况可参阅随书光盘中的手册《CP 343-2_343-2P AS-i Master Manual》。

表 12-4　模拟量 AS-i 从站的地址区

数据记录区	对应的模拟量 AS-i 从站	数据记录区	对应的模拟量 AS-i 从站
DS140	1～16 号从站	DS144	17～31 号从站
DS141	5～20 号从站	DS145	21～31 号从站
DS142	9～24 号从站	DS146	25～31 号从站
DS143	13～28 号从站	DS147	29～31 号从站

下面是 OB1 调用 DS140 访问 1 号模拟量从站（2AO）和 2 号模拟量从站（4AI）的程序：

程序段 3：读写模拟量从站的数据

```
CALL   "RD_REC"                               //调用 SFC 59 读取 2 号 AI 从站的数据
  REQ            :=TRUE                       //读请求
  IOID           :=B#16#54                    //B#16#54 为外设输入，B#16#55 为外设输出
  LADDR          :=W#16#3C                    //CP模块地址 (十进制数 60)
  RECNUM         :=B#16#8C                    //DS140 的数据记录号
  RET_VAL        :=MW6                        //错误代码
  BUSY                :=M0.3                        //为 1 表示读取过程未完成
  RECORD         :=P#DB2.DBX0.0 BYTE 16       //存放读取的数据的地址区
L      DB2.DBW     8                          //将 2 号 AI 从站通道 0 的数据
T      MW     20                              //传送到 MW20
L      MW     24                              //将 MW24 中的数据
T      DB2.DBW    18                          //写入 1 号 AO 从站的通道 1
CALL   "WR_REC"                               //调用 SFC 58 将数据写入 1 号 AO 从站
  REQ            :=M0.4                       //写请求信号
  IOID           :=B#16#55                    //B#16#54 为外设输入，B#16#55 为外设输出
  LADDR          :=W#16#3C                    //CP模块地址 (十进制数 60)
  RECNUM         :=B#16#8C                    //DS140 的数据记录号
  RECORD         :=P#DB2.DBX16.0 BYTE 8       //存放要写入的数据的地址区
```

```
RET_VAL     :=MW8                    //错误代码
BUSY        :=M0.5                   //为 1 表示写过程未完成
```

本例程只有一个 AO 从站和一个 AI 从站，为了读取 2 号 4AI 从站的数据，占用了 16B 数据区。如果将 AI 从站组态为 5 号从站，用数据记录区 DS141 来读取它，只需要读取 8B 数据。由此可见，合理分配模拟量从站的地址是非常重要的。

如果有多个模拟量输入从站，其站地址应连续，以减少读取数据记录的次数。如果有多个模拟量输出从站，按同样的方法处理。

4．CP 343-2 的 ASI 命令接口

通过命令接口，用户程序可以控制 AS-i 的主站，例如控制主站的操作模式，读取从站的参数，修改从站的地址或参数。项目 ASI343_2 中的 FC 7（ASI_3422）提供了 AS-i 的用户接口。CPU 通过 FC 7 发送数据请求报文，CP 343-2P 发送数据响应报文。在数据请求报文中使用不同的命令代码（见随书光盘中的文件《CP 343-2_CP 343-2P Manual》的表 6-3），得到不同的响应数据，有的数据请求报文没有响应数据。下面是 OB1 中用来改变从站地址的程序。

程序段 4：

```
      A     M     80.1
      JCN   m001                     //M80.1 为 0 时跳转
      L     B#16#D                   //改变从站地址的命令代码
      T     MB    100
      L     0                        //从站原有的地址
      T     MB    101
      L     5                        //从站的新地址
      T     MB    102
      CALL  FC    7
        ACT       :=TRUE             //为 1 状态时执行命令
        STARTUP   :=FALSE            //如果在CPU启动时调用FC 7，该参数应为 1，
                                       反之应为 0
        LADDR     :=W#16#3C          //CP模块的起始地址（60）
        SEND      :=P#M 100.0 BYTE 3     //存放命令数据的发送缓冲区
        RECV      :=MB12             //存放接收到的返回数据的缓冲区
        DONE      :=M1.0             //发送正确完成时为 1
        ERROR     :=M1.1             //错误标志位
        STATUS    :=MD14                 //状态双字
m001:   NOP   0
```

5．在 OB82 中读取从站的诊断数据

SFC 51 的参数 SZL_HEADER 的数据类型为 STRUCT（结构），在 OB82 的局部变量表最下面的空白行生成临时变量 t_header，数据类型为 STRUCT。双击打开它后，输入该结构的字元素 LENTHDR 和 N_DR，数据类型均为 Word。LENTHDR 是读取的 SSL 或局部 SSL 的数据记录的长度，N_DR 是读取的数据记录的编号。

数据记录 DS1 的长度为 16B，0～3 号字节是数据记录 0（DS0）的内容，与 OB82 的局部变量的 LD8 中的诊断数据相同（见 CP 342-2 用户手册的 6.6 节）。4～6 号字节为固定值（常数）。0～7 号从站出错时，7 号字节的对应位为 1，1 号从站对应于第 1 位，7 号从站对应于第

7 位。8～15 号从站出错时，8 号字节的对应位为 1，依此类推。0B～31B 号从站出错时，第 11～14 号字节的对应位为 1。下面是 OB82 中的程序：

程序段 1：从站诊断

```
L     60                              // CP 343-2 的模块地址
L     #OB82_MDL_ADDR
<>I
BEC                                   //不是 CP 343-2 产生的中断则结束块调用
CALL  "RDSYSST"                       //调用 SFC 51
  REQ          :=TRUE                 //请求信号为 1
  SZL_ID       :=W#16#B3              //读取模块的诊断信息（数据记录DS1）
  INDEX        :=W#16#3C              //CP模块地址 (60)
  RET_VAL      :=MW10                 //错误代码
  BUSY         :=M0.6                 //为 1 表示读过程未完成
  SZL_HEADER :=#t_header              //局部数据区定义的结构变量
  DR           :=P#M 50.0 BYTE 16     //存放读出的数据记录 1 的目标地址
O     M    57.1                       //从站 1 出错
O     M    57.2                       //从站 2 出错
O     M    57.3                       //从站 3 出错
O     M    57.4                       //从站 4 出错
O     M    61.3                       //从站 3B 出错
=     M    70.0                       //错误标志
```

12.4 使用 DP/AS-i Link 20E 的 AS-i 网络的组态与编程

1．DP/AS-i Link 20E 模块

DP/AS-i Link 20 和 DP/AS-i Link 20E 是 DP/AS-i 网关，用来链接 DP 和 AS-i 网络。DP/AS-i Link 20E 具有扩展的 AS-i 功能，其面板见图 12-24，模块上面是 PROFIBU-DP 9 针连接器和 AS-i 接线端子。

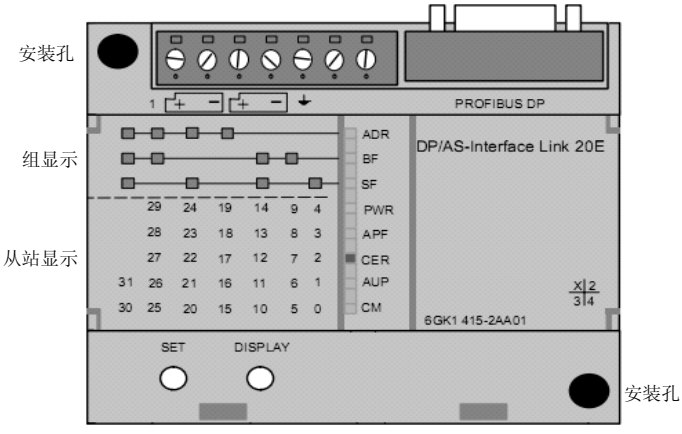


图 12-24　DP/AS-i Link 20E

2．设置 PROFIBUS-DP 站地址

设置 PROFIBUS-DP 站地址的步骤如下：

1）断开与 DP 主站的连接，或令 CPU 为 STOP 模式。

2）连续按“DISPLAY”按钮，直到ADR LED亮。它下面的 7 个LED用来显示DP站地址。最下面的CM LED对应于 1，最上面的BF LED对应于 64（2^6）。亮的LED对应的地址值相加，就是模块的PROFIBUS站地址值。例如在CM、CER和SF LED亮时，PROFIBUS站地址为1+4+32=37。

3）按“DISPLAY”按钮，返回状态显示，原有的 DP 地址不变。按“SET”按钮，代表地址 64 的 BF LED 闪烁。按“SET”按钮确认该位的地址值，该位的 LED 亮；按“DISPLAY”按钮该位的 LED 熄灭，同时下一位的 LED 闪烁。

4）重复上述步骤，设置各位的地址值。

5）设置完所有的位后，被设置为 1 的所有 LED 的颜色交替变化时，按“SET”按钮，设置的 DP 地址被采用。按“DISPLAY”按钮，新的站地址被取消。

3．读取 AS-i 从站信息

读取 AS-i 从站信息的方法与 CP 243-2 的相同，用面板上的 LED 显示读取的从站状态。ADR、BF 和 SF 组成的“组显示”LED 用来显示当前选中的从站的列，PWR 到 CM 这 5 个 LED 用来显示选中的列的哪些从站处于活动状态。假设“组显示”中的 BF 和 SF LED 亮（SF 为最低位），表示选中了第 3 组（从右往左第 3 列从站），此时如果“从站显示”LED 中的 PWR 和 CER LED 亮，第 3 列从站中的 14 和 12 号从站是活动的（active）从站。

LED 常亮表示从站类型为标准从站或 A 类从站，闪烁表示为 B 类从站。

4．AS-i 网络组态

在 STEP 7 中创建一个项目（见随书光盘中的例程 ASI_Link），在 HW Config 中，将电源模块、CPU 和信号模块插入机架，CPU 模块为 CPU 315-2DP（见图 12-25）。

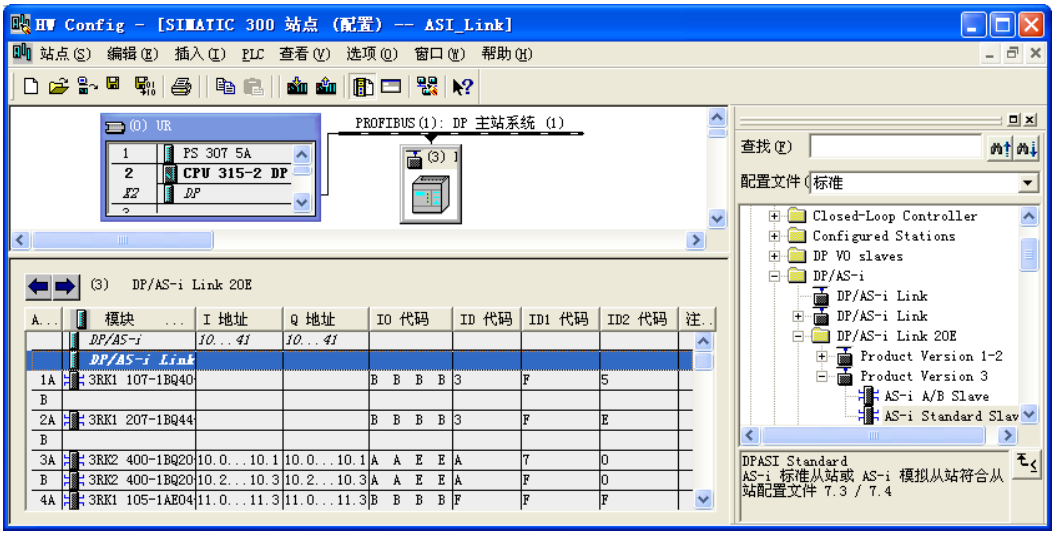


图 12-25　组态 DP/AS-i Link

双击机架中 CPU 模块内标有 DP 的行，点击出现的对话框的“常规”选项卡中的“属性”

按钮，在出现的对话框的“参数”选项卡中，点击“新建”按钮，生成一条 PROFIBUS-DP 网络。采用默认的参数，CPU 315-2DP 为 DP 主站，站地址为 2，网络的传输速率为 1.5 Mbit/s，配置文件为“DP”。点击“确定”按钮，返回 HW Config。

打开图 12-25 右边硬件目录窗口的文件夹“\PROFIBUS DP \DP/AS-i \DP/AS-i Link 20E”，将其中的“Product Version 3”拖放到左边窗口的 PROFIBUS 网络线上。在自动打开的“属性－PROFIBUS 接口”对话框中，设置它的 DP 站地址为 3，点击“确定”按钮，返回 HW Config。

用 DP/AS-i Link 20E 面板设置的 DP 从站地址，应与 HW Config 组态的站地址相同。

选中 3 号从站，双击下面表格中的第一行，打开 DP/AS-i Link 20E 属性对话框，可以设置它的地址区，上传 AS-i 从站的组态信息，设置诊断中断和自动寻址编程功能等属性。

将右边硬件目录窗口中的“AS-i Standard Slave”（标准从站）拖放到左边下面窗口的 1A 所在的行。双击该行，用出现的对话框的“组态”选项卡的“模块”选择框选择需要的从站（见图 12-26 的左图）。点击“选择”按钮，在出现的“从站选择对话框”选中 2AO 从站。

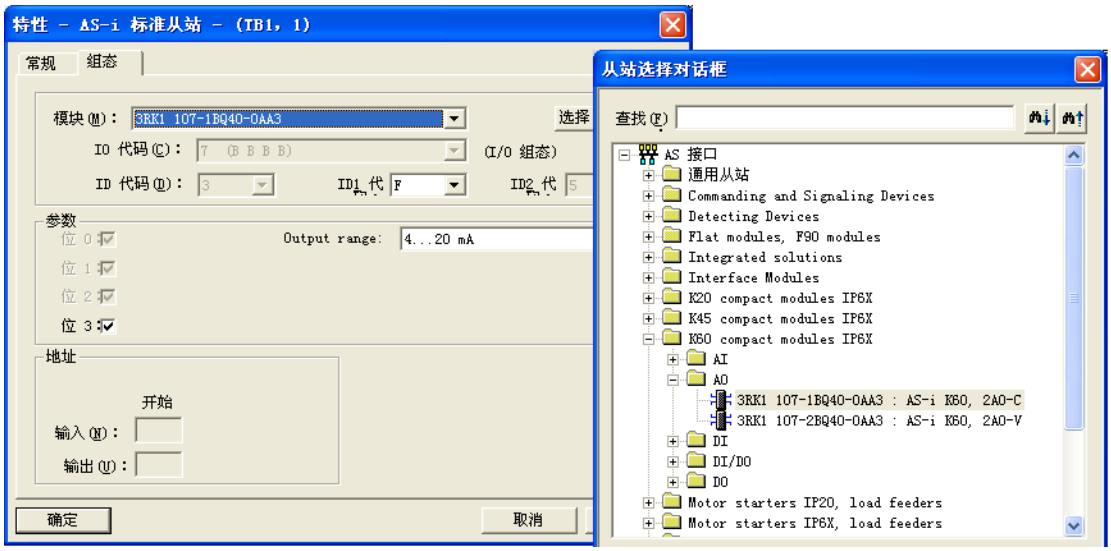


图 12-26　选择 AS-i 从站

本例组态的 AS-i 从站与项目 ASI343_2 的相同。1 号从站是 2AO 模拟量输出从站，2 号从站是 4AI 模拟量输入从站，3A 和 3B 号从站是 A/B 型数字量 DI/DO 从站。4 号从站是 DI/DO 数字量标准从站。

5．DP/AS-i Link 20E 的编程

CPU 读写 AS-i 从站的程序与项目 ASI343_2 的基本上相同（见 12.3.2 节）。其区别仅在于 DP/AS-i Link 20E 的输入、输出地址区的起始字节地址为 10（见图 12-25 下面表格的第一行）。具体的程序见项目中的 OB1 和 OB82。

12.5　练习题

1．AS-i 网络有什么特点？

2．哪些设备可以作 AS-i 的主站？

3．通过双重地址赋值，AS-i 主站最多可以处理多少个从站？最多可以处理多少点数字量输入和数字量输出？

4．AS-i 从站出厂时预置了什么地址？怎样设置从站的地址？

5．用 STEP 7-Micro/WIN 的 ASI 向导组态一个 AS-i 系统，设置若干个标准从站、AB 型从站和模拟量从站，编写通信程序。

6．用 STEP 7 组态 CP 343-2P 为主站的 AS-i 网络，设置若干个标准从站、AB 型从站和模拟量从站，编写通信程序。

7．用 STEP 7 组态 DP/AS-i Link 20E 为主站的 AS-i 网络，设置若干个标准从站、AB 型从站和模拟量从站，编写通信程序。

第 13 章　OPC 通信

13.1　OPC 通信概述

1．OPC 的基本概念

OLE是Object Linking and Embedding（对象链接与嵌入）的缩写，是微软为Windows操作系统、应用程序之间的数据交换开发的技术。OPC（OLE for Process Control，用于过程控制的OLE）是嵌入式过程控制标准，是用于服务器/客户机链接的开放的接口标准和技术规范。

不同的供应商的硬件有不同的标准和协议，OPC 作为一种工业标准，提供了工业环境中信息交换的统一标准的软件接口，这样数据用户不用为不同厂家的数据源开发驱动程序或服务程序。最终用户也不用安装各种设备的驱动程序，显著地降低了系统集成的成本。

OPC 是一种开放式系统接口标准，用于在自动化和 PLC 应用、现场设备和基于 PC 的应用程序（例如 HMI 或办公室应用程序）之间，进行简单的标准化数据交换（见图 13-1）。

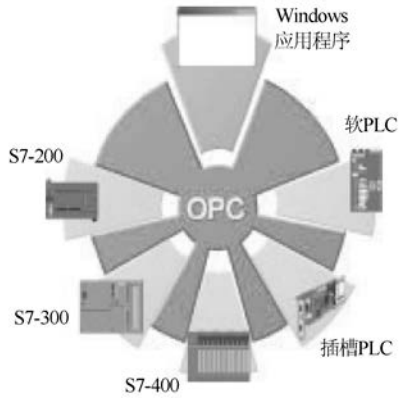


图 13-1　SIMATIC 的 OPC 连接

通过 OPC，可以在计算机上监控、调用和处理 PLC 的数据和事件，允许基于 Windows 的应用程序直接访问不同厂商设备中的过程数据。例如 OPC 接口可以用于将装有 IT 功能的 SIMATIC PLC 连接到 MES（制造执行系统）。

服务器（Server）与客户机（Client）的关系有些像 DP 从站与主站的关系。服务器在通信过程中是被动的，它总是等待客户机发起数据访问。OPC 将数据源提供的数据以标准方式传输到客户机应用程序。

OPC 允许 Windows 应用程序访问过程数据，从而能够轻松地连接不同制造商生产的设备和应用程序。OPC 提供了开放的、与供应商无关的接口，容易使用的客户机/服务器组态，在控制设备（例如 PLC）、现场设备和基于 PC 的应用程序（例如 HMI 或办公应用程序）之间提供标准化的数据交换。

OPC 服务器为连接 OPC 客户机应用程序提供接口。客户机应用程序执行对数据源（例如 PLC 中的存储器）的访问，多个 OPC 客户机可以同时访问同一个 OPC 服务器。

2．SIMATIC NET 软件简介

随书光盘中的 SIMATIC NET 是 PG/PC 通信处理器（CP）的驱动软件和参数化软件。

如果使用 CP 5611/CP 5512，或普通的以太网卡等通信卡，STEP 7 集成的“设置 PG/PC 接口”工具支持这些通信卡。

下列情况需要使用 SIMATIC NET：

1）SIMATIC NET 包含 CP 5613、CP 5614 和 CP 1613、CP 1616 的驱动程序，STEP 7 集成的“设置 PG/PC 接口”工具不支持上述通信卡。将这类 CP 卡插入计算机，在安装 SIMATIC NET 时，将自动地为它们安装驱动程序。

2）如果控制系统的上位计算机软件（例如某些组态软件）不支持西门子的通信协议，不能直接访问 S7-300/400 PLC，需要用 SIMATIC NET 的 OPC 功能来解决这一问题。

3）大型复杂的控制系统有多台计算机和多台 PLC。可以用 SIMATIC NET 来组态 PC 站，然后在 NetPro 中分别组态各 PLC 和各 PC 站点之间的连接。8.3.3 节给出了一个组态 PC 站的例子。

4）使用冗余设计的容错自动化系统 S7-400H 和 CP 1613、CP 1616 时，必须用 SIMATIC NET 来组态冗余通信。

3．站组态编辑器

站组态编辑器（Station Configuration Editor）随软件 SIMATIC NET 提供。它是 PC 站的全新、简单、一致和经济的调试和诊断解决方案。通过使用 NCM PC 或 STEP 7，PC 可以像 SIMATIC S7 设备一样进行组态，并通过网络下载组态信息。

NCM PC 随 SIMATIC NET 软件光盘提供，它是用户为 PC 站组态通信服务的工具。STEP 7 包含 NCM PC，在 STEP 7 的硬件组态环境中可以组态 PC 站。

4．OPC 服务器

OPC 服务器随 SIMATIC NET 软件光盘提供。OPC 服务器有下列功能：

- 数据访问接口 2.05。
- 报警和事件接口 1.02（单一事件）。
- 不同制造商自动化产品之间的标准化。
- 用于不同部件的相同的、用户友好的用户接口。
- 可访问工业以太网和广域网中的计算机。
- 通过客户机接口（C++）的高性能数据存取。
- 通过自动化接口（VB）的OCX数据控制。
- 通过XML DA接口实现互联网通信。
- 变量（条目，Item）的成组化，可以在很短的时间内进行大容量的数据存取。

5．OPC 通信服务

SIMATIC NET OPC 服务器支持以下通信服务：PROFINET IO、PROFINET CBA、MPI、PROFIBUS-DP，使用 MPI、DP 或工业以太网的 S7 通信和 S5 兼容的通信，使用工业以太网的 SNMP（简单网络管理协议）。

SIMATIC NET 除了提供 OPC 服务器外，还提供用于组态和测试 OPC 连接的站组态编辑

器和 OPC Scout，使用这些工具可以将 SIMATIC S7 产品连接到其它 OPC 兼容的应用程序。

13.2 基于 MPI 和 PROFIBUS 的 OPC 服务器与 PLC 的通信

13.2.1 用站组态编辑器组态 PC 站

断电时将 CP 5613 插入计算机主板的插槽内，然后安装 SIMATIC NET。安装后双击打开控制面板中的“设置 PG/PC 接口”，选中出现的对话框中间的选择框内的“CP 5613_5614（MPI)”，点击“确定”按钮退出。此时 CP 5613 处于编程器（PG）模式。

下面介绍基于 MPI 网络 S7 连接的 OPC 服务器与 S7-300/400 的通信的组态方法。

作者使用的软件是 SIMATIC NET 2007 和 STEP 7 V5.4 SP3.1 中文版。安装好 SIMATIC NET 软件后，在计算机的桌面上可看到“Station Configurator”图标，同时在 Windows 的任务栏中出现图标。双击其中的某个图标，打开站组态编辑器（见图 13-2）。如果已经打开了 STEP 7，在组态之前应关闭它。

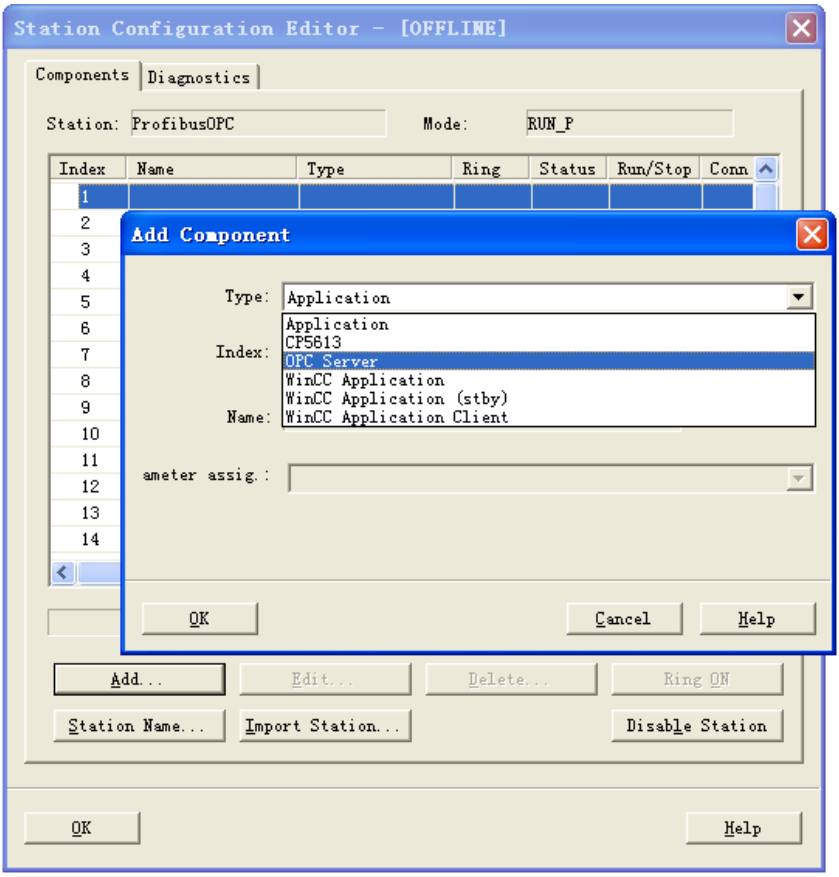


图 13-2 添加 OPC Server

选中编辑器中的 1 号插槽（Index 1），点击“Add”按钮，在出现的“Add Component”（添加组件）对话框中，选中“OPC Server”（OPC 服务器），点击“OK”按钮完成添加操作。

选中 3 号插槽，点击“Add”按钮，在出现的“Add Component”对话框中（见图 13-3），用“Type”（类型）选择框选择添加 CP 5613。下面的选择框只能选择“CP5613_CP5614（PROFIBUS）”。

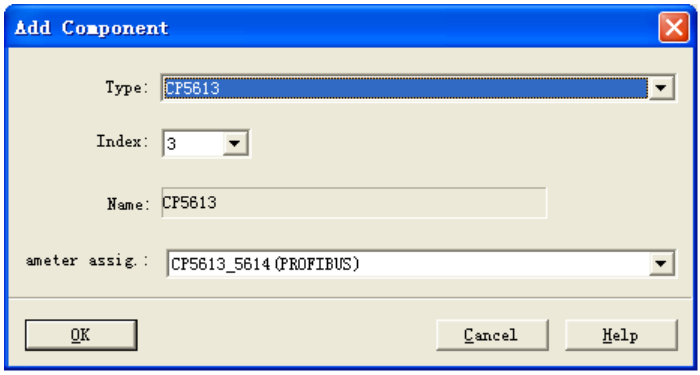


图 13-3　添加 CP 5613

点击“OK”按钮，出现的对话框提示在组态时有关组件的通信不能处于激活状态。

点击“OK”按钮，在图 13-4 所示的组件属性对话框中设置 CP 5613 的 MPI 站地址为 1，波特率为 187.5 kbit/s。建议将 CP 5613 的总线配置文件（Bus profile）设置为“Standard”（标准），也可以设为“DP”。如果 CP 5613 作 DP 主站，则“Bus profile”应设置为“DP”。点击“OK”按钮，设置结束。

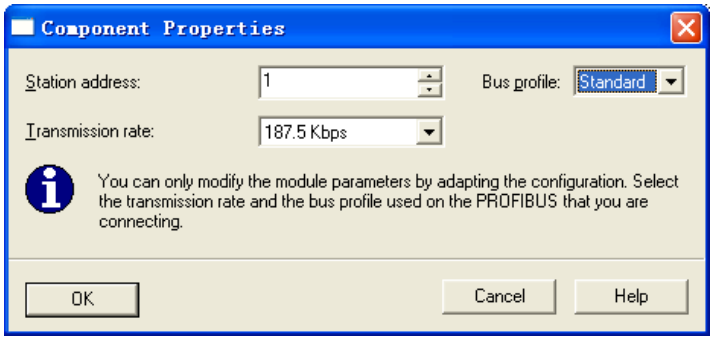


图 13-4　设置 CP 5613 的参数

点击图 13-2 中的“Station Name”按钮，用出现的对话框（见图 13-5）设置 PC 站的名称为“MPIOPC”，点击“OK”按钮确认。PC 站的名称应与 STEP 7 中组态的名称相同。

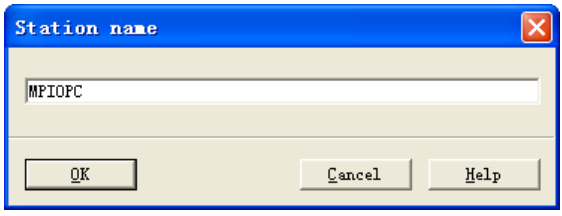


图 13-5　设置 PC 站的名称

图 13-6 是组态好的站组态编辑器的第 1～3 行，第一行的 Status 列的图标上有红色的叉。点击“OK”按钮，关闭站组态编辑器。

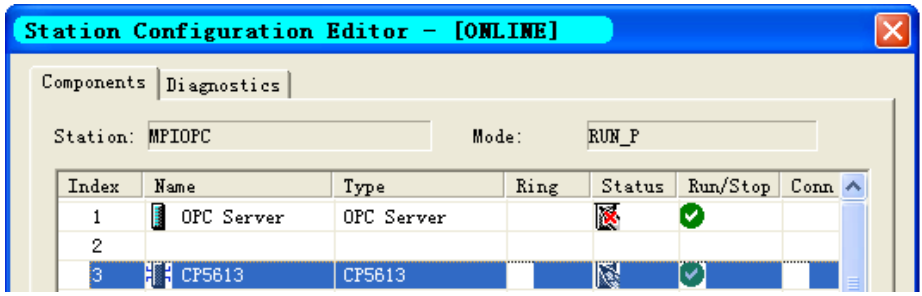


图 13-6　站组态编辑器

13.2.2　组态控制台

组态控制台（Configuration Console）是组态设置和诊断的核心工具，用于 PC 硬件组件和 PC 应用程序的组态和诊断。

正确完成 PC 站的硬件组态后，点击 Windows 工具栏左边的“开始”按钮，执行菜单命令“开始”→“所有程序”→“SIMATIC”→“SIMATIC NET”→“Configuration console”，打开组态控制台。选中图 13-7 左边窗口的文件夹“\Modules\CP5613”中的“General”图标，可以看到 CP 5613 已从默认的 PG（编程器）模式切换到 Configuration mode（组态模式），插槽号（Index）自动指向 3。

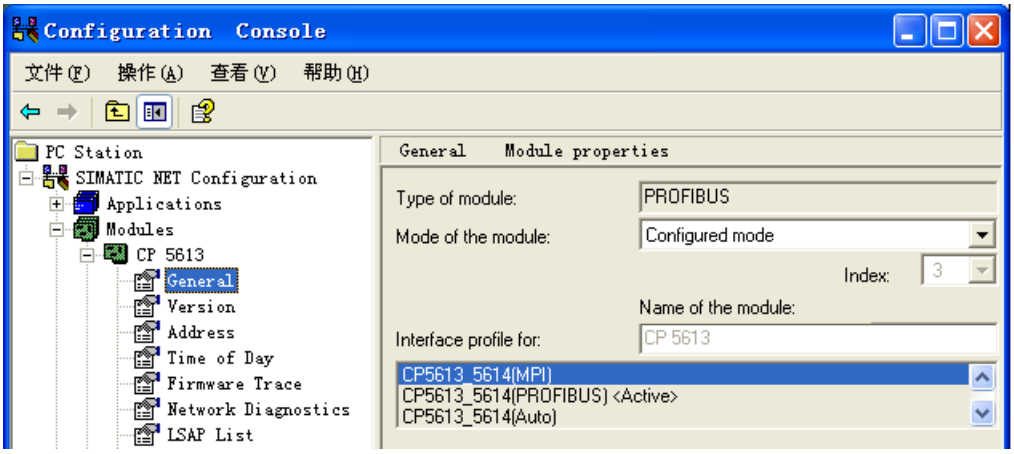


图 13-7　组态控制台

对于较早的 SIMATIC NET V6.1 或 V6.0 版本的软件，需要在上述的对话框中，手动将模块的模式（Mode of the module）从 PG 模式切换到组态模式，并设置 Index 号。

选中左边窗口的“Access Points”（访问点，见图 13-8），双击右边窗口的“S7ONLINE”，选中出现的“S7ONLINE 属性”对话框中的“PC internal（local）”后，点击“确定”按钮。这一设置是为 PC 站组态的下载作准备的。

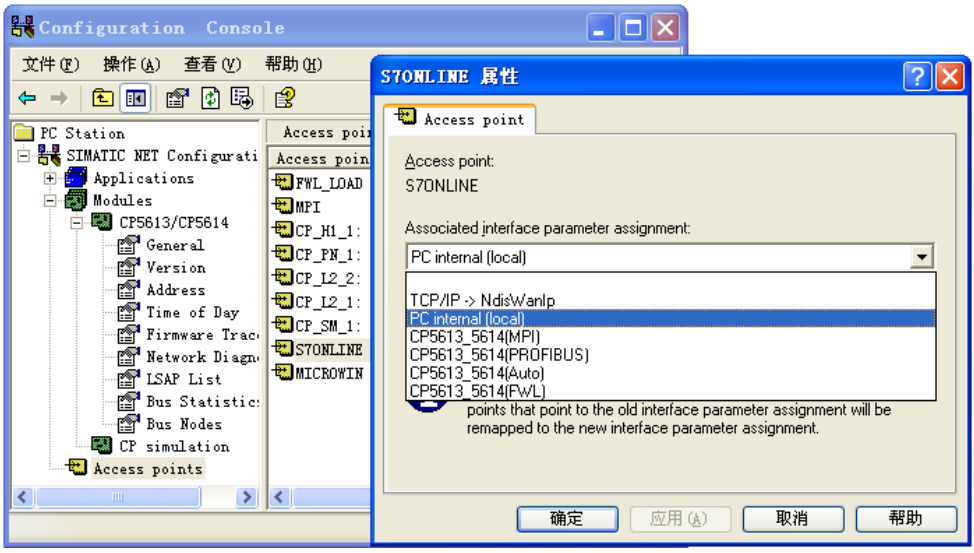


图 13-8　组态访问点

选中组态控制台左边窗口中的“Network Diagnostics”，可以对网络进行诊断（见图 13-9）。

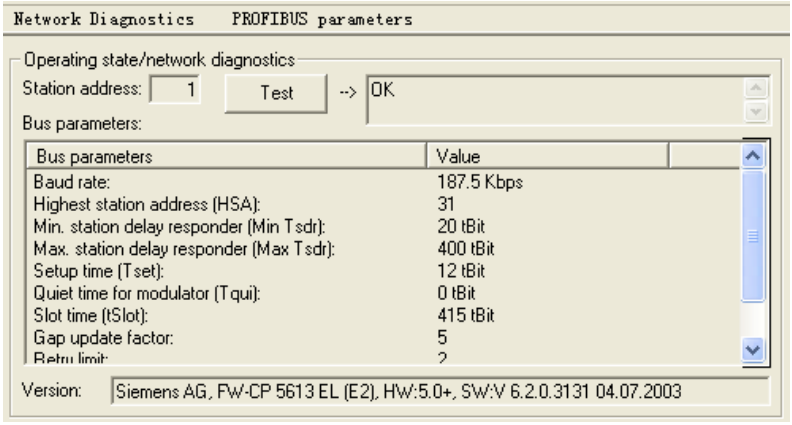


图 13-9　网络诊断

选中组态控制台左边窗口中的“Bus Nodes”，可以查看当前网络上在线的站点。组态结束后，关闭组态控制台。

13.2.3　在 STEP 7 中组态 PC 站点和 PLC

1．组态 PC 站点

打开 SIMATIC 管理器，执行菜单命令“文件”→“新建”，创建一个名为“MPI_OPC”的新项目。用右键点击项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC PC 站点”，插入一个 PC 站。必须将 PC 站默认的名称改为 Station Configuration Editor 中命名的“MPIOPC”。选中它以后，双击右边窗口的“组态”图标（见图 13-10），打开 HW Config（硬件组态）窗口。

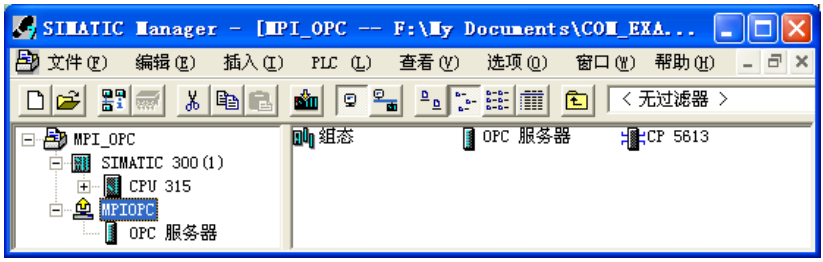


图 13-10　SIMATIC 管理器

在 HW Config 中（见图 13-11），将左边的硬件目录窗口的文件夹“\SIMATIC PC Station\CP PROFIBUS\CP 5613”中的 SW V6.0 SP5 插入 3 号槽。打开硬件目录窗口中的文件夹“\SIMATIC PC Station\User Application\OPC Server”，将其中的 SW V6.3（见图 13-12）插入 1 号槽。

