

第一章 SIMATIC H 系统介绍

1.1 SIMATIC H 系统发展的历史

1.1.1 H 系统的定义

在现代工业的各个领域，要求拥有一种能够满足经济、环保、节能的高度自动化系统，同时，具有冗余及故障安全功能的可编程控制器是针对最高等级的控制需求。

H(高可靠性)系统，通过将发生中断的单元自动切换到备用单元的方法实现系统的不中断工作，H 系统通过部件的冗余实现系统的高可靠性。

F(故障安全)系统，通过将发生中断的系统切换到安全状态(通常为停车)来避免造成对生命、环境和原材料的破坏。

FH 或 HF(故障安全和高可靠性)系统，通过将发生故障的通道关闭，保证系统无扰动运行。

S7-400H 是西门子提供的最新冗余 PLC。由于他是 SIMATIC S7 家族的一员，这意味 S7-400H 拥有所有 SIMATIC S7 具有的先进性。

1.1.2 SIMATIC H 系统的发展

西门子 SIMATIC H 产品发展列表：

1986: S5-150H 带串行数据传输。

1987: S5-150H 带并行数据传输。

1990: S5-155H 带 CPU946R/947R。

1991: S5-115H 带 CPU942H。

1992: S5-155H 新功能(支持 2-00-3 数字和模拟输入)。

1994: S5-155H 带 CPU948R 和新功能。

1997

- 四月：分离机架 S5-155H。
- 九月：S5-155H Lite 版本。

1998

- 一月：IM153-3 连接 S5-115H,155H 和 S7 软冗余。
- 五月：S7 软冗余。
- 七月：S7-400 冗余电源。
- 十二月：带 CPU417H 的 S7-400H 系统 Beta 版发布。

1999

- 五月：带 CPU417H 的 S7-400H 系统全面发布。

2000

- 八月：H-CPU 在 S7-400F 中使用。

- 十二月：带 CPU414H 的 S7-400H 系统全面发布。

2001

- 二月：Y-Link 连接单通道 DP-Slaves。

2003

- 一月：支持冗余 DI/DO AI/AO 模件。

2004 新 CPU

- 十二月：V4 CPU 414-4H/417-4H 全面发布。
 - 程序执行性能增加
 - CPU414-4H 大约为原 CPU 的 1.5 倍
 - CPU417-4H 大约为原 CPU 的 2.5 倍
 - 更多的内存
 - CPU414-4H 集成 1.4MByte
 - CPU417-4H 集成 20MByte
 - 更高的稳定性

内存带有自动错误探查和修复 (EDC)

- 同步电缆长度增加
 - 以前 500m
 - 短距离同步模件最长 10m
 - 长距离同步模件最长 10Km

2005

- 二月：在 S7-400F 系统中使用 H-CPU V4。

2007

十月：V4.5 CPU 414-4H、417-4H 全面发布

- 工作内存增大
 - CPU414-4H 集成 2.8M Byte
 - CPU417-4H 集成 30M byte
- 执行速度加快
 - CPU414-4H 大约为原来的 1.33 倍
 - CPU417-4H 大约为原来的 1.4 倍
- 新增一款 V4.5 412-3H CPU
 - 集成 768K 内存

1.2 故障率的计算方法

系统发生故障的频率和时间的关系可以用浴盆曲线来表达，如图 1-1 所示。。

1.2.1 浴盆曲线原理

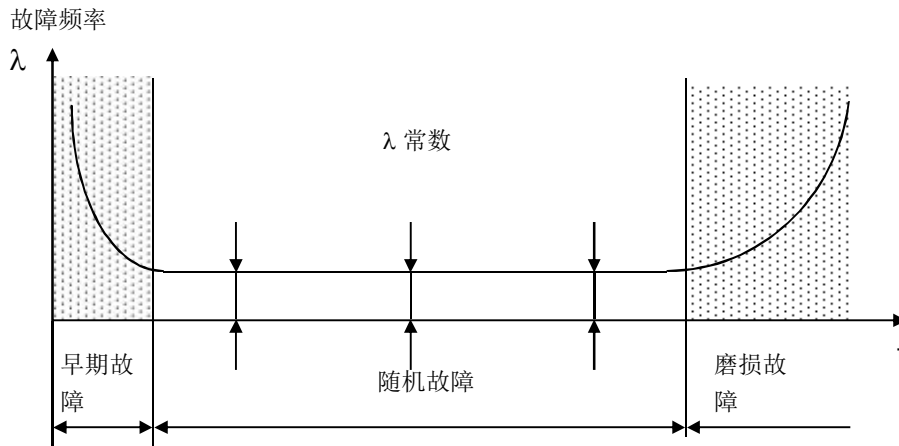


图 1-1 浴盆曲线

从该曲线可以看出，系统故障率在系统早期投用和晚期老化后的故障率较高，而在使用中间段时随机故障率相对恒定。

1.2.2 故障率计算公式

$$\lambda \approx \frac{C}{N \cdot \Delta t}$$

C=在考虑的时间范围 Δt 内，发生故障的部件数

N=整个使用的部件数

Δt =考虑的时间范围

1.2.3 平均无故障时间 MTBF

$$MTBF=1/\lambda$$

1.2.4 可靠性计算公式

$$A_S=MTBF/(MTBF+MDT)$$

MDT=平均故障时间（或

MTTR=平均修复时间)

举例:

- MTBF=100h, MDT=0.5h-→A=99.5%!
- MTBF=1year, MDT=24h-→A=99.7%

因此, 考虑系统的可靠性需同时考虑 MTBF 和 MDT。

1.2.5 如何增加系统的可靠性

从可靠性公式中可以看出, 增加系统的可靠性可以从提高 MTBF 和降低 MDT 两个方面进行。

1.2.5.1 增加系统的稳定性

增加稳定性, 可从如下环节考虑:

- 设备生产商
 - 使用高质量部件
 - 使用具有更高标准的部件
 - 预烧
 - 抗过载保护
 - 质量控制
 - 冗余
- 工厂设计人员
 - 网络结构
 - 冗余安装
 - 符合安装条件需要
 - 在合适的环境条件下使用
- 工厂操作人员
 - 维护
- 快速故障诊断
 - 自动故障诊断和定位 (自测试)
 - 具有诊断功能
 - 诊断工具的稳定性
 - 训练有素的维护人员
- 快速修复
 - 系统不停机情况下修复 (在线修复)
 - 修复工程容易
 - 快速备件发送
 - 训练有素的专业人员

1.2.5.2 整个系统的 MTBF

对于串行系统而言，系统故障发生率是各部件故障发生率之和，如图 1-2 所示。

举例：

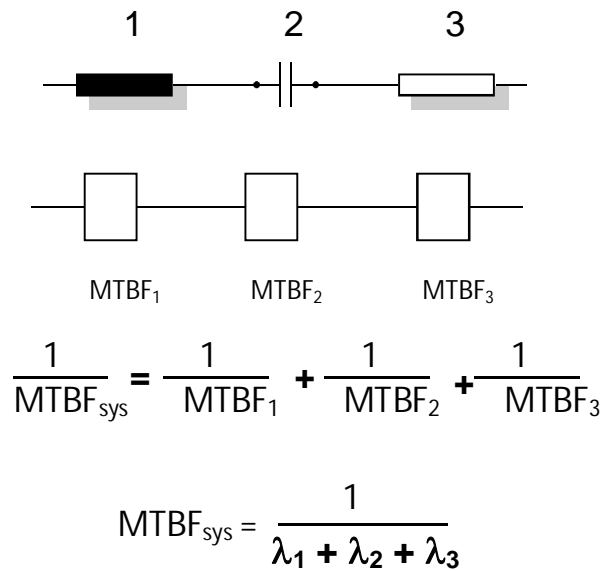


图 1-2 串行系统的 MTBF

可见部件越多 MTBF 时间越小。

SIMATIC H 系统的平均无故障时间：

CPU: 15 年

IO 模件: 50 年

平均修复时间: 大约 4 小时，对于工厂设计通常 10 年进行一次大修，更换部件。

注：

MTBF 的计算方法没有一个确定的标准，各个产品生产商的计算方法不同。

1.2.5.3 故障容错性比较

对于单机操作和冗余操作方式下的 H 系统故障容错性比较，如图 1-3 所示：

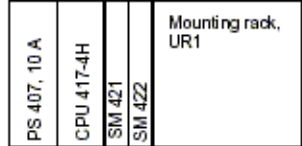
Fault-Tolerant CPU in Single Operation (e.g. CPU 417-4H)	Factor
	1

图 1-3a

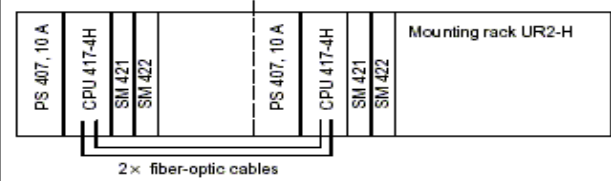
Redundant CPU 417-4 H in split mounting rack	Factor
	57

图 1-3b

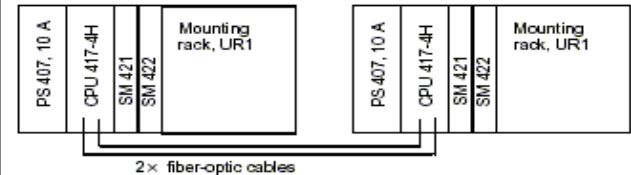
Redundant CPU 417-4H in separate mounting racks	Factor
	59

图 1-3c

以单机操作的系数为 1，那么安装与 H-RACK 机架上（2X9 槽，中间物理分隔）的 H 系统可靠性提高 57 倍，如果 H 系统分别安装于完全分开的机架，例如普通九槽机架，可靠性将提高 59 倍。

1.3 SIMATIC H 系统的优点及应用领域

1.3.1 冗余系统的目的

使用冗余系统的目标是减少因一个错误或系统维护而导致的产品损失。停车成本越高，越值得采用冗余系统。通常投资冗余系统较高的费用会因避免的产品损失而很快地返还。

软冗余：

在许多应用场合，对于工厂单元的冗余质量和控制范围并不完全必要使用一个专用的冗余系统来执行，例如切换时间较长或丢失部分信息，但并不影响控制过程。通常来说，简单的软冗余机制就可将因一个错误事件导致的故障控制过程切换到一个备用系统上而继续进行。利用可选软件包“SIMATIC S7 Software Redundancy”软冗余可在 S7-300 和 S7-400 标准系统中运行。这样配置的软冗余系统可用于能够容忍切换延迟在秒级的控制过程，例如：水厂工作、水处理或交通控制。

冗余 I/O:

当两个单独的模块被组态并以冗余对方式使用时，则采用冗余 I/O。这种方式能够得到最高的可靠性，因为系统可以忍受单个 CPU 和单个信号模块发生故障。

冗余 I/O 是通过使用功能块库“function I/O redundancy”来执行的。这些功能块只能在 S7-400H 系统中使用。

H 系统的优点:

避免由于单个 CPU 故障造成系统瘫痪，无扰动切换，不会丢失任何信息。需要 H 系统高可靠性的原因:

- 处理贵重原料
- 停车或不合格产品的成本昂贵
- 控制系统瘫痪导致重新开车的费用高
- 无需监视和维护人员的操作场合

1.3.2 H 系统的工业应用领域

S7-400H 系统是用于高程度的可靠性和容错能力的场合，例如:

能源开采和配送（石油、天然气、电力）

- 电力
- 管输
- 离岸
- 区域加热系统

化学制品、电力化工、石化和采矿工业

环保工程

- 水处理
- 垃圾焚化

纸浆和造纸

钢铁

食品和包装

玻璃工业

半导体工业（应用）

交通

- 隧道自动化
- 海底隧道自动化

机场

- 跑道照明
- 行李输运

1.3.3 S7-400H 系统的架构

S7-400H 系统的冗余结构确保了任何时候的系统可靠性，例如所有的重要部件都是冗余配置。这包括了冗余的 CPU、供电模块和用于冗余 CPU 通信的同步模块。根据特定的自动化控制过程需要，还可以配置冗余客户服务器、冗余通讯介质、冗余接口模块 IM153-2 等，如图 1-4 所示。

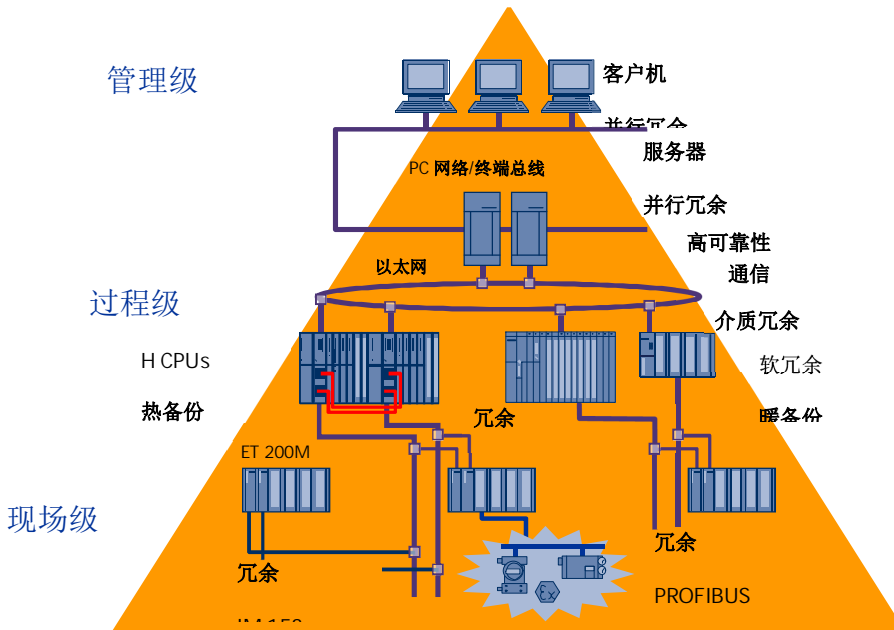


图 1-4 S7-400H 的冗余架构

1.3.4 SIMATIC S7-400H 系统受益于用户

对于用户来说，可以将 H 系统作为普通的系统，因为 H 系统具有隐藏的冗余性

- 透明的编程（与非冗余系统编程方法一致）
- 标准的系统参数化
- 标准的处理方法

S7-400H 系统可无限制使用所有的 SIMATIC 编程语言，并可作为 F 和 FH 系统的平台。

1.3.5 冗余原理

S7-400H 系统中两个 CPU 同时工作，CPU 没有切换时间。I/O 为 2002 主动冗余，同时工作。

1.3.5.1 S7-400H 系统亮点

1. 平滑的主从切换

2. 自动事件同步
3. 集成的错误识别和错误定位功能
4. 操作期间可对系统进行修改
5. 类似标准 CPU 的在线编程
6. 下载程序时，只考虑单个 CPU，程序可自动拷贝到另一个 CPU 中。
7. CPU 修复后自动再进入。
8. 运行中所有部件可更换。

1.3.5.2 无扰动主从切换

- CPU 无切换时间
- IM153 切换时间参考 PROFIBUS 参数
- 切换期间输出保持
- 切换期间无信息或报警/中断丢失。

CPU 切换条件标准

主站故障（供电、机架、CPU）

DP 链或 DP 从站接口模块故障都不会强制 CPU 切换。

1.3.5.3 同步原理

H 系统采用西门子专利的事件同步方式进行同步。保证同步的有效性的同时又不会增加 CPU 的运算负担。

同步事件包括：

- 过程映象区更新
- I/O 直接访问
- 中断、报警
- 更新计时器
- 使用通讯功能时的数据改变

这种同步方式给客户带来的好处：

- 用户不用考虑怎样实现同步
- 无命令限制
- 从标准 CPU 到高可靠性 CPU 用户程序容易移植

无扰动切换

- 无信息丢失
- 无报警/中断丢失

1.3.5.4 全面的自测试功能

SIMATIC H 系统的自测试功能处理的范围包括：CPU、内存、同步连接。

测试形式：

1. H 系统启动时的自测试
 - 完全的测试
2. 循环模式的自测试
 - 作为背景任务永久执行
 - 在指定范围时间内完成（缺省 90 分钟），如图 1-5 所示：

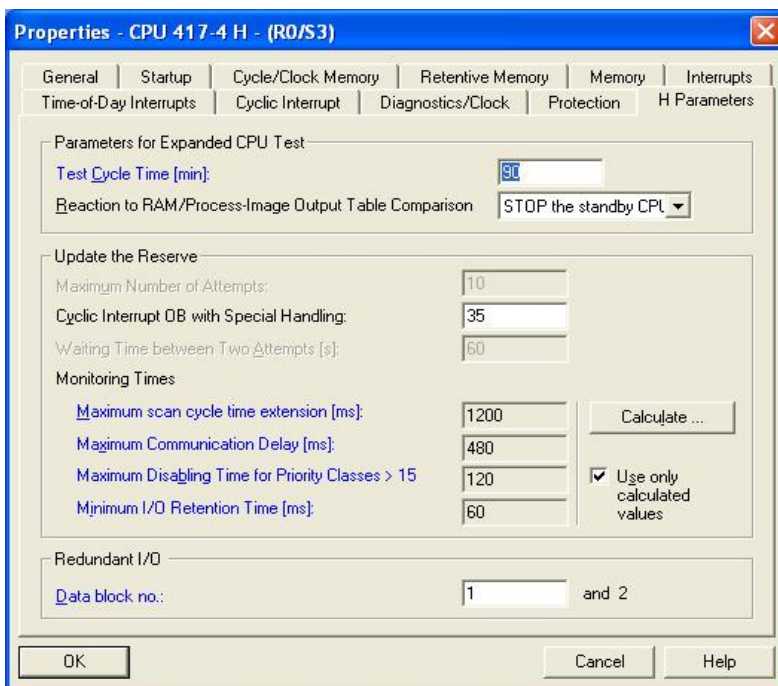


图 1-5 设置 H 系统自检测参数

1.3.5.5 在线编程

S7-400H 系统可以在线进行编程修改。与标准系统的在线修改一样，修改后的程序被下载到连接的 CPU 中，通过光纤同步模块将程序传输到另一个 CPU 中。可以通过 MPI、PROFIBUS 或 Ethernet 接口在线修改程序。

1.3.5.6 在线修改系统参数及添加、删除硬件功能（H-CIR）

在 S7-400H 系统运行中可以在线修改系统参数、添加、删除硬件，可在线修改的范围包括：CPU 内存组态

添加或删除:

- 中央 I/O 或 CP
- DP 从站
- PA 接口和 PA 从站
- Y-Link 和其从站
- 模块化 DP 从站中模件

CPU 参数 (蓝色标注的参数可修改)

1.3.5.7 模件更换

运行模式下可添加和删除的模件:

- I/O 和 CP
- 同步模件
- 冗余 IM153-2
- 冗余电源

在停电状态下能够替换的冗余部件:

- 标准电源
- 中央 IM
- CP PROFIBUS
- CPU(更换后可自动更新程序和数据)

1.3.5.8 备份 CPU 与主 CPU 同步建立的过程

CPU 故障后, 替换的备份 CPU 与主 CPU 同步连接自动建立, 备份 CPU 发出 Link-up 请求, 主站在禁止删除、拷贝和生成块功能后将所有数据发送给备份 CPU。备份 CPU 执行自测试, 然后向主站发出更新请求。主站在终止已组态连接的通讯和禁止低级别的报警后, 拷贝动态数据给备份 CPU。主站运行用户程序, 在禁止所有报警和中断后向已 Link-up 的备份 CPU 发送上次更新后发生改变的动态数据。备份 CPU 接收主 CPU 的输入、输出、定时器、计数器和内存位信息, 主 CPU 使能报警/中断和通讯, 主、备 CPU 进入到冗余、同步操作过程。同步连接的建立如图 1-6 所示:

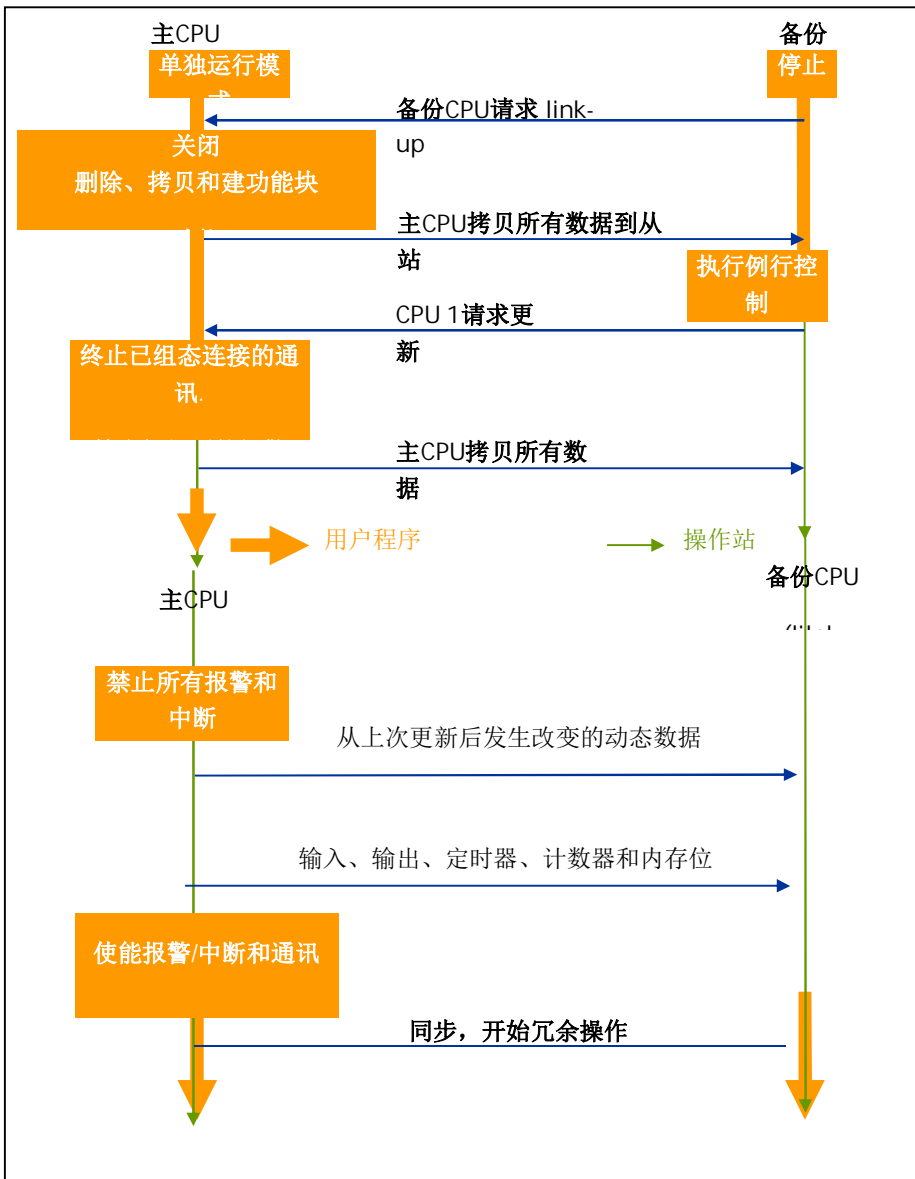


图 1-6

第二章 S7-400 H 硬件组态

以例子的形式介绍 S7-400H 系统的组态过程

2.1 例子所需硬件和软件

硬件：

一套 S7-400H PLC，包括

- (1) 1 个安装机架 UR2-H
- (2) 2 个电源模板 PS 407 10A
- (3) 2 个容错 CPU，CPU414-4H 或 CPU 417-4H
- (4) 4 个同步子模板
- (5) 2 根光缆

一个 ET200M 分布式 I/O 设备，包括

- (6) 2 个 IM 153-2
- (7) 1 个数字量输入模板
- (8) 1 个数字量输出模板

必备的附件，如 PROFIBUS 屏蔽电缆及网络连接器等。

软件：

STEP 7 V5.3 SP2 标准版(已集成冗余选件包)或更高版本。

2.2 硬件安装

- (1) 设置机架号

CPU V3 版本，通过同步子模板上的开关设置；

CPU V4 版本，通过 CPU 背板上的开关设置；

CPU 通电后此机架号生效。

- (2) 将同步子模板插到 CPU 板中。

- (3) 连接同步光缆

将两个位于上部的同步子模板相连；

将两个位于下部的同步子模板相连；

在打开电源或启动系统之前要确保 CPU 的同步光缆已经连接。同步光纤的连接如图 2-1 所示：

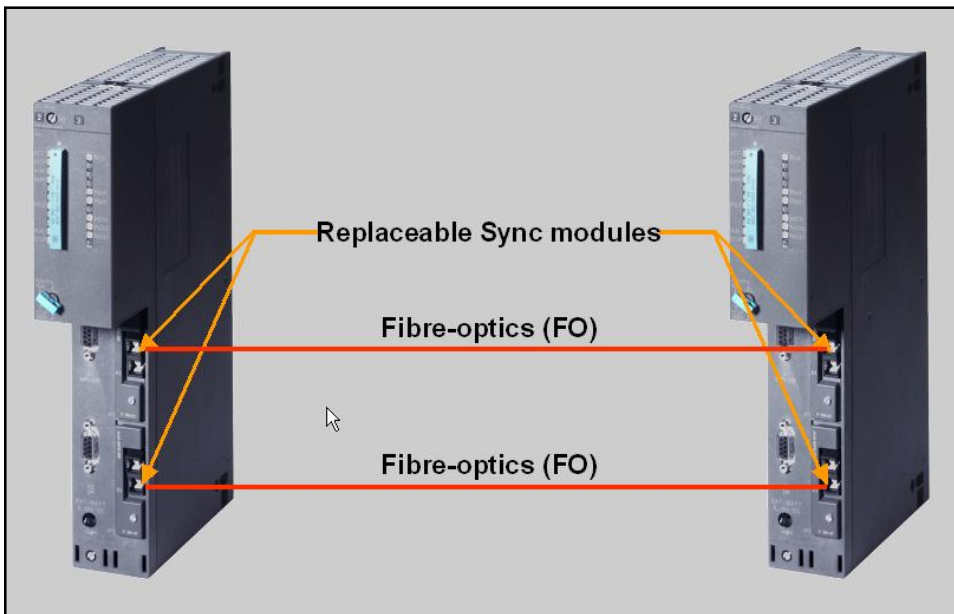


图2-1 S7-400H 同步光纤的连接

(4) 组态分布式 I/O 站 ET200M，使其作为具有切换功能的 DP 从站。

(5) 将编程器连到第一个容错 CPU（CPU0）上，此 CPU 为 S7-400H 的主 CPU。

(6) 通电后 CPU 自检

CPU 第一次通电时，将执行一次 RAM 检测工作，约需 3 分钟。这段时间内 CPU 不接收通过 MPI 接口来的数据，并且 STOP LED 灯闪烁。如果有备用电池，再次通电时不再做此项检查工作。

(7) 启动 CPU

装入程序后执行一个热启动操作：首先启动主 CPU，然后启动热备 CPU。

2.3 使用 STEP 7 进行组态

2.3.1 创建项目组态 S7-400H

在 STEP7 中新建一个项目，在 Insert 菜单下的 Station 选项中选择 SIMATIC H Station，添加一个新的 S7-400H 的站，如图 2-2 所示：

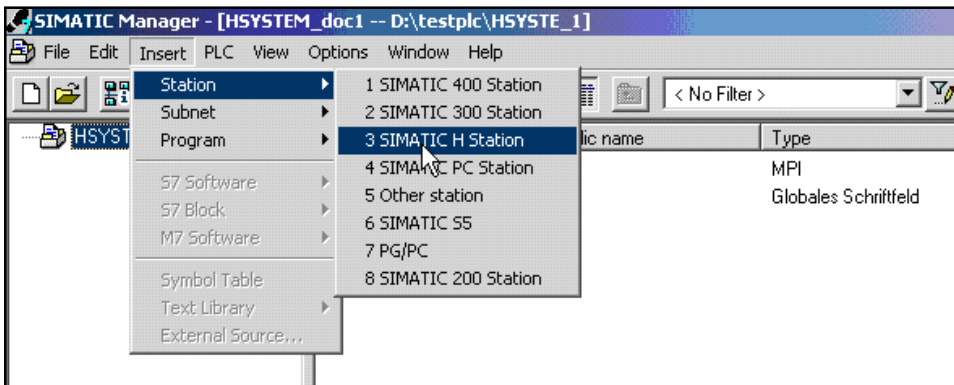


图 2-2 创建项目和添加 S7-400H 站

2.3.2 配置硬件

- (1) 在 S7-400H 站目录下双击 Hardware 打开硬件配置。
- (2) 添加一个 UR2 H 机架，如图 2-3 所示：

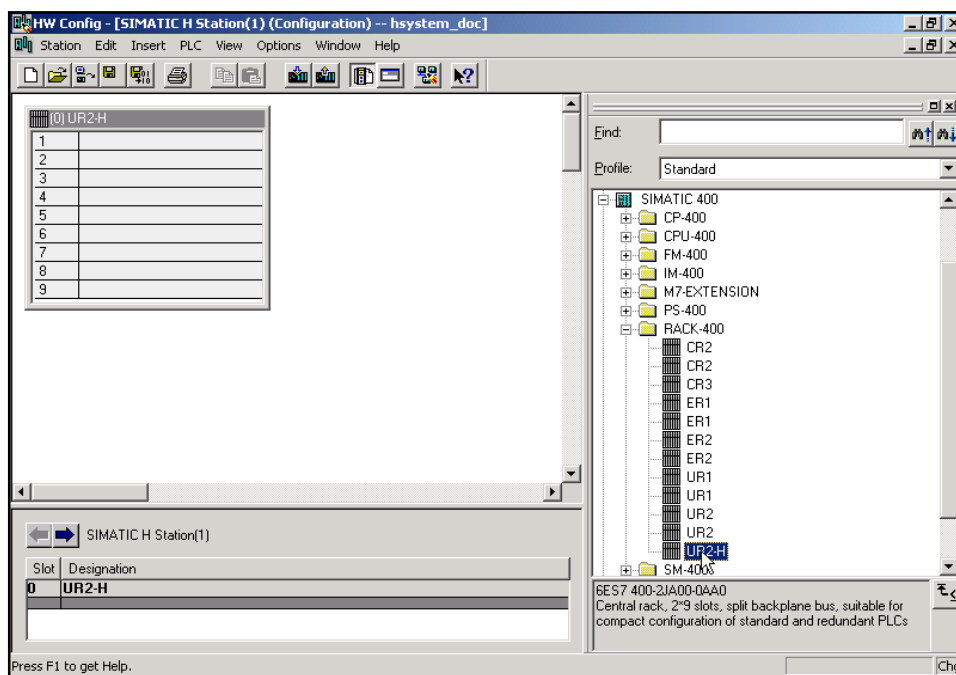


图2-3 添加UR2H机架

(3) 配置电源和CPU，并设定CPU上PROFIBUS DP主站的地址，本例为2，如图2-4所示：

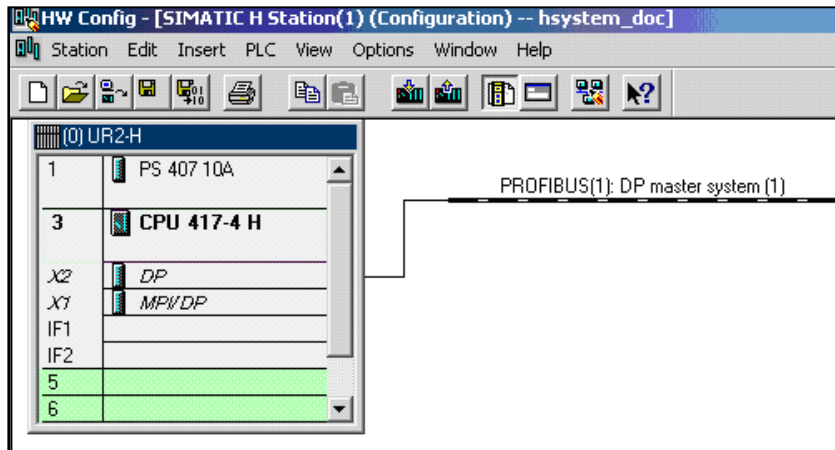


图2-4 添加S7-400H CPU

(4) 添加同步子模板到IF1和IF2槽位上。

(5) 添加以太网网卡并配置MAC网络地址，如图2-5所示：

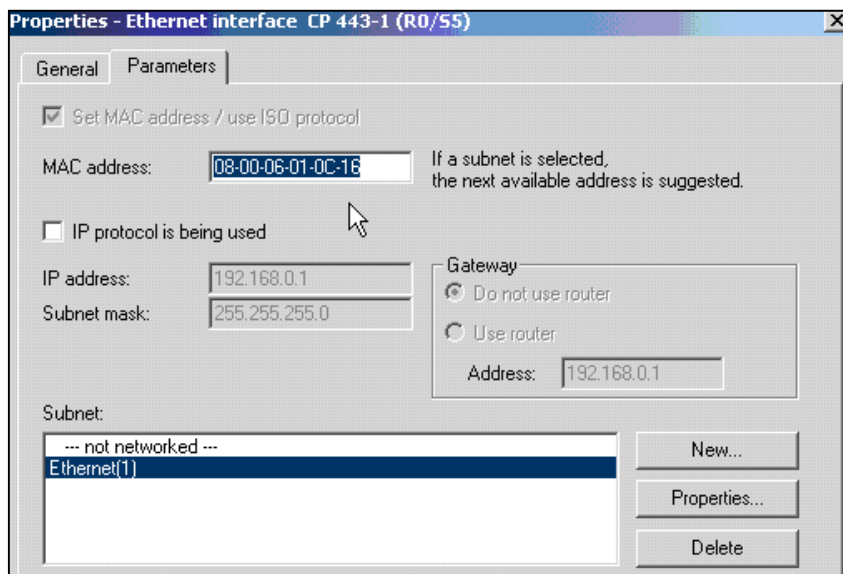


图2-5 配置以太网模板CP443-1

只有以太网可以与 HMI 系统 WINCC 通信。

(6)将机架 0 的硬件配置拷贝，粘贴，复制机架 1 并调整网络参数，如：以太网的 MAC 地址等，在硬件组态中出现两个机架，如 2-6 所示：

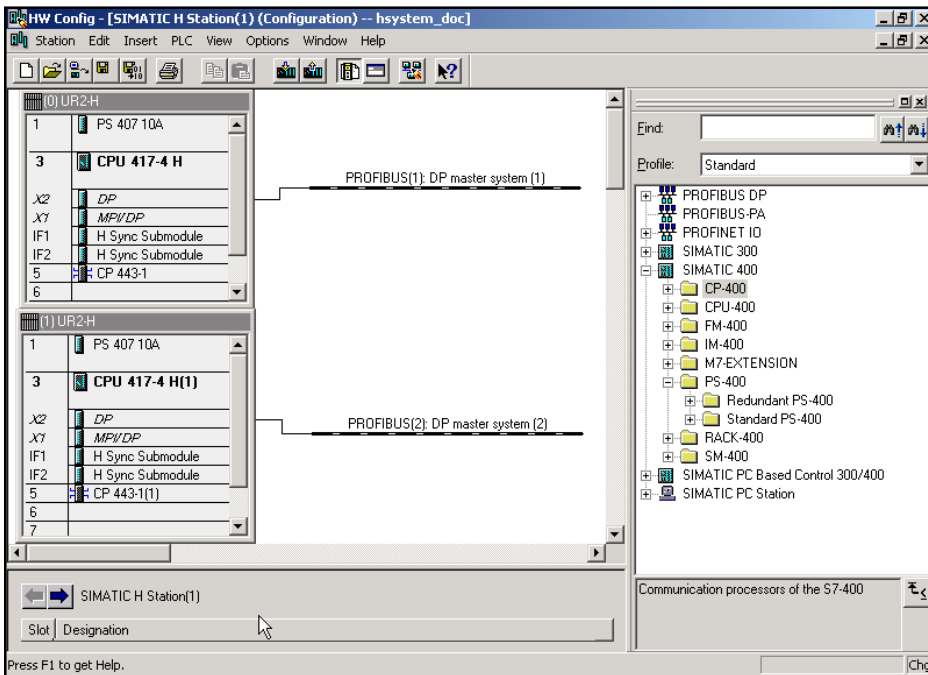


图2—6 S7-400H的硬件配置图

2.3.3 系统参数设置

容错站中的模板参数赋值与 S7-400 标准站中的模板参数赋值没有什么区别。

对于中央处理器单元只需对 CPU0（机架 0 上的 CPU）设定 CPU 参数，所设定的数值将自动分配给 CPU1（机架 1 上的 CPU）。除以下参数外 CPU1 的设置不能更改：

