

常问问题 • 2/2014

# STEP 7 下冗余 IO 编程

冗余 IO

## 目录

<b>1 概述</b> .....	<b>3</b>
<b>2 支持冗余 IO 的模块</b> .....	<b>3</b>
<b>3 冗余 IO 接线原理图</b> .....	<b>4</b>
<b>4 冗余 IO 的两种冗余方式</b> .....	<b>6</b>
<b>5 冗余 IO 功能块库</b> .....	<b>7</b>
<b>6 冗余 IO 编程</b> .....	<b>8</b>
6.1 创建项目 .....	8
6.2 CPU 属性设置 .....	9
6.3 IO 卡件属性设置 .....	11
6.3.1 AI 卡件属性设置 .....	11
6.3.2 AO 卡件属性设置 .....	13
6.3.3 DI 卡件属性设置 .....	14
6.3.4 DO 卡件属性设置 .....	16
6.4 冗余 IO 编程 .....	18
6.4.1 插入相应的 OB 组织块 .....	18
6.4.2 加入冗余 IO 功能块 .....	18
<b>7 常见问题解答</b> .....	<b>25</b>
<b>附表 1</b> .....	<b>27</b>
<b>附表 2</b> .....	<b>29</b>
<b>附表 3</b> .....	<b>30</b>

## 1 概述

为了提高系统可靠性，除了使用冗余 CPU，还可以使用冗余 IO 模板，容许某个信号模板或者信号通道发生故障时，不会影响输入输出信号的正常工作。要实现冗余 IO 功能，需要使用支持冗余 IO 的模块，在 STEP 7 中也需要特定的组态和编程。

冗余 IO 的概念：

当系统包含两套 IO 模块，且这些模块被组态为冗余对并作为冗余对操作时，即被视为冗余 I/O 模块。当一个 IO 模块或者通道出现故障时，系统会自动处理其冗余模块或通道的值。所以说，冗余 I/O 的使用提高了系统的冗余程度。如图 1-1。

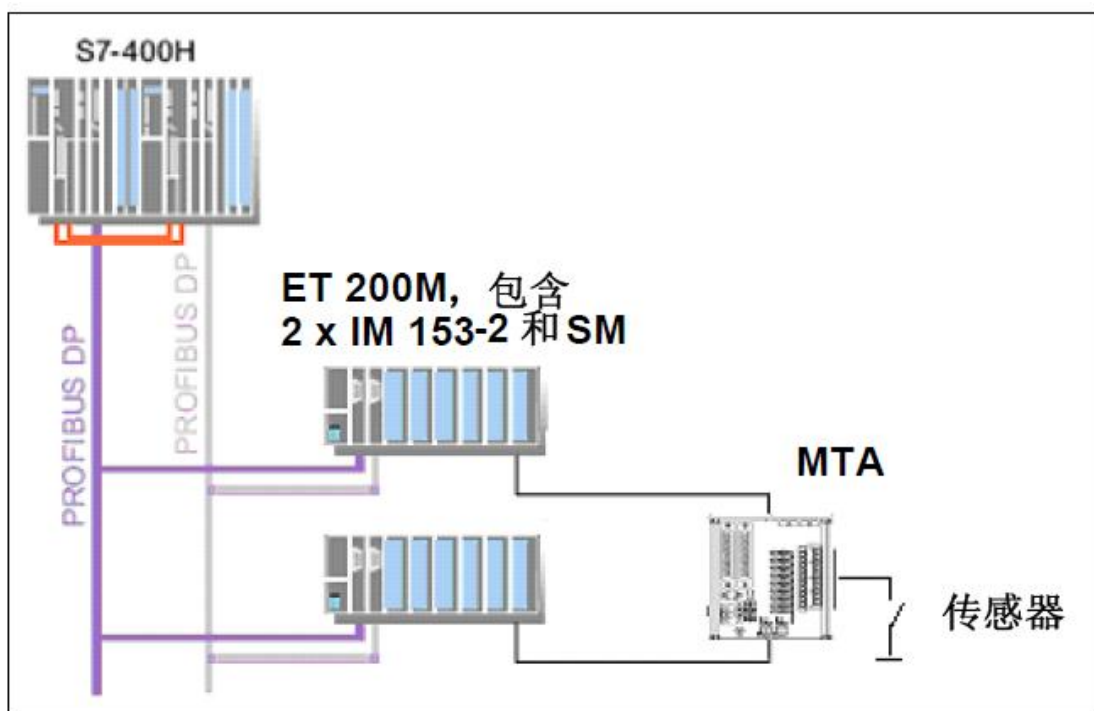


图 1-1 冗余 IO 功能的标准结构

## 2 支持冗余 IO 的模块

要实现冗余 IO 的功能，必须选择支持冗余配置的卡件（不是所有卡件均支持冗余配置），而通常情况下建议选择 ET200M 的方式来实现冗余配置，ET200M 中支持冗余配置的卡件见附表 1。

**注意：**STEP 7 中和 PCS 7 中支持的冗余 IO 模块是不同的。这里所列订货号都只适用于 STEP 7 编程。并且，随着新硬件的推出，订货号可能会发生改变。详细情况请参考最新的硬件选型样本，或咨询西门子服务热线。

### 3 冗余 IO 接线原理图

冗余 IO 卡件和现场信号或执行机构之间的连接需要借助外围冗余电路或者西门子 MTA 板。下面就模拟量输入/输出、数字量输入/输出卡件分别介绍具体的接线。

#### ➤ 模拟量输入

将一个传感器连接到一对冗余模拟量输入模板时，可以有以下几种方式：

1. 电压测量：将电压传感器并联到模拟量输入模板；
2. 间接电流测量：使用外部电阻将电流转换为电压以便并联到电压模拟量输入模板；
3. 直接电流测量：使用 BZX85 或者 1N47..A (齐纳 1.3W) 系列的二极管，将电流 (+/- 20mA, 0...20mA, 4...20mA) 串联进电流模拟量输入模板；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。

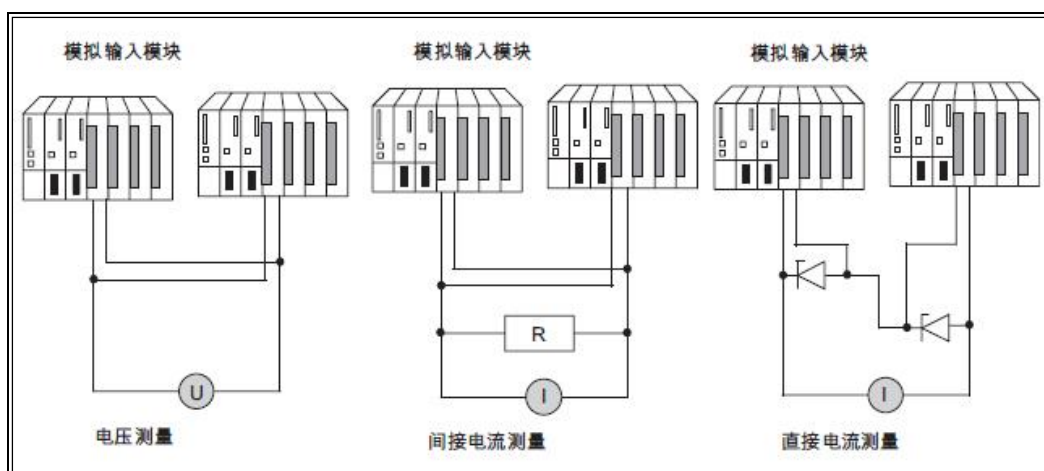


图 3-1 冗余模拟量输入接线方式

**注意：**不同订货号的模拟量输入模块支持的冗余接线方式不同。比如，**6ES7331-7KF02-0AB0** 不适用于直接电流测量。

#### ➤ 模拟量输出：

可以用并行的两个模拟量输出模板的两个输出实现对一个执行机构的冗余输出。选用 1N4003...1N4007 系列二极管比较合适，或其它技术指标满足  $V_r \geq 200V$  并且  $I_F \geq 1A$  的二极管，具体接线方式如下图所示。

冗余操作要求模拟量输出模板采用电流输出（0 -20mA, 4-20mA），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。

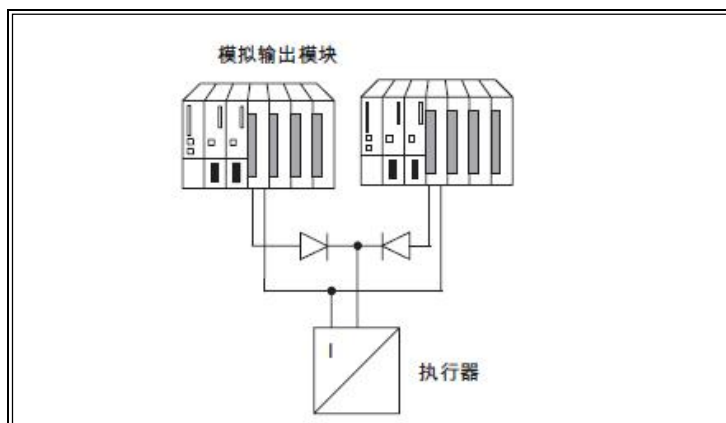


图 3-2 冗余模拟量输出接线方式

➤ 数字量输入：

现场数字信号可以参考如下图所示直接并行接入两卡件的相应通道，无需外围电路。

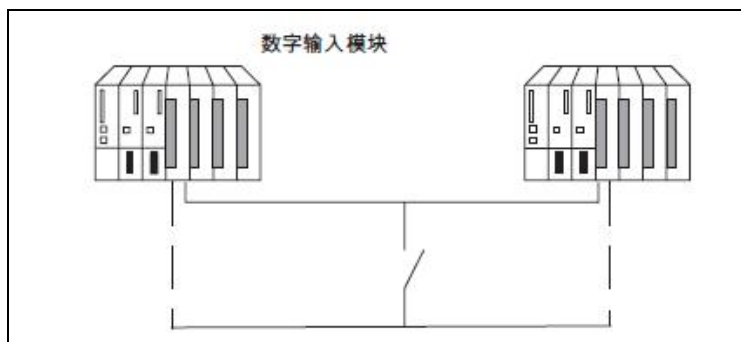


图 3-3 冗余数字量输入接线方式

➤ 数字量输出：

通过并行连接两个数字量输出模板实现执行器的冗余控制。如果需要使用外部二极管，选用 1N4003...1N4007 系列二极管比较合适，或其它技术指标满足  $V_r \geq 200V$  并且  $I_F \geq 1A$  的二极管。

