

操作指南 • 3/2019

# PCS 7 中 CMT 使用入门指南

PCS7, CFC 编程, 控制模块类型, 控制模块, 工厂层级, 过程标签类型

<http://support.automation.siemens.com/CN/view/zh/109766191>

# 目录

<b>1</b>	<b>概述.....</b>	<b>3</b>
<b>2</b>	<b>使用 CMT 进行编程 ( 应用示例 ) .....</b>	<b>4</b>
2.1	软件环境说明 .....	4
2.2	创建控制模块类型 .....	4
2.2.1	从 CFC 创建 CMT .....	4
2.2.2	从过程标签类型创建 CMT .....	6
2.2.3	直接在主数据库创建 CMT .....	8
2.3	变体和替代选择 .....	11
2.3.1	变体 ( Variant ) -可选对象.....	11
2.3.2	替代对象 ( Alternative objects ) .....	13
2.4	CMT 同步 CM ( 实例 ) .....	15
2.4.1	在 CMT 中新增对象 ( FB ) .....	15
2.4.2	更改属性.....	19
2.4.3	更改参数值 .....	22
2.5	功能 ( 实例特殊扩展 ) .....	23
2.6	状态和命令 .....	26
2.7	连接类型.....	29
2.7.1	互连到 CM 的参数.....	29
2.7.2	引用 CM 参数.....	30
2.7.3	引用块变量 .....	31
2.7.4	引用全局变量.....	32

# 1 概述

PCS7 V8.0 开始提供了一种新的概念类型-控制模块类型，在 PCS7 的 SIMATIC Manager 中进行 CFC 编程时，通过使用预先定义好的控制模块类型达到快速灵活组态的目的。

控制模块和控制模块类型具有如下优势：

- 可以通过同步功能实现控制模块类型到控制模块实例的改变更新，同时控制模块实例中的特殊修改，在同步过程中不会丢失，保持控制模块实例的特性。
- 在控制模块类型中可以配置可选块，因此可以通过一个控制模块类型去创建不同结构的控制模块实例
- 控制模块类型和控制模块的使用使得能够和 Advanced ES、Plant Automation Accelerator 以及 COMOS 之间进行扩展数据交换。

在文档中会涉及到诸如控制模块类型、过程标签类型等的概念，为了便于描述有时会使用如下的缩略语进行表示：

序号	缩略语	描述
1	CMT	Control Module Type-控制模块类型
2	CM	Control Module-控制模块
3	PTT	Process Tag Type-过程标签类型
4	AdvES	PCS7 Advanced Engineering-PCS7 高级工程师站
5	COMOS	一款一体化软件解决方案，具有统一的数据平台，可在所有项目阶段为工厂设计人员、工厂运营人员、企业管理人员以及合作伙伴提供符合其特定需求的连续数据流。
6	PAA	Plant Automation Accelerator

表 1-1 缩略语

## 2 使用 CMT 进行编程（应用示例）

### 2.1 软件环境说明

操作系统：Windows 10 Enterprise 2015 LTSB

软件版本：PCS7 V9.0 SP1


### 2.2 创建控制模块类型

CMT 可以通过以下三种方式创建：

- 从 CFC 创建
- 从过程标签类型创建
- 直接从主数据库创建 CMT

注意：CMTs 必须放在主数据库中

#### 2.2.1 从 CFC 创建 CMT

- 1、在菜单栏选择“视图→工厂视图”（View→Plant View），将项目切换到工厂视图。拷贝现有 CFC 到主数据库中某个层级（如果没有做好的 CFC，也可以直接在库的工厂层级中插入新的 CFC）。
- 2、打开该 CFC，在菜单栏选择“视图→工艺 I/O”（View→Technological I/O）或工具栏中的  按钮，激活工艺 I/O 的窗口。

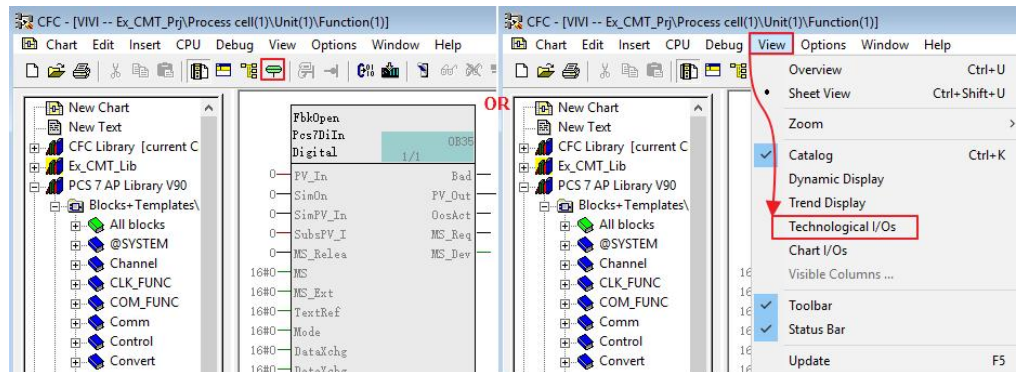


图 2-1 激活工艺 I/Os 窗口

- 3、在工艺 I/O 窗口的左侧空白处单击右键在弹出的快捷菜单中选择“插入新对象 → 控制模块”（Insert new object → Control module）。

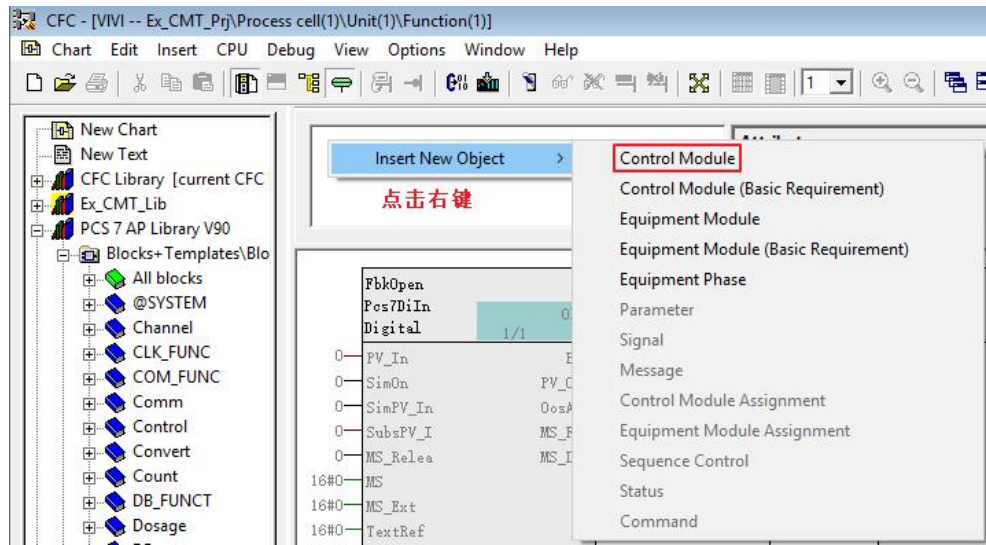


图 2-2 插入控制模块

4、最重要的一步首先分配 CMT 的名称。在工艺 I/O 窗口右侧的名字/分配点击右键选择增加图表。

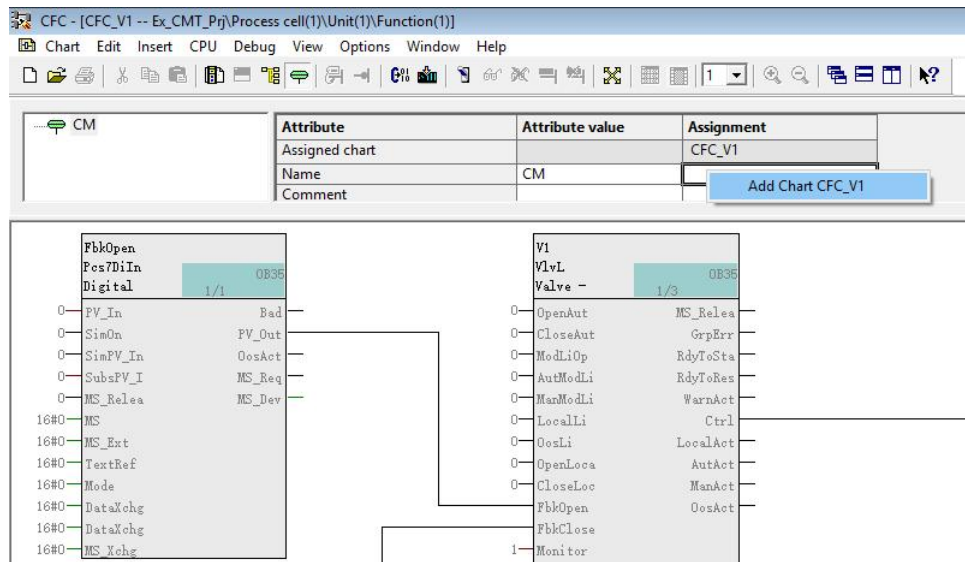
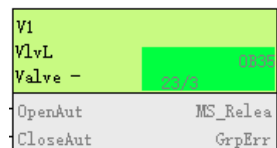


图 2-3 分配 CMT 的名称

**注意：**如果忽略第四步，则在 AdvES, PAA 或 Comos 中给 CM 分配的名字只能传递到工艺 I/O 而不是图表级。

**注意：**与工艺 IO 有连接的对象在 CFC 中以绿色显示。CMT 中的所有 FBs 会自动连接到工艺 IO（因为“ Sampling time”）



## 2.2.2 从过程标签类型创建 CMT

- 1、在主数据库中选择一个要在其中创建 CMT 的层级文件夹，右键选择：“工厂类型（Plant Types）→通过过程标签创建控制模块类型...（Create Control Module Type from Process Tag Type...）”。

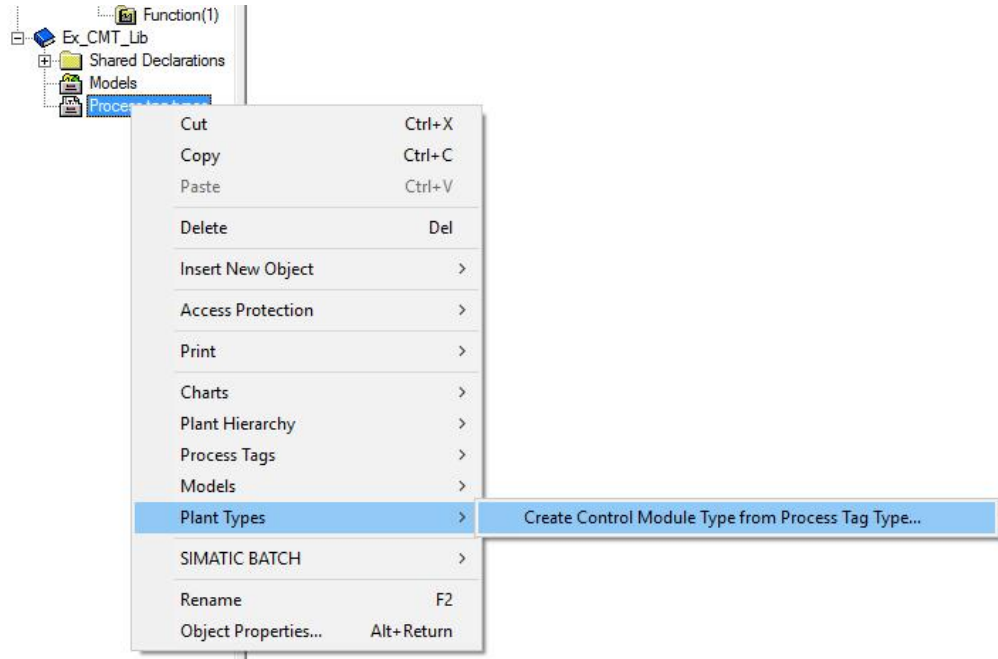


图 2-4 从过程标签类型创建 CMT

- 2、此时会弹出一个对话框显示主数据库中所有可用的过程标签类型。

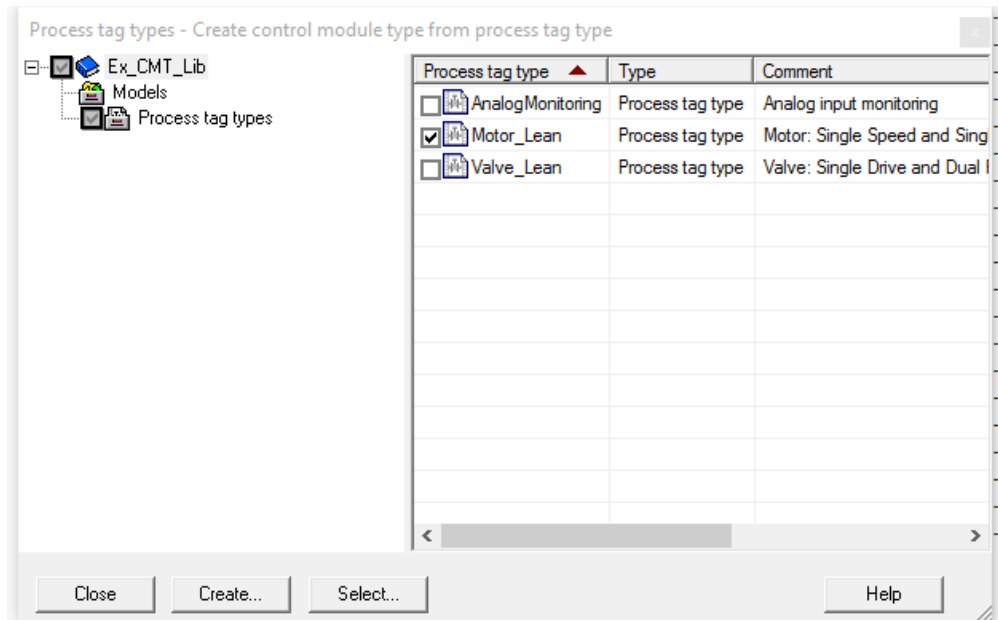


图 2-5 选择可用的过程标签类型

- 如果对话框中显示有 PTTs→请跳到第五步继续
- 如果对话框中显示没有 PTTs→请从第三步继续

- 3、点击“选择...”（Select...）按钮，弹出“打开库”（Open Library）的对话框，选中包含用于创建 CMT 的 PTTs 的库，点击“确定”（OK）进行确认。

