

PCS7软件使用方法

西门子 SIMATIC PCS 7(Process Control System 7)系统是完全无缝集成的自动化解决方案。可以应用于所有工业领域,包括过程工业、制造工业、混合工业以及工业所涉及的所有制造和过程自动化产品。作为先进的过程控制系统,SIMATIC PCS7 形成了一个带有典型过程组态特征的全集成系统。

3.1 PCS7 软件简介

SIMATIC Manager 是一种用于工程工具以及所有过程控制系统 SIMATIC PCS 7 工程任务的组态界面的集成平台,所有 SIMATIC PCS 7 项目都可在此进行管理和归档。SIMATIC Manager 提供有组态自动化系统、I/O 以及通信网络的工具,并可以从电子产品目录中进行必要硬件的选型。该工具可通过 3 个视图在编制工厂项目的同时,可以支持不同的任务:

- Component view(部件视图)用于组态自动化系统部件、总线部件或过程 I/O 等;
- Plant view(工厂视图)用于工厂的层级结构化;
- Process object view(过程对象视图)作为所有测量点/过程对象的开发环境。

Plant View 工厂视图(工厂层级)用于根据工艺流程图结构化和显示一个项目。使用按工艺划分的项目结构,可以实现快速的工艺查找。从工艺层级,也可以进入操作员站区和过程图像层级,这也是工厂层级识别过程对象的基础。并可自动生成工厂单元概览,其中可定位和管理所有可操作和可显示的 CFC 功能块。

使用 SIMATIC Manager 的 Process Object View(过程对象视图),可以通览测量点。它可以树形结构显示工厂的工艺层次,以表格形式显示所有测量点和过程对象(一般数据、参数、信号、报文、过程图像对象和测量值归档)。所有工厂层次中选中的对象都可以显示在表格中,以便非常方便地使用编辑、过滤、更换、导入和导出功能进行处理。

包含过程控制应用的库。在库中可以对预组态和测试的功能块、面板及符号进行管理。并成为自动化解决方案图形化组态的基本元素。使用这些库元素,可以显著降低工程造价和项目成本。其中提供有丰富的功能块,从简单的逻辑和驱动功能块,到集成有报警和 HMI 功能的工艺块(例如 PID 控制器、电机或阀门),直到用于根据 PROFIBUS PA 行规 3.0 PROFIBUS 现场设备的集成功能块(包括过程值状态的标准化评价)。

多项目工程与组态。使用多项目工程工具,可以将一个大型复杂的项目按工艺流程分为几个子项目,以便能够同时由几个小组进行项目作业。为此,在 SIMATIC Manager 中定义有一个“多项目”,每个项目都可随时插入“多项目”中,或从中删除。同样,也可以对项目进行合并和划分(Branch & Merge)。从属于“多项目”的子项目可以保存在一个中央服务器上,也可以保存在本地工程师站中进行编辑。由此,不会由于网络访问而影响工程与组态的性能。

划分和合并(Branch & Merge)是一种多项目工程功能,可以从工艺角度,对项目进行划分和组合。功能图或工厂单元都可复制到另外一个项目中,并进一步进行编辑。原对象中同名的功能图在使用 Branch & Merge 进行项目组合时被覆盖。非针对项目的连接在划分时,一般用于互锁,将成为文字上的连接。在项目合并时,这些文字连接包括由用户自己输入的连接都可通过按键来关闭。

顺序功能图(SFC)用于图形化组态非连续生产流程的顺序控制系统。每个 SFC 图都有输入和输出,用于状态信息的控制。根据需要,SFC 图可以直接在 CFC 中进行定位和连接。所需要的 CFC 块接口只需简单的操作即可进行选择,并与步链中的步(step)和转移条件(transition)连接。使用 ISA S88 状态管理系统,根据 SFC 图,可以组态最多 8 个单独的顺序链,用于单 SFC 中的特殊处理,例如用于 HOLDING(保持)、ABORTING(异常中止)的状态,用于 SAFE STATE(安全状态)或不同的运行模式。SFC 组态可以通过用户友好的编辑功能以及功能强大的测试和 IBS 功能进行。

连续功能图(CFC)用于图形化组态连续自动化功能。通过功能强大的自动路由和集成的 HMI 消息组态功能,可以对预定义的功能块进行定位、组态和连接。在建立一个新 CFC 功能图时,会建立一个带有该图名的流程组。该流程组可以自动插入所有安装在该图中的块中。通过转换,可以对流程顺序进行优化。使用算法首先计算最佳的块顺序,然后是流程组的顺序。

SFC 类可以像常用功能块一样,在 CFC 图中进行定位、组态和连接。因此其功能非常灵活,只需更改原有块,即可自动进行其他块的相应更改。

自动块连接。“生成块驱动器”功能可以自动生成和定位诊断块,用于自动化系统 S7-400 的输入输出部件,该自动化系统的分布式 I/O 采用 ET200M、ET200iS、ET200S、ET200X,以及根据 PROFIBUS PA 行规 3.0 的 PROFIBUS 现场设备和采用 HART 通

信的现场设备。根据符号寻址,使用向导进行块连接。

图形化设计器和面板设计器。操作员站的工程设计数据使用 SIMATIC Manager 管理。所有用于操作和监控的相关测量数据,例如报警信息和变量都可在定义自动化功能的同时生成。对于设备绘制器的图形化生成,可以使用功能强大的图形化设计器。除了标准图形块(面板)以外,还可以自行组态用于测量点或设备部件的监控和操作面板。使用面板设计器,可以快速而简单地生成这种面向客户的面板。

OS 过程控制操作界面,带有可移动的窗口。操作员站是过程控制系统 SIMATIC PCS7 的人机界面,用于用户过程的窗口。操作员站的架构非常灵活,可以适配于不同的工厂规格和客户要求。由此可实现单站系统以及客户机/服务器架构的多站系统的完美协同。操作员站的系统软件根据过程对象(PO)的数量有多种选择。

- 每个 OS 单站可有 250、2000、3000 或 5000 个 PO;
- 每个 OS 服务器可有 250、2000、3000、5000 或 8500 个 PO。

为满足更高要求或系统扩展,过程对象的数量随时可以通过附加的 PowerPack 增加。

OS 的优点如下:

- 使用可扩展的硬件和软件部件,灵活、模块化的设计,可以用于单站和多站系统;
- 功能强大的操作员站基于安装有 Microsoft Windows 2000 的标准 PC 技术,可以用于办公和工业环境;
- 客户机/服务器多用户系统,最多可有 12 个 OS 服务器/服务器对,每个可针对 5000 个过程对象,以及每个服务器/服务器对最多可有 32 个 OS 客户机;
- 基于 Microsoft SQL 服务器的功能强大的归档系统,具有循环归档功能和集成归档备份功能,使用归档服务器可以进行选择;
- OS Health Check,用于监控重要的服务器应用程序;
- 在线修改,不会影响正在运行的程序,通过有选择性地加载冗余服务器,可以进行在线测试;
- 优化的 AS/OS 通信,AS 应答周期为 500 ms,只有在数据变化之后才进行数据传输,抑制抖动报警;
- 用户界面友好的过程控制,较高的运行安全性,采用多屏幕技术;
- 通过在报警信息中组合状态或模拟值,扩展的状态显示;
- 报警优先级作为附加属性,用于筛选重要的报警信息;
- 集中用户管理,访问控制和电子签名;
- 对系统总线上所连接的下位系统进行寿命周期监控;
- 基于 UTC(通用时间同步)的系统范围内的时间同步功能。