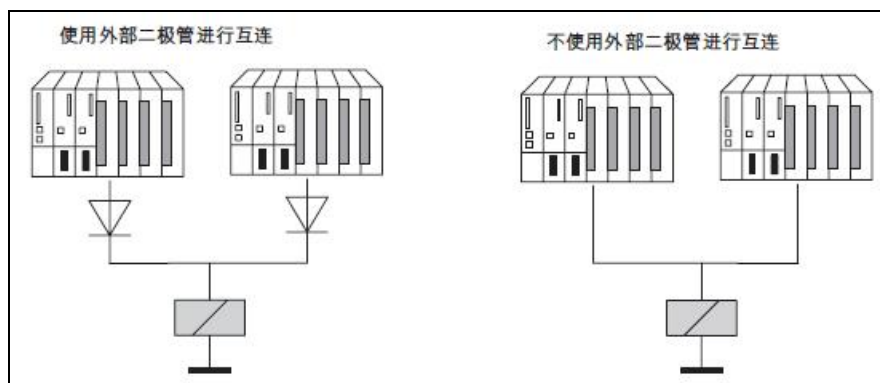


图 3-4 冗余数字量输出接线方式

注意：不是所有的DO卡件均需要加外围的二极管电路连接现场执行器，详情请参考附表2的内容。

更多关于冗余IO的接线注意事项，请参考S7-400H 手册的10.5.2章节内容；不同模板的接线方式，请参考S7-400H 手册的附录F。该手册可以访问以下链接下载：

<http://support.automation.siemens.com/cn/view/zh/1186523>

为了简便、快速、可靠地将传感器和执行器等现场设备连接到 ET 200M 远程 I/O 站的 I/O 模块（包括上述的冗余IO配置），西门子提供预置好的MTA端子板 (Marshaled Termination Assemblies 编组端接部件)，它们可显著降低布线和调试的成本与工作量，并防止出现接线错误（如图3-5）。

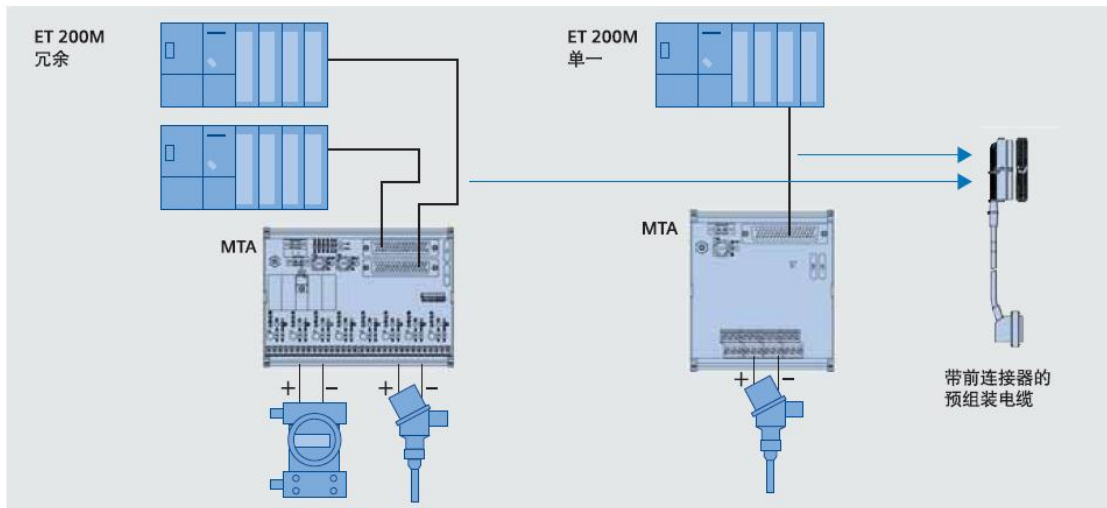


图 3-5 MTA 端子接线图例

每个MTA模块都针对ET 200M 系列中的特定I/O模块而定制（见附表3）。MTA 型号还可用于某些标准I/O 模块以及冗余和安全相关I/O 模块；MTA 通过长度为3m或8m 的带前连接器的预组装电缆与I/O模块相连。

关于MTA板的更多信息可以参考MTA 产品目录，下载链接如下：

<http://www.ad.siemens.com.cn/download/searchResult.aspx?searchText=22091986>

#### 4 冗余IO的两种冗余方式

西门子冗余IO有两种冗余方式，分别代表了西门子IO冗余的不同发展阶段：

- ✓ 模板冗余
- ✓ 通道冗余

二者的区别在于，当使用模板冗余时，模板上某一个通道故障后，整个模板将被钝化，信号读取或输出将切换到另一模块。使用通道冗余时，模板上某一个通道故障，只会钝化该通道，不会钝化该模板，不会影响该模板其他通道的信号读取和输出，其详细特性如下。

模板冗余的原则：

- ✓ 冗余总是应用于整个模块，而不是单个通道；
- ✓ 当在第一个冗余模块上发生通道错误时，整个模块及其通道都切换到钝化状态；

- ✓ 如果第一个冗余模块上发生错误，导致第一个模块钝化后，这时在第二个冗余模块上发生的通道错误，不会再导致第二个冗余模块被钝化。这就意味着第二个冗余模块的其它正常通道仍旧可以继续使用；

通道冗余IO的原则：

- ✓ 单个通道错误只会导致相关的通道钝化，而不会使整个模块钝化。也就是说不会影响模块其它通道的正常工作。

因此，通道冗余的方式提高了系统在以下情况下的实用性：

- 1、传感器经常发生故障
- 2、检修耗时长
- 3、一个模块上有几个通道出现故障

从上可以看出，通道冗余与模板冗余相比有其特有的优越性。模板冗余和通道冗余的方式代表了西门子IO冗余技术发展的不同阶段，不同版本的Step 7支持不同的IO冗余方式。详细信息请参考下述相应章节的内容。

**注意：**上述中钝化的概念为，当信号出现差异或者系统检测出模板或通道故障，系统不再读取或输出该模板或者通道的值；与钝化相对应的动作为解钝，指系统检测出故障消除后，再次重新读取或输出该模板或通道的值。

## 5 冗余IO功能块库

从上述章节中了解到西门子冗余IO技术有两种不同的冗余方式。其不同的IO冗余方式在系统中通过不同的冗余IO功能块库体现，也分为模板冗余功能块库和通道冗余功能块，不同的Step 7版本支持的冗余库也不尽相同。

关于不同STEP 7版本下可以调用的冗余IO功能库，如下表1所示：

表 5-1 不同 STEP 7 版本下的冗余 IO 功能库对照表

	模板冗余IO功能库	通道冗余IO功能库	
STEP 7 V5.4以前	Redundant IO (V1)	N/A	N/A
STEP 7 V5.4之后	Redundant IO (V1)	Redundant IO CGP	
STEP 7 V5.4 SP4	Redundant IO MGP V30	Redundant IO CGP V40	Redundant IO CGP V50

其中：

- ✓ STEP 7 V5.3 SP1只支持模板IO冗余，不支持通道IO冗余。在STEP 7 V5.3 SP1（STEP 7 V5.4以前版本）下只能调用Redundant IO (V1) 模板冗余功能库。

- ✓ 在STEP 7 V5.4之后，既可以调用Redundant IO (V1) 模板冗余功能库，也可以调用Redundant IO CGP通道冗余功能库。
- ✓ 从STEP 7 V5.4 SP4开始，出现了Redundant IO MGP V30, Redundant IO CGP V40 和Redundant IO CGP V50 三个库。Redundant IO MGP V30，类似于之前的Redundant IO (V1)，支持模板冗余；Redundant IO CGP V40，类似于之前的Redundant IO CGP，对某些特定模板支持通道冗余；Redundant IO CGP V50，对所有支持冗余IO的模板都支持通道冗余。具体见附表1。

不同冗余功能块库的函数块号完全相同，使用方法也完全相同，但是版本不同。库函数包括以下功能块：

表 5-2 冗余 IO 库功能块及其说明

	符号名	注释
FC450	RED_INIT	冗余IO初始化功能块
FB450	RED_IN	用于读取冗余输入的功能块
FB451	RED_OUT	用于控制冗余输出的功能块
FB453	RED_STATUS	用于冗余状态信息的功能块
FB452	RED_DIAG	用于诊断冗余IO的功能块
FC451	RED_DEPA	冗余IO去钝化功能块