- CPU 的 MPI 地址
- 集成 PROFIBUS DP 接口的站地址和诊断地址
- I/O 地址区中的模板

在 I/O 地址区编址的模板必须完全在过程映象内或完全在过程映象外，否则不能保证数据的一致性。

CPU 参数设置

(1) 点击 Cycle/Clock memory（循环/时钟存储器）选项栏，如图 2—7 所示，设置 CPU 循环处理参数。

建议设置：

扫描循环监视时间尽可能长（例如 6000 ms）

过程输入映象尽可能小（稍大于实际使用的输入点数）

过程输出映象尽可能小（稍大于实际使用的输出点数）

出现 I/O 访问错误时调用 OB 85：只对于输入错误和输出错误

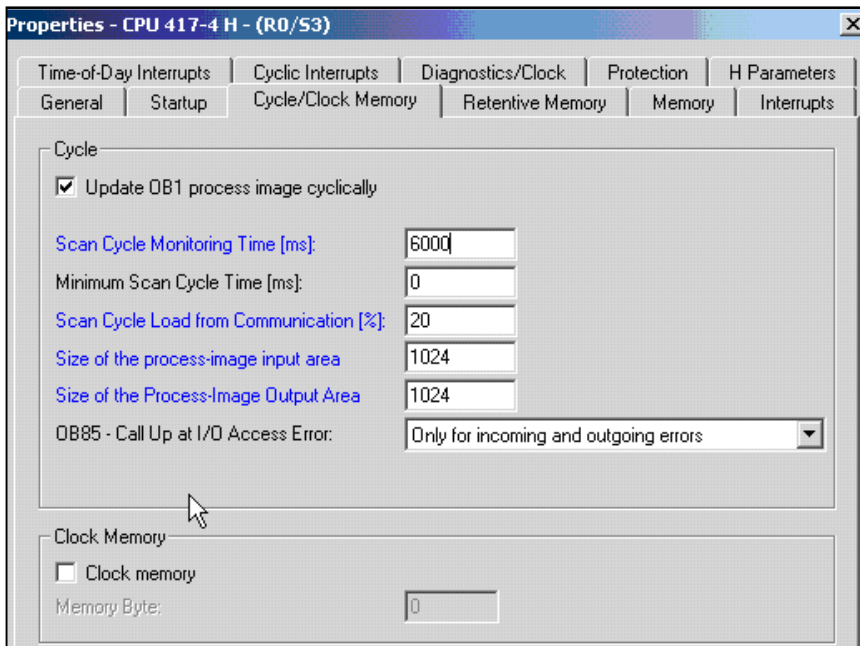


图2—7 “Cyclic/Clock Memory”参数配置

(2)设置诊断缓冲区中的报文数量

在“Diagnostics/Clock（诊断/时钟）”选项栏中可以设置诊断缓冲区的报文数量

建议设定较大数值，例如：1000。

(3) 模块的监控时间

在“Startup（启动）”选项栏中，可以指定模块监视时间，它取决于容错站的配置。如果监视时间太短，CPU 将在诊断缓冲区中输入 W#16#6547 事件。

参数的传输时间取决于以下因素：

- 总线系统的传输速率（传输速率高=> 传输时间短）
- 参数和系统数据块的大小（参数长=> 传输时间长）
- 总线系统上的负载（从站多=> 传输时间长）

建议设置：600（对应于 60 秒）

(4) CPU 自检周期

在“H Parameter（冗余系统参数）”选项栏中，配置 CPU 后台自检的周期。可选范围为 10 分钟到 60000 分钟。

建议设置：使用缺省值 90 分钟，如图 2—8 所示：

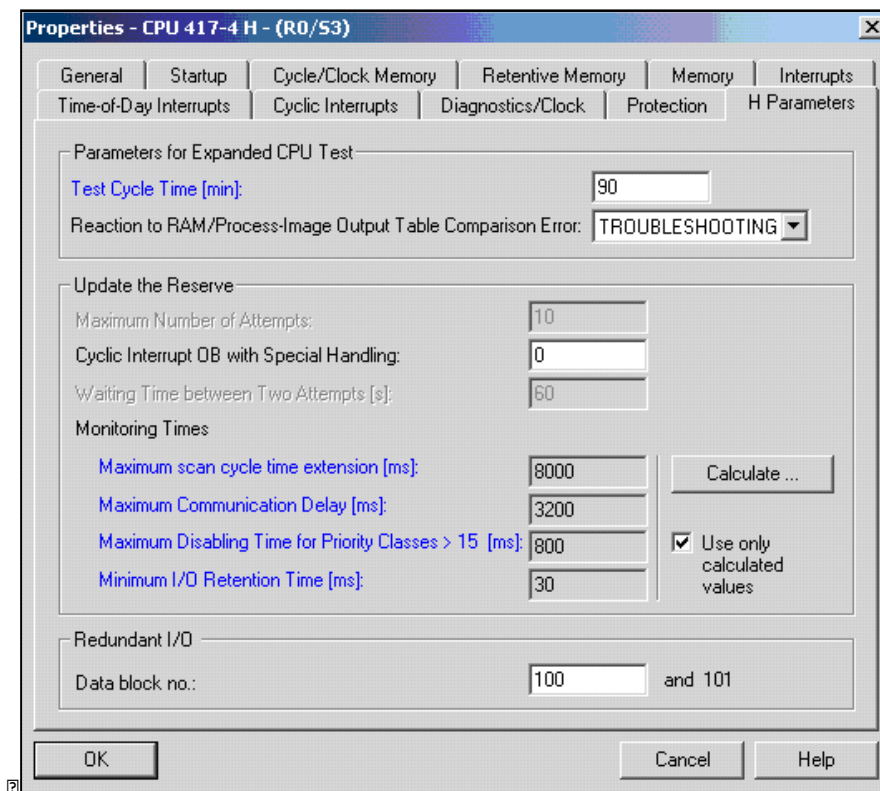


图2—8 “H Parameter”参数配置

2.3.4 配置 ET200M 站

(1)点击 DP 总线“master: DP master system(1)”,在硬件目录 PROFIBUS DP 下, 选择一个 IM153-2 的站点, 双击添加一个 ET200M 站。

(2)设定 ET200M 站的地址。

(3)在 ET200M 站上添加 I/O 模块。

从站配置后的如图 2—9 所示:

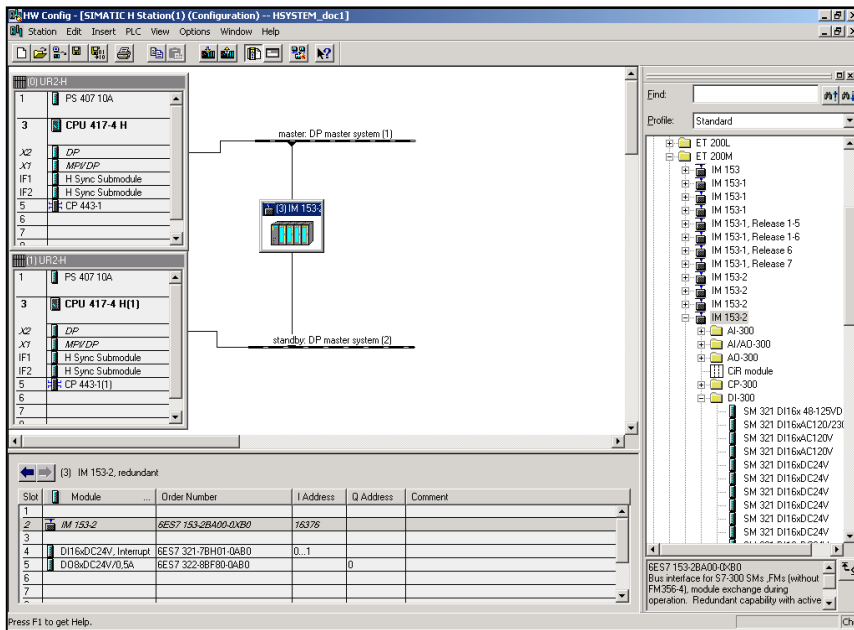


图2-9 配置ET200M站

2.3.5 配置 Y-Link 耦合器

- (1) 点击 DP 总线“master: DP master system(1)”，在硬件目录的 DP/PA Link 下选择 IM157，并双击添加一个站。
- (2) 设定 Y-Link 的站地址
- (3) 选择将 Y-Link 设置为一个 DP/DP 耦合器或 DP/PA 耦合器如图 2-10 所示：

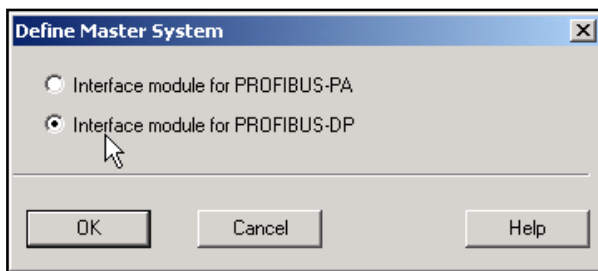


图2-10 选择Y-Link的类型

- (4) 在 Y-Link 后的 PROFIBUS 总线上添加单一总线接口的从站站点，如：Masterdrive 等。组态完成的界面如图 2-11 所示：

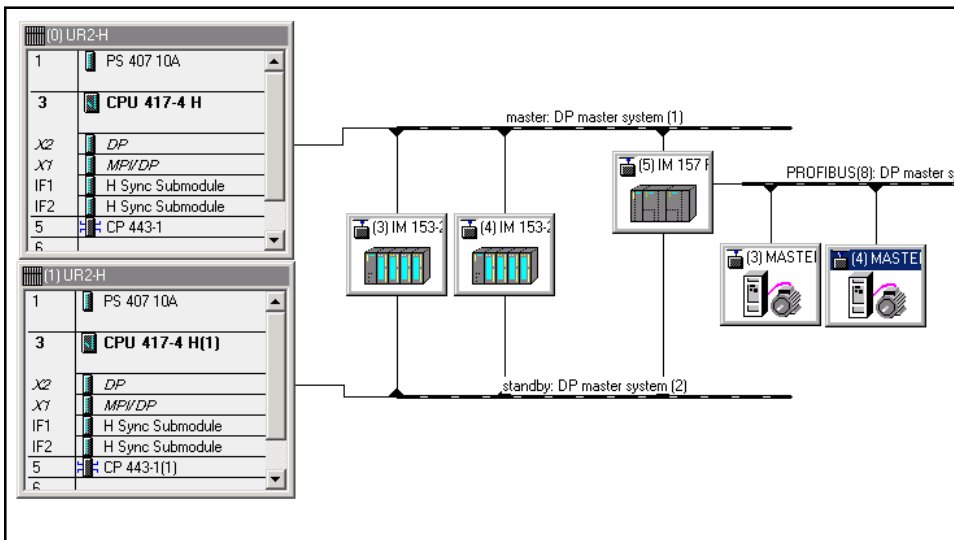


图2—11 Y-Link配置图

注意:

在修改硬件配置后或退出 HW Config 之前一定要进行编译。

2.3.6 添加错误诊断 OB 块

以下错误 OB 块必须装入 S7-400H 的 CPU 中, OB70、OB72、OB80、OB82、OB83、OB85、OB86、OB87、OB88、OB121 和 OB 122。如果没有装载这些 OB, H 系统

在出现错误时可能会进入 STOP 状态, 这些 OB 块另一个功能可以对事件信息进行诊断, OB 块的诊断功能见表 2—1 所示。

表 2—1 H 系统 OB 块的上面

OB no.	故障/出错类型	故障原因	操作系统的响应	故障指示灯
OB70	I/O 冗余故障	<ul style="list-style-type: none"> · PROFIBUS DP 冗余故障 · PROFIBUS DP 切换 	OB 70 调用 如果没有装载 OB CPU 不会进入 STOP 方式	
OB72	CPU 冗余故障	<ul style="list-style-type: none"> · CPU 冗余故障 · 主从切换 · 同步故障 · 同步模块故障 · 更新取消 · 校验错误 (例如 RAM, PIQ) 	OB 72 调用 如果没有装载 OB CPU 进入 STOP 方式	

OB80	定时错误	<ul style="list-style-type: none"> • 用户程序 OB1 和所有中断以及出错 OB 超过规定最大循环时间 • OB 请求出错 • 起始信息缓冲区溢出 • 时间出错中断 	<p>"INTF"指示灯亮直到故障被清除</p> <p>OB 80 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	INTF
OB82	诊断中断	具有中断能力的一个 I/O 模板报告一个诊断中断	<p>OB 82 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	EXTF
OB83	插/拔中断	插入一个电源模板或拔出一个电源模板并插入一个不正确的模板类型，如果在缺省参数设置的 CPU 处于 STOP 方式时只拔出所插入的电源模板，EXTF 指示灯不亮，如果没有再插入电源模板指示灯将闪亮	<p>OB 83 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	EXTF
OB85	优先级错误	<ul style="list-style-type: none"> • 调用了优化级但没有相应的 OB • 对于 SFB 调用背景数据块丢失或故障 	<p>OB 85 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	INTF
		<ul style="list-style-type: none"> • 在过程映象更新过程中出错 		EXTF
OB86	机架/站故障	<ul style="list-style-type: none"> • 扩展机架中的电源故障 • DP 线路故障 • 耦合线路故障，接口模板丢失或故障线路中断 	<p>OB 86 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	EXTF
OB87	通讯错误	<ul style="list-style-type: none"> • 状态信息不能被输入到数据块中 • 不正确的帧标识符 • 帧长度出错 • 非法的全局标识号 • 数据块访问出错 	<p>OB 87 调用</p> <p>如果没有装载 OB CPU 进入 STOP 方式</p>	INTF

OB88	取消处理	一个程序块的处理被取消取消的可能原因是 <ul style="list-style-type: none"> • 嵌套深度太深 • 主站控制继电器嵌套深度太深 • 同步故障嵌套深度太深 • 块调用 i 栈嵌套深度太深 • 块调用 b 栈嵌套深度太深 • 本地数据分配出错 • 未知指令 • 带有名称的子指令超出块范围 	OB 88 调用如果没有装载 OB CPU 进入 STOP 方式	INTF
OB121	编程错误	用户程序中的机器代码出错 <ul style="list-style-type: none"> • BCD 转换错误 • 范围长度出错 • 范围出错 • 调整出错 • 写出错 • 计时器编号出错 • 计数器编号出错 • 块编号出错 • 块未装入 	OB 121 调用如果没有装载 OB CPU 进入 STOP 方式	INTF
OB122	存取出错	模板故障 SM FM CP	EXTF 指示灯亮直到故障被清除	EXTF

2.4 存储器复位

下载用户程序前最好将存储器复位，步骤如下：

- (1) 设置模式选择器至 STOP 设置

结果：STOP 指示灯亮

- (2) 设置模式选择器到 MRES 位置并保持

结果：STOP 指示灯将熄灭 1 秒钟亮 1 秒钟，然后保持点亮状态。

- (3) 将模式选择器置于 STOP 位置，然后在接下来的 3 秒内拨至 MRES，然后再拨回 STOP 位置

结果：STOP 指示灯以 2 Hz 至少闪亮 3 秒钟，执行存储器复位然后持续点亮。

2.5 STEP7 程序下装

- (1) 点击 SIMATIC MANAGER->OPTION->SET PG/PC,选择相应的通讯接口。

(2) 在 SIMATIC MANAGER->PLC 下选择“Download”将用户程序装入 CPU0 中。在同步连接建立后，CPU0 中的用户程序通过同步光纤自动传送到备份 CPU1 中。

(3) 将模式选择器开关拨到 RUN-P 位置启动 S7-400H。首先启动 CPU0，然后启动 CPU1。结果 CPU0 作为主 CPU 启动，CPU1 作为热备 CPU 启动。在热备 CPU 建立同步链接并更新缓存区数据后，S7-400H 转换到冗余工作方式并执行用户程序。

第三章 S7-400H 系统通讯

3.1 S7-400H 与 S7-400H 之间的通讯

3.1.1 H 系统之间的通讯链路

冗余通讯增强互连系统之间可用性最简单的办法是采用冗余系统网络，当一条网络链路损坏时，系统可自动切换到另外一条网络链路。其网络的连接可以根据需要组成如下不同的网络结构。

(1)在 H 系统的每个机架上分别插入 1 块 CP443-1 组成双以太网，如下图 3-1 所示：

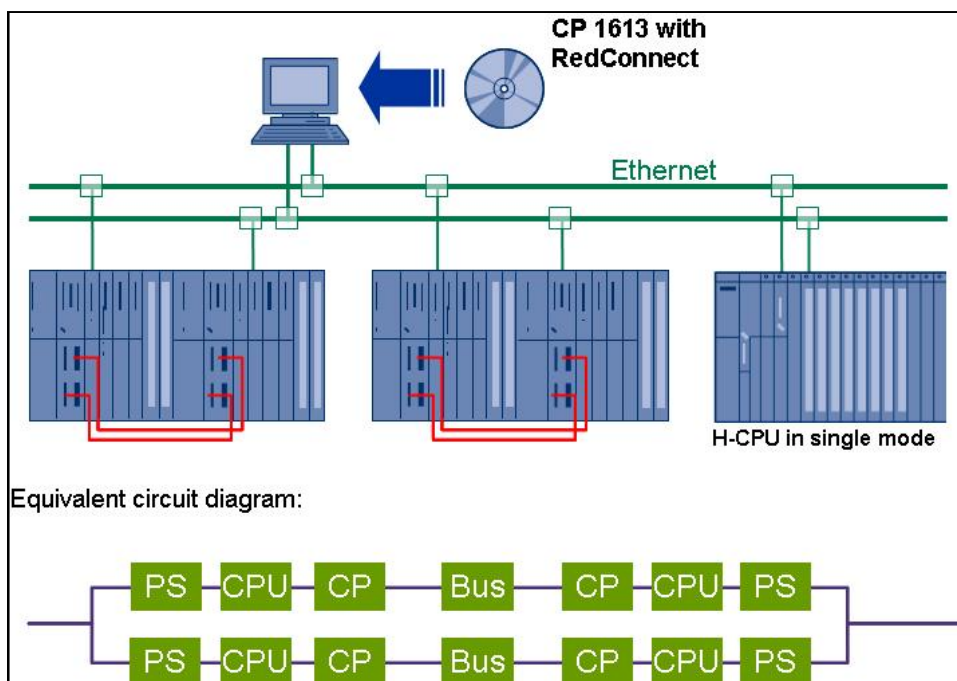


图 3-1 H 系统双以太网结构(一)

(2)在 H 系统的每个机架上分别插入 2 块 CP443-1 组成双以太网，如图 3-2 所示：

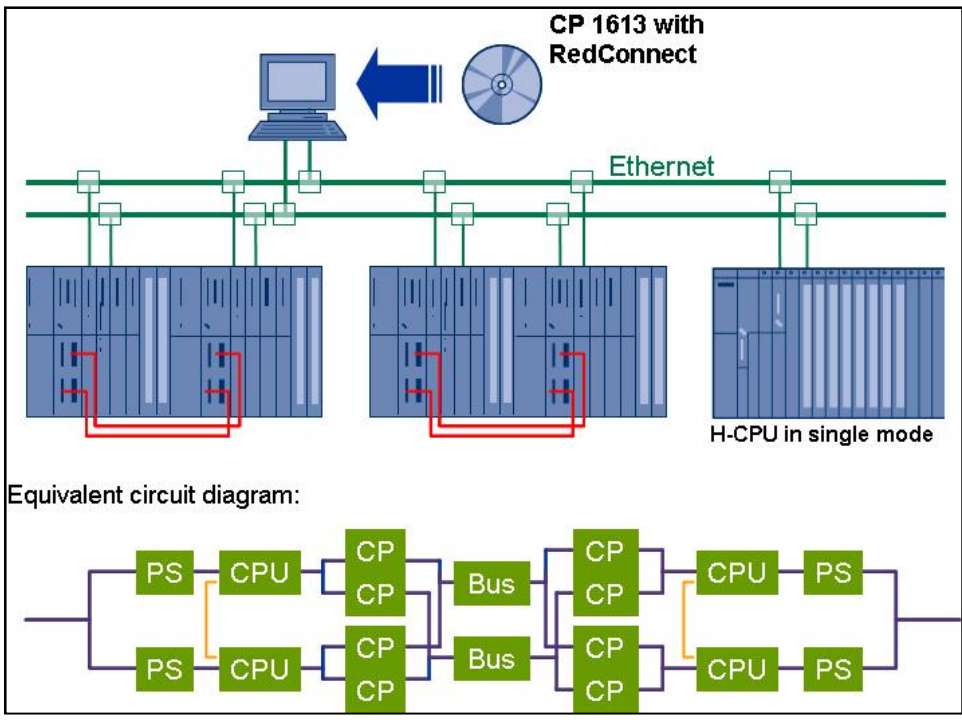


图 3-2 H 系统双以太网结构(二)

(3)在 H 系统的每个机架上分别插入 1 块 CP443-1 组成单以太网,如图 3-3 所示:

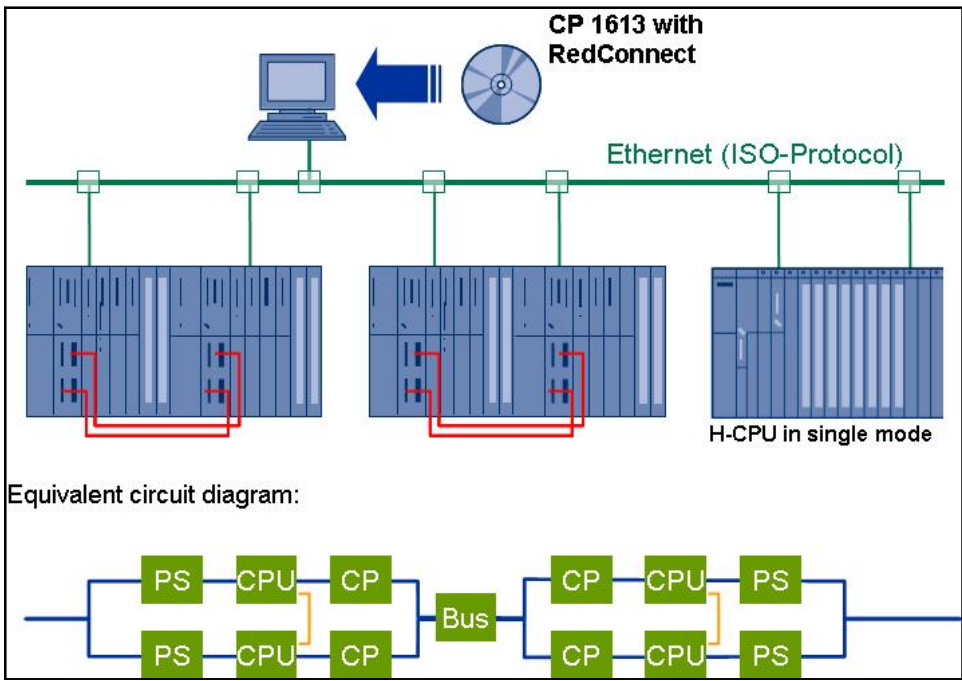


图 3-3 H 系统以太网结构(三)

(4)在 H 系统的每个机架上分别插入 1 块 CP443-1 组成以太环网,如图 3-4 所示:

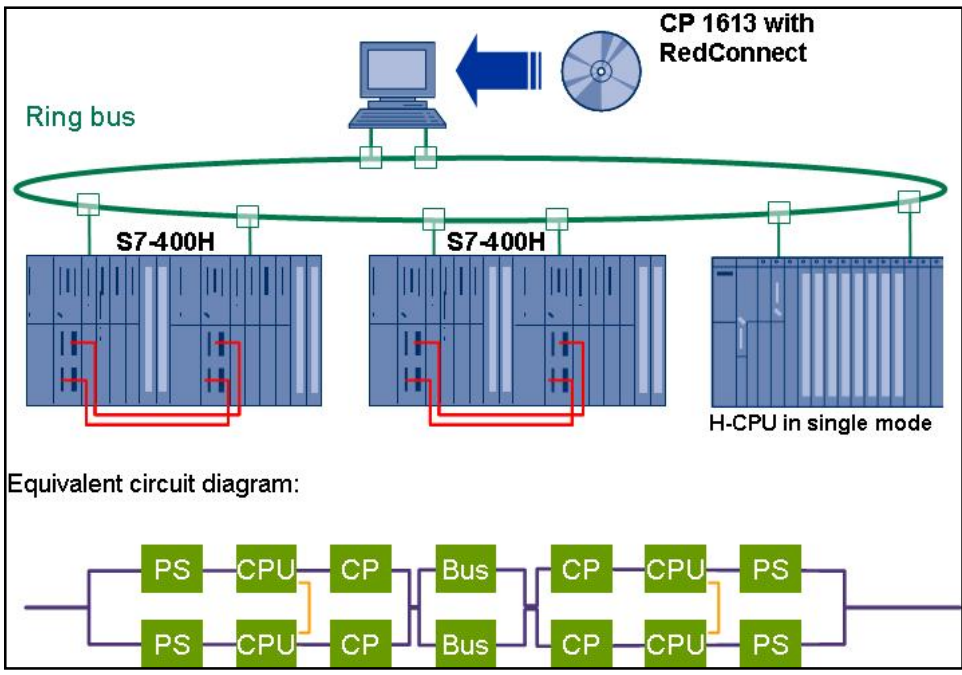


图 3-4 H 系统以太网结构(四)

(5)在 H 系统的每个机架上分别插入 1 块 CP443-1 组成以太双环网,如图 3-5 所示:

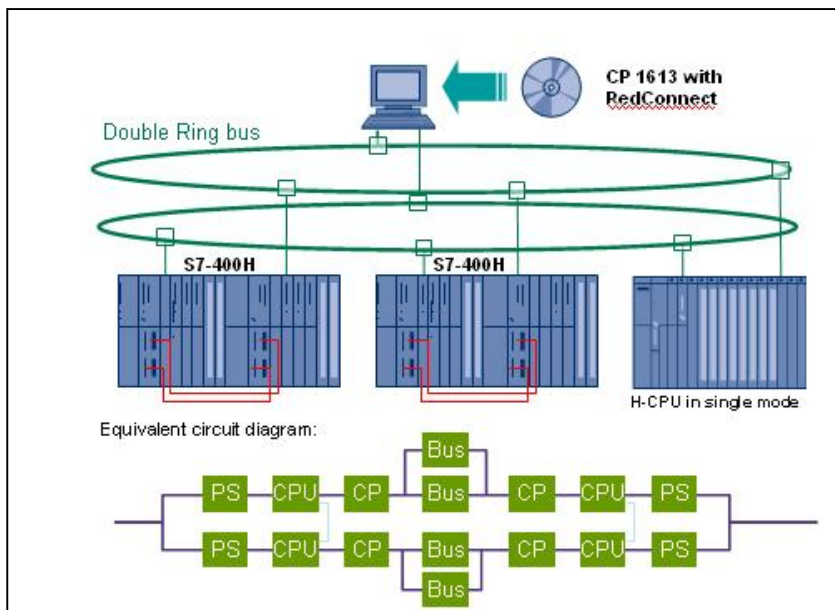


图 3-5 H 系统以太网结构(五)

(6)在 H 系统的每个机架上分别插入 2 块 CP443-1 组成双以太网双环网,如图 3-6 所示:

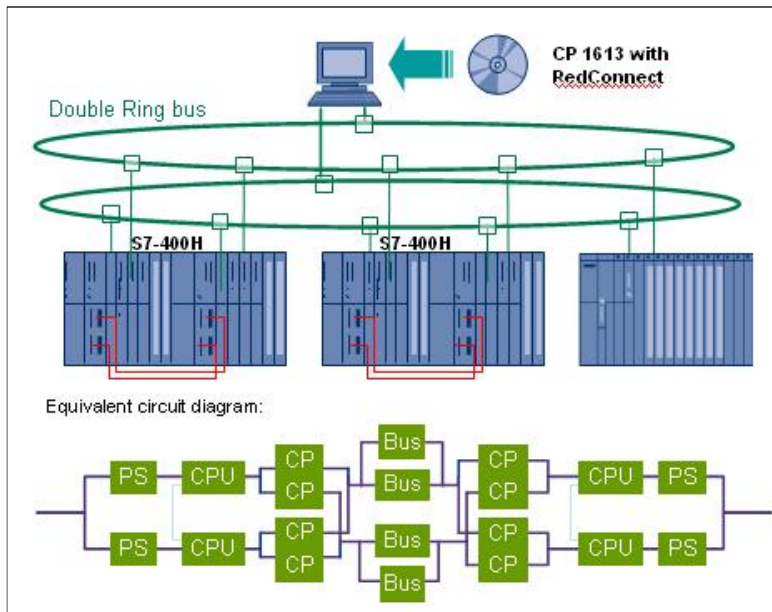
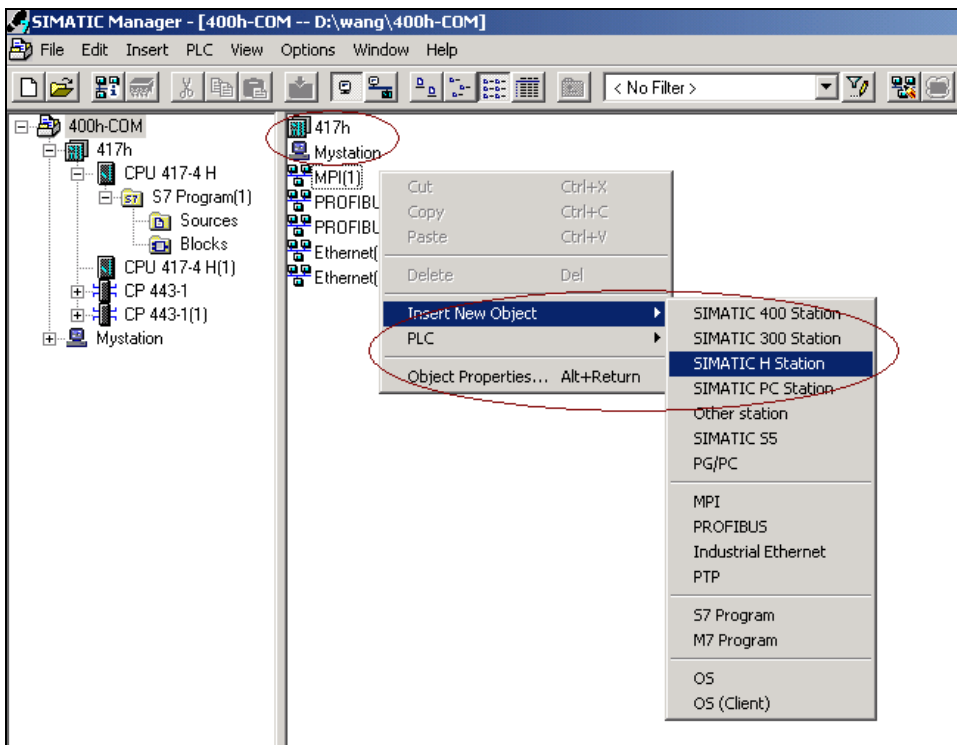


图 3-6 H 系统以太网结构(六)

3.1.2 H 系统之间的通讯配置

以图 3-4 为例,即 H 系统的每个机架上分别插入 1 块 CP443-1 组成以太网环网,来介绍 H 系统的通讯配置。

(1)运行 SIMATIC Manager 来创建一个新的项目,示例中为 400H-COM。插入一个新的 H 站。示例中将其命名为 417h,如图 3-7 所示。



图

3-7 建立417h站

(2) 通过双击硬件或使用右键弹出菜单, 打开 417h 的硬件组态 (HW Config), 根据实际硬件进行组态。如图3-8所示。

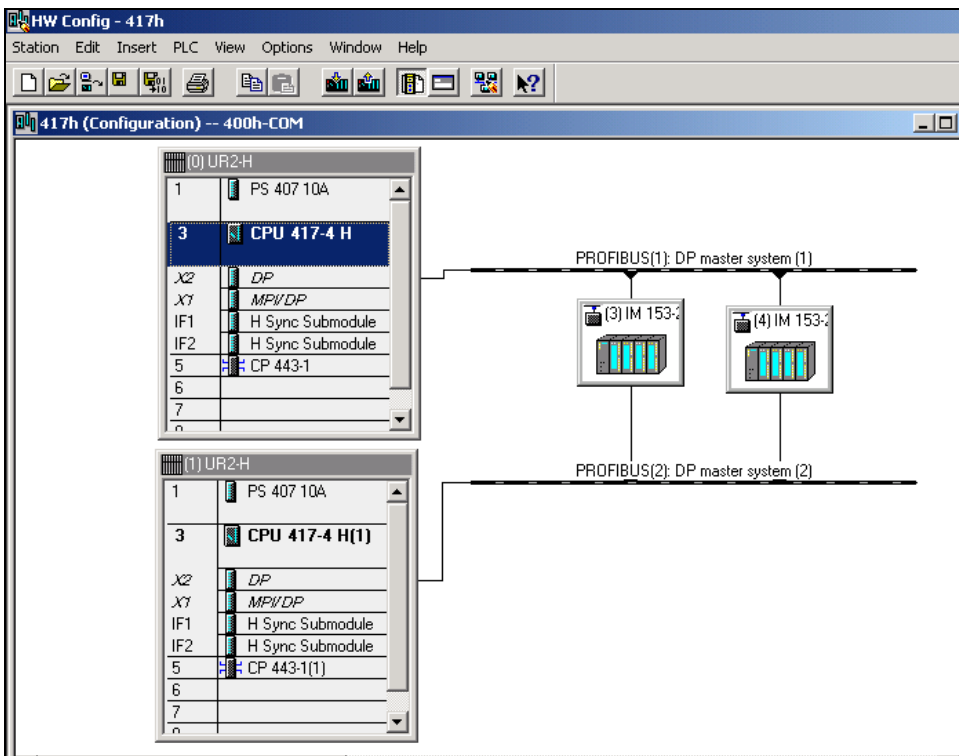


图3

-8 417h站的硬件组态

(3)通过分别右击CP443-1和CP443-1(1)“Object properties”来建立以太网,并配置CP443-1属性,如图3-9所示:

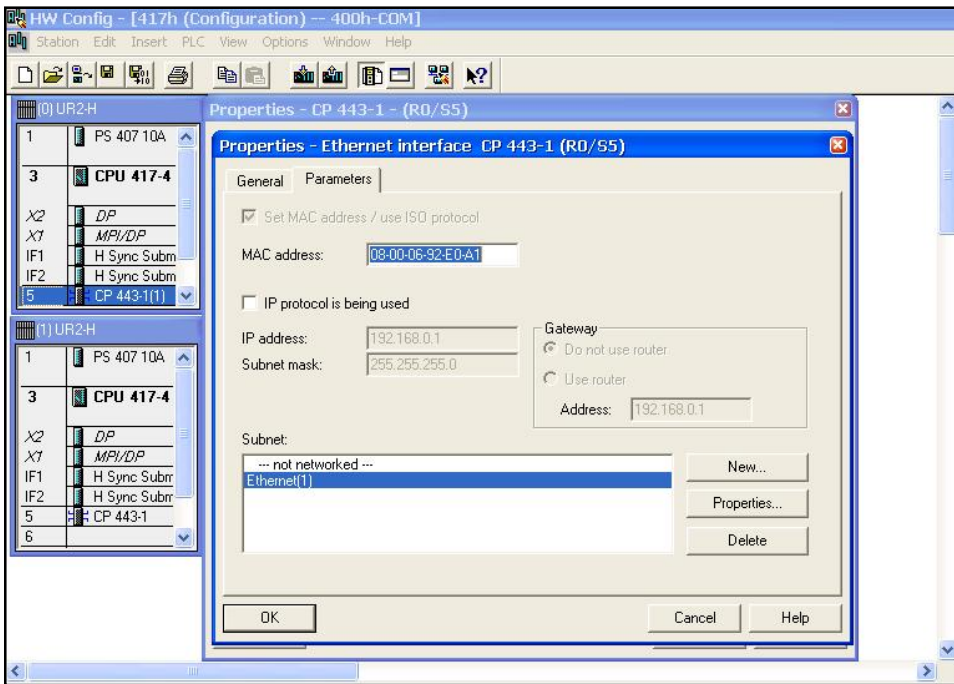



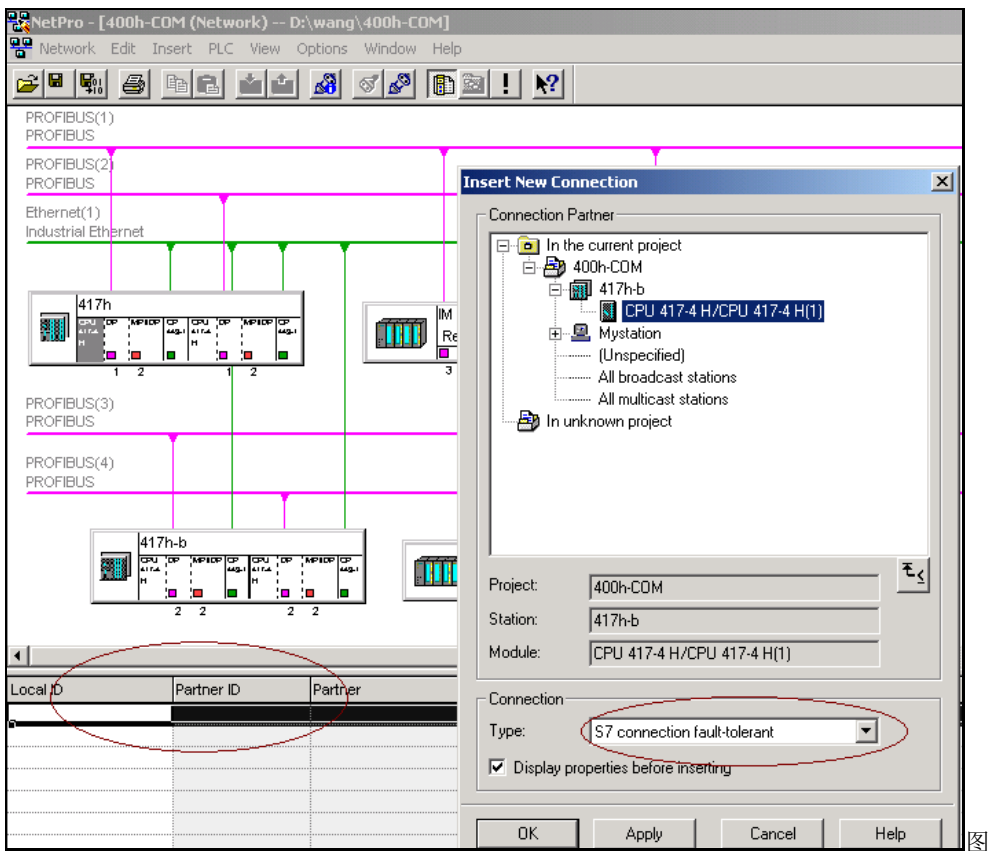
图3 -9 建立以太网并配置CP443-1属性

(4)组态完毕后，可进行编译并通过MPI或以太网下载到CPU中。

(5)通过如上步骤建立第二个冗余站417h-b。

(6)H系统的网络组态

在 417-4h 和 417h-b 组态完毕之后，则可进入网络组态。通过点击“Configure Network”图标  进入网络组态界面。选择 417h 的 CPU,并在下栏的网络连接里点击鼠标右键插入新的连接“S7 Connection fault tolerant”连结，如图 3-11 所示。