3.2 PCS7 系统组态

西门子 SIMATIC PCS7 系统为工业自动化和控制提供了很大范围的硬件、软件、组态、配置和诊断工具。一个典型的 PCS7 系统包括工程师站(ES)、操作员站(OS)、自动化站(AS)。使用 PCS7 软件首先需要进行如下组态：

- 硬件组态：在 ES 站中对 AS 站进行硬件配置识别；
- PC 站组态：建立 ES、OS 和 AS 之间的通信连接，并使用 WiCC 组态软件监控 AS 站及其现场运行状况；
- 网络组态：打通 AS 和 OS、ES 之间的网络连接。
- 以下简要说明 ES、AS、OS 三个站的功能及其作用。

1. ES——工程师站

PCS7 项目是在 PCS7 工程师站上设计的，工程师站安装有 PCS7 组态工具，可以和自动化站和操作员站进行通信。在 ES 中组态 PCS7 项目可以分成 2 个阶段：AS engineering(AS 组态)和 OS engineering(OS 组态)。AS 组态包括工厂层级、功能块、CFC(Sequential Function Chart, 连续功能图)、SFC(Sequential Control Chart, 顺序功能图)和硬件及通信组件配置的设计。使用“Compiling OS(编译 OS)”功能，可以自动得到 OS 组态的项目数据。OS 组态是设计操作功能和图形。

2. AS——自动化站

一个自动化站可由电源(PS)、中央处理单元(CPU)、通信处理器(CP)和输入、输出模块组成。CPU 处理操作系统和程序。PCS7 CPU 是从西门子 S7 400 CPU 系列中选择的。它们通过 System/Plant Bus(系统/工厂总线)与 ES 和/或 OS 服务器进行通信。AS 也有一个通信端口通过 Profibus DP 与现场设备进行通信。如图 3.1 是一个连接到分布式 I/O 的自动化系统典型例子。

“Block(块)”在 PCS7 中是一个很重要的概念。表 3.1 是用于系统中的块汇总表。

3. OS——操作员站

OS 可以是指 OS 服务器、OS 客户机、OS 项目。“服务器”通常是指物理机器。一个 OS 项目可以是 Single-user(单用户)项目、Multi-user(多用户)项目或者 Client(客户机)项目。“单用户或多用户系统”是指一个 OS 项目的类型。

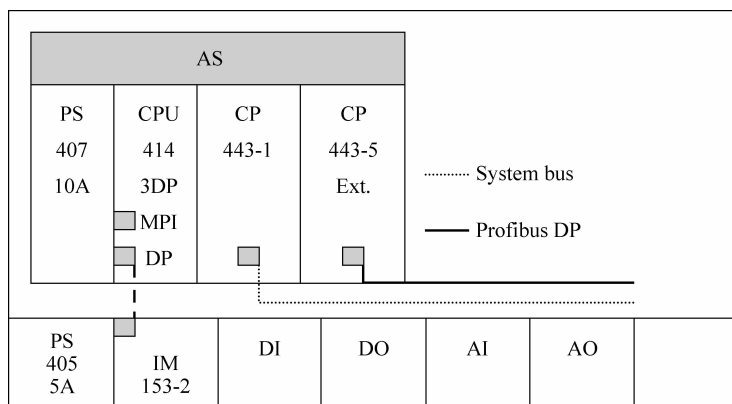


图 3.1 一个自动化站

表 3.1 PCS7 中的块

块	功 能 简 述
组织块(OB)	OB 确定用户程序的结构
系统功能块(SFB)和系统功能调用(SFC)	SFB 和 SFC 集成在 S7 CPU 中,允许访问一些系统功能
功能块(FB)	FB 是带有一个“memory”(存储器)来存储静态变量的块,用户可以设计自己的 FB
功能调用(FC)	FC 包含频繁使用的例行程序,FC 没有存储器
背景数据块(instance DB)	当调用一个 FB/SFB 时,背景数据块与块相关联,它们在编译期间自动建立
数据块(DB)	DB 是存储用户数据的数据区域,除了赋值给功能块的数据外,共享数据可以被任何块定义和使用

OS 组态是在一个 ES 中实施的,它包含了下列功能:

- 图形对象的设计(按钮、幻灯、趋势、面板等)。
- SFV: Sequential Function chart Visualisation(SFC 可视化),用于在 OS 运行系统中显示 SFC。
- 数据归档(变量和消息)以及长期数据存储的设计。
- 报表的设计。系统和过程数据的打印输出。
- 用户管理。针对不同的操作任务,分配和控制用户访问权限。
- 冗余。配置与主服务器连接的第 2 个 OS 服务器。如果两个服务器之一发生故障,第 2 个服务器即承担整个系统的控制。故障服务器恢复工作后,所有消息和过程文档的内容被复制和同步。

- 时间同步。一个 OS 在运行时间里作为时间主站,对连接到系统总线和终端总线的所有其他 OS 和 AS,以当前时间进行时间同步控制。
- 设备状态监视。设备状态监视用于持续地监视独立系统(OS 和 AS),并在运行系统中以屏幕显示的方式显示其结果。
- 与其他应用软件的连接。PCS7 OS 为用户的解决方案提供开放式接口。这样,就可以把 PCS7 OS 集成于复杂的、全公司范围内的自动化解决方案中。

本书以 PCS7 软件 V6.1 版本的组态过程为例,详细的组态步骤参见 PCS7 组态实验。

3.3 PCS7 编程

STEP7 标准软件包配置了梯形图 LAD、语句表(即 IEC1131-3 中的指令表)STL 和功能块图 FBD 三种基本编程语言,通常它们在 STEP7 中可以互相转换。此外,STEP7 还有多种编程语言作为可选软件包,如 CFC、SCL(西门子中的结构文本)、S7 Graph 和 S7 HiGraph。这些编程语言中,LAD、FBD 和 S7 Graph 为图形语言,STL、SCL 和 S7 HiGraph 为文字语言,CFC 则是一种结构块控制程序流程图。PCS7 软件中常用的编程语言为 SCL、CFC 和 SFC 三种,以下详细说明。

3.3.1 使用 SCL 编制程序

SIMATIC 为 S7-300/400 提供结构化控制语言 SCL(Structure Control Language),使得 PLC 编程更加容易。S7 SCL 编程语言符合 IEC1131-3 标准定义的文本高级语言 ST(Structured Text,结构化文本)PLC 开放标准。S7 SCL 是一种像 PASCAL 的高级语言,用于在 SIMATIC S7 中 PLC 的编程。S7 SCL 补充和扩展了 STEP 7 编程软件及其中的梯形图、功能块和语句表的编程语言。