在相应版本中使用相应的冗余库时，有如下注意事项：

- ✓ 在一个SIMATIC H 站上，只能使用一个版本的冗余IO功能库，例如，不能既调用V30库的功能块，又调用V40库的功能块；
- ✓ 不论是模板冗余IO，还是通道冗余IO，都只能以模板为单位进行冗余配置。换言之，一个模板上，或者都是冗余通道，或者都是非冗余通道；
- ✓ 如果要更换冗余IO功能块库，务必彻底删除已有的FB 450-453和FC 450-451的功能块库，插入新的功能块库，并且重新调用、生成背景DB块，并重新装载项目。

## 6 冗余IO编程

不论采用何种上述冗余IO功能块库，STEP 7下的编程方式都完全相同。这里以STEP 7 V5.4 SP4中编程为例进行详细介绍，并附录示例程序。

### 6.1 创建项目

项目中插入Simatic H Station，并组态相应硬件。此处以两个ET200M站（DP 地址分别是3和4），每个站各有8个IO模板，上下一一对应互为冗余配置，共有8个冗余IO模块对（如下图）为例进行介绍。



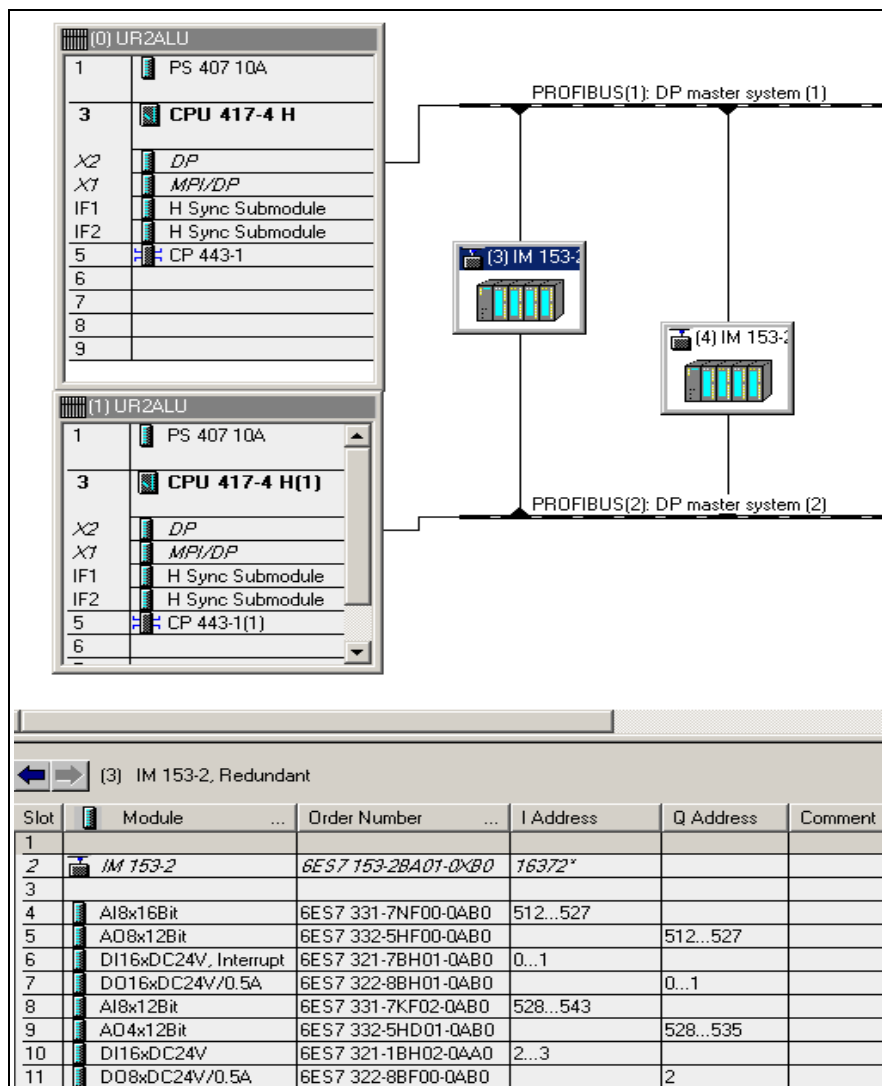


图 6-1 冗余站点硬件组态

为最大程度增加系统冗余度，并方便实际现场调试，建议：

- ✓ 建议将互为冗余的IO模块分别安装在两个ET200M站上；
- ✓ 建议将冗余模块对放在两个ET200M站的相同槽位上，便于调试管理；

## 6.2 CPU属性设置

打开硬件组态，双击CPU，打开“Properties”对话框，按照如下所示分别进行设置。

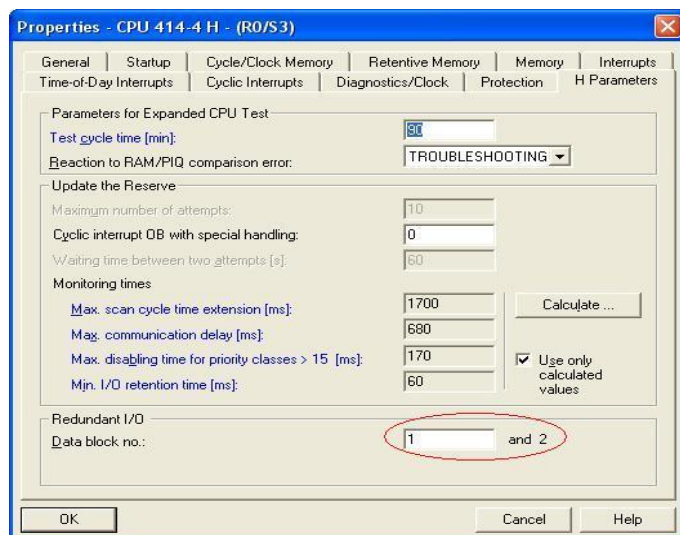


图 6-2 CPU H parameters 设置

要实现冗余I/O功能，系统需要使用两个DB块来存储冗余I/O的一些相关的数据，这两个DB块号需要在HW Config 的“CPU属性 -> H 参数”中设置。如上图所示，默认的DB块号为DB1和DB2。冗余I/O程序运行时会自动创建这两个DB块，用户不需要也不允许手动创建。**注意：用户程序中不需要也不能手动创建该两DB块，否则会引起系统内部错误，甚至造成CPU停机。**

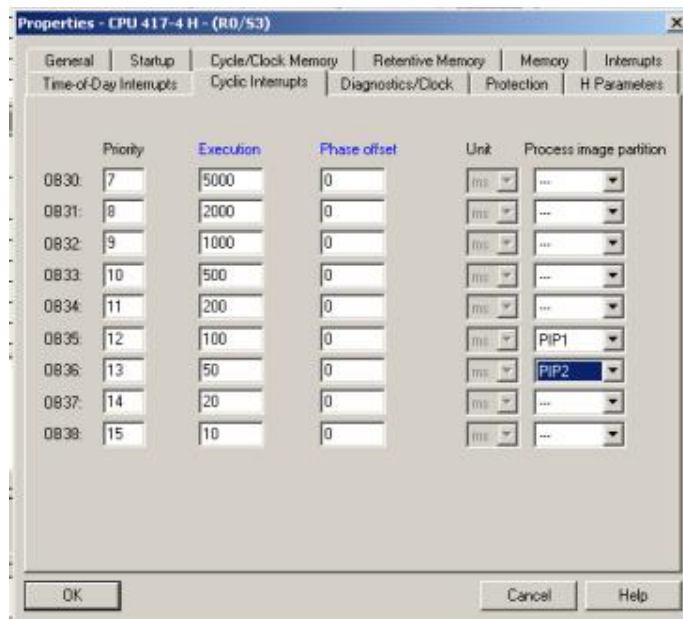


图 6-3 CPU Cyclic Interrupts 设置

参考上图分别设置OB35和OB36的Process image partition为PiP1和PiP2，循环时间分别为100ms和50ms。

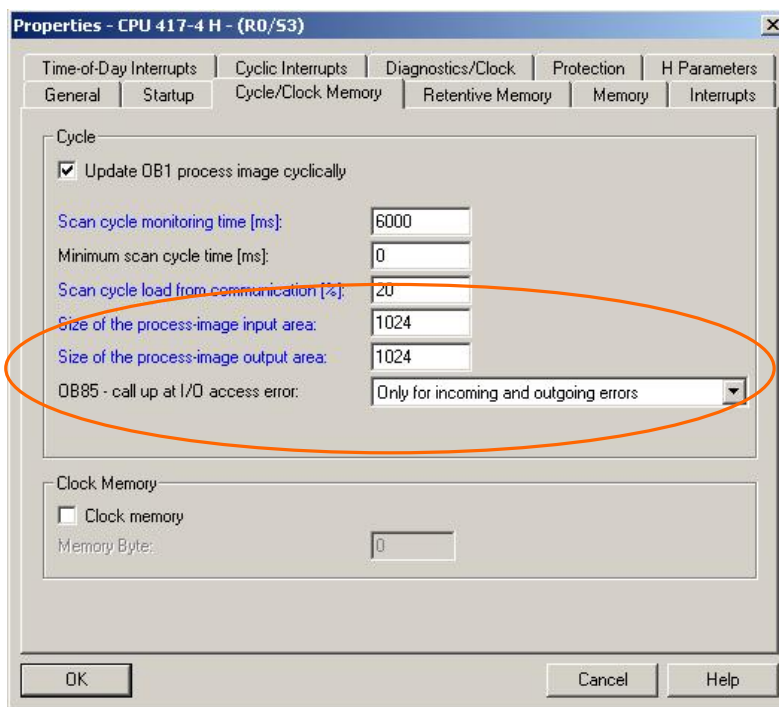


图 6-4 CPU Cycle/Clock Memory 设置

在使用冗余I/O时，应将HW-Config>CPU41x-H>Cycle/Clock Memory下

- OB85-Call Up at I/O Access Error 设定为Only for incoming and outgoing errors;
- Size of process image input/Output Area参数应完全覆盖所有IO卡件的地址（例如，所有IO卡件的地址范围为0~543，则该数值至少需要设置为544）。