3-11 建立网络连接

这样 H 系统之间就建立了共 4 条通讯链路，如图 3-12-所示。

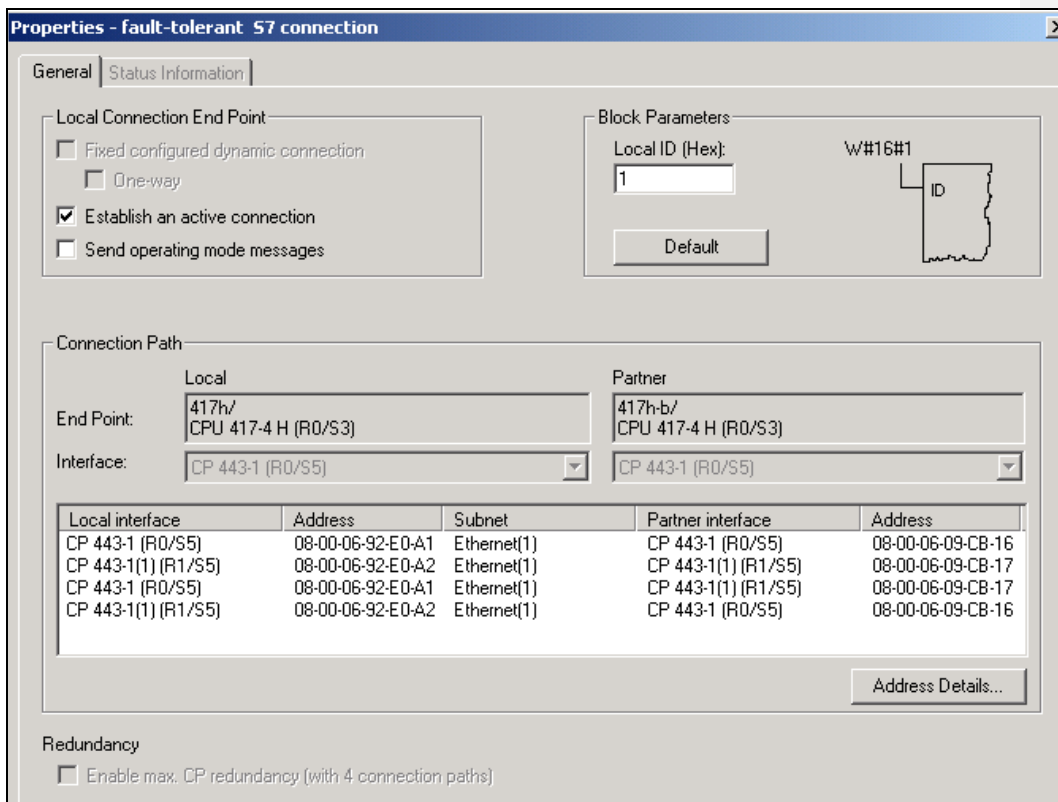


图3-12 H系统之间4条通讯链路

(7)存盘编辑网络连接，如图 3-13 所示：

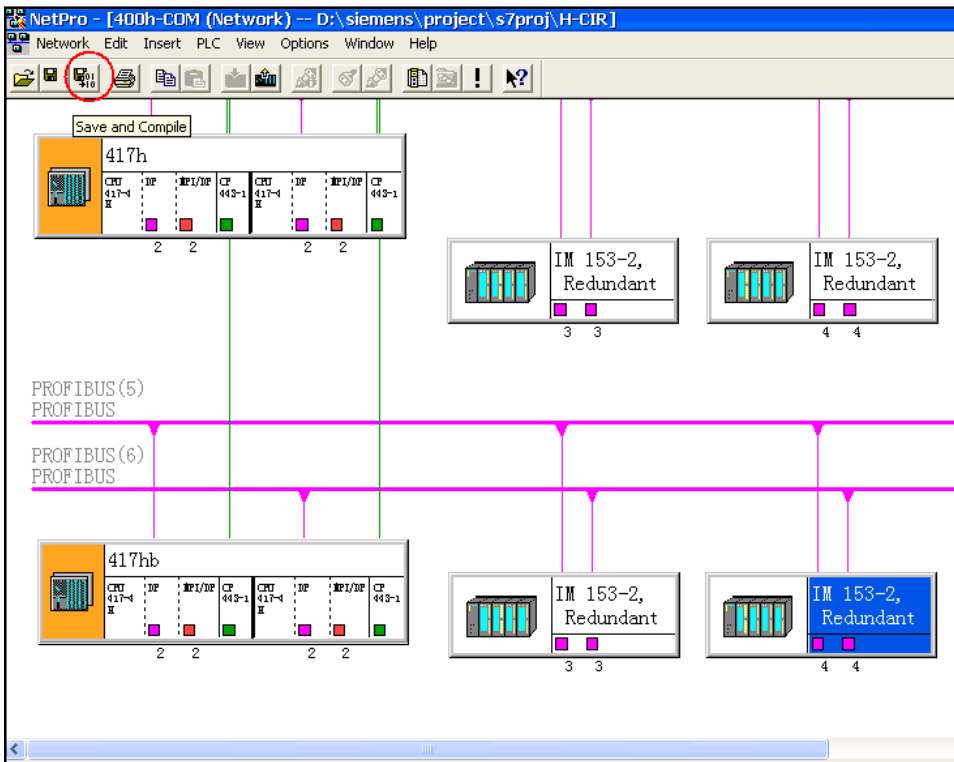


图 3-13 存盘编辑网络连接

(8)分别下载417h站和417hb的网络连接，如图3-14所示：

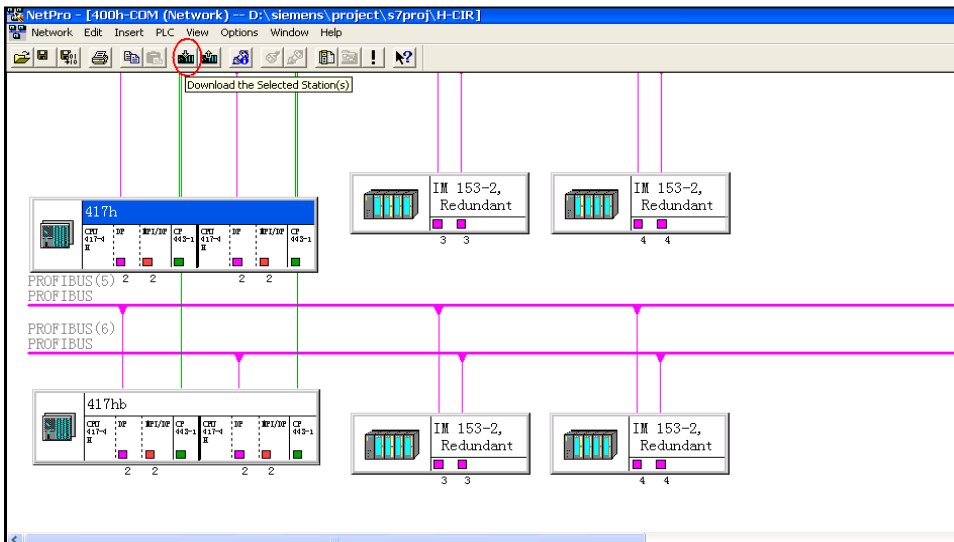


图 3-14 分别下载 417h 站和 417hb 的网络连接

3.1.3 H 系统通讯功能块的调用

H 系统之间可以通过 BSEND/BRCV 实现通讯数据之间安全的数据传送，也可以通过 USEND/URCV 实现通讯数据之间快速的、无确认的数据传送，用 PUT/GET 亦可实现只在单边编程就可以读/写通讯对象的数据。以下系统功能和功能块可以适用于 H 系统之间的通讯。

(1) 传送和接收功能

SFB 8 USEND	向通讯对象发送数据 (无确认)
SFB 9 URCV	从通讯对象接收数据 (无确认)
SFB 12 BSEND	向通讯对象发送数据 (有确认)
SFB 13 BRCV	从通讯对象接收数据 (有确认)
SFB 14 GET	从通讯对象读取数据
SFB 15 PUT	向通讯对象写数据

(2) 控制功能

SFB 19 START	使通讯对象重启
SFB 20 STOP	使通讯对象停止
SFB 21 RESUME	使通讯对象继续工作

(3) 监控功能

SFB 22 STATUS	提供通讯对象的工作状态
SFB 23 USTATUS	接收通讯对象的工作状态

(4) 查询功能

SFC 62 CONTROL	查询连接的状态
----------------	---------

以 BSEND/BRCV 为例，简单介绍 H 系统之间的数据交换。

在 417H 站中调用 SFB12

CALL "BSEND", DB12

REQ :=M10.0

R :=M10.1

```
ID :=W#16#1
R_ID :=DW#16#1
DONE :=M1.0
ERROR :=M1.1
STATUS:=MW20
SD_1 :=IB0
LEN :=MW8
```

在 417H-B 站中调用 SFB13

```
CALL "BRCV", DB12
EN_R :=M100.0
ID :=W#16#1
R_ID :=DW#16#1
NDR :=M20.0
ERROR :=M20.1
STATUS:=MW22
RD_1 :=QB0
LEN :=MW0
```

参数描述如下:

REQ: 沿触发。

EN_R: 为 1 时接收

R_ID: 区分双字, 用于与同一连接其他块区分, 发送与接收必须一致。

SD_I、RD_I: 发送区与接收区。

ERROR: 错误位。

STATUS: 通讯状态字。

这样, 417h 站 IB0 的数据就可以传送到 417hb 站的 QB0。

3.2 S7-400H 与标准 CPU 之间的通讯

3.2.1 H 系统与标准 CPU 的通讯链路

H 系统与标准 CPU 的通讯有两种网络连接方式：一种是通过以太网通讯，如图 3-14 所示，另一种通过 Y-Link 连接到 PROFIBUS 网络，如图 3-15 所示。

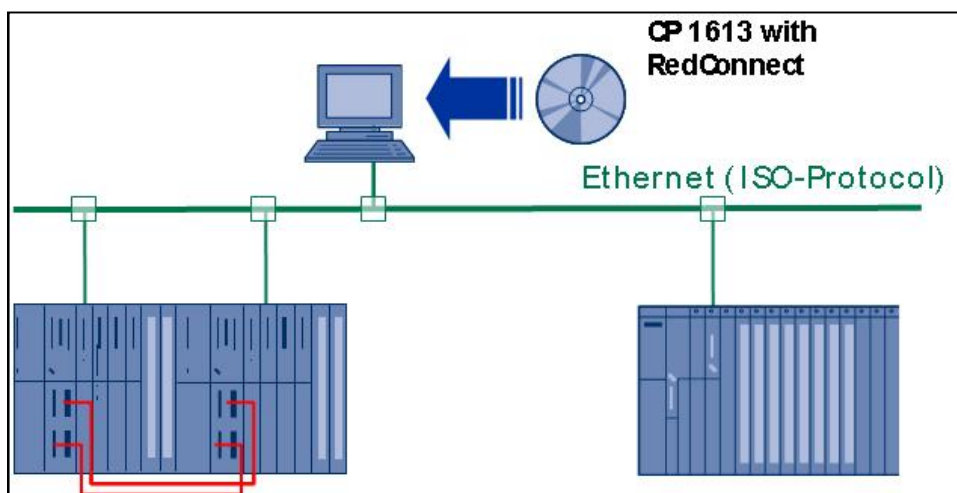


图 3-14 H 系统与标准 CPU 的以太网通讯链路

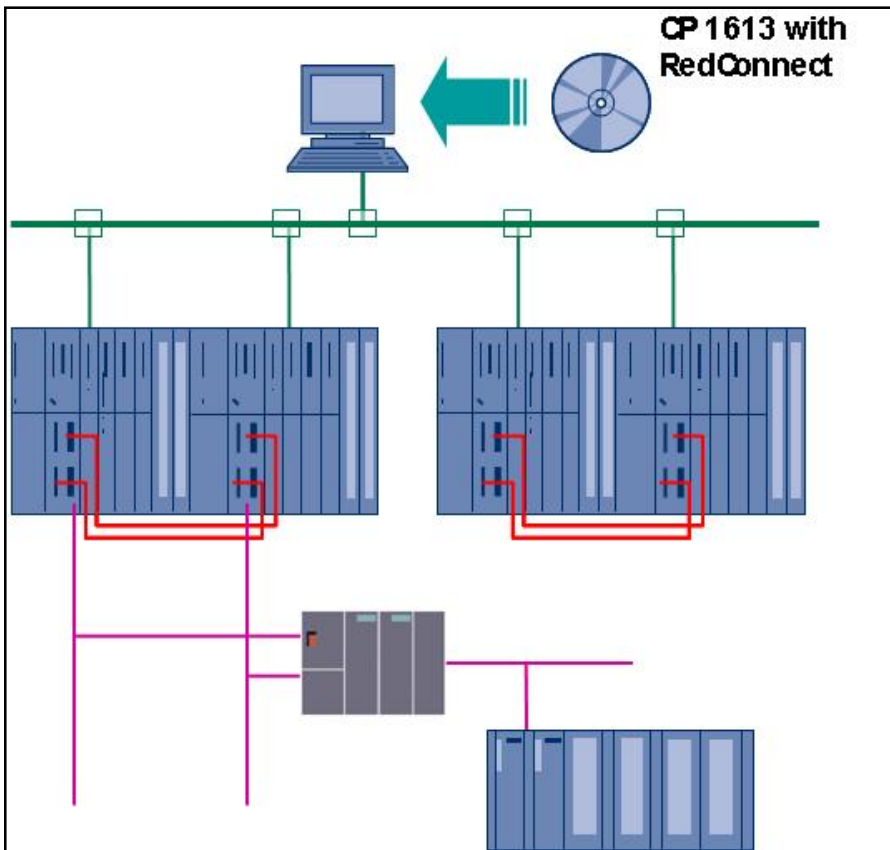


图 3-15 H 系统与标准 CPU 的 Profibus 通讯链路

3.2.2 S7-400H 与标准 CPU 之间的通讯配置

S7-400H 与带有 CP 模板的标准 CPU 之间的 Profibus 通讯配置可参考 Profibus 主从通讯配置; 如果 S7-400H 与标准 CPU 之间是通过与标准 CPU 集成的 DP 口通讯时,则需要标准 CPU 的 Profibus-dp 的 GSD 文件,此时可以把标准 CPU 当作 Profibus 从站进行配置即可,对于 S7-400H 与标准 CPU 之间的 Profibus 通讯配置本文就不再描述,

H CPU 与标准 CPU 之间的以太网通讯配置和 S7-400H 与 S7-400H 之间的通讯配置类似。先对 S7-400H 和标准 CPU 硬件组态完成后,则可进入网络组态。通过选择菜单和图标进入网络组态的界面。在进入网络组态后,点击鼠标右键插入新的连接“S7 Connection”连接,插入连接的步骤图 3-16 所示。

Kommentar [c1]:

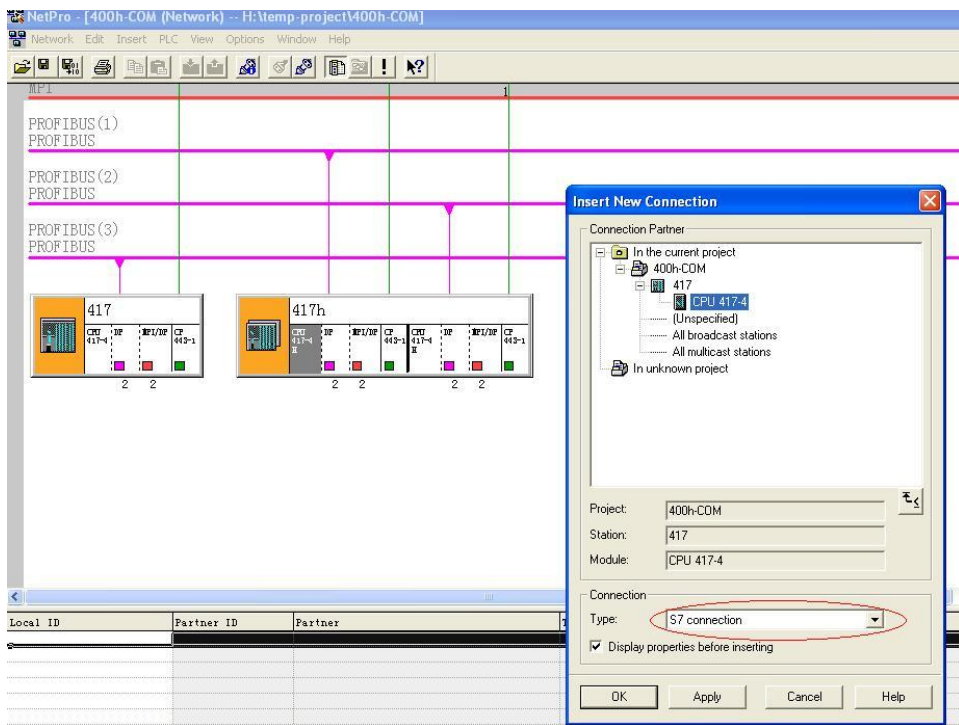


图3-16 H CPU与标准CPU之间的以太网通讯配置

这样 S7-400H 与标准 CPU 之间的通讯就建立，连接共 1 条通道，如图 3-17、3-18 所示。

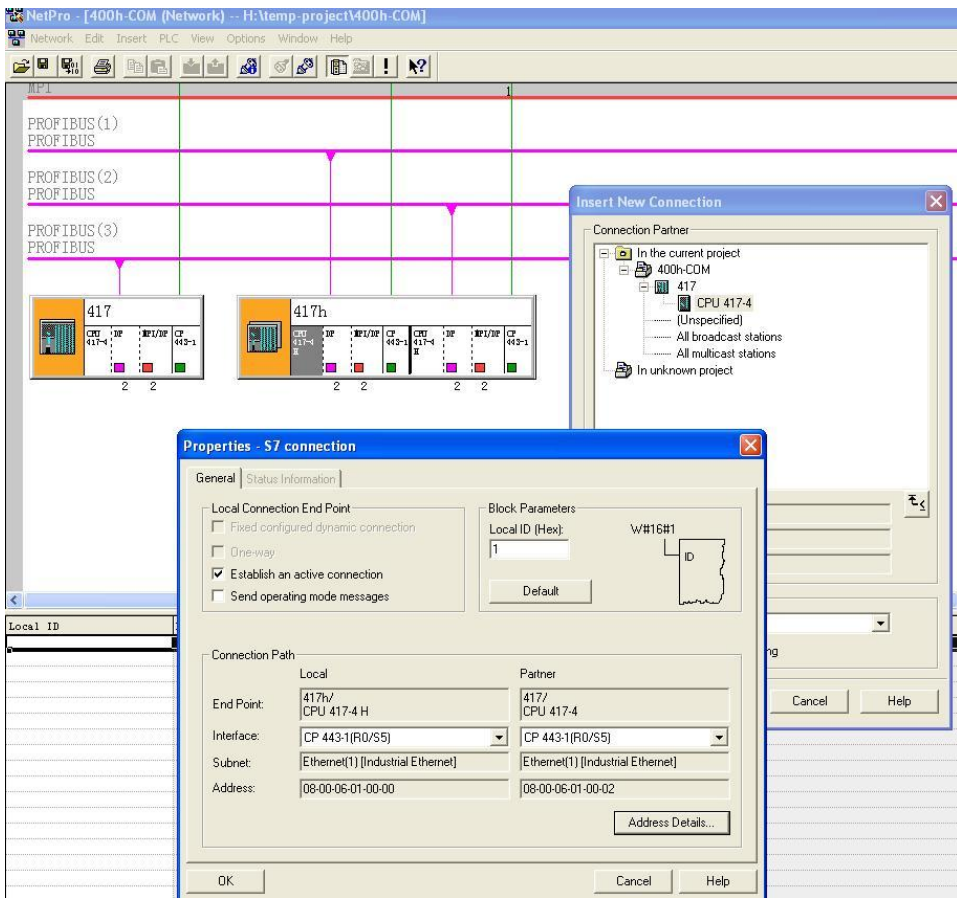


图 3-17 S7-400H 与标准 CPU 之间的以太网链路

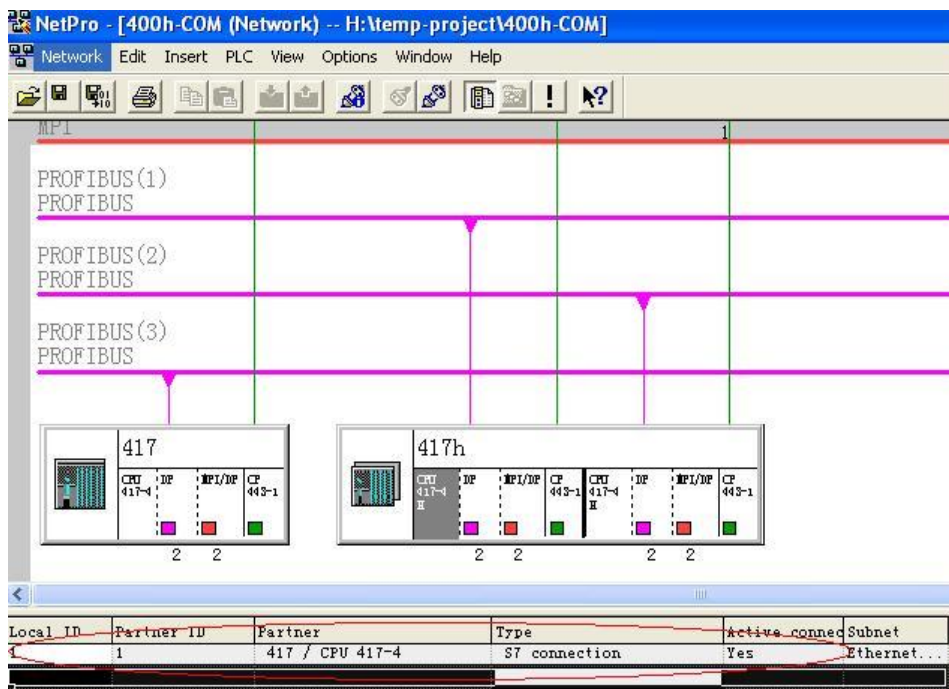


图 3-18 S7-400H 与标准 CPU 之间的以太网通讯连接

3.2.3 S7-400H 与标准 CPU 通讯功能块的调用

H 系统之间可以通过 BSEND/BRCV 实现通讯数据之间安全的数据传送，也可以通过 USEND/URCV 实现通讯数据之间快速的、无确认的数据传送，用 PUT/GET 亦可实现只在单边编程就可以读/写通讯对象的数据。以下系统功能和功能块可以适用于 S7-400H 与标准 CPU 之间的通讯：

(1) 传送和接收功能

SFB 8 USEND 向通讯对象发送数据 (无确认)

SFB 9 URCV 从通讯对象接收数据 (无确认)

SFB 12 BSEND 向通讯对象发送数据 (有确认)

SFB 13 BRCV 从通讯对象接收数据 (有确认)

SFB 14 GET 从通讯对象读取数据

SFB 15 PUT 向通讯对象写数据

(2) 控制功能

SFB 19 START 使通讯对象重起

SFB 20 STOP 使通讯对象停止

SFB 21 RESUME 使通讯对象继续工作

(3)监控功能

SFB 22 STATUS 提供通讯对象的工作状态

SFB 23 USTATUS 接收通讯对象的工作状态

(4)查询功能

SFC 62 CONTROL 查询连接的状态

以下以 BSEND/BRCV 为例，简单介绍 H 系统之间的数据交换。

在 417H 站中调用 SFB12

```
CALL "BSEND", DB12
```

```
REQ :=M10.0
```

```
R :=M0.1
```

```
ID :=W#16#1
```

```
R_ID :=DW#16#1
```

```
DONE :=M1.0
```

```
ERROR :=M1.1
```

```
STATUS:=MW20
```

```
SD_1 :=IB0
```

```
LEN :=MW8
```

在 417 站中调用 SFB13

```
CALL "BRCV", DB12
```

```
EN_R :=M100.0
```

```
ID :=W#16#1
```

```
R_ID :=DW#16#1
```

```
NDR :=M20.0
```

```
ERROR :=M20.1
```

STATUS:=MW22

RD_1 :=QB0

LEN :=MW0

参数描述

REQ: 沿触发。

EN_R: 为 1 时接收

R_ID: 区分双字, 用于与同一连接其他块区分, 发送与接收必须一致。

SD_I、RD_I: 发送区与接收区。

ERROR: 错误位。

STATUS: 通讯状态字。

这样, 417H 站 IB0 的数据就可以传送到 417B 站的 QB0。

注意: 由于 S7-400H 与标准 CPU 之间不能建立“S7 connection fault-tolerant”链接, 标准 CPU 的 CP443-1 通讯卡只能与 S7-400H 的一个 CP443-1 通讯卡建立链接, 那么当 S7-400H 侧建立链接的 CP443-1 出现故障、网络中断或 CPU 故障时, 将无法通讯。但是 S7-400H 与单个冗余 CPU 之间则可以建立容错链接, 因此建议在需要单 CPU 和冗余 CPU 之间通讯的情况下, 建议为单 CPU 配置单个的冗余 CPU 型号。

Kommentar [c2]:

3.3 H 系统与 WinCC 的通讯

3.3.1 H 系统与 WinCC 连接

H 系统与 WinCC 的通讯可根据不同要求采用单路径、双路径、四路径连接等三种方式。

(1) H 系统与 WinCC 的单路径连接(如图 3-19)

H 系统与 WinCC 的单路径连接需要的软、硬件:

1. SIMATIC H 站, 两机架每个机架 1 个通讯卡 (2 CP)。
2. PC 机插 1 块 CP1613 工业以太网卡。
3. STEP7 V5.3 (或 STEP7 V5.2 加冗余软件包), 1 套
4. SIMATIC NET 软件 V6.2 SP1, 1 套
5. WinCC V6.0 SP2, 1 套
6. S7-REDCONNECT V6.2, 1 套

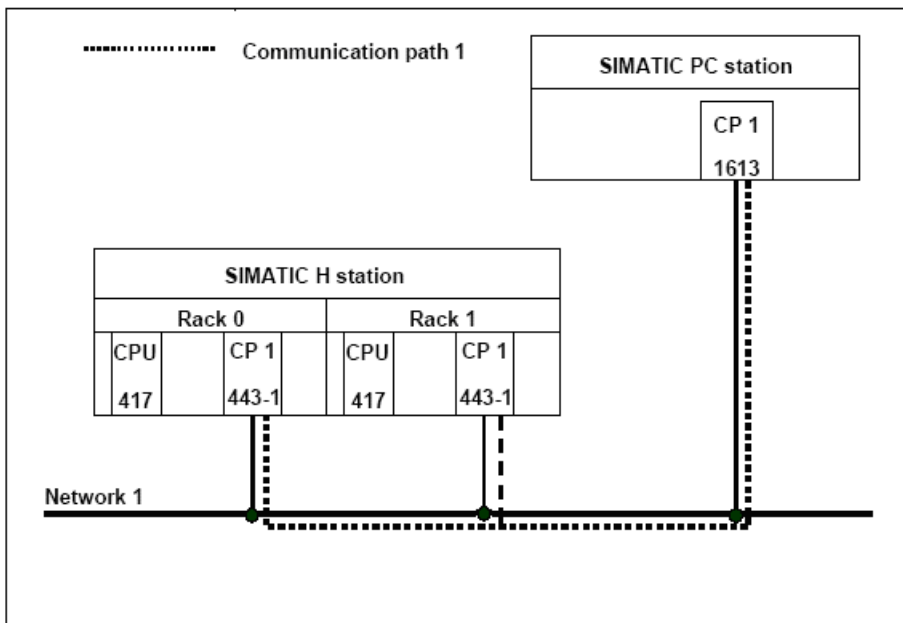


图 3-19 H 系统与 WinCC 的单路径连接

(2) H 系统与 WinCC 的双路径连接,如图 3-20 所示:

H 系统与 WinCC 的双路径连接需要的软、硬件:

1. SIMATIC H 站, 两机架每个机架 1 个通讯卡 (2 CP)。
2. PC 机插 2 块 CP1613 工业以太网卡。
3. STEP7 V5.3 (或 STEP7 V5.2 加冗余软件包), 1 套
4. SIMATIC NET 软件 V6.2 SP1, 1 套
5. WinCC V6.0 SP2, 1 套
6. S7-REDCONNECT V6.2, 1 套

这种连接采用双路径, 任一器件的损坏, 系统可以自动从另一路径进行通讯。

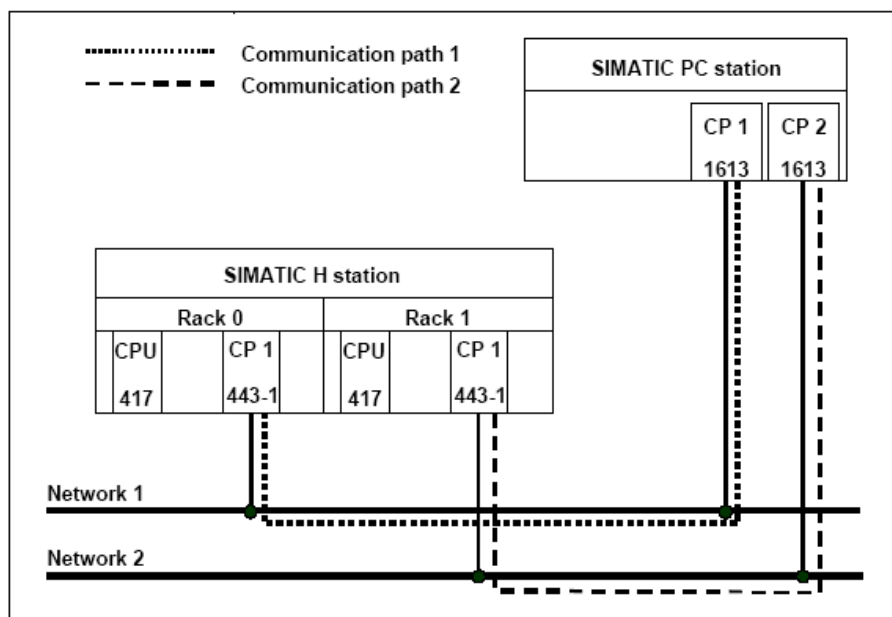


图 3-20 H 系统与 WinCC 的双路径连接

(3) H 系统与 WinCC 的四路径连接，如图 3-21 所示

H 系统与 WinCC 的四路径连接需要的软、硬件：

1. SIMATIC H 站，两机架每个机架 2 个通讯卡（4 CP）。
2. PC 机插 2 块 CP1613 工业以太网卡。
3. STEP7 V5.3 (或 STEP7 V5.2 加冗余软件包), 1 套
4. SIMATIC NET 软件 V6.2 SP1, 1 套
5. WinCC V6.0 SP2, 1 套
6. S7-REDCONNECT V6.2, 1 套

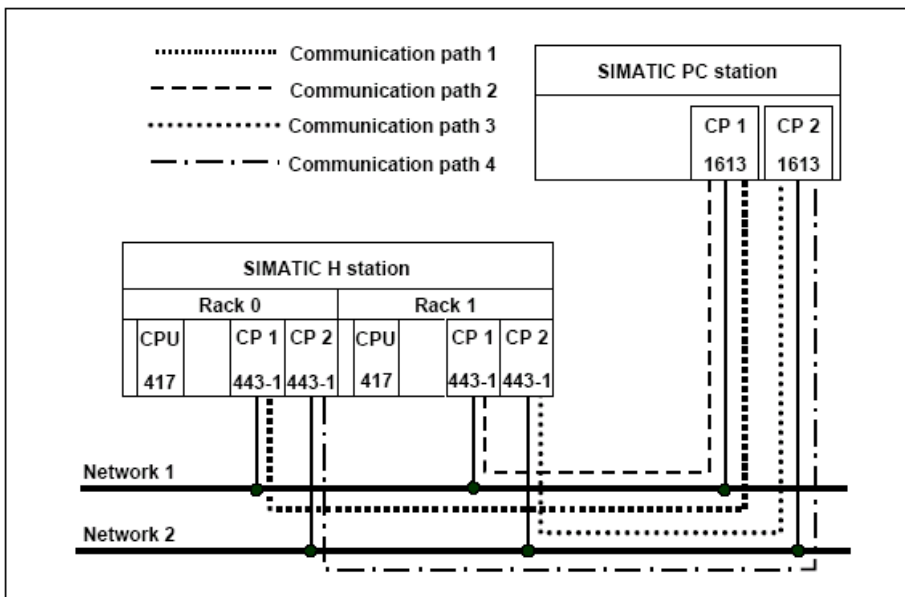


图 3-21 系统与 WinCC 的四路径连接

3.3.2 组态通讯过程

下面我们以单路径通讯为例,即H系统与单CP1613的PC站通过单以太网段进行通讯的例子。双路径或四路径的通讯可参考此例。

- (1) 首先组态400H, 400H站组态的步骤和3.1.2一样, 示例中项目名称为400h-COM。400H站命名为417h。

组态完毕后, 可进行编译并通过MPI或以太网下载到CPU中。

- (2) 向项目中插入一个PC站。

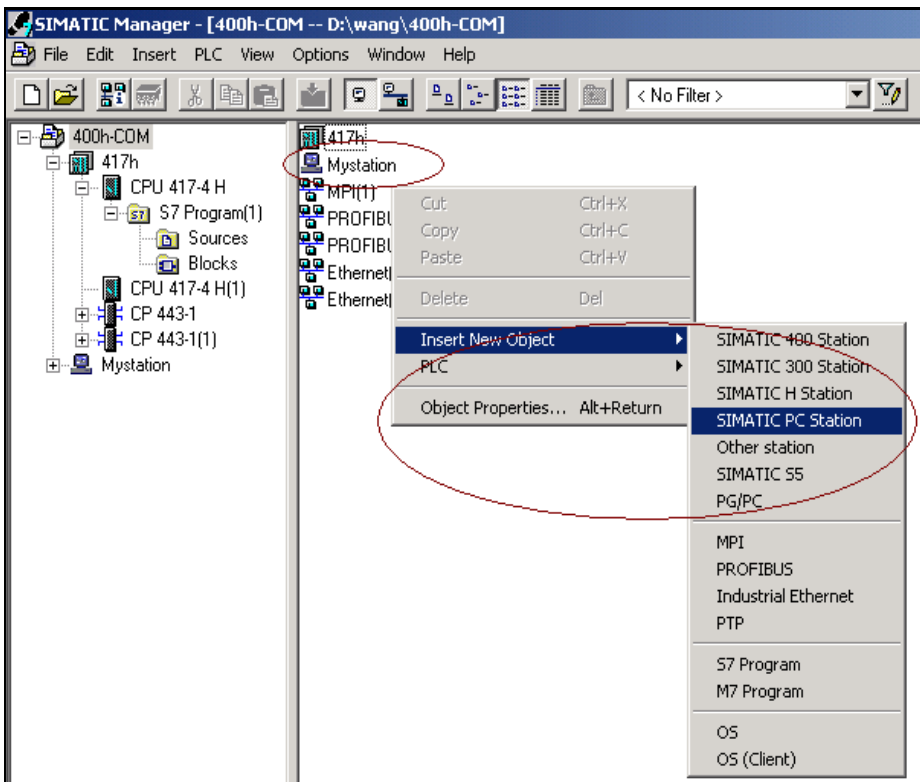


图3-22 建立PC站

在本例中，PC站的名称为：MyStation。然后，对PC站进行组态，如图3-23、3-24所示。

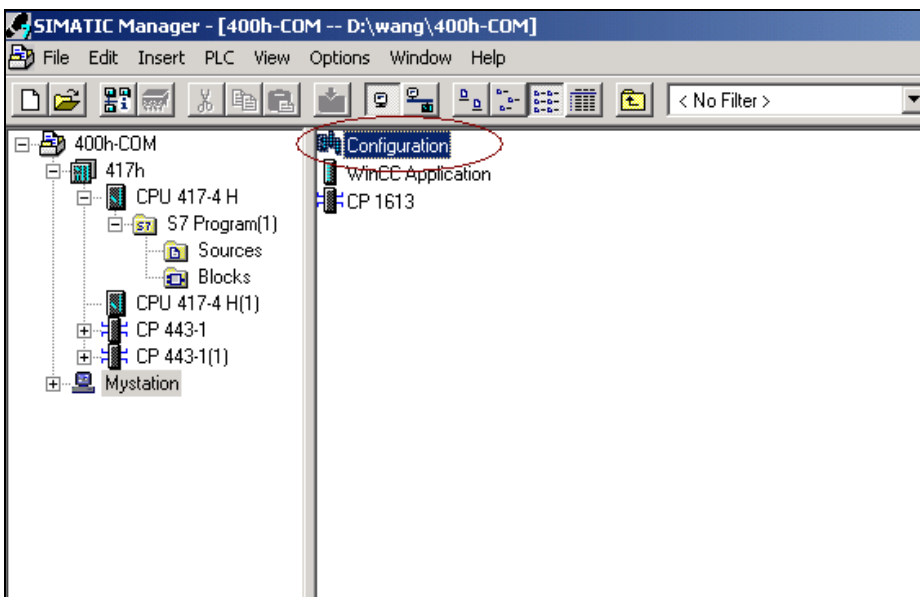


图3-23 组态PC站(1)

如同在417H硬件组态相同，在本例中分别插入Wincc Application、CP1613。

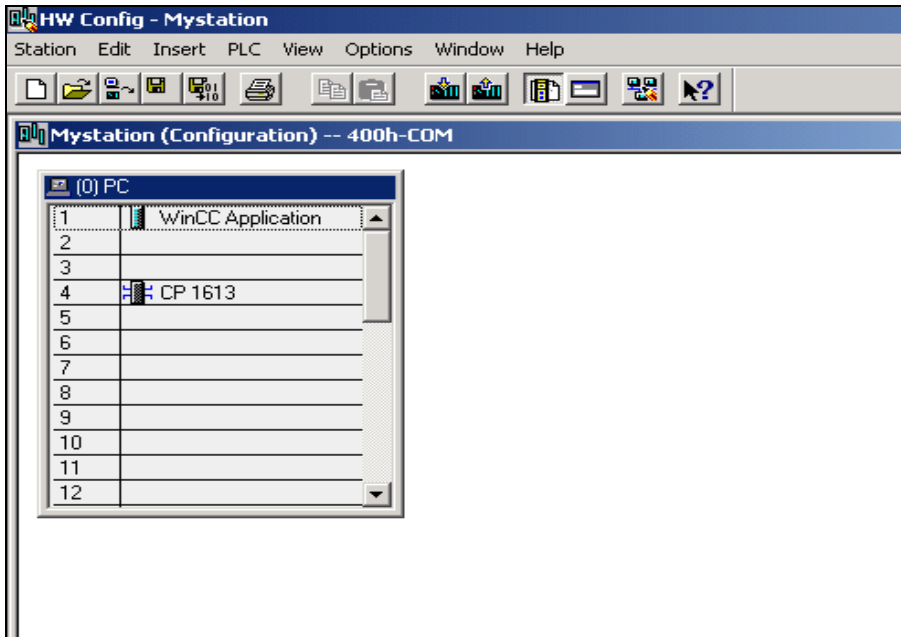


图3-25组

态PC站(2)

设置 CP1613 的网络配置。如图 3-26 3-27 所示。

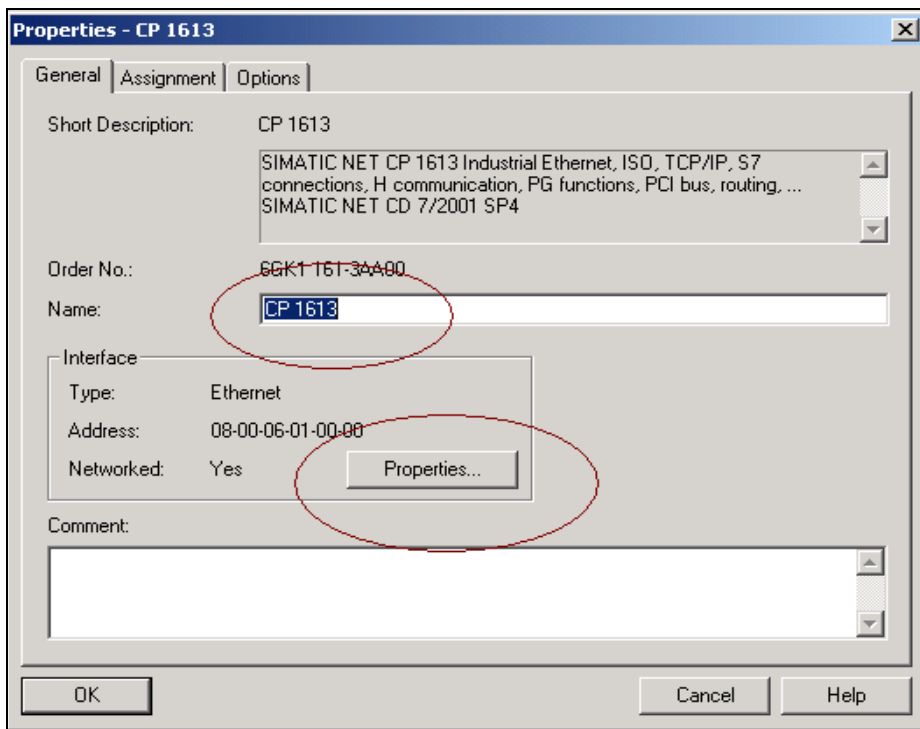


图3-26 设置CP1613的网络配置(1)