S7 SCL 集成在 STEP 7 中,支持 STEP 7 的块概念,可以应用 S7 SCL 创建 OB、FC、FB、DB 和 UDT 等 STEP 7 的块。在一个 S7 程序中,应用 S7 SCL 编制的块可以和其他 STEP 7 编程语言编制的块结合使用,互相调用;同时,S7 SCL 块也可以保存在库中,用于其他语言的编程。

由于 S7 SCL 程序为 ASCII 码的文本文件,容易导入、导出。另外,S7 SCL 块可以编译为 STEP 7 语句表(STL)编程语言,但是如果 S7 SCL 的块保存在 STL 中,则不能再由 S7 SCL 进行编辑。

1. S7 SCL 的特点

S7 SCL 提供强大的开放环境,其开放环境包含的组件如下:

(1) 编辑器,用于编写包含 FC、FB、OB、DB 和 UDT 等程序,程序员编程时有强大的函数支持;

(2) 批处理编译器,用于编译程序,生成 MC7 机器码。生成的 MC7 机器码可以运行在 CPU 314 以上的所有 S7-300/400CPU 中;

(3) 调试器,用于在源程序中寻找存在的逻辑编程错误。

由于每个组件都运行在 Windows 环境下,每个单独的组件都很简单、方便,可以利用 Windows 操作系统的优势。

S7 SCL 具有高级编程语言的所有优势,同时还具有许多特性,有助于结构化编程,例如:

(1) 具有块库,库中有丰富的预定义块,例如,系统函数、IEC 函数、转换函数等。

(2) S7 SCL 编辑器提供了多种编程模板:块(FB、DB 等)模板及其调用;块注释、参数和常量模板;控制结构模板(IF,CASE,FOR,WHILE,REPEAT 等)等。

(3) 高级编程语言元素,可以简单、快捷地创建语言结构,且不易出错。

另外,全符号化编程、注释、基本的和预定义数据类型、交叉参考显示、输入程序自动缩进,以及不同语言元素设置不同的语法颜色等特性,使得程序更加清晰易读。

S7 SCL 高级语言层面上的调试器有如下功能:连续监视程序执行,分别设置断点逐步监视,进入程序内部进行调试。

2. SCL 中的功能块

SCL 功能块的结构如图 3.2 所示。

1) 功能块块头

图 3.3 中列举了功能块块头所包含的内容。



图 3.2 SCL 中功能块的结构

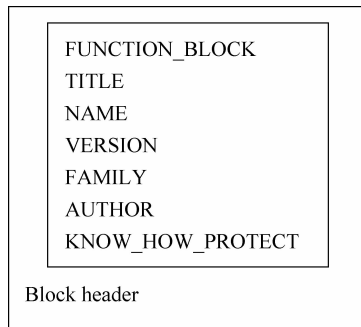


图 3.3 功能块块头

以下详细说明功能块块头中各部分的作用：

(1) FUNCTION_BLOCK

FUNCTION_BLOCK 定义了功能块的序号或者符号名称。如果它是一个符号,例如 SIM_VAL,那么,在 SCL 代码中编译这个功能块之前,必须在符号表中给这个名称分配一个序号(例如 FB501)。当把此功能块拖放到 CFC 上时,这个符号将出现在每个调用功能块的实例上,如图 3.4 所示。

将FUNCTION_BLOCK SIM_VAL
名称改为FB501

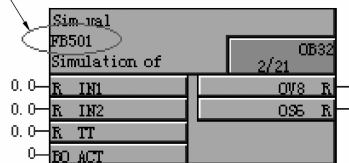


图 3.4 块类型名称

(2) 标题 TITLE

在 PCS7 中不使用这个信息,然而在 SIMATIC Manager 中,这个信息将显示在功能块对象属性的注释 comment 区域中。建议输入和 FUNCTION_BLOCK 一样的名称,如图 3.5 所示。当在 Block 文件夹下右击功能块通过菜单 Object Properties 将会调出功能块的属性窗口。

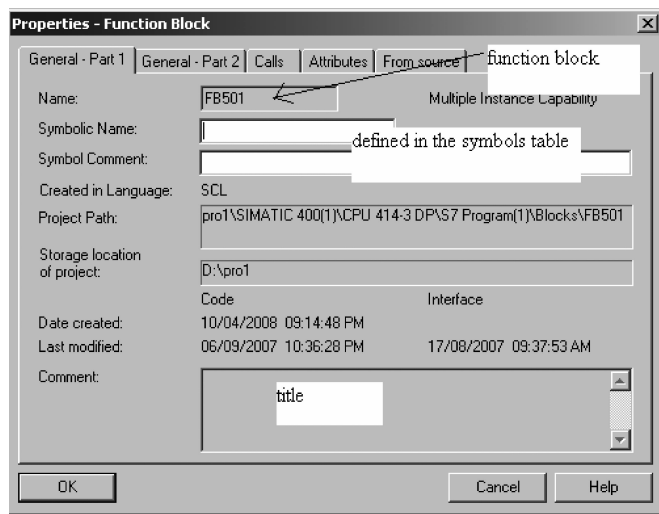


图 3.5 功能块标题

(3) 名称 NAME

在此处输入和 FUNCTION_BLOCK 一样的名称。如果要使用在线帮助,这个名称(连同 FAMILY)将构成在在线帮助系统中查找这个功能块帮助文本位置的关键词。

(4) 版本 VERSION

版本号的范围是从 0.0~15.15。

(5) 组别 FAMILY

如果要将功能块放在一个单独的库中,而且在这个库中将这此功能块安排在不同的组,则为功能块输入一个组别名称,名称不能超过 8 个字符。如图 3.6 所示。如果要使用在线帮助,FAMILY 和 NAME 将构成查找这个功能块帮助文本位置的关键词。