### 6.3 IO卡件属性设置

#### 6.3.1 AI 卡件属性设置

双击AI卡件，打开“Properties”对话框。针对各个菜单分别设置。

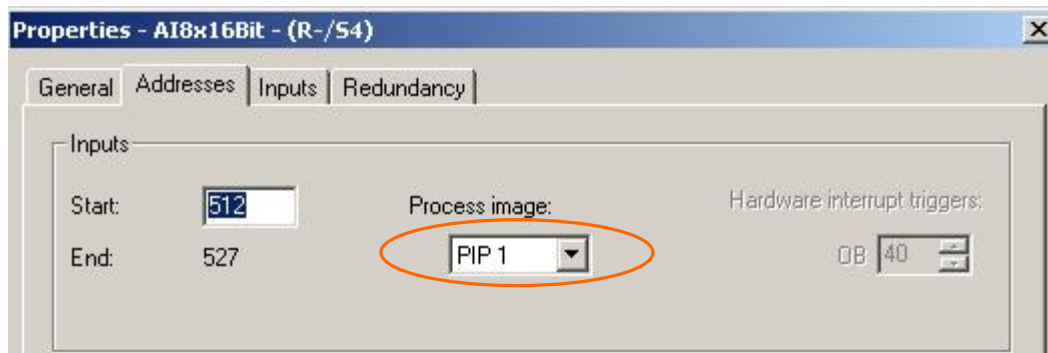


图 6-5 AI 卡件地址设置

对于带诊断功能的AI模板，选中Diagnostics Interrupt和Group Diagnostics。此例中若输入类型为1~5VDC还可设定通道的断线检测（with Check for Wire Break）。对于没有使用的通道或通道组，一定要在信号的测量类型中选择deactivated，关闭该通道组的所有诊断功能：

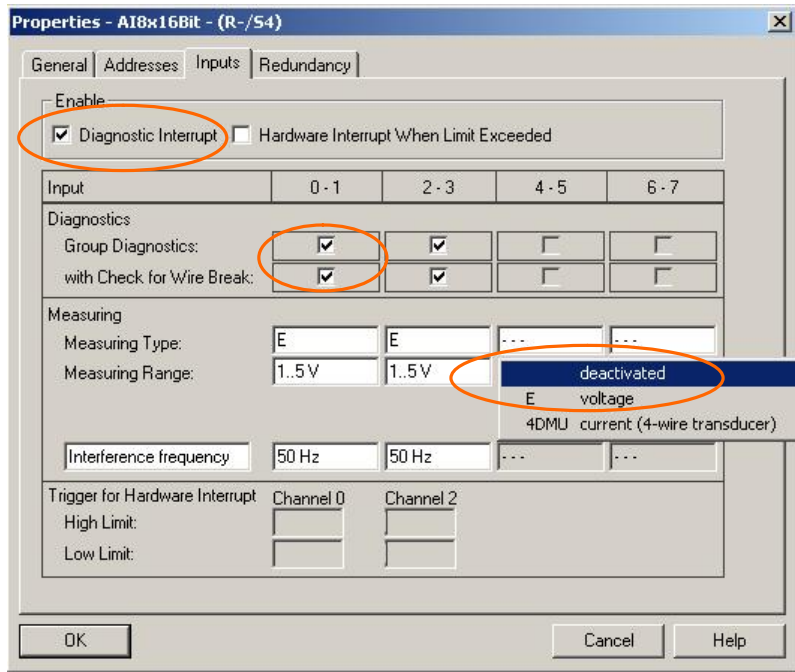


图 6-6 AI 卡件 Inputs 设置

注意：未使用的通道组必须**Deactivated**，如果某通道组中有部分通道使用部分不使用（如上图4通道使用，5通道不使用），则在激活该通道组的前提下，需要将同一通道组中已使用通道的信号通过并联（电压型信号）或串联（电流型信号）连接到未使用的通道上！

设置冗余模块对：

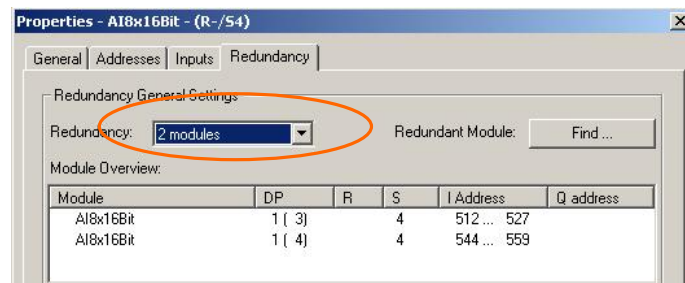


图 6-7 AI 卡件冗余对选择

应设定以下参数以组态模拟量输入模板冗余运行：

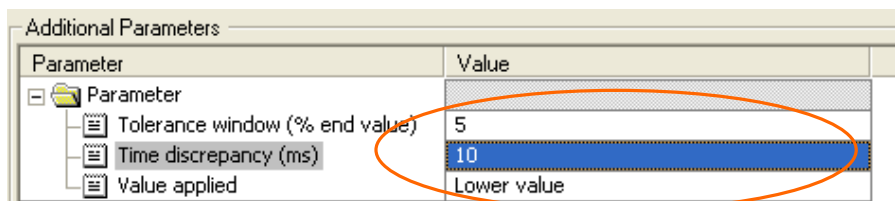


图 6-8 冗余 AI 参数设置

- **Tolerance window** 容差阈值（组态为测量范围最终数值的百分比）  
如果两个模拟值的偏差在容差阈值内则系统认为这两个模拟量采集正常；

➤ **Time discrepancy** 差异时间（冗余输入信号超出容差阈值的最大允许时间）

如果两个模拟量输入的偏差超过容差阈值，模板（或通道）不会被立刻钝化，直到组态的差异时间到后，模板（或通道）才会被钝化，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。

➤ **Value applied** 应用值

系统将检查两个模拟量输入值的偏差是否在组态的容差阈值内。如果在容差阈值内，系统会根据组态选择小值或大值做为应用值，并写入冗余通道的低地址过程映像区。如果两个模拟量输入存在差异而且是第一次差异，将被标记并且开始计算差异时间，在差异时间期间最近的有效值将被写入低地址过程映像区，并在程序中使用该应用值。当差异时间到时，应用值所在模板（通道）有效，另外一个模板被钝化。如图14所示，设定应用值为两个模拟量输入中的小值，系统将该应用值写入低地址的过程映像区，并且在程序中使用该数值，而大值的模板（或通道）被钝化。如果在差异时间内两个模拟量输入的偏差再次低于容差阈值，则差异时间停止计算并且被复位，系统将分析冗余输入信号。模板钝化时，会在CPU的诊断缓冲区中生成一条诊断信息。

### 6.3.2 AO 卡件属性设置

双击AO卡件，打开“Properties”对话框，针对各个菜单分别设置。

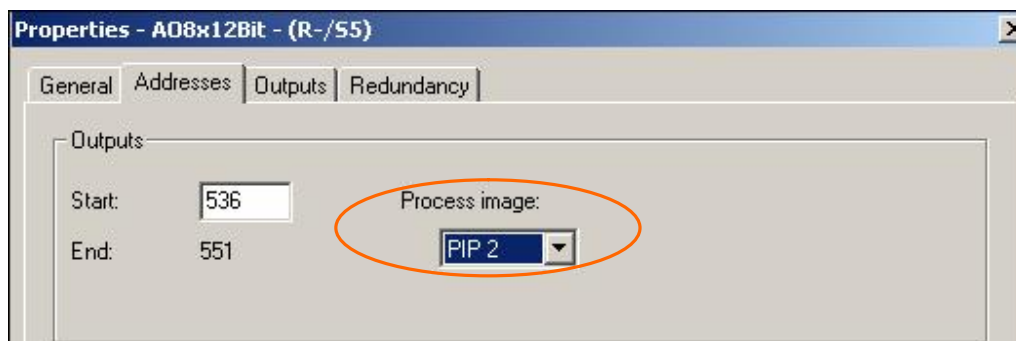


图 6-9 AO 卡件地址设置

模板属性中必须选中Diagnostics Interrupt和Group Diagnostics，并且必须组态为电流输出（0到20mA，4到20mA）（如下图），才能实现AO冗余功能。