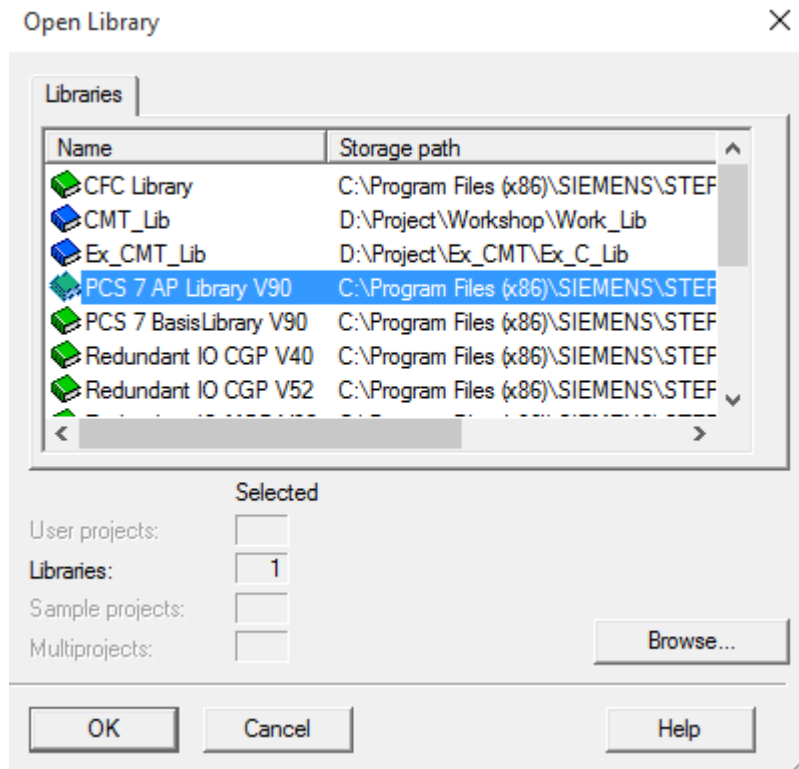


图 2-6 选择用于创建 CMT 的 PTTs 的库

- 4、此时会弹出一个包含所选库中所有过程标签类型的对话框，通过点击复选框选择所需要的过程标签类型。

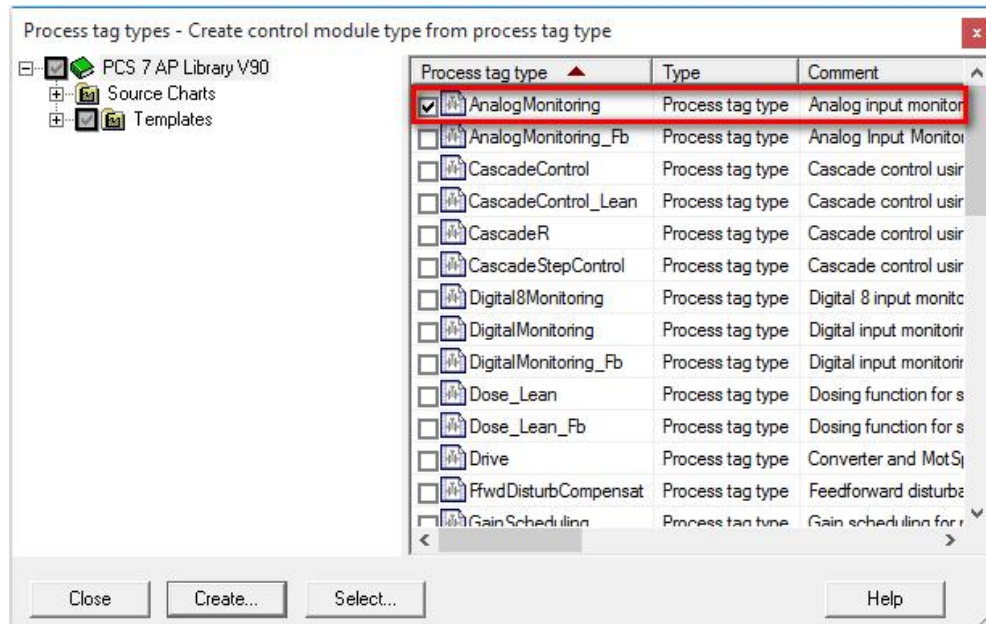


图 2-7 选择需要的过程标签类型用于创建 CMT

- 5、点击“创建”（Create）按钮，从所选的 PTTs 生成相关的 CMTs。

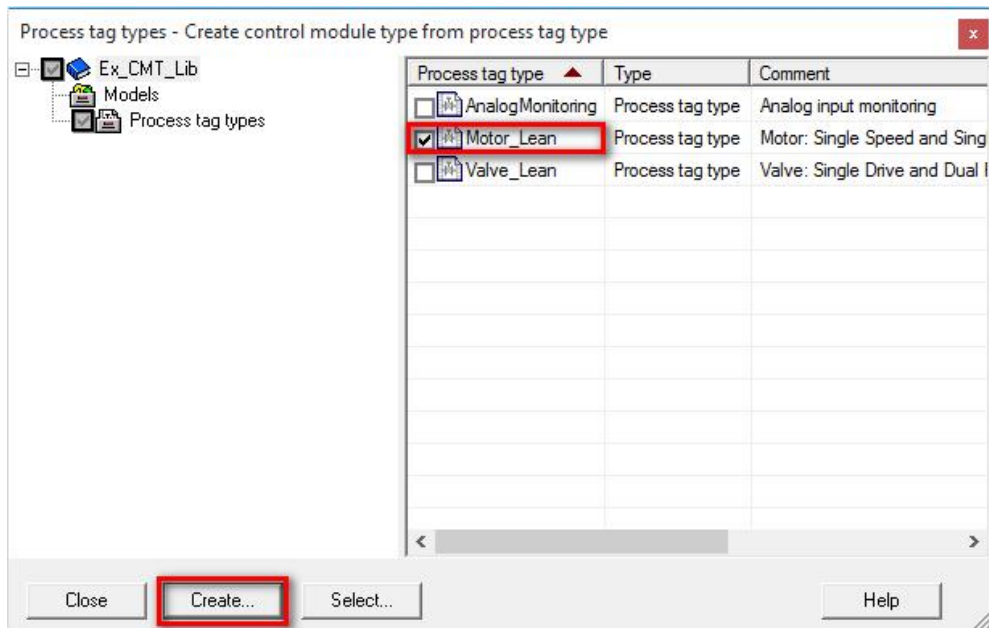


图 2-8 根据选中的过程标签类型创建 CMT

**注意：**用于生成 CMTs 的 PTTs 保持不变，如果 PTTs 位于主数据库中创建 CMT 的同一个层级文件夹，CMTs 生成后 PTTs 依然位于该文件夹下。

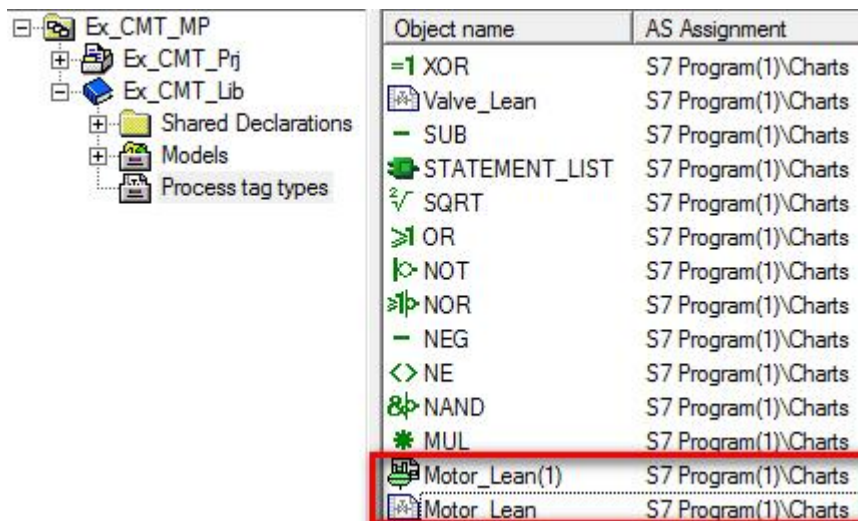


图 2-9 PTTs 和 CMTs 位于同一目录情况

### 2.2.3 直接在主数据库创建 CMT

1. 工厂视图下，在主数据库的层级文件夹下插入空白的 CMT。



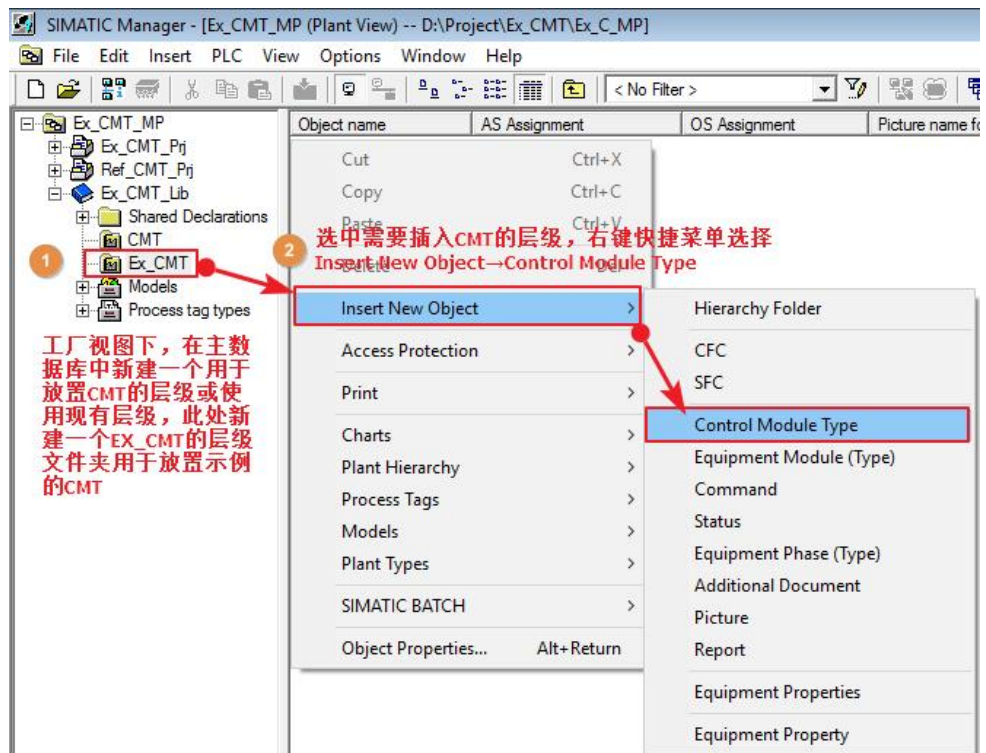


图 2-10 创建 CMT

- 修改生成的 CMT 的名字为自己想要的名字，例如 Ex\_Motor。双击打开该 CMT，从左侧主数据库或 APL 库中添加驱动块、电机块及联锁块等，以满足电机模板的控制要求，如图 2-11 所示。

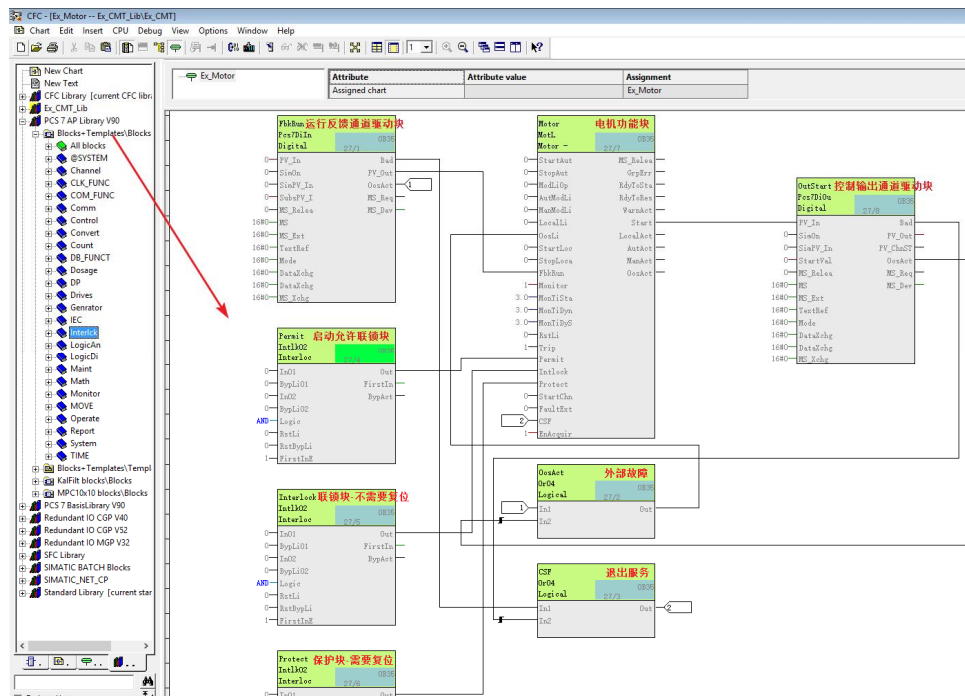


图 2-11 CMT 中添加功能块

- 使用拖拽的方式将需要的块和参数等添加到工艺 I/Os 中，如图 2-12 所示。

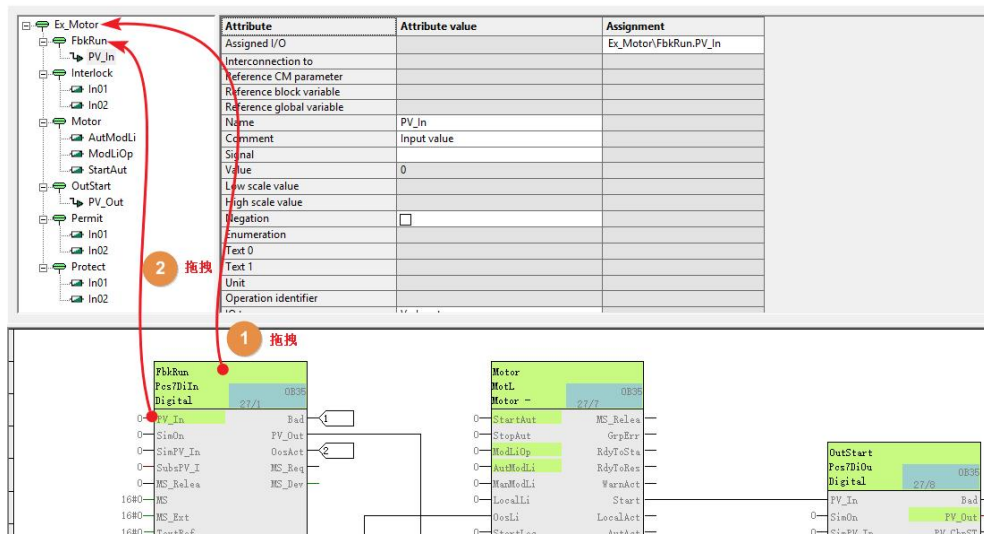


图 2-12 CMT 中添加工艺 I/Os

**注意：**该文档中接下来的内容均基于该方式下创建的 CMT 为例进行说明。

4. 如果将来要在 Advanced ES, PAA 或 COMOS 中修改消息, 需要在 CMT 的工艺 I/O 中添加消息, 如下图所示在需要添加消息的 CM 处点击右键:

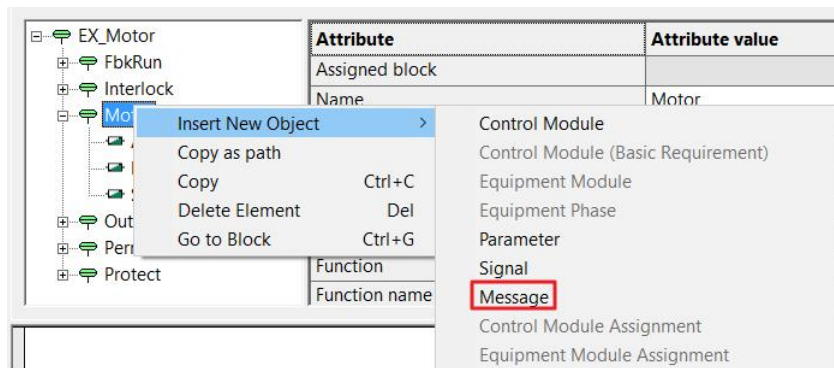


图 2-13 CMT 中添加消息

5. 组态消息的步骤如下图所示, 首先将消息对应的报告块(功能块)选中拖拽到“Assigned block/Assignment”。然后为消息标识符(Message identifier)赋值: SIG1,SIG2,...SIG8(例如输入 SIG1)。此时插入消息的 Name, Message class, Priority, Event 等会自动从对应的报告块获取默认值(可以根据需要进行修改)。

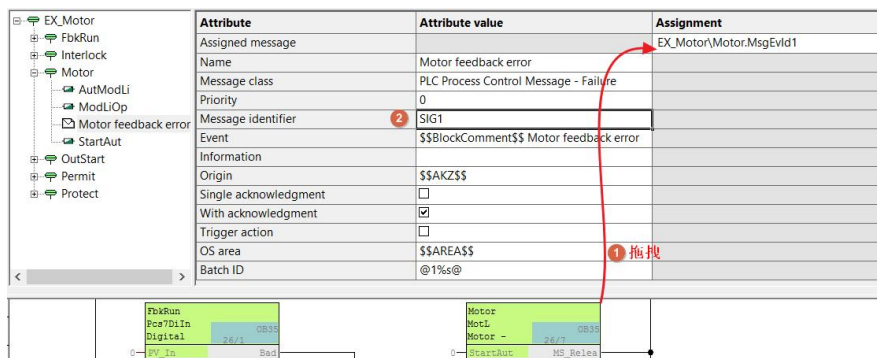


图 2-14 CMT 中组态消息

## 2.3 变体和替代选择

### 2.3.1 变体 (Variant) - 可选对象

在 Chap. 2.2 中已经创建了一个 CMT 并且添加了所需的功能块和工艺 I/Os。其中该 CMT 所包含的 Permit、Interlock、Protect 三个联锁块可能并不是所有的电机都需要亦或只是包含其中的一个或两个。此时，可以使用 CMT 的“Optional”选项。

1. 打开 Chap. 2.1 中新建的 CMT，激活 Permit、Interlock、Protect 的可选选项-“Optional”。

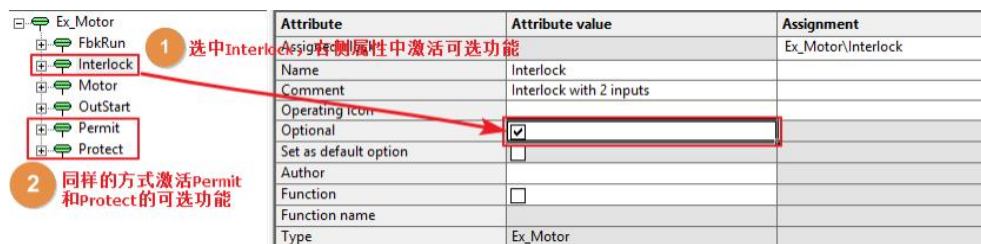


图 2-15 CMT 激活联锁块的可选功能

2. 实例化 CMT 到当前项目的对应层级文件夹下，即使用 CMT 创建 CM，并重命名为需要的设备位号。

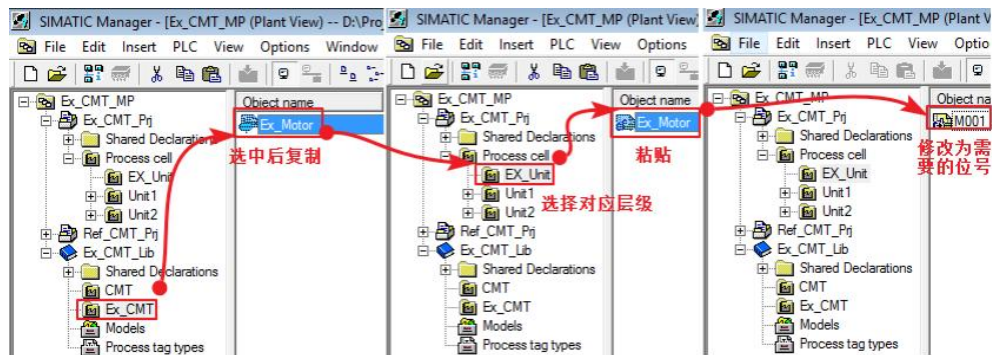


图 2-16 根据 CMT 创建 CM

3. 打开所创建的 CM 并显示工艺 I/Os（如果没有显示参考 Chap. 2.2.1 的第 2 步），激活变体功能：工艺 I/O 最上层右键快捷菜单选择“Variants...”。

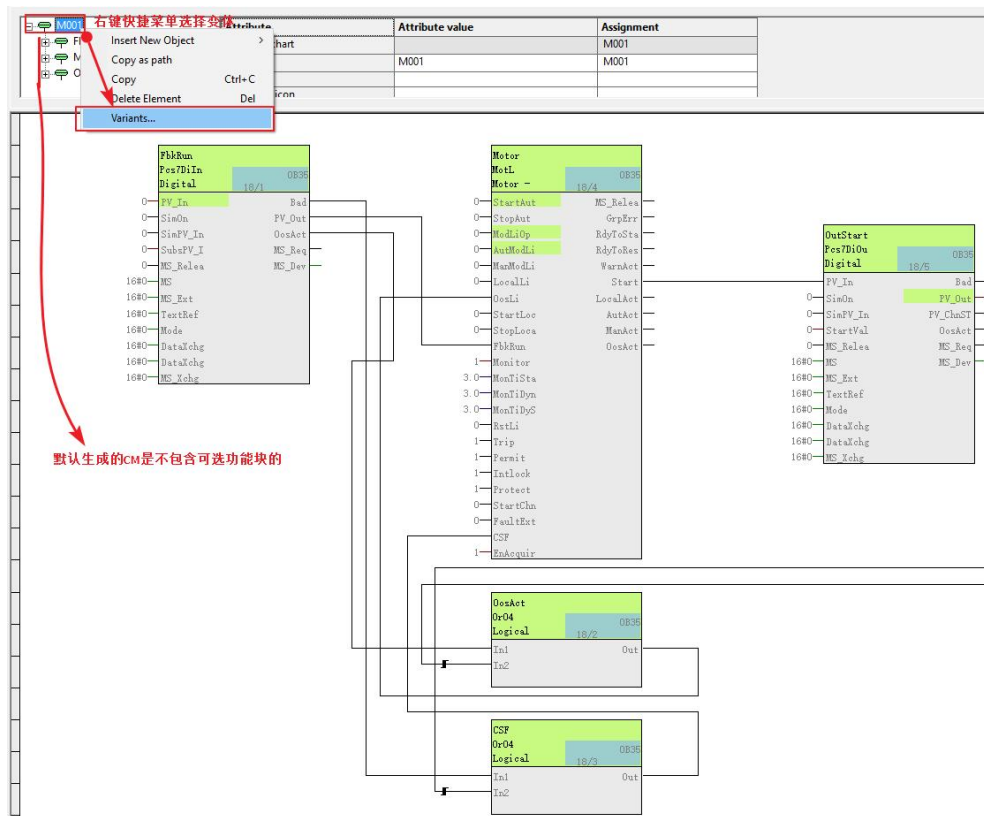


图 2-17 打开变体对话框

- 在弹出的对话框中，选择需要的联锁块，例如选择 **Permit** 和 **Interlock** 两个联锁块。

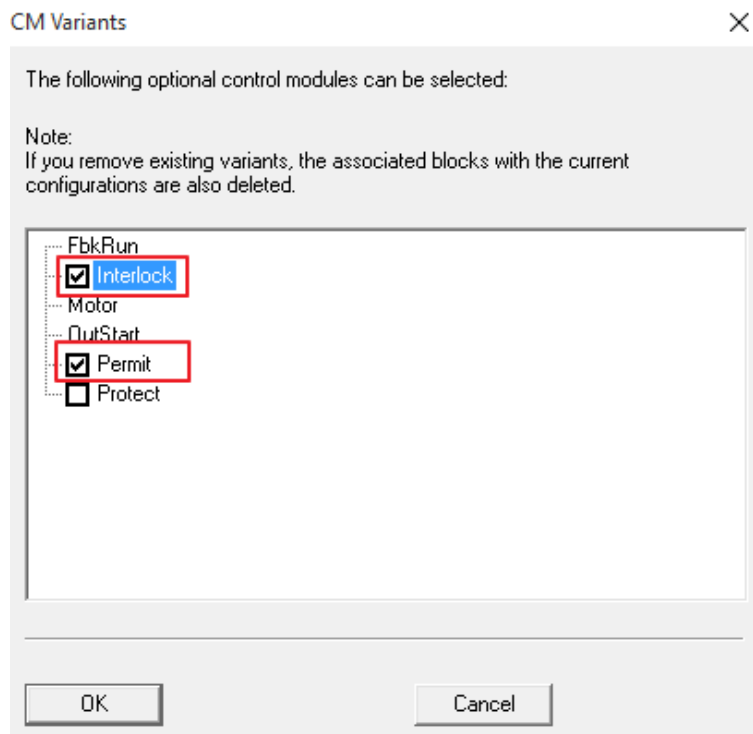


图 2-18 选择可选块

5. 结果如下图所示，Permit 和 Interlock 两个联锁块将显示在 CM 中。

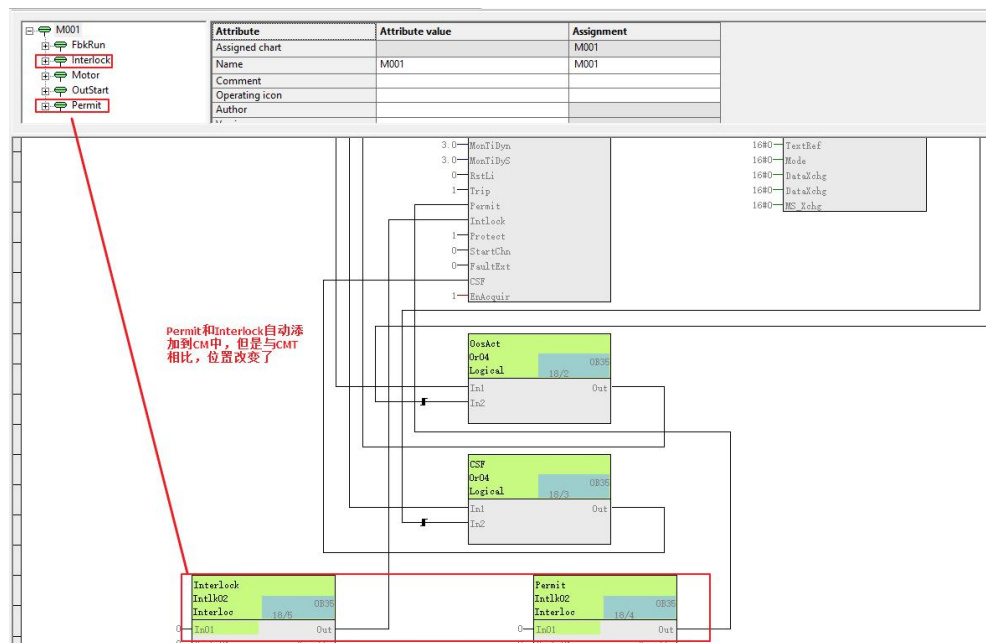


图 2-19 选择可选块后的结果

### 2.3.2 替代对象 (Alternative objects)

Chap. 2.2.3 中的 CMT 实例化时有可能有的设备 Interlock 管脚需要连接的联锁条件 >2 个，此时再使用 Intlk02 就不够用了，需要 Intlk04 或 Intlk08。另外有一部分设备可能只有 <2 个的联锁条件，不想使用 Intlk04 或 Intlk08 这种联锁块。此时可以使用 CMT 的替代对象的功能。（另外在创建模拟量监视的 CMT 时对于多种信号类型所需驱动块不同的情况也可以使用替代功能完成。）

- 在上述创建的 CMT 基础上，增加一个 Intlk04 的功能块，命名为 Interlock\_04。Interlock\_04 和 Interlock\_02 的“Optional”选项都要勾选，否则工艺 I/Os 中两个输出不能和一个输入互连。

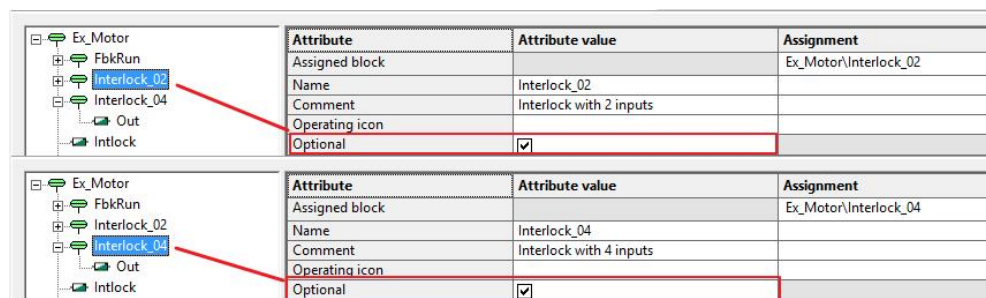


图 2-20 激活可选功能

- 将 Motor 块的 Intlock、Interlock\_04 的 Out (In01-In04 等视需要添加)、i、Interlock\_02 的 Out 管脚添加到工艺 I/Os，Motor 块的 Intlock 已经和 Interlock\_02.Out 的管脚存在互连，再将 Interlock\_04.Out 管脚和 Intlock 管脚在工艺 I/Os 中进行互连（CFC 中一个输入是不能连接两个输入的）