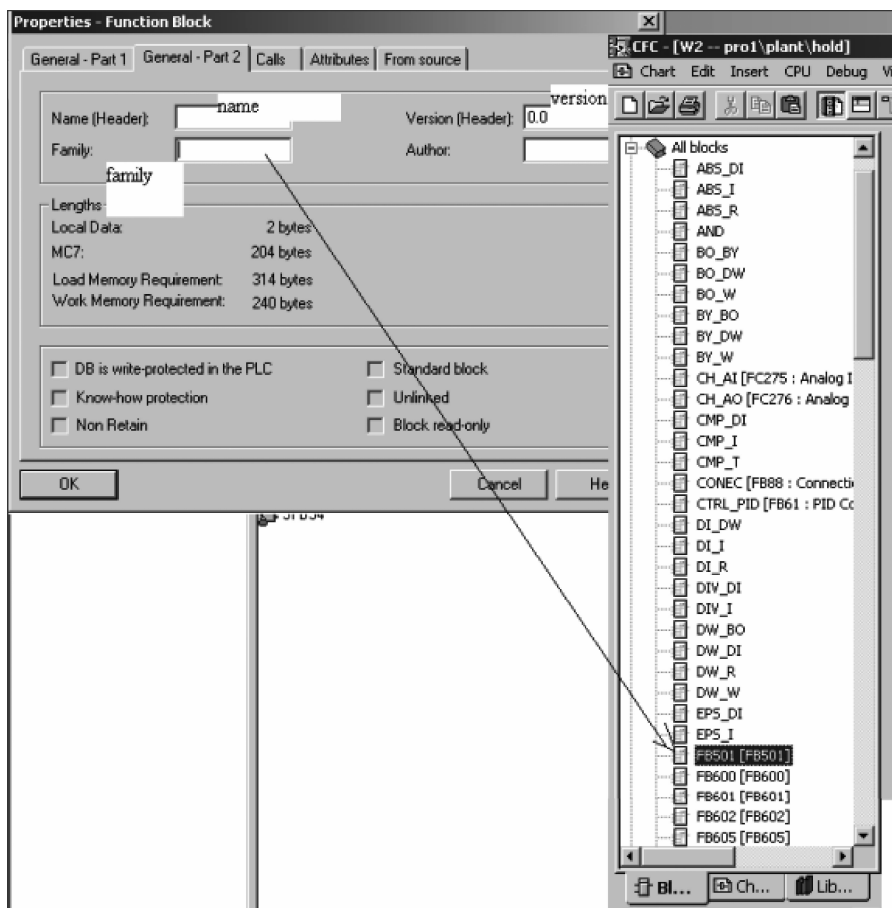


图 3.6 功能块组别

(6) 作者 AUTHOR

这个属性通常包含这个功能块作者的姓名和部门。在 PCS7 兼容的功能块中,这还可以作为以下两个用途:

- 如果要将功能块放在一起形成一个库,为这个库中所有的功能块输入一个共同的名字,名称不能超过 8 个字符。
- 如果要使用在线帮助,也会使用这个名字查找相关的帮助文件的位置。

(7) 专有技术保护 KNOW_HOW_PROTECT

如果输入了这个属性,那么,只能在 SIMATIC Manager 功能块对象属性中显示功能块的属性而不能修改属性。在这个项目之外,这个功能块本身只能由无相应源文件的 Block 编辑器打开,而不能再用 SCL 打开。

2) 功能块属性

系统属性被分为功能块系统属性和参数系统属性。功能块系统属性应用于整个功能块,参数系统属性应用于每个单独的功能块参数。系统属性是相应的应用程序之间的有关接口属性,用于控制和协调相关应用程序之间的功能。例如,只需在功能块中进行一次消息组态,就可以在其他任何地方使用,例如 OS 系统中。在消息组态,通信组态和人机界面中都会用到系统属性。

功能块属性列表如表 3.2 所列。

表 3.2 功能块属性列表

系 统 属 性	意 义	默 认 值
S7_m_c	指定能否在 OS 站中控制和监视该功能块	False
S7_tasklist	包含一个 OB 列表(例如启动和错误 OB),CFC 会将这个块装载到这些 OB 中	装载不超过一次
S7_alarm_ui	消息服务器的标识符: S7_alarm_ui := “0”标准消息对话框; S7_alarm_ui := “1”PCS7 消息对话	S7_alarm_ui := “0”
S7_tag	如果这个系统属性的值为 false,这个功能块将不进入 OS 的标签列表。这对于那些只发送消息而没有面板的功能块非常有用。如果这个系统属性不存在而且该功能块又具有系统属性 S7_m_c,这个功能块将被输入到过程变量列表中	False
S7_driver	这个属性用作信号预处理驱动块,在 SIMATIC Manager 中通过 CFC 功能 Genetate module drivers,它可以自动与相应的功能块连接	False or “chn”
S7_read_back	定义调用功能块是否具有 CFC 中的 Chart>Readback 功能,当这个系统属性的值为 false 时,不能回读调用功能块的参数	True

在 SCL 中,可以在功能块块头部分之前定义功能块属性,如图 3.7 所示。

系统属性显示在 SIMATIC Manager 下功能块对象属性的 Attributes 选项卡中,如果该功能块没有写保护,还可以在此处修改系统属性,如图 3.8 所示。图 3.7 中的例子不包含 KNOW_HOW_PROTECT。因此,在图 3.8 中是可以修改系统属性的。

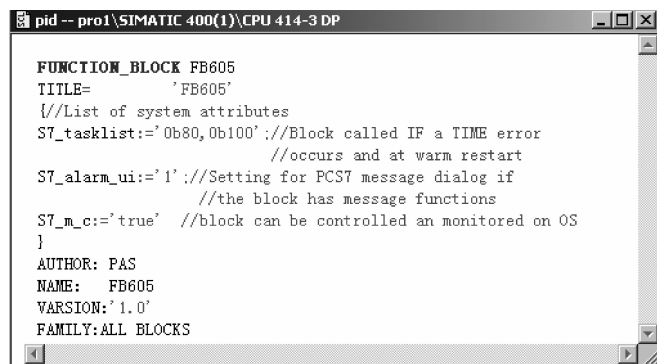


图 3.7 定义功能块属性

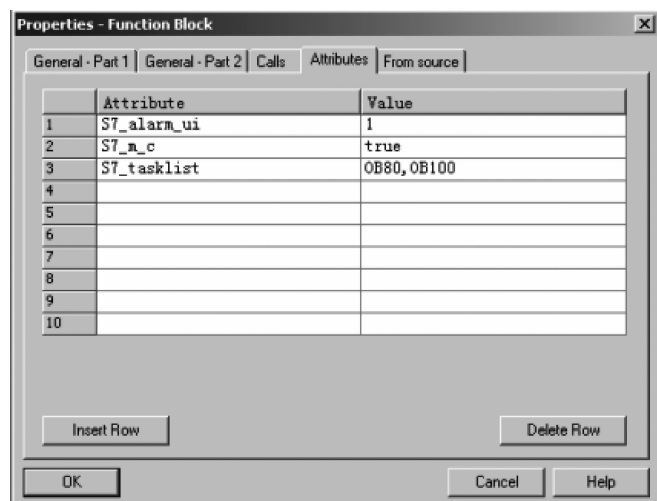


图 3.8 显示和编辑块属性

3) 变量参数声明部分

(1) 功能块参数

功能块参数(Block I/O)定义了功能块的接口,其中一些参数也应用到 OS 画面中。功能块参数分为输入(Input)参数、输出(Output)参数和输入输出(In_out)参数。输入输出参数可以通过功能块算法进行读写操作。在 VAR_INPUT 和 END_VAR 之间定义输入变量。