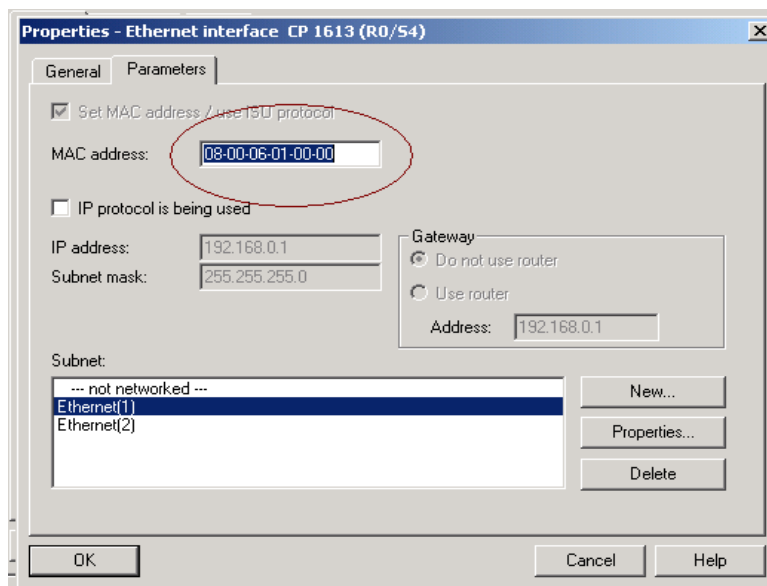


图3-27 设置CP1613的网络配置(2)

PC 站组态完毕后，进行编译检查是否有组态错误。

若没有组态错误，则在“Configuration console”中进行组态。

(3) 运行“Configuration Console”编辑器,组态CP1613，如图3-28所示。

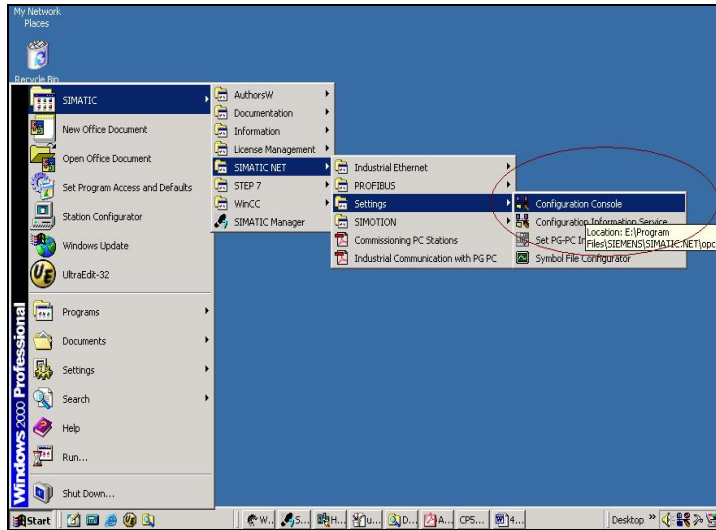


图3-28 组态CP1613

在控制台组态编辑器画面中，需要对CP1613进行组态。首先，需要把“Mode of Module ”的选项设置为：“Configured mode”。然后，把Index选项设置为与PC站相应的槽号，选择应用“Apply”按钮，如下图3-29。

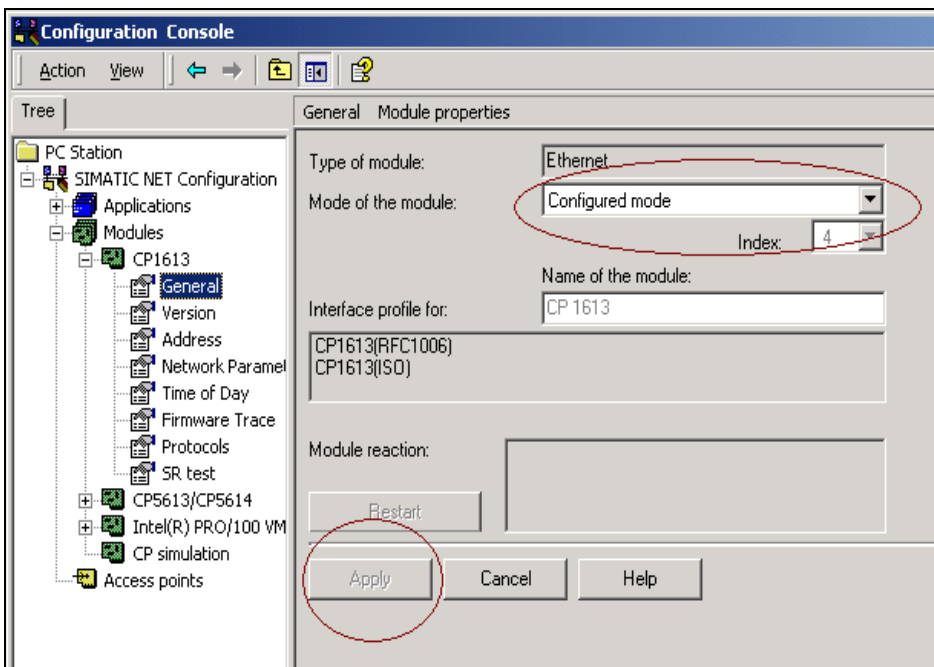


图3-29 组态CP1613的工作模式

- (4) 使用“Station Configuration Editor”进行编辑。双击工具条上的图标或者双击桌面上的“Station Configuration Editor”图标进入编辑画面，如图 3-30 所示。

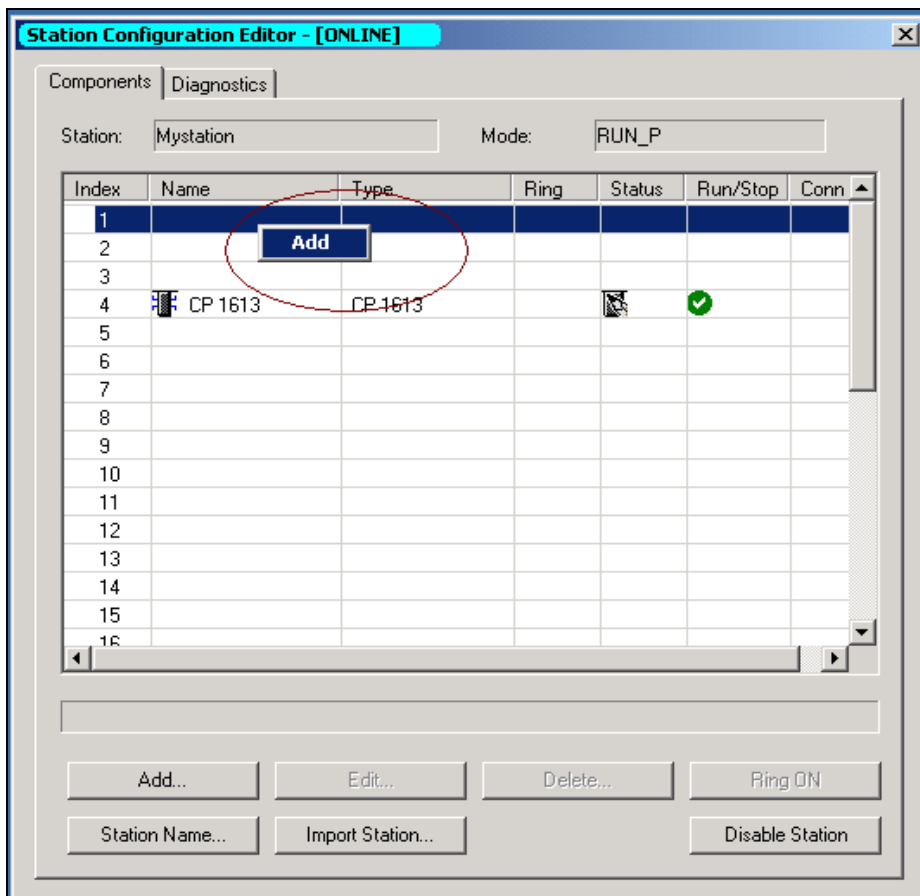


图 3-30 编辑 PC Station

进入Station Configuration Editor中即可见到在Configuration Console中配置的CP1613。对应于PC Station中的硬件组态，在Station Configuration Editor的相应槽位上添加WinCC Application。添加WinCC Application只需在相应槽位上单击右键进行选择。如图3-31所示。

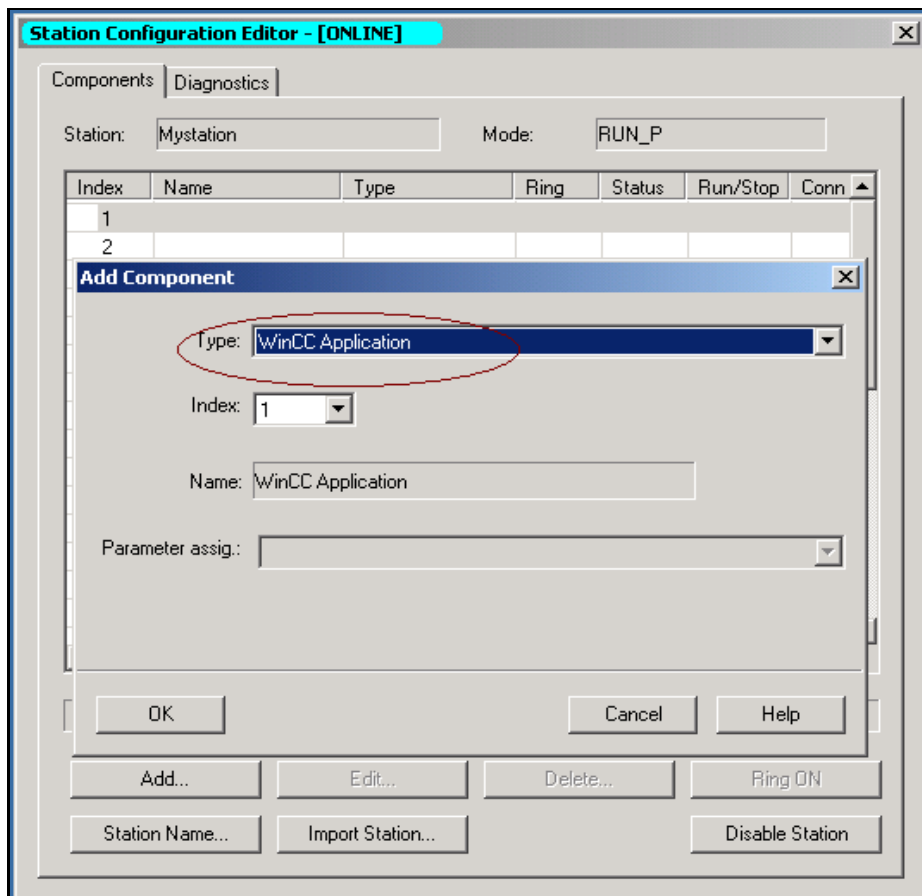


图 3-

31 在 PC Station 中建立“WinCC Application”

Station Configuration Editor完成后如图3-32.

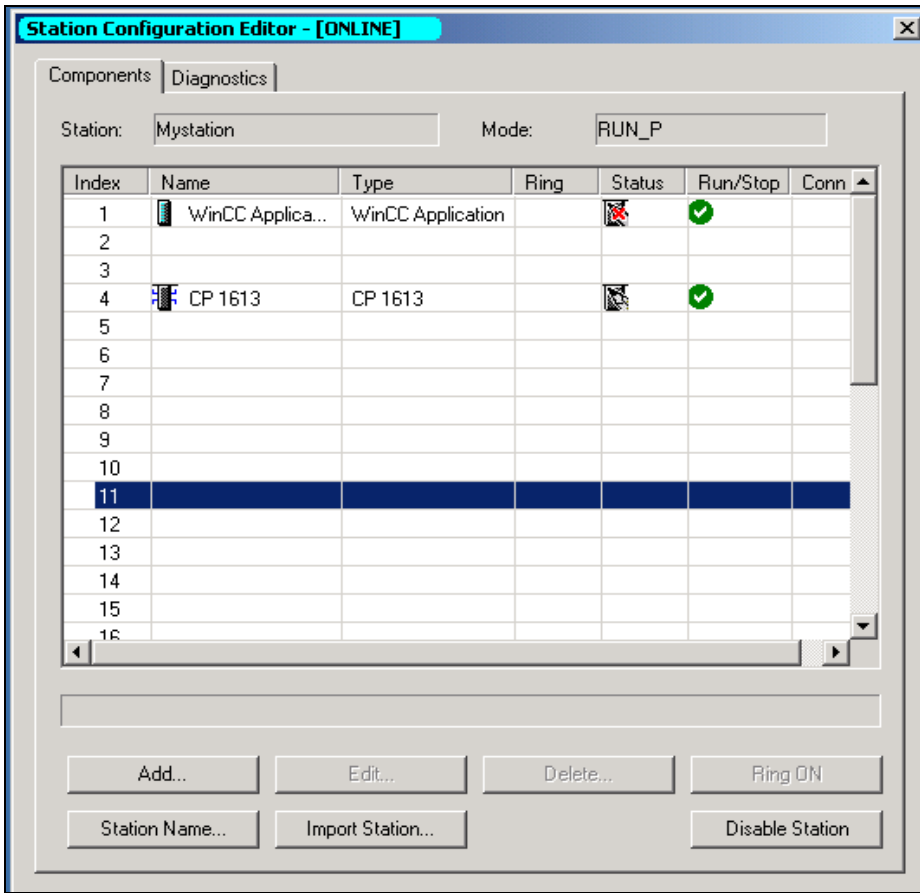


图3-32

配置完成PC Station

“Configuration Console”集成了访问点的设置选项。通过把S7-ONLINE的访问点设为“pc internal(local)”，系统会自动地选择通讯路径，如图3-33所示。

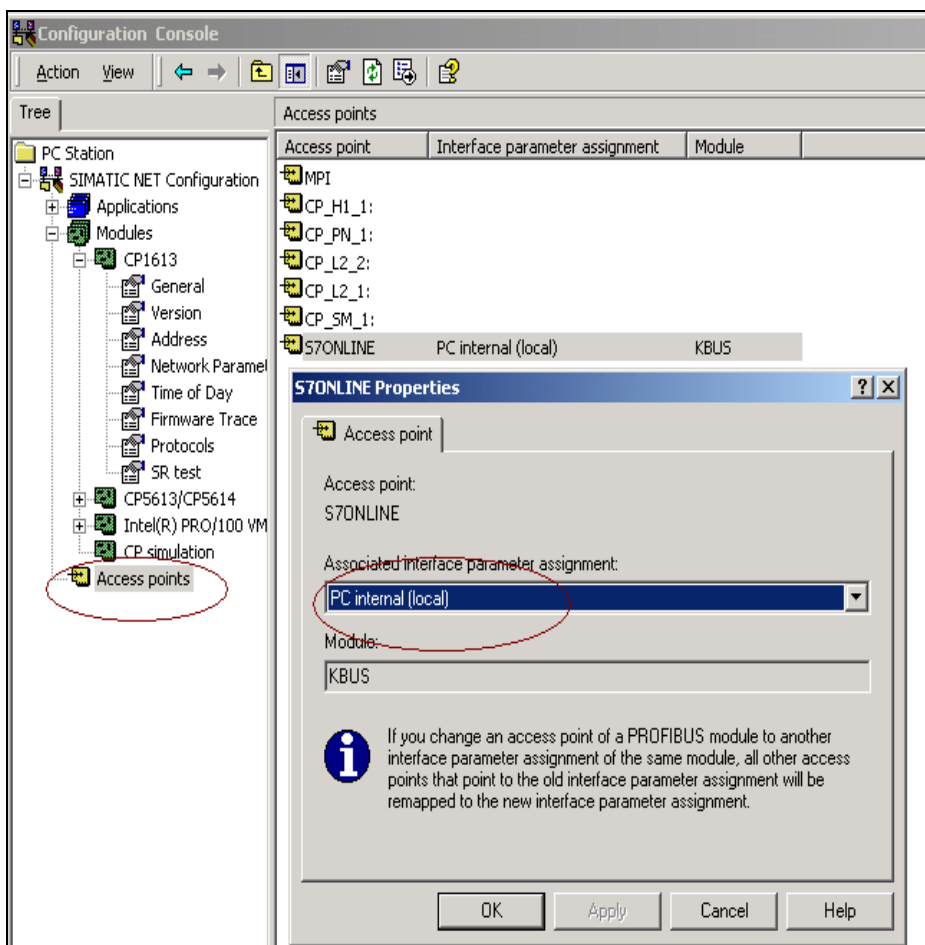


图3-33在“Configuration Console”设置访问点

设置完成后即可从“HW Config”下载 MyStation 硬件配置组态。

同时应注意“Station Configuration Editor”中的 station name 必须于 SIMATIC Manager 中 SIMATIC PC STATION 的名字（即 MyStation）一致；否则，不能正确下装。

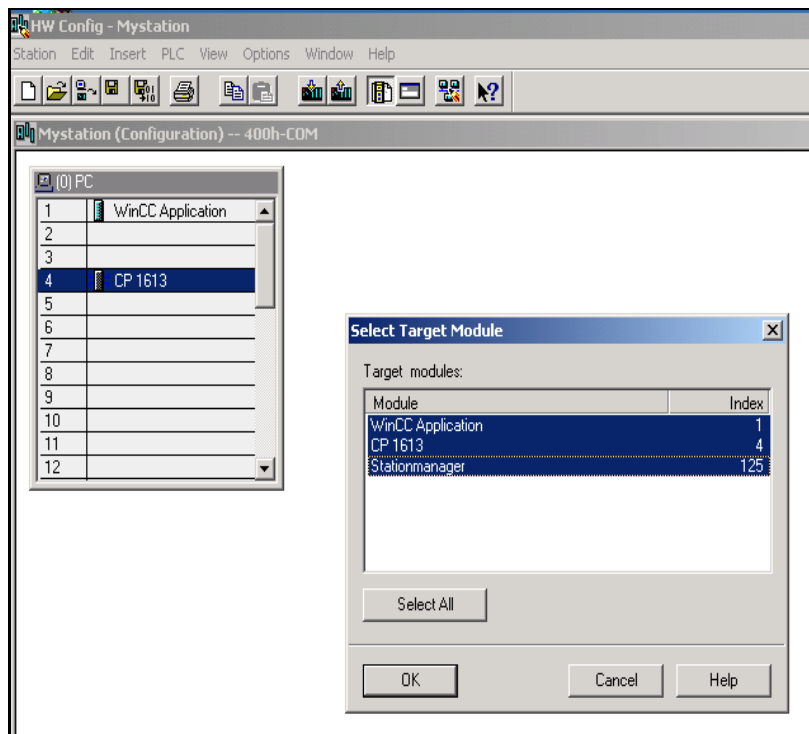


图3-34 下载PC站

PC 站中 WinCC Application 和 CP1613 的槽号与 Station Configuration Editor 一样。

下载完成后，Station Configuration Editor 中的状态如图3-35所示。

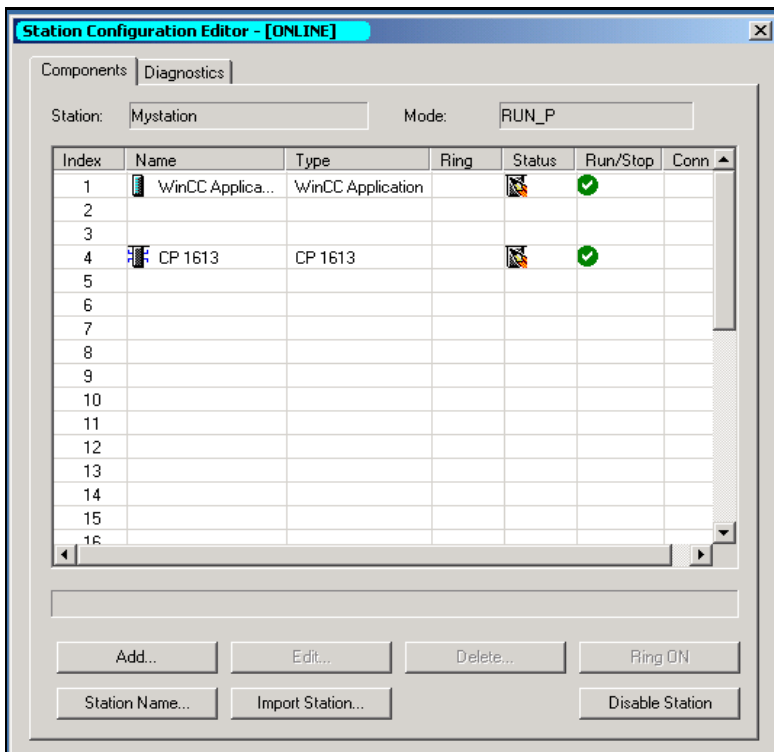


图3-35 下载完成后PC站状态

(5) 网络组态

在 417-4H 及 PC 站组态完毕之后，则可进入网络组态。通过选择菜单和图标进入网络组态的界面，如图 3-36 所示。

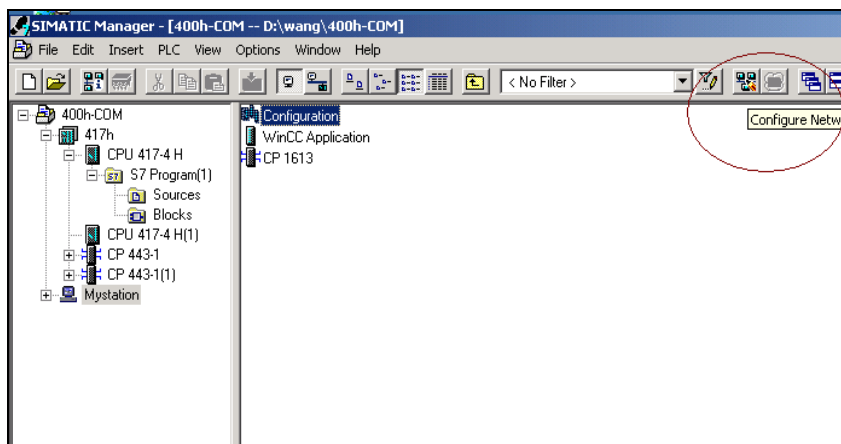


图3-36 网络组态

在进入网络组态后，就可以建立 417-4H 与 MyStation /WinCC Application 之间的连接。如图 3-

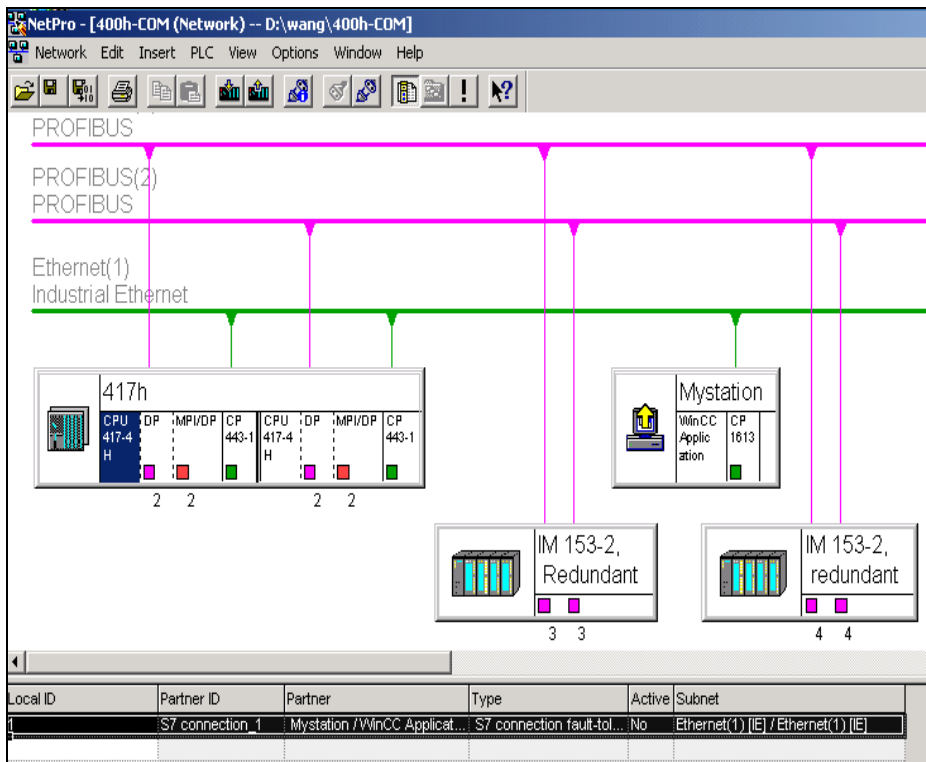


图3-37建立417-4H与MyStation /WinCC Application 之间的连接

(6) 向MyStation 和417h中下装网络组态信息。

在网络组态编辑器画面，选中MyStation按“Download”按钮，将MyStation的硬件配置及冗余连接下装到“Station Configuration Editor”中。选中417H按“Download”按钮，将417h站的连接下装到CPU中。

(7) 组态WinCC，建立冗余连接。点击Add New Driver,插入S7 Protocol Suite。

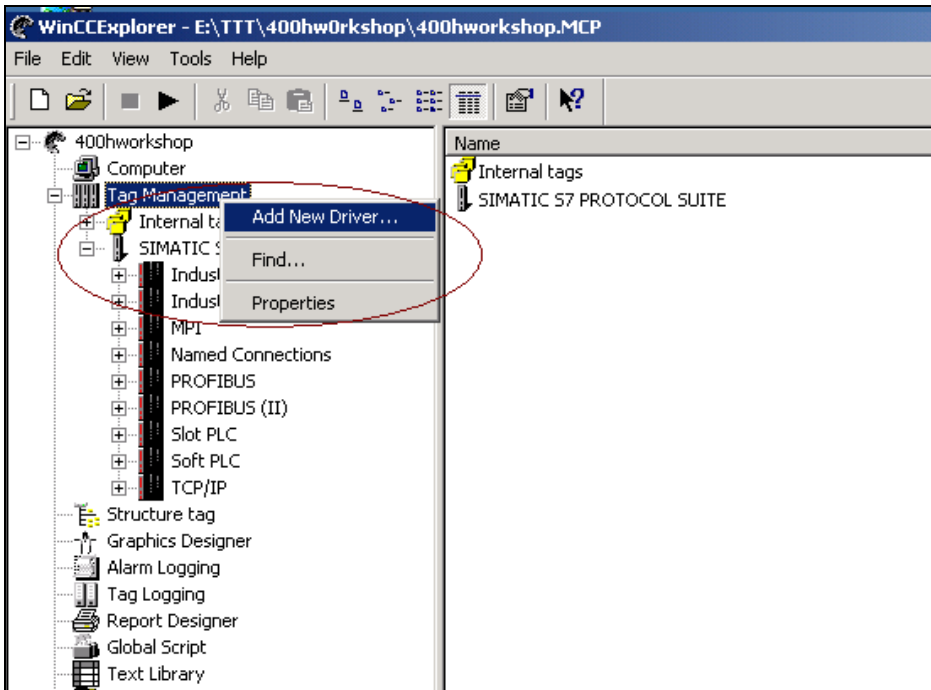


图3-38 组态WinCC，建立冗余连接

在“S7 Protocol Suite”下的“Named Connection” 选项中插入“New Driver Connection”，如图3—39 所示：

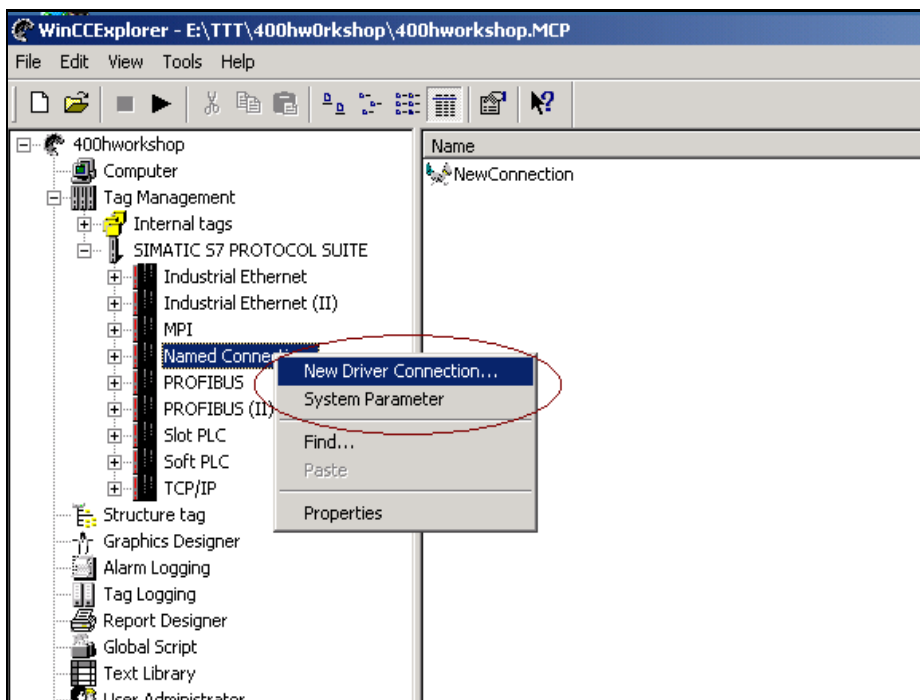


图3-39 建立“New Driver Connection”。

设置New Driver Connection 的属性 ,如图3-40所示。

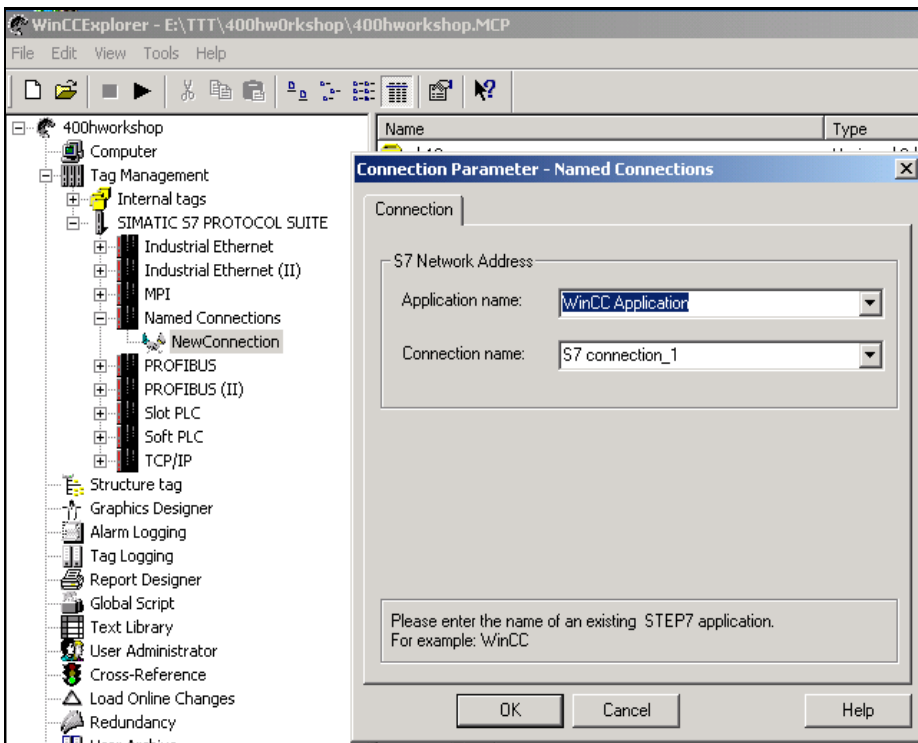


图3-40 设置New Driver Connection 的属性

其中“connection name”为“NetPro”中所建立的connection 名字，即“S7 connection_1”。此时，即可通过冗余连接进行通讯了。

(8)打开图形编辑器组态画面进行测试，如图3-41所示：

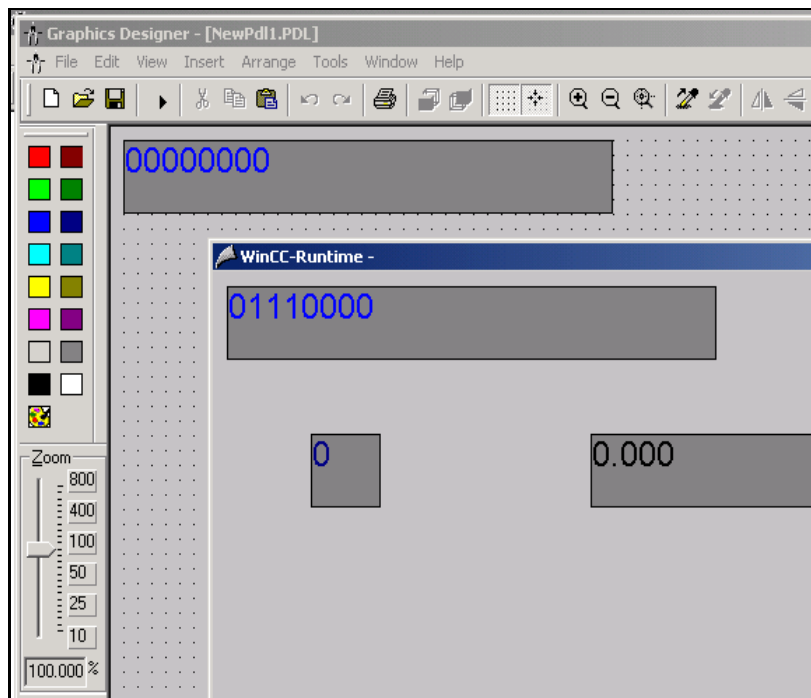


图3-41 WinCC 通讯测试

3.4 OP 与 S7-400H 的连接的解决方案

3.4.1 S7-400H 及 OP 的工作机制

S7-400H 由两个的控制器组成(CPU0 和 CPU1)，两者之间通过光纤连接来实现同步，在正常情况下，两个 CPU 都处在运行模式，其中一个作为主 CPU 在运行，另一个作为备用 CPU 在运行(冗余模式)，如果主 CPU 中发生了错误，系统会自动切换到备用 CPU 上，此时，备用 CPU 运行在单机模式，一旦产生错误的 CPU 修复完毕投入运行，修复的 CPU 作为备用 CPU，这种主备关系会一直保持的下次切换。

OP 和控制器通过 MPI、Profibus 或者工业以太网连接，一个 OP 可以连接到一条 (MPI 或 DP 或 IE) 总线，OP 通过这条总线可以和几个控制器相连，在此总线上的 OP 及控制器的地址必须是唯一的。如图 3-42 所示，这种接法是错误的，因为在下面的总线上，有 2 个地址是相同的。

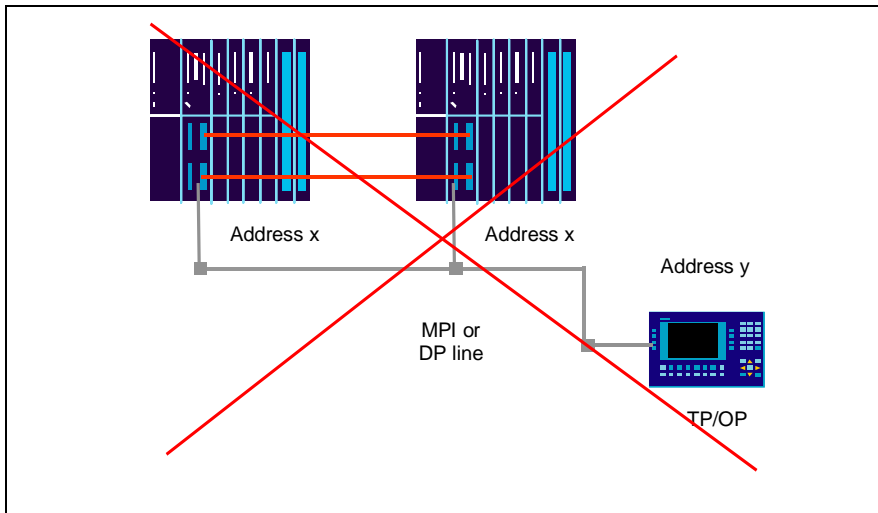


图 3-42 冗余 CPU 的地址不能相同

如果一个 OP 要连接几个 CPU，这些 CPU 一定要有不同的地址，如图 3-43 所示：

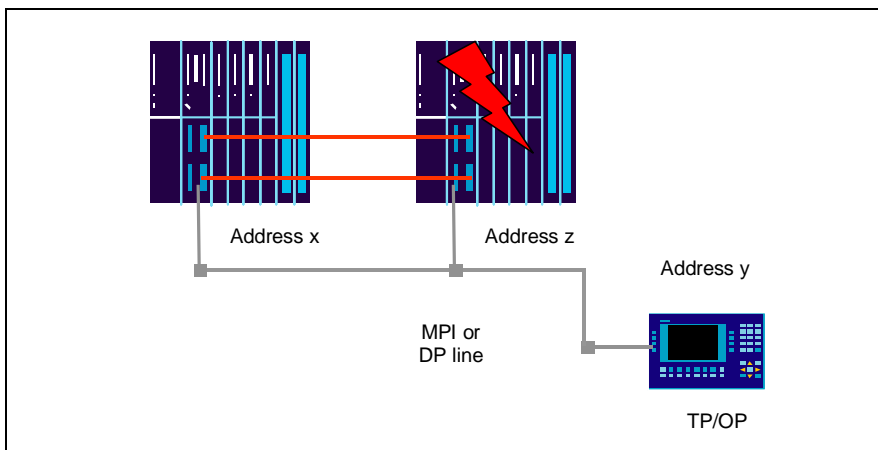


图 3-43 冗余 CPU 的地址不能相同

在这种配置中，会有以下问题，如果一个 CPU 故障，OP 会显示一个它试图去连接一个故障 CPU 的错误信息，OP 不能正常工作。在标准 CPU 配置中，如果 CPU 发生故障，整个设备都不能操作，这种情况是可以接受的。在 H-CPU 系统中，如果一个 CPU 发生故障，另一个还在运行，此时系统处在单机模式，是 S7-400H 系统的一种正常状态，所以 OP 此时必须处在操作的状态。下面介绍 OP 连接 H 系统的主要方法。

3.4.2 OP 连接到 H-System 一边

一个 OP 连接到其中的一边，如果这边的 CPU 发生故障，OP 就不能再进行操作显示，如图 3-44 所示：

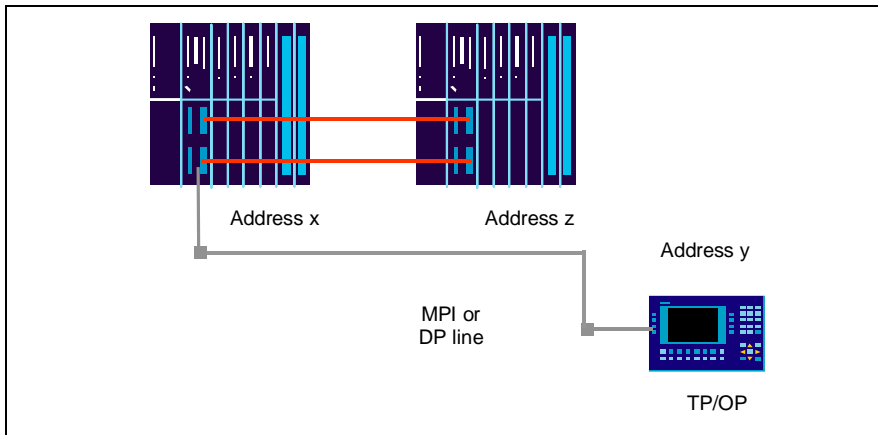


图 3—44 连接单边 CPU

3.4.3 连接冗余的 OP

最简单的方法是每个控制器各连一个 OP，这种解决方案价格比较贵，但也最能反映冗余的原理，因为 OP 硬件也是冗余的，这种方式两边的地址要设的完全一样。如图 3—45 所示：