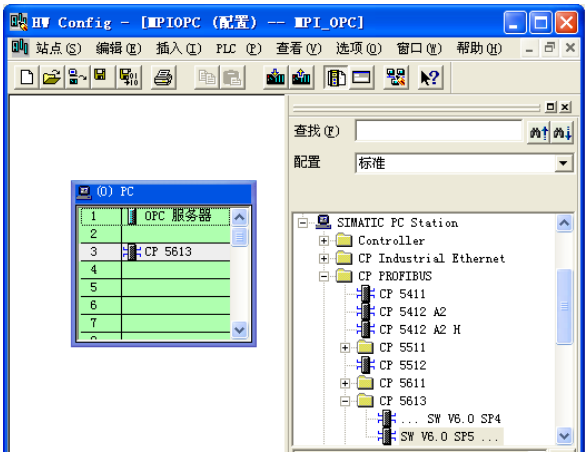


图 13-11　组态 PC 站点

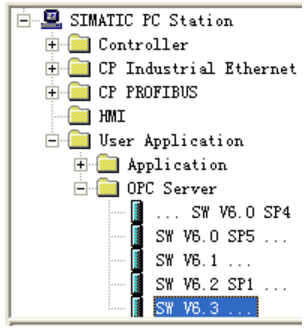


图 13-12　硬件目录窗口

2．组态 CP 5613 的网络参数

双击机架第 3 行的 CP 5613，打开其属性对话框（见图 13-13），用“类型”选择框将 CP 5613 的接口设为 MPI。点击“属性”按钮，在打开的 MPI 接口属性对话框中，将 CP 5613 连接到 MPI 网络上。设置 CP 5613 的 MPI 地址为 1。

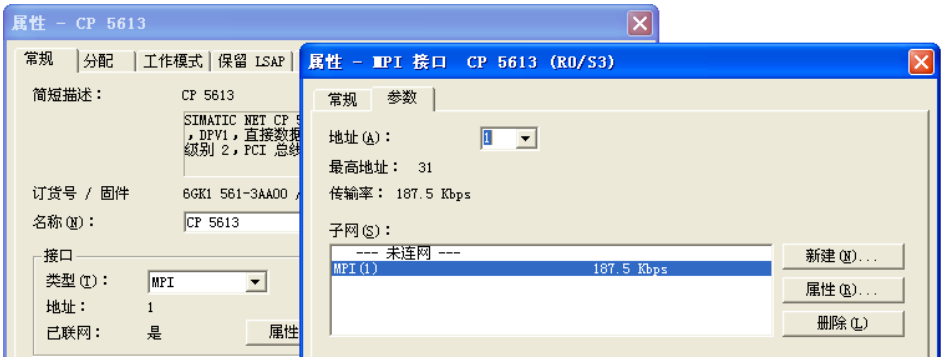


图 13-13　CP 5613 的参数设置

完成 PC 站组件设置后，点击工具栏上的“保存和编译”按钮，保存组态信息。

3．组态 S7-300 站

在 SIMATIC 管理器中，用右键点击项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”，插入一个 S7-300 站。选中它以后，双击右边窗口的“硬件”图标，打开 HW Config 窗口。将 Rack（导轨）拖放到左边的窗口。在机架中插入 CPU 315、电源模块和 I/O 模块。

在 SIMATIC 管理器中生成组织块 OB35 和数据块 DB 1，在 DB 1 中生成一个数组。将图 13-14 中的程序输入 OB1，用组态软件画面上的起动按钮 M0.0 和停止按钮 M0.1 控制电动机 Q4.0。

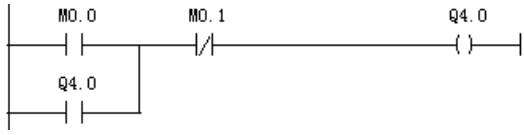


图 13-14　梯形图

在 OB35 中编写程序，每 100ms 分别将 MW10 和 DB1.DBW0 的值加 1 和加 2。

4．建立连接

编译无误后，点击（组态网络）按钮，打开 NetPro（网络组态）窗口（见图 13-15）。双击 CPU 315 所在的小方框，在出现的 CPU 属性对话框中，将 CPU 315 的 MPI 接口连接到 MPI 网络上，默认的站地址为 2。

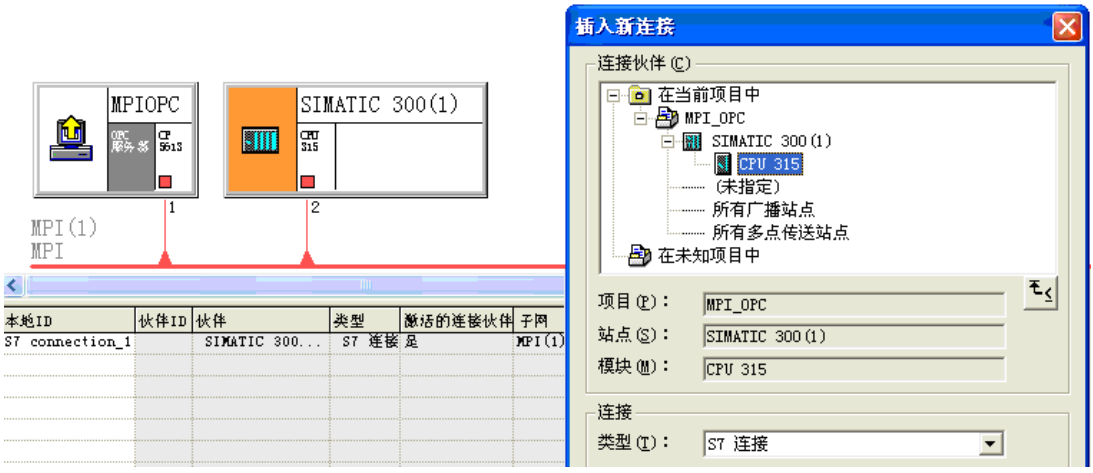


图 13-15　组态 S7 连接

用鼠标选中 OPC Server，双击下面的连接表的第一行，在自动打开的“插入新连接”对话框中，默认的连接对象为 CPU 315，默认的连接类型为 S7 连接。

点击“确定”按钮，生成一个 S7 单向连接，图 13-16 是出现的连接属性对话框。点击“地址详细信息”按钮，可以看到详细的地址信息，其中的“插槽”是 CPU 所在的插槽号。

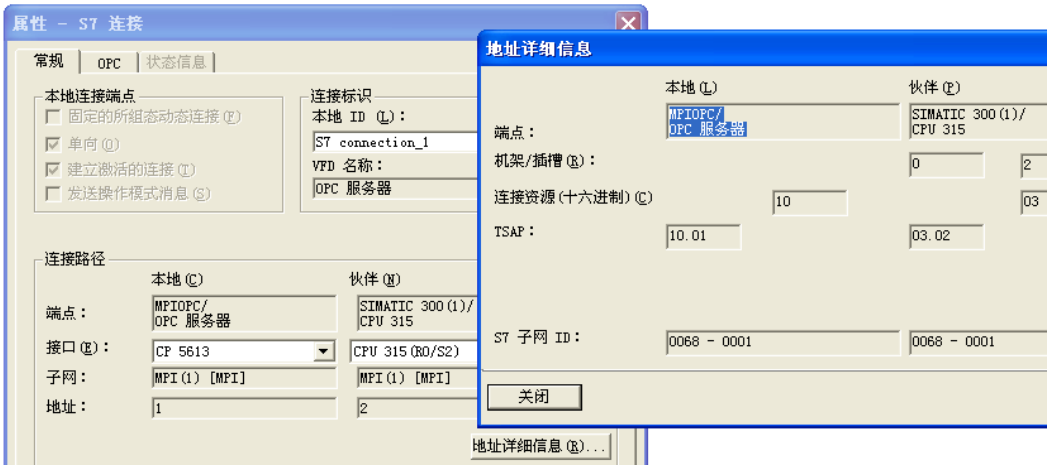


图 13-16　S7 连接属性对话框

最后点击“确定”按钮，返回 NetPro。点击（编译并保存）按钮，保存组态信息。

5．下载组态数据

完成上述组态任务后，用 PROFIBUS 电缆连接 CP 5613 和 CPU 315 的 MPI 接口。

分别选中 CPU 和 PC 站，点击 NetPro 工具栏上的（下载）按钮，将组态数据下载到 CPU 315 和 PC 站。选中 SIMATIC 管理器左边窗口的“块”文件夹，下载程序块和组态信息。

下载成功完成后，打开站组态编辑器窗口（见图 13-17），检查组件状态。图 13-6 中第一行 Status 列的图标上红色的“×”消失，Conn 列出现的连接图标表示连接被激活。

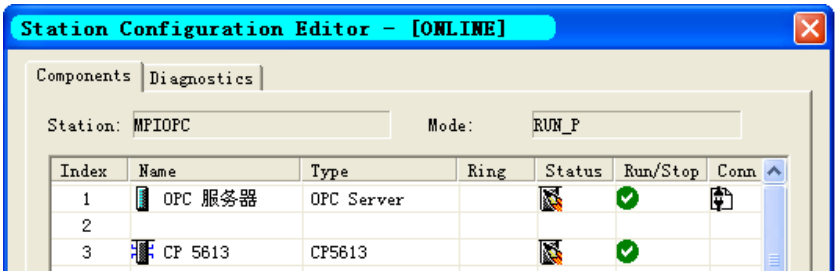


图 13-17　PC 站点的在线状态

13.2.4　在 OPC Scout 中生成 OPC 的条目

1．生成 OPC 的组

OPC Scout 工具随 SIMATIC NET 软件一起提供。下载了 PC 站的组态后，可以用它生成 OPC 的组（Group）和条目（Item，或翻译为“项”），进行 OPC 服务器和 PLC 的数据通信测试。

执行 Windows 菜单命令“开始”→“所有程序”→“SIMAITC”→“SIMATIC Net”→“OPC Scout”，打开 OPC Scout（见图 13-18）。双击图中的“New group”，在出现的“ADD Group”（添加组）对话框中输入组的名称“MPI_OPC”。可以用选择框修改以 ms 为单位的请求刷新的速率，默认值为 500ms，点击“OK”按钮确认。

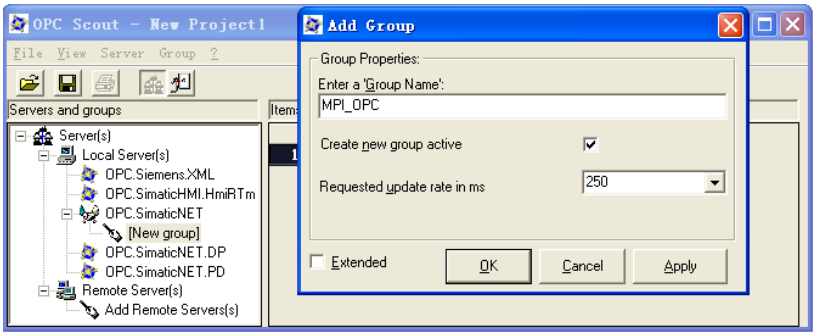


图 13-18　在 OPC Scout 中添加组

2．定义 OPC 的条目

双击已添加的组 MPI_OPC，弹出的“OPC Navigator”（OPC 浏览器）对话框（见图 13-19）显示出所有的连接协议。双击“S7”，显示出在 NetPro 中组态 PC 站时建立的名为“S7 connection_1”的连接（见图 13-15）。子文件夹“\objects”中是 PLC 的各数据区组成的对象树（objects tree），在 STEP 7 中创建的数据块 DB 1 也会在对象树中出现。

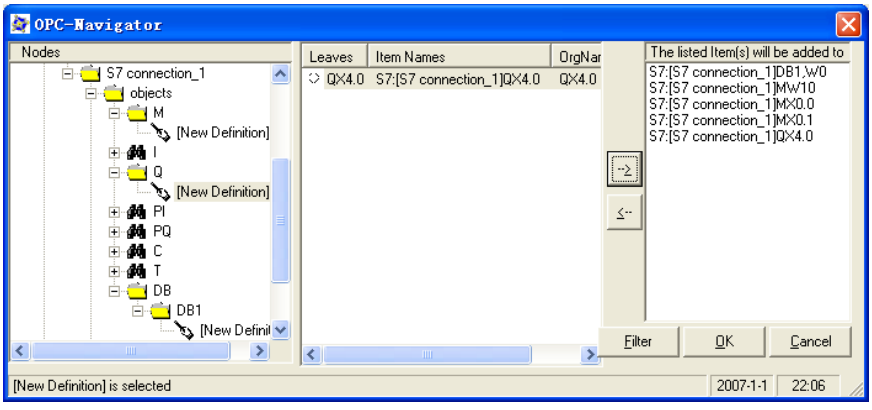


图 13-19　OPC 浏览器

点击 M 区前面带“+”号的小方框，如果“New Definition”（新定义）图标上有红叉标记出现，并不表示有什么问题。双击“New Definition”，在打开的“Define New Item”对话框中（见图 13-20），设置变量的数据类型（Datatype）为 X（位变量），还需要设置字节地址（Address）、位编号（Bit No.）和数据个数（No.Values）。定义完成后，点击“OK”按钮确认。新定义的条目（Item）出现在 OPC Navigator 中间的窗口，左边窗口 M 区的红叉自动消失。

用同样的方法，定义 M0.1、Q4.0 和 MW10。后者的数据类型为 W，No.Values 为 1。点击图 13-19 中的“-->”按钮，将选中的中间窗口的条目移到的右边窗口。选中右边窗口中的某个条目，点击“<--”按钮，可以将它移到中间窗口。

打开对象树中的 DB 1，双击出现的“New Definition”（新定义）图标，在打开的“Define New Item”对话框中（见图 13-21），设置变量的数据类型（Datatype）为 W（字）、地址（Address）为 0、数据个数（No.Values）为 1。定义完成后，点击“OK”按钮确认，生成的条目的名称

为 S7:[S7 connection_1]DB1,W0。

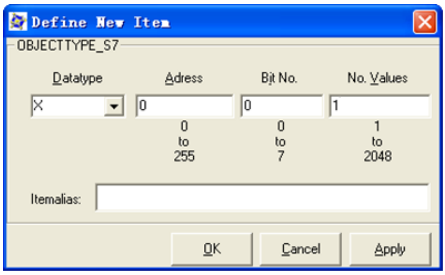


图 13-20　定义新条目

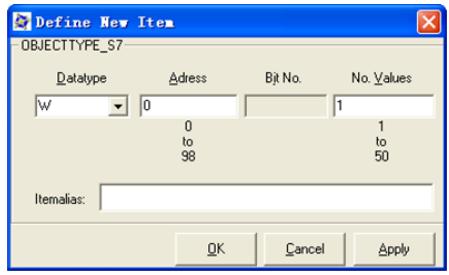


图 13-21　定义新条目

点击图 13-19 中的“OK”按钮，右边窗口中的条目被连接到 OPC 服务器上（见图 13-22）。点击工具栏上发光的灯泡图标，激活当前的组，左边窗口的 MPI_OPC 图标变为绿色。如果条目的“Quality”为“good”，表示已经建立 OPC 服务器与 PLC 的 S7 连接，可以对条目进行读写操作。

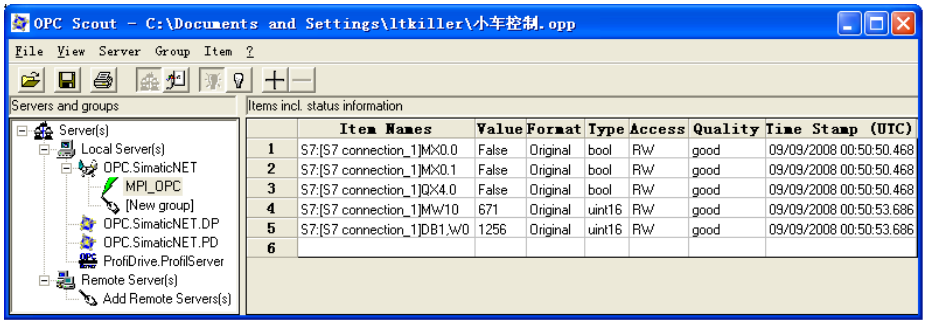


图 13-22　OPC Scout

3．检查变量与 PLC 的连接

将 CPU 315 切换到 RUN 模式，在 CPU 315 每 100ms 执行一次的 OB35 中，分别将 MW10 和 DB1.DBW0 加 1 和加 2，在图 13-22 中可以看到它们的值在不断地动态变化。

双击某个条目（例如 MW10）的“Value”列，可以在出现的对话框中（见图 13-23），对选中的条目进行写操作。点击“OK”按钮，可以看到改写的效果。

OPC Scout 与 PLC 之间的通信与是否打开 STEP 7 无关。

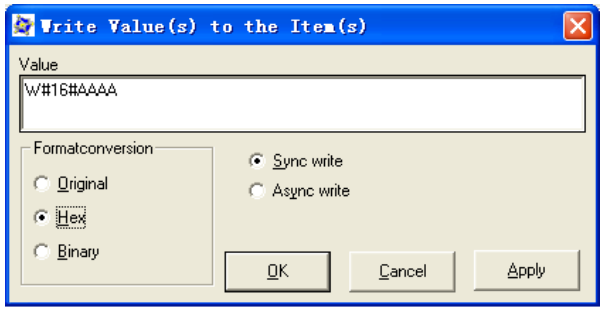


图 13-23　对变量进行写操作

图 13-23 中的 Sync write（同步写方式）的实现较为简单，客户机向服务器发出写请求，然后等待服务器返回信息，当客户机较少，而且与服务器交换的数据量比较少的时候，可以采用这种方式。然而当网络堵塞或有大量的客户机访问时，会造成系统的性能下降。

异步写方式（Async write）的实现较为复杂，客户机向服务器发出写请求后，服务器立刻返回信息，表示请求已被接受，客户机可以进行其他处理。服务器完成读写操作后，通知客户机程序操作完成，并传递相应的信息。因此异步方式的效率更高，能避免多客户机大数据量请求产生的阻塞，可以最大限度地节省 CPU 和网络的资源。

执行菜单命令“File”→“Save Project”，用文件 MPI_OPC.opp 保存 OPC 的组（Group）的设置，以后可以打开该文件，迅速地恢复对组的设置。

13.2.5 基于 PROFIBUS 网络的 OPC 通信的组态

基于 PROFIBUS 和 MPI 的 OPC 通信的组态方法基本上相同。

作者作实验使用的 CP 卡为 CP 5611，在站组态编辑器和 HW Config 中组态 CP 5611 时，总线类型为 Profibus，波特率为 1.5 Mbit/s，总线配置文件可以选 DP 或“标准”。STEP 7 的项目名称为 DP_OPC，OPC Scout 中保存的文件为 DP_OPC.opp（见随书光盘中的同名文件，和同名的 STEP 7 项目）。硬件结构和网络组态见图 13-24 和图 13-25。在站组态编辑器和 STEP 7 中组态的 PC 站的名称均为 DP_OPC。

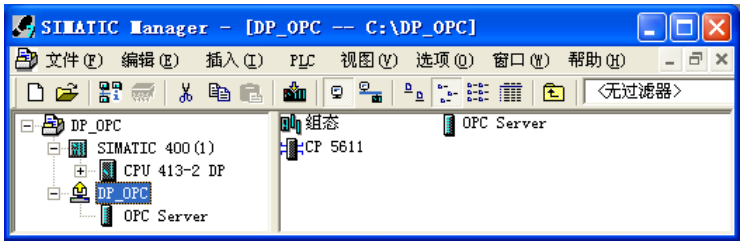


图 13-24　SIMATIC 管理器

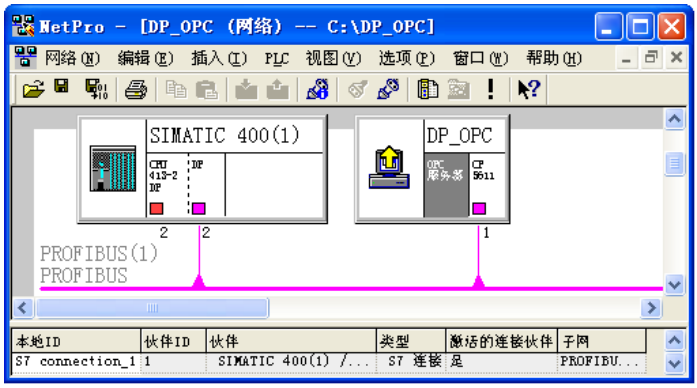


图 13-25　组态 S7 连接

在站组态编辑器和 STEP 7 的 HW Config 中组态 PC 站时，第 1 槽为 OPC Server，第 3 槽为 CP 5611。在下载和运行时，用 PROFIBUS 电缆连接 CP 5611 和 CPU 的 PROFIBUS 接口。

在组态控制台（Configuration Console）和 OPC Scout 中的组态过程和实验结果，与本节前面介绍的组态过程和实验结果几乎完全相同。

13.3　基于 OPC 的组态软件与 S7-300 的通信组态

本例程使用的组态软件是 V6.51 版的组态王，首先安装组态王，在安装驱动程序时，可以只安装常用的 PLC 的驱动程序。

1．生成 OPC 服务器

安装完成后，双击桌面上的“组态王 6.51”图标，出现“组态王工程管理器”对话框（见图 13-26）。点击工具栏上的“新建”图标，生成一个名为“组态王 OPC”的新工程（即新项目）。

图 13-26　组态王工程管理器

双击新生成的工程，出现的对话框提示“未找到加密狗，请装好加密狗然后再试，或忽略进入延时方式”。点击“忽略”按钮，出现的对话框提示将进入演示方式，程序将在两小时后关闭。点击“确定”按钮，打开工程浏览器（见图 13-27）。

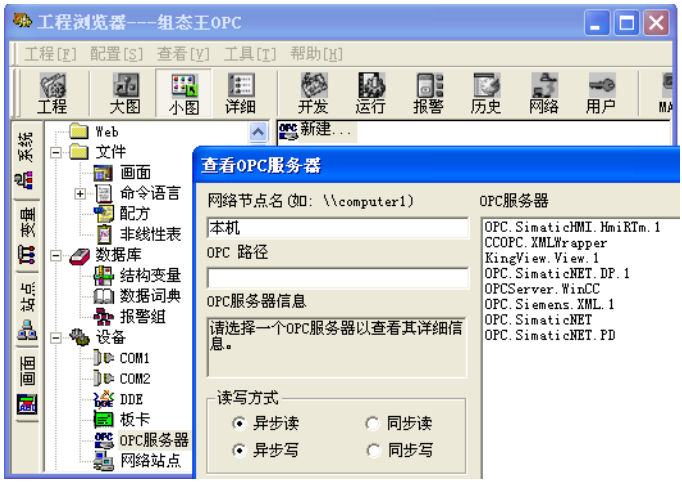


图 13-27　生成 OPC 服务器

选中工程浏览器左边窗口中的“OPC 服务器”，双击右边窗口中的“新建”，在出现的“查看 OPC 服务器”对话框的 OPC 服务器列表中，选中“OPC. SimatcNET”，点击“确定”按钮，创建名为“本机\ OPC. SimatcNET”的 OPC 服务器。

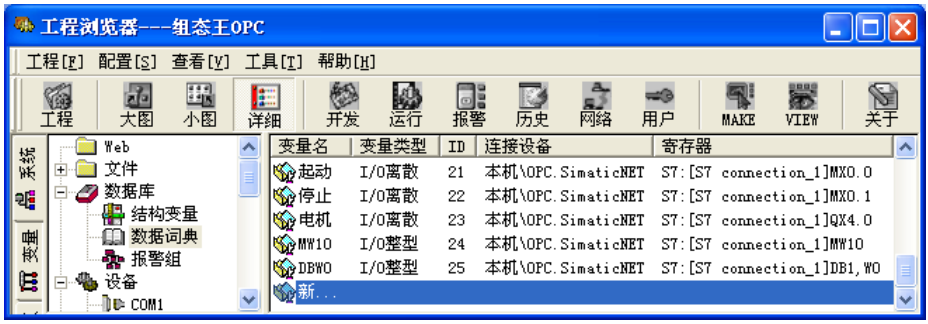


图 13-28　数据词典

2．定义 OPC 服务器中的变量

选中组态王工程浏览器左边窗口的“数据词典”（见图 13-28），双击右边窗口最下面的“新建”。在出现的“定义变量”对话框中（见图 13-29），设置变量的名称为“起动”，变量类型为“I/O 离散”。

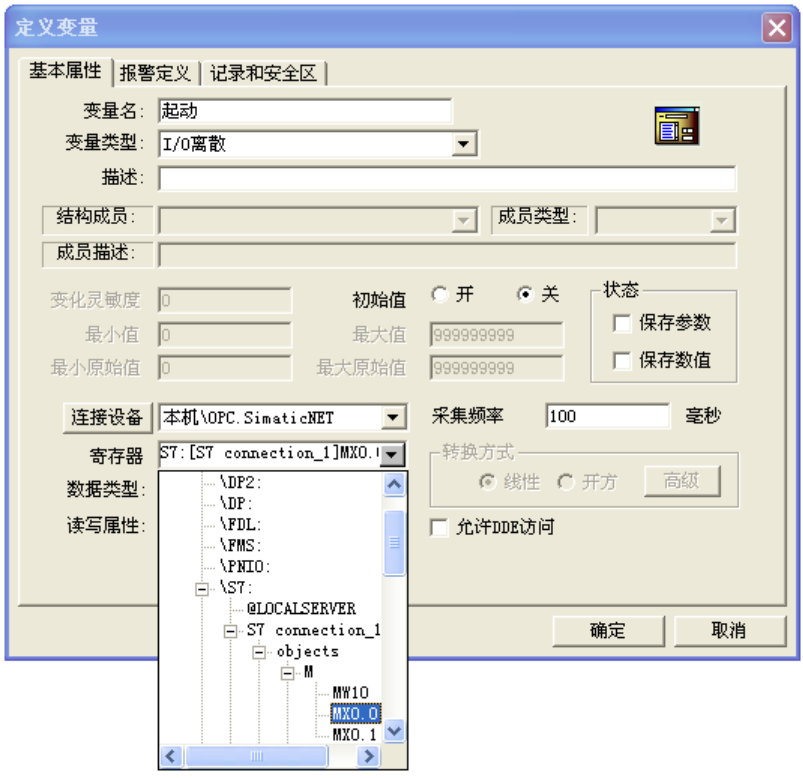


图 13-29　定义变量 M0.0

用“连接设备”选择框选中“本机\OPC.SimatcNET”。点击“寄存器”选择框右边的按钮，在打开的列表中，选中在 OPC Scout 定义的条目“MX0.0”。此外设置数据类型为 Bit，读写属性为“读写”，采集频率为 100ms。点击“确定”按钮，在数据词典中将会出现新生成的变量。用同样的方法生成变量“停止”（M0.1）和“电机”（Q4.0）。

双击“新建”图标，生成名为“DBW0”的变量（见图 13-30），变量类型为“I/O 整数”，数据类型为 USHORT（无符号 16 位字）。点击“寄存器”选择框右边的按钮，在打开的列表中，选中用 OPC Scout 定义的条目“DB1,W0”（DB1.DBW0）。其他的设置与变量“起动”相同，点击“确定”按钮结束设置。用类似的方法生成变量“MW10”。

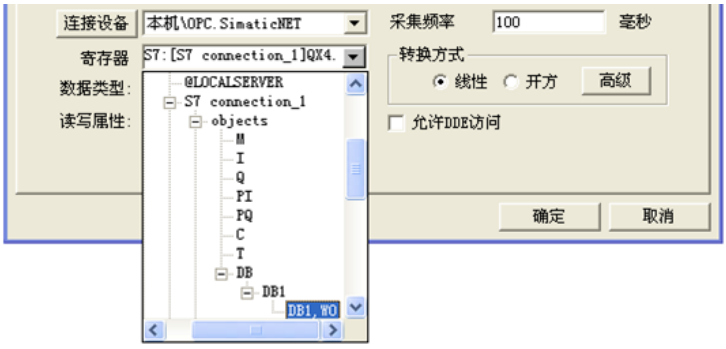


图 13-30　定义变量 DB1.DBW0

3．组态显示和修改字变量的字符串

选中工程浏览器左边窗口中的“画面”，双击右边窗口中的“新建”，在出现的对话框中设置画面的名称为“初始”。点击“确定”按钮，打开开发系统，将画面的背景色改为白色。图 13-31 是组态好的画面上的元件。两个字符串“#####”分别用来显示和修改 MW10 和 DB1.DBW0。

图 13-31　组态王的画面（局部）

双击 MW10 右边的字符串，点击出现的“动画连接”对话框中的“模拟值输入”按钮（见图 13-32），打开“模拟值输入连接”对话框。点击带问号的按钮，设置变量的名称。此外还需要设置变量的取值范围。点击“模拟值输出”按钮，设置需要显示的变量为 MW10。用同样的方法设置画面上 DB1.DBW0 右边的字符串的动画连接属性。

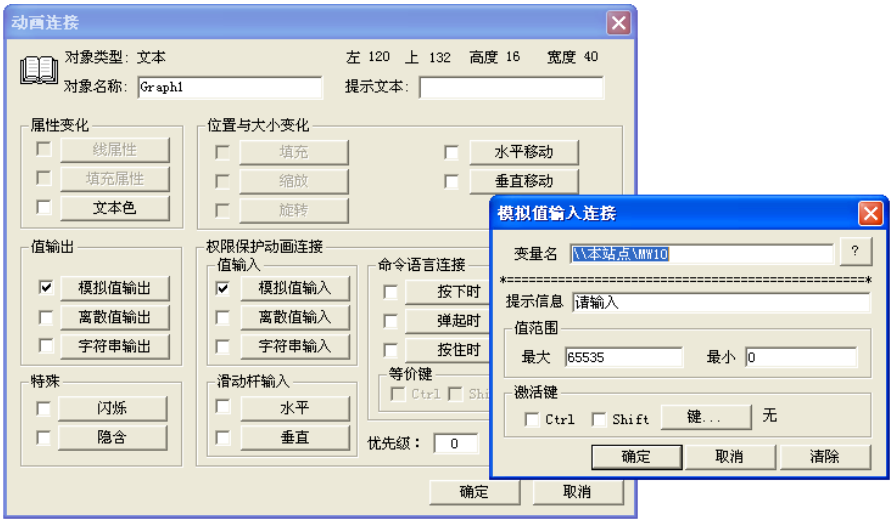


图 13-32　组态显示和修改 MW10 的动画连接

4．组态指示灯

画面上的指示灯用来显示变量“电机”（Q4.0）的状态。执行菜单命令“图库”→“打开图库”，选中打开的图库管理器左边窗口中的“指示灯”，双击右边窗口中的某个指示灯，出现一个随鼠标移动的“Γ”形光标，用鼠标左键点击画面，选中的指示灯被放置到画面上，用鼠标调节指示灯的大小和位置。

双击指示灯，打开“指示灯向导”对话框（见图 13-33）。设置指示灯连接的变量为“电机”。点击“报警色”按钮，将报警色（变量为 0 状态时的颜色）由红色改为深绿色。“正常色”为浅绿色。

5．组态按钮

选中工具箱中的按钮，用鼠标在画面上生成一个按钮。用鼠标右键点击该按钮，执行出现的快捷菜单中的“字符串替换”命令，将按钮上的文本改为“起动”。

双击刚生成的按钮，在打开的“动画连接”对话框中，点击“按下时”按钮（见图 13-32），在出现的“命令语言”对话框中（见图 13-34），输入命令“起动=1;”，点击“确认”按钮返回“动画连接”对话框。用同样的方法，设置在按钮弹起时执行命令“起动=0;”这样的按钮属于点动按钮。按钮按下时变量“起动”为 1 状态，放开时为 0 状态。用同样的方法生成和设置“停止”按钮。

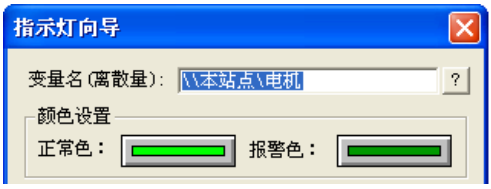


图 13-33　指示灯向导

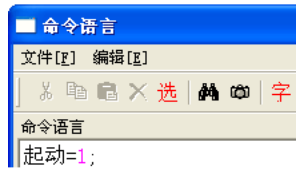


图 13-34　命令语言对话框

6．验证 PLC 与组态王的 OPC 通信

用 MPI 通信电缆连接 CP 5613 卡与 CPU 315 的 MPI 接口，令 PLC 运行在 RUN 模式，组态软件也进入运行模式。这样组态王可以通过内嵌的 OPC 客户端与 SIMATIC NET 软件中的 OPC 服务器交换数据，从而可以监控处于低层的 S7-300。具体方法如下：

执行组态王开发系统中的菜单命令“文件”→“全部存”，再执行菜单命令“文件”→“切换到 View”，打开运行系统，进入演示方式。在运行系统中执行命令“画面”→“打开”，在出现的“打开画面”对话框中，选中要打开的初始画面。点击“确定”按钮，出现处于运行状态的画面（见图 13-35）。可以看到，两个字符串分别动态地显示 CPU 315 中变量 MW10 和 DB1.DBW0 不断增加的值。

图 13-35　运行状态的画面

双击某个动态变化的字符串，用出现的对话框修改变量的值。点击“确定”按钮，可以看到修改的效果，PLC 和 OPC Scout 中该变量的值随之而变。

点击“起动”按钮和“停止”按钮，PLC 中变量 M0.0 和 M0.1 的值随之而变。由于 PLC 的 OB1 中程序的作用（见图 13-14），使 Q4.0 的状态变化，画面中指示灯的状态随之而变。

同时打开组态王的运行系统和 OPC Scout，可以看到画面上的变量和 OPC Scout 中的变量

基本上同步变化。用 OPC Scout 修改 MW10 或 DB1.DBW0 的值，在画面上可以看到修改的结果。即使关闭 OPC Scout，也不会影响组态软件与 PLC 之间的 OPC 通信。

7．组态时常见问题的原因

在执行 PC 站组态下载的过程中，可能会出现“Cannot reach station manager”的错误信息提示，可能的原因如下：

1）在 Configuration Console 中，S7 Online 的访问点（Access Points）没有设为“PC Internal（Local）”。

2）站组态编辑器中虚拟 PC 站的名称与 STEP 7 中 PC 站的名称不符。

3）站组态编辑器中虚拟 PC 站组件（CP 卡和 OPC 服务器）的插槽号与 STEP 7 硬件组态中的插槽号不同。

13.4 基于以太网的 OPC 服务器与 PLC 的通信

13.4.1 组态 PC 站

本例使用计算机主板集成的普通 10M/100M 以太网卡与 PLC 通信。

双击 Windows 的任务栏上的图标。打开站组态编辑器（见图 13-2）。选中编辑器中的 1 号插槽（Index1），点击“Add”按钮，在出现的“Add Component”（添加组件）对话框中，选中“OPC Server”（OPC 服务器），点击“OK”按钮完成添加操作。

用同样的方法，在 3 号槽添加对应于普通网卡的“IE General”（见图 13-36 中下面的图）。作者使用的网卡的名称为“NVIDIA nForce Networking Controller”。

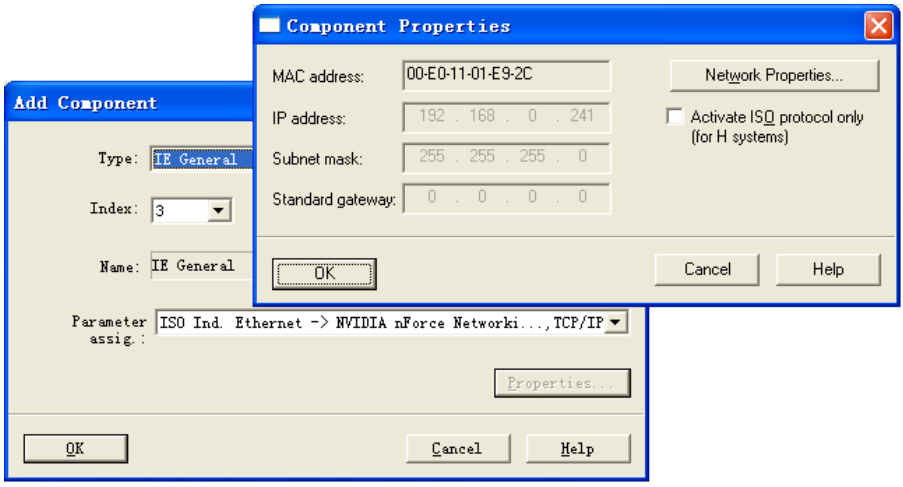


图 13-36 添加普通网卡

插入 IE General 后，弹出其属性对话框（见图 13-36 上面的图），点击“Network Properties”（网络属性）按钮，打开计算机的“网络连接”对话框（见图 10-13），可以对网卡的参数（例如 IP 地址和子网掩码等）进行组态，具体的操作方法见 10.2 节。

点击图 13-2 中的“Station Name”按钮，用出现的“Station name”对话框设置 PC 站的名称为“IE_OPC”。点击“OK”按钮，完成 PC 站的硬件组态。

点击 Windows 下面的工具栏左边的“开始”按钮，执行菜单命令“开始”→“所有程序”→“SIMATIC”→“SIMATIC NET”→“Configuration console”，打开组态控制台。选中左边窗口的文件夹“\Modules\NVIDIA nForce Network”中的“General”图标（见图 13-37），可以看到，普通网卡已从默认的 PG（编程器）模式切换到 Configuration mode（组态模式）。

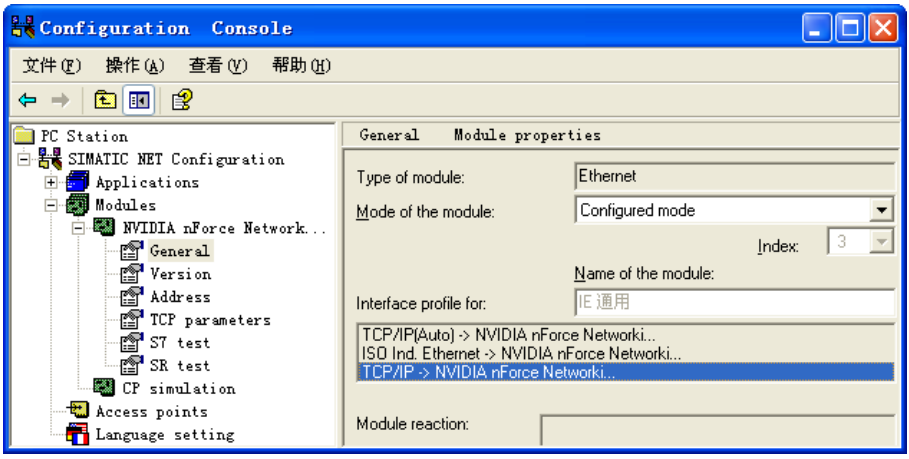


图 13-37　组态控制台

选中左边窗口的“Access Points”（访问点）图标，双击右边窗口的“S7ONLINE”，选中出现的“S7ONLINE Properties”对话框中的“PC internal（local）”，点击“OK”按钮。