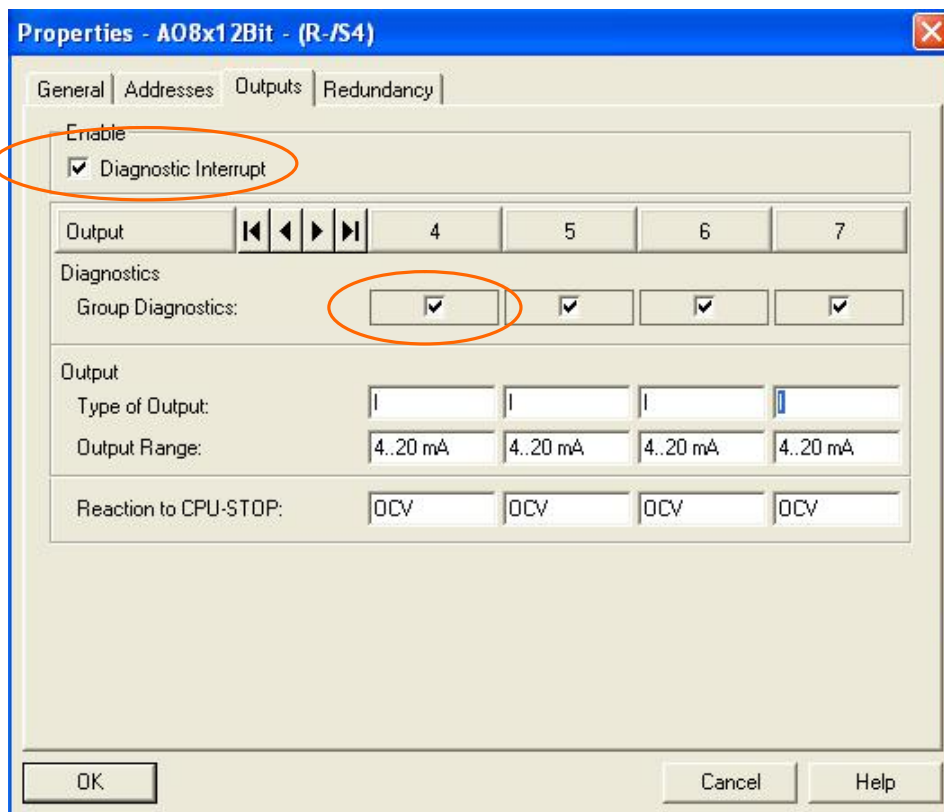


图 6-10 AO 卡件 Outputs 设置

注意：参考AI卡件的设置，没有使用的通道需要**Deactivated**，关闭该通道的诊断功能。

设置冗余模块对：

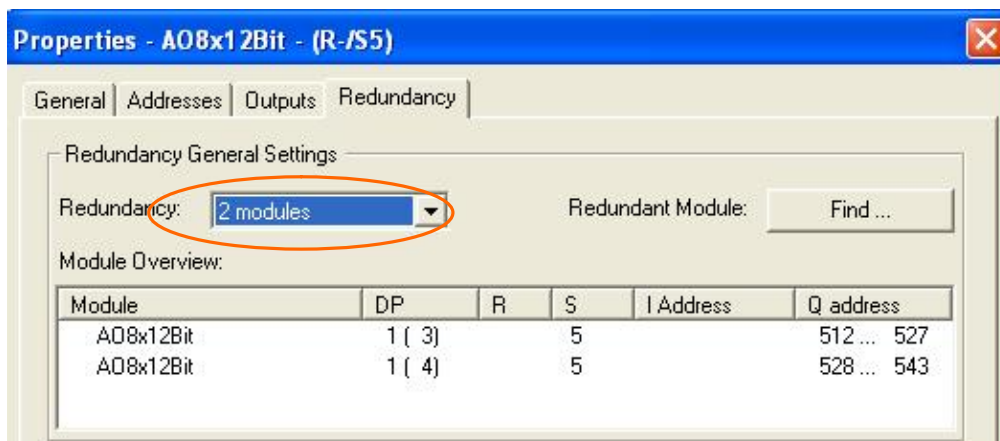


图 6-11 AO 卡件冗余对选择

### 6.3.3 DI 卡件属性设置

双击DI卡件，打开“Properties”对话框。针对各个菜单分别设置。

为卡件分配过程映像分区：



图 6-12 DI 卡件地址设置

对于带诊断功能的DI卡件，需要勾选Diagnostic interrupt:

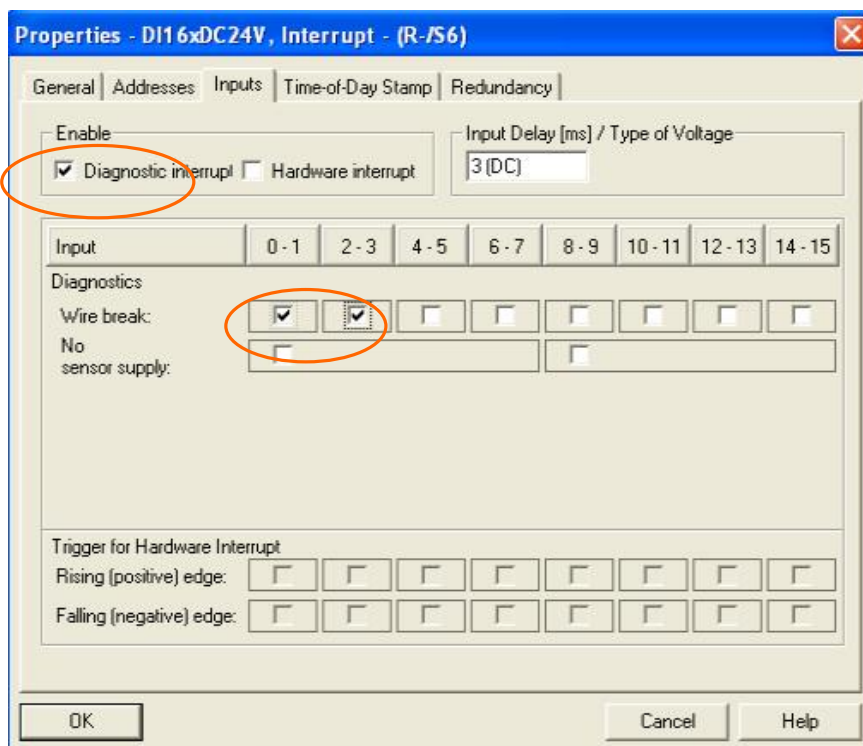


图 6-13 DI 卡件的 Inputs 设置

**注意：**上述的Wire break诊断功能需要根据实际情况激活或去激活，不使用的DI通道需要关闭其对应的所有通道诊断功能。

现场信号如果为干触点，并仍然希望使用Wire break断线诊断功能，则需要将一个分流电阻就地连接在传感器触点上：

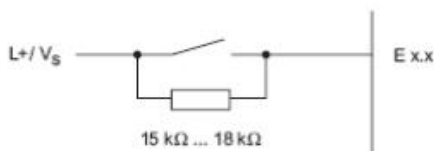


图 6-14 干触电附加断线诊断功能

具体可以参考手册“SIMATIC S7-300模块数据”，章节3.10，可在如下地址下载：  
<http://support.automation.siemens.com/CN/view/zh/8859629>。

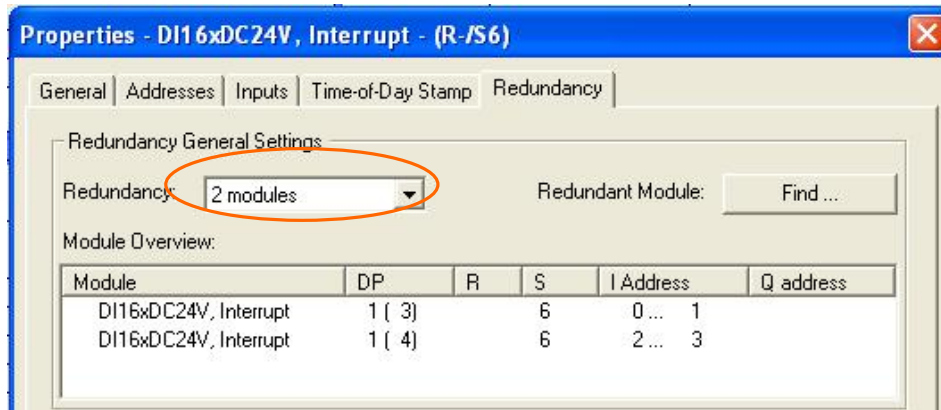


图 6-15 冗余 DI 冗余对选择

应设定以下参数以组态数字量输入模板冗余运行：

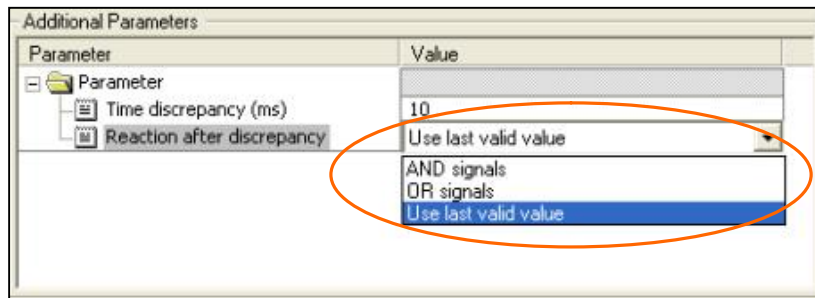


图 6-16 冗余 DI 参数设置

➤ **Time discrepancy** 差异时间：

允许冗余输入信号不同的最大时间，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。

➤ **Reaction after discrepancy** 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相等。如果数值相等，则写入该冗余通道的低地址过程映像区。如果两个输入信号存在差异并且是首次差异，将被标记并开始差异计时。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像中。在差异时间内，如果数值再次相等，则差异时间停止计算并且被复位。如果在组态的差异时间过后仍存在差异，说明已发生故障。