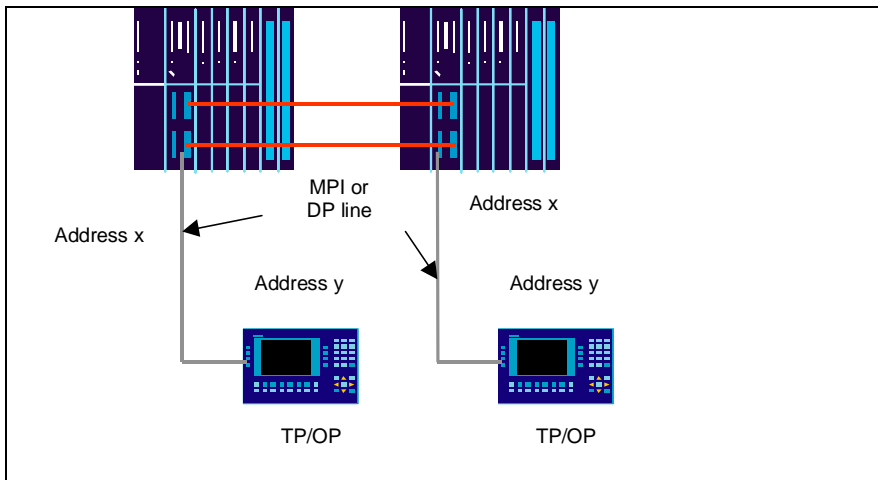


图 3—45 分别连接每个 CPU

3.4.4 通过控制中继器电源切换 OP 到 H-System 任一边

(1) OP 到 H-System 接线

一般情况下，OP 和 S7-H 的子系统都连在同一 MPI/Profibus 网段上，我们通过中继器分成几个独立的段，通过切换中继器的电源，可以把 OP 切到连 H-System 一边，这种方法，两个 CPU 的地址必须相同。如图 3—46 所示：

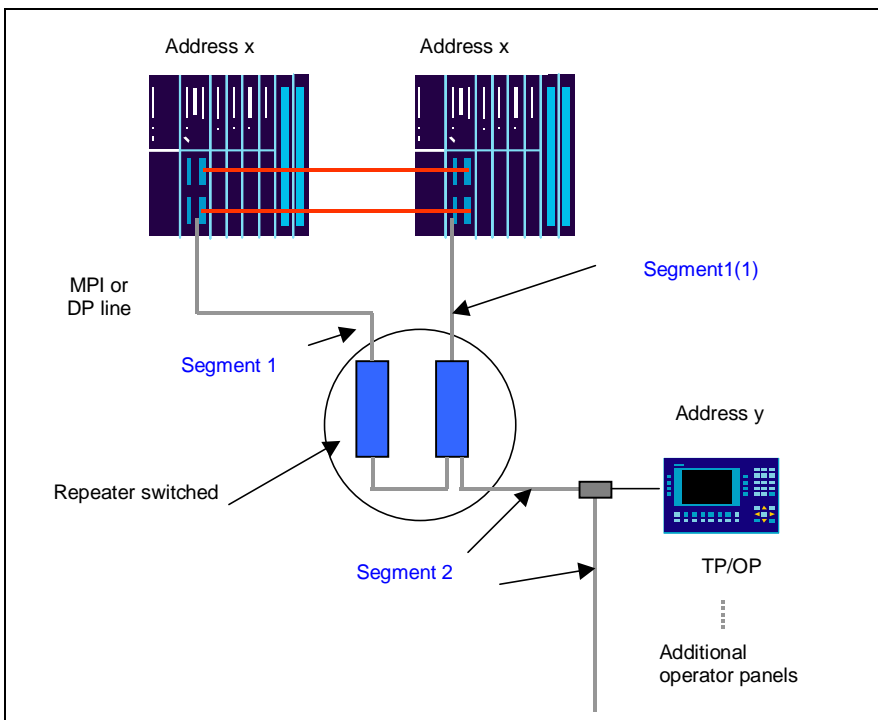


图 3—46 利用中继器 OP 连接 H 系统

(2) 中继器的接线

OP 连到哪个控制器是由中继器的电源切换来决定的，例如，在中继器的电源供应上，用一个两路转换器可以确保其中一个中继器接通电源，中继器的切换可以通过手动切换，也可以通过编程用数字量输出自动切换。中继器供电如图 3—47 所示：

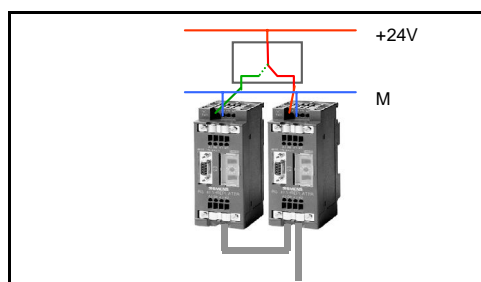


图 3—47 中继器电源的连接方法

(3) 信号通道

在运行过程中，OP 连接 CPU 的通道通过编程在 CPU 故障时自动切换，或者是人为的手动切换，OP 和 H-System 之间的信号通道如图 3—48 所示：（注：CPU 上标注箭头的是故障或故障通道，虚线表示当前信号通道）

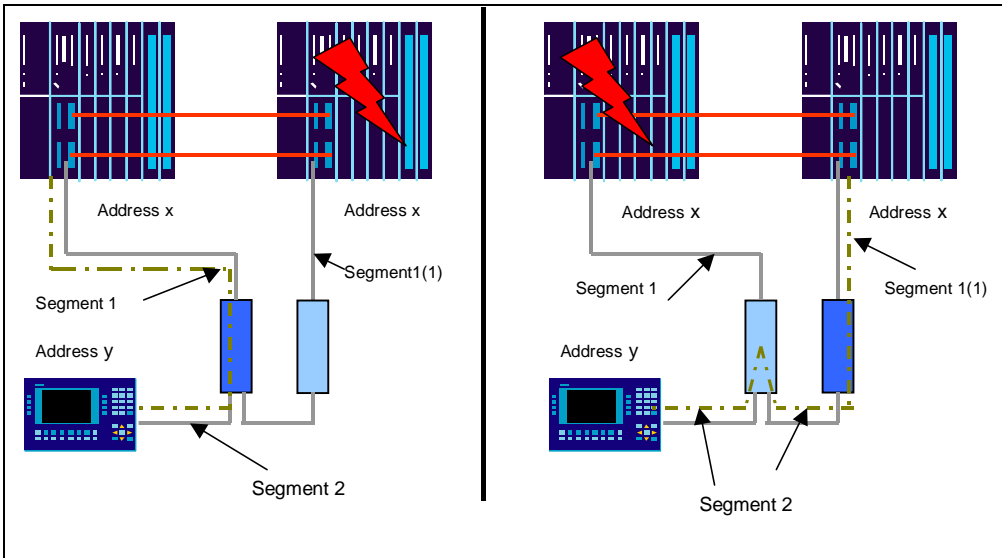


图 3—48 通过中继器的信号通道

3.4.5 在不使用 Repeater 的情况下直接连接 OP 到 H-System

使用该方法的前提条件：

1. 两 CPU 和 OP 处于同一个网络上（MPI 或工业以太网，DP 方式不支持）；
2. 使用 WinCC Flexible 对 OP 进行组态；

具体组态情况请参考如下网上链接内容：

<http://support.automation.siemens.com/CN/view/zh/23842654>

第四章 S7-400H 系统的 H-CIR 功能

概述:

对于 S7-400H 系统而言,除了 在系统运行过程中对出现故障的可热插拔部件进行更换,CPU414H 和 CPU417H 还允许在不中断系统程序运行的情况下对系统的硬件配置进行修改,这一功能特别适用于需要修改或增加系统硬件而不能停机的工艺控制要求。本文就这一功能的实现通过实例来加以说明。

4.1 使用 H-CIR 功能前的几点说明

4.1.1 可能进行的修改

- (1) 在中央机架上修改 CPU 的某些参数和增加 CP 模板
- (2) 更改 CPU 的存储部件,如: 更换 RAM 卡
- (3) 添加或删除远程的输入/输出站, 如: ET200M、DP/PA、Y_Link 等
- (4) 重新组态模板参数

4.1.2 不能进行的修改

- (1) 冗余模式 DP 主站的传输速率
- (2) S7-400H 的通信连接
- (3) 西门子硬件规范中要求必须切断系统电源才能安装的部件

4.1.3 注意事项

- (1) 原则上硬件组态的修改也会导致用户程序的修改
- (2) 确认 CPU 的存储器能否满足新的配置和用户程序的使用
- (3) 确认可能进行的修改对工艺过程的控制没有任何的不利影响
- (4) 确认修改前的 H 系统是在冗余模式下运行
- (5) 所进行的修改必须符合西门子相关的硬件和软件要求。
- (6) 对于因为需要增加硬件所作的配置修改,用户应充分考虑因硬件连接而产生的影响
- (7) 用户程序必须包含必要的中断组织块

4.2 H-CIR 功能原理

对于 H 系统而言,修改其硬件配置重新下载会导致冗余模式的停止,系统会转入单机运行模式,在单机模式下,可选择由其中一个 CPU 控制系统运行,对另一个 CPU 进行组态下载,利用 H 系统切换模式功能实现两个 CPU 运行模式的自动转换,使重新组态后的 CPU 接替控制任务继续控制系统运行,并传送修改参数到原 CPU,从而实现在不中断系统程序运行的情况下对系统的硬件配置进行修改。由于所使用 STEP7 版本的不同,本文以两中不同的实例分别加以描述。

4.3 H-CIR 功能举例(STEP V5.3 SP2 以下)

本例通过修改 CPU417-4H 的硬件组态为例介绍其调试过程，如在 ET200M 从站中增加 I/O 模块，修改系统配置。

4.3.1 系统结构

本例包含一个 CPU417-4H 主站和两个 ET200M 从站,其硬件配置如图 4-1 所示:

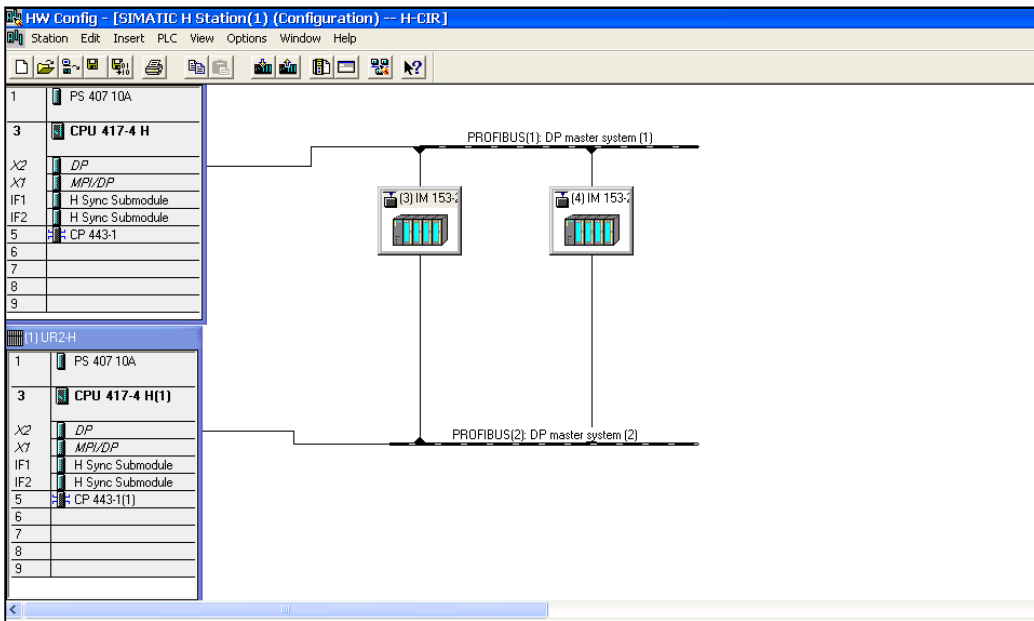


图 4-1 S7-400H 硬件配置图

4.3.2 例子所需硬件和软件

硬件:

- (1) S7-400 CPU417-4H
- (2) 通信处理器 CP443-1
- (3) DP 从站 ET200M
- (4) Profibus 电缆及编程设备

软件:

STEP7 V5.3 SP1

注:若使用 STEP7 V5.3 SP2 以上版本,其功能的实现亦可请参阅 H-CIR 功能举例(二)

4.3.3 实例步骤

(1).建立一个名为 H-CIR 的项目。创建 CPU417-4H 的站,如图 4-2 所示:

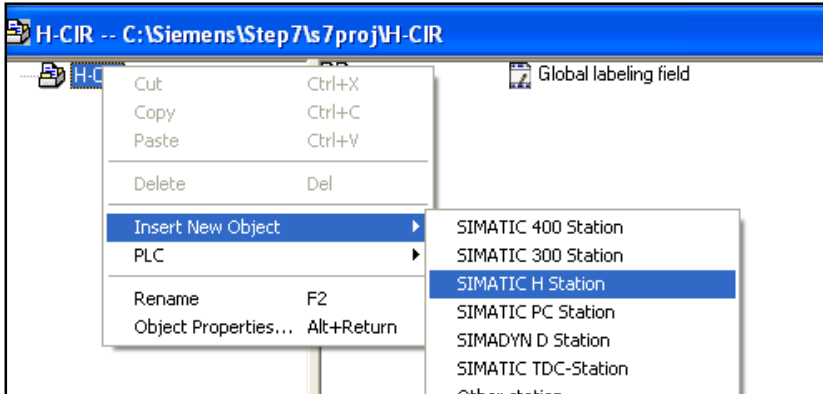


图 4-2 建立 H 站

(2).在硬件组态中,按实际硬件组态 CPU417-4H,ET200M 及相关网络,编译存盘,下载硬件组态,图 4-3 所示:

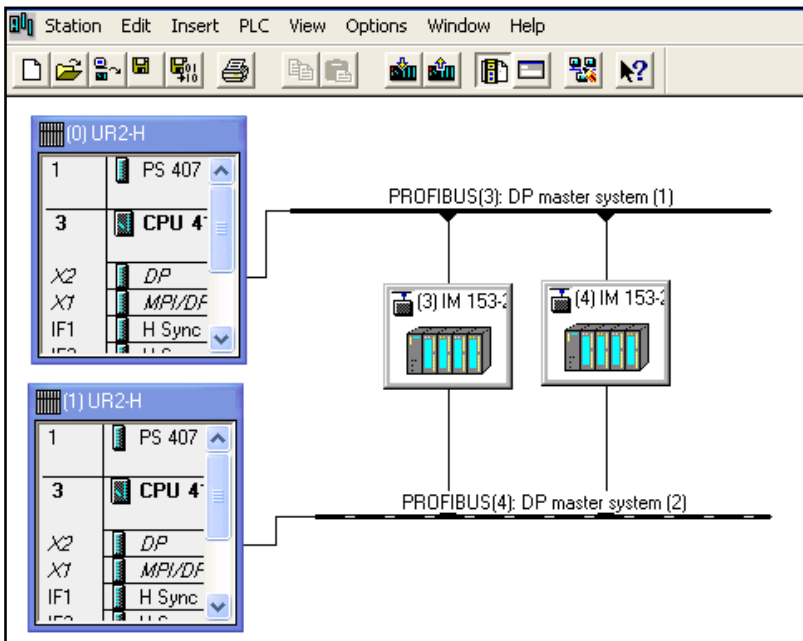


图 4-3 配置 H 站硬件

注意:

CPU 属性中,只有带有蓝色标志参数可以在线修改,如图 4-4 所示框中参数可以在线修改。

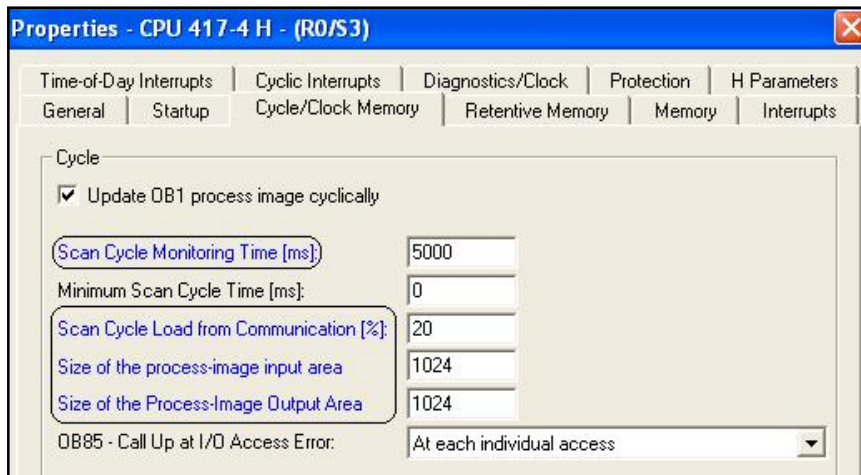


图 4—4 H-CPU 可以在线修改的参数

- (3) 在 Blocks 下添加相关的 OB 块，OB70、72、73、80、81、82、83、84、85、86、87、121、122，在更换硬件和修改参数时不会导致停机。
- (4) 编辑用户程序并下载，使 CPU417-4H 运行在冗余模式下。
- (5) 更改硬件，例如在 ET200M 中增加 I/O 模块，编译存盘。
- (6) 在“Simatic Manager”下选上“CPU417-4H”，点击 PLC 菜单栏下的“Diagnostic/Setting>Operating Mode...”，如图 4—5 所示：

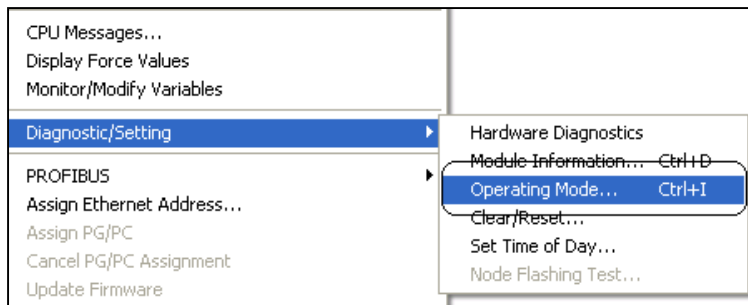


图 4—5 选择操作模式

- (7) 选择处于“Standby”模式下的 CPU417-4H(1)，点击“Stop”命令，如图 4—6 所示：

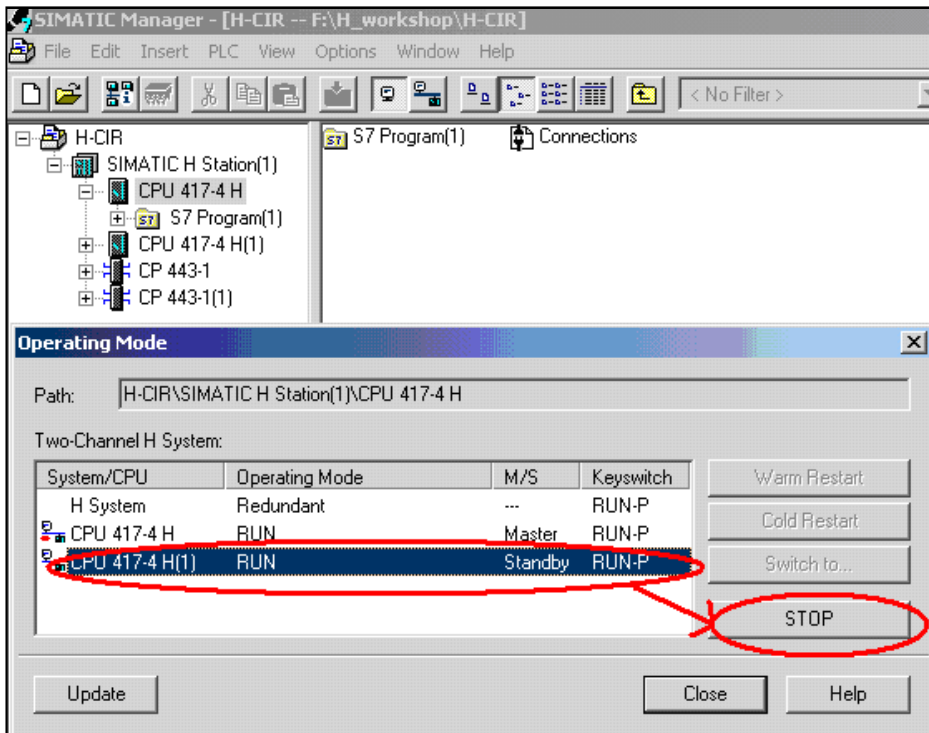


图 4-6 停止备份底 CPU

(8) 系统处于单机运行模式（Solo Mode），如图 4-7 所示：

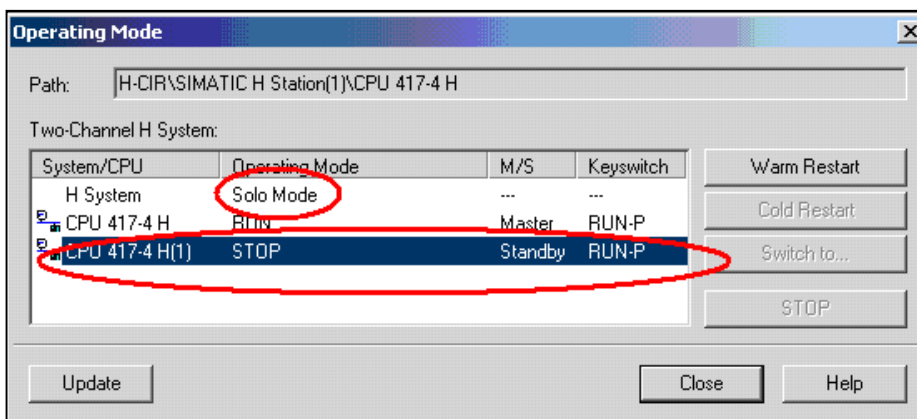


图 4-7 H 系统的 SOLO 模式

(9) 在硬件组态中选择已处于停止状态的 CPU417-4H(1)下载，如图 4-8 所示：

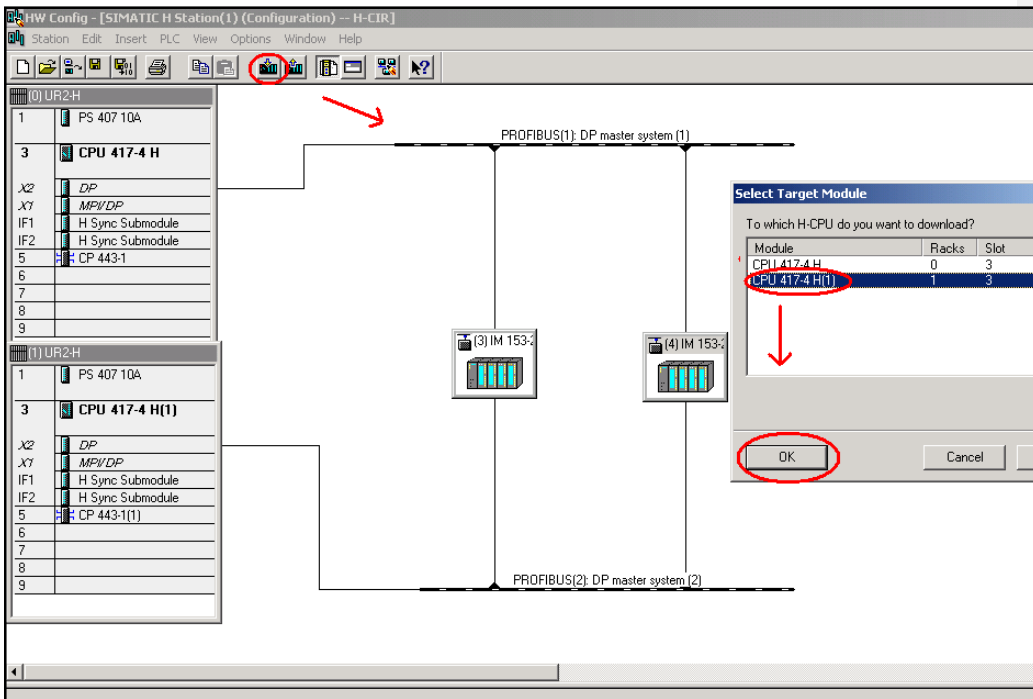


图 4-8 下载组态信息到处于停止模式的 CPU 中

(10) .下载完成后，在“Simatic Manager”下选上 CPU417-4H，点击 PLC 菜单栏下的“Diagnostic/Setting->Operating Mode...,”进入操作模式选择界面。

(11) .选择“H System”,点击“Switch to ...”命令，如图 4-9 所示：

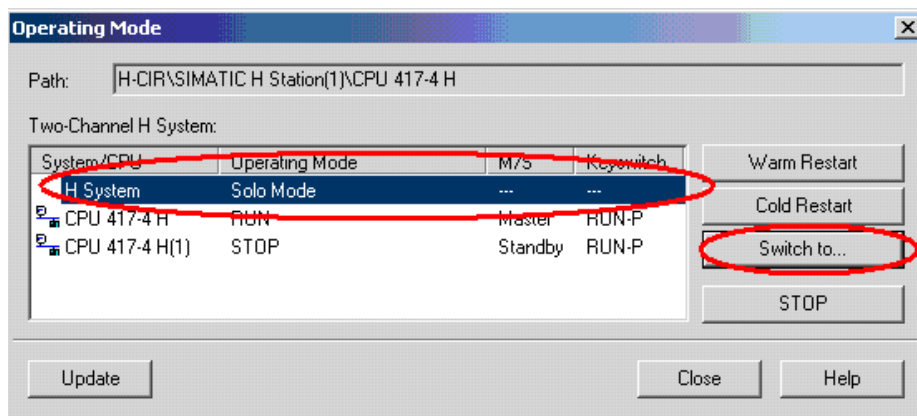


图 4-9 选择 Switch to 操作

(12) .选择“altered configuration”，点击“Switch”命令，弹出切换界面如图 4-10 所示：

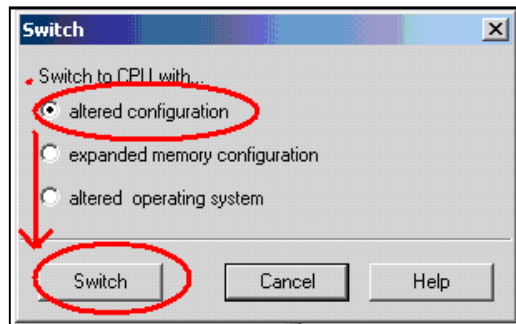


图 4—10 Switch t 界面

(13) .点击“Switch”键后弹出提示框，在确认不会因为错误造成危险后，点击“OK”键进行 CPU 切换操作。

(14) .系统执行 Switch 功能，master 和 Standby 的 CPU 相互切换，原先的 master CPU 切换为 Standby CPU 并处于停止模式，原先的 Standby CPU 切换为 master CPU 并带有从新配置的参数开始运行，系统仍处于单机运行模式。

(15) .选择已处于“STOP”模式下的 CPU 417-4H，并对其进行热启动。

(16) .启动完成，系统再一次处于冗余的运行模式。

(17) .编辑因硬件更改而需修改的用户程序，从而完成 H-CIR 功能。

4.4 H-CIR 功能举例(STEP V5.3 SP2 及以上版本)

使用 STEP7 V5.3 SP2 及以上版本，参数及模块的在线修改操作更加方便，通过修改 CPU417-4H 的硬件组态为例介绍其调试过程，例如在 ET200M 从站中增加 I/O 模块，修改系统配置。

4.4.1 操作步骤

(1) .与上例的配置相同，硬件组态完成后，编辑用户程序并下载，使 CPU417-4H 运行在冗余模式下。

(2) .更改硬件，在 ET200M 中增加 I/O 模块，编译存盘。

注意：

CPU 属性参数中只有带有蓝色标记的参数可以在线修改

(3) .在硬件组态中,点击下载命令,并选择“Download station configuration in Run mode”模式，下载修改后的硬件组态,如图 4—11 所示:

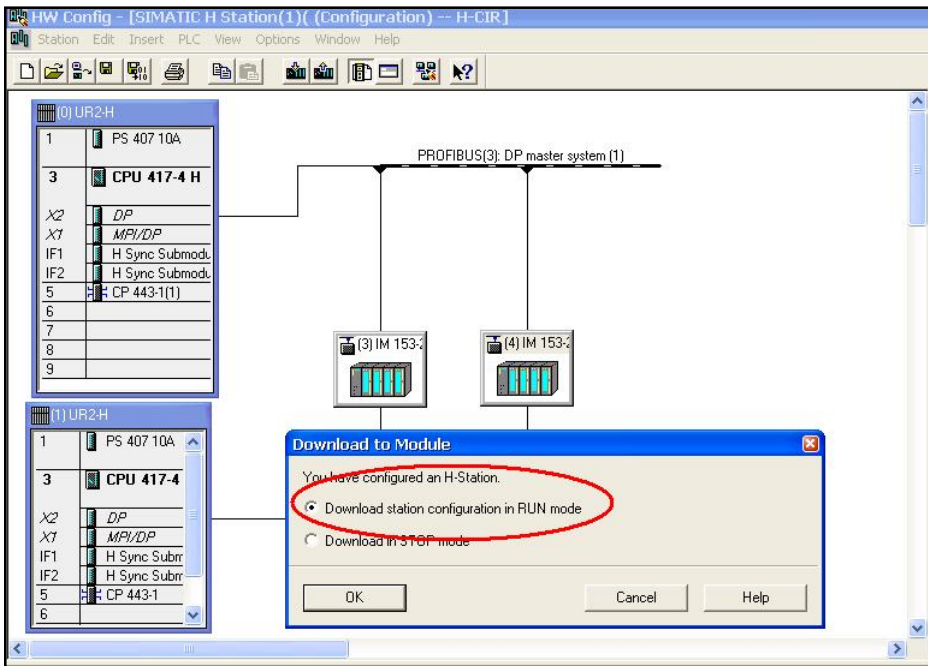


图 4-11 下载组态信息

(4) 选择处于“Reserve”模式的 S7-400H CPU,选择“Automatically continue”选项,点击“Next”键如图 4-12 所示:

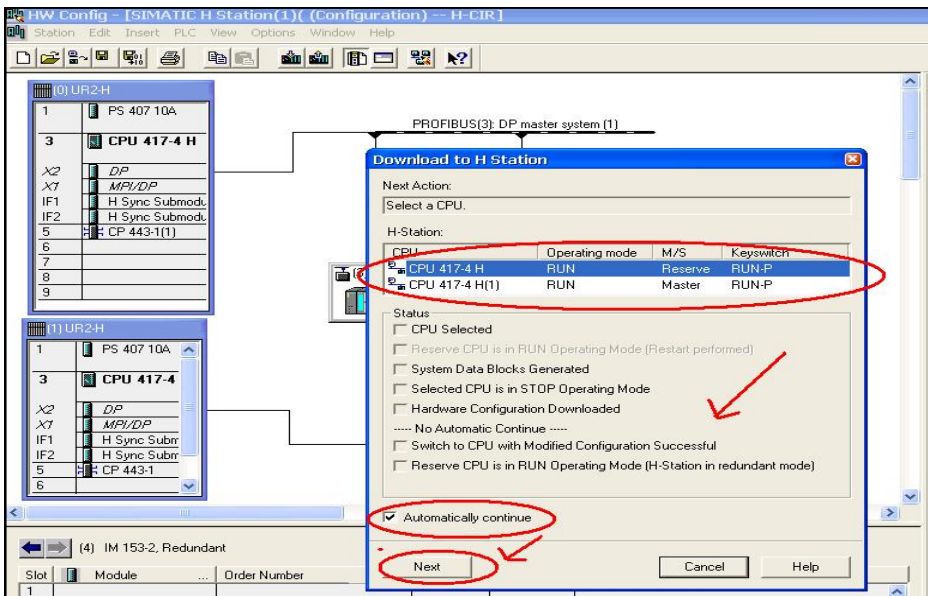


图 4-12 切换操作 (1)

(5) .选择的 S7-400H CPU 处于停机状态并完成硬件下载工作，如图 4-13 所示:

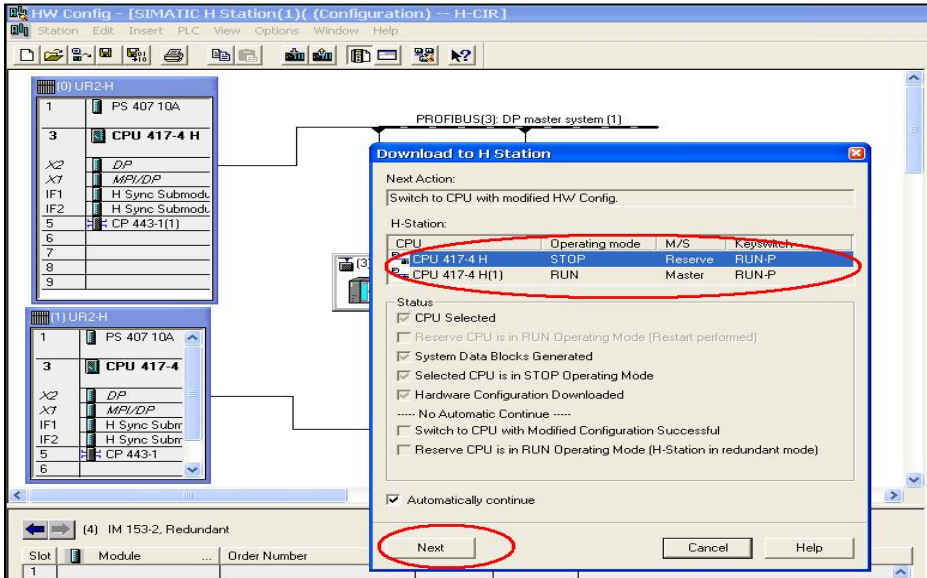


图 4-13 切换操作 (2)

(6) .继续点击“Next”键,系统将自动完成切换、热启动等功能，如图 4-14 所示:

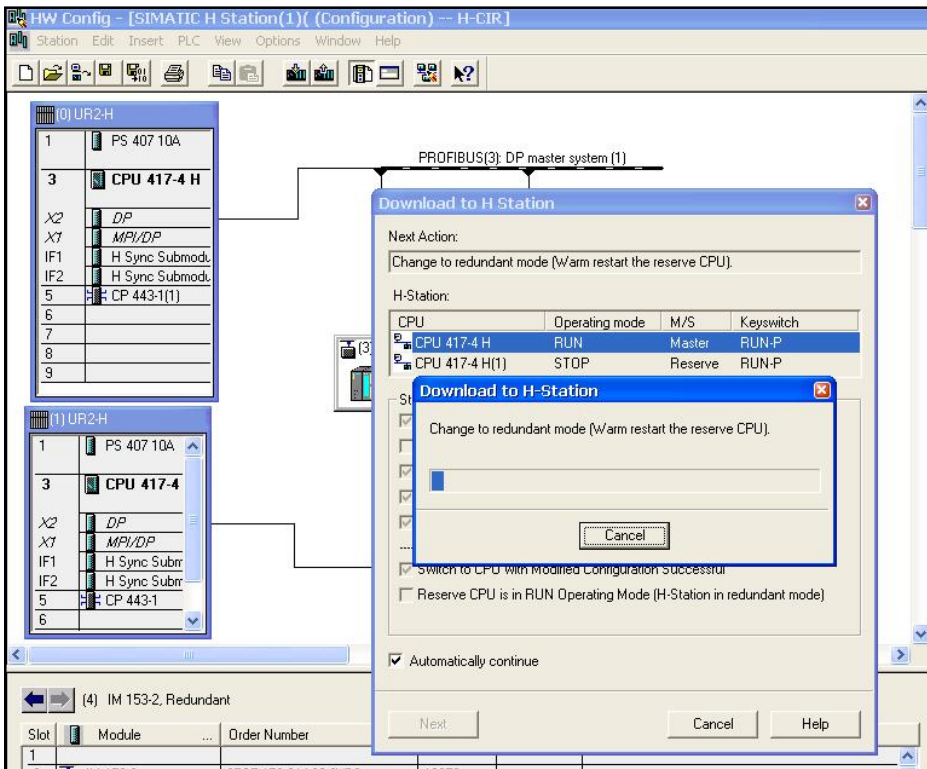


图 4-14 切换操作 (3)

- (7) .启动完成，系统再一次处于冗余的运行模式。
- (8) .编辑因硬件更改而需修改的用户程序，从而完成 H-CIR 功能。

总结:

1. 对于 H 系统的 CIR 功能而言，它不同与标准型 S7-400 的 CIR 功能，其工作原理和实现方法均有所不同。
2. 本文仅以 S7-400H 系统的 PLC 的 CIR 功能而作描述，它并不完全适用于 S7-400H 的 PCS7 系统和 S7-400FH 系统。
3. 冗余系统的在线修改必须考虑到因参数修改错误而造成控制系统停机的后果，本文所述内容仅作参考。并不承当作为工程理论使用而引起的一切后果及责任。

第五章 S7-400H 系统信息及诊断

在生产控制中，通常需要对 S7-400H 系统的信息和状态进行监控，例如监控 CPU 的主从状态、操作状态、DP 从站与主站的通信状态等信息。在程序中可以通过对状态的判断进行必要的操作，也可以将系统信息上传到 HMI，便于操作及维护人员监控。通过功能块可以将系统信息读出，下面介绍读取不同系统信息调用功能块的方法。

5.1 利用 SFC51(SSL-ID W#16#xy71)读出 H 系统信息

SSL-ID (SYSTEM STATUS LIST)系统状态目录，利用 SFC51 可以读出在 SSL-ID 指定的 PLC 系统信息，当 SSL-ID 等于 W#16#0071 时，表示需要读出 S7-400H 系统当前的状态，可以在 OB1 或循环中断组织块 OB3X 中调用，调用 SFC51 的例子如图 5-1 所示：

```
CALL "RDSYSST"          SFC51
REQ      :=M1.1
SZL_ID   :=W#16#71
INDEX    :=W#16#0
RET_VAL  :=MW2
BUSY     :=M1.2
SZL_HEADER:=DB1.SZL_HEADER  P#DB1.DBX0.0
DR       :=DB1.DR           P#DB1.DBX4.0
```