(2) 参数的系统属性列表

表 3.3 给出了参数属性列表。

表 3.3 参数属性列表

系 统 属 性	作 用	意 义	默认值
S7_asmpletime	时间响应	如果参数具有这个系统属性,则自动分配调用循环 OB 的循环时间。当编译 CFC 功能图时,必须选中复选框“Update sampling time”	False
S7_dynamic	CFC	如果参数具有这个系统属性,在 CFC 的测试模式下(Watch)自动注册为测试方式	False
S7_edit	CFC	这个属性决定能否在 SIMATIC Manager 的过程对象视图 Process Object View 中编辑参数	False
S7_link	CFC	这个属性决定能否在 CFC 功能图中互连参数	True
S7_param	CFC	这个属性决定能否在 CFC 功能图中设置参数值	True
S7_visible	CFC	对于一个参数如果这个系统属性被设置为 False,将不在 CFC 功能图中显示	True
S7_qc		这个参数具有质量代码	False
S7_contact		这个属性定义 SFC Type 相关的特定参数	False
S7_m_c	OCM	这个属性决定能否从 OS 控制或者监视参数	False
S7_shortcut	OCM	包含了最长为 16 个字符的参数标识符。这个名称也可以在 OS 的面板中显示	
S7_string_0	OCM	这个系统属性只与数据类型为 BOOL 的输入参数(或者输入/输出参数)有关。它包含了最长为 16 个字符的文本,可以在操作面板上作为操作员文本(例如 Open Valve)显示。当操作员选择这个功能时,这个参数被设置为 0	
S7_string_1	OCM	这个系统属性只与数据类型为 BOOL 的输入参数(或者输入/输出参数)有关。它包含了最长为 16 个字符的文本,可以在操作面板上作为操作员文本(例如 Close Valve)显示。当操作员选择这个功能时,这个参数被设置为 1	
S7_unit	OCM	这个属性包含了参数单位,最长为 16 个字符。这个单位可以在 CFC 的功能块 I/O 中显示	
S7_server	Server	这个接口参数指定一个服务器。消息服务器: S7_server := “alarm_archive”	无服务器调用
S7_a_type	Server	这个接口参数是消息类型 x 的输入消息号或者输入归档号	无服务器调用

(3) 本地变量

① 临时变量

临时变量只是在一次功能块调用的过程中有效。换言之,在每次功能块调用时,必须重新计算临时变量。在 VAR_TEMP 和 END_VAR 之间定义临时变量。

② 静态变量

相对于临时变量,静态变量在多次功能块调用的过程中一直保持它的值,除非在功能块算法中改变该变量的值。在 PCS7 兼容功能块中,如果要在功能块中调用现有功能块,无论是自己创建的还是标准的,这些变量尤为重要。在这种情况下,必须应用一个多实例功能块。通过将一个被调用功能块定义为一个静态变量来完成。在成功编译调用功能块之前,被调用功能块必须存在 S7 程序的 Block 文件夹中。如果想要使得被调用功能块的参数对外可见并且可以连接,必须将被调用功能块的参数复制到调用功能块。在 VAR 到 END_VAR 之间定义静态变量。

(4) 代码部分

SCL 的代码位于 BEGIN...END_FUNCTION_BLOCK 之间(对于功能块 FB), BEGIN...END_FUNCTION(对于功能 FC)或 BEGIN...END_ORGANIZATION_BLOCK 之间(对于组织块 OB)等。

3. 调整系统属性

调整功能块包括根据应用调整系统属性。可以修改功能块 I/O 的下列系统属性,例如:

- 参数的可见性(S7_visible);
- 参数的动态视图(S7_dynamic);
- 二进制数值的文本(S7_string_0,S7_string_1);
- 模拟量的文本(S7_unit,S7_shortcut);
- 参数的互连性(S7_link,例如对于控制器的 GAIN,TN,TV)等;
- 不能改变消息属性(S7_server,S7_a_type)。

如果要向 OS 中传送参数,属性 S7_m_c 是必需的。因此,如果更改属性 S7_m_c,对应面板的功能将受到影响(由于 OS 中的变量不再存在而造成连接丢失)。

为了调整功能块的系统属性,可以在 Block 编辑器中修改系统属性,可以通过双击功能块调出该编辑器。如果功能块被保护,将显示一个警告信息。可以忽略该信息,因为某些属性仍然可以被修改。在 Block 编辑器环境下,可以继续按照图 3.9 中的指示修改属性。

对于项目中现有的功能块和库进行调整之后,可以在其基础上创建更多的功能、功能块和功能图。建议将新建的功能块存放在主数据库中。

4. SCL 编辑器

SCL 不能在 plant view 中插入,只能在 comenent view 中插入;可以在 S7 Program 的 Sources 文件夹下插入一个新的 SCL 文件,然后双击这个文件以打开 SCL 编辑器。

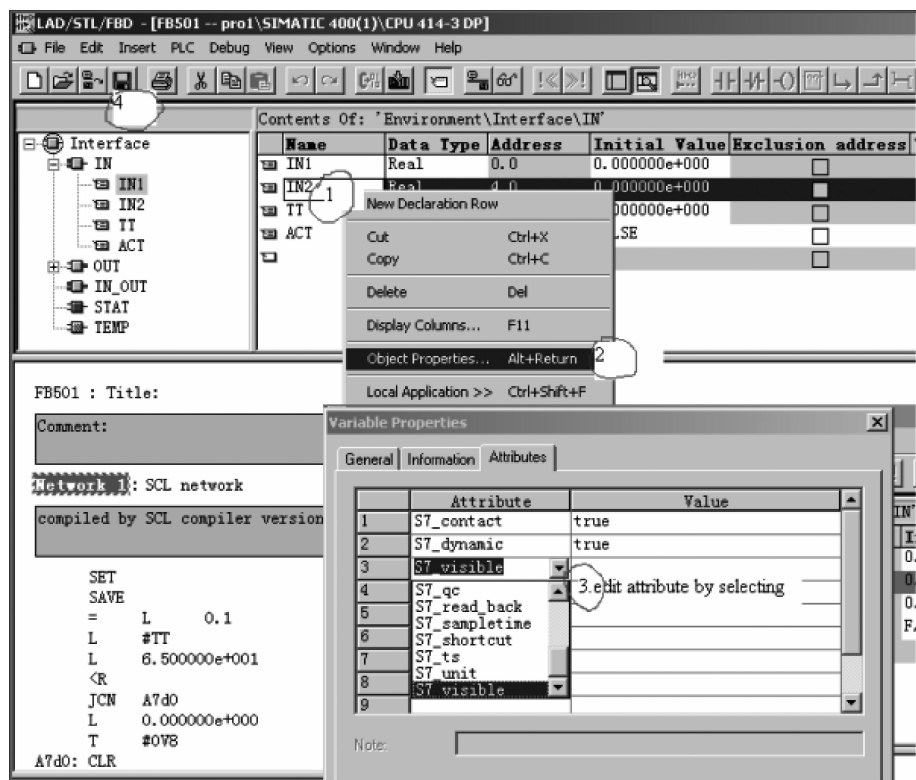


图 3.9 调整系统属性

1) 插入块模板

如图 3.10 所示,当开始使用 SCL 时,可以插入一个块模板,例如程序块(OB,FB 或者 FC),块头(注释)以及数据声明(参数)等。块模板将通过 SCL 语法指导编程。

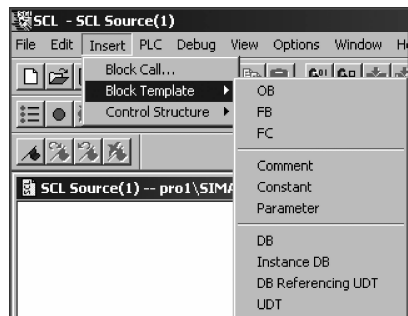


图 3.10 插入一个基本 SCL 句法

2) 在 SCL 源文件中插入调用块 Block Call

通过菜单命令 Insert→Block Call 可以方便地从 SCL 源文件中调用功能和功能块。在这个源文件中 SCL 自动地将被调用的功能块复制到 S7 程序中并插入这个功能块中。如图 3.11 所示。