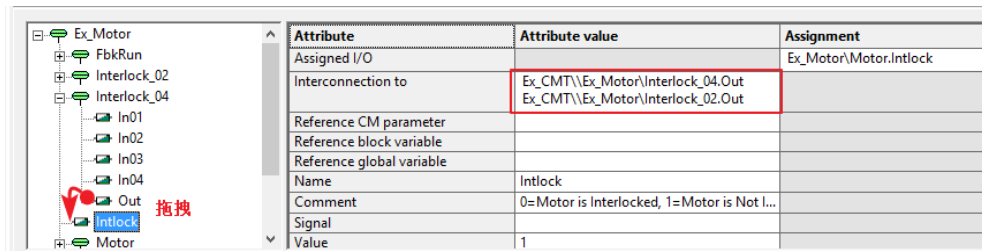


图 2-21 工艺 I/Os 中两个输出连接到同一个输入

- 实例化该 CMT，即根据 CMT 生成 CM，参考 Chap. 2.3.1，并修改为自己需要的位号，此处假设为 M002

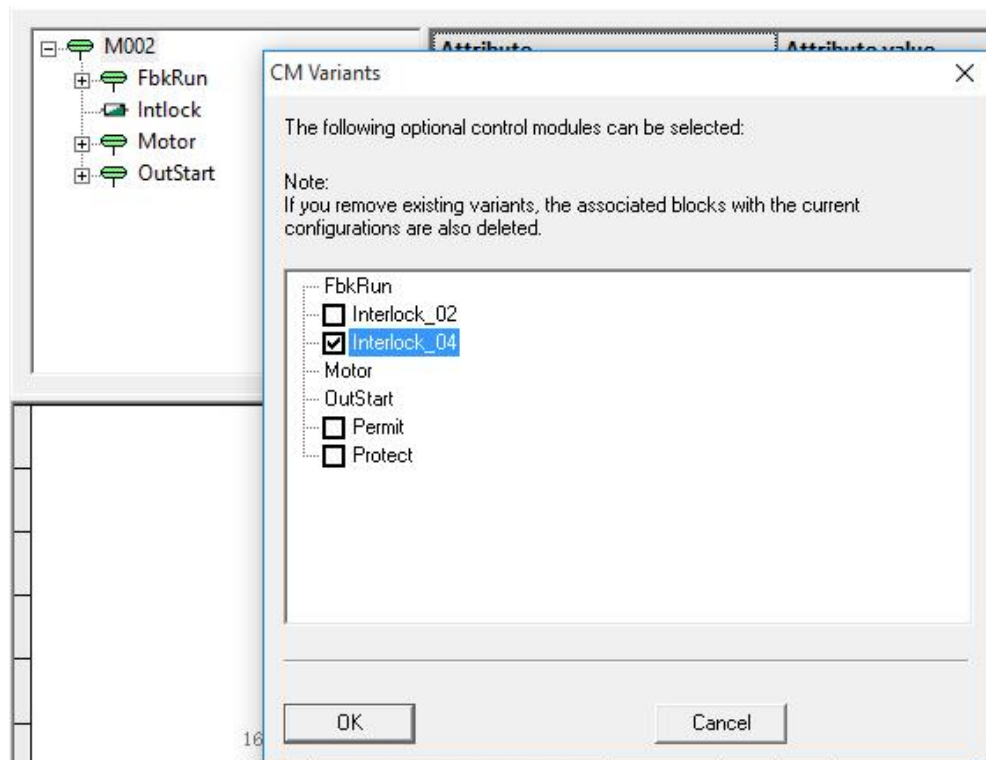


图 2-22 变体中选择所需的可选块。

- 打开 M002，变体中根据实际需要选择所需的 Interlock 块。此处假设选择 Interlock\_04。如图 2-20 所示。

**注意：**可以同时激活 Interlock\_02 和 Interlock\_04，但是只有两个中的一个和 Intlock 管脚互连。

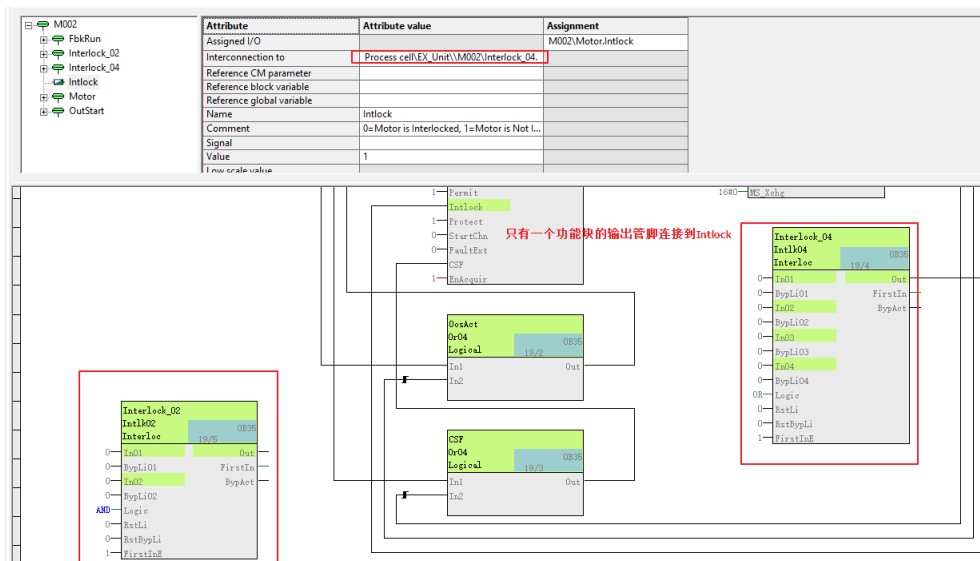


图 2-23 变体中替代块激活多个时结果

## 2.4 CMT 同步 CM（实例）

通过 CMT 创建的 CM，可以利用同步功能将 CMT 中更改的内容同步给已经存在的 CMs（即使一个 CMT 对应多个 CM）。同步功能需要使用 SIMATIC Version Cross Manager 的授权。

信号和参数在具有如下属性时不会被同步。

- S7\_m\_c=true
- 分配给工艺 I/Os
- 存在交叉过程变量的连接
- 连接到实例化的块

### 2.4.1 在 CMT 中新增对象（FB）

#### 目标

- CMT 中新增一个 FB
- 将改变同步到实例

#### 步骤

1. 打开主数据库中 Ex\_CMT 层级中的 CMT Ex\_Motor
2. 插入一个 Pcs7DiIn 驱动块，并修改块名称为 Fault，将相应管脚和 CMT 中已经存在功能块互连：Fault.PV\_Out--Motor.Trip，Fault.Bad--CSF.In3，Fault.OosAct—OosAct.In3。添加 Fault.PV\_In 到工艺 I/Os。

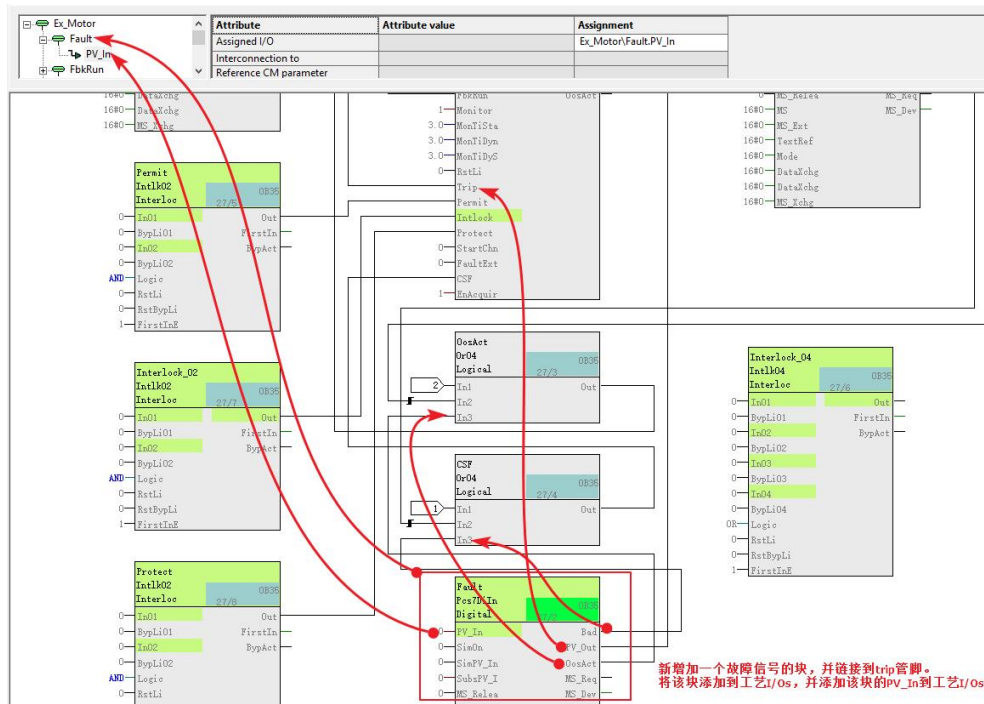


图 2-24 CMT 中新增故障通道块

3. 进行同步：工厂视图下选中需要同步的项目→右键快捷菜单→Plant Types →Synchronize，在弹出的对话框中选择选择要执行同步的 CMT，按需选择带有运行顺序和图形坐标选项。