选中左边窗口的“Address”图标，右边窗口是网卡的地址信息，可以看到网卡的 MAC 地址、IP 地址和子网掩码。

13.4.2　在 STEP 7 中组态 PC 站和 PLC

1．组态 PC 站点

打开 SIMATIC 管理器，执行菜单命令“文件”→“新建”，创建一个名为“IE_OPC”的新项目。用右键点击项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC PC 站点”，插入一个 PC 站。PC 站的名称必须与 Station Configuration Editor 中命名的站的名称相同，将 PC 站默认的名称“SIMATIC PC 站点（1）”改为“IE_OPC”（见图 13-38）。选中它以后，双击右边窗口中的“组态”，打开 HW Config（硬件组态）窗口。

图 13-38　SIMATIC 管理器

在 HW Config 中，打开左边的硬件目录窗口的文件夹“\SIMATIC PC Station\CP Industrial Ethernet\IE General”，将其中的 SW V6.2 SP1 插入 3 号槽。将文件夹“\SIMATIC PC Station\User Application\OPC Server”中的 SW V6.3 插入 1 号槽。

2．组态通用网卡的网络参数

双击机架第 3 行的“IE 通用”，打开其属性对话框（见图 13-39）。点击“属性”按钮，打开以太网接口属性对话框。点击“新建”按钮，新建一条以太网。网卡的 IP 地址和子网掩码应与图 10-13“本地连接”中组态的相同。

图 13-39　组态通用网卡

完成 PC 站组件设置后，点击工具栏上的（保存和编译）按钮，保存组态信息。

3．组态 S7-300 站

在 SIMATIC 管理器中，用右键点击项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”，插入一个 S7-300 站。选中它以后，双击右边窗口的“硬件”图标，打开 HW Config 窗口。将导轨拖放到左边的窗口，在机架中插入 CPU 313C-2DP、电源模块、I/O 模块和 CP 343-1 IT。双击 CP 的属性对话框中的“属性”按钮，在打开的“属性 – Ethernet 接口”对话框中，采用默认的 IP 地址和子网掩码（见图 13-40）。选中“子网”列表框中的“Ethernet（1）”，将 CP 连接到以太网上。

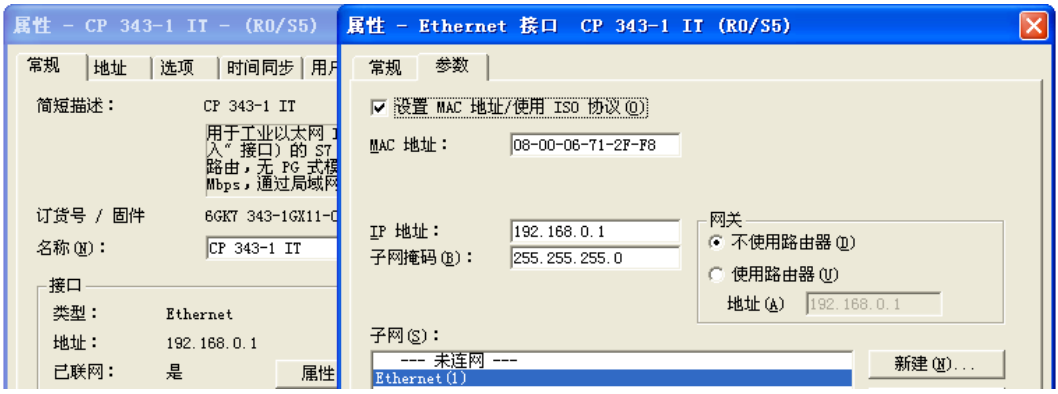


图 13-40　CP 343-1 IT 的以太网接口属性对话框

在 SIMATIC 管理器中生成组织块 OB35 和数据块 DB 1，在 DB 1 中生成一个数组。在 OB1 中编写梯形图程序（见图 13-14），用组态软件画面上的起动按钮 M0.0 和停止按钮 M0.1 控制电动机 Q4.0。

在 OB35 中编写程序，每 100ms 分别将 MW10 和 DB1.DBW0 的值加 1 和加 2。

4．建立连接

编译成功后，点击（组态网络）按钮，打开 NetPro（网络组态）窗口（见图 13-41）。可以看到 CPU 315-2DP 和 PC 站已经连接到以太网上了。

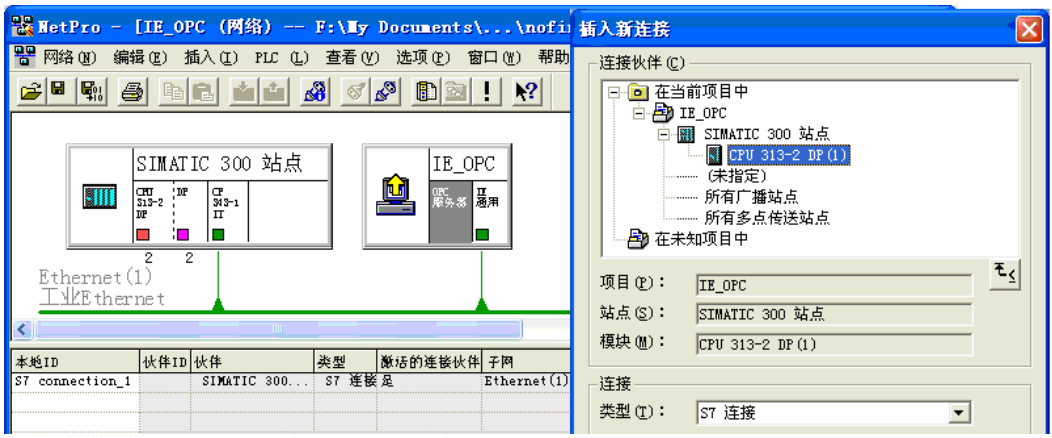


图 13-41　组态 S7 连接

用鼠标选中 PC 站中的 OPC 服务器，双击下面的连接表的第一行，在自动打开的“插入新连接”对话框中，默认的连接对象为 CPU 313C-2DP，默认的连接类型为 S7 连接。点击“确定”按钮，生成一个单向 S7 连接。最后点击（编译并保存）按钮，保存组态的结果。

5．下载组态数据

完成上述组态任务后，点击工具栏上的（下载）按钮，用 MPI 接口将组态信息下载到 CPU 和 PC 站点。选中 SIMATIC 管理器左边窗口的“块”文件夹，下载程序块和组态信息。将 PG/PC 接口切换到 TCP/IP 协议，用交叉连接的 RJ 45 电缆连接计算机网卡和 CP 343-1 IT 的以太网接口。

下载完成后，打开 Station Configuration Editor 窗口，检查组件状态。OPC 服务器插槽的 Conn 列出现的连接图标说明连接被激活（参见图 13-17）。

13.4.3　在 OPC Scout 中生成 OPC 的条目

1．生成 OPC 的组

在 Windows 的桌面上，执行菜单命令“开始”→“所有程序”→“SIMAITC”→“SIMATIC Net”→“OPC Scout”，打开 OPC Scout（见图 13-18）。

双击图中的“New group”，在出现的“ADD Group”（添加组）对话框中，输入组的名称“IE_OPC”，点击“OK”按钮确认。

2．定义 OPC 的条目

双击新建的组 IE_OPC，弹出的“OPC-Navigator”对话框（见图 13-42）显示出所有的连接协议。双击“S7”图标，显示出用 NetPro 组态 PC 站时建立的名为“S7 connection_1”的连接。子文件夹“\objects”内是 PLC 各数据区组成的对象树。

双击对象树中的 New Definition，生成条目 M0.0（见图 13-20）、M0.1、MW10 和 DB1.DBW0。点击“-->”按钮，将图 13-42 中间窗口选中的的条目移到右边窗口。

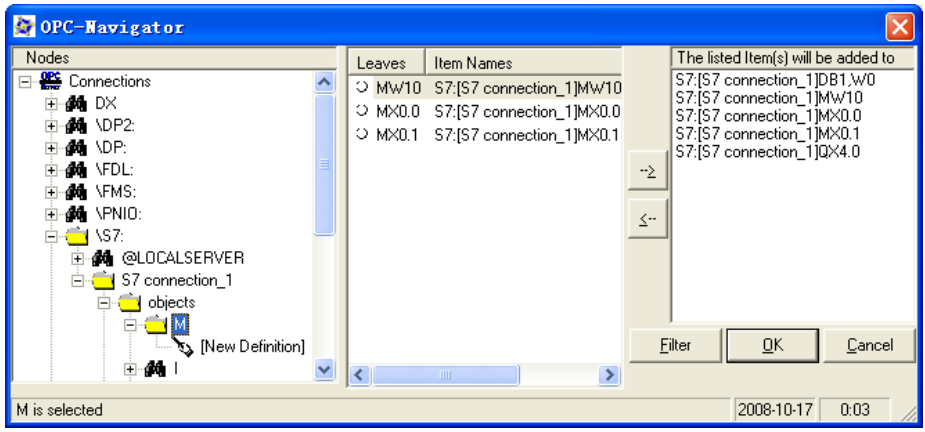


图 13-42　OPC 浏览器

点击图 13-42 中的“OK”按钮，右边窗口中的条目被连接到 OPC 服务器（见图 13-43）。点击 OPC Scout 的工具栏上发光的灯泡图标，激活当前的组，左边窗口的 IE_OPC 图标变为绿色。如果条目的“Quality”为“good”，表示已经建立 OPC 服务器与 PLC 的 S7 连接，可以对条目进行读写操作。

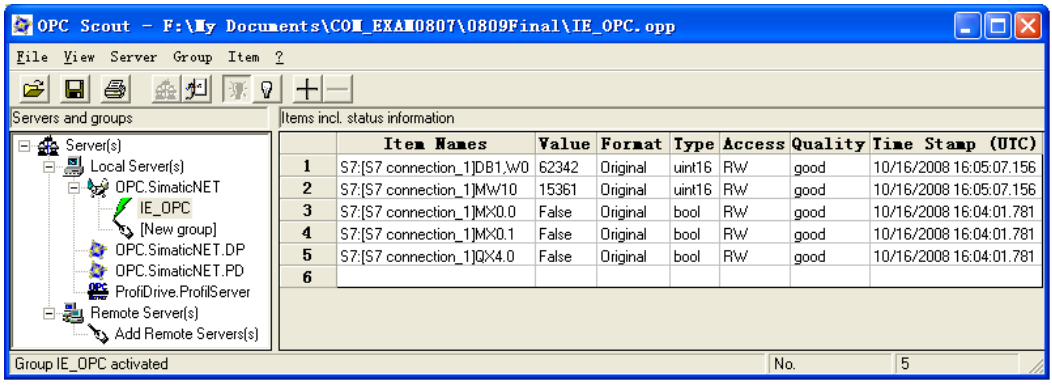


图 13-43　运行中的 OPC Scout

将 CPU 313C-2DP 切换到 RUN 模式，每 100ms 执行一次的 OB35 中的程序将 MW10 和 DB1.DBW0 加 1 和加 2，在图 13-43 中可以看到它们的值在不断地动态变化。CP 343-1 IT 的“RX/TX”LED 在不停地快速闪动。双击某个条目（例如 MW10）的“Value”列，可以在出现的对话框中，对选中的条目进行写操作。

完成上述操作后，在组态软件中就可以使用定义的 OPC 条目了。

13.5 练习题

1. 什么是 OPC？OPC 有什么作用？
2. 什么情况下需要使用软件 SIMATIC NET？
3. 怎样组态带 OPC 服务器的 PC 站？
4. 为了实现 OPC 服务，怎样在 STEP 7 中组态 PLC 和 PC 站点？
5. 怎样用 OPC Scout 生成 OPC 的条目？
6. 怎样验证 OPC 与 PLC 的连接？
7. 怎样用 OPC 实现 PLC 与组态软件的通信？
8. 怎样组态基于以太网的 OPC 服务器与 PLC 的 S7 通信？

第 14 章　MPI 网络通信

14.1　MPI 网络简介

1．MPI 网络

MPI 是多点接口（Multi Point Interface）的缩写，每个 SIMATIC CPU 的第一个通信接口都集成了 MPI 通信协议。

MPI 的物理层是 RS-485，最大传输速率为 12 Mbit/s，默认的传输速率为 187.5 kbit/s。

PLC 通过 MPI 能同时连接运行 STEP 7 的编程器/计算机（PG/PC）、人机界面（HMI）、SIMATIC S7、M7 和 C7（见图 14-1）。每个 CPU 可以使用的 MPI 连接总数与 CPU 的型号有关，为 6～64 个。例如，CPU 312 为 6 个，CPU 417 为 64 个。

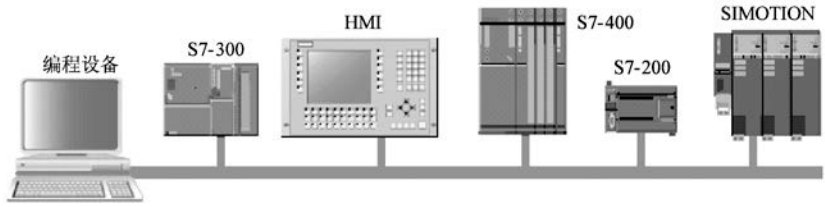


图 14-1　MPI 网络

两个站点之间没有其他站点时（见图 14-2），MPI 站到中继器的最大距离为 50m，中继器之间的最大距离为 1000m。最多可以加 10 个中继器，两个站之间的最大距离为 9100m。

如果在两个中继器之间有 MPI 站点，则每个中继器只能扩展 50m。

图 14-2　带中继器的 MPI 网络

MPI 网络使用 PROFIBUS 总线连接器和 PROFIBUS 总线电缆。位于网络终端的站，应将其总线连接器上的终端电阻开关扳到 On 位置。网络中间的站应将总线连接器上的终端电阻开关扳到 Off 位置。

为了实现计算机与 PLC 的通信，计算机应配置一块 MPI 卡，或使用 PC/MPI、USB/MPI 适配器。应为每个 MPI 节点设置 MPI 地址（0～126），编程设备、人机界面和 CPU 的默认地址分别为 0、1、2。MPI 网络最多可以连接 125 个站。

在 S7-300 中，MPI 总线在 PLC 中与 K 总线（通信总线）连接在一起，S7-300 机架上 K 总线的每一个节点（包括功能模块 FM 和通信处理器 CP）也是 MPI 的一个节点，有自己的 MPI 地址。在 S7-400 中，MPI（187.5 kbit/s）通信模式被转换为内部 10.5 Mbit/s 的 K 总线。S7-400 只有 CPU 有 MPI 地址，其他智能模块没有 MPI 地址。

2．MPI 的通信服务

MPI 网络可以提供下列通信功能：

1）PG/OP（编程器/操作面板）通信功能。

2）小数据量的全局数据（简称为 GD）通信，不需要编程。

3）最多 76B 的小数据量 S7 基本通信。

4）内置的、经济的 S7 通信。

3．S7-300/400 的通信接口

大多数 S7-300 CPU 集成的第一个通信接口是 MPI 接口，某些 S7-300 CPU 和 S7-400 CPU 集成的第一个通信接口可以设置为 MPI 接口或 DP 接口。型号中有“2DP”的 S7-300/400 的 CPU 集成的第二个通信口是 DP 接口，没有 S7 基本通信和全局数据通信功能。型号中有“2PN/DP”的 CPU 集成有一个 MPI/DP 接口和一个 PROFINET 以太网接口。

14.2 全局数据通信

通过全局数据（Global data，GD）通信，同一个 MPI 子网中最多 15 台 S7-300/400 和 C7 之间可以周期性地相互交换少量的数据。每个 CPU 都可以通过全局数据通信访问其他 CPU 的过程输入、过程输出、存储器标志位（M）、定时器、计数器和数据块中的数据。

全局数据通信具有下列特点：

1）硬件成本低。使用 CPU 集成的 MPI 接口，不需要增加通信处理器（CP），就可以实现多 CPU 之间的通信，对 CPU 也没有特殊的要求，因此这是一种经济而有效的通信方式。

2）使用简单方便。只需要在 STEP 7 的网络组态工具 NetPro 中，用全局数据表对全局数据通信组态，设置各 CPU 之间用于交换数据的地址区和通信速率等参数，运行时 CPU 的操作系统就可以实现周期性的全局数据交换，不需要用户对全局数据通信编程。

3）可以实现事件驱动的全局数据通信。即只是在事件发生时（例如在某个数字量信号的上升沿），才调用系统功能（SFC）来发送数据。这样可以有效地减少通信网络的负载。

4）传输的数据量较少。S7-400 之间每个全局数据包最多 54B，S7-300 最多传送 22B。因此适合 CPU 之间传输少量数据时使用。

5）全局数据通信采用广播方式来传输数据，数据被接收后不返回确认信息，不能保证通信数据的完整性和准确性。可以设置状态双字来监控通信的状态，获取错误信息。如果要求进行可靠的数据交换，应使用 S7 通信或其他通信服务。

14.2.1 硬件与网络组态

1．生成一个 STEP 7 项目

打开 SIMATIC 管理器，新建一个项目，项目名称为 MPI_GD_1，全局数据通信的项目在随书光盘的文件夹“\Project\MPI_GD”中。

选中 SIMATIC 管理器中生成的“SIMATIC 400 站点”，双击右边窗口的“硬件”图标，打开 HW Config，将电源模块“拖放”到机架的 1 号槽，CPU 413-2DP 插入 4 号槽，从 8 号槽开始添加 I/O 模块。

点击工具栏上的按钮，保存和编译组态信息，组态信息被自动保存到系统数据中。可以在 HW Config 中用按钮下载组态信息，也可以在 SIMATIC 管理器中，下载编译成功后“块”文件夹中生成的系统数据。

2．组态 S7-300 站点

用鼠标右键点击 SIMATIC 管理器左边窗口最上面的项目图标，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”。选中左边窗口中出现的“SIMATIC 300（1）”图标，用鼠标左键双击右边窗口中的“硬件”图标，打开 HW Config。

将硬件目录窗口的文件夹“\SIMATIC 300\RACK-300”中的导轨（Rail）拖放到硬件组态窗口，电源模块“拖放”到 1 号槽，CPU 315-2DP 模块插入 2 号槽。因为没有接口模块，3 号槽空着。从 4 号槽开始添加 I/O 模块。最后点击按钮，编译并保存组态信息。

3．网络组态

关闭 HW Config，点击 SIMATIC 管理器的工具栏上的按钮，打开网络组态工具 NetPro，可以看到一条自动生成的标有 MPI（1）的网络，和没有与网络相连的两个站，图 14-3 是已经连接好的 MPI 网络。

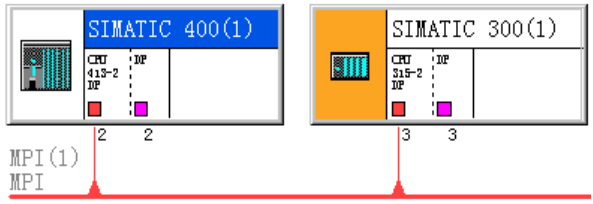


图 14-3　NetPro 中已连接好的 MPI 网络

双击某个站的 CPU 方框中的小红方块，打开 MPI 接口属性对话框（见图 14-4），用鼠标选中“参数”选项卡的“子网”列表中的“MPI（1）”，该行的背景变为深蓝色，点击“确定”按钮，CPU 被连接到 MPI（1）子网上。选中“未连网”后点击“确定”按钮，将断开 CPU 与 MPI（1）子网的连接。点击“确定”按钮返回 NetPro，可以看到该 CPU 是否连接到 MPI 网络。

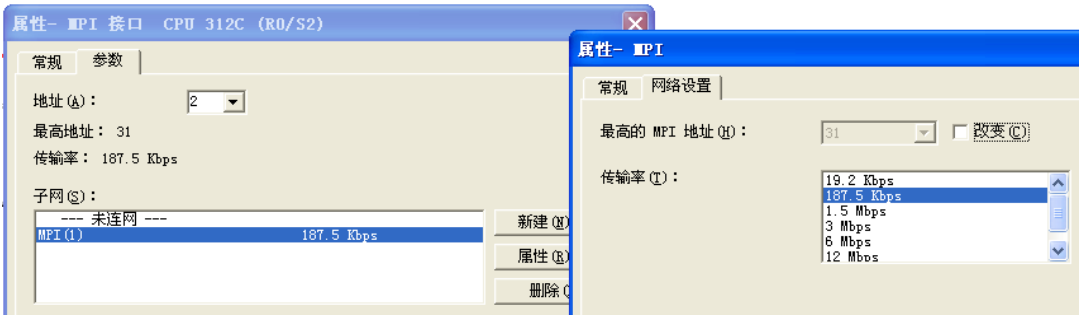


图 14-4　MPI 接口属性对话框

也可以在 NetPro 中，将图 14-3 的 CPU 方框中代表 MPI 接口的小红方块“拖放”到 MPI 网络上，该站便被连接到网络上了，这是一种方便快捷的连接方法。也可以用“拖放”连接点的方法，断开 CPU 与 MPI 网络之间的连接。

可以在“参数”选项卡设置 MPI 站地址，各站的 MPI 地址应互不重叠。点击图 14-4 中的“新建”按钮，可以生成一条新的子网。点击“删除”按钮，可以删除选中的“子网”列表框中的子网。

选中“子网”列表框中的 MPI 网络后，点击“属性”按钮，在打开的“属性 –MPI”对话框中（见图 14-4），可以设置选中的子网的属性，例如在“常规”选项卡中修改子网的名称和编号，在“网络设置”选项卡中设置子网的传输速率，默认的传输速率为 187.5 kbit/s。点击复选框“改变”，小方框中出现“√”后，可以通过设置最高站地址来优化 MPI 子网。可以采用 STEP 7 指定的最高 MPI 地址的默认值。

除了用 NetPro 组态 MPI 网络之外，也可以在 HW Config 中，双击机架中 CPU 所在的行，点击打开的 CPU 属性对话框的“常规”选项卡（见图 14-5）中的“属性”按钮，在打开的 MPI 接口属性对话框中，设置 MPI 接口和 MPI 网络的参数。

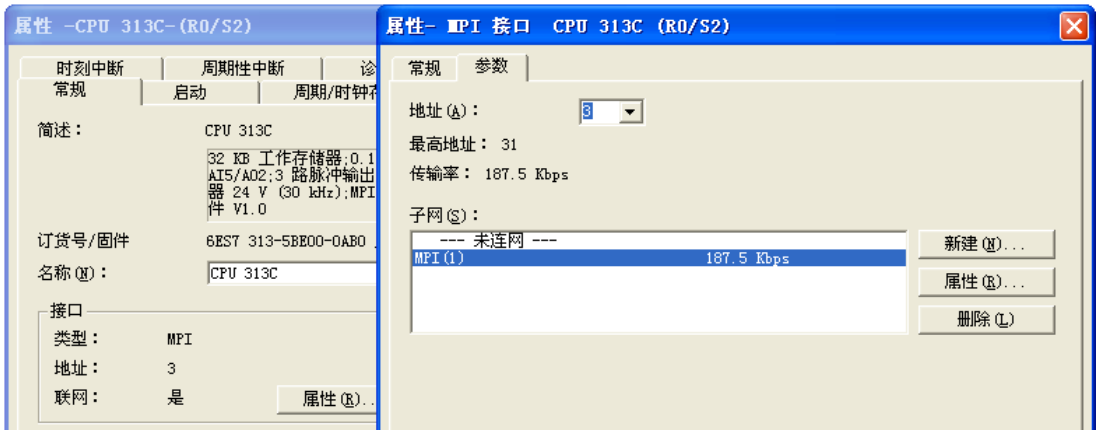


图 14-5　MPI 接口属性对话框

14.2.2　全局数据通信组态

通过全局数据（Global data，GD）通信，可以实现同一个 MPI 子网中的 S7-300/400 和 C7 之间的循环数据交换，传送少量的数据。本节将全局数据通信简称为 GD。

1．全局数据的结构

（1）全局数据环

参与收发全局数据包的 CPU 组成了全局数据环（GD Circle）。CPU 可以向同一个环内的其他 CPU 发送数据或接收数据。在一个 MPI 网络中，最多可以建立 16 个 GD 环。每个 GD 环最多允许 15 个 CPU 参与全局数据交换。

（2）全局数据包

每个全局数据占全局数据表中的一行。同一个全局数据环中，具有相同的发送站和接收站的全局数据的字节数之和如果没有超出允许值，在编译时自动组成一个全局数据包（GD

Packet)。GD 包和 GD 包中的数据均有编号，例如，GD1.2.3 是 1 号 GD 环的 2 号 GD 包中的 3 号数据。

（3）CPU 的全局数据通信功能

S7-300 CPU 可以发送和接收的 GD 包的个数(4 个或 8 个)与 CPU 的型号有关，有 S7-300 参与的每个 GD 包最多 22B 数据。S7-400 CPU 可以发送 8 个或 16 个 GD 包，接收 16 个或 32 个 GD 包，S7-400 之间每个 GD 包最多 54B 数据。S7-400 CPU 具有对全局数据交换的控制功能，支持事件驱动的数据传送方式。

2．生成和填写 GD 表

用鼠标右键点击 NetPro 中的 MPI 网络线，执行弹出的快捷菜单中的“定义全局数据”命令。在出现的 GD 表对话框（见图 14-6）中，对全局数据通信进行组态。

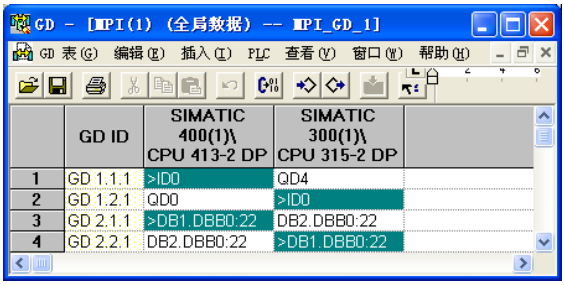


图 14-6　全局数据表

双击“GD ID”（GD 标识符）右边的灰色单元，在出现的“选择 CPU”对话框左边的窗口中（见图 14-7），打开 SIMATIC 400 站点，双击其中的“CPU 413-2DP”图标，自动关闭“选择 CPU”对话框，CPU 413-2DP 站点出现在全局数据表最上面一行指定的方格中。用同样的方法，在最上面一行生成 CPU 315-2DP 站点。将光标放在表头中两列之间的边界线上，出现水平方向的双向箭头后，按住鼠标左键左右拖动，可以调整列的宽度。

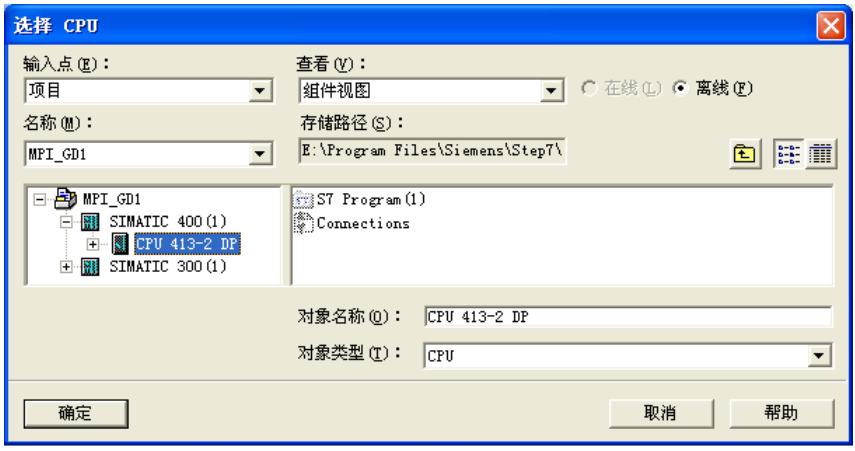


图 14-7　选择 CPU 对话框

在表头下面的第一行生成一个全局数据，将 CPU 413-2DP 的 ID0 发送到 CPU 315-2DP 的 QD4。GD ID 列的 GD 标识符是编译后生成的。

首先用鼠标右键点击 CPU 413-2DP 下面的单元格，执行出现的快捷菜单中的“发送器”命令，该方格变为深色，同时在单元的左端出现符号“>”，表示在该行中 CPU 413-2DP 为发送站，在该单元中输入要发送的全局数据的地址 ID0。只能输入绝对地址，不能输入符号地址，包含定时器和计数器地址的单元只能作为发送方。在每一行中应定义一个并且只能有一个 CPU 作为数据的发送方，可以有一个或多个站接收。同一行中各个单元接收或发送的字节数应相同。

也可以用下面的方法来设置发送地址：选中表格中的某个单元，点击工具栏上的  按钮，该单元变为深色，同时在单元的左端出现符号“>”，然后在该单元输入发送区的地址。用左键选中 CPU 315-2DP 下面的单元，直接输入 QD4，该单元的背景为白色，表示在该行中 CPU 315-2DP 是接收站。

用同样的方法，在表格的第 2 行组态，将 CPU 315-2DP 的 ID0 发送给 CPU 413-2DP 的 QD0。

选中 SIMATIC 管理器左边 SIMATIC 400 站点的“块”文件夹，用鼠标右键点击右边窗口的空白处，执行出现的快捷菜单中的命令，生成共享数据块 DB 1 和 DB 2。为了定义数据块的大小，打开数据块，删除自动生成的临时占位符变量，生成一个 22B 的数组。

用同样的方法，在 CPU 315-2DP 的“块”文件夹中生成共享数据块 DB 1 和 DB 2，用数组定义它们的大小。

做好上述准备工作后，在全局数据表的第 3 行生成一个全局变量，将 SIMATIC 400 站的 DB1.DBB0～DB1.DBB21 的 22B 数据发送到 SIMATIC 300 站的 DB2.DBB0～DB2.DBB21（见图 14–6）。在第 4 行生成一个全局变量，将 SIMATIC 300 站的 DB 1 的 22B 数据发送到 SIMATIC 400 站的 DB 2。

变量的复制因子用来定义连续的数据区的长度，例如，DB1.DBB0:22 表示从 DB 1 的 DBB0 开始的 22B 数据，MW0:11 表示从 MW0 开始的 11 个字。注意必须使用英文的冒号，如果使用中文的冒号将会出错，因为在计算机内部，两种冒号使用不同的编码。

如果全局数据包由若干个连续的数据区组成，一个连续的数据区占用的空间为数据区的字节数加上两个头部说明字节。一个单独的双字占 6B，一个单独的字占 4B，一个单独的字节占 3B，一个单独的位也占 3B。值得注意的是，第一个连续数据区的两个头部说明字节不包括在 22B 之内。例如，一个全局数据包中的 DB2.DBB0:10 和 QW0:5 一共占用 22B。

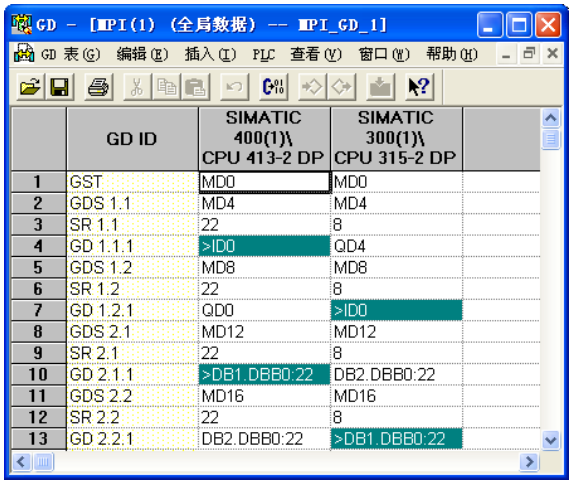
在运行时，发送方 CPU 自动地周期性地将指定的地址区中的数据发送到接收方指定的地址区。例如，图 14–6 第 3 行意味着 CPU 413-2DP 定时地周期性地将 DB 1 中的数据发送到 CPU 315-2DP 的 DB 2。CPU 315-2DP 对它自己的 DB 2 访问，就好像在访问 CPU 413-2DP 的 DB 1 一样。

完成全局数据表的输入后，应点击工具栏上的  按钮，对它进行第一次编译。图 14–6 的“GD ID”列中的 GD 标识符是在编译时自动生成的。编译生成了全局数据环和全局数据包。

3．设置扫描速率和状态双字的地址

扫描速率用来定义 CPU 刷新全局数据的时间间隔。编译后执行菜单命令“查看”→“扫描速率”，每个数据包将增加标有“SR”的行（见图 14–8），用来设置该数据包的扫描速率（1～255）。“SR”的后面是全局数据环和全局数据包的编号。再次执行该命令，将会隐藏扫描速率所在的各行。扫描速率的单位是 CPU 的扫描循环周期，S7-300 默认的扫描速率为 8，S7-400 默认的扫描速率为 22，用户可以修改它们。如果选择 S7-400 的扫描速率为 0，表示是事件驱

动的 GD 数据传输。



	GD ID	SIMATIC 400(1)\ CPU 413-2 DP	SIMATIC 300(1)\ CPU 315-2 DP
1	GST	MD0	MD0
2	GDS 1.1	MD4	MD4
3	SR 1.1	22	8
4	GD 1.1.1	>ID0	QD4
5	GDS 1.2	MD8	MD8
6	SR 1.2	22	8
7	GD 1.2.1	QD0	>ID0
8	GDS 2.1	MD12	MD12
9	SR 2.1	22	8
10	GD 2.1.1	>DB1.DBB0:22	DB2.DBB0:22
11	GDS 2.2	MD16	MD16
12	SR 2.2	22	8
13	GD 2.2.1	DB2.DBB0:22	>DB1.DBB0:22

图 14-8　设置扫描速率与状态双字

发送器的扫描速率为 22，表示 CPU 每隔 22 个扫描周期，在扫描周期检查点发送一次 GD 包。接收器的扫描速率为 8，表示 CPU 每隔 8 个扫描周期，在扫描周期检查点接收 GD 包。更确切地说，在设定的地址区输入已接收的 GD 包。

扫描速率值太小，将会使网络通信超载。建议采用默认的扫描速率，或确保扫描周期与扫描速率的乘积大于 0.5s。对于更高的通信要求，应采用其他通信方式，例如通过 PROFIBUS-DP 进行通信。

可以用 GD 数据传输的状态双字来检查数据是否被正确地传送，编译后执行菜单命令“查看”→“GD 状态”，在出现的 GDS 行中可以给每个数据包指定一个用于状态双字的地址。再次执行该命令，各状态双字所在的行将会消失。

最上面一行的全局状态双字 GST 是各 GDS 行中的状态双字相“或”的结果。状态双字中使用的各位的意义见表 14-1，被置位的位将保持其状态不变，直到它被用户程序复位。

表 14-1　GD 通信的状态双字

位号	说　明	状态位设定者	位号	说　明	状态位设定者
0	发送方地址区长度错误	发送或接收 CPU	5	全局数据包 GD 对象遗漏	接收 CPU
1	发送方找不到存储 GD 的数据块	发送或接收 CPU	6	接收方发送方数据长度不一致	接收 CPU
3	全局数据包在发送方丢失 全局数据包在接收方丢失 全局数据包在链路上丢失	发送 CPU 发送或接收 CPU 接收 CPU	7	接收方地址区长度错误	接收 CPU
			8	接收方找不到存储 GD 的数据块	接收 CPU
			11	发送方重新启动	接收 CPU
4	全局数据包语法错误	接收 CPU	31	接收方接收到新数据	接收 CPU

状态双字使用户程序能及时了解通信的有效性和实时性，增强了系统的故障诊断能力。

设置好扫描速率和状态双字的地址后，应点击工具栏上的按钮，对全局数据表进行第二次编译，使扫描速率和状态双字地址包含在组态数据中。

关闭全局数据表，返回 NetPro，点击工具栏上的按钮，保存和编译组态信息。

4．编写验证通信的程序

为了检验通信是否成功，在组态全局数据表时，用某个站的 ID0 来控制通信伙伴的 QD0（CPU 413-2DP）或 QD4（CPU 313C-2DP）。在运行时用接在输入模块端子上的小开关来改变 ID0 的值，观察通信伙伴对应的输出点的状态是否随之而变。

在 CPU 413-2DP 的初始化组织块 OB100 中，将组态时指定的 DB1 中的字预置为 16#4444，将组态时指定的 DB 2 中的 22B 数据接收区清零。

为了观察周期性数据传输的动态效果，在每 100ms 执行一次的 OB35 中，将数据发送区的第一个字 DB1.DBW0 加 1。

CPU 315-2DP 的程序与 CPU 413-2DP 的基本上相同，只是在 OB100 中，将组态时指定的 DB 1 中的字预置为 16#3333。

5．通信错误组织块 OB87

在使用通信功能块或全局数据（GD）通信进行数据交换时，如果出现下列通信错误，操作系统将调用 OB87：

1）接收全局数据时，检测到错误的帧标识符（ID）。

2）全局数据通信的状态信息数据块不存在或太短。

3）接收到非法的全局数据包编号。

如果没有生成和下载 OB87，CPU 将会切换到 STOP 状态。OB87 的局部变量 OB87_FLT_ID 给出了错误代码。选中 SIMATIC 管理器中的 OB87 之后，按计算机的<F1>键，在出现的 OB87 的在线帮助中，可以查看错误代码的详细信息。