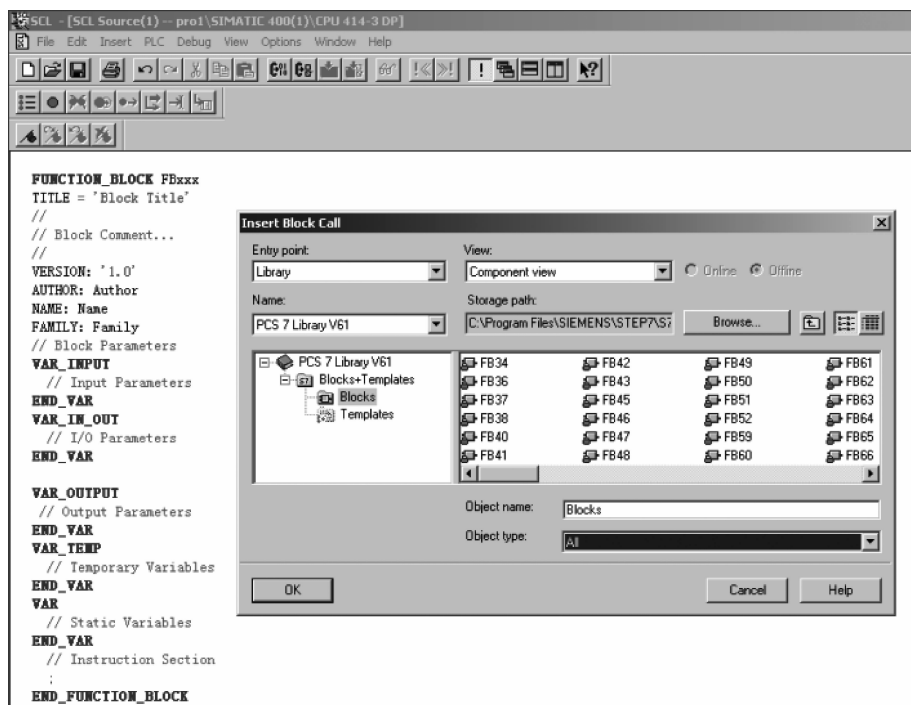


图 3.11 在 SCL 中调用功能块

3) SCL 控制语句

(1) IF 语句

下面给出了 IF...THEN 语句的一个例子。

```
IF I1.1 THEN
N := 0
SUN := 0
OK := FALSE; //Set OK flag to FALSE
ELSIF START = TRUE THEN
N := N + 1
SUM := SUM + N
ELSE
OK := FALSE
END_IF
```

(2) CASE 语句

下面给出了 CASE 语句的一个例子。

```
CASE TW OF
1:DISPLAY := OVEN_TEMP
2:DISPLAY := MOTOR_SPEED
3:DISPLAY := GROSS_TARE
QW4 := 16#0003
4..10: DISPLAY := INT_TO_DINT(TW)
QW4 := 16#0004
11,13,19: DISPLAY := 99
QW4 := 16#0005
ELSE:
DISPLAY := 0
TW_ERROR := 1
END_CASE
```

(3) FOR 语句

下面给出了 FOR 语句的一个例子。

```
FUNCTION_BLOCK FOR_EXA
VAR
INDEX: INT
IDWORD: ARRAY[1..50] OF STRING
END_VAR
BEGIN
FOR INDEX := 1 TO 50 BY 2 DO
IF IDWORD[INDEX] = 'KEY' THEN
EXIT
END_IF
END_FOR
END_FUNCTION_BLOCK
```

(4) WHILE 语句

下面给出了 WHILE 语句的一个例子。

```
FUNCTION_BLOCK WHILE_EXA
VAR
INDEX: INT
IDWORD: ARRAY[1..50] OF STRING
END_VAR
BEGIN
INDEX := 1
WHILE INDEX <= 50 AND IDWORD[INDEX] <> "KEY" DO
```

```

INDEX := INDEX + 2
END_WHILE
END_FUNCTION_BLOCK

```

5. 块中图

像在 SCL 中一样,需要为功能图的输入输出(其将变为功能块参数)和功能块定义属性。如图 3.12 所示,说明如何打开变量属性 Variable Properties 对话框给变量定义系统属性,前提是有一个定义好的功能块输入输出,如图 3.13 所示,打开对话框之后,显示出供选择的参数属性。

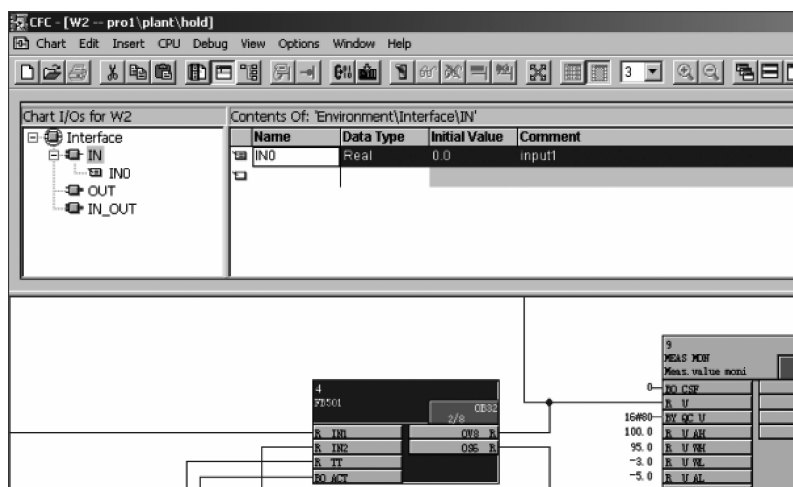


图 3.12 打开变量属性 Variable Properties 对话框

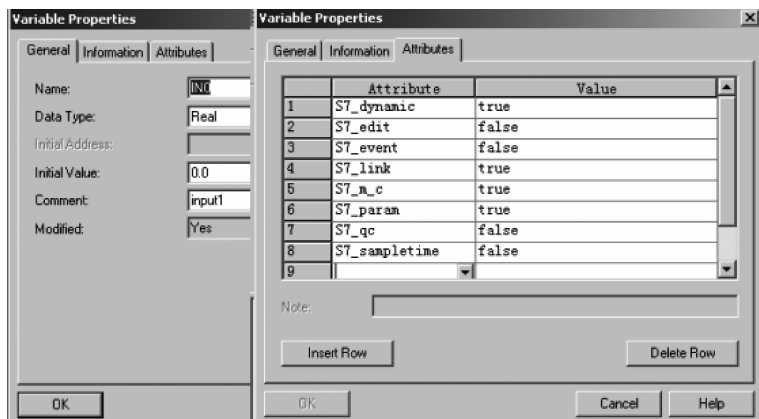


图 3.13 在变量属性 Variable Properties 对话框中定义参数属性

在选取功能图输入输出并为输入输出定义了参数属性之后,依照菜单路径选择功能: Compile→Chart as Block Type…。接着在打开的编译对话框 Compile chart as block type 中,需要定义功能块块头和功能块属性。如图 3.14 和图 3.15 所示。