图 5-1 调用 SFC51(SSL-ID W#16#0071)例子程序

SFC51 的参数解释如下，

REQ : 为 1 是读取 SZL_ID 指定的系统信息，本例中 M1.1 为 1 时启动读请求。

SZL_ID : 指定需要读取的系统信息，本例为 W#16#71，H 系统当前状态。

INDEX : 本例中没有意义。

RET_VAL : 调用 SFC51 的状态字。

BUSY : 为 1 时表示读进程没有完成。

SZL_HEADER: 输出系统信息存储的数据记录区号及长度，结构数据类型。

DR : 指定输出系统信息存储在 CPU 的地址区。

参数 SZL_HEADER 与 DR 的地址区在 DB1 中建立，如 5-2 所示：

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	SZL_HEADER	STRUCT	
+0.0	LENTHDR	WORD	W#16#0
+2.0	N_DR	WORD	W#16#0
=4.0		END_STRUCT	
+4.0	DR	ARRAY[1..16]	
*1.0		BYTE	
=20.0		END_STRUCT	

图 5-2 参数 SZL_HEADER 与 DR 的地址区

参数 SZL_HEADER 为一个结构数据，包括两个字，第一个字输出系统信息长度，例如 W#16#10 表示输出 16 个字节，第二个字输出存储系统信息的数据记录区，例如 W#16#1 表示数据记录区为 1。参数 DR 为存储系统信息的地址区，数据类型为指针，长度必须大于参数 SZL_HEADER 第一个字输出的信息长度。本例中当 M1.1 为时，读取的系统信息存储在 DB1.DBB4~DB1.DBB20 16 个字节中。用户可以对 16 个字节长度的系统信息进行分析和处理，系统信息内容如下：

内容	长度	含义
Redinf	2 bytes	冗余信息 W#16#0011: 单机 H CPU 运行 W#16#0012: H 系统 2 备 1 运行
Mwstat1	1 byte	状态字节 1 Bit 0: 保留 Bit 1: 保留 Bit 2: 保留 Bit 3: 保留 Bit 4: 机架 0 中 CPU 的状态 =0: 从 CPU =1: 主 CPU Bit 5: 机架 1 中 CPU 的状态 =0: 从 CPU

=1: 主 CPU

Bit 6: 保留

Bit 7: 保留

Mwstat2 1 byte

状态字节2

Bit 0: 同步连接状态 01:

CPU 0 和CPU 1同步

=0: 不可能

=1: 可能

Bit 1: 0

Bit 2: 0

Bit 3: 保留

Bit 4: =0: CPU没有在机架0

=1: CPU在机架0上

(冗余模式: bit 4 = 0)

Bit 5: =0: CPU没有在机架1

=1: CPU在机架1上

(冗余模式: bit 5 = 0)

Bit 6: 保留

Bit 7: 主从切换是否从新使能

=0: 否

=1: 是

Hsfinfo 2 bytes

SFC 90 "H_CTRL"状态字

Bit 0: =0: 从新使能没有激活

=1: 从新使能激活

Bit 1: =0: 从站Updating使能

=1: 从站Updating没有使能

Bit 2: =0: Link-up 模式没有使能

=1: Link-up 模式使能

Bit 3: 保留

Bit 4: 保留

Bit 5: 保留

Bit 6: 保留

Bit 7: 保留

Bit 8: 保留

Samfehl 2 bytes

保留

Bz_cpu_0 2 bytes

CPU在机架0的模式

W#16#0001: 停止 (update)

W#16#0002: 停止 (reset memory)

W#16#0003: 停止(self-initialization)

W#16#0004: 停止(internal)

W#16#0005: 启动(cold restart)

W#16#0006: 启动(warm restart)

W#16#0007: 启动(hot restart)

W#16#0008: 运行(solo mode)

W#16#0009: 运行(redundant mode)

W#16#000A: HOLD模式

W#16#000B: LINK-UP模式

W#16#000C: UPDATE模式

W#16#000D: 故障

W#16#000E: 自检测

W#16#000F: 没有开机

Bz_cpu_1 2 bytes

CPU在机架1的模式

(与 bz_cpu_0相同)

Bz_cpu_2 2 bytes	保留
Cpu_valid 1 byte	信息变量 bz_cpu_0 和 bz_cpu_1 有效性 B#16#01: bz_cpu_0 有效 B#16#02: bz_cpu_1 有效 B#16#03: bz_cpu_0 和 bz_cpu_1 有效
hsync_f 1 byte	连接质量的状态 (只有mwstat2 bit 0为1时有效) <ul style="list-style-type: none">● Bit 0: 上部插孔的同步模块光纤连接质量被限制● Bit 1: 下部插孔的同步模块光纤连接质量被限制Bit 2 到 7: 0

5.2 利用 SFC51(SSL-ID W#16#xy75)读出 H 系统可切换 DP 从站的信息

当 SSL-ID 等于 W#16#0C75 时,表示需要读出 S7-400H 系统可切换 DP 从站 (ET200M) 的状态, SFC51 可以在 OB1 或循环中断组织块 OB3X 中调用, 调用 SFC51 的例子如图 5—3 所示:

```
CALL "RDSYSST"          SFC51
REQ      :=M1.1
SZL_ID   :=W#16#C75
INDEX    :=W#16#1FF5
RET_VAL  :=MW2
BUSY     :=M1.2
SZL_HEADER:=DB1.SZL_HEADER P#DB1.DBX0.0
DR       :=DB1.DR          P#DB1.DBX4.0
```

图 5—3 调用 SFC51(SSL-ID W#16#0C75)例子程序

与读取 SSL-ID W#16#xy71 的方法相同, 参数 SZL_ID 变成 W#16#C75, 参数 INDEX 为从站的诊断地址, 例如从站诊断地址为 8181, 转换十六进制为 W#16#1FF5。参数 SZL_HEADER 和 DR 结构与上例相同, 读出的信息长度同样是 16 个字节, 本例中当 M1.1 为时, 读取的系统信息存储在 DB1.DBB4~DB1.DBB20 16 个字节中。用户可以对 16 个字节长度的 DP 从站信息进行分析 and 处理, 信息内容如下:

内容	长度	含义
adr1_bgt0	1 word	DP 从站接口模块第一个地址区, 它的主站接口模块插在机架 0 上: DP 主站的 PROFIBUS 系统 ID 号和从站站号。

adr2_bgt0	1 word	DP从站接口模块第二个地址区，它的主站接口模块插在机架0上： DP主站的槽号和子槽号（集成在CPU的主站模块的接口号）
adr1_bgt1	1 word	DP从站接口模块第一个地址区，它的主站接口模块插在机架1上： DP主站的PROFIBUS系统ID号和从站站号。
adr2_bgt1	1 word	DP从站接口模块第二个地址区，它的主站接口模块插在机架1上： DP主站的槽号和子槽号（集成在CPU的主站模块的接口号）
Res	2 words	保留
Logadr	1 word	DP从站的诊断地址： <ul style="list-style-type: none"> ● Bits 0 到 14: 诊断地址 ● Bit 15: I/O标识符 (0 = 输入, 1 = 输出)
Slavestatus	1 word	通信状态： <ul style="list-style-type: none"> ● Bit 0 = 1: 机架0的DP主站模块不能访问从站接口模块 ● Bit 1 = 1: 机架1的DP主站模块不能访问从站接口模块 ● Bits 2 到 7: 保留(每一个位为0)

5.3 利用 SFC87 读出冗余连接状态信息

S7-400H 之间可以建立冗余的连接，在单一网络上，两个站建立的连接可能为 4 个，如图 5-4 所示：

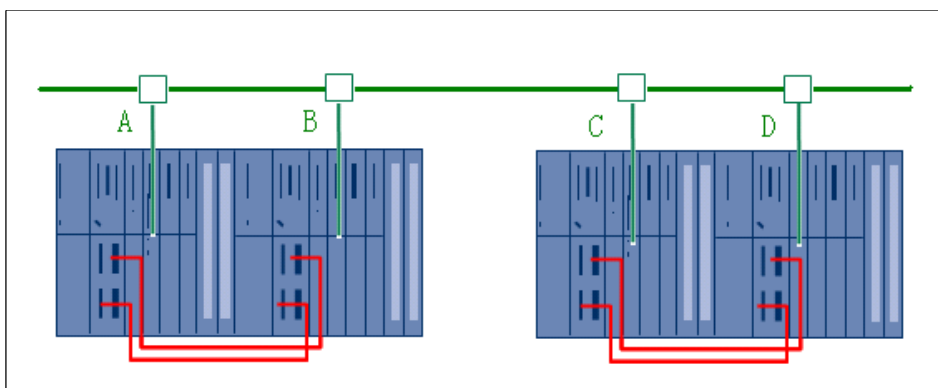


图 5-4 单一网络建立连接

如果网卡分别为 A、B、C、D，可能的连接为 A-C、A-D、B-C、B-D。如果在冗余的网络上四个网卡的连接数只能为两个，以图 5-4 为例，如果 D 网卡有故障，数据可以通过 A-C、B-C 的连接进行通信，A-D、B-D 通信故障，故障信息可以通过调用 SFC87 读出，连接建立之后，可以查看到建立的四个 S7 连接，如图 5-5 所示：

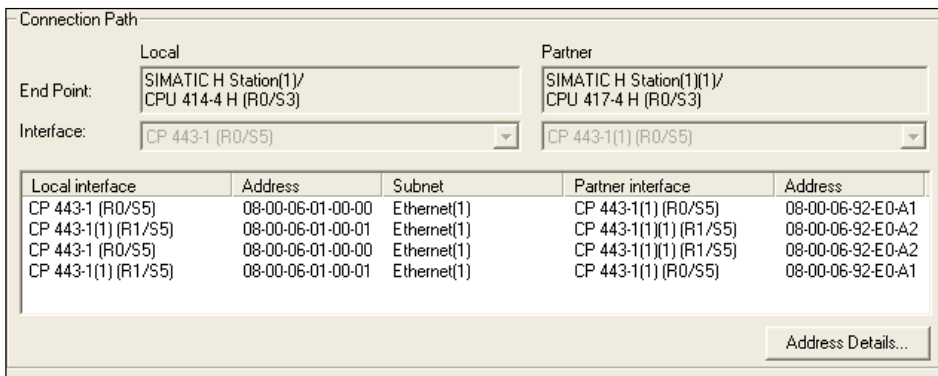


图 5-5 冗余的网络连接

从上到下的连接数定义为 0、1、2、3，通过 SFC87 的调用同时可以监控主连接和备有的连接数。SFC87 可以在 OB1 或循环中断组织块 OB3X 中调用，调用的例子如图 5-6 所示：

```

CALL "C_DIAG"           SFC87
  REQ    :=M1.1
  MODE   :=MB2
  RET_VAL:=MW4
  BUSY   :=M1.3
  N_CON  :=MW6
  CON_ARR:=P#M 20.0 BYTE 512
  
```

图 5-6 调用 SFC87 例子程序

SFC87 的参数解释如下，

REQ : 为 1 时，诊断任务激活。

MODE : 操作模式。操作模式有四种：

W#16#0 不传送连接状态到数据接收区，只传送确认信息到操作系统。

W#16#1 传送连接状态到数据接收区，传送确认信息到操作系统。

W#16#2 只在连接状态改变时传送连接状态到数据接收区，传送确认信息到操作系统。

W#16#3 只传送连接状态到数据接收区，不传送确认信息到操作系统。

RET_VAL：调用 SFC87 状态返回值。

BUSY：任务没有完成。

N-CON：指示上一次连接状态（DIS_PCON）变化的连接数。冗余连接可以建立多个，这里指其中一个连接。

CON_ARR：连接状态接收区。以 CPU 最大的连接数计算，每一个连接占用八个字节。本例以 CPU417-4H 为例，CPU 的连接数为 64 个，每一个连接需要占用八个字节，总共 512 个字节。如果连接状态输出区小，在 RET_VAL 值中将有错误信息。

参数 CON_ARR 其实是一个数组变量，每一个数组单元对应 CPU 的一个连接，数组的单元由一个八个字节的结构变量组成，表示每一个连接的状态。输出八个字节的连接状态含义如下：

内容	长度	含义
CON_ID	WORD	在 NETPRO 中组态的连接号，输出 W#16#FFFF 表示连接没有配置，CON_ARR[i].DIS_PCON 被置位表示连接被从新配置或被删除。
STAT_CON	BYTE	S7 连接或 S7 冗余连接当前状态，可能的输出值： B#16#00：S7 连接没有建立。 B#16#10：S7 冗余连接没有建立。 B#16#01：S7 连接正在被建立。 B#16#11： S7 冗余连接正在被建立。 B#16#02：S7 连接被建立。 B#16#12：S7 冗余连接被建立，但是连接没有冗余（例如 D 通 信处理器故障，只能建立 A-C、B-C 连接）。 B#16#13：S7 冗余连接被建立
PROD_CON	BYTE	当前正在运行连接的子连接，可能的值为 0、1、2、3。
STBY_CON	BYTE	备用连接的子连接，可能的值为 0、1、2、3。B#16#FF 表示没有备份连接，只有 S7 冗余连接有备用连接。

DIS_PCON	BOOL	CON_ARR[i].STAT_CON 输出值由 W#16#12 -> W#16#13 或 W#16#13 -> W#16#12 转变后被置 1。其他状态的变化不作考虑。 当 MODE=B#16#01 或 02 时，状态输出到接收区 CON_ARR 后该位 被复位；当 MODE=B#16#03 时，该位不变化。
RES0	BYTE	保留(B#16#00)
RES1	BYTE	保留(B#16#00)

5.4 利用 SFC90 控制 H 系统的连接状态和检测

当冗余 CPU 其中一个出现故障，这时只有一个 CPU 工作，更换故障的 CPU 上电后，需要向主 CPU 发送 link-up 请求，主 CPU 检测到 link-up 请求，Update 从 CPU，从 CPU 进入工作状态。link-up 和 Update 过程将影响主 CPU 对快速过程的处理，例如在 Update 过程中信号状态的变化不能被识别（执行 link-up 和 Update 的时间参考 H 手册），在这种情况下，通过 SFC90 可以终止 link-up 和 Update 过程，快速过程执行完成后再使能 link-up 和 Update 过程，通过调用 SFC90 还可以使 H-CPU 执行对数据存储区的检查，SFC90 在 OB1 或循环中断组织块 OB3X 中调用，调用的例子如图 5-7 所示：

```

CALL "H_CTRL"                                SFC90
REQ      :=M1.1
MODE     :=MB2
SUBMODE  :=MW4
RET_VAL  :=MW6
BUSY     :=M1.2

```

图 5-7 调用 SFC90 例子程序

SFC90 的参数解释如下：

REQ : 为 1 时，控制任务激活。

MODE : 操作模式。操作模式有七种：

B#16#3 取消 Link-up 过程。

B#16#4 使能 Link-up 过程。

B#16#1 取消 Updating 过程。

B#16#2 使能 Updating 过程。

B#16#20 取消在 SUBMODE 中指定的循环自检测部分。一个自检测部分只能取消一次。

B#16#21 添加在 SUBMODE 中指定的循环自检测部分。一个自检测部分在取消后才能添加。

B#16#22 立即执行在 SUBMODE 中指定的自检测部分。

SUBMODE : 操作子模式。操作子模式有六种:

- 0 SP7 – ASIC – 检测
- 1 Code 存储器检测
- 2 Data 存储器检测
- 3 操作系统代码校验和检测
- 4 Code 块校验和检测
- 5 冗余操作中对数据块、M 区、计数器、计数器的比较

RET_VAL : 调用 SFC90 状态返回值。

BUSY : BUSY=1 指示任务没有完成。

5.5 利用 FC125 监控 DP 从站工作状态

在程序中调用 FC125 可以判断系统中 DP 从站和模块的错误。它可以检测到丢失和故障的从站, S7-400H 具有冗余的 PROFIBUS 总线, 所以必须在 CPU 中调用 2 次, 分别监控两条 PROFIBUS (通过参数 DP_MASTERSYSTEM 识别) 总线上的从站。FC125 需要在 OB1 中调用, 在 OB1 中调用的程序如图 5-8 所示:

```

CALL "GLOBAL_DP_DIAG"                                FC125
CHECK_ACTIV          :=TRUE
EXTERNAL_DP_INTERFACE :=FALSE
DP_MASTERSYSTEM      :=1
DATA_FIELD           :="GLOBAL_DIAG_DB".DATA_FIELD    P#DB100.DBX0.0
SUM_SLAVES_DIAG      :="GLOBAL_DIAG_DB".SUM_SLAVES_DIAG DB100.DBW50
LIST_SLAVES_NOT_PRESENT:= "GLOBAL_DIAG_DB".LIST_SLAVES_NOT_PRESENT P#DB100.DBX52.0
LIST_SLAVES_ERROR    := "GLOBAL_DIAG_DB".LIST_SLAVES_ERROR P#DB100.DBX68.0
RETVAL               := "GLOBAL_DIAG_DB".RETVAL        DB100.DBW84
BUSY                 := "GLOBAL_DIAG_DB".BUSY          DB100.DBX86.0

CALL "GLOBAL_DP_DIAG"                                FC125
CHECK_ACTIV          :=TRUE
EXTERNAL_DP_INTERFACE :=FALSE
DP_MASTERSYSTEM      :=2
DATA_FIELD           :=DB101.DATA_FIELD              P#DB101.DBX0.0
SUM_SLAVES_DIAG      :=DB101.DBW50
LIST_SLAVES_NOT_PRESENT:=DB101.LIST_SLAVES_NOT_PRESENT P#DB101.DBX52.0
LIST_SLAVES_ERROR    :=DB101.LIST_SLAVES_ERROR       P#DB101.DBX68.0
RETVAL               :=DB101.DBW84
BUSY                 :=DB101.DBX86.0

```

图 5—8 调用 FC 125 例子程序

下面对每个参数进行说明。

CHECK_ACTIVE :为 1 时开始检测从站。

EXTERNAL_DP_INTERFACE :选择 DP 主站的接口，0 表示集成在 CPU 上的 DP 主站，1 表示外部 DP 主站接口，如 CP443-5 等。

DP_MASTERSYSTEM :PROFIBUS 的 ID 号，在硬件组态中可以查看到 PROFIBUS 的 ID 号。H 站有两条 PROFIBUS 网络，需要调用 FC125 两次，分别赋值两个 PROFIBUS 的 ID 号，数据类型为 INT。

DATA_FIELD :50 个字节的数据区，仅供 FC125 内部使用，数据类型为 POINTER。

SUM_SLAVES_DIAG :从站诊断的总数，数据类型为 INT。

LIST_SLAVES_NOT_PRESENT:检查到丢失的从站（软件已经组态但是现场没有连接硬件），16 个字节的数据区，每一个数据位对应一个从站，置 1 表示一个从站丢失，如果丢失的从站返回或维修好，相对应的位复位。数据类型为 POINTER。

LIST_SLAVES_ERROR :故障的从站（部分模块有故障，其他模块可以连续运行），16 个字节的数据区，每一个数据位对应一个从站，置 1 表示一个从站故障，如果故障的从站返回或维修好，相对应的位复位。数据类型为 POINTER。

RETVAL :调用 FC125 时的状态返回值，数据类型为 INT。

BUSY :为 1 表示当前正在执行。

例如，本例中检查到丢失的从站地址区为 DBB52~DBB67 16 个字节，每一字节 8 个位，每一个位表示一个从站，如果位被置 1，表示从站丢失。图 5-9 为从站编号的分配表，注意字节与站号排列次序：

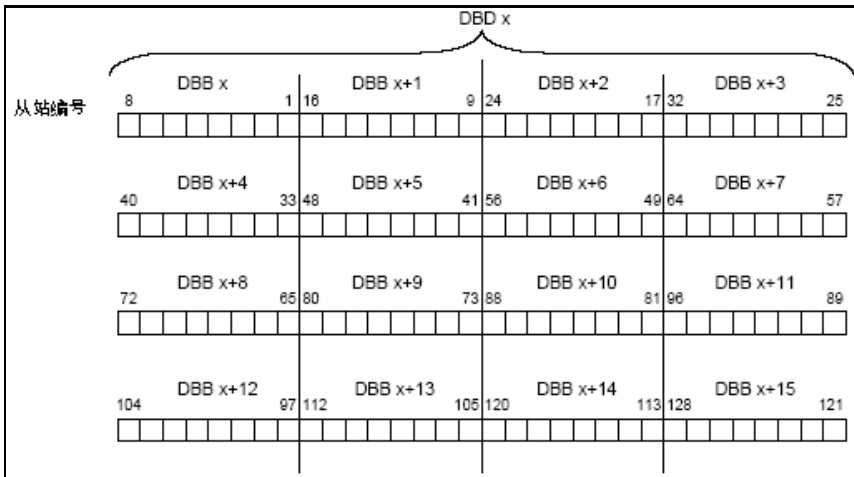


图 5-9 从站编号的分配表

在编程界面下选择菜单栏中 PLC->Monitor/Modify Variables，如图 5-10 所示：

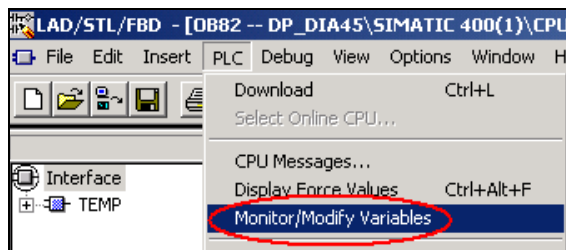



图 5-10 打开变量监控表

点击工具栏中的 ，观察在线诊断结果，如图 5-11 所示：

	Address	Symbol	Displ	Status value
1	DB100.DBD	52 "GLOBAL_	BIN	2#0000_1100_0000_0000_0000_0000_0000_0000
2	DB100.DBD	68 "GLOBAL_	BIN	2#0000_0000_0000_0000_0000_0000_0000_0000
3	DB101.DBD	52	BIN	2#0000_1100_0000_0000_0000_0000_0000_0000
4	DB101.DBD	62	BIN	2#0000_0000_0000_0000_0000_0000_0000_0000

图 5-11 监控变量

在第 1 栏中，显示 ID1 的 PROFIBUS 没有检测到的从站为 3, 4 号从站。

在第 3 栏中，显示 ID2 的 PROFIBUS 没有检测到的从站为 3, 4 号从站。

通过 FC125 可以获得故障的从站，从而排除故障。同时数据存储区可以作为接口被其它程序读取和调用（如 WINCC），更直观地在界面中显示故障从站和错误信息。

调用版本不同的 FB125 块可能会检测不到故障的从站

5.6 利用 OB 块诊断 S7-400H 系统

通过组织块 OB70、OB72、OB73 中的临时变量可以对 S7-400H 系统进行诊断，以 OB70 为例介绍诊断信息的读取方法。OB70 为 I/O 冗余错误，当 DP 主站错误、从站接口模块错误以及主站连接从站 I/O 的更改都会调用 OB70。在程序中建立 OB70，打开 OB70 可以看到接口参数，如图 5-12 所示：

Contents of Environment \Int		Name	Data T	Adresse
Interface	TEMP	OB70_EV_...	Byte	0.0
		OB70_EV_CLASS		
		OB70_FLT_ID	Byte	1.0
		OB70_PRI...	Byte	2.0
		OB70_OB_...	Byte	3.0
		OB70_RESERVE	Word	4.0
		OB70_INFO_1	Word	6.0
		OB70_INFO_2	Word	8.0
		OB70_INFO_3	Word	10.0
		OB70_DATE_TIM	Da...	12.0

图 5-12 OB70 接口参数

接口参数解释如下:

变量	类型	描述
OB70_EV_CLASS	BYTE	事件等级和 ID: ·B#16#72: 离开事件 (故障清除) ·B#16#73: 进来事件 (出现故障)
OB70_FLT_ID	BYTE	故障代码(可能的值:B#16#A2, B#16#A3)
OB70_PRIORITY	BYTE	优先级; 在 STEP7 硬件组态中可以分配
OB70_OB_NUMBR	BYTE	OB 号(70)
OB70_RESERVED_1	WORD	保留
OB70_INFO_1	WORD	与错误代码有关
OB70_INFO_2	WORD	与错误代码有关
OB70_INFO_3	WORD	与错误代码有关
OB70_DATE_TIME	DATE_AND_TIME	OB 块调用的时间

下列事件触发 OB70 的调用:

OB70_EV_CLASS	OB70_FLT_ID	OB70 开始事件
B#16#72	B#16#A2	DP 主站故障
B#16#73/B#16#72	B#16#A3	冗余的 DP 从站故障/清除

故障代码 B#16#A2, B#16#A3 的描述如下:

故障代码	位	描述
·B#16#A2-		
OB70_INFO_1:		有故障 DP 主站的逻辑地址
OB70_INFO_2:		保留
OB70_INFO_3:	0 to 7	保留
	8 to 15	有故障 DP 主站的系统 ID

·B#16#A3

OB70_INFO_1:	DP 主站的逻辑地址
OB70_INFO_2:	有故障 DP 从站:
0 to 14	S7 从站的逻辑地址或其他 DP 从站的诊断地址
15	I/O 识别, 从站的数据类型
OB70_INFO_3:	有故障 DP 从站:
0 to 7	站号
8 to 15	主站的系统 ID

如果触发 OB70 调用的事件出现, 可以将事件信息传送到全局变量 (例如 M 数据区) 中进行分析, OB70 程序例子如图 5-13 所示:

```
L   #OB70_EV_CLASS
T   MB   20

L   #OB70_FLT_ID
T   MB   21

L   #OB70_INFO_1
T   MW   22

L   #OB70_INFO_2
T   MW   24

L   #OB70_INFO_3
T   MW   26
```

图 5-13 OB70 程序示例

通过程序将触发 OB70 调用的事件信息记录到 MB20-MB27 中, 可以为后续的判断及显示使用。利用相同的方法, 通过 OB72、OB73 可以读取 S7-400H 系统其他的诊断信息, 这里不作描述。

第六章 冗余 I/O

6.1 冗余 I/O 的定义

当 I/O 模块以每两个一组组态成冗余对运行时，I/O 模板即认为是冗余配置。这意味着 S7-400H 系统的所有元件都可以冗余。使用冗余 I/O 可以使 H 系统提高可用性。

6.2 冗余 I/O 配置

以下为几种采用冗余 I/O 的典型配置

6.2.1 中央和扩展设备中的 I/O 冗余

信号模板成对安装在 CPU 0 和 CPU 1 子系统中，如图 6-1 所示：

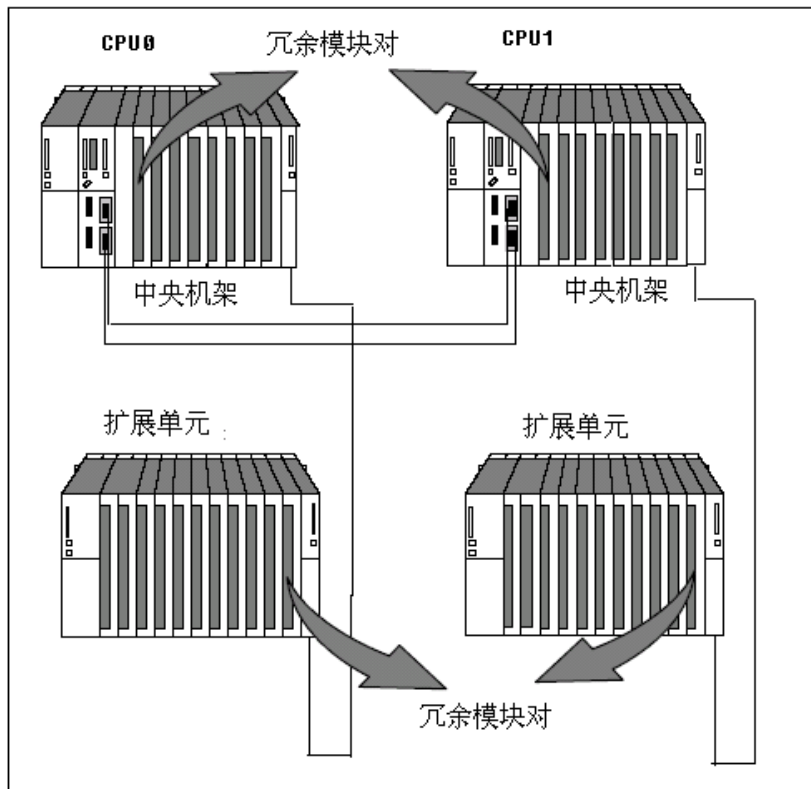


图6-1 中央和扩展设备中的I/O冗余

这样使 S7-400 的 I/O 模块具有冗余性。

注意：

当一个 CPU 因故障掉电时，连接的 I/O 模块被认为故障而不能被识别。

6.2.2. 单边 DP 从站中的 I/O 冗余

信号模板成对安装在带有有源底板总线的分布式 I/O 设备 ET200M 中，如图 6-2 所示：

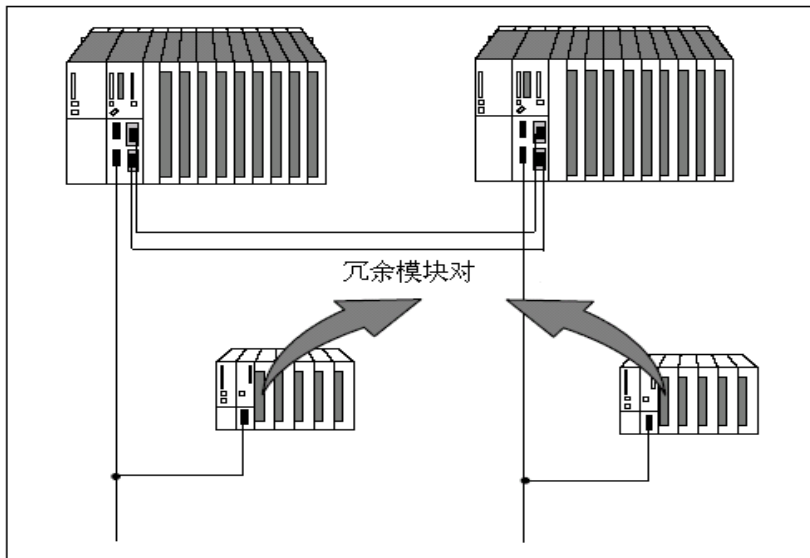


图6-2 单向DP从站中的I/O冗余

注意：

当其中一个从站PROFIBUS断线或主站CPU因故障掉电时，连接的I/O模块被认为故障而不能被识别。

6.2.3. 切换式DP 从站中的I/O冗余

信号模板成对安装在带有有源底板总线的分布式I/O设备ET200M中，如图6-3所示：

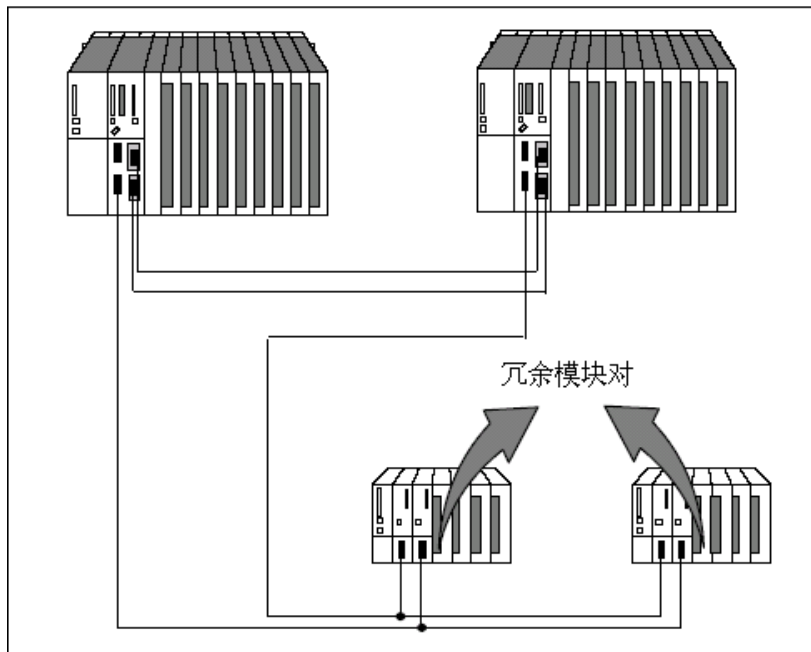


图6-3 切换式DP 从站中的I/O冗余

在这种方式下，即使连接I/O从站CPU故障也不会影响I/O模块的可用性。

6.2.4. 单机模式H CPU中的I/O冗余

连接方式如图6-4所示：

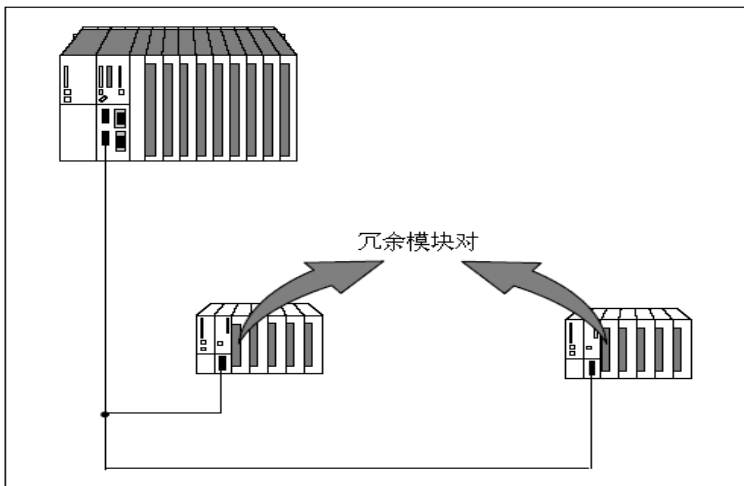


图6-4 单一模式冗余I/O

注意：

当主站CPU因故障而停机时，I/O冗余不能实现。

6.3 “I/O 冗余” 库函数及使用

6.3.1 “I/O冗余” 库函数

“I/O冗余”(Redundant IO(V1)) 库函数，随可选 H 包(STEP7 V5.3已包含)一起提供，以支持I/O模板冗余功能。

从Step7 V5.4开始，同时提供了“I/O 冗余” (Redundant IO(CGP)) 库函数，可以实现I/O通道冗余，提高了系统的可用性。

模板冗余与通道冗余的区别在于，当使用模板冗余时，模板上某一个通道故障，那么整个模板将被钝化，而使用通道冗余时，模板上某一个通道故障，不会影响到该模板其他通道的状态。

注意：在使用Step7 V5.4以上版本编程时，在整个程序中只能使用同一个库函数，即Redundant IO(V1)与Redundant IO(CGP)的功能块不能同时使用。

两个I/O冗余库函数的功能块名称完全相同，使用方法也完全相同。

库函数包含以下功能块：

- FC 450 RED_INIT 初始化功能
- FC 451 RED_DEPA 触发去钝化
- FB 450 RED_IN 读取冗余输入功能块

- FB 451 RED_OUT 控制冗余输出功能块
- FB 452 RED_DIAG 冗余 I/O 诊断功能块
- FB 453 RED_STATUS 冗余状态信息功能块

6.3.2 “I/O冗余”功能块的使用

在使用这些功能块前,首先在硬件配置中配置冗余 I/O 模块,然后参考表 6-1 在相应的 OB 块中调用对应的功能块。

表 6-1 可调用 I/O 冗余功能块的 OB 块

功能块	调用 OB
FC450 “RED_INIT”	1. OB72 CPU 冗余故障 FC450 只在事件发生后执行 B#16#33: 操作员进行主从切换 2. OB80 超时故障 FC450 只在启动事件 B#16#0A“重新设置恢复运行”后执行 3. OB100 暖重起 4. OB102 冷重起 如果冗余 I/O 连接到处于单机模式的冗余 CPU 时,在 OB80 中调用 FC450 功能块.
FC451“RED_DEPA”	当插入任一模块时,在 OB83 中调用 FC451, 可以使钝化模块维修后自动解除钝化.
FB450“RED_IN”	OB1 循环程序 OB30-OB38 定时中断
FB451“RED_OUT”	OB1 循环程序 OB30-OB38 定时中断
FB452“RED_DIAG”	OB72 CPU 冗余中断 OB82 诊断中断 OB83 插拔中断 OB85 程序运行错
FB453“RED_STATUS”	

6.4 冗余 I/O 的硬件安装和配置

如果要使用冗余 I/O，推荐按以下方法进行

- (1) 插入所有冗余使用的模板
- (2) 使用 HW Config 在对象属性中配置模板的冗余属性
- (3) 查找每个模板的冗余模板或使用缺省设置

集中配置:

将模板插入偶数机架插槽 X 中,冗余模板插在同一插槽的下一奇数机架中。如果模板插入插槽 X 的奇数机架中,建议在前一偶数机架同一插槽中插入冗余模板。

单边 DP 从站中的分布式配置:

如果从站中的模板插入插槽 X 中并且 DP 主站系统为冗余配置,建议冗余模板插在另一侧具有相同 PROFIBUS 地址的从站插槽 X 中。

单机模式切换式 DP 从站中的分布式配置

如果从站中的模板位于某一个 DP 地址的插槽 X 中,建议冗余模板位于具有下一 PROFIBUS 地址的从站插槽 X 中。

- (4) 设置 I/O 模板的其它冗余参数

用户程序可以处理的有效数值总是处于两个冗余模板的低位地址,因此应用程序只能使用低位地址而高位地址中的数值与应用程序无关。

注意:

冗余模板必须在输入或输出的过程映像区中设置,通常在相关的过程映像区能访问到。

必须设置冗余 I/O 的 DB 并分配未被使用的 DB 号,这些 DB 不是 FB 450 “RED_IN” 或 FB 451 “RED_OUT” 的背景数据块。

当用冗余模板时,在 “HW Config --> Properties CPU 41x-H>” Cycle/Flag” 栏中,设置如下参数: “OB 85 call on I/O access error --> Only incoming and outgoing errors”

6.5 冗余信号模板类型及型号

6.5.1 支持模板冗余的模板类型及型号

6.5.1.1 冗余数字量输入模板

参考表 6-2:

表 6-2 具有冗余功能的数字量输入模板

模板	订货号
中央机架:双通道冗余 DI	
DI 16xDC 24V alarm	6ES7 421-7BH01-0AB0

DI 32xDC 24 V	6ES7 421-1BL0x-0AA0
DI 32xUC 120V	6ES7 421-1EL00-0AA0
分布式:双通道冗余 DI	
DI16xDC 24 V, interrupt	6ES7 321-7BH00-0AB0
DI16xDC 24 V, interrupt	6ES7 321-7BH01-0AB0
DI16xDC 24 V	6ES7 321-1BH02-0AA0
DI32xDC 24 V	6ES7 321-1BL00-0AA0
DI 8xAC 120/230V	6ES7 321-1FF01-0AA0
DI 4xNAMUR [Ex ib]	6ES7 321-7RD00-0AB0
DI 16xNamur	6ES7 321-7TH00-0AB0
DI 24xDC 24 V	6ES7 326-1BK00-0AB0
DI 8xNAMUR [Ex ib]	6ES7 326-1RF00-0AB0

6.5.1.2.冗余数字量输出模板

参考表6-3:

表6-3 具有冗余功能的数字量输出模板

模板	订货号
中央机架:双通道冗余DO	
DO 32xDC 24V/0.5A	6ES7 422-7BL00-0AB0
DO 16xAC 120/230V/2A	6ES7 422-1FH00-0AA0
分布式:双通道冗余DO	
DO8xDC 24 V/0.5 A	6ES7 322-8BF00-0AB0
DO8xDC 24 V/2 A	6ES7 322-1BF01-0AA0
DO32xDC 24 V/0.5 A	6ES7 322-1BL00-0AA0
DO8xAC 120/230 V/2 A	6ES7 322-1FF01-0AA0
DO 16x24 V/10 mA [Ex ib]	6ES7 322-5SD00-0AB0
DO 16xDC 24V/0.5A	6ES7 322-8BH01-0AB0
DO 10xDC 24 V/2 A	6ES7 326-2BF01-0AB0