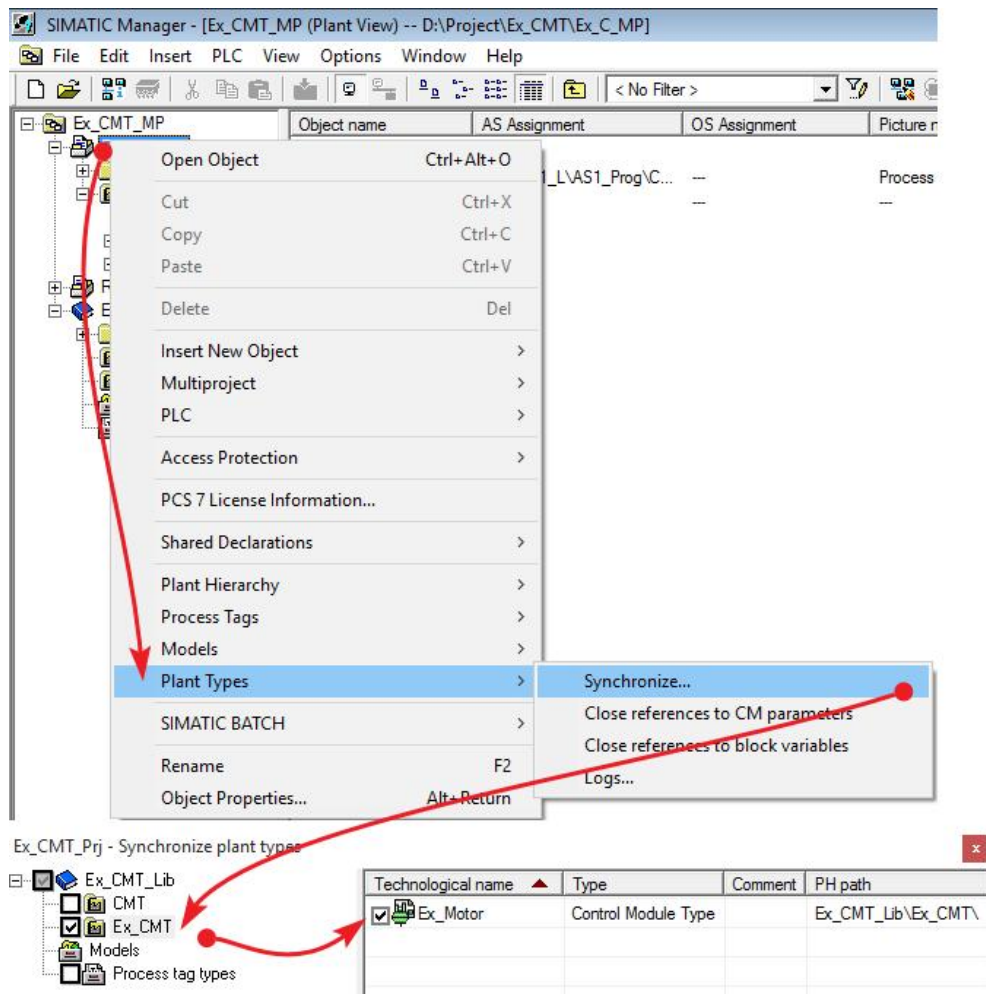


图 2-25 CMT 同步

4. 检查数据，并列出不同之处。

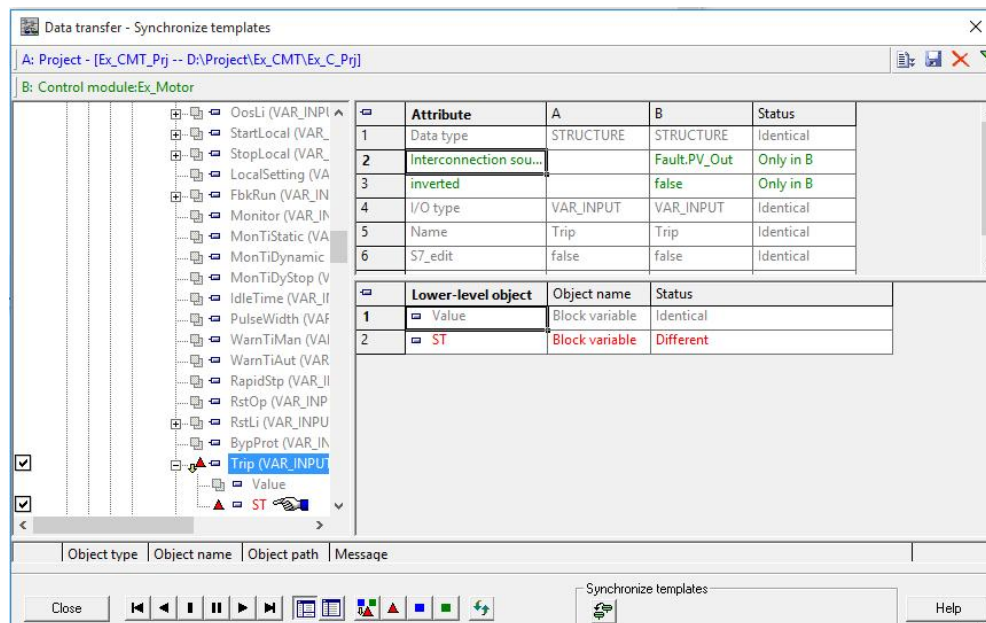


图 2-26 检查数据-同步不同之处

5. 在不同之处点击右键，可以从快捷菜单中的 open project 选项同时打开 CMT 和 CM

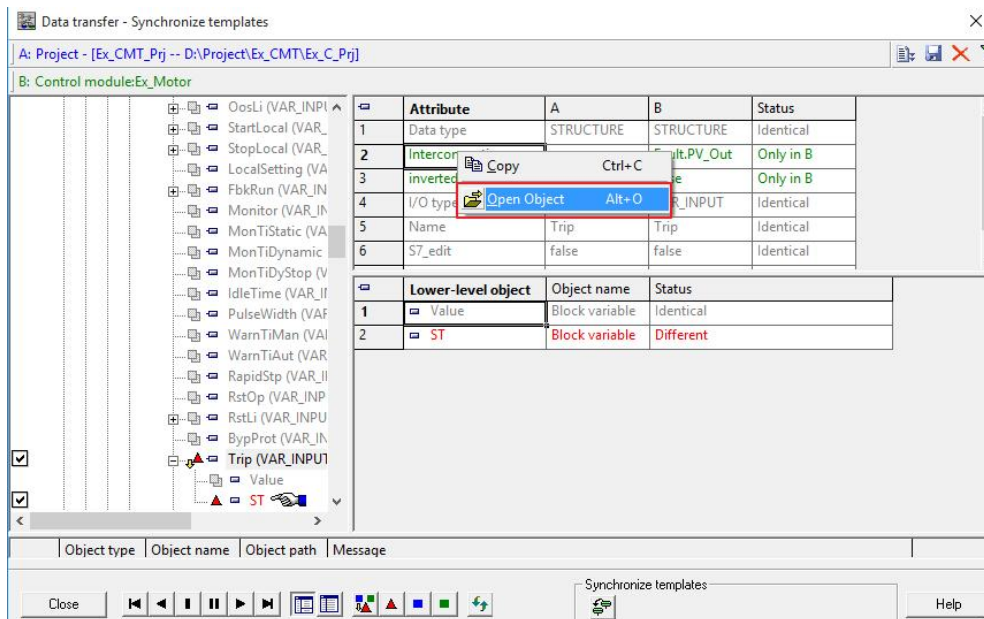


图 2-27 比较窗口打开 CMT 和 CM 快捷菜单

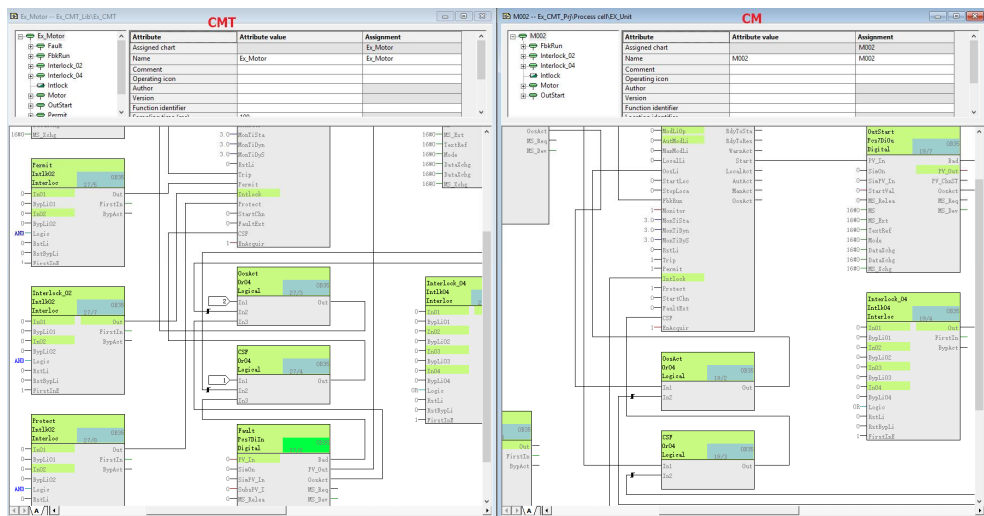


图 2-28 从同步对话框快速打开 CMT 和 CM

6. 点击同步按钮，执行同步。同步完成后在实例中检查同步结果。

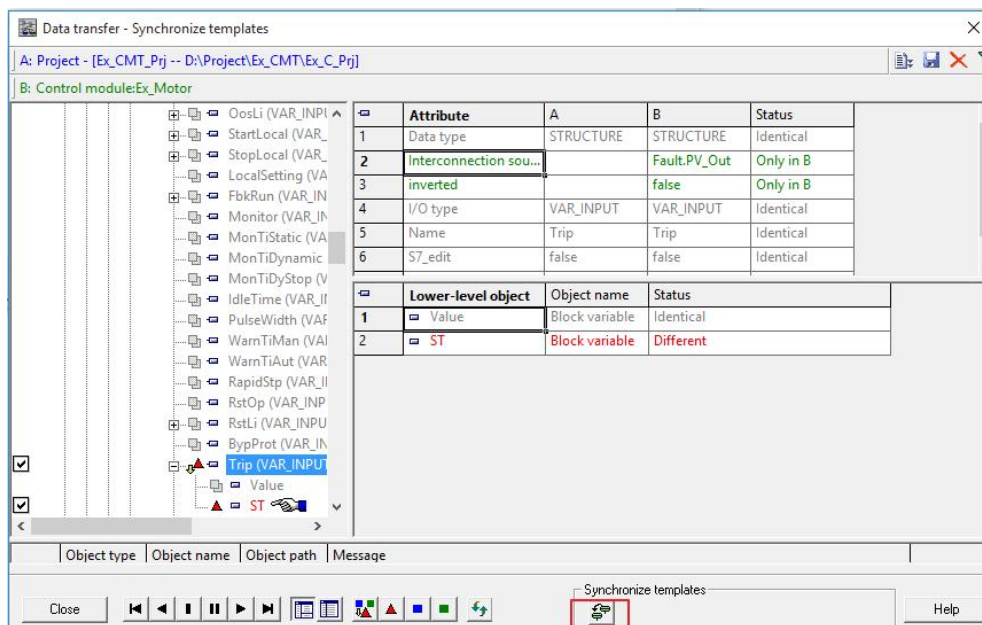


图 2-29 执行同步

## 2.4.2 更改属性

### 目标

- 更改 CMT 和 CM 中 FBs 和 parameters 的属性（例如 Comment, Block group, Text0/Text1）
- 同步 CMT 到实例

### 更改 FB 的 Comment 以及 Block Group 属性

1. 打开主数据库中 Ex\_CMT 层级中的 CMT: Ex\_Motor
2. 打开 Motor 块的属性，将 Motor 块的注释从“ Motor - Large” 修改为 “ Test\_Comment\_Motor”，并修改 Block Group 属性为“ HidenGroup1”
3. 打开一个实例 CM: M002，修改其中的 Motor 块 Comment 为 “ Test\_Comment\_M002”，并修改 Block Group 属性为“ HidenGroup2”。

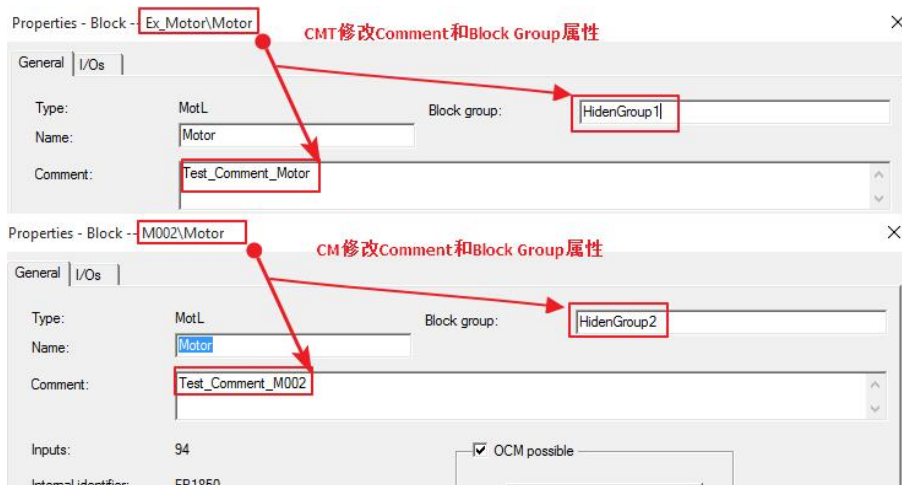


图 2-30 更改 CMT 和 CM 的 Comment 及 Block Group

4. 执行 CMT 同步，同步后的结果如下图和下表所示：

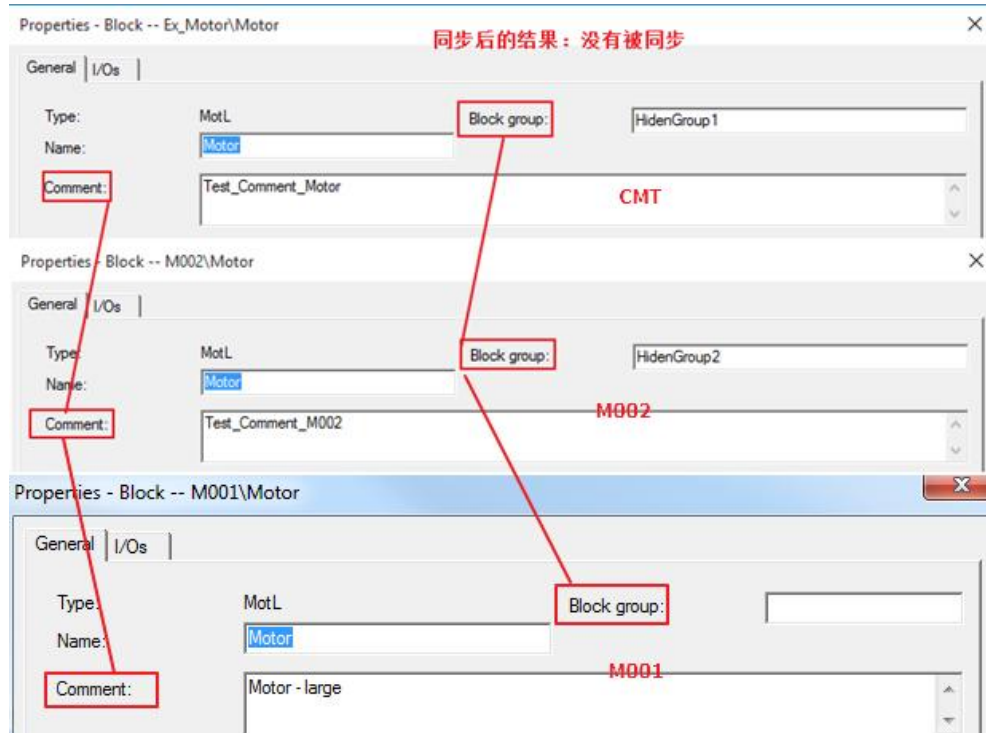


图 2-31 更改 CMT 和 CM 的 Comment 及 Block Group 后的同步结果

	在 CMT 中所作修改	在 CM 中所作修改
同步结果	未同步，CM 中的值不会被修改	未同步，CM 中的值不会被修改

表 2-1 Comment 同步结果

更改带 S7\_string 属性的 I/O 的 Text0/Text1

1. 打开 CMT: Ex\_Motor，修改 Permit 块的 In01 管脚的 Text0 文本为：CMT\_Per\_In01\_0，Text1 对应文本保持为空不变。

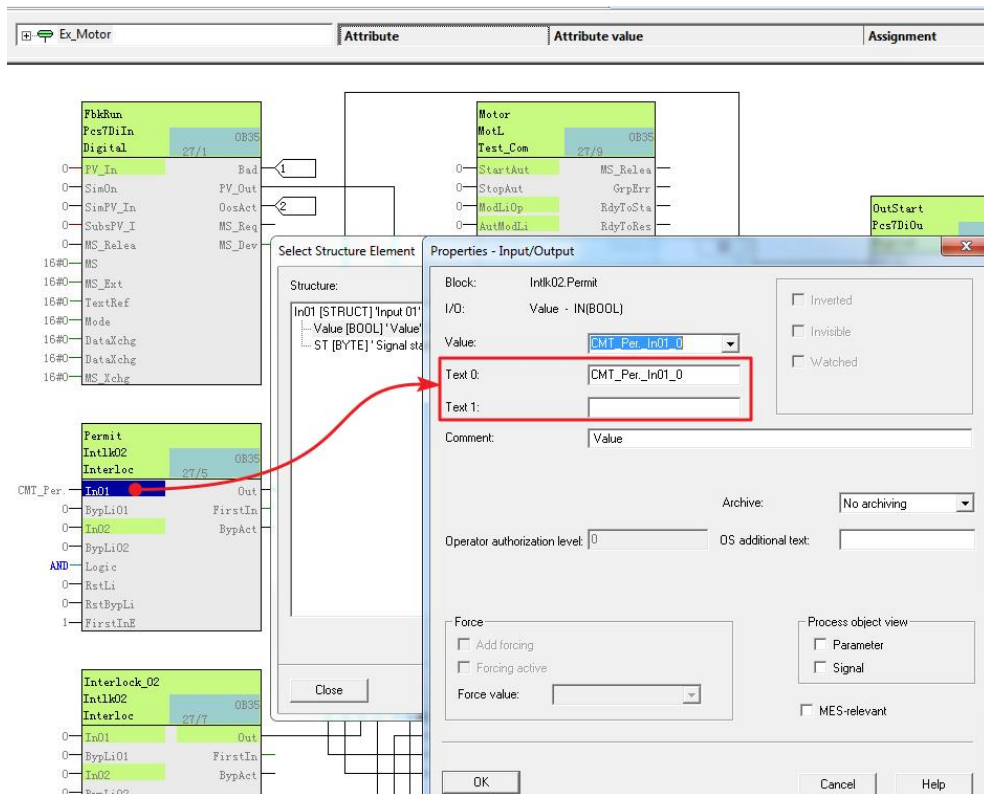


图 2-32 修改 CMT 中 Permitt 的 In1 管脚 Text0 文本

2. 此处为 CMT 实例 M001 的 Permitt 块的 In1 管脚的 Text0 和 Text1 分别输入文本：CM\_Per\_In01\_0 和 CM\_Per\_In01\_1。