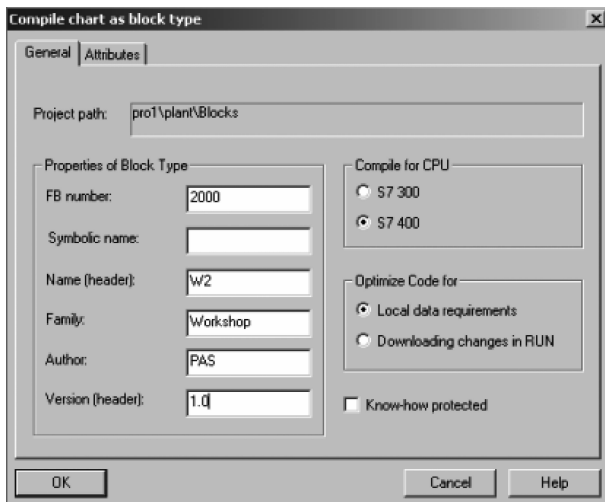


图 3.14 定义功能块块头

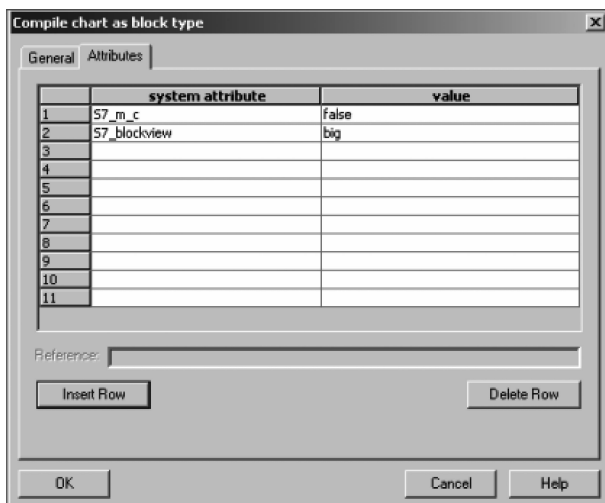


图 3.15 定义功能块属性

一旦一个功能图变成了一个功能块,它将成为一个独立的功能块。可以将其插入到 CFC 中,或者在其他项目中使用该功能块等。

3.3.2 使用 CFC 编制程序

1. CFC 编程语言概述

连续功能图(Continuous Function Chart,CFC)是一种简洁的图形组态工具,主要用于连续过程自动化控制任务的组态,可以无缝集成到 STEP7 软件中。CFC 组态以系统配置的已编程的块为基础,这些块以库的形式保存,每个块都可以重复使用,用户可根据实际工艺要求设置不同的块参数,避免大量重复性的工作,从而提高工程效率。

每个 CFC 图表文件由多个分区构成(可以任意添加新的分区,由字母 A~Z 任意排列),每个分区由 6 个表单组成。功能块之间的互连可以在相同或不同的 CFC 图表文件之间的相同或不同分区的表单上进行,连接标记由系统自动标出。因此,采用 CFC 可以完成很复杂的大型控制任务。

CFC 还具有图中图功能,即将同类的控制功能或方案组态成模板 CFC 图表,用户可以直接在其他 CFC 图表中插入模板 CFC 图表,直接对模板图表进行连接,可以节省大量的组态时间。

在 CFC 编辑器中,使用图形化工具从库中选择系统已编程的块,拖动相应块到图表(用户编程区)中,然后使用鼠标将其相互连系起来即可完成程序编制。用户无须了解语法等技术细节或系统资源的分配情况,而只需专注于项目的规划组态。

块具有运行时间属性,系统对运行时间属性有默认的设置,但是用户可以单独定义每个块的运行时间属性。

用户创建需要的所有功能,通过单击就可以生成可执行的机器代码,将代码下载到 PLC,就可以应用 CFC 的测试功能对代码进行测试。

1) CFC 图表

图表是 CFC 编辑器的基本工作单元,每个图表在 CPU 中都有唯一的名字。每个图表可以包含由 A~Z 命名的多达 26 个图表分区,新创建的图表只包含一个图表分区,如图 3.16 所示,用户可以根据需要添加图表分区。

图表文件每个分区包含 6 个表单,排成两列,每列 3 个表单,如图 3.16 所示。每个表单包含中心工作区和页栏。在工作区,用户可以放置块、互连块或图表等。页栏在中心工作区的左右两边,包括当前活动表单,不包含到互连对象(块/图表块 I/O、共享地址和运行时间组等)的引用。

可以将一个 CFC 图表插入到另一个 CFC 图表中,即图中图功能,称为图表嵌套。插入的图表可以像其他图表一样打

Sheet1		Sheet4	
Sheet2		Sheet5	
Sheet3		Sheet6	

图 3.16 CFC 图表

开、编辑和单独修改。也可以将一个图表封装起来,作为“图表块”提供图表 I/O。对于每个图表,用户可以决定哪些块 I/O 引出作为图表 I/O。

2) CFC 中的块

用户可以自定义图表中不同组件的显示,例如对于块可以选择类型名称或块的 FB/FC 标识等是否显示在“块头”中,是否显示 I/O 的数据类型,以及名称或注释等;另外,块及页栏的宽度也可以由用户设置。

(1) 块类型

每个块都有一个表明算法、类型名称和数据接口(输入/输出参数)的类型定义。其中,类型名称是功能缩写或者是功能的首字母,例如:

CTUD(COUNT UP and DOWN)功能为边沿控制的加减计数器;

COUNT_P 功能为二进制信号正跳沿加减(根据设置的不同)计数器;

ADD_R 功能为对输入值进行加运算,输出运算结果。

类型定义还指定输入/输出参数的数据类型,输入/输出参数即块的输入和输出。数据类型指的是输入/输出取值的类型和范围,例如:

BOOL: 布尔类型,只能取 0 或 1;

STRING: 字符串类型,字符串可以作为取值。

创建算法块时必须声明 FB、FC 或基本操作块(Basic Operation, BOP)。BOP 与 FC 类似,同样没有存储区,BOP 是 CFC 编程组件,用作 AND 和 OR 等简单功能。

(2) 块实例

插入一个块到 CFC 图表即创建了该块的实例。所谓实例,即块类型的一个副本,其功能和所定义的块类型相同,但是不同实例可以有不同的块参数,更改实例的参数不会对块类型的功能定义产生影响。对于每个特定的块类型,可以创建任意数量的块实例。

应用实例的概念对块类型进行集中修改,可以自动在各个相应的块实例中同步更新。

(3) 复合块(复实例块)