6．下载与运行

第二次编译成功完成后，需要将组态好的信息单独下载到各 CPU。比较方便的是使用计算机上安装的 CP 卡（例如 CP 5611 或 CP 5613），通过 MPI 网络下载和监控通信过程。

首先在 HW Config 中分别下载各 CPU 的组态信息，然后用 PROFIBUS 电缆连接编程用的计算机和两台 PLC 的 MPI 接口，令各台 PLC 均处于 STOP 模式。在 GD 表编辑器中，点击  按钮，在下载对话框中（见图 14-9），如果选中“下载到所有 CPU”，则组态信息将依次自动地下载到所有的 CPU。如果选中“下载到所选 CPU”，则只下载到选中的 CPU。

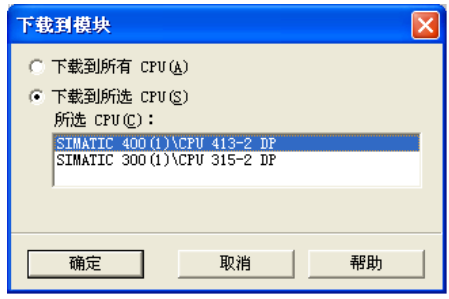


图 14-9　下载对话框

下载完成后将各 CPU 切换到 RUN 模式，各 CPU 之间将开始自动地交换全局数据。在循环周期结束时，发送方的 CPU 发送数据，在循环周期开始时，接收方的 CPU 将接收到的数据传送到组态时指定的地址区。

由图 14-6 可知，CPU 413-2DP 和 CPU 315-2DP 的 ID0 分别控制对方的 QD0 或 QD4，运行时改变某台 PLC 输入点的状态，可以观察对方对应的输出点是否随之而变。

在运行时同时打开两个站的变量表（见图 14-10 和图 14-11），调节它们的大小后，在屏幕上同时显示各变量表中的数据。其中的“状态值”是 STEP 7 读取的 CPU 中的数据。

可以看到，变量中的 DB2.DBW0 的值在不断地增大。

图 14-10　CPU 413-2DP 的变量表

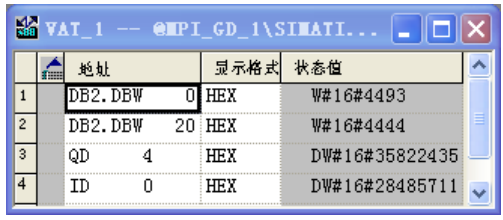


图 14-11　CPU 315-2DP 的变量表

7．两台 S7-300 之间的全局数据通信

随书光盘中的项目 MPI_GD_2 用于两台 CPU 315-2DP 之间的全局数据通信。其全局数据表（见图 14-12）的第 2 行和第 3 行中的全局数据具有相同的发送方和接收方，这两个全局数据的字节总数也没有超过允许的范围（22B）。因此在第一次编译时，它们被指定为全局数据包 GD1.2 中的两个全局数据 GD1.2.1 和 GD1.2.2。

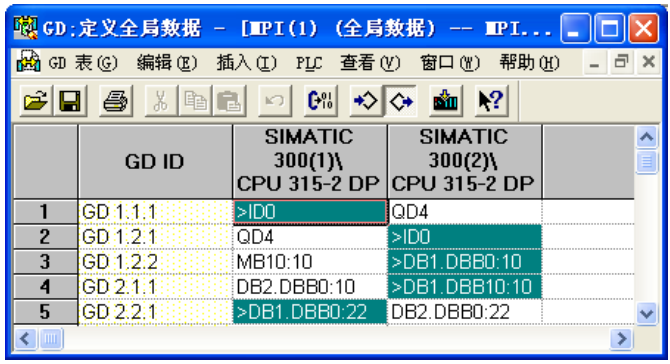


图 14-12　全局数据表

8．两台 S7-400 之间的全局数据通信

随书光盘中的项目 MPI_GD_A 用于两台 CPU 413-2DP 之间的全局数据通信。S7-400 之间的每个全局数据包允许最多传送 54B 的数据（见图 14-13），超出此范围时将会出现错误提示信息。在通信双方的初始化组织块 OB100 中，数据块 OB1 的前 54B 组成的发送数据区分别被预置为 16#4131 和 16#4132。在循环中断组织块 OB35 中，DB1.DBW0 被加 1 后发送给通信伙伴的 DB2.DBW0。

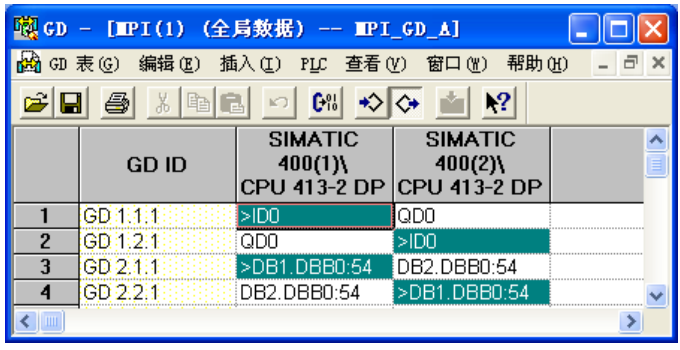


图 14-13　全局数据表

在运行时监控通信双方的变量表（见图 14-14 和图 14-15），可以看到双方接收到的DB2.DBW0 在不断变化，DB 2 中接收到的最后一个字 DB2.DBW52 的值是通信伙伴在 OB100 中预置的值。扳动接在输入模块输入端的小开关，可以看到通信伙伴对应的输出点的状态随之而变。

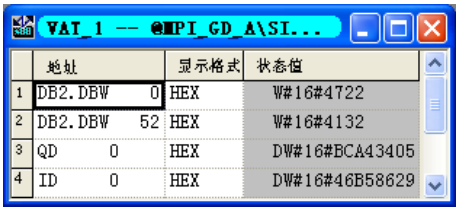


图 14-14　2 号站的变量表

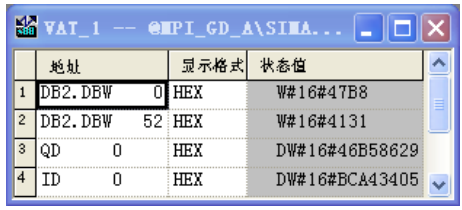


图 14-15　3 号站的变量表

14.2.3　3 个站之间的全局数据通信组态

在 SIMATIC 管理器中新建一个项目，项目名称为 MPI_GD_3（见随书光盘中的同名例程）。

在 SIMATIC 管理器中创建一个 SIMATIC 400 站点，CPU 模块为 CPU 413-2DP，设置它的 MPI 站地址为 2。创建两个 SIMATIC 300 站点，CPU 模块均为 CPU 315-2DP。设置它们的 MPI 站地址分别为 3 和 4，在 NetPro 中，将 3 个站都连接到 MPI 网络上（见图 14-16）。

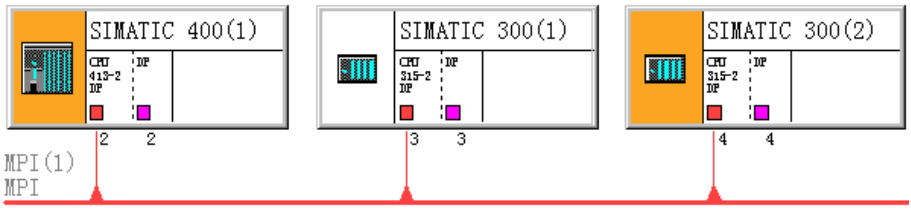


图 14-16　NetPro 中已连接好的 MPI 网络

用鼠标右键分别点击 SIMATIC 管理器（见图 14-17）左边窗口各站点的“块”图标，执行出现的快捷菜单中的命令，生成共享数据块 DB 1 和 DB 2，为了定义数据块的大小，在各数据块中生成一个数组。

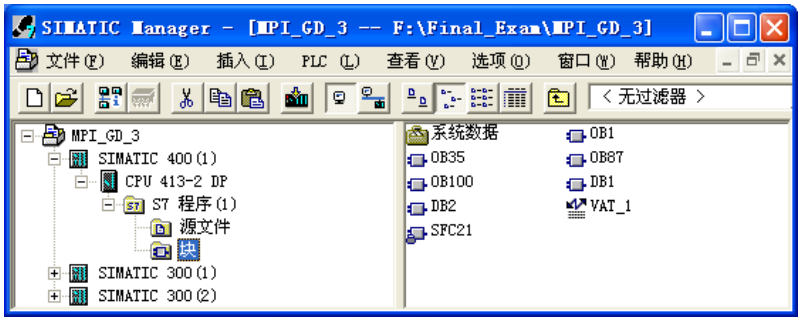
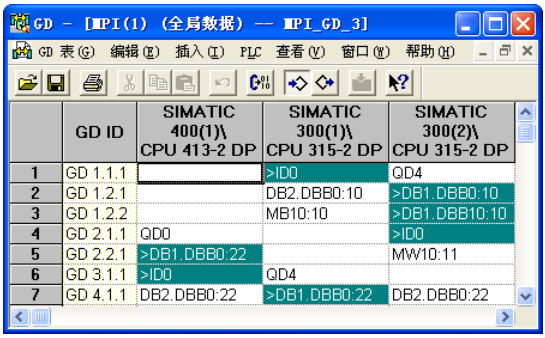


图 14-17　SIMATIC 管理器

用鼠标右键点击 NetPro 中的 MPI 网络线，执行弹出的快捷菜单中的“定义全局数据”命

令。在出现的全局数据表中（见图 14-18）对全局数据通信进行组态。


GD - [MPI(1) (全局数据) -- MPI_GD_3]

GD 表(G) 编辑(E) 插入(I) PLC 查看(V) 窗口(W) 帮助(H)

	GD ID	SIMATIC 400(1)\ CPU 413-2 DP	SIMATIC 300(1)\ CPU 315-2 DP	SIMATIC 300(2)\ CPU 315-2 DP
1	GD 1.1.1		>ID0	QD4
2	GD 1.2.1		DB2.DBB0:10	>DB1.DBB0:10
3	GD 1.2.2		MB10:10	>DB1.DBB10:10
4	GD 2.1.1	QD0		>ID0
5	GD 2.2.1	>DB1.DBB0:22		MW10:11
6	GD 3.1.1	>ID0	QD4	
7	GD 4.1.1	DB2.DBB0:22	>DB1.DBB0:22	DB2.DBB0:22

图 14-18　全局数据表


VAT_1 -- @MPI_GD_...

地址	显示格式	状态值
DB2.DBW 0	HEX	W#16#7E77
DB2.DBW 20	HEX	W#16#3001
ID 0	HEX	DW#16#2D673767
QD 0	HEX	DW#16#BB563528

图 14-19　2 号站的变量表

组态好全局数据后，点击工具栏上的按钮，对全局数据进行编译。从编译后的 GD ID 可以看出，两台 S7-300 组成了 1 号全局数据环，表中的第 2 行和第 3 行全局数据的发送方和接收方相同，组成了全局数据包 GD1.2。CPU 413-2DP 和第 2 块 CPU 315-2DP 组成了 2 号全局数据环。CPU 413-2DP 和第 1 块 CPU 315-2DP 组成了 3 号全局数据环。3 块 CPU 组成了 4 号全局数据环。各 CPU 的 OB100 和 OB35 中的程序见随书光盘中的项目 MPI_GD_3。

将组态信息和程序分别下载到各 CPU，然后用 PROFIBUS 电缆连接编程用的计算机和 3 台 PLC 的 MPI 接口。将各 CPU 切换到 RUN 模式，各 CPU 之间开始自动地交换全局数据。

图 14-19～图 14-21 是系统运行时复制的各站点的变量表。


VAT_1 -- @MPI_GD_2\...

地址	显示格式	状态值
DB2.DBW 0	HEX	W#16#7E64
DB2.DBW 8	HEX	W#16#3002
ID 0	HEX	DW#16#32664232
QD 4	HEX	DW#16#2D673767
MW 10	HEX	W#16#3002
MW 18	HEX	W#16#3002

图 14-20　3 号站的变量表


VAT_1 -- @MPI_GD_2\...

地址	显示格式	状态值
DB2.DBW 0	HEX	W#16#7E79
DB2.DBW 20	HEX	W#16#3001
ID 0	HEX	DW#16#BB563528
QD 4	HEX	DW#16#32664232
MW 10	HEX	W#16#6AFD
MW 30	HEX	W#16#4001

图 14-21　4 号站的变量表

14.2.4　事件驱动的全局数据通信的组态与编程

使用 SFC 60“GD_SEND”和 SFC 61“GD_RCV”，S7-400 之间可以用事件驱动的方式发送和接收 GD 包，实现全局数据通信。应在全局数据表中组态用事件驱动方式发送的 GD 包，并将该 GD 包的扫描速率设置为 0。

打开 SIMATIC 管理器，新建一个名为“MPI_GD_B”的项目（见光盘中的同名例程）。

用鼠标右键点击项目图标，执行出现的快捷菜单中的命令，插入一个 SIMATIC 400 站点（见图 14-22）。用鼠标双击右边窗口中的“硬件”图标，打开硬件组态工具 HW Config，将电源模块、CPU 413-2DP 和信号模块插入机架。

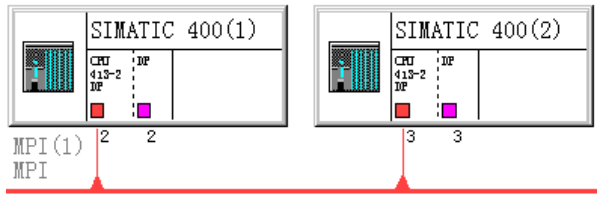


图 14-22　组态好的 MPI 网络

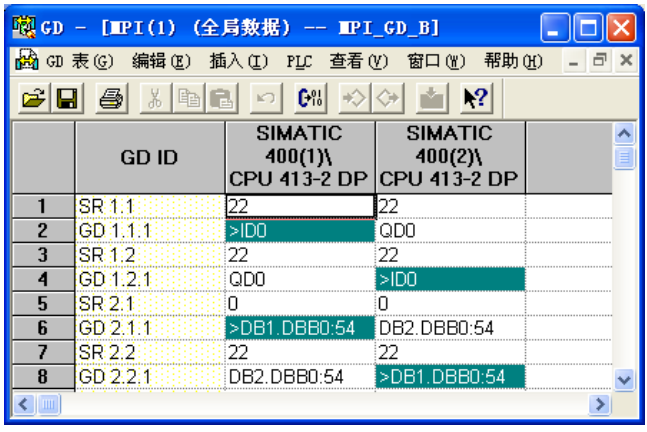
双击 S7-400 的机架中 MPI/DP 所在的行，点击打开的对话框中的“属性”按钮，在打开的 MPI 接口属性对话框中，将 CPU 连接到 MPI 网络上，采用默认的 MPI 地址 2 和默认的网络参数，传输速率为 187.5 kbit/s。

在 SIMATIC 管理器中生成另一个 S7-400 站。在 HW Config 中，将 CPU 413-1 和电源模块、信号模块插入机架，设置 CPU 的 MPI 地址为 3，将它连接到 MPI 网络上。

在 SIMATIC 管理器生成数据块 DB 1、DB 2 和组织块 OB100、OB35、OB87，在各数据块中生成一个数组。各 CPU 的 OB100 和 OB35 中的程序见随书光盘中的项目 MPI_GD_3。

关闭 HW Config，点击 SIMATIC 管理器的工具栏上的按钮，打开网络组态工具 NetPro，可以看到 MPI 网络上的两个站（见图 14-22）。

用鼠标右键点击 NetPro 中的 MPI 网络线，执行弹出的快捷菜单中的“定义全局数据”命令。用出现的 GD 表（见图 14-23）对全局数据通信组态，将双方的 ID0 发送到对方的 QD0。将双方的 DB 1 中的 54B 数据发送到对方的 DB 2。



	GD ID	SIMATIC 400(1)\ CPU 413-2 DP	SIMATIC 400(2)\ CPU 413-2 DP
1	SR 1.1	22	22
2	GD 1.1.1	>ID0	QD0
3	SR 1.2	22	22
4	GD 1.2.1	QD0	>ID0
5	SR 2.1	0	0
6	GD 2.1.1	>DB1.DBB0:54	DB2.DBB0:54
7	SR 2.2	22	22
8	GD 2.2.1	DB2.DBB0:54	>DB1.DBB0:54

图 14-23　全局数据表

为了实现事件驱动的全局数据通信，将全局数据包 GD2.1 的扫描速率（SR2.1）设置为 0。

SFC 60 和 SFC 61 可以在用户程序中的任何一点被调用，全局数据表中设置的扫描速率不受调用 SFC 60 和 SFC 61 的影响。

SFC 60 和 SFC 61 可能被更高优先级的块中断。为了保证全局数据交换的连续性，在调用 SFC 60 之前，调用 SFC 39“DIS_IRT”或 SFC 41“DIS_AIRT”，来禁止或延迟更高优先级的中断和异步错误。执行完 SFC 60 后，调用 SFC 40“EN_IRT”或 SFC 42“EN_AIRT”，允许处理高优先级的中断和异步错误。

下面是 2 号站的 CPU 413-2DP 的 OB1 中的程序，用 SFC 60 发送数据包 GD2.1。

```
程序段 1：延迟处理高中断优先级的中断和异步错误
          CALL   "DIS_AIRT"                    //调用 SFC 41
            RET_VAL      :=MW100               //返回的错误信息
程序段 2：发送全局数据包 GD 2.1
          A      I         0.0
          FP     M         1.0
          JCN    _001                         //不是 I0.0 的上升沿则跳转
          CALL   "GD_SND"                     //调用 SFC 60
            CIRCLE_ID    :=B#16#2             //组态时设置的GD环编号
            BLOCK_ID     :=B#16#1             //组态时设置的GD包编号
            RET_VAL      :=MW102              //返回的错误信息
_001:     NOP    0
程序段 3：允许处理高中断优先级的中断和异步错误
          CALL   "EN_AIRT"                    //调用 SFC 42
            RET_VAL:=MW104                    //返回的错误信息
```

下面是 3 号站的 OB1 调用 SFC 61 接收数据包 GD2.1 的程序：

```
          CALL   "GD_RCV"                     //调用 SFC 61
            CIRCLE_ID    :=B#16#2             //组态时设置的GD环编号
            BLOCK_ID     :=B#16#1             //组态时设置的GD包编号
            RET_VAL      :=MW106              //返回的错误信息
```

将组态信息和程序分别下载到各 CPU，然后用 PROFIBUS 电缆连接编程用的计算机和两台 PLC 的 MPI 接口。将两台 CPU 切换到 RUN 模式，CPU 之间开始自动地交换全局数据。

图 14-24～图 14-26 是系统运行时复制的各站点的变量表。进入 RUN 模式后，通信双方开始周期性地将 ID0 传送给对方的 QD0，3 号站周期性地将 DB 1 中的数据传送到 2 号站的 DB 2。

3 号站的 DB2 的前 54B 数据被 OB100 初始化为 0。在 2 号站的 I0.0 的上升沿调用 SFC 60，将 2 号站的 DB 1 中的数据传送到 3 号站的 DB 2。

图 14-24　2 号站的变量表

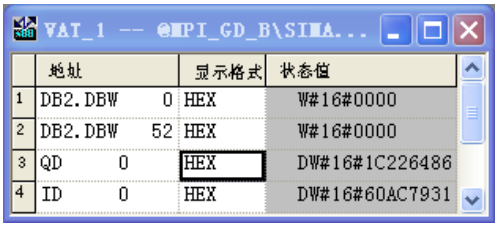


图 14-25　3 号站刚开始进入 RUN 模式的变量表

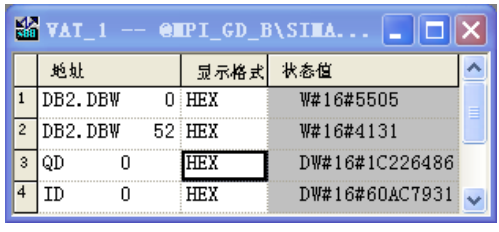


图 14-26　3 号站事件触发数据发送后的变量表

图 14-25 和图 14-26 是 I0.0 第一个上升沿触发数据发送之前和触发发送之后的 3 号站的变量表。2 号站的 DB 1 的前 54B 数据被 OB100 初始化为 16#4131，OB35 将 DB1.DBW0 不断加 1。在每次 I0.0 的上升沿，2 号站的 DB 1 中的数据被发送给 3 号站的 DB 2，3 号站的

DB2.DBW0 的值才会变化。

14.3 S7 基本通信

14.3.1 S7 基本通信概述

S7 基本通信服务通过调用系统功能（SFC）和不需组态的 S7 连接进行数据交换，只能用于 MPI 网络。这些 SFC 可以访问所有 S7 和 C7 PLC 中的数据，发送最多 76B 的数据给 MPI 网络中的 S7 PLC、HMI 或 PC。S7 基本通信的 SFC 集成在 CPU 的操作系统中，并用 SFC 提供用户程序的软件接口。S7 基本通信不能与其他子网中的站进行通信。

1．不需要组态的连接

连接是指两个通信伙伴之间为了执行通信服务建立的逻辑分配，而不是指两个站之间物理媒体（例如电缆）的连接。连接分为需要组态的静态连接和不需要组态的动态连接。

PG（编程器）通信和 S7 基本通信不需要对连接组态，这种连接也称为动态连接。通过通信块 SFC 65～SFC 68 的参数，指定通信伙伴的地址、触发通信的信号，并确定完成数据传输后该连接是继续保持或终止。

在同一时刻，一个不需要组态的连接只能用于一个通信伙伴，与不同的通信伙伴的连接可以一个接一个地建立和终止。完成与某一通信伙伴的数据传输后，可以连接其他通信伙伴，因此可以相继访问的通信伙伴的数量不受连接资源的限制。但是 CPU 同时建立的连接总数不能超过可以同时使用的最大连接个数的限制。由于在通信过程中必须考虑连接的建立和断开，因此降低了网络的数据传输能力。

CPU 进入 STOP 模式时，所有已建立的连接被终止。

2．用于 S7 基本通信的系统功能

S7 基本通信的系统功能（SFC）分为两类：

1）I_GET 与 I_PUT（见表 14–2）用于 CPU 与同一个 S7 站的其他有通信功能的模块之间交换数据。SFC 名称中的“I”表示内部。

2）X_SEND、X_RCV、X_GET 与 X_PUT 用于 S7 CPU 与其他具有通信功能的模块之间交换数据，通信伙伴在同一个 MPI 子网内，但不是在同一个站内。块的名称中的“X”表示外部。

用于 S7 基本通信的 SFC 可以在所有的 S7-300/400 CPU 上运行，S7-300/400 CPU 还可以用 X_GET 和 X_PUT 来读写 S7-200 CPU 中的数据。S7-200 在 S7 基本通信中只能作服务器，因此不需要对 S7-200 组态和编程。

表 14–2　用于 S7 基本通信的 SFC

外部通信块	内部通信块	简　　述
SFC 65 X_SEND SFC 66 X_RCV		将数据安全地传送到通信伙伴，数据传输在通信伙伴的接收功能（X_RCV）接收完数据才结束
SFC 67 X_GET	SFC 72　I_GET	读取通信伙伴的变量，无需通信伙伴编程
SFC 68 X_PUT	SFC 73　I_PUT	将变量写入通信伙伴的存储区，无需通信伙伴编程
SFC 69 X_ABORT	SFC 74　I_ABORT	结束一个已存在和没有传输数据的连接，通信双方释放相应的连接资源

3．S7 基本通信 SFC 的公用参数的说明

（1）输入参数 REQ

REQ（请求激活）是电平触发的控制参数，REQ 为 1 时触发任务。

（2）输入参数 REQ_ID

仅 SFC 65 和 SFC 66 使用 REQ_ID 来识别发送的数据。下列情况下，需要在接收端使用参数 REQ_ID：

1）发送端的 CPU 用不同的 REQ_ID 调用几次 SFC 65，将数据发送到同一个通信伙伴。

2）不同的 CPU 调用 SFC 65，将数据发送到同一个通信伙伴。

根据 REQ_ID 可以将接收到的数据保存到不同的存储区。

（3）输入参数 CONT

输入参数 CONT（Continue，继续）的值如果为 1（TRUE），表示任务完成后继续保持与通信伙伴的连接。如果要在两个站之间周期性地交换数据，可以令 CONT 为 1。建立的连接可以用 SFC 69“X_ABORT”来终止。

如果在调用 SFC 时令 CONT = 0，连接在数据传输完成后被终止，又可以使用该连接来与一个新的通信伙伴交换数据。这种方式可以确保只是在实际使用时才占用连接资源。

14.3.2 需要双方编程的 S7 基本通信

S7 基本通信不用组态静态连接，也不用在全局数据表中组态。

如果需要发送的数据超过 76B，可以将数据分为若干个数据包来发送，即多次调用 SFC 65“X_SEND”。每次调用的接收方的 MPI 地址 DEST_ID 相同，但是数据包标识符 REQ_ID 不同。接收方用 SFC 66“X_RCV”接收数据后，根据接收到的 REQ_ID 判别收到的是哪一包数据，并分别存放到不同的地址区。

1．组态硬件和网络

在 STEP 7 中创建一个项目“MPI_UC_1”，用 S7 基本通信在 S7-300 和 S7-400 之间交换数据。要求将它们的 DB 1 和 DB 2 中 76B 的数据发送给对方的 DB 3，接收方将它们分别保存到 DB 4 和 DB 5。随书光盘中 S7 基本通信的例程在文件夹“\Project\MPI_UC”中。

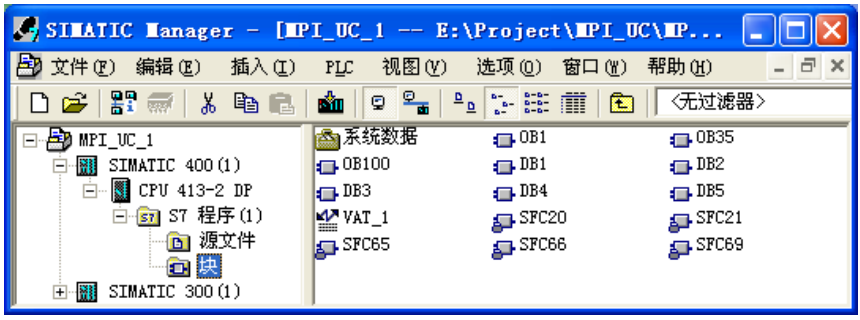


图 14-27　SIMATIC 管理器

生成两个站，CPU 分别为 CPU 413-2DP 和 CPU 315-2 DP（见图 14-27），将它们连接到 MPI 网络上，它们的 MPI 站地址分别为 2 和 3。通信双方都需要调用通信块，一方调用 X_SEND 来发送数据，另一方调用 X_RCV 来接收数据。这种通信方式适用于 S7-300/400 之间的通信。

2．编写发送数据的程序

如果在 OB1 中调用 SFC 65“X_SEND”，在 REQ 信号 M1.0 为 1 时的每个扫描周期调用一次 SFC 65，发送数据的频率太快，将会加重 CPU 的负担。因此在循环中断组织块 OB35 中调用 SFC 65，每隔一定的时间间隔（默认值为 100ms）调用两次 SFC 65。

下面是 2 号站的 OB35 中的程序：

```
程序段 1：准备要发送的数据
      L     DB1.DBW    0
      +     1
      T     DB1.DBW    0                    //每 100ms 将 DB1.DBW0 加 1
      L     DB2.DBW    0
      +     2
      T     DB2.DBW    0                    //每 100ms 将 DB2.DBW0 加 2
程序段 2：发送 DB 1 中的数据
      CALL  "X_SEND"                        //调用 SFC 65
        REQ          :=M1.0                 //发送请求，该参数为 1 时发送
        CONT         :=TRUE                 //发送完成后保持连接
        DEST_ID           :=W#16#3               //接收方的MPI地址
        REQ_ID            :=DW#16#1              //数据包标识符
        SD           :=P#DB1.DBX0.0 BYTE 76 //存放要发送的数据的地址区
        RET_VAL      :=MW12                 //通信状态字，返回的故障信息
        BUSY              :=M1.1                 //为 1 时正在发送，为 0 时发送完成
程序段 3：发送 DB 2 中的数据
      L     ID         0
      T     DB2.DBD    2                    //用 ID0 控制对方的 QD4
      CALL  "X_SEND"                        //调用 SFC 65
        REQ          :=M1.2                 //发送请求，该参数为 1 时发送
        CONT         :=TRUE                 //发送完成后保持连接
        DEST_ID           :=W#16#3               //接收方的MPI地址
        REQ_ID            :=DW#16#2              //数据包标识符
        SD           :=P#DB2.DBX0.0 BYTE 76 //存放要发送的数据的地址区
        RET_VAL      :=MW14                 //通信状态字，返回的故障信息
        BUSY              :=M1.3                 //为 1 时正在发送，为 0 时发送完成
程序段 4：断开动态连接
      CALL  "X_ABORT"                       //调用 SFC 69
        REQ          :=M1.4                 //请求断开连接
        DEST_ID           :=W#16#3               //对方的MPI地址
        RET_VAL      :=MW16                 //返回的故障信息
        BUSY         :=M1.5                 //为 1 表示任务未完成
```

3．编写接收数据的程序

下面是 2 号站的 OB1 中接收数据的程序：

```
程序段 1：从 MPI 接收数据
      CALL  "X_RCV"                         //调用 SFC 66
        EN_DT        :=M0.0                 //接收使能位
```

```
      RET_VAL    :=MW2                          //接收状态字，返回的错误代码
      REQ_ID          :=MD4                     //接收到的数据包的标识符
      NDA        :=M0.1                         //为 1 时表示有新的数据包
      RD         :=P#DB3.DBX0.0 BYTE 76         //存放接收的数据的地址区
程序段 2：保存接收到的数据
      A     M     0.1
      JCN   L3                                 //没有新的数据包则跳转
      L     MD    4                            //取接收到的数据包标识符
      L     1
      ==D                                      //比较数据包标识符是否为 1
      JCN   L1                                 //不是 1 号数据包则跳转
      CALL  "BLKMOV"                           //调用 SFC 20 保存接收到的数据
        SRCBLK     :=P#DB3.DBX0.0 BYTE 76      //源数据区
        RET_VAL    :=MW8                       //状态字
        DSTBLK     :=P#DB4.DBX0.0 BYTE 76      //目标数据区
      JU    L3                                 //无条件跳转
L1:   L     MD    4                            //取接收到的数据包标识符
      L     2
      ==D                                      //比较数据包标识符是否为 2
      JCN   L3                                 //不是 2 号数据包则跳转
      CALL  "BLKMOV"                           //调用 SFC 20 保存接收到的数据
        SRCBLK     :=P#DB3.DBX0.0 BYTE 76      //源数据区
        RET_VAL    :=MW10                      //状态字
        DSTBLK     :=P#DB5.DBX0.0 BYTE 76      //目标数据区
      L     DB5.DBD    2
      T     QD    0                            //用对方的 ID0 控制本站的 QD0
L3:   NOP   0
```

为了观察数据传输的动态效果，将发送站的 ID0 传送给第 2 个数据包中的 DB2.DBD2，它对应于 3 号站接收并保存在 DB5.DBD2 中的数据。用它来控制接收站的 QD0（CPU 413-2DP）或 QD4（CPU 315-2DP）。在运行时改变 ID0 的状态，观察对方的 QD0 或 QD4 是否随之而变。

4．初始化程序

在 2 号站的初始化程序 OB100 中，调用 SFC 21，将存放发送数据的 DB 1 和 DB 2 的各个字预置为 16#4131 和 16#4132，将接收数据的 DB 4 和 DB 5 的各个字清零。

5．两个站程序的区别

3 号站的程序与 2 号站的基本上相同，二者的区别如下：

1）在 3 号站的 OB35 中，X_SEND 和 X_ABORT 中的通信伙伴的 MPI 地址 DEST_ID 为 W#16#2。每次中断将 DB1.DBW0 和 DB2.DBW0 分别加 3 和加 4。

2）在 3 号站的初始化程序 OB100 中，将发送数据的 DB 1 和 DB 2 的各个字预置为 16#3151 和 16#3152。

6．察看动态连接的个数

执行菜单命令“PLC”→“诊断/设置”→“模块信息”，在 CPU 的“模块信息”对话框的“通讯”选项卡中（见图 14–28），可以看到 CPU 的最大连接个数、PG、OP、S7 和其他通信（本例为 S7 基本通信）占用的连接个数。连接个数变化后，需要点击“更新”按钮，才能

看到变化后的连接的个数。

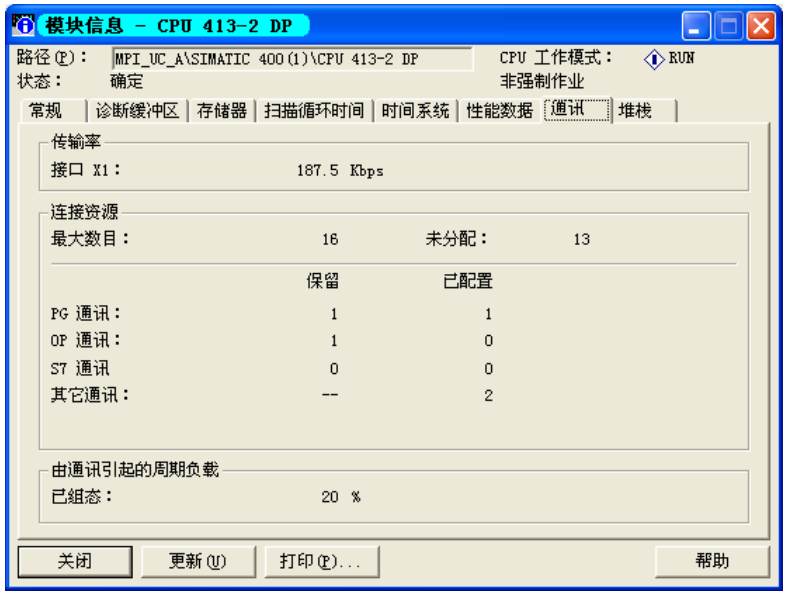


图 14-28　模块信息中的“通讯”选项卡

7．运行与监控

将组态信息和程序分别下载到各 CPU，然后用 PROFIBUS 电缆连接编程用的计算机和两台 PLC 的 MPI 接口，将两块 CPU 切换到 RUN 模式。

在运行时同时打开两个站的变量表（见图 14-29 和图 14-30），其中的 M1.0 和 M1.2 是发送使能位，M0.0 是接收使能位，M1.4 是 X_ABORT 的使能位，用来断开连接。

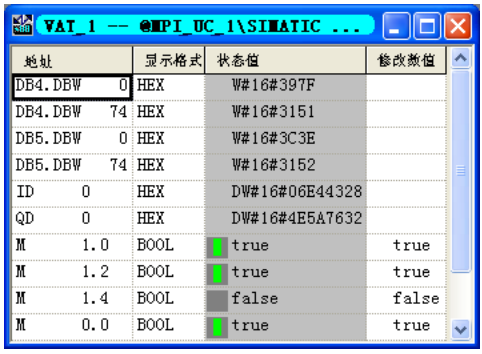


图 14-29　CPU 413-2DP 的变量表

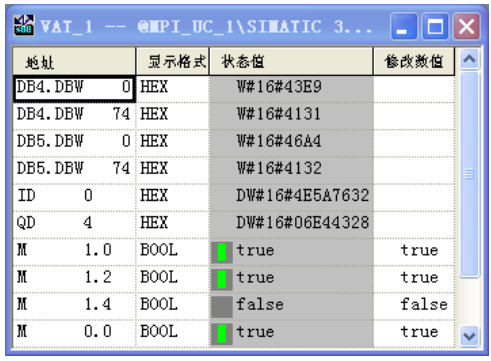


图 14-30　CPU 315-2DP 的变量表

在运行时用变量表监视接收的 DB 4、DB 5 的第一个字 DBW0 和最后一个字 DBW74。在每 100ms 执行一次的 OB35 中，两个数据包发送的第一个字 DB1.DBW0 和 DB2.DBW0 分别被加 1～4，通信伙伴用 DB4.DBW0 和 DB5.DBW0 来保存它们。

（1）初始状态

开始运行时双方的 SFC 的使能位均为 0 状态（false），动态连接数为 0，没有传输数据，

用来保存接收的数据的 DB 4、DB 5 中的字和 QD0、QD4 均为 0。

（2）建立连接

令 3 号站的发送使能位 M1.0 为 1，建立了一个动态连接。2 号站的接收使能位 M0.0 必须同时为 1，才能接收到 3 号站发送的数据，2 号站的 DB4.DBW0 的值才会不断增大。

因为每个站既要发送数据，又要接收数据，最多需要两个连接。令 2 号站的某个发送使能位为 1，将建立第二个动态连接。

（3）连接的保持

如果令 3 号站的发送使能位为 0，或者令 2 号站的接收使能位 M0.0 为 0，都会中断数据传输，2 号站的 DB4.DBW0 或 DB5.DBW0 的值停止变化。因为 SFC 65 的参数 CONT 为 1（TRUE），即使发送数据使能位 M1.0 变为 0，数据传输中断，连接仍然保持。

（4）断开连接

将 3 号站的 X_ABORT 的使能位 M1.4 置位为 1，将断开一个已建立的连接。M1.4 为 1 时，即使双方的 3 号站的发送使能位和 2 号站的接收使能位均为 1，也不能传输数据，DB4.DBW0 或 DB5.DBW0 停止变化。双方的 M1.4 均为 1 状态时，数据传输完全停止，两条连接均断开，模块信息中的“其他通信”的连接数变为 0。

（5）连接的恢复

令 X_ABORT 的使能位 M1.4 为 0，在对应的发送使能位为 1 时，将会恢复被断开的连接。

8．两台 S7-300 之间的通信

随书光盘中的项目 MPI_UC_3 调用 X_SEND 和 X_RCV，来实现两块 CPU 315-2DP 之间的 S7 基本通信，硬件、通信的组态、编程和监控的方法与项目 MPI_UC_1 基本上相同，详细的情况见随书光盘中的项目。