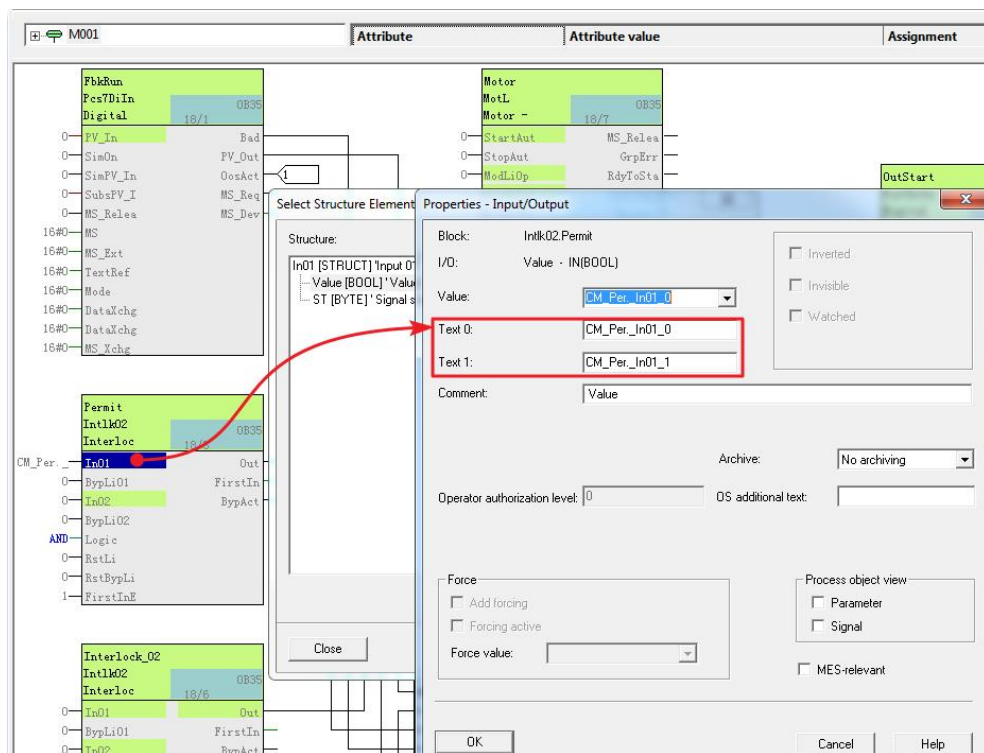


图 2-33 修改 M001 中 Permitt 的 In1 管脚 Text0 和 Text1 文本

3. 执行 CMT 同步，同步结果如图和下表所示。

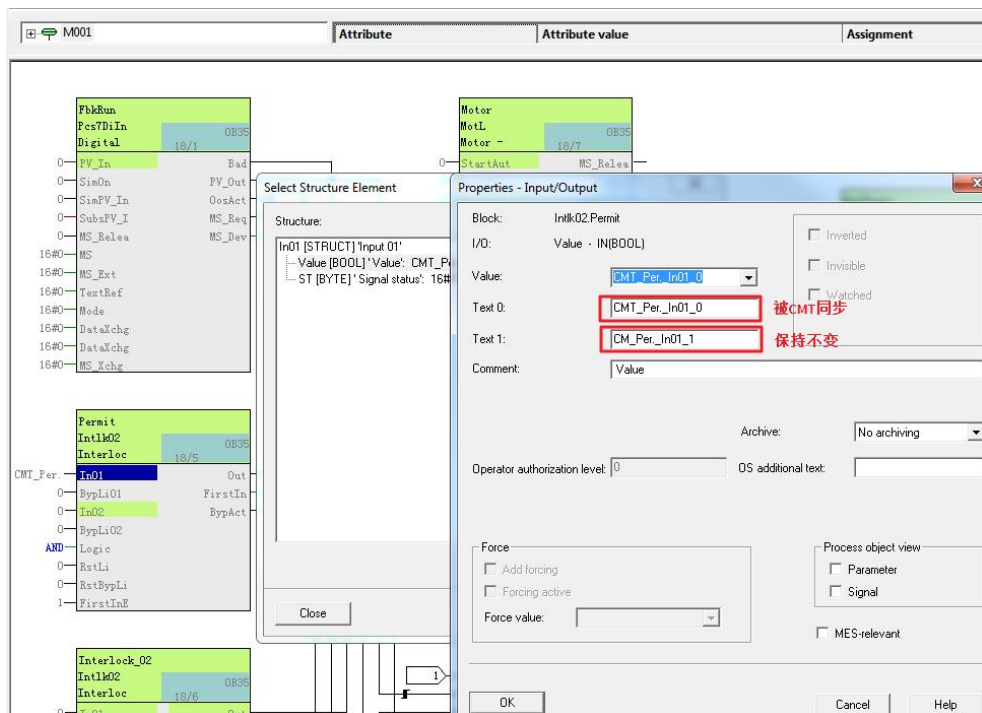


图 2-34 Text0 和 Text1 同步结果

	在 CMT 中所作修改	在 CM 中所作修改
同步结果	如果在 CMT 中定义了 Text0 或 Text1，则同步时总是将 CMT 中定义内容同步到 CM 中。	如果在 CMT 中没有为 Text0 或 Text1 定义文本，则 CM 中对于 Text0 或 Text1 的更改将会保持

表 2-2 Text0 和 Text1 同步结果

2.4.3 更改参数值

目标

- 更改 CMT 和/或 CM 的参数值
- 同步 CMT 到 CM 实例

更改参数值步骤

1. 在项目主数据库中新建一个 FB 块，该块各输入管脚具有不同属性。新建一个 CMT 命名为 Ex\_Sync\_Para，并在该 CMT 中添加该功能块。其中输入管脚 in\_TED 添加到工艺 I/Os。

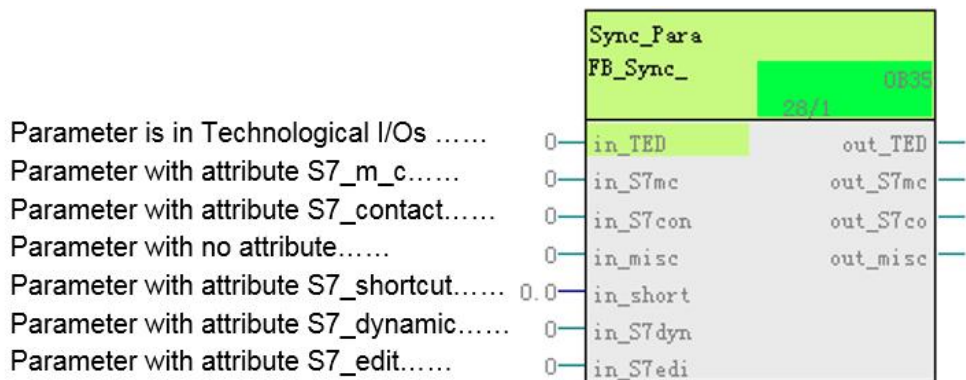


图 2-35 管脚具有不同属性的 CMT

2. 使用 CMT 创建实例：“ Sync\_Para\_1”。
3. 修改实例“ Sync\_Para\_1” 中各个输入管脚的值，然后执行 CMT 同步。
4. In\_misc, in\_short, in\_S7dynamic, in\_S7edit 管脚值被 CMT 同步。




图 2-36 管脚具有不同属性的 CM 同步结果

## 2.5 功能（实例特殊扩展）

### 目标

- 创建一个逻辑组合的功能块
- 将其创建为 Function
- 在实例 CM 中使用该 Function
- 修改 Function 并将其同步到实例

步骤,

1. 在主数据库创建一个新的 CMT: Ex\_Function\_Logic, 在该 CMT 中不使用子 CM, 所有的参数直接拖拽到 Technological I/Os 的最上层。激活 “Function” 功能, 该 CMT 的图标将变为  Ex\_Function\_Logic。

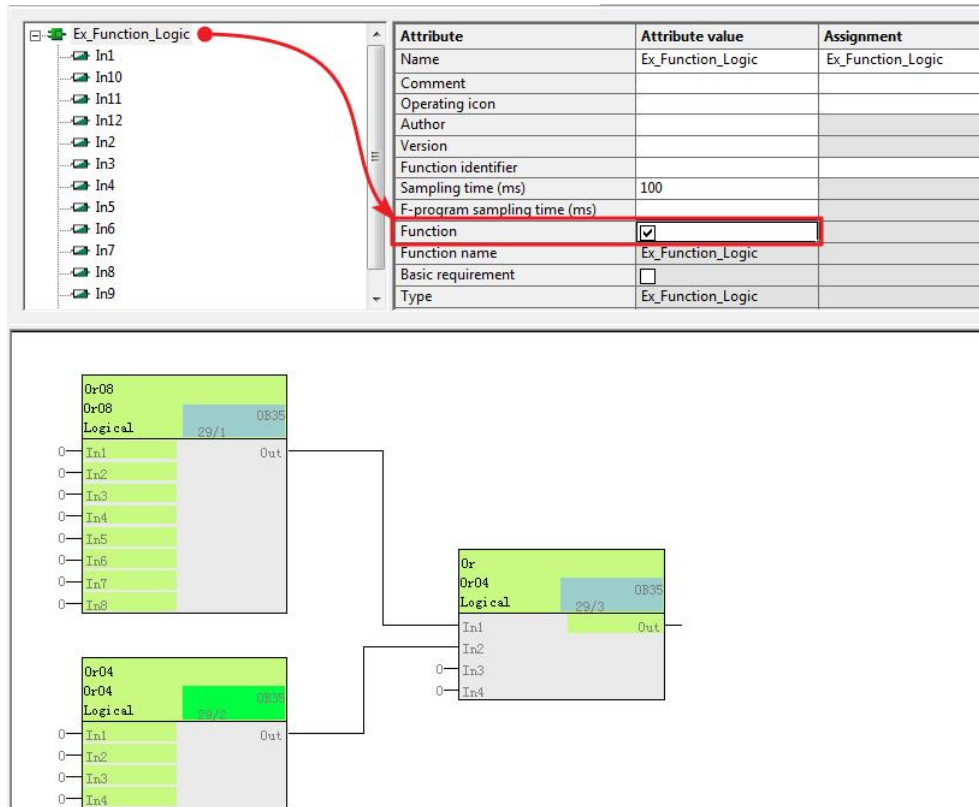


图 2-37 创建 Function

2. 打开之前创建的 CM 实例 M001, 将新建的 Function: Ex\_Function\_Logic 添加到 M001 的工艺 I/Os 中。并将 Function 的输出管脚 Out 连接到 Motor 的 Protect 管脚。



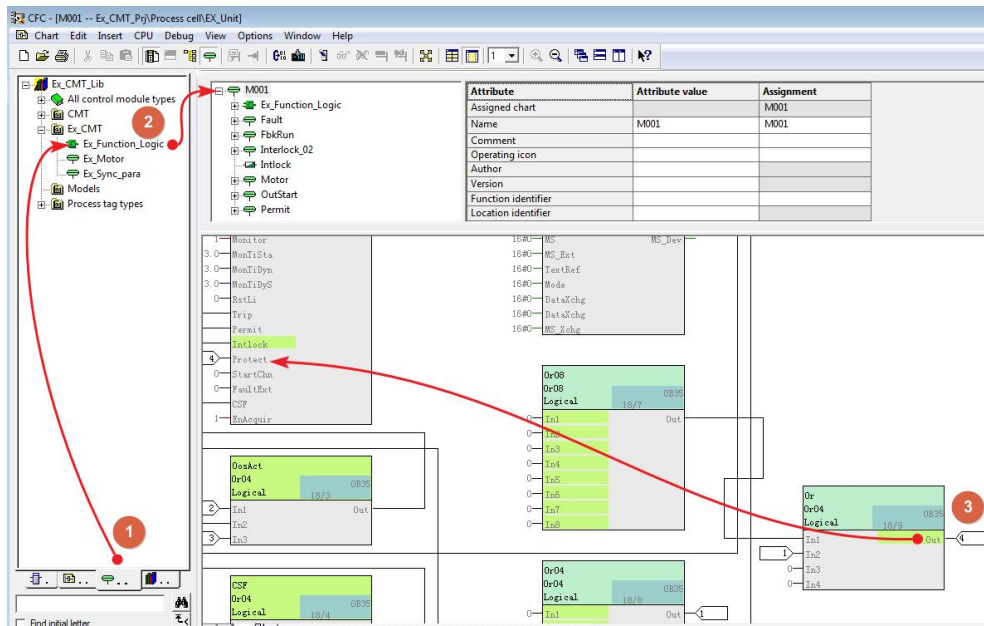


图 2-38 将 Function 添加到 CM，并连接相应管脚

3. 打开之前定义的功能，添加一个新的 Logic 块，并将输入管脚添加到工艺 I/Os，新增 Or04 块的输出连接到之前的 Or 的 In3 输入管脚。（注意如果激活 Function 功能后，工艺 I/Os 中的管脚无法重命名，如果要修改工艺 I/Os 的管脚名可以先不激活 Function，修改完毕后重新激活 Function 功能。）

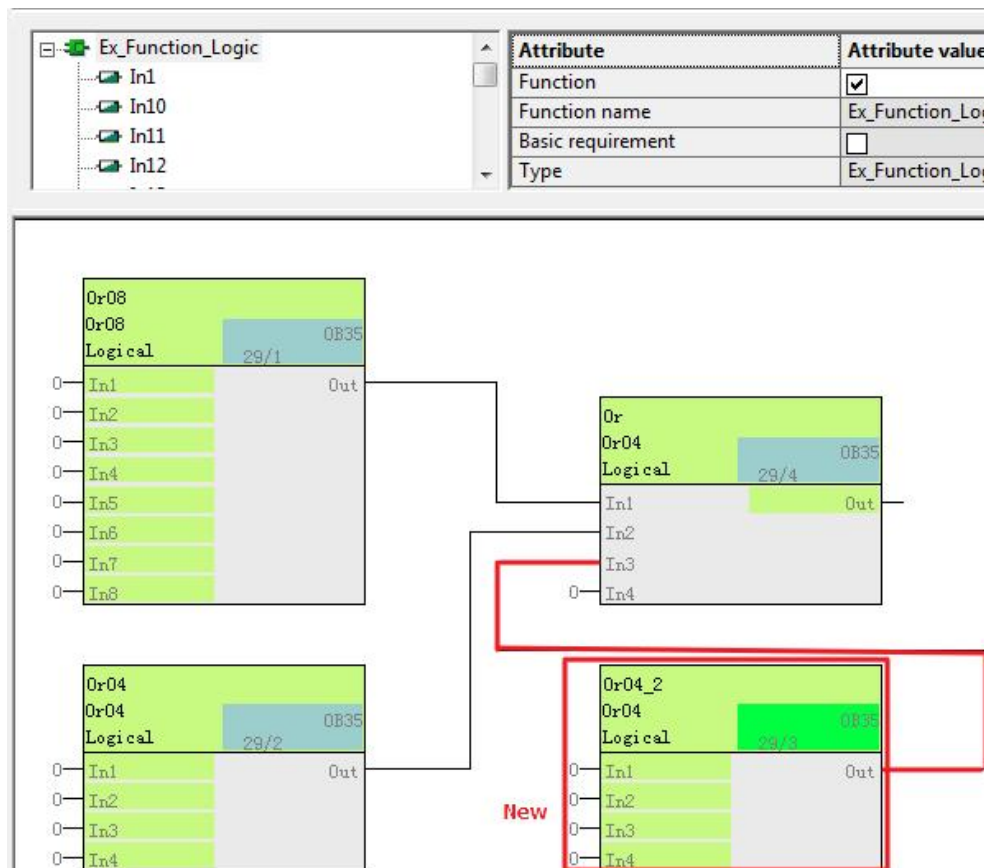


图 2-39 修改 Function

4. 执行 CMT 同步，将 Function 的修改同步到 CM。同步完成后，打开之前调用 Function 的实例 M001，Function 中的修改已经同步到 M001。