6.5.1.3.冗余模拟量输入模板

参考表6-4:

表6-4 具有冗余功能的模拟量输入模板

模板	订货号
中央机架:双通道冗余 AI	
AI 6x16 Bit	6ES7 431-7QH00-0AB0
分布式:双通道冗余 AI	
AI8x12Bit	6ES7 331-7KF02-0AB0

AI 8x16Bit	6ES7 331-7NF00-0AB0
AI 4x15Bit [Ex ib]	6ES7 331-7RD00-0AB0
AI 6x13Bit	6ES7 336-1HE00-0AB0

6.5.1.4.冗余模拟量输出模板

参考表6-5:

表6-5具有冗余功能的模拟量输出模板

模板	订货号
中央机架:双通道冗余 AO	
无	
分布式:双通道冗余AO	
AO4x12 Bit	6ES7 332-5HD01-0AB0
AO8x12 Bit	6ES7 332-5HF00-0AB0
AO4x0/4...20 mA [Ex ib]	6ES7 332-5RD00-0AB0

6.5.2.支持通道冗余模板类型及型号

参考表6-6

表6-6具有通道冗余功能的模板

模板	订货号
中央机架:双通道冗余 AO	
无	
分布式:双通道冗余模板	
AI8x16 Bit	6ES7 331-7NF00-0AB0
AO8x12 Bit	6ES7 332-5HF00-0AB0
DI16xDC 24 V, interrupt	6ES7 321-7BH01-0AB0
DO 16xDC 24V/0.5A	6ES7 322-8BH01-0AB0

注意：上述所列支持冗余设计的模板，可能会因为选择的系统的不同（Step7还是PCS7系统），以及新硬件的推出会有所不同。此处仅做参考，详细情况请参考当时最新的硬件选型目录，或从西门子相关部门获得最新信息。

6.6 冗余 I/O 参数设置及配置方式

6.6.1 使用数字量输入模板作为冗余 I/O

组态数字量输入模板时应设定以下参数：

- 差异时间：冗余输入信号可以不同的最大允许时间，如果在组态的差异时间过后输入数值仍存在差异说明信号已发生故障。
- H 系统对输入数值差异的响应

首先检查成对冗余模板的输入信号是否匹配。如果数值匹配，统一的数值将被写入输入过程映象的低位数据存储区。如果存在差异并且是首次存在差异，将被标记并开始差异计时。在差异时间期间，最近的匹配值（非差异值）将被写入低位地址模板的过程映象中。该步骤将重复直到在差异时间内数值再次匹配或超出差异时间。如果在组态的差异时间过后仍存在差异说明已发生故障，可根据以下策略查找故障页：

- （1）在差异时间期间最近的匹配值作为结果保留
- （2）一旦差异时间到显示以下出错报文

错误代码 7960：“Redundant I/O:discrepancy time at digital input expired, error not yet localized”（冗余 I/O 数字量输入差异时间到，还没有查到故障）。在静态出错映象中不能进行钝化和输入。在出现下一信号变换之前，应在差异时间到后进行组态的响应。

- （3）如果出现另一个信号变换，发生变换的模板将是正常模板，另一个模板将被钝化。

6.6.1.1 使用非冗余传感器的冗余数字量输入模板

将非冗余传感器的数字量输入模板安装在1002配置中，如图6-5所示：

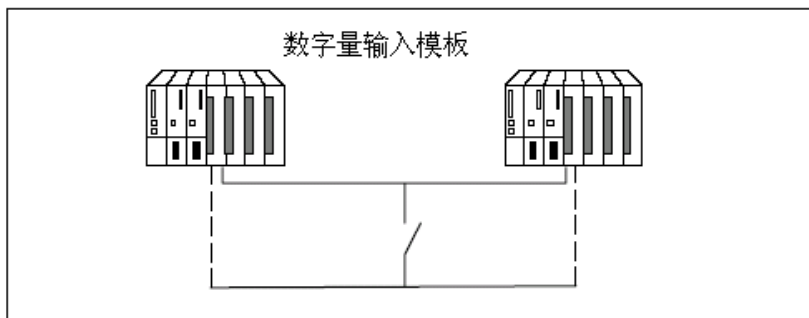


图6-5连接非冗余传感器的数字量输入信号

6.6.1.2 使用冗余传感器的冗余数字量输入模板

将带有冗余传感器的数字量输入模板安装在1002配置中，如图6-6所示：

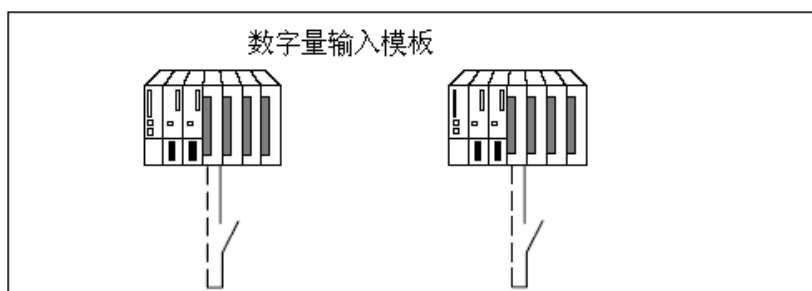


图6-6连接冗余传感器的数字量输入信号

使用冗余传感器可增加其可用性。差异分析可以检测除了非冗余电源故障以外的所有错误,可以通过安装冗余电源进一步增加系统的可用性。

在将一个传感器连接到几个数字量输入模板时冗余模板必须具有相同的基准电位。

6.6.2 冗余数字量输出模板

通过并行连接两个数字量输出模板或安全型数字量输出模板的两个输出(1002 配置)可以实现执行器的冗余控制。如图6-7所示:

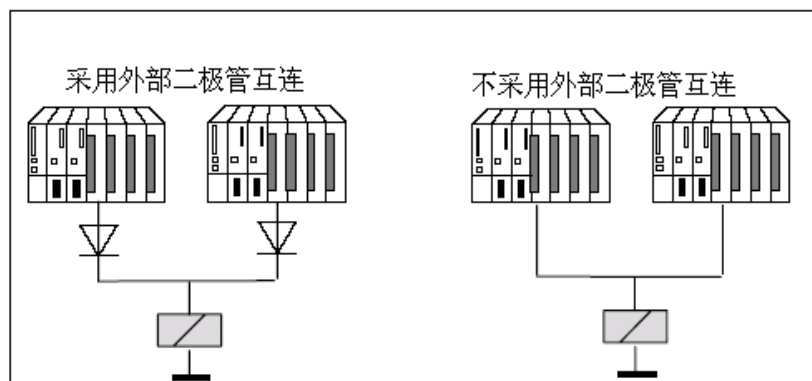


图6-7 1002 配置中的冗余数字量输出模板

数字量输出模板必须使用同一电源,通过二极管互连或无二极管互连:

表6-6列出了在冗余模式中可以通过二极管连接的数字量输出模板。

表6-6 输出模块连接类型

模块	外部需采用二极管	外部不需要采用二极管
6ES7 422-7BL00-0AB0	X	-
6ES7 422-1FH00-0AA0	-	X

6ES7 326-2BF01-0AB0	X	X
6ES7 322-1BL00-0AA0	X	-
6ES7 322-1BF01-0AA0	X	-
6ES7 322-8BF00-0AB0	X	X
6ES7 322-1FF01-0AA0	-	X
6ES7 322-8BH00-0AB0	-	X
6ES7 322-5SD00-0AB0	X	-

注：“X”表示选择“-”表示不可能

二极管电路的接线信息:

1N4003 ... 1N4007系列二极管比较合适,或其它技术指标满足 $V_r \geq 200\text{ V}$ 并 $I_F \geq 1\text{ A}$ 的二极管。

建议将模块的机架地和负载地分开,两种电路应当互连到等电势地。

6.6.3 使用模拟量输入模板作为冗余 I/O

应设定以下参数以组态模拟量输入模板冗余运行:

- 容差阈值(组态为测量范围最终数值的百分比)

如果两个模拟值都在容差阈值内则它们相等。

- 差异时间(冗余输入信号可以超出容差阈值的最大允许时间)

在组态的差异时间到后,如果还有输入数值的差异将出现故障。

如果将同一传感器连接到两个模拟量输入模板,差异时间的缺省设置通常即已足够.如果连接不同的传感器,尤其是连接温度传感器时必须增加差异时间。

- 应用值

应用值是指输入到用户程序中的两个模拟量输入数值中的数值。

系统将检查这两个模拟值是否在组态的容差阈值内.如果在容差阈值内应用值将被写入输入过程映象的低位数据存储区。如果存在差异并且是第一次差异,将被标记并且差异时间开始启动。在差异时间期间最近的有效值将被写入低位地址模板的过程映象中,并应用于当前过程。当差异时间到时,具有组态标准值的模板将被声明有效,其它模板被钝化。如果两个模板中的最大值被组态为标准值,该数值将被用于进一步程序执行,其它模板被钝化。如果设定了最小值,该模板将最小值用于过程而具有最大值的模板被钝化。不管何时被钝化的模板都将在诊断缓冲区里进行记录。如果在差异时间内差异被中止,系统将分析冗余输入信号。

6.6.3.1 接入非冗余传感器的冗余模拟量输入模板

在1002配置中接入非冗余传感器的模拟量输入模板,方法如图6-8所示:

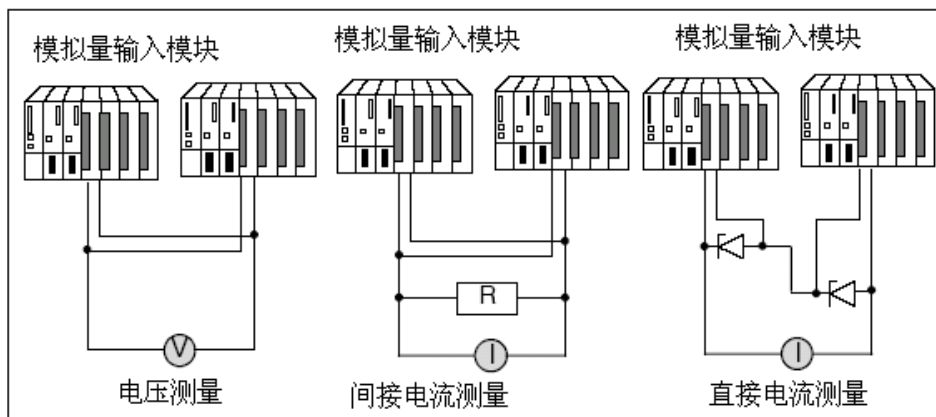


图6-8 带有一个传感器的10Q2 配置中的冗余模拟量输入模板

在将一个传感器连接到几个模拟量输入模板时应注意以下事项:

- 将电压传感器并联到模拟量输入模板(左图)。
- 可以使用外部阻抗将电流转换为电压以便使用并联的电压模拟量输入模板(右图)。
- 2线变送器从外部供电以便能在线维护模板。
- 使用冗余安全型模拟量输入模板可增加其可用性。

6.6.3.2用于间接电流测量的冗余模拟量输入模板

根据图6-8可适用以下模拟量输入的连接

- 相应的检测器类型可以是带电压输出的有源测量变送器、热电偶。
- 接入测量变送器、热电偶模块在硬件配置中去掉断线诊断功能。
- 检测器类型可以是有源4线测量变送器和无源2线测量变送器,输出范围 $\pm 20\text{mA}$ 、 $0\text{...}20\text{mA}$ 和 $4\text{...}20\text{mA}$ 。2线测量变送器通过外部辅助电源供电。
- 电阻和输入电压范围的选择准则应是测量精度、数值格式、最大分辨率和诊断。
- 除了所列选项以外,其它符合欧姆定律的输入电阻和电压组合也是可以的。但是应注意这样的混合可能数值格式、诊断功能和分辨率的降低。测量误差也主要取决于特定模板的分流电阻。
- 使用误差为 $\pm 0.1\%$ 和TK 15ppm的测量电阻。

6.6.3.3用于直接电流测量的冗余模拟量输入模板

根据图6-8右,模板量输入模块的线路连接要求

- 检测器类型可以是有源4线测量变送器和无源2线测量变送器输出范围 $\pm 20\text{mA}$ 、 $0\text{...}20\text{mA}$ 和 $4\text{...}20\text{mA}$ 。2线测量变送器通过外部辅助电源供电。
- 断线诊断功能只支持 $4\text{...}20\text{mA}$ 输入范围,其它单极或双极输入范围不支持。
- 只要满足模块指定的电压要求,可以选择任一BZX85 or 1N47..A(齐纳1.3W)系列的二极管,它的反锁电流越小越好。

- 这种接线电路包含指定的二极管有最大1μA的反锁电流所导致的基本测量误差,在20mA的范围和16位精度内,有2位的误差.

6.6.3.4 带有冗余传感器的冗余模拟量输入模板

对于双冗余传感器最好使用1002配置的安全型模拟量输入模板,如图6-9所示:

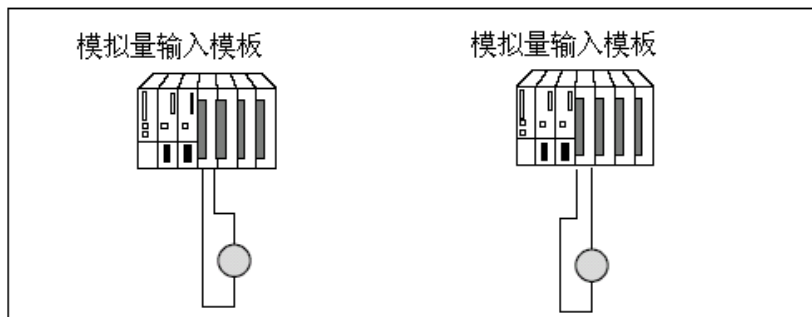


图6-9 带有2个传感器的1002配置中的冗余模拟量输入模板

使用冗余传感器可增加其可用性

差异分析也可以检测除非冗余有载电源故障以外的外部错误。

表6-7列出了可以冗余使用的模拟量输入模板连接带冗余或非冗余传感器。

表6-7模拟量模板连接类型

模板	冗余传感器	非冗余传感器
6ES7 431-7QH00-0AB0	X	X
6ES7 336-1HE00-0AB0	X	-
6ES7 331-7KF02-0AB0	X	X
6ES7 331-7NF00-0AB0	X	X
6ES7 331-7RD00-0AB0	X	X

注：“X”表示选择“-”表示不可能

6.6.4 冗余的模拟量输出模板

可以用并行的两个模拟量输出模板(1-of-2结构)的两个输出实现对一个执行机构冗余控制输出,如图6-10所示:

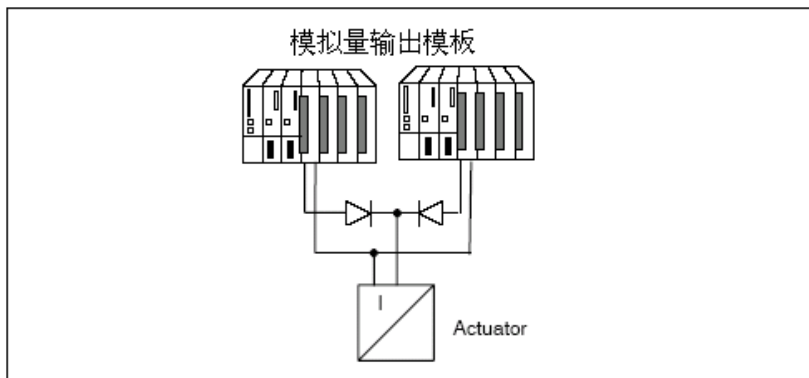


图6-10 冗余模拟量输出模板的接线

二极管电路的说明

- 1N4003 ... 1N4007系列二极管比较合适,或其它技术指标满足 $V_r \geq 200\text{ V}$ 并 $I_F \geq 1\text{ A}$ 的二极管。
- 建议将模块的机架地和负载地分开.两种电路应当互连到等电势地。

冗余操作要求模拟量输出模板采用电流输出(0 to 20 mA, 4 to 20 mA), 每块模板输出控制值的一半, 总的两块模板输出为控制值的全值。如果其中的一块模板检测有故障, 冗余的另一块模板输出为控制值的全值。由这个错误导致的输出模板的浪涌电流可以忽略。

6.7 模板的去钝化

输入信号在差异时间后仍然出现偏差, CPU将没有变化的模块钝化, 钝化的模板可以由以下事件重新激活

- 冗余系统启动
- 冗余系统切换到冗余运行模式
FB 452 "RED_DIAG"初始化后在系统切换到冗余运行模式时去钝化。该功能需要在OB 72(CPU冗余错)调用, FB 452 "RED_DIAG"也需要在OB 82 (诊断中断), OB 83 (拔插中断), OB 85 (程序运行错)。确保冗余I/O功能块的正确操作。
- 在运行过程中修改系统参数。
- 当至少有一个冗余模板通过功能 FC 451 "RED DEPA"解除钝化时。

当发生这些事件之一时, 去钝化将在 FB 450 "RED IN"中进行。在所有模板都去钝化后,将在诊断缓冲区中记录。

如果在单边集中式设备或单边 DP从站中使用冗余 I/O,在站故障/恢复或更换故障模板后必须去钝化。冗余模板使用 FC 451 可以触发一个完全去钝化。

6.7.2 使用状态字确定钝化的状态

状态字/控制字FB_RED_IN.STATUS_CONTROL_W 位于 FB 450 RED_IN 的背景数据块中，状态字节可以提供有关冗余 I/O 的状态信息。参考表6-8。

表6-8模板的钝化信息

位	含义
状态字节字节 1	
0	备用
1	备用
2	0 = 无可用模拟量输出模板 1 = 至少有一个可用模拟量输出模板
3	0 = 没有被 OB 85 钝化 1 = 至少有一个被 OB 85 钝化
4	0 = 没有被 OB 82 钝化 1 = 至少有一个被 OB 82 钝化
5	0 = 没有可用通道信息 1 = 有可用通道信息
6	0 = 没有钝化模板 1 = 至少有一个模板钝化
7	0 = 没有运行完全去钝化 1 = 运行完全去钝化

采用 MODUL_STATUS_WORD 确定模板对的钝化状态，MODUL_STATUS_WORD 位于 FB 453 RED_STATUS 的背景数据块中，有两个状态字节可提供模板对的状态，参考表6-9。

表6-9冗余模板的信息

位	含义
状态字节字节 1	
0	0 = 由 OB 82 触发的低位模板钝化 1 = 没有由 OB 82 触发的低位模板钝化

1	0 = 由 OB 82 触发的高位模板钝化 1 = 没有由 OB 82 触发的高位模板钝化
2	0 = 上溢或下溢对于模拟量输入模板 1 = 没有上溢或下溢
3	0 = 有可用通道信息 1 = 没有可用通道信息
4	0 = 差异时间到对于输入模板 1 = 差异时间未到
5	0 = 模板对存在差异对于输入模板 1 = 模板对没有差异
6	0 = 低位模板被钝化 1 = 低位模板去钝化
7	0 = 高位模板被钝化 1 = 高位模板去钝化
状态字节字节 2	
0	备用
1	备用
2	0 = OB 85 的输出事件后没有使能低位模板的去钝化 1 = OB 85 的输出事件后使能低位模板的去钝化
3	0 = OB 85 的输出事件后没有使能高位模板的去钝化 1 = OB 85 的输出事件后使能高位模板的去钝化
4	0 = OB 82 的输出事件后没有使能低位模板的去钝化 1 = OB 82 的输出事件后使能低位模板的去钝化
5	0 = OB 82 的输出事件后没有使能高位模板的去钝化 1 = OB 82 的输出事件后使能高位模板的去钝化
6	0 = 由 OB 85 触发的低位模板钝化

	1 = 没有由 OB 85 触发的低位模板钝化
7	0 = 由 OB 85 触发的高位模板钝化 1 = 没有由 OB 85 触发的高位模板钝化

6.8 编程顺序和例程

下面以一个简单的例子说明如何编程，本例程所使用的软件版本为STEPV5.3 SP2。过程如下：

- 打开SIMATIC Manager >新建一个项目RED_IO File>new...>输入项目名称例如 RED_IO，插入一个SIMATIC H Station，双击Hardware开始硬件配置。
- 依次插入机架，电源模块，CPU，同步模块，以太网卡，ET200M 站。
- 在ET200M站插入冗余的模块。

6.8.1 对冗余的DI模块进行配置

- (1) 双击DI模块>选择Redundancy tab>2 module>选择与该DI模块冗余的模块
- (2) 根据项目要求设置Time discrepancy(ms)差异时间
- (3) 根据项目要求设置reaction after discrepancy,有三种选择
 - ◇ AND Signals:如超过差异时间,冗余的DI通道信号相与后,进行程序处理。
 - ◇ OR Signals:如超过差异时间,冗余的DI通道信号相或 进行程序处理。
 - ◇ Use last valid value:如超过差异时间,取上次有效值进行程序处理。

设置好后点击 OK键确认。如图6-11所示：

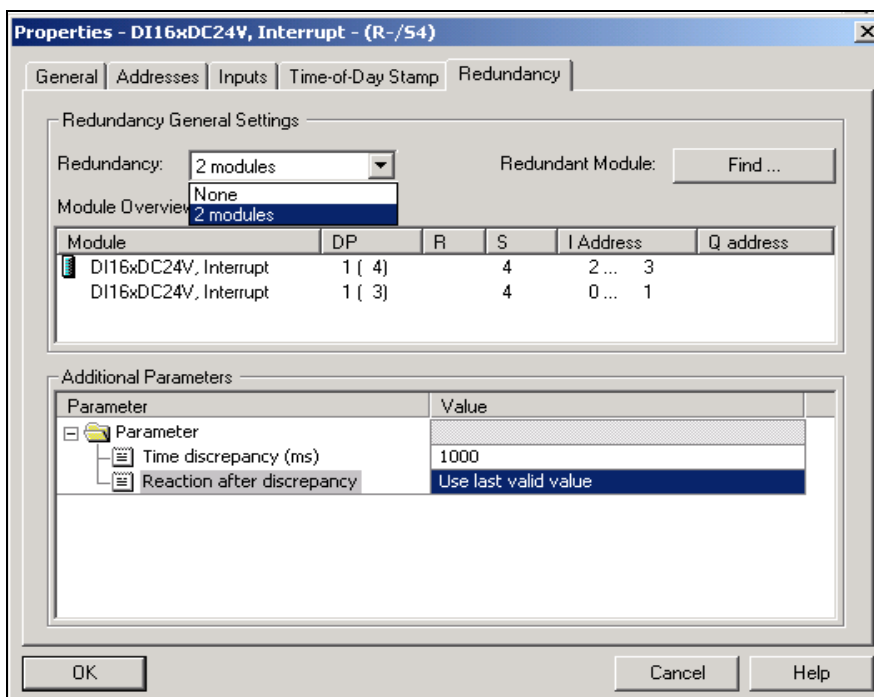


图6-11 冗余DI模板配置

6.8.2 对冗余的DO模块进行配置

双击DO模块>选择Redundancy tab>2 module>选择与该DO模块冗余的模块。设置完成后点击 OK 键确认，如图6-12所示：

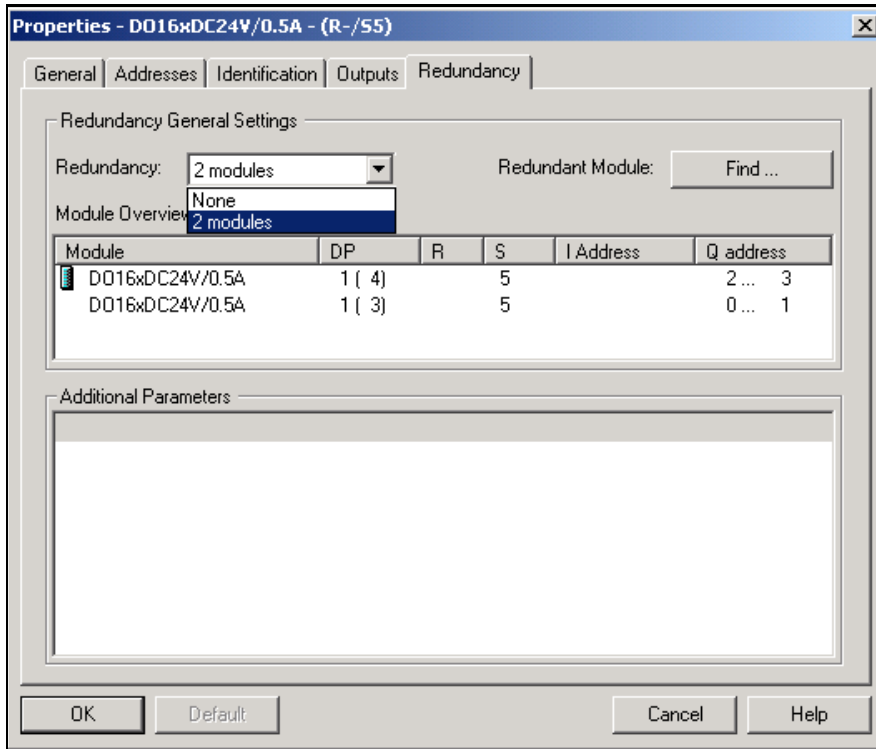


图6-12 冗余DO模板配置

6.8.3 对冗余的AI模块进行配置 (详细参数值设定见报标题6.6.3)

- (1) 双击AI模块>选择Redundancy tab>2 modules>选择与该AI模块冗余的模块。
- (2) 选择Inputs>选择相应的输入类型.(两个模块必须相同输入类型)
 - Tolerance windows(组态为测量范围最终数值的百分比)
 - Discrepancy time差异时间(冗余输入信号可以超出容差窗值的最大允许时间)
 - Applied value适用值:指输入到用户程序中的两个模拟量输入数值中的数值.可选择较低的值或较高的值

设置完成后点击 OK键确认，如图6-13所示：

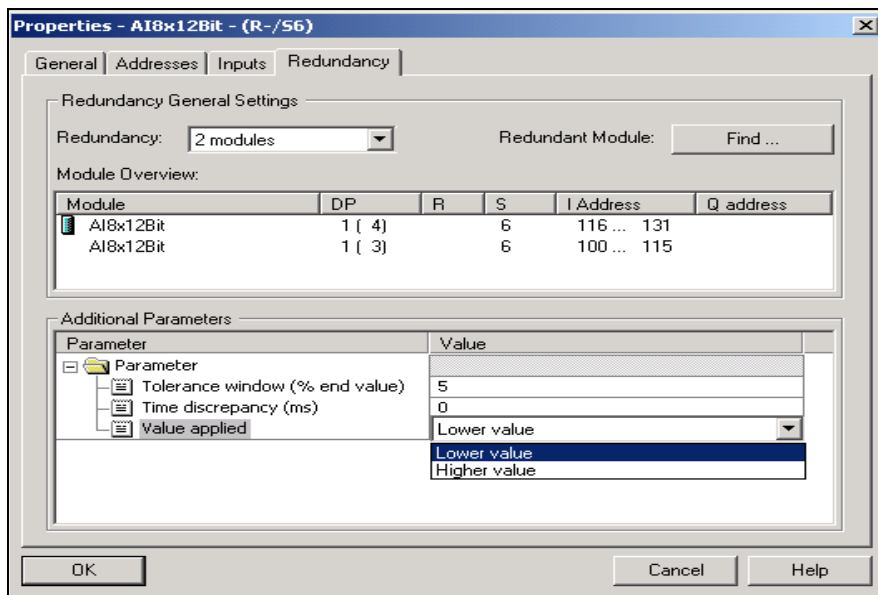


图6-13 冗余AI模板配置

6.8.4 冗余的AO模块进行配置

双击AO模块>选择Redundancy tab>2 modules>选择与该AO模块冗余的模块。设置好后点击 OK键确认，如图6-14所示：

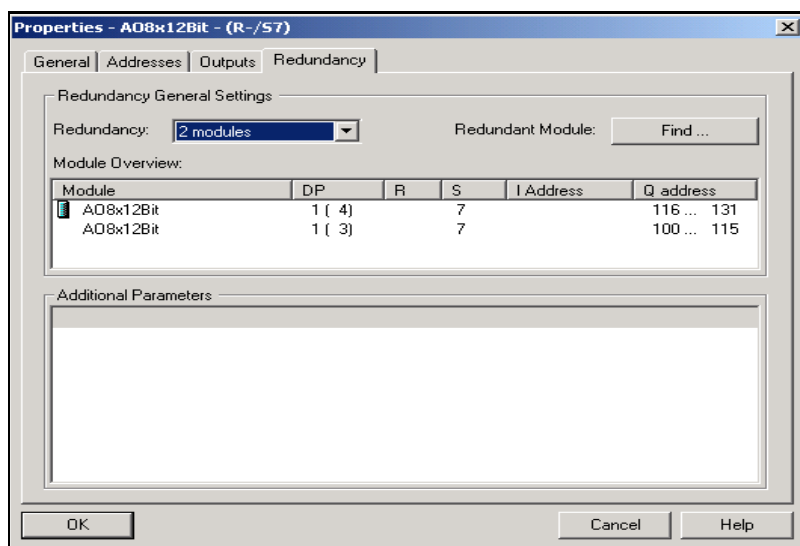


图6-14 冗余AO模板配置

所有硬件配置完成后,编译下载硬件组态到CPU中。

6.9 编程

- 在BLOCKS下插入错误处理组织OB块OB70、OB72、OB73、OB80、OB82、OB83、OB84、OB85、OB86、OB87、OB121、OB122，防止当错误发生时,冗余CPU进入停止模式。
- 分别在OB72、OB80、OB100中调用FC450“RED_INIT”。
- 分别在OB72，OB82,OB83,OB85中调用FB452“RED_DIAG”，如图6-15所示：

```
CALL "RED_INIT"  
RETURN_VAL:=MW204  
EXT_INFO :=MW206  
  
CALL "RED_DIAG", DB452  
RETURN_VAL:=MW216  
EXT_INFO :=MW218
```

图 6-15 OB72 程序示例

- 在OB83中调用FC451“RED_DEPA”,如图6-16所示：

```
CALL "RED_DEPA"  
RETURN_VAL:=MW40  
EXT_INFO :=MW42
```

图 6-16 OB83 程序示例

- 在OB1循环程序或OB30-OB38循环中断中先调用FB450“RED_IN”，再编写冗余I/O处理程序，程序最后调用FB451“RED_OUT”。如图6-17所示：

```
CALL "RED_IN", DB450  
RETURN_VAL:=MW30  
EXT_INFO :=MW32  
  
I  
//用户应用程序  
  
CALL "RED_OUT", DB451  
RETURN_VAL:=MW32  
EXT_INFO :=MW36
```

图 6-17 OB1 程序示例

- 下载全部程序，观察实验结果。

6.10 SIMATIC PCS7 MTA (Marshaled Termination Assemblies)

在6.6中已经介绍了冗余AI,AO,DI,DO的接线方式，由于每个通道都需要单独接线，对于用户来说，比较烦琐。SIMATIC PCS 7 Marshaled Termination Assemblies (MTA)提供了简单、快速且可靠地将现场设备、传感器和执行器连接到ET 200M 冗余远程I/O 模块的手段。它们有助于大大降低布线和调试的工作量及费用，并可避免接线错误。（同样使用于STEP7）

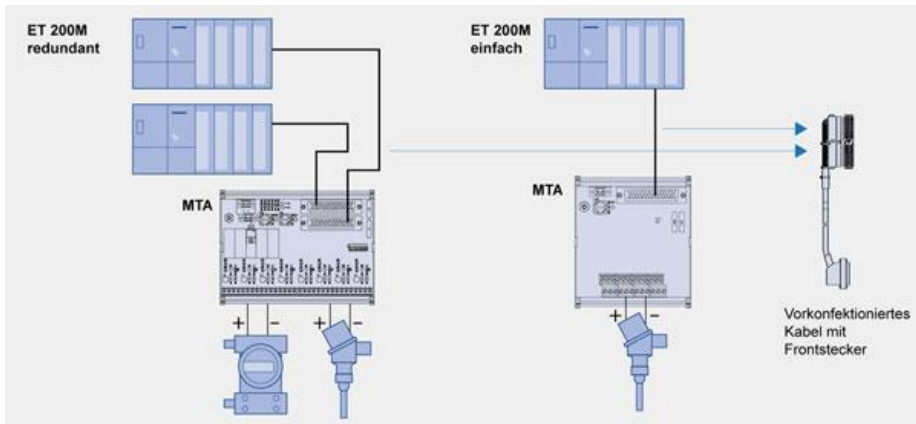


图 6-17 MTA 接线示意图

每种 SIMATIC PCS 7 MTA 专门对应于特定的 ET 200M I/O 模块(请参阅‘设计’下面的对应关系表)。MTA 设计适用于标准 I/O 模块以及冗余和故障安全 I/O 模块。SIMATIC PCS 7 MTA 通过 3 米或 8 米预装配电缆连接至 I/O 模块。

MTA 类型	输入输出范围	MTA 订货号	ET200M模板订货号	连接电缆订货号
8 Channels, AI	1 - 5 V 0/4 - 20 mA	6ES7650-1AA51-2XX0	6ES7331-7NF00-0AB0	6ES7922-3BD00-0AS0 (3 m) 6ES7922-3BJ00-0AS0 (8 m)
8 Channels, AO	4 - 20 mA	6ES7650-1AB51-2XX0	6ES7332-5HF00-0AB0	6ES7922-3BD00-0AS0 (3 m) 6ES7922-3BJ00-0AS0 (8 m)
16 Channels, DO	24 V DC, 0.5 A	6ES7650-1AD10-2XX0	6ES7322-8BH01-0AB0	6ES7922-3BD00-0AT0 (3 m) 6ES7922-3BJ00-0AT0 (8 m)
6 Channels F-AI (failsafe)	4 - 20 mA	6ES7650-1AH51-5XX0	6ES7336-1HE00-0AB0	6ES7922-3BD00-0AS0 (3 m) 6ES7922-3BJ00-0AS0 (8 m)
16 Channels, DI	24 V DC	6ES7650-1AC11-3XX0	6ES7321-7BH01-0AB0	6ES7922-3BD00-0AM0 (3 m) 6ES7922-3BJ00-0AM0 (8 m)
24 Channels F-DI (failsafe)	24 V DC	6ES7650-1AK11-7XX0	6ES7326-1BK00-0AB0 and 6ES7326-1BK01-0AB0	6ES7922-3BD00-0AS0 (3 m) 6ES7922-3BJ00-0AS0 (8 m)
10 Channels F-DO (failsafe)	24 V DC, 2 A	6ES7650-1AL11-6XX0	6ES7326-2BF01-0AB0	6ES7922-3BD00-0AN0 (3 m) 6ES7922-3BJ00-0AN0 (8 m)
10 Channels DO relay (failsafe)	110 - 220 V AC, 5 A; 24 V DC, 5 A	6ES7650-1AM30-6XX0	6ES7326-2BF01-0AB0	6ES7922-3BD00-0AS0 (3 m) 6ES7922-3BJ00-0AS0 (8 m)

表6—7 支持冗余模板的MTA及电缆订货号

第七章 软件冗余的原理和配置

7.1 软件冗余基本信息介绍

软件冗余是 Siemens 实现冗余功能的一种低成本解决方案，可以应用于对主备系统切换时间为秒级的控制系统中。

7.1.1 系统结构

Siemens 软件冗余系统的软件、硬件包括：

- (1) 1 套 STEP7 编程软件（V5.2 或更高）加软冗余软件包(V1.x)；
- (2) 2 套 PLC 控制器及 I/O 模块，可以是 S7-300（313C-2DP，314C-2DP，31X-2DP）或 S7-400（全部 S7-400 系列 CPU）系统；
- (3) 3 条通讯链路，主系统与从站通讯链路（PROFIBUS 1）、备用系统与从站通讯链路（PROFIBUS 2）、主系统与备用系统的数据同步通讯链路（MPI 或 PROFIBUS 或 Ethernet）；
- (4) 若干个 ET200M 从站，每个从站包括 2 个 IM153-2 接口模块和若干个 I/O 模块；Y-Link 不能用于软冗余系统；
- (5) 除此之外，还需要一些相关的附件，用于编程和上位机监控的 PC-Adapter（连接在计算机串口）或 CP5611（插在主板上的 PCI 槽上）或 CP5511（插在笔记本的 PCMCIA 槽里）、PROFIBUS 电缆、PROFIBUS 总线链接器等。

系统架构如图 7-1 所示：

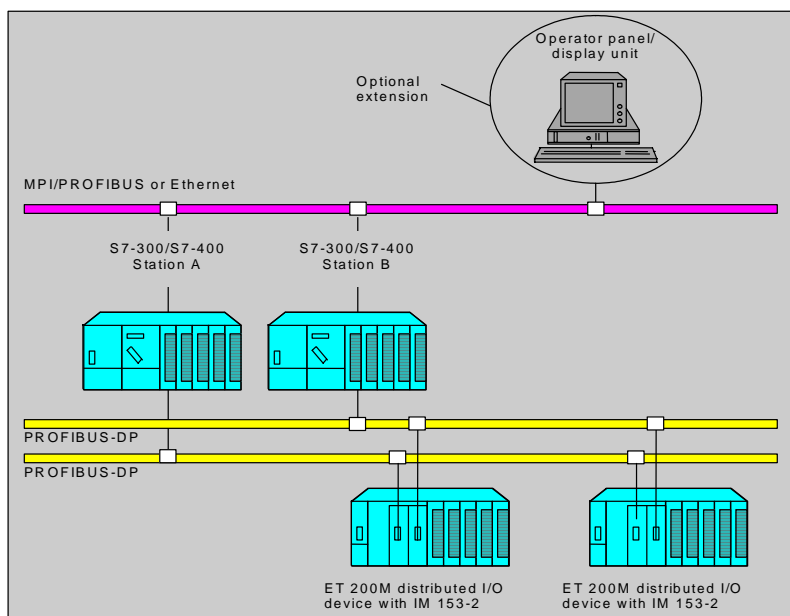


图 7-1 软冗余的系统架构

可以看出，系统是由两套独立的 S7-300 或 S7-400 PLC 系统组成，软冗余能够实现：

主机架电源、背板总线等冗余；PLC 处理器冗余；PROFIBUS 现场总线网络冗余（包括通讯接口、总线接头、总线电缆的冗余）；ET200M 站的通讯接口模块 IM153-2 冗余。

软冗余系统由 A 和 B 两套 PLC 控制系统组成。开始时，A 系统为主，B 系统为备用，当主系统 A 中的任何一个组件出错，控制任务会自动切换到备用系统 B 当中执行，这时，B 系统为主，A 系统为备用，这种切换过程是包括电源、CPU、通讯电缆和 IM153 接口模块的整体切换。系统运行过程中，即使没有任何组件出错，操作人员也可以通过设定控制字，实现手动的主备系统切换，这种手动切换过程，对于控制系统的软硬件调整，更换，扩容非常有用，即 Altering Configuration and Application Program in RUN Mode。