14.3.3 只需一个站编程的 S7 基本通信

1．SFC 68“X_PUT”和 SFC 67“X_GET”

除了上述使用 SFC 65“X_SEND”和 SFC 66“X_RCV”的双边通信之外，还可以实现单边通信。编写程序的一方的 CPU 是客户机，另一方作服务器，后者不需要编写程序。客户机是通信的主动方，服务器是被动方。

在 S7 基本通信中，S7-300 或 S7-400 都可以作客户机或服务器，S7-200 只能作服务器。

在 STEP 7 中创建一个项目（见随书光盘中的例程 MPI_UC_2），生成两个站，CPU 分别为 CPU 413-2DP 和 CPU 315-2 DP，将它们连接到 MPI 网络上，它们的 MPI 站地址分别为 2 和 3。

在单边通信中，客户机（CPU 413-2DP）调用 SFC 68“X_PUT”来写服务器（CPU 315-2DP）中的地址区，调用 SFC 67“X_GET”来读取服务器中的数据。在读取服务器数据时，客户机首先要发送读取命令给服务器。

2．客户机的程序设计

如果在 OB1 中调用 SFC 68“X_PUT”和 SFC 67“X_GET”，在它们的通信请求信号 REQ 为 1 时，每个扫描周期都要执行一次读、写操作。为了减少发送读、写命令的次数，在循环中断组织块 OB35 中调用 SFC 68“X_PUT”和 SFC 67“X_GET”。下面是客户机 CPU 413-2DP 的 OB35 中的程序：

程序段 1：准备要发送的数据

```
L       DB1.DBW     0
+       1
T       DB1.DBW     0                   //每 100ms 将 DB1.DBW 加 1
L       ID      0
T       DB1.DBD     2                   //用本站的 ID0 控制对方的 QD4
```

程序段 2：用 SFC 68 写对方的数据区

```
CALL    "X_PUT"                         //调用 SFC 68
  REQ           :=M0.0                  //写数据使能信号
  CONT          :=TRUE                  //发送完成后保持连接
  DEST_ID       :=W#16#3                //对方的MPI地址
  VAR_ADDR      :=P#DB2.DBX0.0 BYTE 76  ///通信伙伴要写入数据的地址区
  SD            :=P#DB1.DBX0.0 BYTE 76  //存放本站要发送的数据的地址区
  RET_VAL       :=MW2                   //通信状态字，返回的错误信息
  BUSY          :=M0.1                  //为 1 时写数据未完成
```

程序段 3：用 SFC 67 读对方的数据区

```
CALL    "X_GET"                         //调用 SFC 67
  REQ           :=M0.2                  //读数据使能信号
  CONT          :=TRUE                  //读数据完成后保持连接
  DEST_ID           :=W#16#3                //对方的MPI站地址
  VAR_ADDR      :=P#DB1.DBX0.0 BYTE 76  //要读取的通信伙伴的地址区
  RET_VAL       :=MW4                   //通信状态字，返回的错误信息
  BUSY          :=M0.3                  //为 1 时读数据未完成
  RD            :=P#DB2.DBX0.0 BYTE 76  //本站存放读取的数据的地址区
L       DB2.DBD     2
T       QD          0                   //用对方的 ID0 控制本站的 QD0
```

程序段 4：断开连接

```
CALL    "X_ABORT"                       //调用 SFC 69
  REQ           :=M0.4                  //使能信号
  DEST_ID           :=W#16#3                //对方的MPI地址
  RET_VAL       :=MW6                   //返回的错误信息
  BUSY          :=M0.5                  //为 1 表示操作未完成
```

如果在 3 号站调用 SFC 67“X_GET”和 SFC 68“X_PUT”读写 2 号站的系统数据区，可以使用与上面基本上相同的程序，只需要将 SFC 中的 DEST_ID（对方的 MPI 地址）改为 W#16#2。

SFC 69“X_ABORT”可以中断一个由 SFC“X_SEND”、“X_GET”或“X_PUT”建立的连接。如果上述 SFC 的操作已经完成（BUSY = 0），调用 SFC 69“X_ABORT”后，通信双方的连接资源被释放。

CPU 413-2DP 的初始化程序 OB100 调用 SFC 21，将发送数据的 DB 1 的各个字预置为 16#4444，将接收数据的 DB 2 的各个字清零。

3．服务器程序设计

下面是服务器（CPU 315-2DP）的 OB1 中的程序。

程序段 1：

```
L     ID      0
T     DB1.DBD     2                    //用本站的 ID0 控制对方的 QD0
L     DB2.DBD     2
T     QD      4                        //用通信伙伴的 ID0 控制本站的 QD4
```

CPU 315-2DP 的初始化程序 OB100 调用 SFC 21，将存放发送数据的 DB 1 的各个字预置为 16#3333，将存放接收数据的 DB 2 的各个字清零。

在 CPU 315-2DP 的 OB35 中，每 100ms 将 DB1.DBW 加 2。

4．运行与监控

将组态信息和程序分别下载到各 CPU，然后用 PROFIBUS 电缆连接编程用的计算机和两台 PLC 的 MPI 接口，将各 CPU 切换到 RUN 模式。在运行时同时打开两个站的变量表，M0.0、M0.2 和 M0.4 分别是 X_PUT、X_GET 和 X_ABORT 的使能位。

在运行时用变量表监视通信双方存放接收到的数据的 DB 2 的第一个字 DBW0 和最后一个字 DBW74（见图 14-31 和图 14-32）。

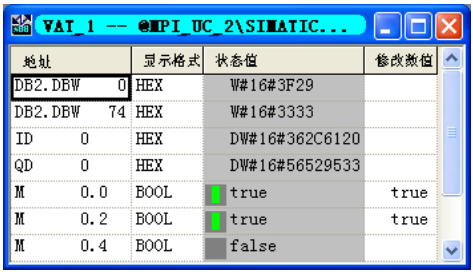


图 14-31　CPU 413-2DP 的变量表

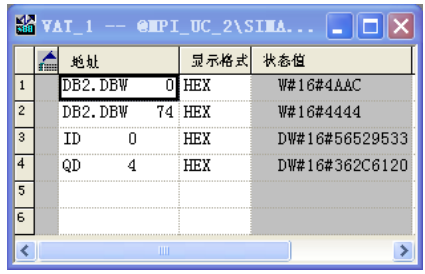


图 14-32　CPU 315-2DP 的变量表

读、写使能信号 M0.0 和 M0.2 为 1，X_ABORT 的使能信号 M0.4 为 0 时，X_PUT 和 X_GET 正常运行。因为 X_PUT 和 X_GET 的参数 CONT 均为 1（TRUE），即使读、写使能位 M0.0 和 M0.2 由 1 状态变为 0 状态，连接仍然保持。可以通过置位 M0.4 来使能 SFC 69，断开已建立的连接。

由 CPU 的“模块信息”对话框的“通讯”选项卡可知，X_PUT 和 X_GET 共用一个“其他通讯”的动态连接。X_PUT、X_GET 的 REQ 均变为 0 时，通信停止，但是仍然占用一个“其他通讯”动态连接。ABORT 的使能信号为 1 时，动态连接数为 0。ABORT 的使能信号变为 0 后，再次用 M0.0 或 M0.2 启动通信，又占用一个动态连接。

5．两台 S7-300 之间的通信

随书光盘中的项目 MPI_UC_4 调用 X_PUT 和 X_GET，来实现两块 CPU 315-2DP 之间的 S7 基本通信，硬件、通信的组态、编程和监控的方法与项目 MPI_UC_2 基本上相同，详细的情况见随书光盘中的项目。

14.3.4　S7 基本通信 SFC 综合应用例程

1．对通信的基本要求

创建一个项目（见随书光盘中的例程 MPI_UC_5），生成 3 个站，一个站的 CPU 为 CPU

413-2DP，另外两个站的 CPU 为 CPU 315-2 DP（见图 14-34），在 NetPro 中将它们连接到 MPI 网络上，它们的 MPI 站地址分别为 2、3、4（见图 14-35）。

要求通信按下述的顺序进行：

1）2 号站调用 X_PUT 写 3 号站的数据。

2）2 号站调用 X_GET 读 3 号站的数据。

3）2 号站调用 X_SEND 向 3 号站发送数据，3 号站调用 X_RCV 接收数据。

4）2 号站调用 X_ABORT 断开与 3 号站的连接。

5）2 号站调用 X_SEND 向 4 号站发送数据，4 号站调用 X_RCV 接收数据。

6）2 号站调用 X_ABORT 断开与 4 号站的连接。

上述通信操作周期性地反复进行。

在通信过程中，应保证 2 号站同时只能调用一个通信 SFC。通信程序用顺序控制设计法来编写，顺序功能图（请参阅参考文献[2]的 5.2 节）如图 14-33 所示。

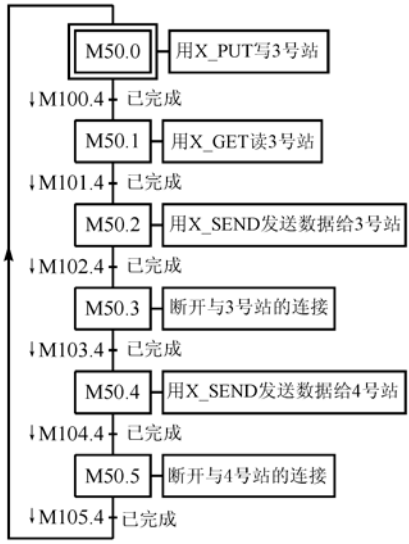


图 14-33　2 号站的顺序功能图

通信 SFC 执行的时间较长，可以通过通信 SFC 的输出参数 BUSY（M10x.4）的状态来判断通信任务是否结束。BUSY 为 1 状态时，表示通信任务尚未完成，为 0 状态时表示通信任务结束。因此用 SFC 的 BUSY 信号（M10x.4）的下降沿作为相邻步之间的转换条件。顺序功能图中转换条件 M10x.4 左边的“↓”表示下降沿有效。

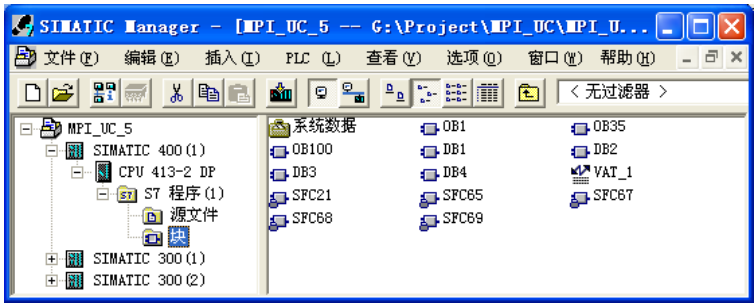


图 14-34　SIMATIC 管理器

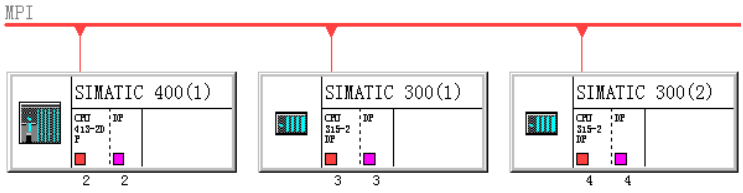


图 14-35　MPI 网络

2．顺序控制程序

在 2 号站的初始化程序 OB100 中，将顺序功能图中初始步对应的 M50.0 置位为 1，将其余各步对应的存储器位复位为 0，具体程序如下：

```
      L     0
      T     MB      50
      SET
      =     M       50.0
```

下面是用参考文献[2]的 5.4 节中的方法编写的 2 号站的 OB1 中的顺序控制程序：

```
程序段 1：
      A     M       50.0        //调用 X_PUT 将数据写入 3 号站
      A     M       100.4
      FN    M       100.5       //在 BUSY 信号的下降沿，通信任务已结束
      S     M       50.1        //将后续步置位为活动步
      R     M       50.0        //将前级步复位为不活动步
程序段 2：
      A     M       50.1        //调用 X_GET 读 3 号站的数据
      A     M       101.4
      FN    M       101.5       //在 BUSY 信号的下降沿，通信任务已结束
      S     M       50.2        //将后续步置位为活动步
      R     M       50.1        //将前级步复位为不活动步
程序段 3：
      A     M       50.2        //调用 X_SEND 将数据发送到 3 号站
      A     M       102.4
      FN    M       102.5       ///在 BUSY 信号的下降沿，通信任务已结束
      S     M       50.3        //将后续步置位为活动步
      R     M       50.2        //将前级步复位为不活动步
程序段 4：
      A     M       50.3        //调用 X_ABORT 断开与 3 号站的连接
      A     M       103.4
      FN    M       103.5       //在 BUSY 信号的下降沿，操作结束
      S     M       50.4        //将后续步置位为活动步
      R     M       50.3        //将前级步复位为不活动步
程序段 5：
      A     M       50.4        //调用 X_SEND 将数据写入 4 号站
      A     M       104.4
      FN    M       104.5       //在 BUSY 信号的下降沿，通信任务已结束
      S     M       50.5        //将后续步置位为活动步
      R     M       50.4        //将前级步复位为不活动步
程序段 6：
      A     M       50.5        //调用 X_ABORT 断开与 4 号站的连接
      A     M       105.4
      FN    M       105.5       //在 BUSY 信号的下降沿，操作结束
      S     M       50.0        //将后续步置位为活动步
      R     M       50.5        //将前级步复位为不活动步
```

3．2 号站的通信程序

为了防止通信块执行的时间间隔过快，在 2 号站每 100ms 执行一次的 OB35 中，用各步对应的存储器位 M50.0～M50.5 来分别调用对应的通信块。下面是 2 号站的 OB35 中的程序：

程序段 1：使发送的数据动态变化

```
      L     DB1.DBW    0
      +     1
      T     DB1.DBW    0
      L     DB4.DBW    0
      +     2
      T     DB4.DBW    0
```

程序段 2：在第 1 步调用 X_PUT，将数据写入 3 号站

```
      A          M     50.0
      JCN        m001                          //不是第 1 步则跳转
      L       ID     0
      T       DB1.DBD    2                     //用 ID0 控制 3 号站的 QD4
      CALL   "X_PUT"                           //调用 SFC 68，将数据写入 3 号站
        REQ          :=TRUE                    //写数据使能信号
        CONT         :=TRUE                    //写数据完成后保持连接
        DEST_ID      :=W#16#3                  //对方的MPI地址
        VAR_ADDR     :=P#DB2.DBX0.0 BYTE 76    //通信伙伴要写入数据的地址区
        SD           :=P#DB1.DBX0.0 BYTE 76    //存放本站要发送的数据的地址区
        RET_VAL      :=MW130                   //通信状态字，返回的故障信息
        BUSY         :=M100.4                  //为 1 时写数据未完成
      BEU                                      //块结束
```

程序段 3：在第 2 步调用 X_GET，读 3 号站的数据

```
m001: A          M     50.1
      JCN        m002                          //不是第 2 步则跳转
      CALL   "X_GET"                           //调用 SFC 67
        REQ          :=TRUE                    //读数据使能信号
        CONT         :=TRUE                    //读数据完成后保持连接
        DEST_ID      :=W#16#3                  //对方的MPI站地址
        VAR_ADDR     :=P#DB1.DBX0.0 BYTE 76    //要读取的通信伙伴的地址区
        RET_VAL      :=MW132                   //通信状态字，返回的故障信息
        BUSY         :=M101.4                  //为 1 时读数据未完成
        RD                 :=P#DB2.DBX0.0 BYTE 76      //本站存放读取的数据的地址区
      L       DB2.DBD    2
      T       QD     0                         //用 3 号站的 ID0 控制本站的 QD0
      BEU                                      //块结束
```

程序段 4：在第 3 步调用 X_SEND，向 3 号站发送数据

```
m002: A          M     50.2
      JCN        m003                          //不是第 3 步则跳转
      CALL   "X_SEND"                          //调用 SFC 65
        REQ          :=TRUE                    //发送请求，该参数为 1 时发送
        CONT         :=TRUE                    //发送完成后保持连接
        DEST_ID      :=W#16#3                  //接收方的MPI地址
        REQ_ID       :=DW#16#1                 //数据包标识符
        SD           :=P#DB3.DBX0.0 BYTE 76    //存放要发送的数据的地址区
        RET_VAL      :=MW134                   //通信状态字，返回的故障信息
        BUSY         :=M102.4                  //为 1 时正在发送，为 0 时发送完成
```

```
        BEU                                          //块结束
程序段 5：在第 4 步调用 X_ABORT 断开与 3 号站的连接
m003:   A       M           50.3
        JCN     m004                                 //不是第 4 步则跳转
        CALL    "X_ABORT"                            //调用 SFC 69
          REQ           :=TRUE                       //使能信号
          DEST_ID       :=W#16#3                     //3 号站的MPI地址
          RET_VAL       :=MW136                      //返回的错误信息
          BUSY          :=M103.4                     //为 1 表示操作未完成
        BEU                                          //块结束
程序段 6：在第 5 步调用 X_SEND，向 4 号站发送数据
m004:   A       M           50.4
        JCN     m005                                 //不是第 5 步则跳转
        L       ID          0
        T       DB4.DBD     2                        //用 ID0 控制 4 号站的 QD4
        CALL    "X_SEND"                             //调用 SFC 65
          REQ           :=TRUE                       //发送请求，该参数为 1 时发送
          CONT          :=TRUE                       //发送完成后保持连接
          DEST_ID       :=W#16#4                     //接收方的MPI地址
          REQ_ID        :=DW#16#2                    //数据包标识符
          SD            :=P#DB4.DBX0.0 BYTE 76       //存放要发送的数据的地址区
          RET_VAL       :=MW138                      //通信状态字，返回的故障信息
          BUSY          :=M104.4                     //为 1 时正在发送，为 0 时发送完成
        BEU                                          //块结束
程序段 7：在第 6 步调用 X_ABORT 断开与 4 号站的连接
m005:   A       M       50.5
        JCN     m006                                 //不是第 6 步则跳转
        CALL    "X_ABORT"                            //调用 SFC 69
          REQ           :=TRUE                       //使能信号
          DEST_ID       :=W#16#4                     //4 号站的MPI地址
          RET_VAL       :=MW140                      //返回的错误信息
          BUSY          :=M105.4                     //为 1 表示操作未完成
m006:  NOP      0                                    //块结束
```

4．3 号站和 4 号站的程序

下面是 3 号站的 OB1 接收数据的程序：

```
程序段 1：接收 2 号站发送的数据
        CALL    "X_RCV"                              //调用 SFC 66
          EN_DT         :=M0.0                       //接收使能位
          RET_VAL       :=MW2                        //接收状态字，返回的错误代码
          REQ_ID        :=MD4                        //接收到的数据包的标识符
          NDA           :=M0.1                       //为 1 时表示有新的数据包
          RD            :=P#DB3.DBX0.0 BYTE 76       //存放接收的数据的地址区
        L       DB2.DBD     2
        T       QD      4                            //用 2 号站的 ID0 控制本站的 QD4
        L       ID      0
```

```
T       DB1.DBD    2                    //用本站的 ID0 控制 2 号站的 QD0
```

下面是 4 号站的 OB1 中接收数据的程序：

程序段 1：接收 2 号站发送的数据

```
CALL   "X_RCV"                          //调用 SFC 66
  EN_DT        :=M0.0                   //接收使能位
  RET_VAL      :=MW2                    //接收状态字，返回的错误代码
  REQ_ID       :=MD4                    //接收到的数据包的标识符
  NDA          :=M0.1                   //为 1 时表示有新的数据包
  RD           :=P#DB3.DBX0.0 BYTE 76   //存放接收的数据的地址区
L       DB3.DBD    2
T       QD         4                    //用 2 号站的 ID0 控制本站的 QD4
```

3 号站和 4 号站的 OB35 每 100ms 将 DB1.DBW0 加 1，程序如下：

程序段 1：使发送的数据动态变化

```
L       DB1.DBW    0
+       1
T       DB1.DBW    0
```

5．通信数据区的初始化

1）2 号站的初始化组织块 OB100 调用 SFC 21，将 X_PUT 的源数据区 DB 1 的各个字预置为 W#16#4001，将写入 3 号站和 4 号站的 DB 3 和 DB4 的各个字分别预置为 W#16#4003 和 W#16#4004。将存放用 X_GET 读取的数据的 DB 2 的各个字清零。

2）3 号站的初始化组织块 OB100 将 CPU 413-2DP 用 X_GET 读取的 DB 1 中的字预置为 W#16#3001，将 CPU 413-2DP 用 X_PUT 和 X_SEND 写入数据的 DB 2 和 DB 3 的各个字清零。

3）4 号站的初始化组织块 OB100 将用于接收 CPU 413-2DP 用 X_SEND 发送的数据的 DB 3 的各个字清零。

6．下载与监控

将组态信息和程序分别下载到各 CPU，然后用 PROFIBUS 电缆连接编程用的计算机和 3 台 PLC 的 MPI 接口，将各 CPU 切换到 RUN 模式。

在运行时同时打开 3 个站的变量表（见图 14-36～图 14-38），用变量表监视各个站接收到的数据的第一个字 DBW0 和最后一个字 DBW74，以及本站的 ID0 和 QD0（或 QD4）。

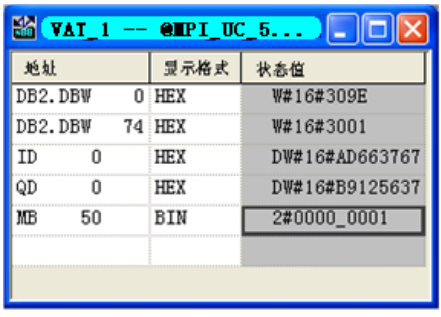


图 14-36　CPU 413-2DP 的变量表

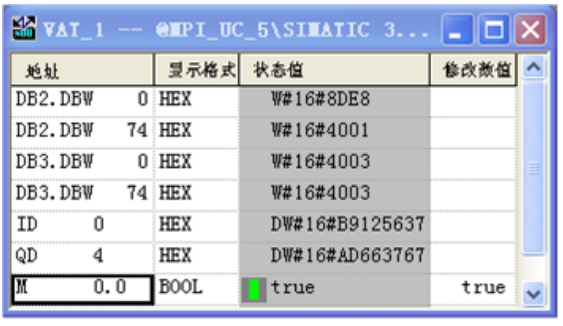


图 14-37　3 号站的变量表

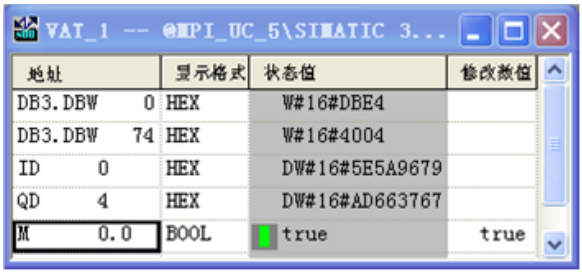


图 14-38　4 号站的变量表

2 号站的 DB 2 用来存放用 X_GET 读取的 3 号站的数据。3 号站的 DB 2 存放的是 2 号站用 X_PUT 写入的数据。

3 号站和 4 号站的 DB 3 是用 X_RCV 接收到的 2 号站用 X_SEND 发送的数据。3 号站和 4 号站的变量表中的 M0.0 是 X_RCV 的接收使能位，用它来控制是否接收数据。

因为用 2 号站的 ID0 控制 3 号站和 4 号站的 QD4，从变量表可以看出，三者的状态完全相同。在控制通信过程的顺序控制程序正常运行时，可以看到 2 号站的变量表中的 MB50 的第 0～5 位（对应于图 14-33 中的各步）在同一时刻只有一位为一，并且这 6 位循环左移，依次轮流变为 1 状态。

14.4　S7-200 与 S7-300 的 MPI 通信

在 S7 基本通信中，S7-300/400 作为客户机，可以用 X_GET 和 X_PUT 来读 S7-200 的 I 区，写 S7-200 的 Q 区。S7-300/400 在读写 S7-200 的 V 存储区（变量存储区）时，用数据块 DB 1 来代替 V 存储器。S7-200 在 S7 基本通信中只能作服务器，不需要编写 S7-200 的通信程序。S7-200 应将需要与 S7-300/400 交换的数据传送到一个连续的 V 存储区，为数据传送做好准备。

1．设置通信参数

实现 S7-200 和 S7-300/400 的 MPI 通信的必要条件是它们采用相同的通信速率，它们的站地址不能重叠。

（1）设置计算机与 S7-200 通信的接口

用 PC/PPI 通信电缆连接台式计算机的 RS-232C 接口和 CPU 224 的 RS-485 接口，打开 S7-200 的编程软件 STEP 7-Micro/WIN V4.0（见图 14-39）。双击左边的指令树的“通信”文件夹中的“设置 PG/PC 接口”图标，在打开的对话框中，选中“PC/PPI cable（PPI）”。

点击“属性”按钮，打开“属性 – PC/PPI cable”对话框，按图 14-39 设置通信参数，传输速率应与 CPU 中的相同才能通信。“本地连接”选项卡中的 COM 口是实际使用的串口。

双击 STEP 7-Micro/WIN 左边窗口的“通信”文件夹中的“通信”图标，打开“通信”对话框，点击其中的“双击刷新”图标。如果计算机可以与 CPU 224 通信，将会显示出 CPU 224。这一步不是实现通信必须的操作。

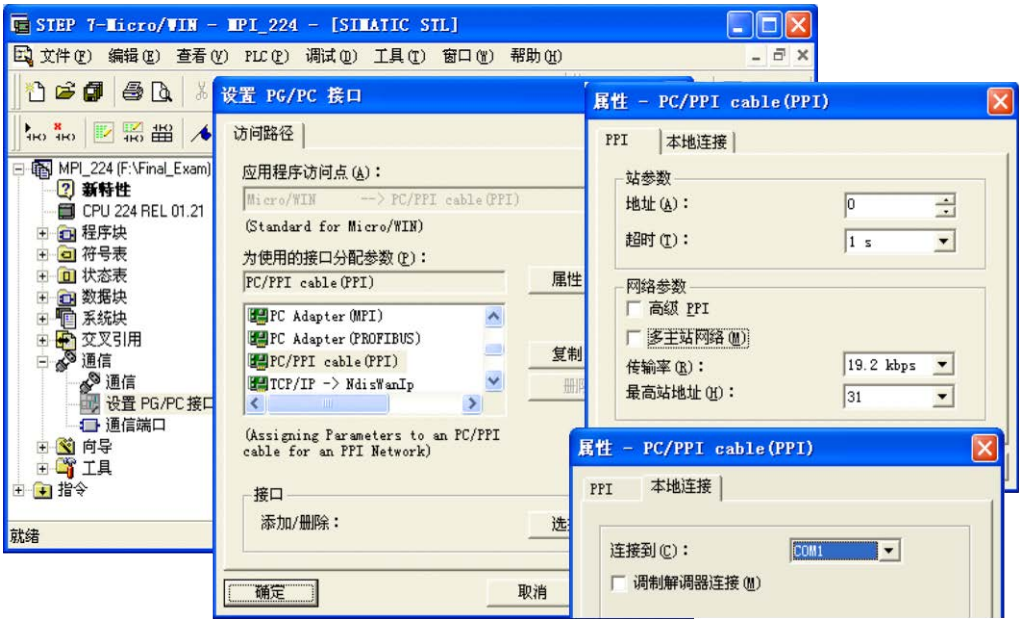


图 14-39　设置 PG/PC 接口

（2）设置 CPU 224 的通信参数

新版的 S7-200 或 S7-200CN 的通信速率可以设置为 9.6kbit/s、19.2kbit/s 和 187.5 kbit/s，作者做试验使用的 CPU 224 的固件版本为 V01.21，不能使用 187.5kbit/s 的波特率，例程中选用的波特率为 19.2kbit/s。S7-200 和 S7-300 默认的 MPI 地址都为 2。双击 STEP 7-Micro/WIN 左边的指令树的“通信”文件夹中的“通信端口”图标，打开系统块中的通信端口对话框（见图 14-40），将 CPU 224 的 MPI 地址设置为 3，通信速率设置为 19.2kbit/s。需要将系统块下载到 CPU 224，设置的参数才会起作用。

图 14-40　设置 S7-200 的通信接口

（3）设置 CPU 315-2DP 的通信参数

在 STEP 7 中，用新建项目向导创建一个项目（见随书光盘中的例程 MPI_224A），将 CPU 315-2DP 连接到 MPI 网络上，设置 MPI 站地址为 2，MPI 网络的传输速率为 19.2 kbit/s。将设置的参数下载到 CPU 315-2DP。

（4）设置 CP 5613 的通信参数

用 PROFIBUS 电缆连接计算机的通信卡 CP 5613、CPU 315-2DP 和 CPU 224 的 MPI 接口，将各设备通电。在 SIMATIC 管理器中，执行菜单命令“选项”→“设置 PG/PC 接口”，设置 CP 5613 使用 MPI 协议，波特率为 19.2 kbit/s。

2．CPU 315-2DP 读写 CPU 224 的 I/O 区的编程

在 CPU 315-2DP 的 OB35 中编写下面的程序，OB35 采用默认的执行周期 100ms。因为参数 REQ 和 CONT 均为 1（TRUE），每 100ms 执行一次 SFC 68“X_PUT”，将本站的 IW0 写入 CPU 224 的 QW0。每 100ms 执行一次 SFC 67“X_GET”，将读取的 CPU 224 的 IW0 保存到本站的 QW4。读写请求信号 REQ 也可以用变量（例如 M10.0）来控制。

程序段 1：将 CPU 315-2DP 的 IW0 写入 S7-200 的 QW0

```
CALL   "X_PUT"                          //调用 SFC 68
  REQ          :=TRUE                   //写数据使能信号
  CONT         :=TRUE                   //写数据完成后保持连接
  DEST_ID      :=W#16#3                 //S7-200 的MPI地址
  VAR_ADDR     :=QW0                    //写目的地址区，S7-200 的QW0
  SD           :=IW0                    //本站的发送数据区
  RET_VAL      :=MW2                    //通信状态字，返回的故障信息
  BUSY         :=M0.1                   //为 1 时写数据未完成
```

程序段 2：读取 S7-200 的 IW0，写入本站的 QW4

```
CALL   "X_GET"                          //调用 SFC 67
  REQ          :=TRUE                   //读数据使能信号
  CONT         :=TRUE                   //读数据完成后保持连接
  DEST_ID          :=W#16#3                   //对方的MPI站地址
  VAR_ADDR     :=IW0                    //读取S7-200 的IW0
  RET_VAL      :=MW4                    //通信状态字，返回的故障信息
  BUSY         :=M0.3                   //为 1 时读数据未完成
  RD           :=QW4                    //读取的数据在本站中的存储地址
```

将程序块和系统数据下载到 CPU 315-2DP，连接好 CPU 224 和 CPU 315-2DP 的通信接口，在 RUN 模式，可以用双方的 IW0 控制通信伙伴的 QW0 或 QW4。

3．CPU 315-2DP 读写 CPU 224 的 V 区的编程

在 STEP 7 中，用新建项目向导创建一个项目（见随书光盘中的例程 MPI_224B），将 CPU 315-2DP 连接到 MPI 网络上，MPI 站地址为 2，MPI 网络的传输速率为 19.2 kbit/s。将设置的参数下载到 CPU 315-2DP。与它通信的 CPU 224 的 MPI 站地址为 3，传输速率为 19.2 kbit/s。

初始化程序 OB100 将 DB 1 的 76B 数据发送区的字预置为 W#16#3333，将 DB 2 的 76B 数据接收区复位为 0。

在 CPU 315-2DP 的 OB35 中调用 SFC 68“X_PUT”，将本站的 DB 1 的 76B 数据发送到通信伙伴的 DB 1 的 DBB100～DBB175，即 CPU 224 的 VB100～VB175。调用 SFC 67“X_GET”，

读取 CPU 224 的 VB200～VB275（即 DB 1 的 DBB200～DBB275）中的数据，将它们存放到 DB 2。执行 OB35 的时间间隔为默认的 100ms。

下面是 CPU 315-2DP 的循环中断组织块 OB35 的程序：

```
程序段 1：准备要发送的数据
        L     DB1.DBW    0
        +     1
        T     DB1.DBW    0                    //每 100ms 将 DB1.DBW0 加 1
        L     IW         0
        T     DB1.DBW    2                    //用本站的 IW0 控制 S7-200 的 QW0
程序段 2：将本站的 DB1 中的数据写入 CPU 224 的 V 区
        CALL  "X_PUT"                         //调用 SFC 68
          REQ          :=TRUE                 //写数据使能信号
          CONT         :=TRUE                 //发送完成后保持连接
          DEST_ID      :=W#16#3               //S7-200 的MPI地址
          VAR_ADDR     :=P#DB1.DBX100.0 BYTE 76  //S7-200 要写入数据的VB100～VB175
          SD           :=P#DB1.DBX0.0 BYTE 76    //存放本站要发送的数据的地址区
          RET_VAL      :=MW2                  //通信状态字，返回的故障信息
          BUSY         :=M0.1                 //为 1 时写数据未完成
程序段 3：读取 CPU 224 的 V 区的数据，保存到本站的 DB 2
        CALL  "X_GET"                         //调用 SFC 67
          REQ          :=TRUE                 //读数据使能信号
          CONT         :=TRUE                 //读数据完成后保持连接
          DEST_ID          :=W#16#3               //对方的MPI站地址
          VAR_ADDR     := P#DB1.DBX200.0 BYTE 76 //要读取S7-200 的VB200～VB275
          RET_VAL      :=MW4                  //通信状态字，返回的故障信息
          BUSY         :=M0.3                 //为 1 时读数据未完成
          RD           := P#DB2.DBX0.0 BYTE 76   //本站存放读取的数据的地址区
        L     DB2.DBW    2
        T     QW         4                    //用 S7-200 的 IW0 控制本站的 QW4
程序段 4：断开连接
        CALL  "X_ABORT"                       //调用 SFC 69
          REQ          :=M0.4                 //使能信号
          DEST_ID          :=W#16#3               //对方的MPI地址
          RET_VAL      :=MW6                  //返回的错误信息
          BUSY         :=M0.5                 //为 1 表示操作未完成
```

4．CPU 224 的程序

CPU 224 的项目文件为 MPI_224.mwp，下面是 CPU 224 的 OB1 中的程序：

```
LD      SM0.1                 //在第一个扫描周期
FILL    16#2222, VW204, 36    //将发送数据区 VW204～VW274 预置为 16#2222
FILL    16#0, VW100, 38       //将接收数据区 VW100～VW174 清零
LD      SM0.5                 //周期为 1s 的时钟脉冲
EU                            //上升沿检测
INCW    VW200                 //每秒钟 VW200 加 1
LD      SM0.0                 //SM0.0 的常开触点一直闭合
```

```
MOVW    IW0, VW202                  //用本站的 IW0 控制 S7-300 的 QW4
MOVW    VW102, QW0                  //用 S7-300 的 IW0 控制本站的 QW0
```

通过双方的程序和通信，双方用本站的 IW0 控制对方的 QW0 或 QW4（见图 14-41），在运行时改变 IW0 的输入值，通信伙伴的 QW0 或 QW4 随之而变。

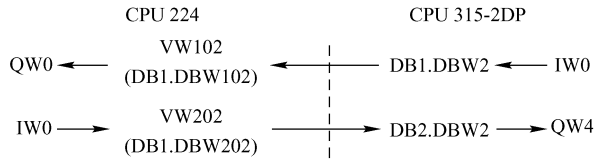


图 14-41　数据传送关系

用 CPU 315-2DP 的变量表监视本站参与通信的存储区，对应于 S7-200 的 VW200 的 DB2.DBW0 的值每 1s 加 1。用数组定义 DB 1 和 DB 2 的大小，在运行时监控 DB 2，可以看到，DB2.DBW4～DBW74 的值均为接收到的 16#2222。

图 14-42　CPU 315-2DP 的变量表

CPU 224 只有一个通信接口，与 S7-300 通信时，不能用 STEP 7 或 STEP 7-Micro/WIN 监控 CPU 224，只能通过 QW0 来观察它的内部变量的情况。例如在 CPU 224 的 OB1 中，将上述程序中的指令“MOVW　VW102, QW0”改为“MOVW　VW104, QW0”，用 VW104 接收到的 S7-300 的 DB1.DBW4 的值控制本站的 QW0，可以通过 QW0 查看 VW104 的值。

也可以将接收到的数据保存在有断电保持功能的 V 区，将 CPU 断电后，断开与 MPI 网络的连接，然后用 PC/PPI 电缆和 S7-200 的编程软件来读取通信时接收到的数据。

14.5　基于 MPI 网络的 S7 通信

S7 通信是需要建立连接的通信，S7 通信可以用于 PROFIBUS-DP、MPI 和工业以太网。这 3 种网络的 S7 通信的组态和编程的方法基本上相同。S7 通信详细的组态和编程的方法见 4.2 节和 4.3 节。

在 CPU 集成的通信接口组成的 MPI 网络的 S7 通信中，S7-400 CPU 通过调用 SFB GET 和 PUT，既可以作服务器（Server），也可以作客户机（Client），进行单向通信，对其他 CPU 的数据进行读/写操作。还可以调用 SFB USEND/URCV、BSEND/ BRCV 发送和接收数据，进行双向通信。S7-300 CPU 的 MPI 接口只能作 S7 通信的服务器，不能作通信的客户机，不能主动发送和接收数据。

14.5.1　单向 S7 通信

在本节的单向 S7 通信例程中，S7-400 作客户机，S7-300 作服务器，客户机调用单向通信块 SFB GET 和 PUT，通过集成的 MPI 接口和 S7 通信，读、写服务器的存储区。服务器是