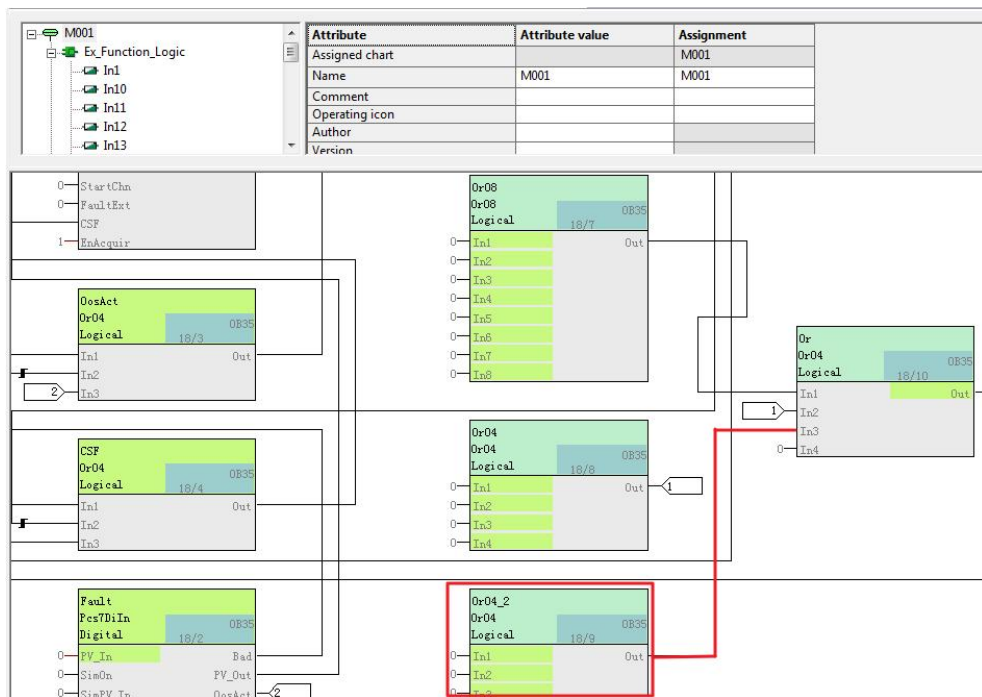


图 2-40 Function 同步结果

## 2.6 状态和命令

### 目标

- 在 CMT 中创建命令和状态

### CMT 中创建命令

1. 打开 CMT: Ex\_Motor。在工艺 I/Os 中添加新对象“Command”（右键快捷菜单），并修改名字为 Mauto。

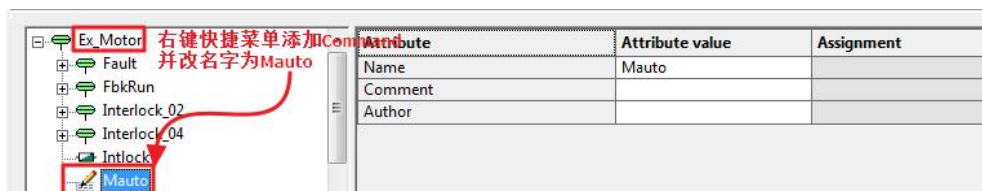


图 2-41 Ex\_Motor 添加 Command

2. 选中命令 Mauto，右键快捷菜单打开其属性。将弹出类似 SFC 的命令窗口。类似操作 SFC 的方式，为参数赋值。

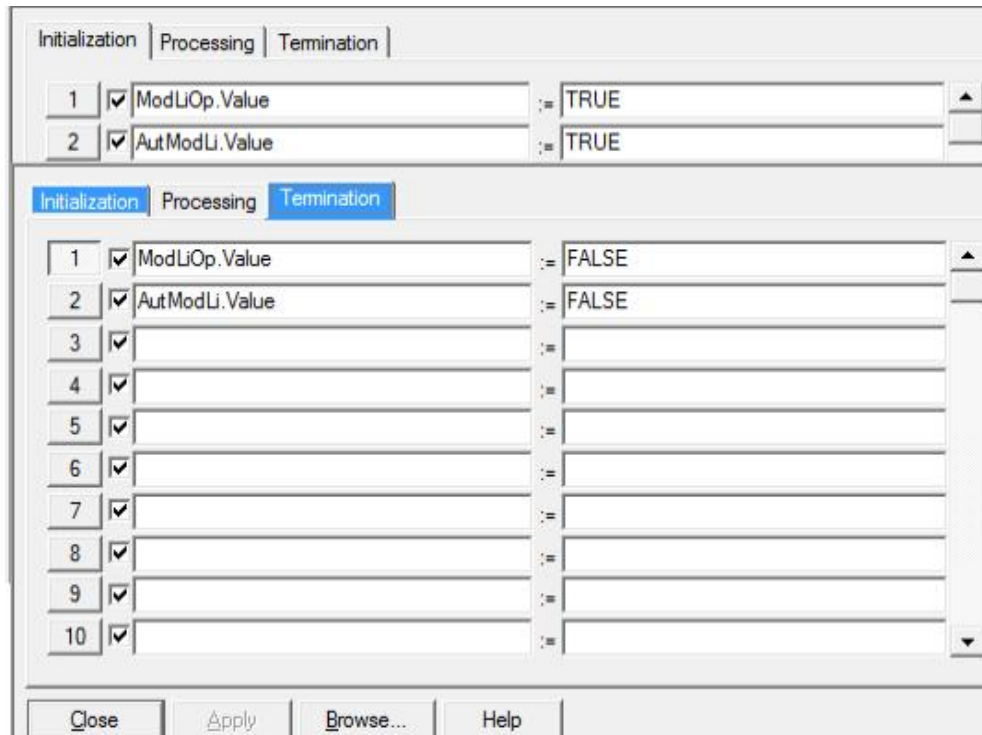


图 2-42 为 Mauto 相关参数赋值

3. 参照 Step1-2 继续添加 Comand “ Mstart” 。

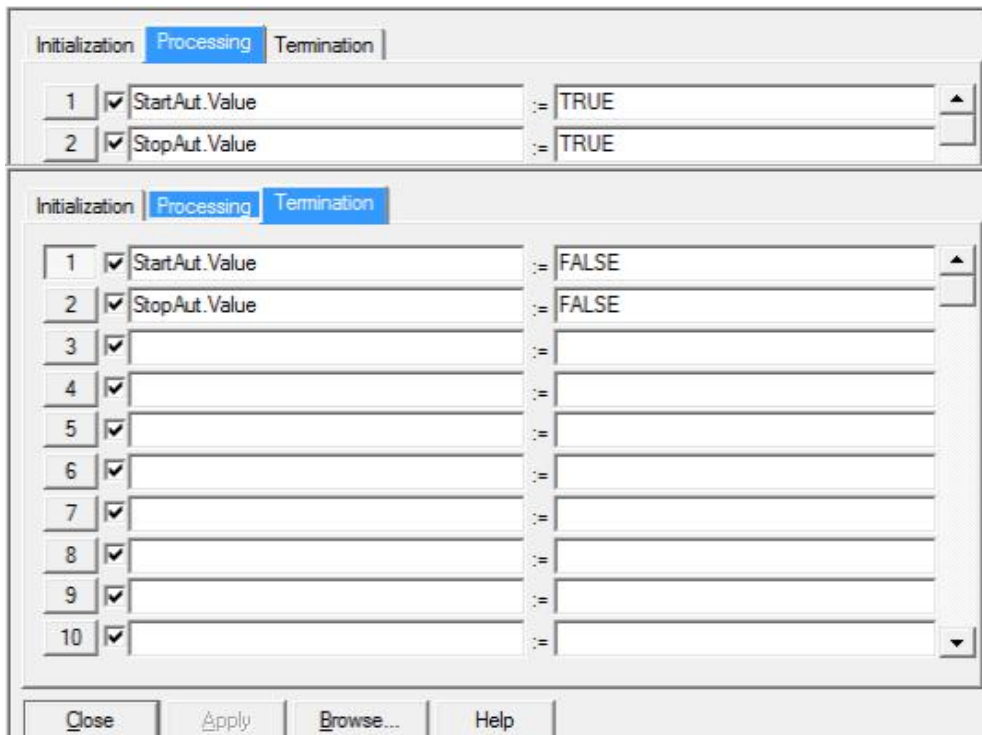


图 2-43 为 Mstart 相关参数赋值

4. 参照 Step1-2 继续添加 Comand “ Mstop” 。

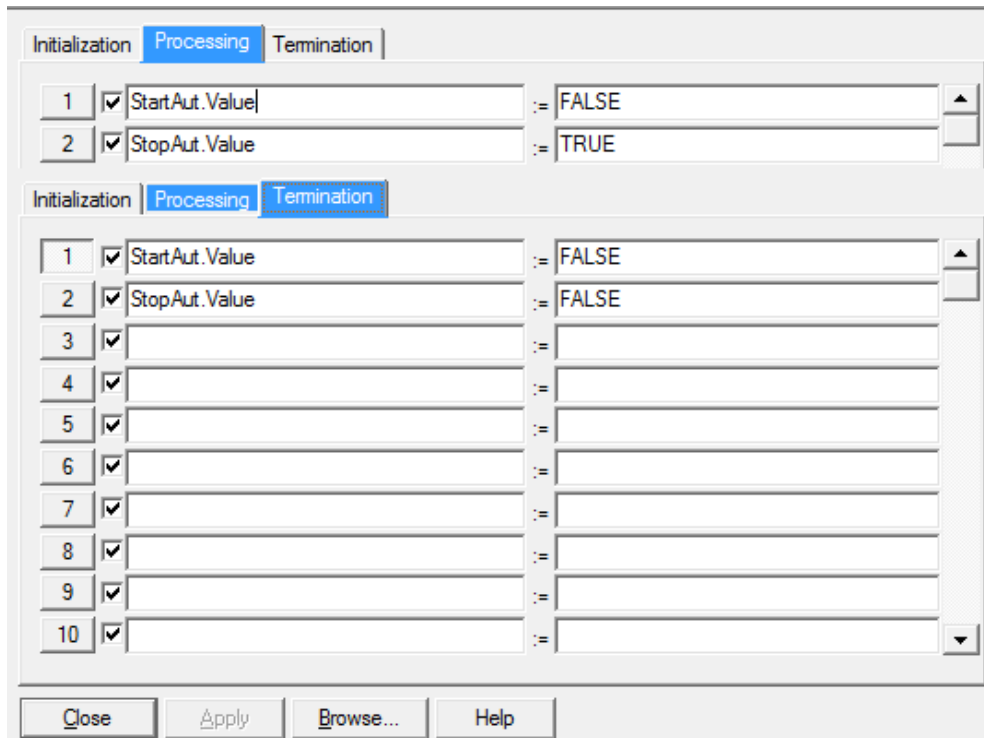


图 2-44 为 Mstop 相关参数赋值

### CMT 中创建状态

1. 打开 CMT: Ex\_Motor。在工艺 I/Os 中添加新对象“ Status”（右键快捷菜单），并修改名字为 IsRun。

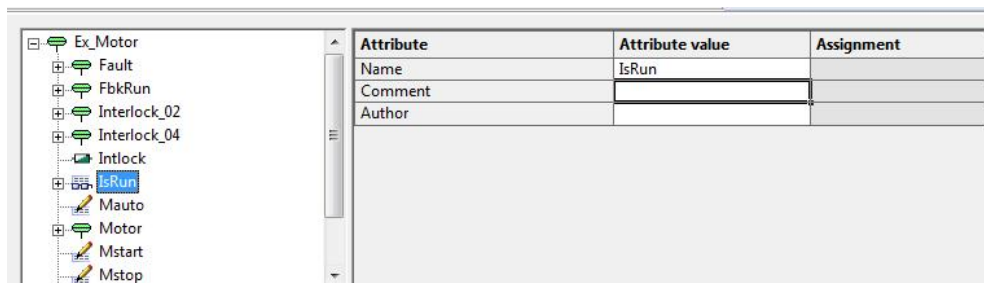


图 2-45 添加 Status

2. 选中状态 IsRun，右键快捷菜单打开其属性。将弹出类似 SFC 的状态窗口。类似操作 SFC 的方式，添加条件。

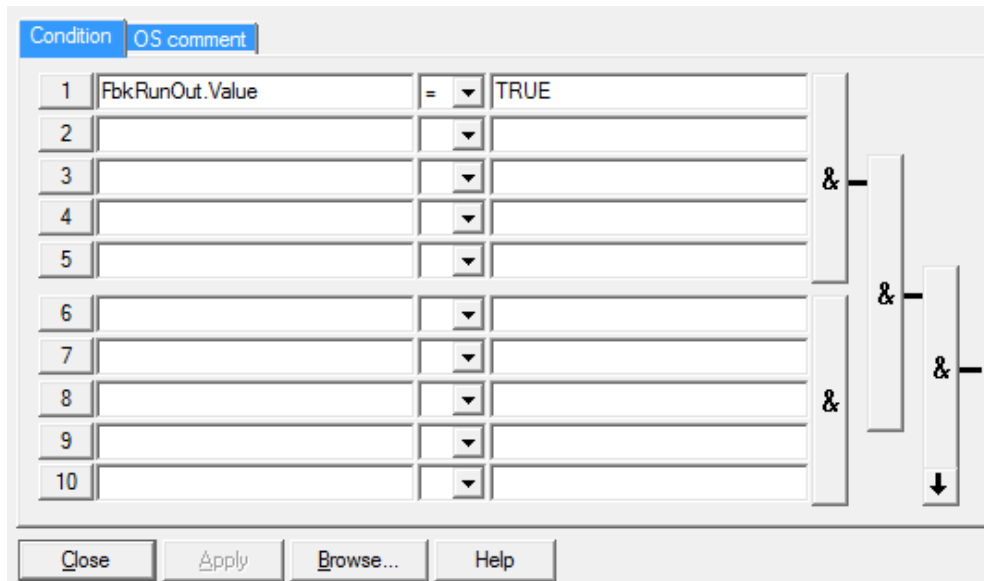


图 2-46 为 Status 添加条件

## 2.7 连接类型

### 2.7.1 互连到 CM 的参数

#### 目标

- 创建工艺 I/Os 之间的互连

#### 步骤

1. 通过拖拽的方式可以创建工艺 I/Os 之间的互连。

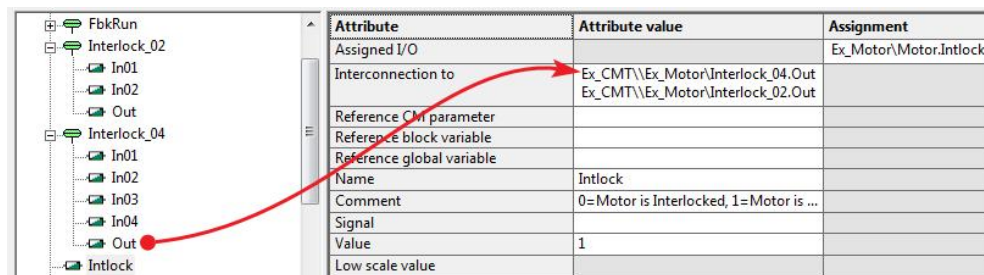


图 2-47 工艺 I/Os 之间互连

2. 也可以通过“ interconnection to” 的属性值处右键快捷菜单打开“ new interconnection”，手动输入目标地址。

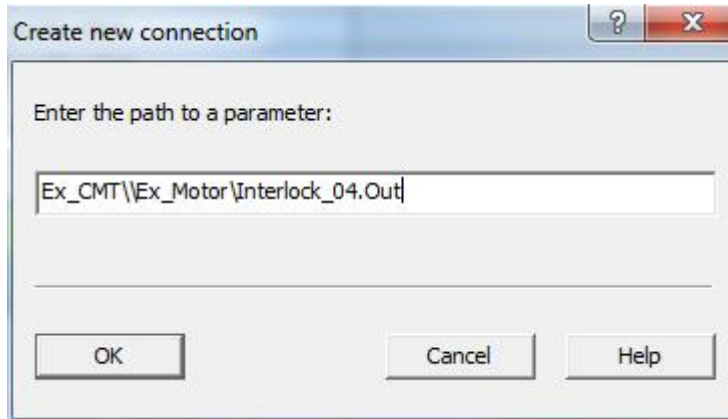


图 2-48 手动输入绝对地址

## 2.7.2 引用 CM 参数

### 目标

- 创建工艺 I/Os 之间的引用

### 步骤

1. 打开 CMT“ Ex\_Motor”，分别在 AutoModLi 和 ModLiOp 管脚的 Reference CM Parameter 参数的属性值处右键快捷菜单选择 New Reference。

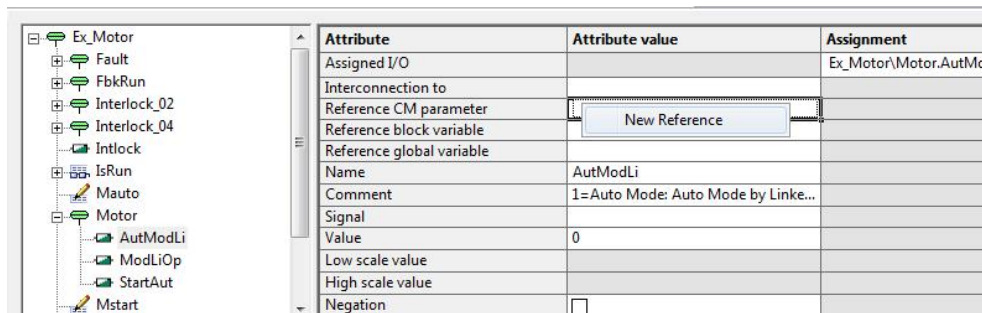


图 2-49 插入新的 CM parameter 引用

2. 将打开创建新引用的对话框，在对话框中输入目标地址的绝对或相对地址。  
(参考 Chap. 2.4.5.2)

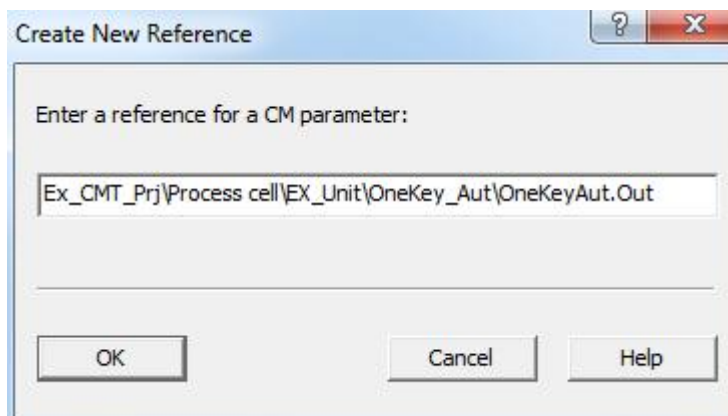


图 2-50 CM parameter 引用的绝对地址

3. 新创建的实例将会包含创建的 CM parameter 引用。注意之前创建的实例在执行同步后也不会包含后来创建的 CM parameter。

Attribute	Attribute value	Assignment
Assigned I/O	新建的实例包含CM parameter引用	M003\Motor.AutModLi
Interconnection to		
Reference CM parameter	Ex_CMT_Pj\Process cell\EX_Unit\OneKey_Aut\OneKeyAut.Out	
Reference block variable		