对于带诊断功能的DI卡件，并且勾选了诊断选项，则发生故障的模板立刻钝化；

对于不带诊断功能的DI卡件，差异时间到后模板不会被钝化；直到出现某块模板的信号变换，发生变换的模板被视为正常模板，另一个模板将被钝化。

### 6.3.4 DO 卡件属性设置

双击DO卡件，打开“Properties”对话框。为 DO卡件分配过程映像分区 (如图23)。



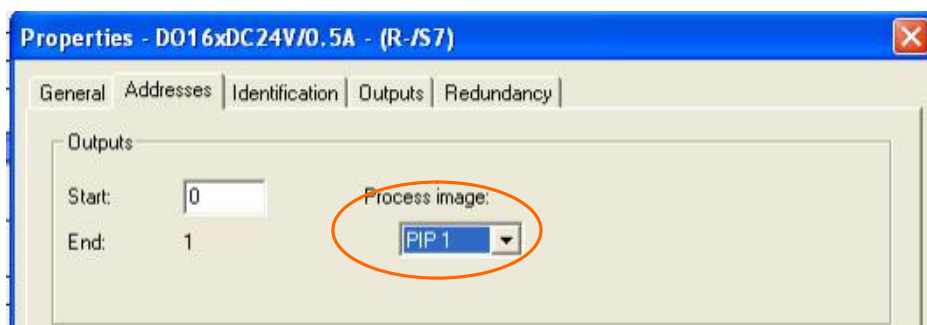


图 6-17 DO 卡件地址设置

如果带诊断功能的DO卡件，对于使用的通道，需要勾选Diagnostic interrupt和Group diagnostics:

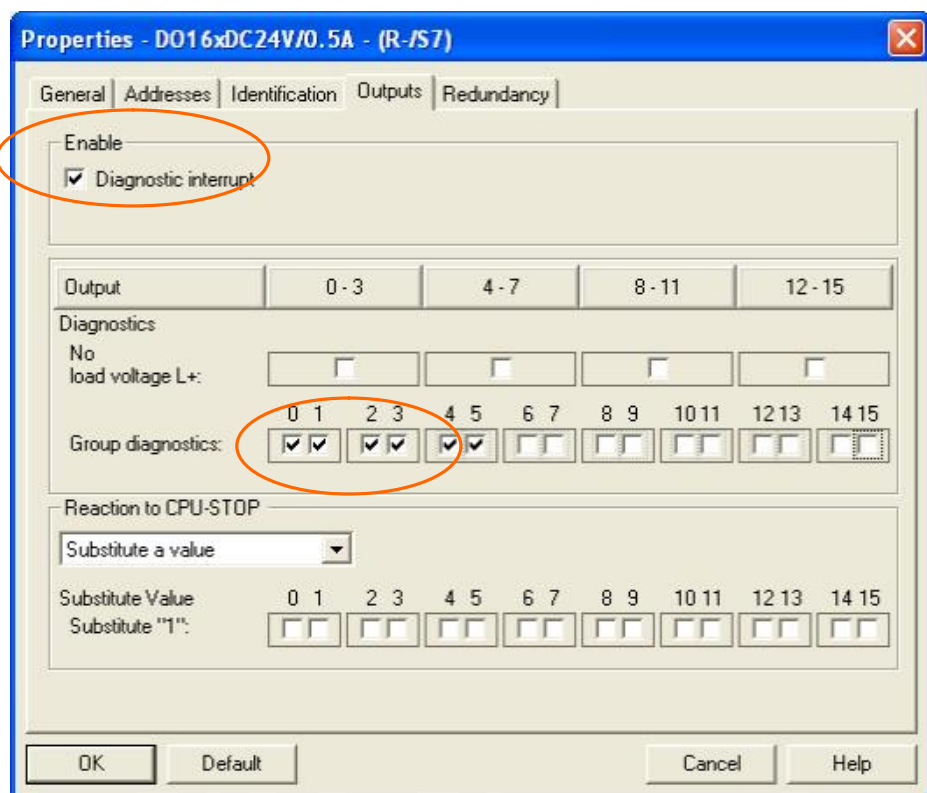


图 6-18 DO 卡件 Outputs 设置

注意：参考DI卡件设置，不使用的通道需要关闭其对应的所有诊断功能。

设置冗余模块对：

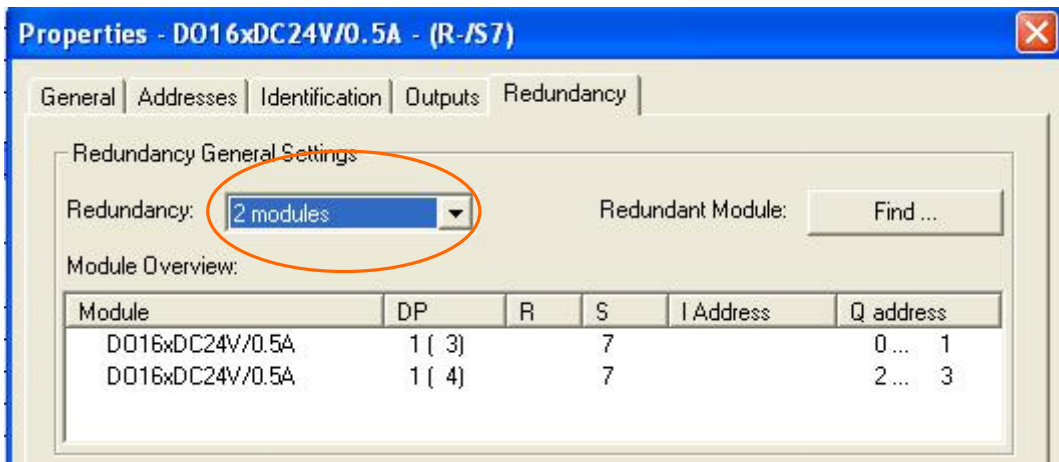


图 6-19 DO 卡件冗余对选择

## 6.4 冗余IO编程

### 6.4.1 插入相应的OB组织块

在Blocks文件夹中插入OB70, OB72, OB80, OB82, OB83, OB85, OB86, OB87, OB88, OB121, OB122组织块, 如图26所述, 否则相应错误一旦发生, 将导致CPU停机。