通信中的被动方，不需要编写通信程序。S7-300 和 S7-400 之间只能建立单向的 S7 连接。

1．S7 连接的组态

在 STEP 7 中创建一个名为 MPI_S7_1 的项目（本节的例程在随书光盘的文件夹“\Project\MPI_S7”中），生成两个站，CPU 模块分别为 CPU 413-2 DP 和 CPU 315-2 DP（见图 14-43）。

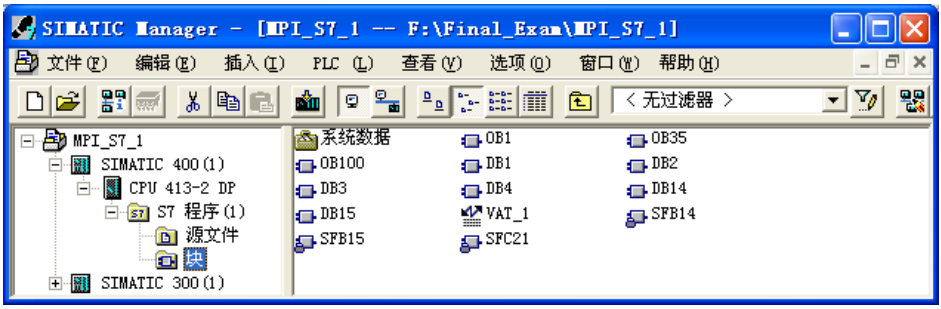


图 14-43　SIMATIC 管理器

点击 SIMATIC 管理器中的按钮，打开网络组态工具 NetPro，将两个站连接到 MPI 网络上，设置它们的 MPI 站地址分别为 2 和 3（见图 14-44）。

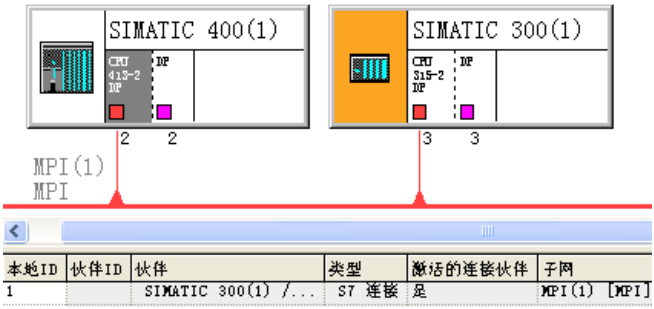


图 14-44　网络与连接的组态

选中 2 号站的 CPU 413-2DP 所在的小方框，在 NetPro 下面的窗口出现连接表。双击连接表中的第一行，在出现的“插入新连接”对话框中（见图 14-45 的左图），系统默认的通信伙伴为同一项目中的 CPU 315-2DP，在“连接”区的“类型”选择框中，默认的连接类型为 S7 连接。

点击“确定”按钮，确认默认的选项，出现“属性 – S7 连接”对话框（见图 14-45 的右图）。在调用通信 SFB 时，将会用到“块参数”区内的“ID”（本地连接标识符）。

组态好连接后，点击工具栏上的按钮，编译并保存网络组态信息。

可以看到连接表中只有“本地 ID”的值（见图 14-44），没有“伙伴 ID”的值。因为是单向连接，只需将连接表信息下载到本地的 CPU。

在连接属性对话框的“本地连接端点”区，可以看到复选框“单向”被自动选中，且为灰色，不能更改。点击“地址详细信息”按钮，可以查看地址的详细信息。

选中图 14-44 中 3 号站的 CPU 所在的小方框，因为是单向连接，连接表中没有连接信息。

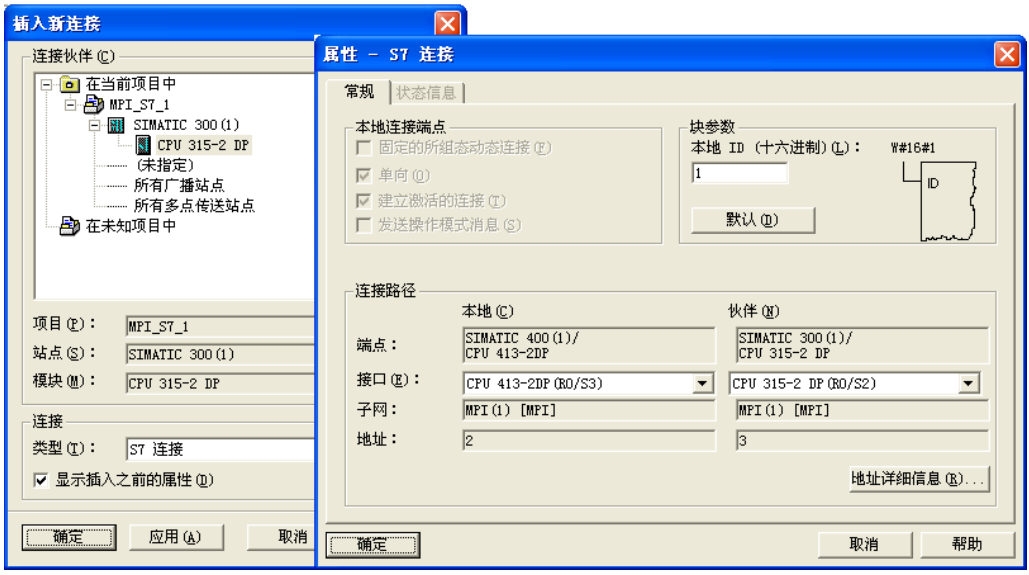


图 14–45　建立新的连接

2．设置时钟存储器字节

系统功能块 GET、PUT 在通信请求信号 REQ 的上升沿时激活数据传输，为了实现周期性的数据传输，用时钟存储器位提供的时钟脉冲作 REQ 信号。

选中 SIMATIC 管理器中的 SIMATIC 400 站点，双击右边窗口中的“硬件”图标，打开 HW Config。双击机架中 CPU 413-2DP 所在的行，在出现的 CPU 属性对话框的“周期/时钟存储器”选项卡中（见图 4–5），点击复选框“时钟存储器”，设置用于时钟存储器的存储器字节为 MB8。MB8 的第一位 M8.1 的周期为 200ms（ON 100ms，OFF 100ms）。

3．单向 S7 通信的编程

在 CPU 413-2DP 的 OB1 中，分别用 M8.1 和将它取反后得到的 M10.0 作为 SFB GET 和 PUT 的通信请求信号 REQ。S7-400 的 SFB PUT/GET 最多可以读、写 4 个地址区中的数据。S7-400 的程序与 4.2 节中的项目 PB_S7_A 的基本上相同。通过 CPU 413-2DP 读、写 CPU 315-2DP 中的数据，实现了用两个站的 ID0 分别控制对方的 QD0 或 QD4。

在 S7-400 每 100ms 循环执行一次的组织块 OB35 中，将 DB1.DBW0 和 DB3.DBW0 分别加 1 和加 2。CPU 315-2DP 的 OB35 的程序与 CPU 413-2DP 的基本上相同。CPU 315-2DP 的 OB1 没有编程。

在 CPU 413-2DP 的 OB100 中，将存放待发送数据的 DB 1、DB 3 和 MW40～MW58 分别预置为 16#4131、16#4133 和 16#4134，将 DB 2、DB 4 和 MW20～MW38 存放接收数据的地址区清零。

在 CPU 315-2DP 的 OB100 中，将存放待发送数据的 DB 1、DB 3 和 MW40～MW58 分别预置为 16#3151、16#3153 和 16#3154，将 DB 2、DB 4 和 MW20～MW38 存放接收数据的地址区清零。

4．通信的监控

将程序和系统数据分别下载到两块 CPU，用电缆连接两块 CPU 和计算机的 MPI 接口，

同时打开两个站的变量表，在屏幕上同时显示两个变量表中的动态数据。

图 14-46 和图 14-47 是在运行时复制的变量表，只监视了各接收地址区的第一个字和最后一个字。在运行时观察到 DB2.DBW0 和 DB4.DBW0（对应于发送方的 DB1.DBW0 和 DB3.DBW0）接收到的数值不断变化。可以用一个站的 ID0 控制另一个站的 QD0 或 QD4。

图 14-46　2 号站的变量表

图 14-47　3 号站的变量表

14.5.2　使用 USEND/URCV 的双向 S7 通信

只有在 S7-400 之间，才能通过集成的 MPI 接口进行 S7 双向通信。调用 SFB USEND/URCV 可以进行快速的、不可靠的数据传送，例如可以用于事件消息和报警消息的传送。

1．项目的生成与组态

在 STEP 7 中创建一个项目（见随书光盘中的例程 MPI_S7_B），生成两个站，CPU 模块均为 CPU 413-2DP，点击 SIMATIC 管理器中的按钮，打开网络组态工具 NetPro，将两个站连接到 MPI 网络上，设置它们的 MPI 站地址分别为 2 和 3（见图 14-48）。

选中 2 号站的 CPU 413-2DP 所在的小方框，在 NetPro 下面的窗口出现连接表。双击连接表的第一行，在出现的“插入新连接”对话框中，系统默认的通信伙伴为同一项目中的另一块 CPU 413-2DP，采用默认的连接类型（S7 连接）。

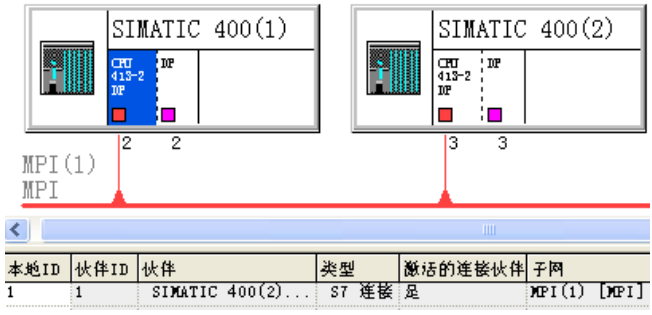


图 14-48　网络与 S7 连接组态

点击“确定”按钮，出现 S7 连接属性对话框（见图 14-49）。复选框“建立激活的连接”是默认的选择。“本地 ID”（本地标识符）的值将会在调用通信块时用到。

因为通信的双方都是 S7-400，STEP 7 自动创建了一个双向连接，在连接表中生成了

“本地 ID”和“伙伴 ID”。选中图 14-48 中 3 号站的 CPU 413-2DP 所在的小方框，可以看到下面窗口的连接表中第一行自动生成的连接参数（见图 14-50）。因为 2 号站和 3 号站是通信伙伴，它们的连接表中的 ID 相同。应将通信双方的连接表信息分别下载到各自的 CPU。

图 14-49　S7 连接属性对话框

本地ID	伙伴ID	伙伴	类型	激活的连接伙伴	子网
1	1	SIMATIC 400(1) / CPU 413-2 DP	S7 连接	否	MPI(1) [MPI]

图 14-50　SIMATIC 400（2）的连接表

2．程序设计

编程时应使用图 14-49 中的 S7 连接的 ID 号。在同一个连接中多次调用通信块时，SFB 中的 R_ID 用于区分同一连接中不同的 SFB/FB 调用，发送方与接收方的 R_ID 应相同。

本项目与 4.3 节的项目 PB_S7_B 的程序基本上相同。

2 号站和 3 号站的 OB1 的程序基本上相同，其区别在于 2 号站调用的 USEND 和 URCV 的 R_ID 分别为 1 和 2，3 号站调用的 USEND 和 URCV 的 R_ID 分别为 2 和 1。

在通信双方每 100ms 循环执行一次的组织块 OB35 中，将需要发送的 DB2.DBW0 加 1 或加 2。在 2 号站的 OB100，将存放待发送数据的 DB 1 中的 10 个字和 MW40～MW58 分别预置为 16#4011 和 16#4014。将存放接收到的数据的 DB 2 中的 10 个字和 MW20～MW38 清零。

3 号站的 OB100 中的程序与 2 号站的基本上相同，区别仅在于 OB100 预置的数据为 16#4021 和 16#4024。

3．通信的监控

将程序和系统数据分别下载到两块 CPU，用电缆连接两块 CPU 和计算机的 MPI 接口，

同时打开两个站的变量表，在屏幕上同时显示两个变量表中的动态数据。图 14-51 和图 14-52 是在运行时复制的变量表。

变量表中的 M0.0 是 URCV 的接收请求信号（EN_R），刚进入 RUN 模式时，M0.0 为 0 状态，禁止接收。双方的 DB 2、MW20～MW38 和 QD0 等接收数据区中的数据均为 0。

在变量表中 M0.0 所在的行的“修改数值”列输入“1”，点击鼠标确认后变为“true”，点击工具栏上的（激活修改变量）按钮，新的值被写入 CPU，并在“状态值”列显示出来。在时钟脉冲 M8.0 的上升沿，每 100ms USEND 发送一次数据。M0.0 为“true”时才能接收对方发送的数据，双方的 DB2.DBW0 不断动态变化。用外接的小开关改变 ID0 的值，通信伙伴的 QD0 的值随之而变。

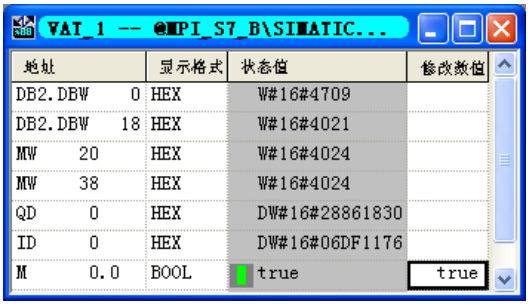


图 14-51　2 号站的变量表

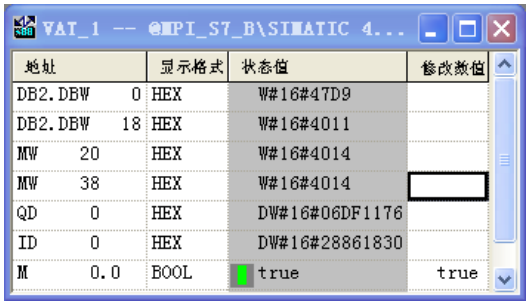


图 14-52　3 号站的变量表

14.5.3　使用 BSEND/BRCV 的双向 S7 通信

使用 SFB BSEND/BRCV，可以进行快速、可靠的数据传送。BSEND/BRCV 不能用于 S7-300 集成的 MPI 接口的 S7 通信。在 STEP 7 中创建一个项目（见随书光盘中的例程 MPI_S7_C），生成两个站，CPU 模块均为 CPU 413-2DP，点击 SIMATIC 管理器中的按钮，打开网络组态工具 NetPro，将两个站连接在 MPI 网络上，设置它们的 MPI 站地址分别为 2 和 3。S7 连接的组态方法与例程 MPI_S7_B 相同。本项目与 4.3.2 节中的项目 PB_S7_C 的程序基本上相同。

在 CPU 的属性对话框设置 MB8 为时钟存储器字节，M8.0 的时钟周期为 100ms。

在 2 号站的初始化程序 OB100 中，预置 BRCV 接收的字节数，将 DB 1 中要发送的数据字预置为 W#16#4131，将 DB 2 存放接收数据的数据区清零。3 号站的 OB100 的程序与 2 号站的基本上相同。区别在于 DB 1 的各个字被预置为 W#16#4132。在通信双方每 100ms 循环执行一次的组织块 OB35 中，将待发送的 DB2.DBW0 加 1 或加 2。

将程序和系统数据分别下载到两块 CPU 后，用电缆连接两块 CPU 和计算机的 MPI 接口，同时打开两个站的变量表，在屏幕上同时显示两个变量表中的动态数据。

刚进入 RUN 模式时，M0.0 为 0 状态，禁止接收，双方的 DB 2 和 QD0（接收数据区）的数据均为 0。M0.0 为 1 状态时允许接收，在时钟脉冲 M8.0 的上升沿，每 100ms BSEND 发送一次数据。

图 14-53 和图 14-54 是在运行时复制的变量表。运行时可以看到双方的 DB2.DBW0 不断动态变化。用接在模块输入端的小开关改变 ID0 的值，通信伙伴的 QD0 的值随之而变。

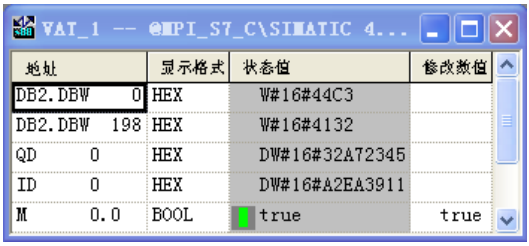


图 14-53　2 号站的变量表

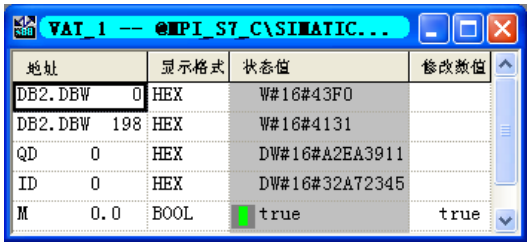


图 14-54　3 号站的变量表

14.5.4　S7 通信的 SFB 综合应用例程

1．项目结构

在 STEP 7 中创建一个项目（见随书光盘中的例程 MpiS7mul），生成两个站，CPU 模块均为 CPU 413-2（见图 14-55）。点击 SIMATIC 管理器中的按钮，打开网络组态工具 NetPro，将两个站连接到 MPI 网络上，设置它们的 MPI 站地址分别为 2 和 3（见图 14-56）。选中 2 号站的 CPU，双击下面的连接表的第 1 行，生成一个双向的 S7 连接，连接 ID 为 1。

图 14-55　SIMATIC 管理器

该项目中名为 SIMATIC 400-SEND 的站，在 OB1 中调用 SFB 8“USEND”和 SFB 12“BSEND”，向名为 SIMATIC 400-RCV 的 3 号站发送数据，调用 SFB 14“GET”和 SFB 15“PUT”读、写 3 号站中的数据。该项目中名为 SIMATIC 400-RCV 的站，在 OB1 中调用 SFB 9“URCV”和 SFB 13“BRCV”接收数据。

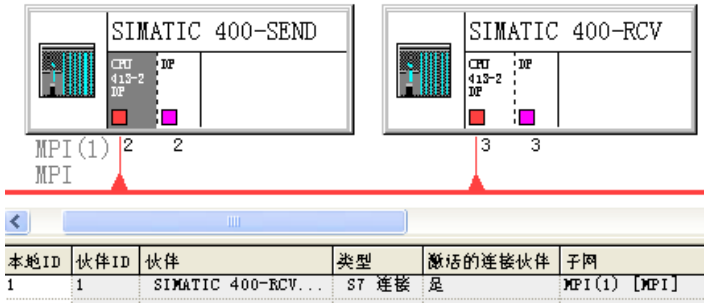


图 14-56　S7 连接组态

2．S7 通信编程中需要考虑的一些问题

因为两台 CPU 之间只建立了一条 S7 连接，所有被调用的通信 SFB 均使用同一个连接 ID，其值为 1。USEND 和 URCV 的 R_ID（发送与接收请求号）为 2，BSEND 和 BRCV 的 R_ID 为 3。PUT 和 GET 不需要设置 R_ID。

在名为 SIMATIC 400-SEND 的 2 号站的 CPU 属性视图的“周期/时钟存储器”选项卡中，设置 MB8 为时钟存储器字节。用周期为 200ms 的两个反相的信号 M8.1 和 M9.0 作为 PUT 和 GET 的通信请求信号 REQ，在这两个信号的上升沿，分别读写通信伙伴的数据区。

在 USEND 和 BSEND 的发送请求信号 REQ 的上升沿发送数据，接收站的 URCV 和 BRCV 的接收使能信号 EN_R 信号一直为 1（true），允许接收数据。

3．发送站的 OB1

下面是发送站的 OB1 中的程序，M8.1 的时钟周期为 200ms。

程序段 1：使 M9.0 与 M8.1 反相

```
AN    M    8.1
=     M    9.0
```

程序段 2：读取对方的数据区

```
CALL   "GET" , DB14                    //调用 SFC 14
  REQ          :=M8.1                  //每 200ms读取一次
  ID           :=W#16#1                //S7 连接号
  NDR          :=M0.1                  //每次读取完成产生一个脉冲
  ERROR        :=M0.2                  //错误标志位
  STATUS       :=MW2                   //通信状态字
  ADDR_1       :=P#DB1.DBX0.0 BYTE 20  //要读取的通信伙伴的 1 号地址区
  ADDR_2       :=P#DB3.DBX0.0 BYTE 20  //要读取的通信伙伴的 2 号地址区
  ADDR_3       :=ID0                   //读取通信伙伴的ID0
  ADDR_4       :=P#M 40.0 BYTE 20      //要读取的通信伙伴的 4 号地址区
  RD_1         :=P#DB2.DBX0.0 BYTE 20  //本站存放读取的数据的 1 号数据区
  RD_2         :=P#DB4.DBX0.0 BYTE 20  //本站存放读取的数据的 2 号数据区
  RD_3         :=QD0                   //用通信伙伴的ID0 控制本站的QD0
  RD_4         :=P#M 20.0 BYTE 20      //本站存放读取的数据的 4 号数据区
```

程序段 3：写对方的数据区

```
CALL   "PUT" , DB15                    //调用 SFC 15
  REQ          :=M9.0                  //每 200ms写一次
  ID           :=W#16#1                //S7 连接号
  DONE         :=M9.1                  //每次写完成产生一个脉冲
  ERROR        :=M9.2                  //错误标志位
  STATUS       :=MW10                  //通信状态字
  ADDR_1       :=P#DB2.DBX0.0 BYTE 20  //通信伙伴要写入数据的 1 号地址区
  ADDR_2       :=P#DB4.DBX0.0 BYTE 20  //通信伙伴要写入数据的 2 号地址区
  ADDR_3       :=QW0                   //将本站的IW0 写入通信伙伴的QW0
  ADDR_4       :=P#M 20.0 BYTE 20      //通信伙伴要写入数据的 4 号地址区
  SD_1         :=P#DB1.DBX0.0 BYTE 20  //存放本站要发送的数据的 1 号地址区
  SD_2         :=P#DB3.DBX0.0 BYTE 20  //存放本站要发送的数据的 2 号地址区
  SD_3         :=IW0                   //用本站的IW0 控制通信伙伴的QW0
  SD_4         :=P#M 40.0 BYTE 20      //存放本站要发送的数据的 4 号地址区
```

程序段 4：用 USEND 发送数据

```
CALL   "USEND" , DB8                    //调用 SFB 8
  REQ          :=M4.0                   //上升沿时发送
  ID           :=W#16#1                 //S7 连接号
  R_ID         :=DW#16#2                //发送与接收请求号
  DONE         :=M4.1                   //任务被正确执行完为 1
  ERROR        :=M4.2                   //错误标志位，为 1 时出错
  STATUS       :=MW6                    //通信状态字
  SD_1         :=P#DB5.DBX0.0 BYTE 20   //存放要发送的数据的 1 号地址区
  SD_2         :=IB2                    //用本站的IB2 控制 3 号站的QB2
  SD_3         :=
  SD_4         :=
```

程序段 5：用 BSEND 发送数据

```
L       IB      3
T       DB6.DBB     0                   //用本站的 IB3 控制 3 号站的 QB3
CALL   "BSEND" , DB12                   //调用 SFB 12
  REQ          :=M14.0                  //通信请求，上升沿时激活数据发送
  R            :=                       //上升沿时中止正在进行的数据发送
  ID           :=W#16#1                 //S7 连接号
  R_ID         :=DW#16#3                //发送与接收请求号
  DONE         :=M14.2                  //任务被正确执行完为 1
  ERROR        :=M14.3                  //错误标志位，为 1 时出错
  STATUS       :=MW16                   //通信状态字
  SD_1         :=P#DB6.DBX0.0 BYTE 200  //存放要发送的数据的地址区
  LEN          :=MW18                   //要发送的数据区的字节长度
```

4．发送站的 OB100

在 OB100 中预置 BRCV 接收的字节数，分别将 DB 1、DB 3、MB40～MB59、DB 5 和 DB 6 中要发送的数据字分别预置为 W#16#4011 和 W#16#4013～W#16#4016，将 DB 2、DB 4 和 MB20～MB39 存放读取的数据的数据区清零。

5．接收站的 OB1

程序段 1：

```
CALL   "URCV" , DB9                     //调用 SFB 9
  EN_R         :=TRUE                   //接收请求，为 1 时接收
  ID           :=W#16#1                 //S7 连接号
  R_ID         :=DW#16#2                //发送与接收请求号
  NDR          :=M1.1                   //任务被正确执行时为 1
  ERROR        :=M1.2                   //发送错误标志位，通信出错时为 1
  STATUS       :=MW6                    //通信状态字
  RD_1         :=P#DB5.DBX0.0 BYTE 20   //存放接收的数据的 1 号地址区
  RD_2         :=QB2                    //用 2 号站的IB2 控制本站的QB2
  RD_3         :=
  RD_4         :=
```

程序段 2：

```
CALL   "BRCV" , DB13                    //调用 SFB 13
```

```
EN_R        :=TRUE                  //为 1 状态时允许数据接收
ID          :=W#16#1                //S7 连接号
R_ID        :=DW#16#3               //发送与接收请求号
NDR         :=M0.1                  //任务被正确执行完为 1
ERROR       :=M0.2                  //接收错误标志位，为 1 时出错
STATUS      :=MW2                   //通信状态字
RD_1        :=P#DB6.DBX0.0 BYTE 200 //存放接收的数据的地址区
LEN         :=MW4                   //接收的数据字节数
L    DB6.DBB    0
T    QB         3                   //用 2 号站的 IB3 控制本站的 QB3
```

6．接收站的 OB100 中的程序

在 OB100 中，预置 BRCV 接收的字节数，分别将 DB 1、DB 3 和 MB40～MB59 存放发送站要读取的数据的地址区预置为 W#16#4021、W#16#4023 和 W#16#4024，将 DB 2、DB 4、DB 5、DB 6 和 MB20～MB39 存放被写入和接收到的数据的数据区清零。

7．OB35 中的程序

为了观察 SFB PUT 和 GET 连续传输数据的动态效果，在通信双方执行周期为 100ms 的组织块 OB35 中，将 DB1.DBW0 和 DB3.DBW0 增加不同的增量（1～4）。在变量表中监控接收到的对应的 DB2.DBW0 和 DB4.DBW0 是否动态变化。

8．通信程序的运行与监控

将程序和系统数据分别下载到两台 CPU 后，用电缆连接两台 CPU 和计算机的 MPI 接口，同时打开两个站的变量表，可以看到变量表中动态变化的数据。

发送站用 SFC“GET”读取接收站的 DB 1、DB 3、ID0 和 MB40～MB59 中的数据，并将它们保存在本站的 DB 2、DB 4、QD0 和 MB20～MB39 中。

发送站用 SFC“PUT”将本机的 DB 1、DB 3、IW0 和 MB40～MB59 中的数据，写入接收站的 DB 2、DB 4、QW0 和 MB20～MB39 中。

发送站用 USEND 将 DB 5 和 IB2 中的数据发送给接收站的 DB 5 和 QB2，用 BSEND 将 DB 6 和 IB3 中的数据发送给接收站的 DB 6 和 QB3。用发送站的变量表（见图 14-57）和接收站的变量表（见图 14-58）监控接收到的 DB 2、DB 4 和 MB20～MB39 的起始字和结束字。此外通信双方的变量表还监控 ID0 和 QD0。

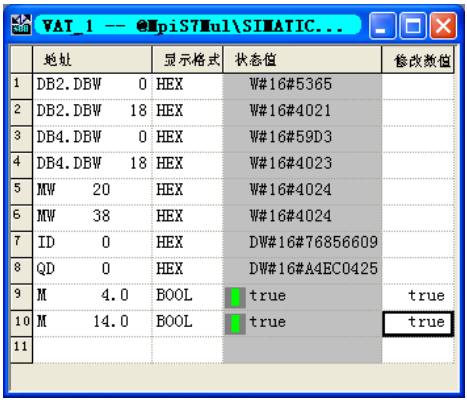


图 14-57　发送站的变量表

图 14-58　接收站的变量表

因为发送站周期性地调用 GET 和 PUT，在两个变量表中可以看到 DB2.DBW0 和 DB4.DBW0 的值不断增大。还可以看到 DB2.DBW18、DB4.DBW18、MW20 和 MW38 在对方的 OB100 中被预置的值。

发送站的 M4.0 和 M14.0 分别是 USEND 和 BSEND 的发送请求信号。需要用变量表将它们的值设置为 1（true），在它们的上升沿发送数据。USEND 和 BSEND 发送数据后，可以根据接收站的 QB2 和 QB3 是否与发送站的 IB2 和 IB3 相同，来检查数据传输是否成功。

14.6 PRODAVE 通信软件的应用

1．PRODAVE 简介

PLC 具有极高的可靠性，一般用来执行现场的控制任务，但是它的人机接口功能较差。PLC 与个人计算机（PC）通过通信连接起来，用 PC 作为上位计算机，实现系统的监控、人机接口，以及与上一级网络（例如工业以太网）的通信等功能，可以使二者的优势互补，组成一个功能强、可靠性高、成本低的控制系统。因此在工业控制系统中，PC 与 PLC 之间的通信是最常见和最重要的通信之一。

实现计算机与 PLC 通信最简便的方法是使用计算机上运行的组态软件，例如西门子公司的 WinCC 或国产的组态软件。组态软件与现场设备（例如 PLC）之间的通信程序是由组态软件生产厂家开发的，用户只需要设置一些通信参数，就可以实现上位计算机与现场设备之间的通信，通信的双方都不需要编写通信程序。但是每个系统都需要购买组态软件，费用较高。有的组态软件不能满足用户的某些特殊要求。

S7-300/400 的 MPI（多点接口）和 S7-200 的 PPI（点对点接口）用于西门子公司控制产品之间的通信，例如安装在 PC 上的 STEP 7 编程软件与 PLC 之间的通信，但是这些通信协议均未公开。

PRODAVE 是用于 PC 与 S7 系列 PLC 之间的数据链接通信的工具箱，可以用于 S7-200、S7-300/400、M7 和 C7 等西门子 PLC。通过下列硬件，可以在 S7 PLC 与 PC 之间方便地建立数据链接（见图 14–59）：

1）用于 PC 的 MPI 通信处理器，例如 CP5511、CP5611 和 CP5613，通信速率最高 12 Mbit/s。

2）用于 S7-300/400 的 PC/MPI 适配器或 USB/MPI 适配器。

3）用于 S7-200 的 PC/PPI 编程电缆。

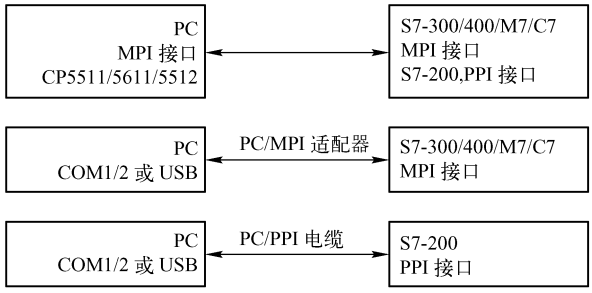


图 14–59　PC 与 PLC 的连接方式

用 PRODAVE 来实现上位计算机与 S7 系列 PLC 的通信是很方便的。PRODAVE 的动态链接库（DLL）提供了大量的基于 Windows 操作系统的 DDL 函数，供用户解决 PLC 与 PC 之间的数据交换和数据处理问题。可以在 VB 或 VC 等编程环境中调用这些函数，来建立或断开通信连接，读写 CPU 的系统存储器，方便地实现计算机与 S7-300/400 的点对点通信。

PRODAVE 有以下特点：

1）使用简单方便，编程人员不需要熟悉复杂的通信协议，通过调用 PRODAVE 提供的动态链接库（DLL）中的函数就可以实现通信。

2）上位计算机用通信函数直接读写 PLC 中的数据，不用编写 PLC 的通信程序。

3）如果使用 PC/MPI 适配器或用于 PC 的通信处理器作通信接口，它们同时还可以兼作编程软件与 PLC 的通信接口。

PRODAVE 的使用方法可以参阅随书光盘中的有关用户手册。

PRODAVE 的函数分为基本函数、数据处理函数和电话服务函数（TeleService Functions）。

基本函数用于建立、断开和激活 PC 与 PLC 的连接，以及读、写 PLC 中的各种数据。数据处理函数用于 PC 中用户数据的转换和处理，与 PC 和 PLC 之间的通信没有直接的关系。电话服务函数用于 PC 通过电话线与 PLC 建立连接，本节主要介绍基本函数。

2．建立与断开连接的函数

1）load_tool 用于建立上位机与 PLC 的连接，初始化适配器，检查是否安装了驱动程序，初始化参数地址，激活选中的通信接口。

2）unload_tool 用于在通信结束时断开 PC 与 PLC 的连接，否则可能引起上位机死机，或者造成上位机系统的异常状况。

3）new_ss 用来激活 PLC 与 PG/PC 的连接，也可以用它来重新建立已经关闭的连接。如果只有一个连接，不必使用 new_ss 函数。

3．读取 PLC 字节的函数

函数 X_field_read 用于读取 PLC 的 X 地址区中从地址 no 开始的 amount 个字节的数据，将它们存放在 PC 的数组变量 value 中。其中的 X 可取 e（输入 I）、a（输出 Q）和 m（位存储器 M）。e 和 a 是德语的缩写。

4．写 PLC 字节的函数

这类函数将存放在 PC 的数组变量 value 中的数据写入 PLC 的 X 地址区从地址 no 开始的 amount 个字节中，X 可以取 a 和 m。

5．读/写数据块的函数

1）d_field_read 读取 PLC 的 db 数据块中从地址 no 开始的 amount 个字节的数据，将它们存放在 PC 的数组变量 value 中。

2）d_field_write 将存放在 PC 的数组变量 value 中的 amount 个字节的数据，写入 PLC 的 db 数据块中从地址 no 开始的区域。

3）db_read/db_write 函数中各变量的意义与 d_field_read/d_field_write 的类似，区别在于 amount 以字为单位。

6．读定时器/计数器字

X_field_read 读取从地址 no 开始的 amount 个定时器或计数器的当前值，将它们存放在 PC 的数组变量 value 中。X 可以取 t（定时器）和 z（计数器）。z 是德语的缩写。

7．写计数器字

z_field_write 将存放在 PC 的数组变量 value 中的 amount 个字的数据，写入 PLC 从地址 no 开始的的计数器区，改写的是计数器的当前值。

8．读/写混合数据

mix_read 最多可以读取 PLC 的 20 个数据，mix_write 最多可以向 PLC 写 20 个数据。需要指明每个数据的地址区类型、长度（字节或字）和地址。

9．标志状态测试

mb_bittest 检测 PLC 内地址为 no 的标志（即位存储器）字节 MB 中的第 bitno 位。返回值 value 与该位的 0/1 状态相同。

10．置位/复位标志

mb_setbit 和 mb_resetbit 分别将 PLC 中地址为 no 的 MB 的第 bitno 位置位和复位。

11．其他通信函数

ag_info 用于读取 PLC 的信息，ag_zustand 用于读取 PLC 的状态，db_buch 用于检测某数据块是否存在。

12．数据处理函数

PRODAVE 为了方便用户，在 komfort.dll 中还提供了与通信无关的数据处理函数，例如位数据与字节数据的转换函数，浮点数格式转换函数，高低字节交换函数，位测试函数和错误信息函数等。

作者曾在某水电站控制系统中将 PRODAVE 用于 S7-300 与上位计算机的通信。

14.7 练习题

1．MPI 网络可以提供哪些通信服务？

2．S7-300/400 CPU 集成的 MPI 接口有哪些通信功能？

3．全局数据通信有什么特点？

4．生成一个项目，组态 CPU 412-2DP、CPU 313C 和 CPU 315-2DP 之间的全局数据通信。

5．S7 基本通信有什么特点？

6．怎样释放动态连接资源？

7．生成一个项目，客户机（CPU 315）用 X_PUT 和 X_GET 读写服务器（CPU 313C）的数据，用双方的 ID0 控制对方的 QD0 或 QD4。

8．生成一个项目，客户机（CPU 313C）用 X_PUT 和 X_GET 读写服务器（CPU 226）的数据，用双方的 MD0 控制对方的 MD10。

9．什么 CPU 能作 MPI 网络的 S7 通信的客户机？

10．PRODAVE 适于在什么场合使用？

第 15 章　其他通信网络与通信服务

15.1　串行通信

15.1.1　串行通信概述

在工业控制系统中，某些现场的控制设备和智能仪表没有标准的现场总线接口，只有串行通信接口。它们往往使用厂家定义的非标准的通信协议，有的使用 Modbus 协议。

串行通信又称为点对点（Point to Point）通信，简称为 PtP 通信。串行通信用于 S7 PLC 和带有串行通信接口的设备（例如计算机、打印机、条形码阅读器、机器人控制系统、扫描仪等）之间传输数据。

1．西门子的串行通信协议

串行通信主要用来与非西门子设备通信。S7-300/400 的串行通信可以使用的通信协议主要有 ASCII driver、3964（R）和 RK512。它们在 7 层 OSI 参考模型中的位置如图 15-1 所示。

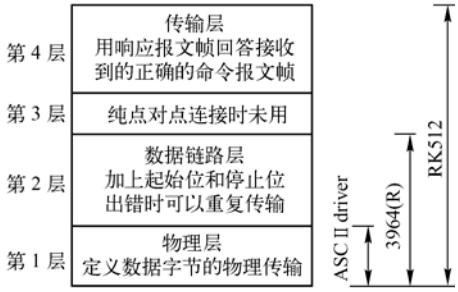


图 15-1　PtP 协议在 OSI 参考模型中的位置

ASCII driver 只使用 7 层 OSI 模型中的物理层，用于控制 CPU 和一个通信伙伴之间的串行连接的数据传输，ASCII driver 可以发送和接收开放式的数据（所有可以打印的 ASCII 字符），提供一种开放式的报文帧结构。ASCII driver 可以用结束字符、帧的长度和字符延迟时间作为报文帧结束的判据。接收方必须在组态时设置一个报文帧的结束判据。