Reference global variable		
Name	AutModLi	
Comment	1=Auto Mode: Auto Mode by Linked or SFC	
Signal		

Attribute	Attribute value	Assignment
Assigned I/O	之前创建的CMT实例不会被同步	M001\Motor.AutModLi
Interconnection to		
Reference CM parameter		
Reference block variable		
Reference global variable		
Name	AutModLi	
Comment	1=Auto Mode: Auto Mode by Linked or SFC	

图 2-51 CM parameter 引用自动在新的实例中创建

4. 如果目标地址存在，引用可以被关闭。工厂视图→选择项目→点击右键快捷菜单→工厂类型→关闭 CM parameter 引用。

Object name	AS Assignment	OS Assignment	Picture name
Ex_CMT_MP			
Sh...	S1_Prog\C...	--	Process cell

Attribute	Attribute value
Assigned I/O	
Interconnection to	Process cell\EX_Unit\OneKey_Aut\OneKeyAut.Out
Reference CM parameter	
Reference block variable	
Reference global variable	
Name	AutModLi

图 2-52 关闭 CM parameter 引用

## 2.7.3 引用块变量

### 目标

- 创建工艺 I/Os 和 CFC-Parameter 之间的引用

## 步骤

1. 打开 CMT “ Ex\_Motor ” 或任意实例 CM。选择工艺 I/Os 中的 StartAut，右侧 Reference block variable 的属性值处新建引用。引用地址为绝对或相对地址（参考 Chap. 2.4.5.2）

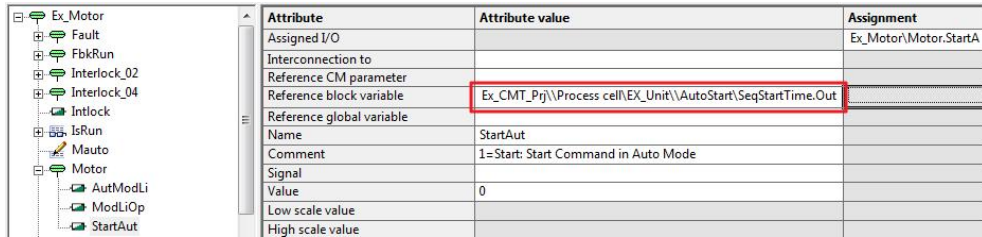


图 2-53 新建 block variable 的引用

2. 使用该 CMT 创建实例 M004，将会包含创建的 block variable 引用。如果是修改的 CM 则省略该步骤。

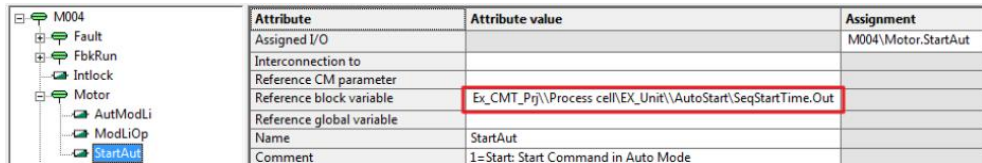


图 2-54 新建实例包含 block variable 的引用

3. 如果引用的目标地址存在，则可以将该引用关闭。Project→右键快捷菜单→Plant Types→Close Reference block variables

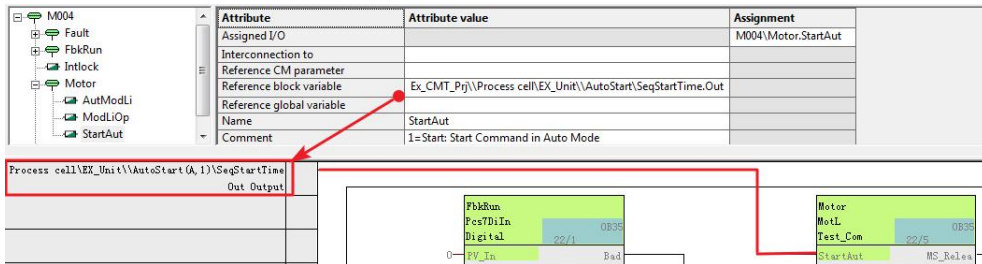


图 2-55 关闭 block variables 引用

## 2.7.4 引用全局变量

### 目标

- 创建工艺 I/Os 和数据块之间的引用

### 步骤

1. 打开 CMT “ Ex\_Motor ” 或其实例，将 MonTiStatic 添加到工艺 I/Os，并未其添加全局变量引用。

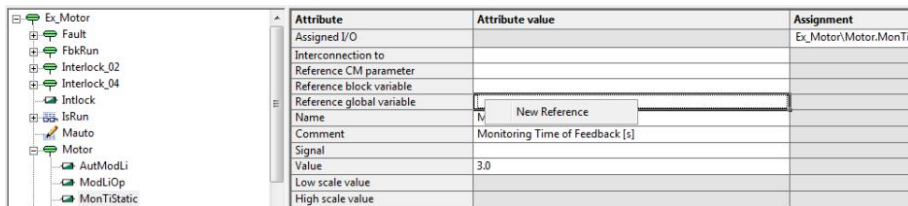


图 2-56 新建全局变量引用



2. 将打开创建新引用的对话框，在目标地址处输入全局变量。

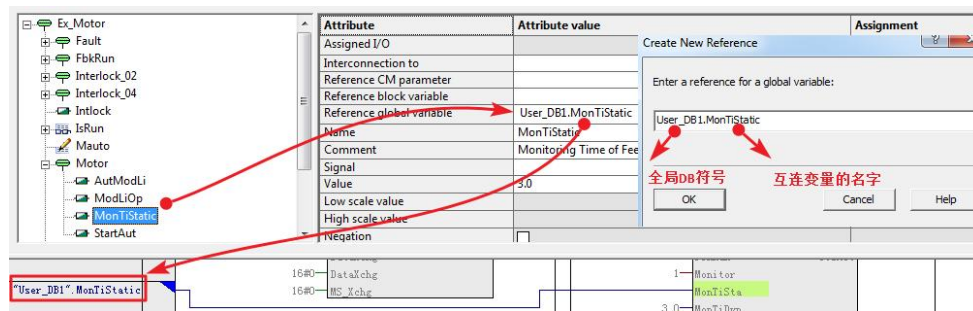


图 2-57 全局变量引用目标地址

3. 新建实例（如果是在 CM 处修改则省略此步骤），将会包含此全局引用，（之前已经存在的 CM 在同步后也不会被修改）。
4. 如果目标地址存在，则可以关闭该引用，关闭步骤如下图所示。

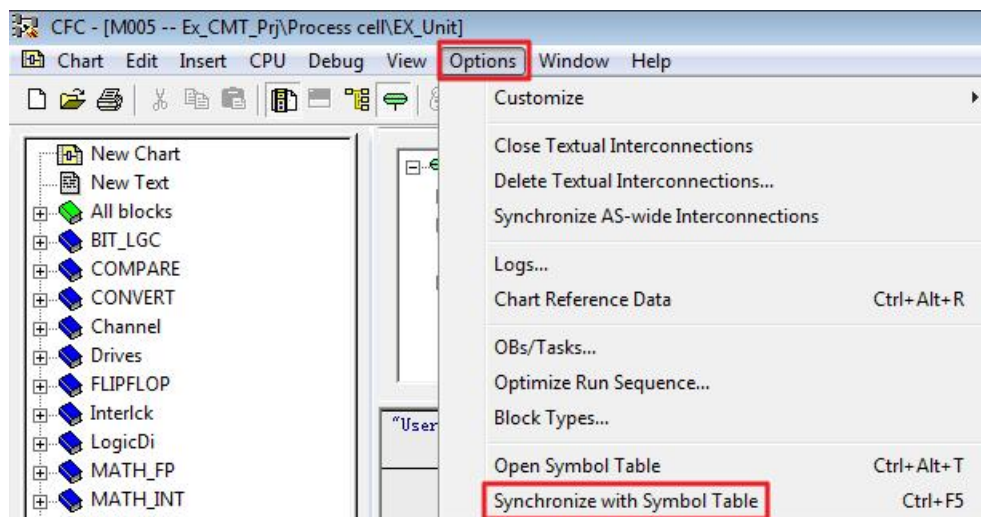


图 2-58 关闭全局变量引用

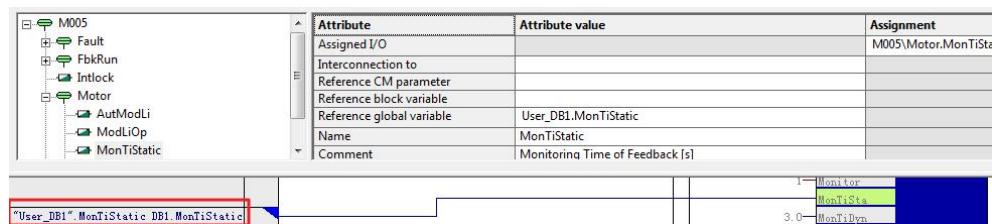


图 2-59 关闭全局变量引用结果