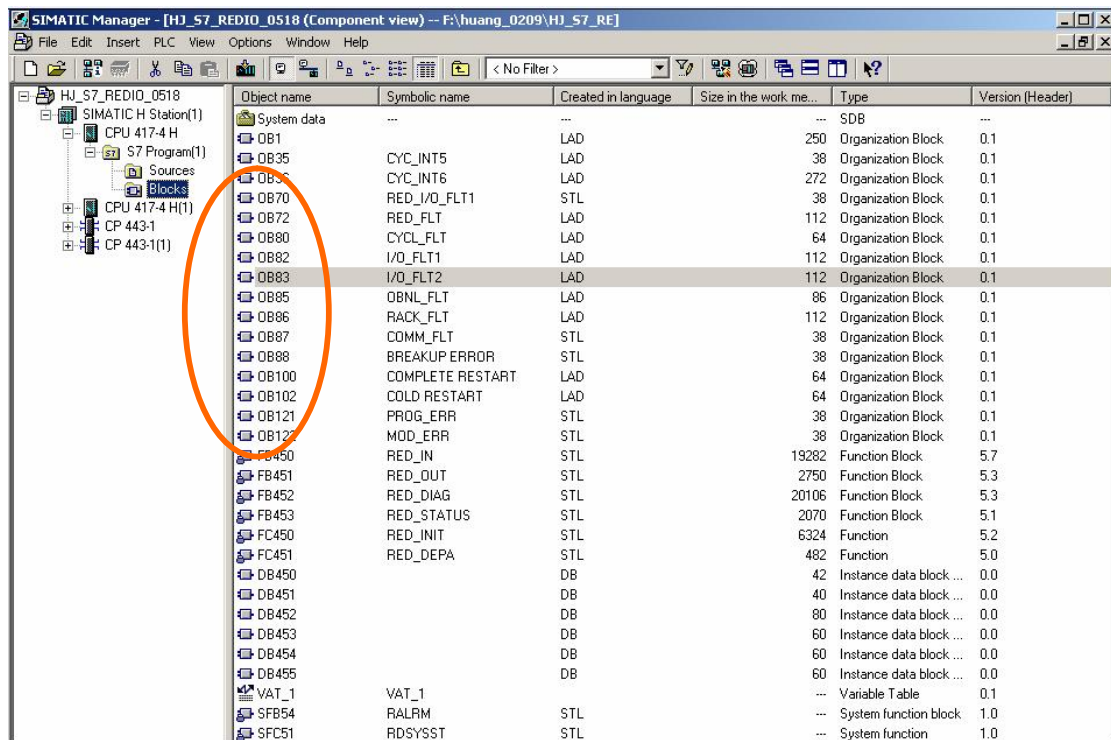


图 6-20 插入相应 OB 组织块

### 6.4.2 加入冗余IO功能块

通过Simatic Manager打开Libraries, 根据需求选择某一个冗余IO库, 并从中复制如下所示所有功能块到当前程序中, 如下图。

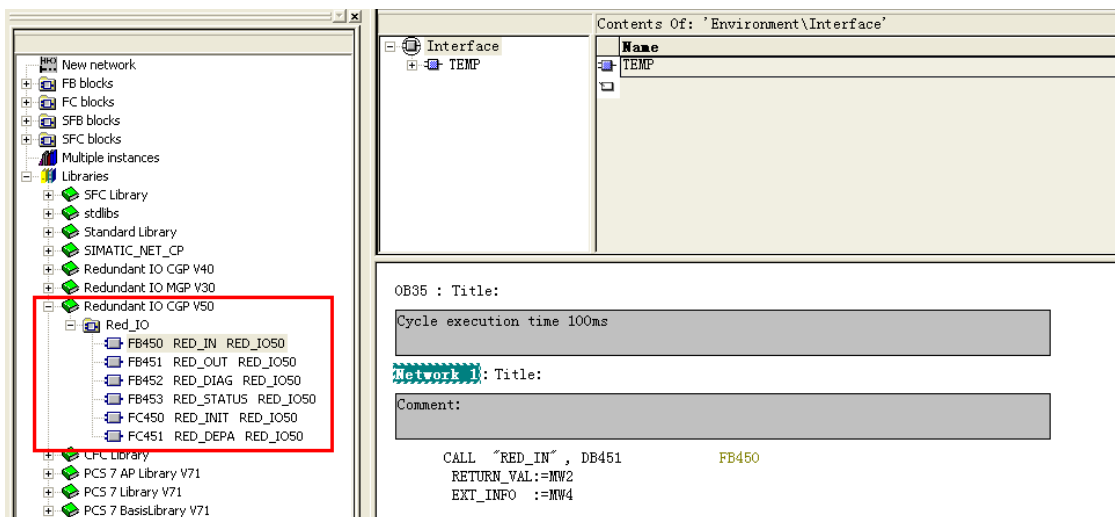


图 6-21 冗余库中需要复制的功能块

### 6.4.3 OB35程序编写

OB35中负责读取/输出AI、DI和DO信号，在OB35的起始位置调用功能块RED\_IN，在OB35的最后位置调用RED\_OUT，RED\_IN和RED\_OUT之间编写信号读取程序，例如，L IW512、A IO.0 = Q0.0等，所有冗余对信号的读取在程序中均只使用低地址，例如，IW512和IW544互为冗余，则程序中读取该冗余信号时，仅使用IW512来读取。

```

OB35 : "Cyclic Interrupt"
Comment:

Network 1: Title:
Comment:

CALL "RED_IN", DB450          FB450
RETURN_VAL:=MW2
EXT_INFO :=MW4

A    M    430.0
=    Q    0.0

L    IW   512
T    MW   1000

CALL "RED_OUT", DB451        FB451
RETURN_VAL:=MW6
EXT_INFO :=MW8

```

图 6-22 OB35 中读取冗余 AI, DI, DO 程序

### 6.4.4 OB36程序编写

此处以OB36中负责输出AO信号为例。在OB36的起始位置调用功能块RED\_IN，在OB36的最后位置调用RED\_OUT，RED\_IN和RED\_OUT之间编写信号读取程序，例如，T

QW512等，所有冗余对信号的读取在程序中均只使用低地址，例如，QW512和QW544互为冗余，则程序中读取该冗余信号时，仅使用QW512来输出。

```
OB36 : "Cyclic Interrupt"
Comment:
Network 1: Title:
Comment:
CALL "RED_IN", DB452          FB450
RETURN_VAL:=MW8
EXT_INFO :=MW10

L    MW 410
T    QW 512

CALL "RED_OUT", DB454        FB451
RETURN_VAL:=MW134
EXT_INFO :=MW136
```

图 6-23 OB36 中读取冗余 AO 程序

#### 6.4.5 OB100（暖重启）、OB102（冷重启）程序编写

OB100和OB102中调用冗余初始化块RED\_INIT。

```
OB100 : "Complete Restart"
Comment:
Network 1: Title:
Comment:
CALL "RED_INIT"              FC450
RETURN_VAL:=MW200
EXT_INFO :=MW202
```

图 6-24 OB100 中调用 RED\_INIT

```

OB102 : "Cold Restart"
Comment:
Network 1: Title:
Comment:

CALL "RED_INIT"                FC450
RETURN_VAL:=MW204
EXT_INFO :=MW206

```

图 6-25 OB102 中调用 RED\_INIT

#### 6.4.6 OB72（CPU冗余故障）程序编写

OB72中应调用冗余初始化块RED\_INIT和冗余诊断块RED\_DIAG。

```

OB72 : Title:
Comment:
Network 1: Title:
Comment:

CALL "RED_INIT"                FC450
RETURN_VAL:=MW90
EXT_INFO :=MW92

CALL "RED_DIAG" , DB452       FB452
RETURN_VAL:=MW94
EXT_INFO :=MW96

```

图 6-26 OB72 中调用 RED\_INIT 和 RED\_DIAG

#### 6.4.7 OB80（超时故障）程序编写

OB80中应调用冗余初始化块RED\_INIT。

```

OB80 : Title:
Comment:
Network 1: Title:
Comment:

CALL "RED_INIT"                FC450
RETURN_VAL:=MW208
EXT_INFO :=MW210

```

图 6-27 OB80 中调用 RED\_INIT

#### 6.4.8 OB82(诊断中断)、OB83(插拔中断)程序编写

OB82和OB83中应调用冗余诊断块RED\_DIAG和冗余去钝块RED\_DEPA。

```
OB82 : Title:
Comment:

Network 1: Title:
Comment:

CALL "RED_DIAG" , DB455      FB452
RETURN_VAL:=MW50
EXT_INFO  :=MW52

CALL "RED_DEPA"             FC451
RETURN_VAL:=MW54
EXT_INFO  :=MW56
```

图 6-28 OB82 中调用 RED\_DIAG 和 RED\_DEPA

```
OB83 : Title:
Comment:

Network 1: Title:
Comment:

CALL "RED_DIAG" , DB456      FB452
RETURN_VAL:=MW60
EXT_INFO  :=MW62

CALL "RED_DEPA"             FC451
RETURN_VAL:=MW64
EXT_INFO  :=MW66
```

图 6-29 OB83 中调用 RED\_DIAG 和 RED\_DEPA

#### 6.4.9 OB85（程序运行错误）程序编写

OB85 中应调用冗余诊断块RED\_DIAG。

```
OB85 : Title:
Comment:

Network 1: Title:
Comment:

CALL "RED_DIAG" , DB457      FB452
RETURN_VAL:=MW70
EXT_INFO  :=MW72
```

图 6-30 OB85 中调用 RED\_DIAG

#### 6.4.10 OB86 (机架故障)程序编写

OB86 中应调用冗余诊断块RED\_DIAG和冗余去钝块RED\_DEPA。

```
OB86 : "Loss Of Rack Fault"
Comment:
Network 1: Title:
Comment:
CALL "RED_DIAG", DB278      FB452
RETURN_VAL:=MW100
EXT_INFO :=MW102
CALL "RED_DEPA"           FC451
RETURN_VAL:=MW104
EXT_INFO :=MW108
```

图 6-31 OB86 中调用 RED\_DIAG 和 RED\_DEPA

#### 6.4.11 程序中冗余状态的诊断

在程序中任意位置都可以通过RED\_STATUS功能块读取某对冗余卡件的冗余状态，以便为维护等提供相应信息。

```

Network 2: Title:
Comment:

CALL "RED_STATUS", DB454      FB453
  IOID      :=B#16#54
  LADDR     :=W#16#200
  RETURN_VAL :=MW14
  EXT_INFO  :=MW16
  ACTIV_L   :=M18.0
  ACTIV_H   :=M18.1
  MODUL_STATUS_WORD:=MW20
  LADDR_L   :=MW22
  LADDR_H   :=MW24

CALL "RED_STATUS", DB455      FB453
  IOID      :=B#16#54
  LADDR     :=W#16#2
  RETURN_VAL :=MW26
  EXT_INFO  :=MW28
  ACTIV_L   :=M30.0
  ACTIV_H   :=M30.1
  MODUL_STATUS_WORD:=MW32
  LADDR_L   :=MW34
  LADDR_H   :=MW36

CALL "RED_STATUS", DB456      FB453
  IOID      :=B#16#55
  LADDR     :=W#16#0
  RETURN_VAL :=MW38
  EXT_INFO  :=MW40
  ACTIV_L   :=M42.0
  ACTIV_H   :=M42.1
  MODUL_STATUS_WORD:=MW44
  LADDR_L   :=MW46
  LADDR_H   :=MW48

```

图 6-32 RED\_STATUS 读取冗余 IO 状态

该功能块的各管脚解释如下：


- ✓ IOID: B#16#54时，表明读取模板为输入信号模板，即AI或者DI；B#16#55时，表明读取模板为输出信号模板，即AO或者DO。
- ✓ LADDR: 为所读冗余模板对的低地址模板的基地址。比如，对于一对冗余AI模板，低地址为512...527，高地址为544...559，LADDR=W#16#200。

更多关于每个管脚的含义，可选中此功能块，按F1，查看帮助文件。



## 7 常见问题解答

Q1: 当用8通道AO卡件时，AO信号为什么前四个通道正常，后四个通道不正常？

A1: AO卡件信号类型的设置是针对单个通道的。在Output中要用  翻到下一页，设置后四个通道的信号类型和诊断选项。

Q2: 检查程序时，如何判断程序中所用功能块出自哪一个冗余IO功能库？

A2: 在SIMATIC MANAGER中，Blocks的folder中，查看功能块的Author，以及Version(Header)是否一致，如图39。下表中列出了不同版本功能块的Author（表3）和Version（表4）。一个控制器下必须保证所有的上述冗余功能块均来自于同一个库。