国内极少有人使用 3964（R）和 RK512 协议，中国期刊网几乎没有有关的文章，因此本节主要介绍 ASCII driver。

2．MODBUS 协议

Modbus 是一种基于串口的通信协议，在工业控制中得到了较为广泛的使用。Modbus 协议是一个主-从协议，采用请求-响应方式，主站发出带有从站地址的请求报文，具有该地址的从站接收到后发出响应报文进行应答。Modbus 协议有 ASCII 和 RTU（远程终端单元）这两种报文传输模式。

3．串行通信处理器

（1）CPU 31xC-2PtP 集成的串行通信接口

CPU 313-2PtP 和 314C-2PtP 有一个集成的 RS422/485 通信接口，可以使用 ASCII 和 3964（R）通信协议；CPU 314C-2PtP 还可以使用 RK512 协议。其他没有集成 PtP 串口的 CPU 的串行通信需要使用 CP 340、CP 341、CP 440 和 CP 441 通信处理器模块。

（2）CP 340 通信处理器

CP 340 通信处理器是串行通信较经济的解决方案，用于 S7-300 和 ET 200M（S7 CPU 作为主站）的串行通信，它有一个通信接口，有 4 种不同的型号。一种模块的通信接口为 RS-232C，可以使用通信协议 ASCII 和 3964（R）。另外 3 种模块的通信接口分别为 RS-232C、20mA（TTY，国内很少使用）和 RS-422/RS-485，可以使用的通信协议增加了打印机驱动程序。此外，所有的串行通信 CP 模块都可以使用与变频器通信的 USS 协议。可以在随书光盘 CP 的用户手册的附录中，查阅 CP 的 RS-232C 和 RS-422/485 连接器的针脚定义，和通信接口的接线图。

（3）CP 341 通信处理器

CP 341 有一个通信接口，有 3 种不同的型号，分别采用不同的通信接口。可以使用的通信协议包括 ASCII、3964（R）、RS 512 协议。

通过安装相应的软件和插在 CP 模块上的硬件加密狗，CP 341 和 CP 441 可以使用下列客户协议：Modbus RTU 主站协议、Modbus RTU 从站协议和 Data Highway 协议。Modbus RTU 协议只能在价格较高的 CP 341 和 CP 441-2 上使用，并且需要为 Modbus RTU 协议单独付费，使用成本较高。

S7-200 集成了 Modbus RTU 主站协议和 RTU 从站协议，用 S7-200 来实现 Modbus RTU 协议要经济得多。用 MPI 接口和 X_PUT、X_GET 来实现 S7-200 和 S7-300/400 之间的通信（见 14.4 节），不需要增加额外的开支。作者编写的《PLC 编程及应用（第 3 版）》详细介绍了 S7-200 使用 Modbus RTU 主站协议和 RTU 从站协议进行通信的方法。

（4）CP 440 串行通信处理器

CP 440 的物理接口为 RS-422/RS-485。最多 32 个节点，最高传输速率为 115.2 kbit/s，通信距离最长 1200m。可以使用的通信协议为 ASCII 和 3964（R）。

（5）CP 441-1/CP 441-2 串行通信处理器

CP 441-1 可以插入一块分别带一个 20mA（TTY）、RS-232C 或 RS-422/485 接口的 IF 963 子模块。可以使用的通信协议有 ASCII、3964（R）和打印机驱动程序。

CP 441-2 可以插入两块 IF 963 子模块，可以使用的通信协议与 CP 341 相同。

15.1.2 使用 ASCII 协议发送和接收数据

1．在编程设备中安装 CP 的组态工具软件

用于点对点通信的 CP 附带的光盘中有 CP 的组态工具、带有 CP 通信功能块和功能的库和程序实例。该光盘的文件在随书光盘的文件夹“CP_PtP_CD”中，安装它以后才能对 PtP CP 组态和编程。安装好之后，在程序编辑器左边窗口的“库”文件夹中，将会出现名为“CP PtP”的文件夹（见图 15-2）。

2．生成项目

在 SIMATIC 管理器中，用新建项目向导创建一个新的项目，项目名称为“ASCII”，CPU

为 CPU 313C-2DP。本章的例程在随书光盘的文件夹“\Project\Chapter15”中。

在 HW Config 中，将电源模块、信号模块和 CP 340-RS422/485 插入机架。CP 340 在 5 号槽，模块的起始地址为 272。双击机架中的 CP 340，点击它的属性对话框中的“参数”按钮，在打开的参数设置对话框中（见图 15-3），用“Protocol”（协议）选择框选中 ASCII 协议。点击信封形状的 Protocol 图标，打开 Protocol 对话框，设置的串口参数见图 15-4，“Even”为偶校验。

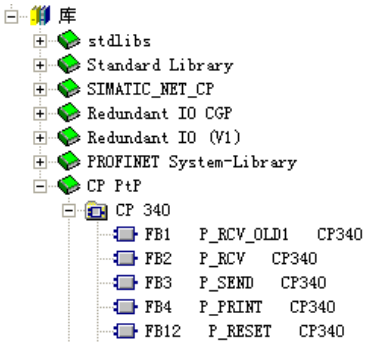


图 15-2　PtP 通信的 FB

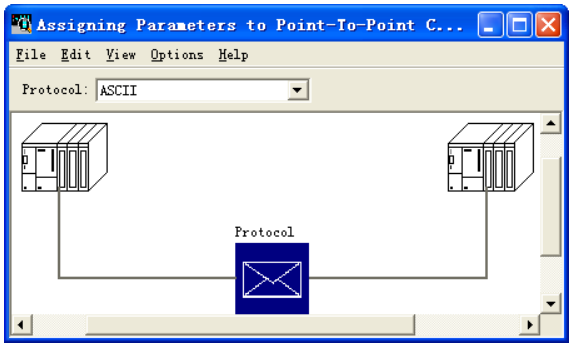


图 15-3　设置 CP 340 的参数

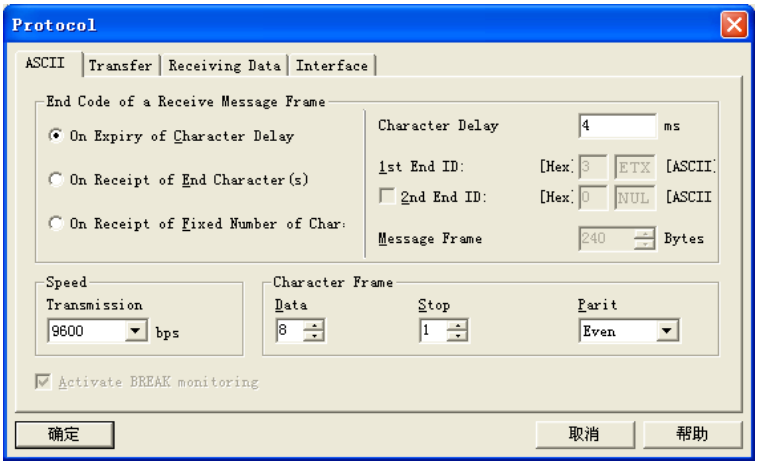


图 15-4　设置 CP 340 的协议参数

在 Interface（接口）选项卡，选中半双工的“RS 485”（见图 15-5）。其他参数均采用默认值。点击 按钮，编译并保存组态信息。

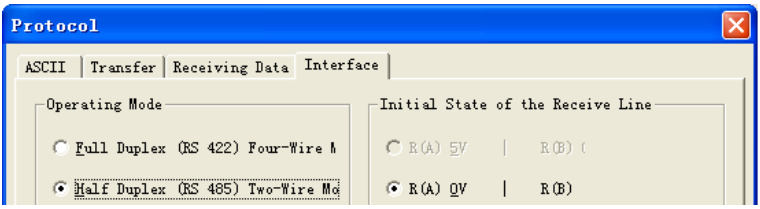


图 15-5　设置 CP 340 的接口参数

双击机架中的 CPU，打开 CPU 的属性对话框。在“周期/时钟存储器”选项卡中（见图 4-5），选中复选框“时钟存储器”，设置用于时钟存储器的存储器字节为 MB8，时钟存储器位 M8.5 的周期为 1s。

3．调用 FB 3 和 FB 2 发送和接收数据

在 SIMATIC 管理器中生成用来保存发送数据和接收数据的数据块 DB 1 和 DB 4，在两个数据块中分别生成一个数组。下面是 OB1 中的程序：

```
      CALL   "P_SEND" , DB3          //调用 FB 3
        REQ          := M8.5         //上升沿时发送数据，每秒发送一次
        R            :=              //复位信号
        LADDR        :=272           //CP 340 的起始地址
        DB_NO        :=1             //存放要发送的数据的数据块编号（DB 1）
        DBB_NO       :=0             //要发送的数据在数据块中的起始地址（DBB0）
        LEN          :=18            //要发送的数据字节数
        DONE         :=M1.1          //发送任务完成时产生一个脉冲
        ERROR        :=M1.2          //错误标志位
        STATUS       :=MW2           //错误状态字
      CALL   "P_RCV" , DB2           //调用 FB 2 接收数据
        EN_R         :=TRUE          //为 1 时允许接收
        R            :=              //复位信号
        LADDR        :=272           //CP 340 的起始地址
        DB_NO        :=4             //存放接收的数据的数据块编号（DB 4）
        DBB_NO       :=0             //接收的数据在数据块中的起始地址（DBB0）
        NDR          :=M1.5          //接收任务完成时产生一个脉冲
        ERROR        :=M1.6          //错误标志位
        LEN          :=MW4           //接收的数据的字节数
        STATUS       :=MW6           //错误状态字
      A（
      L      MW      2
      L      0
      <>I
      ）
      JNB    _001                    //MW2 非 0（发送出错）则跳转
      L      MW      4
      T      MW      8               //保存接收到的字节数
_001: NOP    0
```

在 OB100 中，用 SFC 21 将 DB 1 中的字初始化为 16#1234，将 DB 4 中的字清零。在每 100ms 执行一次的 OB 35 中，将发送的第一个字 DB1.DBW0 加 1。

4．通信测试

如果 CP 340 的接口是 RS-232C，可以用 RS-232C 电缆直接连接 CP 340 和计算机的串口。作者做实验用的 CP 340 的接口是 RS-422/RS-485，CP 340 的接口与计算机的 RS-232C 接口之间用 S7-200 的 PC/MPI 通信电缆来转接。CP 340 的 15 针 RS-422/RS-485 接口的接线图见 CP 340 用户手册的附录 B.3。

做实验时用随书光盘中的串口通信调试软件来发送数据和显示接收的数据。打开该软件后（见图 15-6），执行菜单命令“串口设置”→“串口属性”，设置计算机串口的波特率、数据位、奇偶校验位和停止位等参数，对话框下面的状态栏给出了串口的状态与设置的参数。

执行菜单命令“串口设置”→“打开/关闭串口”命令，打开串口，在“通信记录”文本框中，可以看到 CP 340 每秒一次发送给计算机的 18B 数据，接收的前两个字节（DB1.DBW0）是动态变化的。可以用“清空”按钮清除“通信记录”文本框中的数据。

图 15-6　串口通信调试对话框

在“发送帧”文本框输入要发送的数据后，点击“发送”按钮，数据被发送到 PLC。可以选择用字符串、十进制字节或十六进制字节这 3 种数据格式输入要发送的数据。十进制字节或十六进制字节数据之间用空格隔开，各数据必须在一个字节允许的范围之内。

15.2　S7 路由功能

15.2.1　PG/PC 的 S7 路由功能

1．路由的概念

“S7 路由”是指在西门子工业通信网络中，跨越两个或多个子网（MPI、PROFIBUS 和工业以太网）进行网络访问。S7 路由属于 PG/OP（编程设备/操作员面板）通信服务功能，通过它可以实现跨网络的 PG/OP 通信，例如，用编程计算机的普通以太网卡，访问 PROFIBUS 网络和 MPI 网络中的设备。PG 可以在某个固定点（一般是主控室）访问所有在 S7 项目中组态的 S7 站点，下载用户程序和硬件组态，或者执行测试和诊断功能。还可以用 S7 路由功能实现跨网络的 HMI 与 PLC 的通信。

凡是涉及路由功能，都需要一个或者多个网关（S7 CPU 或通信处理器）。网关是指一个跨接在两个网段上，并且可以实现两个网段之间的数据交换的设备。在 S7 路由中，网关由多个接口连接到两个或更多的子网，这些子网可以使用相同的或不同的通信协议。

图 15-7 中间的 S7-300 站就是一个网关，它的以太网接口和 MPI 接口使该站跨接在以太网和 MPI 网络上。为了使用路由功能，需要组态 PG/PC 站。以太网上的 PG/PC 站可以通过

S7-300 的路由功能访问 MPI 网络上的 S7-400 站。

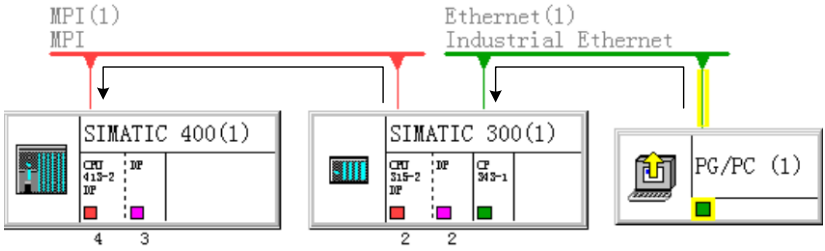


图 15-7　PG/PC 的 S7 路由功能

STEP 7 在组态网络期间，自动地为网关生成特定的“路由表”。路由表是特殊的系统数据，必须下载到网关。此后，编程设备可以通过网关搜索到指定的 PLC 的通信路径。

为了实现路由功能，应在同一个项目中组态网络，CPU 应下载该项目完整的网络组态信息。随书光盘中的文件《S7-300_400 路由功能》列出了支持路由的设备的型号和订货号。

2. PLC 硬件组态

在 SIMATIC 管理器中创建一个新的项目，项目名称为“IE_MPI”（见随书光盘中的同名例程）。在 HW Config 中，将 CPU 315-2DP、电源模块、信号模块和 CP 343-1 插入机架。双击 CP 343-1，在出现的 CP 属性对话框中，点击“常规”选项卡中的“属性”按钮，在打开的以太网接口属性对话框的“参数”选项卡中（见图 15-8），采用 CP 默认的 IP 地址 192.168.0.1 和子网掩码 255.255.255.0，用单选框选中“使用路由器”，CP 和路由器的 IP 地址应在同一个网段内，但是它们的 IP 地址不能相同。点击“新建”按钮，生成一条名为“Ethernet（1）”的以太网，将 CP 343-1 连接到以太网上。点击“确定”按钮，返回 HW Config。

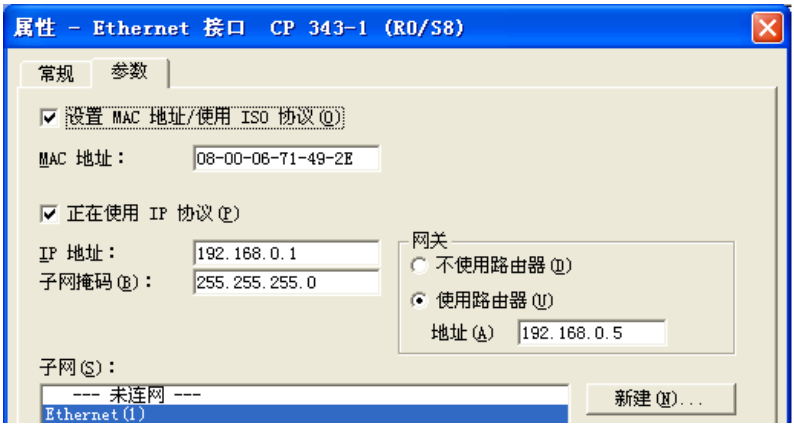


图 15-8　组态 CP 343-1 的以太网接口

CPU 315-2DP 的 DP 地址和 MPI 地址均为 2，CP 343-1 的 MPI 地址为 3。

组态 CP 343-1 的以太网接口时，一定要选中“使用路由器”。如果选中“不使用路由器”，运行时 CP 343-1 的 SF LED 亮。此时双击 HW Config 中的 CP 343-1，打开它的属性对话框。点击“诊断”选项卡中的“运行”按钮，在诊断缓冲区可以看到事件“未找到路由表”。

在 SIMATIC 管理器中生成一个 S7-400 站点。在 HW Config 中，将电源模块、CPU 413-2DP 和信号模块插入机架。CPU 模块的 DP 地址和 MPI 地址如图 15-7 所示。组态结束后，点击工具栏上的按钮，编译并保存组态信息，用 MPI 接口将组态信息下载到各 CPU。

3. 组态 PG/PC

用鼠标右键点击 SIMATIC 管理器中的项目，执行出现的快捷菜单中的命令“插入新对象”→“PG/PC”。双击右边窗口新生成的 PG/PC，打开它的属性对话框。点击“接口”选项卡中的“新建”按钮（见图 15-9），选中出现的“新建接口 – 类型选择”对话框中的“Industrial Ethernet”，点击“确定”按钮，生成一个以太网接口。

图 15-9　生成 PG/PC 的以太网接口

选中接口列表中生成的工业以太网接口，点击“属性”按钮，打开 PG/PC 的以太网接口属性对话框（见图 15-10），设置计算机普通网卡的 MAC 地址和 IP 地址。可以用“设置 PG/PC 接口”对话框的诊断功能读取网卡的 MAC 地址。IP 地址应与“网络连接”中设置的相同（见图 10-13）。如果使用 ISO 协议，可以不设置 IP 地址。

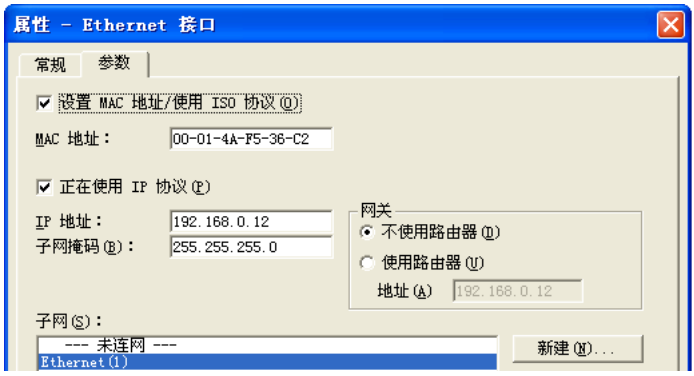


图 15-10　设置 PG/PC 的以太网接口

点击“确定”按钮，返回 PG/PC 属性对话框。打开“分配”选项卡（见图 15-11），选中“PG/PC 中的接口参数分配”列表框中使用 ISO 协议的网卡。点击“分配”按钮，将 ISO 协议“分配”给上面“组态的接口”列表框中的以太网接口。分配后 PG/PC 站连接到网络的接线两边出现黄色区域（见图 15-7）。

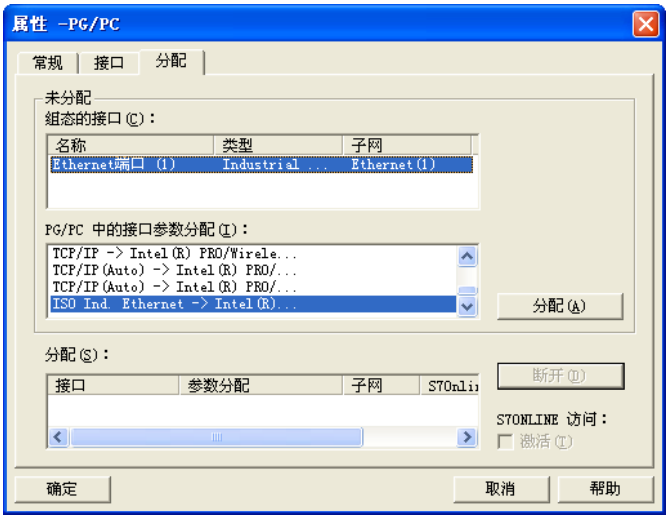


图 15-11　分配 PG/PC 的以太网接口的参数

分配成功后，“未分配”区内“组态的接口”列表框中的以太网接口消失，下面的“分配”列表框中出现以太网接口和分配的参数（见图 15-12）。选中以太网接口后，点击“断开”按钮，可以取消参数分配，返回图 15-11 的状态。

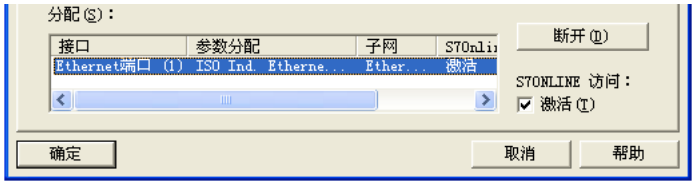


图 15-12　分配后 PG/PC 的以太网接口的参数

这种参数分配不是必要的操作，即使不执行上述分配操作，编程计算机也可以访问 MPI 网络上的 CPU 413-2DP。

4. 组态网络

点击 SIMATIC 管理器工具栏上的按钮，打开网络组态工具 NetPro（见图 15-7），可以看到，PG/PC 和 S7-300 站点已经连接到以太网上。用鼠标左键按住 CPU 中代表 MPI 接口的小红方块，将它拖放到 MPI 网络上。

组态好所有网络连接后，点击工具栏上的按钮，编译并保存组态信息。编译成功后用 MPI 接口将组态信息下载到各 CPU。选中 PG/PC，工具栏上的下载按钮为浅灰色，表示不能执行下载，组态信息不需要下载到 PG/PC。

5. 验证 S7 网关功能

用 PROFIBUS 电缆连接 CPU 315-2DP 和 CPU 413-2DP 的 MPI 接口，用以太网电缆连接计算机和 CP 343-1 的以太网接口。将 CPU 和 CP 的模式转换开关扳到 RUN 位置。待 CPU 和 CP 正常运行后（绿色的 RUN LED 亮），点击 SIMATIC 管理器工具栏上的按钮，打开在线视图（见图 15-13），CPU 和 CP 上的“运行”图标表示编程计算机可以访问以太网和 MPI 网络上的两个 CPU。

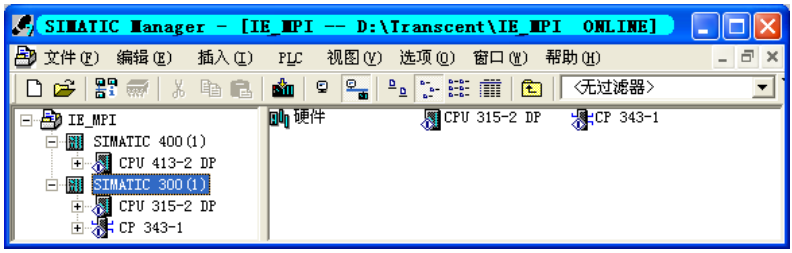


图 15-13　在线视图

在离线的 HW Config 中下载 CPU 413-2DP 的组态信息。在出现的“选择节点地址”对话框中（见图 15-14），可以看到目标站点（CPU 413-2DP）的 MPI 地址，和网关（S7 300 站点）的以太网 MAC 地址。

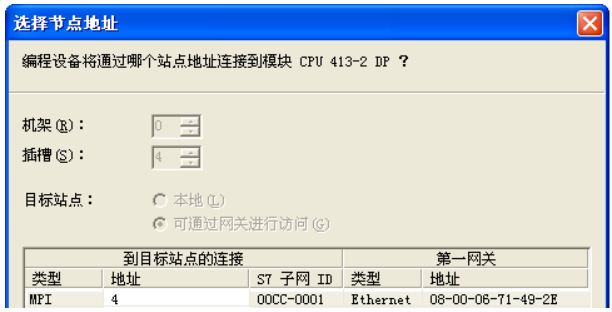


图 15-14　下载时目标站点与网关的地址

实验表明，通过路由功能，可以对以太网和 MPI 网络上的每一个站点进行下载、上载和监控操作。

15.2.2　HMI 的 S7 路由功能

1. HMI 的路由功能

S7 路由属于 PG/OP 通信服务功能，人机界面（HMI）可以通过路由功能，监控它所在的网络之外的其他网络上的 PLC。

图 15-15 中间的 CPU 315-2PN/DP 站点是一个网关，它的以太网接口和 MPI/DP 接口使该站跨接在以太网和 DP 网络上。图中的 HMI 站可以通过 CPU 315-2PN/DP 监控 DP 网络上的 CPU 315-2DP 站。

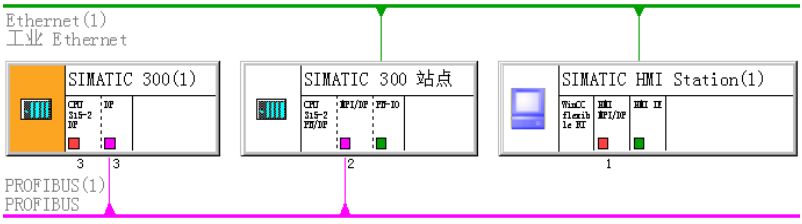


图 15-15　HMI 的 S7 路由功能

2. 生成 SIMATIC 300 站点

在 SIMATIC 管理器中，创建一个新的项目，项目名称为“HMI_Rout”（见随书光盘中

的同名例程），CPU 为 CPU 315-2DP（见图 15-16）。设置 DP 地址和 MPI 地址均为 3。

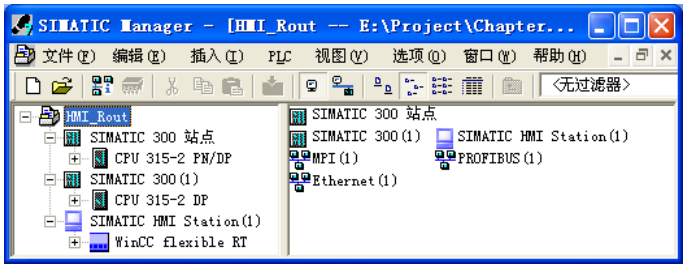


图 15-16　SIMATIC 管理器

3. 生成 HMI 站点

点击 SIMATIC 管理器左边窗口最上面的项目，执行弹出的快捷菜单中的“插入新对象”→“SIMATIC HMI Station”命令，在出现的对话框中设置 HMI 的型号为 MP 277 10" Touch，在 STEP 7 的项目中生成 SIMATIC HMI 站对象（见图 15-16）。所选的 HMI 是有一个以太网接口和一个 MPI/DP 接口的触摸屏。

4. 生成有路由功能的 SIMATIC 300 站点

用鼠标右键点击 SIMATIC 管理器左边窗口中的项目，执行出现的快捷菜单中的命令“插入新对象”→“SIMATIC 300 站点”。选中左边窗口新生成的 300 站点，双击右边窗口中的“硬件”，打开 HW Config。将硬件目录窗口中的导轨拖放到左边窗口，将 CPU 315-2PN/DP 插入 2 号槽。

5. 在 NetPro 中连接站点

点击工具栏上的按钮，打开网络组态工具 NetPro（见图 15-15），显示出尚未连网的两台 S7-300 站点和 HMI 站点。双击 HMI 站点中的以太网接口，点击出现的“HMI IE 属性”对话框的“接口”区中的“属性”按钮，打开以太网接口属性对话框（见图 15-17）。点击“新建”按钮，生成一个以太网，HMI 被连接到以太网上，采用图中默认的参数。

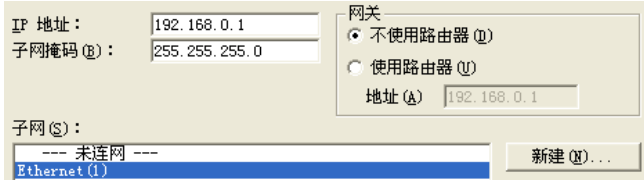


图 15-17　HMI 的以太网接口属性对话框

双击 CPU 315-2PN/DP 站点的以太网接口，点击出现的“PN-IO 属性”对话框的“接口”区中的“属性”按钮，打开以太网接口属性对话框（见图 15-18）。将该站点连接到以太网上，用单选框选中“使用路由器”，设置路由器的 IP 地址。

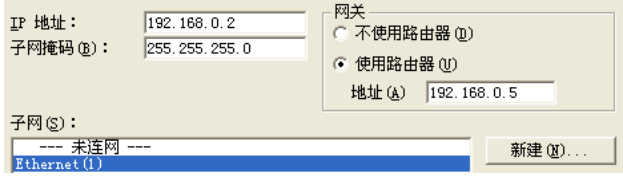


图 15-18　CPU 315-2PN/DP 的以太网接口属性对话框

双击 CPU 315-2PN/DP 站点的 MPI/DP 接口，用“接口”区的“类型”选择框将接口设置为“PROFIBUS”（见图 15-19），点击“属性”按钮，在 PROFIBUS 接口属性对话框中，点击“新建”按钮，生成一个 PROFIBUS 网络，采用默认的站地址 2。

图 15-19　CPU 315-2DP 的 MPI/DP 接口属性对话框

将 CPU 315-2DP 连接到 PROFIBUS 网络上，最后点击工具栏上的按钮，编译和保存组态信息。用 MPI 接口将组态信息下载到各 CPU。

6．打开 WinCC flexible 项目

打开图 15-16 所示SIMATIC管理器左边窗口中的HMI站点，选中“连接”图标，双击右边窗口的“连接”，打开 WinCC flexible的连接表（见图 15-20）。可以看到在STEP 7 组态的HMI与 CPU 315-2PN/DP站点通过以太网建立的“连接_1”。点击“激活的”列右边隐藏的按钮，激活该连接。

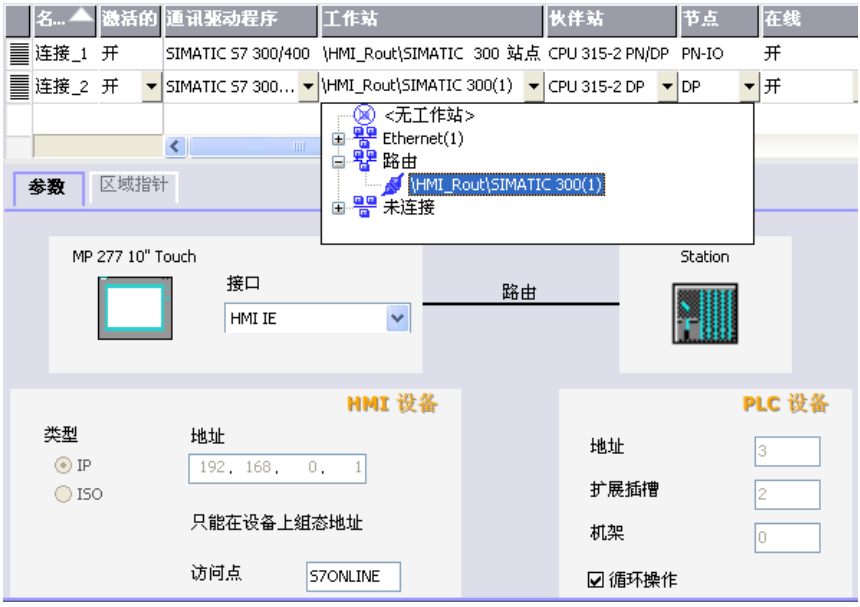


图 15-20　连接属性对话框

7．生成用于路由的连接

双击连接表的空白行，生成名为“连接_2”的新连接。点击该行的“工作站”列右边的按钮，选中出现的对话框的“路由”中的SIMATIC 300（1）站点，“伙伴站”列出现“CPU 315-2DP”。选中新生成的连接，在下面的连接属性对话框的“参数”选项卡中，可以看到HMI与CPU 315-2DP之间的连接线上的“路由”，以及在STEP 7 组态的双方的以太网接口和DP接口的参数。

8．生成 PLC 中的变量

双击图 15-21 右边窗口的项目视图中的“变量”，打开变量表编辑器。双击第一行，生成一个新的变量。点击“符号”列右边的按钮，在打开的对话框左边的窗口中，打开各层的文件夹，可以看到与连接表中两个连接对应的两个 SIMATIC 300 站点。

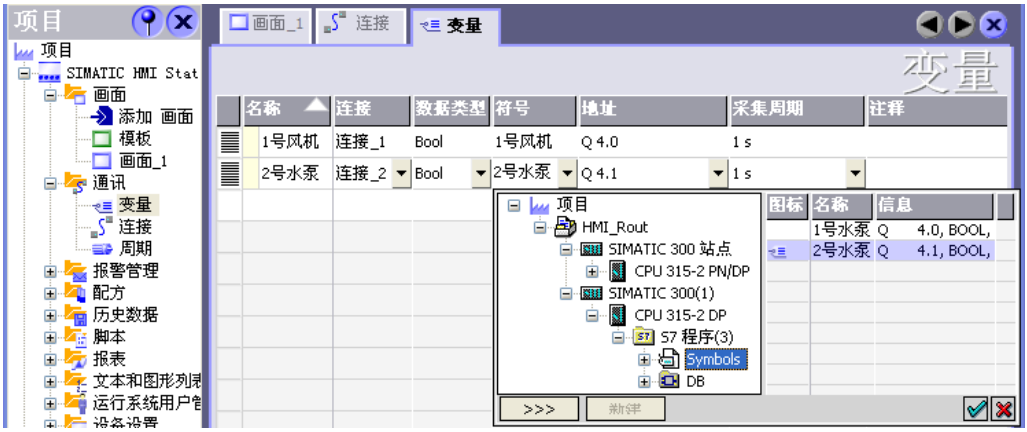


图 15-21　变量表

选中 CPU 315-2DP 站点的符号表（Symbols），点击右边窗口中的某个变量，它被传送到 HMI 的变量表中。由此可知，以太网上的 HMI 可以通过 S7 路由功能访问 DP 网络上的 CPU 315-2DP 中的变量。

15.3　其他网络与通信服务

15.3.1　工业无线局域网

1．工业无线局域网的特点

工业无线局域网（IWLAN）产品 SCALANCE W 基于 IEEE 802.11 标准。无线通信以空间电磁波的形式传输信息，是有线解决方案的有力补充，越来越多地应用到工业领域，例如自动导航车辆和单轨输送机等。

SCALANCE W 设备系列具有 IP 65 防护等级，PROFINET IO 支持工业无线局域网的无线通信。无线局域网具有以下优点：

1）简化了维修工作，减少了维修费用和停机时间。对设备的远程诊断降低了维修成本。

2）可以节省通信电缆，易于更改通信路径。

3）可以解决移动设备的旋转和移动造成的碳刷磨损和导线破裂的问题。可以与腐蚀环境

中的设备建立低成本的连接。

无线模块配备了两个天线，接收器选择接收到的两个信号中最强的一个。

如果传输链路的属性有变化，为了确保数据传输的可靠性，无线模块从最大传输速率切换到较低的传输速率。无线模块还提供 IT 功能，包括使用 SNMP 基于 Web 进行管理、电子邮件或 SMS 消息的发送功能。

2．网络结构

在 WLAN 中，接入点（Access point，见图 15-22）的作用类似于交换机。每个接入点都与其单元中的所有常规节点（即所谓的“客户机”）进行通信，不管它们是固定的还是移动的。另一方面，接入点通过电缆或通过另一个独立的无线网络保持相互之间的连接，因此可以超越无线单元的限制进行通信。W780 模块是各个无线单元的网络交换机，以及工业以太网和 WLAN 网络之间的传输媒体的接入点。

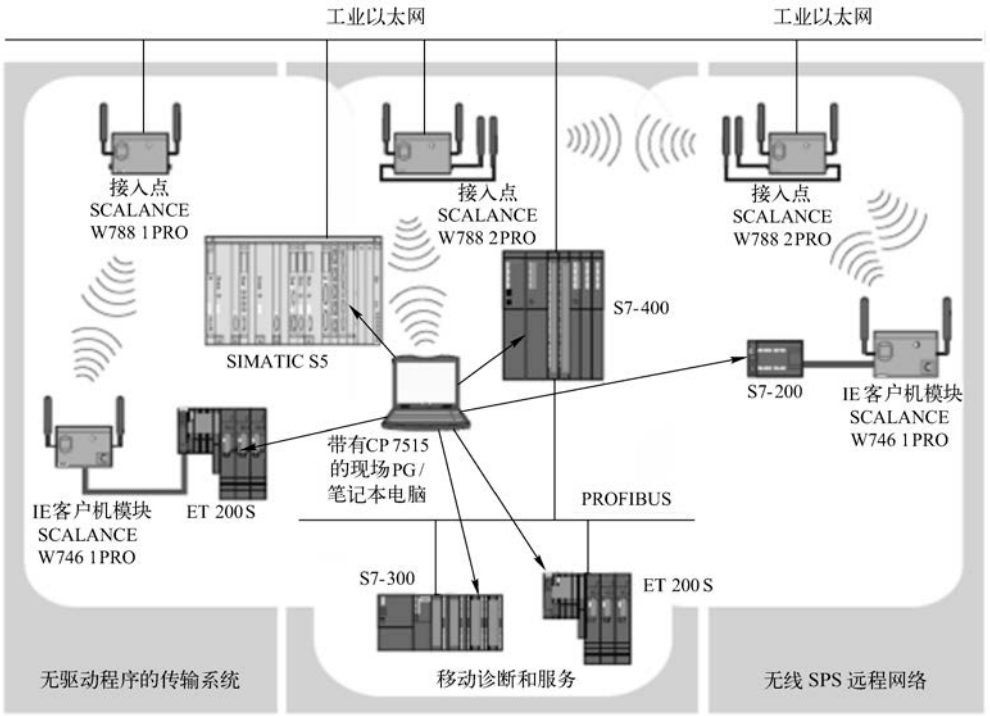


图 15-22　工业无线局域网

有两种不同的 WLAN 网络类型：

（1）基础结构模式 （InfrastrucTRUE mode）

在基础结构模式中，通过公共接入点连接节点进行通信。节点必须登录到接入点，并通过指定的通道传输。这种类型的网络称为基础服务集。

如果无线区域因感应范围太小或可操作节点太少而未达到接入点，可以在公共网络中对两个或多个重叠的基础服务集进行操作。基础结构模式支持在以太网中进行操作。符合 IEEE 802.11 标准的 WLAN 也被称为无线以太网。