7.1.2 系统工作原理

在软冗余系统进行工作时，A、B 控制系统（处理器，通讯、I/O）独立运行，由主系统的 PLC 掌握对 ET200 从站中的 I/O 控制权。A、B 系统中的 PLC 程序由非冗余（non-duplicated）用户程序段和冗余（redundant backup）用户程序段组成，主系统 PLC 执行全部的用户程序，备用系统 PLC 只执行非冗余用户程序段，而跳过冗余用户程序段。

软冗余系统内部的运行过程参考图 7-2。

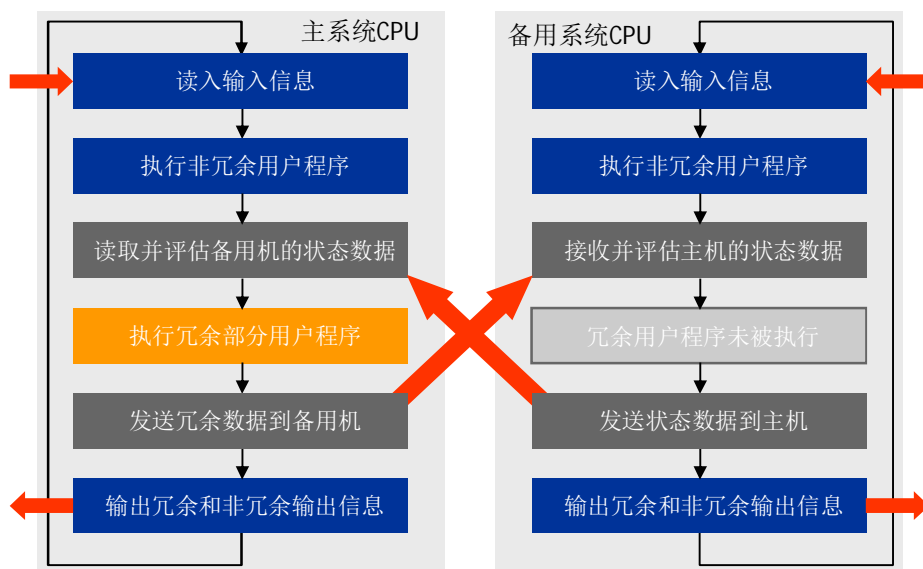


图 7-2 软冗余系统内部的运行过程

主系统的 CPU 将数据同步到备用系统的 CPU 需要 1 到几个程序扫描循环，如图 7-3 所示：

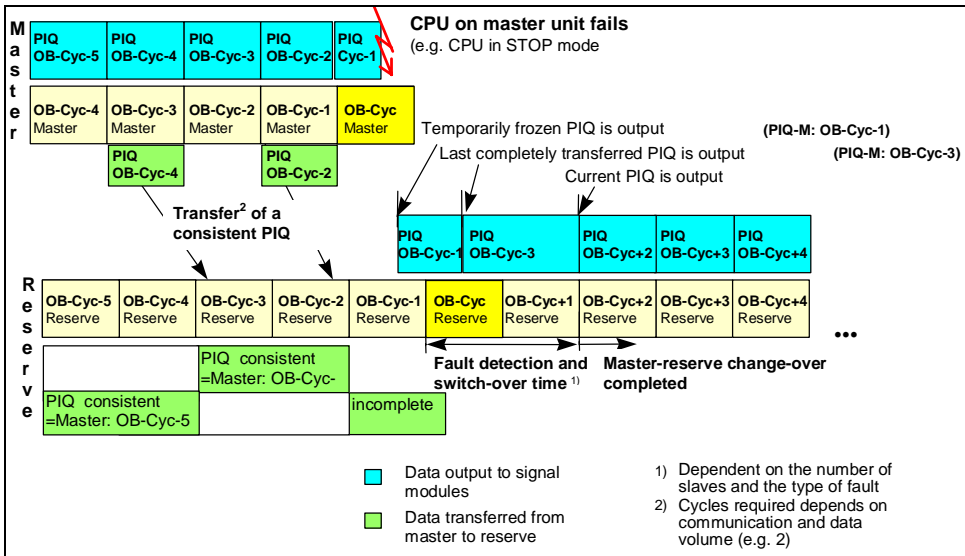


图 7-3 软冗余同步原理

数据同步所需要的时间取决于同步数据量的大小和同步所采用的网络方式，MPI 方式周期最长，PROFIBUS 方式适中，Ethernet 网方式最快。同步通信效率见表 7-1

表 7-1 同步通信效率

PROFIBUS 网络 1.5 Mbaud	Ethernet 以太网 10 Mbaud	MPI网络 187.5 kbaud
每60ms 传送240个字节数据	每48ms 传送 240个字节数据	每152ms 传送76个字节数据

用户需要在初始化程序中（OB100）定义冗余部分的数据区，该数据区可以包括：一个过程映象区（process image area），一个定时器区（IEC timer area），一个计数器区（IEC counter area），一个位地址区（memory address area）和一个数据块区（data block area），S7-300 同步的最大数据量为 8 kBytes，S7-400 同步的最大数据量 64kBytes。

主备系统的切换时间 = 故障诊断检测时间 + 同步数据传输时间 + DP 从站切换时间

如果 CPU 的故障是停机或断电，则故障诊断为大约 100—1000 毫秒，315—2DP 同步 1000 字节的数据所需的时间大约为 200—300ms，8 个 DP 从站的切换时间在 100ms 左右。可以在软冗余手册当中找到关于切换时间的具体说明。

无论控制程序循环扫描到哪里，当前激活的系统（即主系统）随时都会接收并处理报警信息，这样，在主系统 A 与备用系统 B 进行切换过程中产生的 alarm 存在被丢失的可能。

7.1.3 产品订货信息

软件冗余系统相关产品的订货信息参考表 7-2。

表 7-2 软件冗余系统相关产品的订货信息

产品名称	订货号	描述
CPU 系列		
CPU314C-2DP	6ES7 314-6CF00-0AB0	S7-300 系列只能够实现软件冗余，无硬件冗余的功能。
CPU313C-2DP	6ES7 313-6CE00-0AB0	
CPU 31x-2DP	6ES7 315-2AFxx-0AB0	
	6ES7 315-2AG10-0AB0	
	6ES7 316-2AGxx-0AB0 6ES7 318-2AJxx-0AB0	
CPU 412-1	6ES7 412-1XFxx-0AB0	S7-400 全系列的 CPU 都可以应用于软冗余系统；
CPU 412-2	6ES7 412-1FK03-0AB0 6ES7 412-2XGxx-0AB0	
CPU 413-1	6ES7 413-1XGxx-0AB0	S7-400 H 系列的 CPU 属于硬件冗余方式，相对于软冗余，硬件冗余系统切换速度快，主备 CPU 中的数据 and 事件保证完全一致，适于高可靠性应用场合，成本较高
CPU 413-2DP	6ES7 413-2XGxx-0AB0	
CPU 414-1	6ES7 414-1XGxx-0AB0	
CPU 414-2DP	6ES7 414-2XGxx-0AB0	
CPU 414-3DP	6ES7 414-2XJxx-0AB0	
	6ES7 414-3XJxx-0AB0	
CPU 416-1	6ES7 416-1XJxx-0AB0	
CPU 416-2DP	6ES7 416-2XKxx-0AB0	
CPU 416-3DP	6ES7 416-2XLxx-0AB0	
CPU 417-4	6ES7 416-3XLxx-0AB0	
	6ES7 417-4XLxx-0AB0	
CP 通讯处理器系列（数据同步 Redundant-backup link）		

CP 342-5 PROFIBUS通讯模块	6ES7 342-5DA00-0XE0 6GK7 342-5DA02-0XE0	
CP 343-1 Ethernet 通讯模块	6GK7 343-1BA00-0XE0 6GK7 343-1EX11-0XE0	
CP 443-5 Extended PROFIBUS通讯模块	6EK7 443-5DXxx-0XE0	
CP 443-1 ISO Ethernet 通讯模块	6EK7 443-1BXxx-0XE0	
ET200 系列		
2x DP slave interface IM 153-2	6ES7 153-2AA02-0XB0 6ES7 153-2AB01-0XB0	
适于ET200M的所有数字量、模拟量 I/O 模板	参照 S7-300 选型样本和 STEP 7 的硬件组态窗口中 ET200 文件夹中的 I/O 模块	S7-300 模板手册的下载路径： http://support.automation.siemens.com/CN/view/zh/8859629
CP 341	6ES7 341-1xH01-0AE0	串口通讯模板
FM 350	6ES7 350-1AH0x-0AE0	计数器功能模板
相关软件及附件		
STEP7 编程软件	6ES7810-4CC07-0YA5	
软冗余软件包 (Software Redundancy)	6ES7862-0AC01-0YA0	包括冗余功能块的安装软件和授权
PROFIBUS标准电缆	6XV1 830-0EH10	按米订货
RS485总线连接器	6ES7 972-0BA12-0XA0	PROFIBUS 网络连接头
CP5611通讯卡	6GK1 561-1AA00	进行编程和上位机监控的通讯卡

注：

1) 只有 S7-300 315-2DP (除 CPU314C-2DP、CPU313C-2DP) 型号以上的 PLC 才支持软冗余功能, 所有的 S7-400 都支持软冗余功能;

2) 主系统与备用系统的 CPU 型号可以不同, 如主系统采用一套 S7-400 系统, 而备用系统采用一套 S7-300 系统;

7.2 软件冗余系统的调试过程

在计算机上首先安装 STEP7 5.x 软件和软冗余软件包, 软冗余软件光盘包括了冗余功能程序块库、不同系统结构的例子程序和软冗余使用手册。

7.2.1 不同系统结构方式下的功能块:

在安装完软冗余的软件后, 可以在 STEP7 当中找到例子程序和功能程序块库, 如图 7-4 所示:

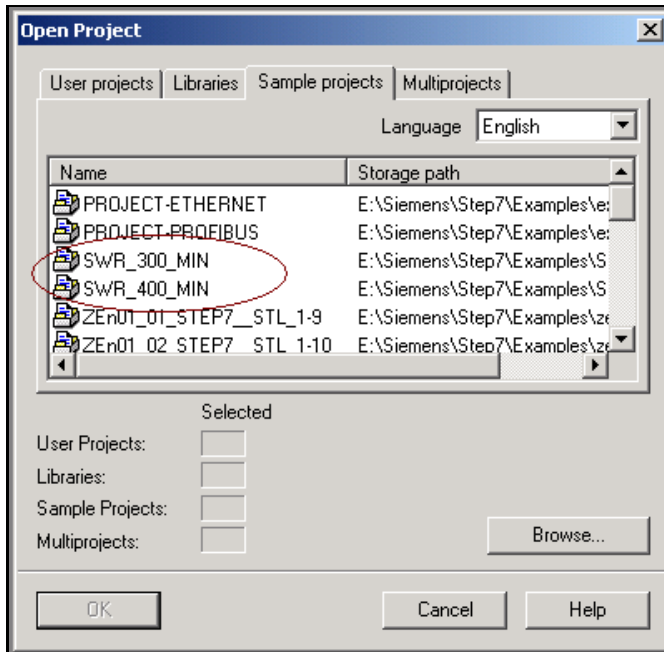


图7-4 软件冗余的例子程序

以上图 7-4 显示的是在 STEP7 中使用 OPEN 菜单打开软冗余的例子程序, 例子程序中采用 MPI 链路实现数据的同步。在 Libraries 找到不同 CPU 以及不同链接方式下可以使用的功能块, 如图 7-5 所示:

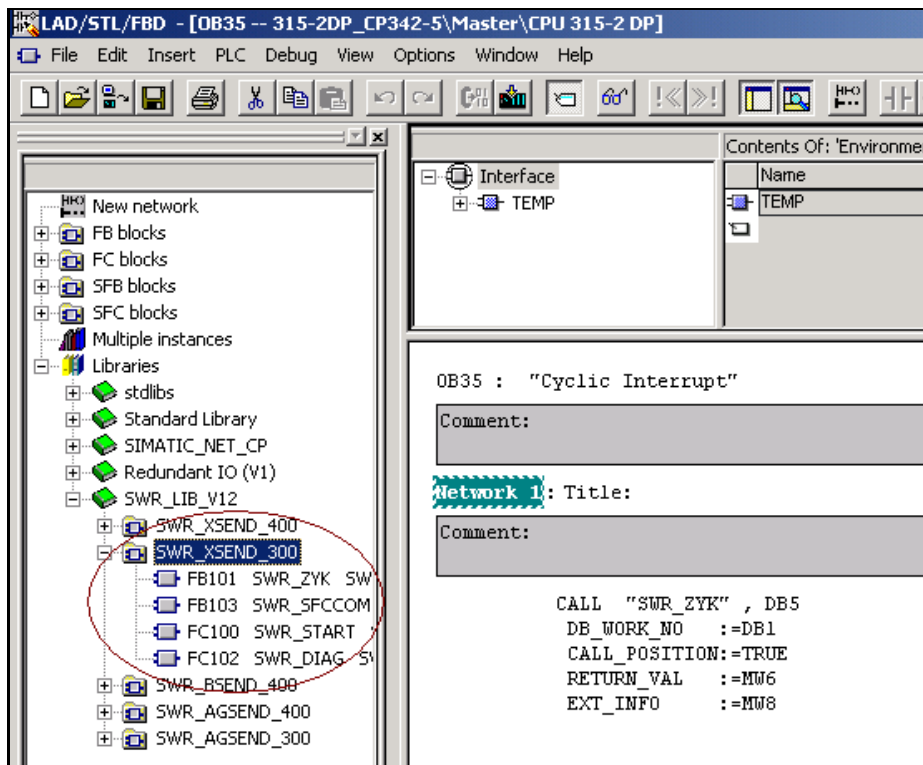


图7-5

软件冗余功能块

下面的两个表格分别说明了采用 S7-300, S7-400 进行软件冗余时, 可采用的不同网络连接以及所需调用的程序功能块包, 参考表 7-3 和 7-4。

表 7-3 S7-300 进行数据同步使用的程序库

Select this package	For this network	And this connection type	Remarks
XSEND_300	MPI	Permanently configured connection	Network connected to MPI interface of CPU
AG_SEND_300	PROFIBUS	FDL connection	Network connected via CP342-5
	Industrial Ethernet	ISO connection	Network connected via CP345-1

表 7-4 S7-400 进行数据同步使用的程序库

Select this package	For this network	And this connection type	Remarks
XSEND_400	MPI	Permanently configured connection	Network connected to MPI interface of CPU
AG_SEND_400	PROFIBUS	FDL connection	Network connected via CP443-5
	Industrial Ethernet	ISO connection	Network connected via CP443-1

BSEND_400	MPI	S7 connection	Network connected to MPI interface of CPU
	PROFIBUS		Network connected via CP 443-5
	Industrial Ethernet		Network connected via CP 443-1

从以上的表格中可以看到，软冗余系统当中可以采用 MPI、PROFIBUS、Ethernet 三种网络实现主系统和备用系统之间的数据同步（Redundant-backup link）。

采用 BSEND_400 包中的功能块进行数据同步时，无论数据同步的连接是什么方式，都需要在 STEP7 的 NETPRO 窗口中组态一个 S7 Connection。除了 BSEND_400 中之外，使用其他功能块：

- 采用 MPI 网络时，直接使用 PLC 的编程口进行数据同步，不需要进行连接（Connection）的组态，但 MPI 数据同步的效率低。
- 采用 PROFIBUS 网络时，需要使用一对 CP 通讯卡（CP342-5 或 CP443-5），在 STEP7 的 NETPRO 窗口中组态主系统和备用系统之间的 FDL 连接。
- 采用 Ethernet 网络时，需要使用一对 CP 通讯卡（CP343-1 或 CP443-1），在 NETPRO 窗口中组态主系统和备用系统之间的 ISO 连接。

冗余功能块的说明参考表 7-5。

表 7-5 冗余功能块说明：

名称	描述
FC 100 'SWR_START'	初始化程序块，定义系统运行的参数。
FB 101 'SWR_ZYK'	循环调用的数据同步功能块，将主系统中的冗余数据复制到备用系统当中
FC 102 'SWR_DIAG'	诊断功能块，在 OB86 中调用，将得到的诊断数据提供给 FB101 使用。
FB 103 'SWR_SFCCOM'	在该块内部调用 SFC 65 'X_SEND' 和 SFC 66 'X_RCV' 功能块，实现采用 MPI 网络的数据同步。
FB 104 'SWR_AG_COM'	在该块内部调用 FC 5 'AG_SEND'，FC 6 'AG_RCV'，实现采用 PROFIBUS 或 Ethernet 网络的数据同步。
FB 105 'SWR_SFBCOM'	在该块内部调用 SFB 12 'BSEND' and SFB 13 'BRCV' 功能块，实现 MPI 或 PROFIBUS 或 Ethernet 网络的数据同步，只能在 S7-400 中调用。
DB_WORK_NO	冗余软件块使用的数据区，用户不需要生成该数据块，也不使用该数据块中的数据；
DB_SEND_NO	用于主系统发送同步数据到备用系统的发送数据区（包括了主系统的 M、T、C、DB 等区域的数据），用户不需要生成该数据块，也不使用该数据块中的数据；

DB_RCV_NO	用于备用系统接收来自自主系统的同步数据的接收数据区，用户不需要生成该数据块，也不使用该数据块中的数据；
DB_A_B_NO	从A站到B站，非冗余同步的数据的收发区；
DB_B_A_NO	从B站到A站，非冗余同步的数据的收发区；
DB_COM_NO	FB101的背景数据块，包括了数据同步链路的状态，控制等信息，用户需要生成该数据块，DBW8为状态字，DBW10为控制字，；
FC 5 'AG_SEND'	PROFIBUS网络中，实现FDL链接，在FB104内部调用的发送块，用户需要生成该程序块，并下载到PLC当中。
FC 6 'AG_RCV'	PROFIBUS网络中，实现FDL链接，在FB104内部调用的接收块，用户需要生成该程序块，并下载到PLC当中。

注：

1) 用 PROFIBUS 或 Ethernet 方式进行数据同步时，需要在 OB100 当中调用 FB101，而 FB101 内部调用了 FB104、FC5、FC6，所以您必须手动地将 FB104、FC5、FC6 插入到项目当中，插入的方法是：在程序中调用一次，再将这条语句删掉。

2) 在 OB100 中调用 FC100 功能块时，PLC 会自动创建一些与 FC100 参数相关的程序段和数据块，所以当你更改了 FC100 的参数时，应该对 PLC 进行 Reset 操作，再重新下载项目的软硬件到 PLC 当中。

7.2.2 实例

例子由两套 315-2DP 和一个 ET200M 从站组成，系统结构如图 7-6 所示：

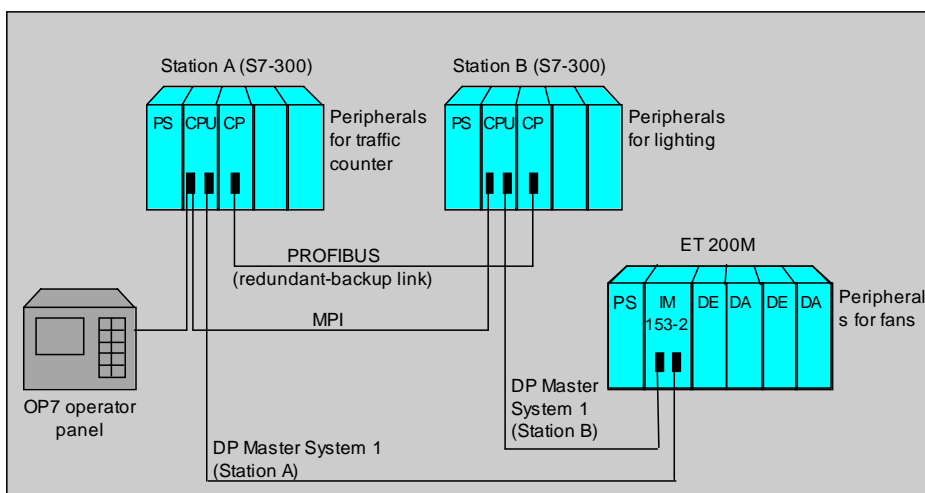


图7-6 系统配置

除了实现冗余功能的 3 条 PROFIBUS 网络外，还有一条 MPI 网络用于上位机监视和控制程序的调试。

以下是生成该例子程序的步骤，可以根据系统的配置情况进行参照：

(1) 插入两个 S7-300 的站，A 和 B，在 A 站和 B 站的硬件组态窗口中，插入 315-2DP 时，要分别创建相互独立的 PROFIBUS 网络（如 A 站为 Master_Line、B 站为 Reserve_Line），使用网络的默认参数，速率为 1.5M，并将站地址设定为 2，A 站和 B 站的 PROFIBUS DP 集成通讯口设定为主站模式（Master Mode）。

(2) 分别在 A、B 站中插入 CP342-5 模块，连接到同一个 PROFIBUS 网络上（Synchronization_Line），地址分别为 4、5，将 CP342-5 设定为 No DP 方式，并记录 CP342-5 的硬件地址 256。

(3) 分别在 A、B 站的硬件组态窗口中插入一个 ET200 从站（一个 IM153-2 模块和一个 16 入/16 出的数字量模板），DP 的地址为 3。

(4) 进入 STEP7 的网络组态窗口 NETPRO 中，选中 A 站的 CPU 点击鼠标右键，插入一个新的链接，选择 FDL Connection，点击 Apply，弹出链接属性窗口，记录链接的 ID，设定 LSAP 为 17，18，存盘编译网络组态。系统的网络结构如图 7-7 所示：

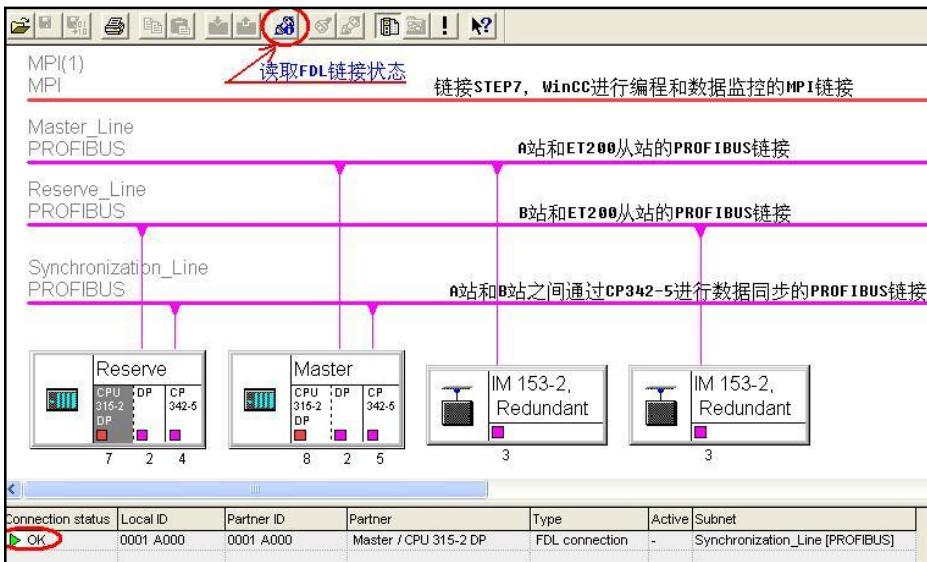


图7-7 系统的网络结构

以上网络结构中包括了 4 条链路：

- 用于上位机编程和监控的 MPI 链路；
- A 站与 ET200 从站 Master Line PROFIBUS 链路；
- B 站与 ET200 从站 Reserve Line PROFIBUS 链路；
- A 站和 B 站之间进行数据同步的 Synchronization Line PROFIBUS 链路。

可以通过点击图中的状态读取按钮，获取当前 FDL 链接的状态，上图的左下角显示当前在线读取 FDL 的状态为 OK。

(5) 在 A 站的 Block 中插入 OB1（主循环程序块）、OB35（定时中断组织块）、OB100（暖启动调用程序块）、OB80（在主系统与备用系统切换时间超时时，调用该块）、OB82 (DP-Slave ET200 站上的 IM153-2 模块出错报警，调用该功能块)、OB83（DP 从站的接口模块与主站链接断开或链接重新建立时调用该块）、OB85（程序运行出错或 DP 从站连接失败调用该块）、OB86（主从站通讯出错调用该块）、OB87（通讯失败调用该块）、OB122（外围设备访问出错调用该块）、OB121 等组织块，并对其中的 OB100、OB35、OB86 进行编程。

(6) 在 OB100 中我们调用 FC 100 ‘SWR_START’进行软冗余的初始化，有几个参数需要特别注意，LADDR 应当和 CP342-5 的硬件地址相一致，例子程序为 256，VERB_ID 必须和在 NETPRO 中创建的 FDL 链路的 ID 号一致，例子程序为 1，DB_COM_NO 建议仍采用 DB5。

FC100 功能块参数的说明和例子见表 7-6，

表 7-6 FC100 参数说明

参数名	数据类型	描述	例子
AG_KENNUNG	字符型	A站应当填‘A’ B站应当填‘B’	‘A’
DB_WORK_NO	数据块名称	软冗余功能实现所需要使用的内部数据块，用户不用生成该块，也无需读写其中的数据	DB1
DB_SEND_NO	数据块名称	存放发送数据所使用的内部数据块，A→B或B→A，用户不用生成该块，也无需读写其中的数据	DB2
DB_RCV_NO	数据块名称	存放接收数据所使用的内部数据块，A←B或B←A，用户不用生成该块，也无需读写其中的数据	DB3
MPI_ADR	整型	对方站的MPI站地址，进行MPI网络数据同步时才有意义	2
LADDR	整型	CP通讯处理器组态的硬件地址，采用PROFIBUS或Ethernet网络进行数据同步时才有意义	256
VERB_ID	整型	网络链接的ID号，在NETPRO窗口中组态的链接的ID值，如FDL Connection、ISO Connection或S7 Connection。	1

DP_MASTER_SYS_ID	整型	DP主站网络的ID号，你可以在NETPRO窗口中双击链接ET200M从站紫色的PROFIBUS主从网络，获取该ID值。	1
DB_COM_NO	数据块名称	FB101所用使用的背景数据块，需用户生成该块，并可以读取状态字DBW8和控制字DBW10，获取系统信息或手动切换主备系统	DB5
DP-KOMMUN	整型	确定链接ET200从站的DP通讯口类型： 1. 使用CPU上集成DP通讯口链接ET200从站； 2. 使用CP通讯卡上的DP通讯口链接ET200从站；	1
ADR_MODUS	整型	CPU分配I/O地址的矩阵的增量，不同CPU的地址矩阵不同： 1, if base addresses 0, 1, 2, 3 ... 4, if base addresses 0, 4, 8, 12 ...	1
PAA_FIRST	整数	ET200M站第一个输出字节的地址	0
PAA_LAST	整型	ET200M站最后一个输出字节的地址，PAA_FIRST到PAA_LAST的字节范围必须是ET200M站上连续定义的。	4
MB_NO	整型	冗余的M数据区的起始字节地址	20
MB_LEN	整型	冗余的M数据区的字节个数，如MB_NO为20，MB_LEN为30，则MB20—MB49这个数据段为冗余数据区；	30
IEC_NO	整型	冗余的IEC定时器、IEC计数器所使用的起始背景数据块	111
IEC_LEN	整型	冗余的IEC定时器、IEC计数器所使用的背景数据块的个数，如IEC_NO设为111，IEC_LEN设为7，则DB111—DB117为冗余定时器、计数器的背景数据块	7
DB_NO	整型	冗余的起始数据块	8
DB_NO_LEN	整型	冗余的数据块个数，如DB_NO为8，DB_NO_LEN为2，则DB8-DB9为冗余的数据块	2
SLAVE_NO	整型	ET200从站的最低站地址	3

SLAVE_LEN	整型	ET200从站的个数，如SLAVE_NO为3，SLAVE_LEN为2，则该软冗余系统连接了站号为3，4的两个ET200从站，站号必须连续。	
SLAVE_DISTANCE	整型	确定所链接ET200从站上IM153-2模块的PROFIBUS DP站地址： 1. 2块IM153-2模块的DP口使用相同的站地址； 2. 2块IM153-2模块的DP口的站地址为n和n+1；	1
DB_A_B_NO	数据块名称	通过数据同步链路（Redundancy Link），A、B站之间还能进行一些非冗余数据通讯，即2个PLC之间的普通数据交换，该参数即定义从A站到B站，交换的非冗余数据块。	DB11
DB_A_B_NO_LEN	WORD	A站到B站，非冗余数据块中交换的Word字数，如DB_A_B_NO设为DB11，DB_A_B_NO_LEN设为W#16#64，则A站的DB11.DBW0-DB11.DBW198被复制到B站的DB11.DBW0-DB11.DBW198，因为W#16#64代表16进制64，即一共复制100个数据字。	W#16#64
DB_B_A_NO	数据块名称	该参数即定义从B站到A站，交换的非冗余数据块。	DB12
DB_B_A_NO_LEN	WORD	B站到A站，非冗余数据块中交换的Word字数，如DB_B_A_NO设为DB12，DB_B_A_NO_LEN设为W#16#64，则B站的DB12.DBW0-DB12.DBW198被复制到A站的DB12.DBW0-DB12.DBW198，因为W#16#64代表16进制64，即一共复制100个数据字。	W#16#64
RETURN_VAL	WORD	调用FC100的返回值，为0代表正常，其他返回值您可以在错误代码表中找到对应的错误原因。	MW2
EXT_INFO	WORD	FC100中内部调用的一些功能块所返回的错误代码	MW4

(7) 一般建议将非冗余程序段编写在 OB1 当中，而将冗余程序段编写在 OB35 当中，这里使用的

```

CALL FB 101, DB5
DE WORK NO      :=DB1
CALL POSITION    :=TRUE
RETURN_VAL     :=MW6
EXT_INFO       :=MW8

U   DB5.DBX     9.1
SPB M001

Instructions for
redundant-backup
application program

M001: CALL FC 1
Instructions for
communication program

CALL FB 101, DB5
DE WORK NO      :=DB1
CALL POSITION    :=FALSE
RETURN_VAL     :=MW10
EXT_INFO       :=MW12

```

是 OB35 的默认属性，即每 100ms 中断触发一次，可以根据实际的需要在 CPU 属性中修改中断的时间间隔。在 OB35 里调用 FB 101 'SWR_ZYK' 功能块，FB101 块中封装了冗余功能的程序段，实现冗余功能。调用 FB101 时，你可以在线地读出 RETURN_VAL 参数的数值，如果为 0，说明冗余链接正常。如果为 8015 说明数据同步的连接不成功，这是一个常见的错误，原因可能是 CP342-5 之间的 FDL 链接建立的不正确或物理链路不通，或者是

FC100 的 VERB_ID 参数与 NETPRO 中的链接 ID 号不一致。当执行“SWR_START”程序块时，系统分配这些数据区，不能用 S7 的定时器和计数器，只能使用 IEC 标准的定时器和计数器。可以在软冗余手册的第三章第 9 节找到对应的诊断信息。OB35 中的程序可以分为 4 个部分，如图 7-8 所示：

A.在循环程序块（OB1 或 OB35）的开始调用 FB101，并将 CALL_POSITION 置为 TRUE

B.可以在 DB5 中得到控制字（DBW10）和状态字 DBW8 的信息。分析状态字中的信息，如果当前站为备用系统，则跳过冗余程序段。

C.冗余程序段。该处为您的冗余程序段

D.在循环程序块（OB1 或 OB35）的结尾调用 FB101，将 CALL_POSITION 置为 FALSE，停止系统冗余程序段。

图 7-8 冗余程序举例

程序执行如下：

第一步（A）启动系统的冗余数据同步功能 →

第二步（B）根据状态字判断是否为主系统，为主系统时才执行第三步，否则跳到第四步 →

第三步（C）为冗余的程序段 →

第四步（D）停止系统的冗余数据同步。

通过对 OB35 中的程序在线监控，得知当前冗余功能成功与否，如图 7-9 所示：

The screenshot displays the OB35 program code in the SIMATIC Manager. The code includes a call to FB101 with the following parameters:

```

CALL "SWR_ZYK" , DB5
DB_WORK_NO      :=DB1
CALL_POSITION:=TRUE
RETURN_VAL      :=MW6
EXT_INFO        :=MW8
    
```

Below the code, there is a variable table with two columns: IN and OUT. The values in the OUT column are 16#0, which are circled in red in the image.

IN	OUT
	16#0
	16#0

At the bottom of the screenshot, there is a ladder logic diagram showing a normally open contact labeled 'DB5.DBX9.1' leading to a coil labeled 'over' with the instruction '{ JMP }'.

图 7-9 FB101 状态值

图 7-9 显示，FB101 的返回值 Return_VAL 和 Ext_INFO 为 0，说明冗余功能正常。

通过 FB101 的背景数据块中的状态字和控制字，可以知道系统的运行的情况和当前哪个系统为主系统，哪个为备用系统，状态字的定义如图 7-10 所示：

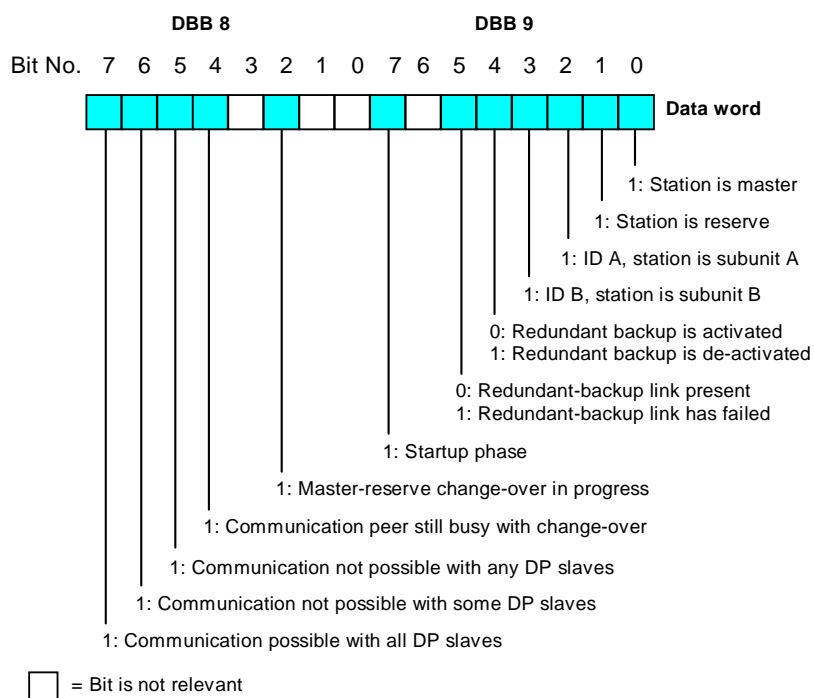


图 7-10 软件冗余状态字

可以通过写控制字中对应的位，起停备用系统与主系统之间的冗余通讯，也可以实现主系统与备用系统之间的手动切换，如图 7-11 所示：

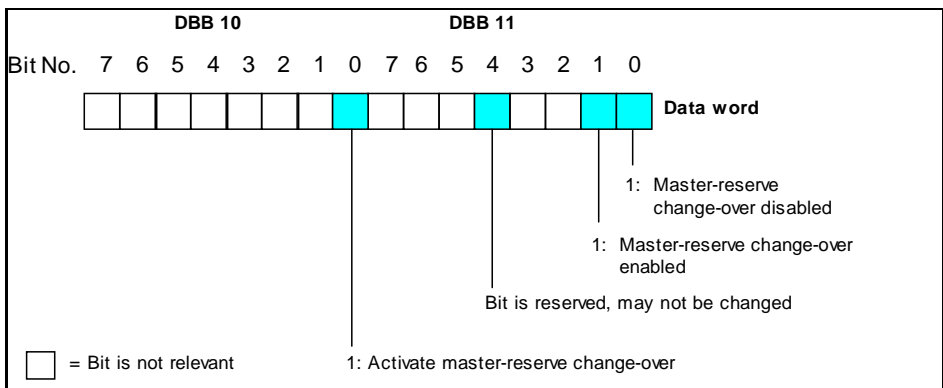


图 7-

11 软件冗余控制字

通过设定 DB5.DBX10.0 为 1，实现主系统与备用系统的手动切换。

(8)在 OB86 中调用诊断功能块 FC 102 'SWR_DIAG'，当系统出现 PROFIBUS 总线错误时，该功能块返回诊断信息，供 FB101 使用。

(9)插入 FB101 内部调用的 FB104，FC5、FC6 等功能块，将所有的程序块下载到 PLC 当中。

(10)在组态软件 WinCC 中创建两个 MPI 链接，分别与主系统和备用系统进行链接，并生成对应的 Tag 变量，如图 7-12 所示：

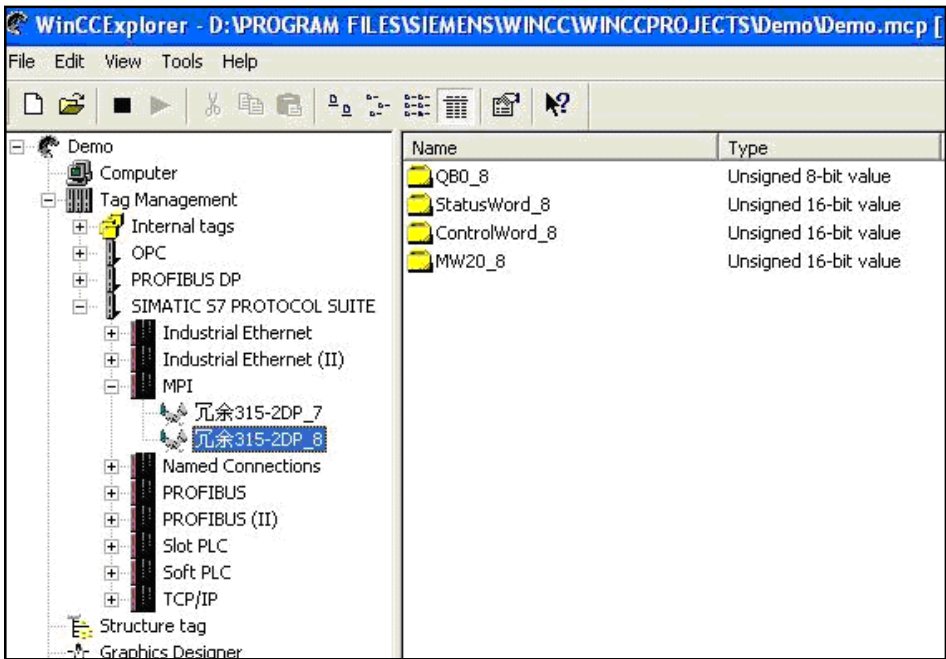


图 7-12 WINCC 中创建连接

利用 WINCC 的向导功能可以创建冗余的连接，通常的情况下，A、B 站冗余的数据连接 WINCC 建立的冗余连接，A 站数据连接 WINCC 专一与 A 站建立的连接，B 站数据连接 WINCC 专一与 B 站建立的连接。

利用 WINCC 的向导功能可以创建冗余的连接，通常的情况下，A、B 站冗余的数据连接 WINCC 建立的冗余连接，A 站数据连接 WINCC 专一与 A 站建立的连接，B 站数据连接 WINCC 专一与 B 站建立的连接。