Object name	Symbolic name	Version (Header)	Name (Header)	Author
FB450	RED_IN	3.5	RED_IN	RED_IO
FB451	RED_OUT	3.2	RED_OUT	RED_IO
FB452	RED_DIAG	3.1	RED_DIAG	RED_IO
FB453	RED_STATUS	3.1	RED_STAT	RED_IO
FC450	RED_INIT	3.2	RED_INIT	RED_IO
FC451	RED_DEPA	3.0	RED_DEPA	RED_IO

图 7-1 判断功能块是否属于同一个库

表 7-1

冗余IO功能块库	Author
Redundant IO(V1)	RED_IO
Redundant IO CGP	RED_IO_1
Redundant IO MGP V30	RED_IO
Redundant IO CGP V40	RED_IO_1
Redundant IO CGP V50	RED_IO50

表 7-2

冗余IO功能块	Redundant IO(V1)	Redundant IO CGP	Redundant IO MGP V30	Redundant IO CGP V40	Redundant IO CGP V50
FB450 RED_IN	3.2	4.6	3.5	4.6	5.7
FB451 RED_OUT	3.2	4.3	3.2	4.3	5.3
FB452 RED_DIAG	3.1	4.6	3.1	4.6	5.3
FB453 RED_STATUS	3.1	4.2	3.1	4.2	5.1
FC450 RED_INIT	3.1	4.1	3.2	4.1	5.2
FC451 RED_DEPA	3.0	4.1	3.0	4.1	5.0

Q3: 为什么CPU 412-3H冗余控制器下进行冗余IO编程后，Master CPU运行起来之后，Standby CPU无法同步，并且同步过程中Master CPU也进入STOP状态，CPU报INTF故障，在单CPU运行情况下冗余IO无法正常工作？

A3: 检查CPU的诊断缓冲区，如提示Error when allocating local data错误，则为CPU的本地数据设置不正确导致冗余功能块无法正常执行。在CPU硬件组态中，Memory菜单中，适当增加提示出错优先级的local data。比如，增加优先级25和28的Local Data。如图40。更多详细信息可以参考文档“如何计算S7400 CPU所需的Local Data大小”。

<http://www.ad.siemens.com.cn/download/searchResult.aspx?searchText=F0302>。

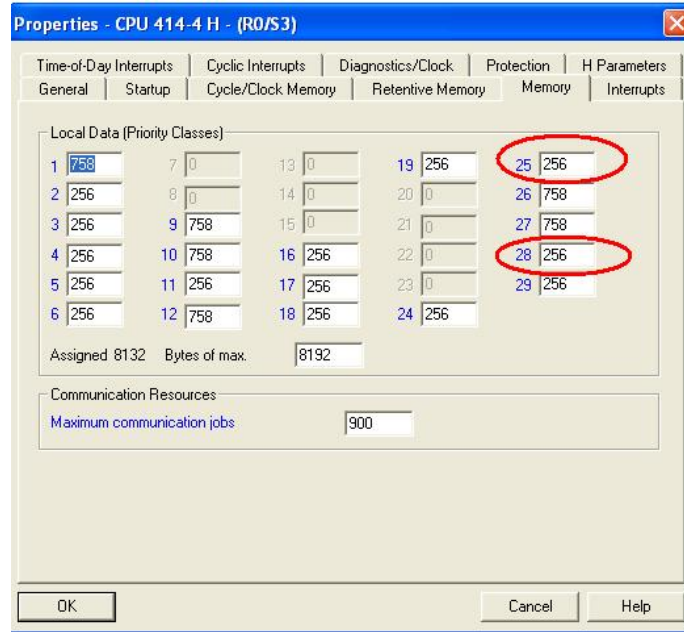


图 7-2 调整 CPU 的 Local Data

## 附表1

### 支持冗余IO的模块列表

分布式双通道冗余DI:

模块	订货号	V30	V40	V50
DI16xDC 24 V, 中断	6ES7321-7BH00-0AB0	X		X
DI16xDC 24 V	6ES7321-7BH01-0AB0	X	X	X
DI16xDC 24 V	6ES7321-1BH02-0AA0	X		X
DI32xDC 24 V	6ES7321-1BL00-0AA0	X		X
DI 8xAC 120/230V	6ES7321-1FF01-0AA0	X		X
DI 4xNamur [EEx ib]	6ES7321-7RD00-0AB0	X		X
DI 16xNamur	6ES7321-7TH00-0AB0	X		X
DI 24xDC 24 V	6ES7326-1BK00-0AB0	X		X
DI 8xNAMUR [EEx ib]	6ES7326-1RF00-0AB0	X		X

分布式双通道冗余DO:

模块	订货号	V30	V40	V50
DO8xDC 24 V/0.5 A	6ES7322-8BF00-0AB0	X		X
DO8xDC 24 V/2 A	6ES7322-1BF01-0AA0	X		X
DO32xDC 24 V/0.5 A	6ES7322-1BL00-0AA0	X		X
DO8xAC 120/230 V/2 A	6ES7322-1FF01-0AA0	X		X
DO 4x24 V/10 mA [EEx ib]	6ES7322-5SD00-0AB0	X		X
DO 4x24 V/10 mA [EEx ib]	6ES7322-5RD00-0AB0	X		X
DO 16xDC 24 V/0.5 A	6ES7322-8BH01-0AB0	X	X	X
DO 10xDC 24 V/2 A, 产品版本V3 以上	6ES7326-2BF01-0AB0	X	X	X

分布式双通道冗余AI:

模块	订货号	V30	V40	V50
AI8x12 位	6ES7331.7KF02.0AB0	X		X
AI 8x16 位	6ES7331-7NF00-0AB0 (从STEP 7 V5.4SP4开始)	X	X	X
AI 8x16 位	6ES7331-7NF10-0AB0	X		X
AI 6xTC 16 位 iso	6ES7331-7PE10-0AB0			X

AI 4x15 位 [EEx ib]	6ES7331-7RD00-0AB0	X		X
AI 6x13 位	6ES7336-1HE00-0AB0	X		X
AI 8x0/4...20mA HART	6ES7331-7TF01-0AB0	X	X	X

分布式双通道冗余AO:

模块	订货号	V30	V40	V50
AO4x12 位	6ES7332-5HD01-0AB0	X		X
AO8x12 位	6ES7332-5HF00-0AB0	X	X	X
AO4x0/4...20 mA [EEx ib]	6ES7332-5RD00-0AB0 *	X		X
AO 8x0/4...20mA HART	6ES7332-8TF01-0AB0	X	X	X
* 在冗余模式下，不能将该模块应用在危险区域。				

附表2

冗余DO卡件是否需要外接二极管对照表

DO模块	需要外接二极管	不需要外接二极管
6ES7326-2BF01-0AB0	X	X
6ES7322-1BL00-0AA0	X	-
6ES7322-1BF01-0AA0	X	-
6ES7322-8BF00-0AB0	X	X
6ES7322-1FF01-0AA0	-	X
6ES7322-8BH01-0AB0	-	X
6ES7322-5RD00-0AB0	X	-
6ES7322-5SD00-0AB0	X	-

X: yes

## 附表3

MTA端子板对照表

MTA类型	MTA的订货号	ET200M模块订货号	IO冗余
8 channels, AI	6ES7 650-1AA51-2XX0	6ES7 331-7NF00-0AB0 (从产品版本 5 开始)	Yes
8 channels, AO	6ES7 650-1AB51-2XX0	6ES7 332-5HF00-0AB0 (从产品版本 3 开始)	Yes
8 channels, AI HART	6ES7 650-1AA61-2XX0	6ES7 331-7TF01-0AB0	Yes
8 channels, AO HART	6ES7 650-1AB61-2XX0	6ES7 332-8TF01-0AB0	Yes
8 channels, AI TC	6ES7 650-1AF51-2XX0	6ES7 331-7PF10-0AB0 (从产品版本 4 开始) 以及 6ES7 331-7PF11-0AB0	No
8 channels, AI RTD	6ES7 650-1AG51-2XX0	6ES7 331-7PF00-0AB0 (从产品版本 8开始) 以及 6ES7 331-7PF01-0AB0	No
16 channels, DO	6ES7 650-1AD10-2XX0	6ES7 322-8BH01-0AB0	Yes
6 channels F-AI HART (safetyrelated)	6ES7 650-1AH61-5XX0 and 2 x 6ES7 650- 1BD51-0XX0	6ES7 336-1HE00-0AB0 (从产品版本 6开始)	Yes
	6ES7 650-1AH61-5XX0	6ES7 336-4GE00-0AB0	Yes
16 channels, DI	6ES7 650-1AC11-3XX0	6ES7 321-7BH01-0AB0 (从产品版本 2开始)	Yes
24 channels F-DI (safety-related)	6ES7 650-1AK11-7XX0	6ES7 326-1BK00-0AB0 以及 6ES7 326-1BK01-0AB0 (从产品版本 1开始)	Yes
10 channels F-DO (safety-related)	6ES7 650-1AL11-6XX0	6ES7 326-2BF01-0AB0 (从产品版本 2开始)	Yes
16 channels DO relay	6ES7 650-1AM30- 3XX0	6ES7 322-8BH01-0AB0 (从产品版本 1开始)	Yes

10 channels F DO relays (safetyrelated)	6ES7 650-1AM31- 6XX0	6ES7 326-2BF01-0AB0 (从产品版本 2开始)	Yes
--	-------------------------	------------------------------------	-----