（2）节点之间直接连接的自组网 （ad hoc network）

符合 IEEE 802.11 标准的最简单的 WLAN 网络称为自组网。各设备的无线网卡不需要大型网络结构，无需用户介入便可以建立快速、简单的网络。自组网用于短距离临时交换数据。

3．冗余模式

将数据通过两个单独的射频无线网卡和两个不同的通道进行传输，便可以实现冗余。

在冗余模式，必须用接入点处理两个无线接口，用两种频率发送数据。接入点相互之间不但以主频率进行通信，而且在另一组天线的第二个通道上进行通信。受到干扰时可以通过其他通道进行连接。

4．访问方法

有线以太网采用 CSMA/CD（带冲突检测的载波侦听多路访问）的访问方法。WLAN 使用的访问方式称为 CSMA/CA（带冲突避免的载波侦听多路访问）。无线 LAN 不是以物理方法侦听通道，而是使用将此通道保留一定时间的通信协议。节点开始发送数据之前，将查询媒体是否空闲。

5．IWLAN 的信息安全

因为无线电波受到反射和衍射重叠等的影响，因此 IWLAN 可能会受到干扰。此外，IWLAN 是“共享媒体”，也就是所有节点都试图访问同一个网络。而在有线“切换式以太网”中，每个节点均有单独的电缆直接连接到交换机，不必与其他节点共享。除非采取其他措施，否则将无法确定哪个节点正在阻塞无线网络并在访问媒体。IEEE 802.11i 用于处理 WLAN 数据传输的安全性，重点是无线数据传输的加密算法和身份验证过程的定义。

6．无线网关

IWLAN/PB LINK PN IO 是用于连接工业以太网和 PROFIBUS 网络的无线网关，支持访问 PROFIBUS 的所有 DP 从站。

15.3.2 广域网

1．SINAUT ST7 的组成

WAN 是地理上分布很广的广域网（Wide Area Network）的缩写。西门子的 SINAUT ST7 为广域网通信提供了强大而灵活的解决方案，其专用的硬件和软件使该系统可以在广域网内实现可靠的数据传输。它由站、节点终端以及一个或多个控制中心组成，系统的各部分用数据传输媒体连接。

（1）站和节点终端

TIM 传输模块是 SINAUT ST7 系统的核心硬件，安装了 TIM 传输模块的 S7-300/400 可以扩展为 SINAUT 站或节点终端。

一个 SINAUT 站可以通过两个接口连接广域网，从而实现通信通道的冗余。这两个接口可以连接相同或不同类型的广域网，例如一个专线网络加一个拨号网络。

（2）控制中心

有 3 种不同形式的控制中心：

1）基于 S7-300/400 的控制系统的控制中心。仅需要站上现有的过程数据的当前过程映像。通过输入命令、设定值或参数可以影响站过程控制。这种控制中心也可以用来扩展 PC 控制中心的功能，例如用操作员面板作数据输出和用作紧急操作系统。

2)基于 WinCC 的 PC 控制中心 SINAUT ST7cc。这是用于 SINAUT ST7 和老系统 SINAUT

ST1 的理想控制中心系统。它专门用于 SINAUT 系统上的事件驱动和时间标记数据传输，通过 WinCC 冗余软件包，可以设置为冗余系统。

3）SINAUT ST7sc 是通过 OPC 连接其他供应商的控制中心。SINAUT ST7 和 ST1 可以通过“数据访问接口”与其他制造商的控制中心系统连接。ST7sc 具备扩展缓冲机制，可以防止在 OPC 客户机出现故障时丢失数据。可以将 ST7sc 连接至非冗余或冗余客户机。

2．SINAUT ST7 可以使用的广域网

下列广域网（WAN）可以用于 ST7 的数据传输：

（1）传统的 WAN

可以用下列传统的 WAN 进行数据传输：专线网络（铜质电缆、光纤电缆、专用的或租用的专线）、专用无线网络、模拟电话网和数字 ISDN 电话网、移动无线网络 GSM。

（2）基于以太网的 WAN

在站和控制中心之间或各个站之间可以通过以太网进行 SINAUT 通信：

1）通过无线局域网和用于以太网的特殊无线设备（例如 SCALANCE W）进行通信。

2）通过光纤导线和具有光纤端口的 SCALANCE X 交换机，最长通信距离可达 26 km。

3）通过使用 ADSL 或 GPRS 的公共网络和互联网进行通信。

3．本地数据存储

TIM 模块可以提供多达 1MB 的存储空间，可以保存那些通信连接或通信伙伴发生故障时不能丢失的数据。提供的存储容量最多可以容纳 32000 个报文。

可以对要发送的数据分配不同的优先级。高优先级可以立即建立拨号连接，低优先级的数据最初将保存在 TIM 中，然后在适当的时候发送，所有这些通信控制功能都是由 TIM 自动实现的。用户需要做的只是在组态时配置必要的连接，以及在程序中定义要传输的数据。

由于 TIM 模块保存的是带时间标记的数据，因此要求数据的接收方必须具有处理带时间标记数据的能力，尤其是涉及到历史数据归档的时候。即使控制系统接收数据延迟了数小时甚至数天，SINAUT ST7cc 或 ST7cs 控制中心都能够确保所有的报警、事件及归档输入使用来自远程站的时间标记。

4．SINAUT ST7 的功能

（1）事件驱动的数据传输

SINAUT ST7 软件提供在 CPU 与 CPU 之间，或 CPU 与 ST7cc 控制中心之间以事件驱动方式进行数据传输。可以显示连接失败、CPU 或控制中心故障。故障排除后，数据将自动更新。

（2）时钟功能

可以通过一台 DCF77 无线时钟或 GPS 时钟接收装置，给整个网络上的 CPU 及 ST7cc 控制中心提供日期和时间，使系统具有准确的时间，并且保证了数据时间标记的一致性。

（3）远程编程和远程诊断

使用 SINAUT 的行业的终端分布区域很广，终端发生故障时，需要工作人员到达现场才能解决。为此 SINAUT ST7 可以提供通过 WAN 网络进行远程编程和远程诊断的功能。SIMATIC 和 SINAUT 为站自动化和 WAN 通信提供的所有诊断和编程功能都可以通过 WAN 执行。

（4）通过短消息发送报警信息给工作人员

CPU 可以发送事件驱动的报警短消息到工作人员的移动电话，同时，工作人员也可以发

送短消息到 CPU 来确认报警事件。

15.3.3 KNX/EIB

1．KNX/EIB 系统概述

KONNEX（KNX）是一种多主站网络，用于家庭和楼宇控制系统的标准化网络通信。它由以下几部分组成：

- EIB（European Installation Bus，欧洲安装总线）。
- EHS（European Home Systems，欧洲家庭系统）。
- BatiBUS技术。

在传统楼宇系统中，各种服务（例如供热和照明）是单独规划和实施的。每项功能（信号和电源）都需要单独的线路。在 KNX/EIB 中，所有服务都是一起规划和实施的。所有操作功能和顺序都由一条公共线路进行控制和监视。

2．KNX/EIB 系统的组成

KNX/EIB 系统（见图 15-23）由以下组件组成：

- 传感器（例如温度传感器和风速表）生成命令并将这些命令打包成报文。

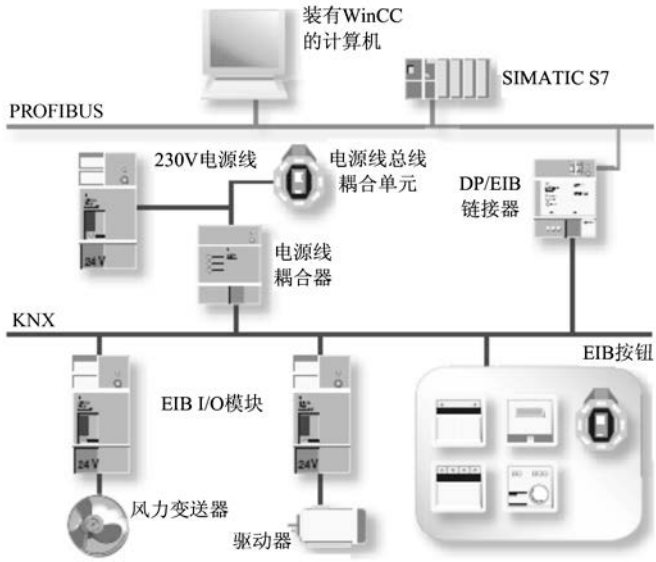


图 15-23　KNX 网络

- 执行器（例如用于照明的接触器和送风管道中的挡板）：执行接收到的报文。

总线将传感器和执行器结合起来。这些设备使用标准化的命令进行通信。使用特定的工程工具软件（ETS）对节点进行编程和组态，ETS 是一种多供应商工具，用于规划、调试和维护 KNX/EIB。ETS 可以确保不同制造商生产的组件相互兼容。

KNX/EIB 的显示功能使用户可以查看系统状态并操作远程控件。附加的程序（例如时间、中断和数据记录工具等）使 EIB 显示器成为一个用于楼宇自动化系统的完整管理站。它可以提供像远程维护、远程读出消耗数据和安全服务之类的 KNX/EIB 服务。

KNX/EIB 可以与其他服务进行无缝集成，例如与电气设备（家用电器等）联网。

KNX/EIB 最明显的特点就是减少了线路，从而带来很多优点：

1）更易于完成楼宇设备的安装，以后的扩展和修改也很容易。

2）如果要求发生变化，无需重新布线，通过重新组态总线节点，就可以改造系统。

3）可以随时分接总线电缆而无需终端电阻。但是总线电缆不能连接成环形。

3．网络结构

KNX/EIB 将网络分为区域、线路以及每条线路上的节点。一个网络最多包含 15 个区域，14000 个节点。线路的最大总长度为 1000 m。

耦合器可以用作线路放大器、线路耦合器和区域耦合器。线路耦合器用来对各线路彼此进行电隔离。此外，线路耦合器还可以过滤报文，只将允许的报文发送到其他线路，这将使总线负载降至最小程度。ETS 可以创建过滤表，然后将这些过滤表装载到耦合器。区域耦合器将一个区域的主线路与整个网络的区域线路连接起来。

在通信负载很大的系统（例如用于可视化或与管理级进行通信的系统）中，通过网关将主线路连接至以太网，可以增加传输量，建立带有更多节点的系统。

在 KNX/EIB 中，数据以串行方式传输，传输速率为 9.6 kbit/s。节点收到报文后将发送一个确认信息。KNX/EIB 支持以下传输媒体：双绞线（铜缆）、无线电、红外、光纤导线、Internet 和 ISDN 等。无线电和红外媒体是对楼宇进行改造的理想选择。

KNX/EIB 通过类似于以太网的报文执行数据交换，即各个总线节点可以相互独立地发送报文。如果发生冲突，各个报文将重新发送，发送节点将收到确认接收的报文。

通过 DP/EIB 连接器，可将 KNX/EIB 网络作为子网连接到 PROFIBUS-DP。也可以通过相应的接口将 KNX/EIB 连接到其他系统，例如其他楼宇自动化系统的控制中心和公共电话网络，例如 ISDN。

KNX/EIB 采用 CSMA/CA（带冲突避免的载波侦听多路访问）方式，与以太网的区别在于采用冲突避免（Collision Avoidance，CA）而不是冲突检测的方式。在每次发送之前，每个节点都会检查媒体是否空闲。

KNX/EIB 和互联网之间的连接变得日益重要，尤其是在家庭领域。这种连接使得即使不在家里也可以检测重要数据，例如室内温度，必要时还可以调节室内的温度值。

15.4 练习题

1．串行通信用于哪些场合？

2．西门子公司的串行通信可以使用哪些通信协议？

3．有哪些模块可以用于串行通信？各有什么功能？

4．S7 路由功能有什么优点？

5．怎样实现 PG/PC 的 S7 路由功能？

6．怎样实现 HMI 的 S7 路由功能？

7．工业无线局域网有什么特点？

8．广域网有什么特点？

9．KNX/EIB 是什么样的网络？

附　录

附录 A　常用缩写词

AI/AO：模拟量输入/模拟量输出。
ASCII：美国信息交换标准代码。
AS-i：执行器传感器接口，一种现场总线。
B：字节（Byte）。
BCD 码：二进制编码的十进制数。
BOOL：布尔变量，或称开关量、数字量。
C7：由 S7-300、操作面板、I/O、通信和过程监控系统组成的控制装置。
CBA：基于组件的自动化。
CP：通信处理器。
CPU：中央处理单元，CPU 模块的简称。
CRC：循环冗余码校验。
CSMA/CD：带冲突检测的载波侦听多路访问。
C 总线：通信总线。
DB：数据块。
DI：背景数据块，或数字量输入。
DO：数字量输出。
DINT：32 位有符号双整数。
DLL：动态链接库。
DP：分布式外部设备，PROFIBUS-DP 的简称。
DPM1：PROFIBUS 的 1 类 DP 主站，控制系统的中央控制器。
DPM2：PROFIBUS 的 2 类 DP 主站，DP 网络中的编程、诊断和管理设备。
DP-V0、DP-V1 和 DP-V2：PROFIBUS-DP 的 3 个版本。
DPV1：DP-V1 的简称。
DTE：数据终端设备。
DW 或 DWORD：32 位无符号双字。
ELM：电气链接模块。
EMC：电磁兼容性。
EN：欧洲标准。
ESM：电气交换模块。
FALSE：数字量的值（0）。
FB：功能块，有专用存储区（背景数据块）的子程序。

FBD：功能块图，PLC 的编程语言。
FC：功能，没有专用存储区的子程序。
FDL：PROFIBUS 的现场总线数据链路层。
FEPROM：Flash EPROM，快闪存储器。
FM：功能模块。
FTP：文件传输协议。
IM：接口模块。
GD：用于 MPI 通信的全局数据。
GSD 文件：常规站说明文件。
h：小时。
HEX：十六进制数。
HMI：人机界面。
HTTP：超文本传输协议。
HW Config：集成在 STEP 7 中的硬件组态工具。
I/O：输入/输出。
IEC：国际电工委员会。
IEEE：美国电气与电子工程师学会。
IM：接口模块。
INT：16 位有符号整数（Integer）。
Intranet：企业内部互联网。
Internet：国际互联网。
IRT：PROFINET 的等时同步实时通信。
ISDN：综合服务数字网。
ISO：国际标准化组织，或一种以太网通信服务。
IT：信息技术。
ITP：工业屏蔽双绞线。
K 总线：通信总线。
L#：32 位双整数常数的符号。
LAD：梯形图。
LAN：局域网。
LED：发光二极管。
LLC：逻辑链路控制。
LSAP：连接服务访问点。
M7：具有 AT 兼容计算机功能的控制器，作为 CPU 或功能模块使用。
MAC：媒体访问控制。
MMC：微存储卡。
MODEN：调制解调。
MPI：多点接口，用于 S7 设备的通信协议。
OB：组织块，操作系统与用户程序的接口。

OB1：用于循环处理的组织块，用户程序中的主程序。

OLE：对象链接与嵌入。

OLM：光学链接模块。

OP：操作员面板。

OPC：用于过程控制的 OLE。

OSI：开放系统互连模型。

OSM：光学交换模块。

P#：地址指针常数，例如 P#M2.0 是 M2.0 的地址。

PA：过程自动化。

PC：个人计算机。

PG/PC：编程器/个人计算机。

PI：外设输入存储区，可以通过它直接访问输入模块。

PLC：可编程序控制器。

PLCSIM：S7-300/400 的仿真软件。

PPI：点对点接口，用于 S7-200 的通信协议。

PQ：外设输出存储区，可以通过它直接访问输出模块。

PROFIBUS：一种现场总线。

PROFINET：基于工业以太网的现场总线。

PS：电源模块。

PtP：点对点通信。

RAM：随机读写存储器。

REAL：实数，又称为浮点数。

ROM：只读存储器。

RT：实时（Run time）。

s：秒。

S5：西门子早期 PLC 的型号。

SAP：服务访问点。

SCADA：监视控制与数据采集。

SDB：系统数据块。

SFB：系统功能块，集成在 CPU 模块中，通过它调用一些重要的系统功能，有专用存储区。

SFC：系统功能，集成在 CPU 模块中，通过它调用一些重要的系统功能，没有专用存储区。

SIMATIC：SIEMENS AG（西门子自动化与驱动集团）的注册商标。

SM：信号模块，数字量输入/数字量输出模块和模拟量输入/模拟量输出模块的总称。

SMS：短消息服务。

SMTP：简单邮件传输协议。

SNMP：简单网络管理协议。

SSL：系统状态表。

STEP 7：S7-300/400 的编程软件。
STL：语句表。
TCP/IP：用于以太网通信的传输控制协议/网际协议。
TIA：全集成自动化。
TP：触摸屏或双绞线。
TRUE：数字量的值（1）。
TSAP：运输层的服务访问点。
UDP：用户数据报协议。
USB：通用串行总线。
VAT：变量表。
WAN：广域网（Wide Area Network）。
WinCC：西门子的上位计算机组态软件。
WLAN：无线局域网。

附录 B　随书光盘内容简介

通信软件
iMap V3.0
S7-PDIAG V5.3
PDM V6.0
SIMATIC NET 2007
Drivemonitor V5.4
CP_PtP_CD：点对点通信处理器的光盘
PC Adapter USB Drive：USB/MPI 适配器的驱动程序
串口通信调试软件
\资料与手册\300_400 软件手册
PLCSIM 使用入门.pdf
S7-PLCSIM V5.4 User Manual.pdf
System Software and Standard Functions Reference Manual.pdf
使用 STEP 7 编程.pdf
梯形图（LAD）编程参考手册.pdf
用于 S7 的系统软件和标准功能参考手册.pdf
语句表（STL）编程参考手册.pdf
\资料与手册\300_400 硬件手册
CPU 31xC 和 CPU 31x 技术规范设备手册.pdf
CPU 31xC 和 CPU 31x 安装操作说明.pdf
S7-300 CPU 31xC 技术功能操作说明.pdf
S7-300 模块数据设备手册.pdf
S7-400 CPU Specifications Manual.pdf

S7-400 CPU 规格设备手册.pdf
S7-400 Module Data Reference Manual.pdf
S7-400 模块规范参考手册.pdf
S7-400 硬件与安装手册.pdf
S7-400H 容错系统手册.pdf
自动化系统 S7-300.pdf
组态硬件和通讯连接手册.pdf
Configuring Hardware and Communication Connection Manual.pdf
\资料与手册\ AS_i
Addressing and Diagnosis Instrument for AS-i.pdf
AS-i Introduction and basics Manual.pdf
CP 243-2 AS-i Master Manual.pdf
CP 343-2_343-2P AS-i Master Manual.pdf
DP_AS-i Link 20E Manual.pdf
如何配置 CP 343-2P.pdf
\资料与手册\ET 200
\ET 200S 模块中文设备手册
ET 200 产品目录 2008.pdf
ET 200B Manual.pdf
ET 200eco PN 操作指导.pdf
ET 200eco 操作指导.pdf
ET 200iS Manual.pdf
ET 200iSP 操作说明.pdf
ET 200M Operating Instructions.pdf
ET 200M 操作说明.pdf
ET 200pro Operating Instructions.pdf
ET 200pro 故障安全模块操作说明.pdf
ET 200pro 操作说明.pdf
ET 200R Manual.pdf
ET 200S 操作说明.pdf
ET 200X BM 147 CPU Manual.pdf
ET 200X Manual.pdf
\资料与手册\PA
DP_PA 耦合器连接器和 Y 型连接器操作说明.pdf
PA 总线阀门定位器与 S7 建立通讯.pdf
PROFIBUS-PA 应用技术手册.pdf
SITRANS T3K PA.pdf
如何配置 PA 总线仪表方案.pdf
智能电器阀门定位器操作说明.pdf

\资料与手册\ PROFIBUS
CP 342-5 / CP 342-5 FO Manual.pdf
CP 443-5 BASIC Manual.pdf
CP 443-5 Ext Manual.pdf
CP 5511 Manual.pdf
DP-DP Coupler User Description.pdf
PROFIBUS CP 组态和调试手册.pdf
Profibus Technology and Application.pdf
S7-CPs for PROFIBUS Configuring and Commissioning Manual.pdf
\资料与手册\产品样本
Industrial Communication Catalog 2009.pdf
ET 200 产品目录.pdf
PROFINET 产品样本 2005.pdf
S7-300 产品目录.pdf
S7-400 产品样本.pdf
Scalance X 交换机选型样本.pdf
SIMATIC 控制器产品手册.pdf
SIMATIC 工业软件.pdf
工业通讯及现场设备产品目录 2004.pdf
工业以太网产品样本 2005.pdf
人机界面系统产品手册.pdf
西门子工厂自动化产品系列.pdf
\资料与手册\第 15 章的手册
CP 340 PtP 打印机驱动程序入门指南.pdf
CP 340 安装与组态手册.pdf
CP 341 设备手册.pdf
CP 440 Manual.pdf
CP 441 Manual.pdf
S7-300_400 路由功能.pdf
\资料与手册\工业以太网与 PROFINET
Configuring Plants in iMap.pdf
CP 343-1 lean 手册.pdf
CP 343-1 PN Manual.pdf
CP 343-1 手册.pdf
From DP to PROFINET IO Programming Manual.pdf
PROFINET CBA 通信入门.pdf
PROFINET CBA 用户程序接口入门.pdf
PROFINET IO Configuration and Diagnostics.pdf
PROFINET 技术和应用系统描述.pdf

Profinet 系统手册.pdf
SCALANCE X-200 操作说明.pdf
SCALANCE X-400 SNMP 管理入门.pdf
SCALANCE X-400 路由指导手册.pdf
从 PROFIBUS DP 到 PROFINET IO 编程手册.pdf
工业以太网 CP 组态与调试手册.pdf
使用 CP 343-1 IT_CP 443-1 IT 的信息技术手册.pdf
\资料与手册\驱动
CBP_CBP2 通信板使用说明书.pdf
MM 420 变频器使用大全.pdf
MM 440 变频器使用大全.pdf
SIMOREG DC Master 使用说明书.pdf
SIMOVERT_MASTERDRIVES 使用大全_上.pdf
SIMOVERT_MASTERDRIVES 使用大全_下.pdf
VLT5000 PROFIBUS Manual.pdf
VLT5000 手册.pdf
\资料与手册\诊断
BT200 Physical Bus Test Device for PROFIBUS-DP.pdf
CPU 31xC 和 CPU 31x 安装操作说明.pdf
Diagnostic Repeater for PROFIBUS-DP Manual.pdf
FB125 HELP.chm
PDIAG 使用入门.pdf
PROFINET IO Diagnostics.pdf
S7-PDIAG Configuring Process Diagnostics Manual.pdf

附录 C　随书光盘中的例程说明

\Project\PB_MS：第 3 章的例程
\Convert：DP 主站与 SIMOVERT MASTERDRIVES 变频器通信。
\Danfoss：DP 主站与丹佛斯变频器通信。
\DCMaster：DP 主站与 DC MASTERDRIVES 通信。
\FC4_CTRL：DP 主站用 FC 4 控制 DP 网络和 DP 从站。
\MM440：DP 主站与变频器 MM 440 通信。
\PB_EM277：DP 主站与 EM 277 通信。
\PB_MS_1：DP 主站与 ET 200 通信。
\PB_MS_2：DP 主站与智能从站通信。
\PB_MS_3：DP 主站调用 SFC 14、15 与智能从站通信。
\PB_MS_4：CPU 413-2DP 与作为从站的 CP 342-5 通信。
\PB_MS_5，CP 443-5 Ext 与作为从站的 CP 342-5 通信。

\PB_MS_6：CP 342-5 作主站与 ET 200 通信。

\Project \PB_S7：第 4 章的 S7 连接例程

\PB_CTRL：CPU 413-2DP 远程监控另一台 CPU 413-2DP。

\PB_CTRL2：CPU 413-2DP 远程监控 CPU 313C-2DP。

\PB_S7_1：CPU 413-2DP 和 CP 342-5 之间的 S7 单向通信。

\PB_S7_2：CP 443-5 Ext 和 CP 342-5 之间的 S7 单向通信。

\PB_S7_A：CPU 413-2DP 和 CPU 313C-2DP 之间的 S7 单向通信。

\PB_S7_B：CPU 413-2DP 之间调用 USEND/URCV 的 S7 双向通信。

\PB_S7_C：CPU 413-2DP 之间调用 BSEND/BRCV 的 S7 双向通信。

\PB_S7_D：CPU 413-2DP 调用 GET/PUT 与 CP 443-5 的 S7 单向通信。

\PB_S7_E：CPU 413-2DP 与 CP 443-5 之间调用 USEND/URCV 的 S7 双向通信。

\Project \PB_FDL：第 4 章的 FDL 连接例程

\FDL_1：CPU 413-2DP 和 CPU 315-2DP 之间的 FDL 通信。

\FDL_2：两台 CPU 315-2DP 之间的 FDL 通信。

\FDL_2Net：两个 PROFIBUS 子网上的 CPU 315-2DP 的 FDL 通信。

\FDL_mul：CPU 315-2DP 之间的 FDL 多点传送方式通信。

\FDL_Pro1：两个项目中的 CPU 315-2DP 之间的 FDL 通信，项目之一。

\FDL_Pro2：两个项目中的 CPU 315-2DP 之间的 FDL 通信，项目之二。

\FDLbroad：CPU 315-2DP 之间的 FDL 广播方式通信。

\FDLfree2：CPU 315-2DP 之间的 FDL 自由第二层通信。

\Project \PB_Others：第 5 章的例程

\DX_1：智能从站之间的 DX 通信。

\DX_2：DX 通信，智能从站接收 ET 200M 发送的数据。

\DX_3：DX 通信，智能从站发送数据给网络上的另一个主站。

\Intrrupt：智能从站调用 SFC 7 触发主站的硬件中断。

\Isochron：ET 200M 和 ET 200S 作从站的恒定总线周期通信。

\SFC_12：DP 主站用 SFC 12 激活和禁止 DP 从站。

\Syncfrez：一组从站的输出同步与输入冻结。

\Project \PB_DIAG：第 6、7 章的例程

\443_Diag：CP 443-5 作主站，诊断 3 个 DP 从站的实例。

\FB_125：调用 FB 125 和 FC 125 诊断 DP 从站。

\FC3_Diag：调用 FC 3 诊断 CP 342-5 的 DP 从站。

\HW_Diag：CPU 313C-2DP 和 3 个 DP 从站的故障诊断实例。

\MS_Diag：DP 主站和智能从站的相互诊断。

\OB_Diag1：CPU 413-2DP 作主站，用 OB 的局部变量诊断 DP 从站的故障。

\OB1SFC13：在 OB1 和 OB82 中调用 SFC 13 诊断 ET 200M 和 ET 200B 的故障。

\PB_MS_7：CPU 313C-2DP 作主站，ET 200 作从站，用于错误 OB 应用实验。

\Repeater：诊断中继器应用例程。

\SFC_13：在 OB82 和 OB86 中调用 SFC 13 诊断 ET 200M 和 ET 200B 的故障。

\SFC_51：调用 SFC 51 读取局部系统状态表，诊断 DP 从站的故障。
\SFC13_S：调用 SFC 13 诊断 ET 200S 的故障。
\Project \Alarm Display：第 8 章的例程
\Alarm_D：调用 SFC 107/108 生成消息，用 WinCC flexible 和 PLCSIM 仿真。
\Alarm_S：调用 SFC 17/18 生成消息，用 WinCC flexible 和 PLCSIM 仿真。
\ReportEr：报告系统错误，用 WinCC flexible 和 PLCSIM 仿真。
\ReptErPC：报告系统错误，用 WinCC 和 PLCSIM 仿真。
\ReptPC1：报告系统错误，WinCC 用 DP 网络监控硬件 PLC，未生成 PC 站点。
\ReptPC2：报告系统错误，HMI 和 WinCC 用 DP 网络监控硬件 PLC，生成 PC 站点。
\S7PDIAG：用 S7_DIAG 生成地址监控的消息。
\SFC_52：生成用户自定义的诊断消息。
\Project \PA：第 9 章的例程
\PA：使用 PDM 组态 PROFIBUS-PA 设备。
\PB_PA1：仅使用 DP/PA 耦合器的 PROFIBUS-PA 通信组态。
\PB_PA2：使用 DP/PA 链接器的 PROFIBUS-PA 通信组态。
\PA_GSD：PA 设备的 GSD 文件。
\Project \Ethernet：第 10 章的例程
\IE_ISO：CP 343-1 IT 和 CP 343-1 的 ISO 连接通信。
\IE_S7_1：CP 343-1 IT 和 CP 343-1 调用 PUT/GET 的 S7 通信。
\IE_S7_2：CP 343-1 IT 和 CP 343-1 调用 USEND/URCV 的 S7 通信。
\IE_S7_3：CP 343-1 IT 和 CP 343-1 调用 BSEND/BRCV 的 S7 通信。
\IE_TCP：CP 343-1 IT 和 CP 343-1 的 TCP 连接通信。
\IE_UDP：CP 343-1 IT 和 CP 343-1 指定通信伙伴的 UDP 连接通信。
\ISOonTCP：CP 343-1 IT 和 CP 343-1 的 ISO-on-TCP 连接通信。
\UDP_MUL：3 个站点之间多点传送的 UDP 连接通信。
\UDPfree：3 个站点之间组态时未指定通信伙伴的 UDP 连接通信。
\Project \PROFINET：第 11 章的例程
\315_2PN：CPU 315-2PN/DP 作控制器的 PROFINET 网络通信。
\CP343_1：CP 343-1 作控制器的 PROFINET 网络通信。
\CP443_1：CP 443-1 作控制器的 PROFINET 网络通信。
\New Plant：iMap 的项目。
\PN_CBA1：用于生成 CBA 组件的项目。
\PN_CBA2：用于生成 CBA 组件的项目。
\Project \ASI：第 12 章的例程
\ASI_Link：基于 DP/AS-i Link 20E 的 AS-i 网络通信。
\ASI343_2：基于 CP 343-2 的 AS-i 网络通信。
ASI243_2.mwp：基于 CP 243-2 的 AS-i 网络通信，S7-200 的项目。
\Project \OPC：第 13 章的例程
\DP_OPC：基于 DP 网络的 OPC 通信的 STEP 7 项目。

\IE_OPC：基于以太网的 OPC 通信的 STEP 7 项目。
\MPI_OPC：基于 MPI 网络的 OPC 通信的 STEP 7 项目。
\组态王 OPC：OPC 通信的组态王项目。
DP_OPC.opp：基于 DP 网络的 OPC Scout 项目文件。
IE_OPC.opp：基于以太网的 OPC Scout 项目文件。
MPI_OPC.opp：基于 MPI 网络的 OPC Scout 项目文件。
\Project \MPI_GD：第 14 章 MPI 全局数据通信例程
\MPI_GD_1：CPU 413-2DP 和 CPU 315-2DP 之间的全局数据通信。
\MPI_GD_2：CPU 315-2DP 之间的全局数据通信。
\MPI_GD_3：CPU 413-2DP 和两台 CPU 315-2DP 之间的全局数据通信。
\MPI_GD_A：CPU 413-2DP 之间的全局数据通信。
\MPI_GD_B：CPU 413-2DP 之间事件驱动的全局数据通信。
\Project \MPI_UC：第 14 章 S7 基本通信例程
\MPI_224A：CPU 315-2DP 与 CPU 224 的 MPI 通信。用 IW0 控制对方的输出点。
\MPI_224B：CPU 315-2DP 通过 MPI 读写 CPU 224 的 V 数据区。
\MPI_UC_1：CPU 413-2DP 和 CPU 315-2DP 调用 X_RCV/X_SEND 的通信。
\MPI_UC_2：CPU 413-2DP 和 CPU 315-2DP 调用 X_PUT/X_GET 的通信。
\MPI_UC_3：CPU 315-2DP 之间调用 X_RCV/X_SEND 的通信。
\MPI_UC_4：CPU 315-2DP 之间调用 X_PUT/X_GET 的通信。
\MPI_UC_5：3 台 PLC 之间 S7 基本通信综合应用例程。
MPI_224.mwp：CPU 224 与 CPU 315-2DP 通信的程序。
\Project \MPI_S7：第 14 章 MPI S7 连接例程
\MPI_CTRL：CPU 413-2DP 远程监控另一台 CPU 413-2DP。
\MPI_S7_1：CPU 413-2DP 和 CPU 313C-2DP 之间的 S7 单向通信。
\MPI_S7_A：CPU 413-2DP 之间调用 PUT/GET 的 S7 单向通信。
\MPI_S7_B：CPU 413-2DP 之间调用 USEND/URCV 的 S7 双向通信。
\MPI_S7_C：CPU 413-2DP 之间调用 BSEND/BRCV 的 S7 双向通信。
\MpiCtrl2：CPU 413-2DP 远程监控 CPU 315-2DP。
\MpiS7Mul：CPU 413-2DP 之间 S7 通信 SFB 综合应用例程。
\Project \Chapter15：第 15 章的例程
\ASCII：CP 340 使用 ASCII 协议通信。
\HMI_Rout：用于 HMI 的 S7 路由功能。
\IE_MPI：用于 PG/PC 的 S7 路由功能。

参 考 文 献

[1] 崔坚等. 西门子工业网络通信指南[M]. 北京：机械工业出版社，2005.

[2] 廖常初. S7-300/400 PLC 应用技术[M]. 2 版. 北京：机械工业出版社，2008.

[3] 廖常初，陈晓东. 西门子人机界面（触摸屏）组态与应用技术[M]. 2 版. 北京：机械工业出版社，2008.

[4] 廖常初. S7-300/400 PLC 应用教程[M]. 2 版. 北京：机械工业出版社，2008.

[5] 廖常初. PLC 编程及应用[M]. 3 版. 北京：机械工业出版社，2008.

[6] 缪学勤. 现场总线国际标准最新进展[J]. 电气时代，2007（8）.

[7] Siemens AG. Industrial Communication Catalog IK PI，2009.

[8] Siemens AG. 用于S7 的系统软件和标准功能参考手册，2007.

[9] Siemens AG. Communication with SIMATIC System Manual，2006.

[10] PROFIBUS Trade Organization PTO. PROFIBUS Technology and Application，Karlsruhe Germany，2002.

[11] J Weigmann, G Kilian. Decentralization with PROFIBUS DP/DPV1[M]. 2nd ed. Erlangen: Publicis Corporate Publishing, 2003.

[12] Siemens AG. System Software for S7-300/400 System and Standard Functions Reference Manual，2006.

[13] Siemens AG. Configuring Hardware and Communication Connections Manual，2007.

[14] Siemens AG. 工业通讯及现场设备产品目录，2004.

[15] Danfoss Group. VLT5000 PROFIBUS Manual，2005.

[16] Siemens AG. Distributed I/O device ET 200M Operating Instructions，2006.

[17] Siemens AG. ET 200B Distributed I/O Station Manual，1999.

[18] Siemens AG. Distributed I/O System ET 200S Manual，2005.

[19] Siemens AG. S7-CPs for PROFIBUS Configuring and Commissioning Manual，2005.

[20] Siemens AG. BT 200 Physical Bus Test Device for PROFIBUS-DP，2004.

[21] Siemens AG. Diagnostic Repeater for PROFIBUS-DP Manual，2002.

[22] Siemens AG. S7-PDIAG Configuring Process Diagnostics Manual，2005.

[23] Siemens AG. From DP to PROFINET IO Programming Manual，2006.

[24] Siemens AG. CP 343-2_343-2P AS-i Master Manual，2008.

[25] Siemens AG. Configuring Plants in iMap Configuration Manual，2006.

[26] Siemens AG. S7-PLCSIM V5.4 User Manual，2007.

ISBN 978-7-111-28256-3

◎策划 时静

◎封面设计 旭洲企划 刘吉维

电气信息工程丛书

- ◇ PLC 编程及应用（第3版）
- ◇ S7-300/400 PLC 应用技术（第2版）
- ◇ 西门子 S7-200 PLC 编程及应用案例精选
- ◇ 西门子人机界面(触摸屏)组态与应用技术
- ◆ 西门子工业通信网络组态编程与故障诊断
- ◇ CS/CJ 系列 PLC 应用基础及案例
- ◇ 欧姆龙 CP1H PLC 应用基础与编程实践
- ◇ IEC 61131-3 编程语言及应用基础
- ◇ Protel 99 SE 原理图与 PCB 设计及电路仿真
- ◇ Protel 99SE 实战详解与技巧
- ◇ Protel 2004 电路原理图及 PCB 设计
- ◇ 基于 Altium Designer 的原理图与 PCB 设计
- ◇ VHDL 数字电路及系统设计
- ◇ OrCAD 电路原理图设计与应用
- ◇ 单片机应用及 C51 程序设计
- ◇ 51 单片机快速上手
- ◇ 数字信号处理器原理、结构及应用基础——TMS320F28x
- ◇ Freescale 9S12 十六位单片机原理及嵌入式开发技术
- ◇ 基于 EDK 的 FPGA 嵌入式系统开发
- ◇ 嵌入式可配置实时操作系统 eCos 开发与应用（第2版）
- ◇ Linux PowerPC 详解——核心篇
- ◇ 典型数控系统应用技术（FANUC 篇）

上架建议：工业技术／电气工程

地址：北京市百万庄大街22号 邮政编码：100037
电话服务
社服务中心：(010)88361066 网络服务
销售一部：(010)68326294 门户网：http://www.cmpbook.com
销售二部：(010)88379649 教材网：http://www.cmpedu.com
读者购书热线：(010)88379203 封面无防伪标均为盗版

ISBN 978-7-111-28256-3
9 787111 282563 >

定价：69.00元（含1DVD）