可以将不同的子功能放在一起,这些子功能本身就是块,放在一起形成复合块。例如,控制器块可以包含消息块和操作员控制块,而操作员控制块本身也可以是复合块。

在 CFC 编辑器中,通过在图表中使块(功能)互连,分配适当的参数创建复合块,则可以将其作为新块类型进行编译。

(4) 输入个数可变的块

对于 CFC 中的某些算法块,其输入个数是可变的,用户可以在 CFC 图表中对其进行更改,如 AND 块的输入个数就是可变的,可以按要求修改为两输入、三输入等。

(5) 块系列

根据功能属性的不同将块进行编组,形成块系列。创建块系列的同时,每个块即得到一个系列标识符。CFC 选项包安装后,默认的块系列有: BIT_LGC、COPARE、

CONVERT、FLIPFLOP、MATH_FP、MATH_INT、MULTIPLX、PMC_FUNC、SHIFT 和 WRD_LGC 等。

(6) 组织块

在 CFC 中,不能插入也不能编辑组织块(OB)。调用运行时间顺序编辑器后,CFC 图表中算法块按执行的顺序显示在 OB 块中。

(7) 重叠块

块插入或移动到图表中,若没有足够的空间时即形成重叠块,这种情况下,新的块与其他对象全部或部分重叠。

重叠块显示为浅灰色,只有移动到表单的空白位置,重叠块的 I/O 才可见。重叠块与其他算法块之间的互连和入口也是不可见的,但是都保存在表单中。

3) 目录

目录是 CFC 编辑器中的一个窗口,窗口中包含可用算法块的树形列表,如图 3.17 所示。通过鼠标拖放可以从目录中插入块和图表到 CFC 图表中。单击工具栏上的 Catalog 按钮可以打开或关闭目录窗口。

单击图 3.17 窗口下方的选项卡或按钮可以实现不同列表的显示切换,其功能如表 3.4 所列。

表 3.4 目录窗口选项卡和按钮

	块列表: 可以插入到图表已存在的(导入的)块、All Blocks 下面按字母顺序排列的 BOP 和当前 S7 程序的块等
	库: 从库中可以选取新块插入到图表中,SIMATIC 管理器中已有库在此处显示
	图表: 当前 S7 程序中的 CFC 图表,用户可以插入(复制)或打开
	未入选的块: 不显示在 CFC 图表中的当前程序中的块。这些块所分配的 CFC 图表也在此显示。只有存在未放置的块时,该目录才显示
	搜索块或图表: 在文本框中输入块或图表的名称,单击此按钮进行搜索,会打开对象所在的文件夹(如块系列所在的文件夹)。搜索时只需输入对象名称的前几个字母,找到名称中有这些字母的对象时,搜索结束。搜索时,会显示一个带进度条的对话框,如果搜索时间太长,可以停止搜索
	关闭文件夹: 单击此按钮可以关闭目录中所有打开的文件夹

4) 操作员监控

过程运行时,现场站产生的消息通常需要传送到操作员监控系统(PCS7)中。通过 CFC 中的消息组态,用户可以直接在块中组态基于事件的消息、文本和属性等。组态现场站时,需要在监控站上创建现场站和监控站通信,即监控所需的必要数据,可以在 SIMATIC 管理器中传递该数据到监控站。

当插入一个带消息特性的块到 CFC 图表时,一个消息也自动创建。这个块拥有默认属性和消息文本的消息结构。也就是说,当时间出现时,PLC 即发送消息,而对于用户来说不需要任何额外的组态,可以将动态的值输入到消息文本中。

可以通过消息组态功能编辑单个块实例的属性,如消息类、消息类型和消息文本,如果“消息”按钮没有激活,则该块无法发送消息。

PCS7 所有的消息块都具备一个操作员监控属性。可以对那些允许操作员输入的块文本进行选择和编辑,通过块属性的“Operator Control and Monitoring...”按钮启动相应对话框,CFC 块实例的 Operator Control and Monitoring 属性也可以进行修改,需要在块对象属性中选择或不选择 Operator Control and Monitoring 选项。

组态完消息后,通过编辑将现场站和监控站通信所需的数据传送到监控站,这些数据传送到一个或多个目标操作员站并通过图形或面板使用。要实现传送,必须安装 AS-OS Engineering 现场站监控站工程软件包。

5) CFC 组态步骤

应用 CFC 组态的一般步骤如下:

- ① 创建工程结构:使用 SIMATIC 管理器在 S7 程序下创建 CFC 图表文件夹;
- ② 创建块,可以来自库中、其他程序或自定义类型的块,并将其导入 CFC 中;
- ③ 在图表中插入块实例;
- ④ 组态块参数并互连:可以指定块的输入/输出参数或者与其他块或共享地址互连;
- ⑤ 调整运行时间参数;
- ⑥ 编译 CFC 图表;
- ⑦ 下载 CFC 程序;
- ⑧ 测试 CFC 程序。

2. 使用 CFC 编辑器

用户首先应用 SIMATIC 管理器创建一个工程 CFC_EXE_1,接着执行菜单命令 Insert→S7 Software→CFC,或选中图表文件夹,右击,在弹出的快捷菜单中选择 Insert→

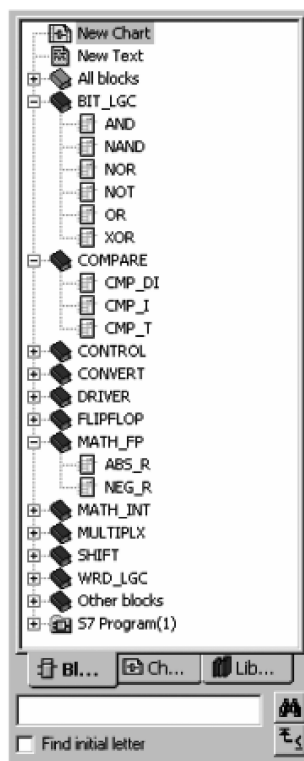


图 3.17 CFC 中的目录

Object→CFC 菜单命令,这样就创建了一个图表 CFC1,并重命名为 Control,如图 3.18 所示。图表名称在图表文件夹中由系统检测是唯一的,最长可以有 22 个字符。

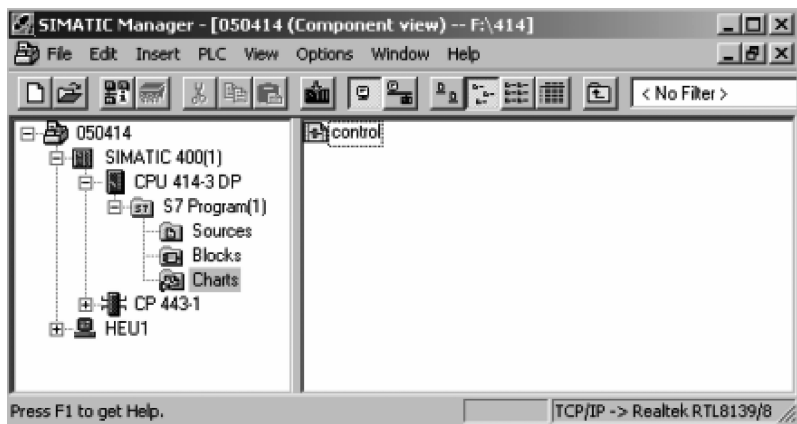


图 3.18 新建图表 Control

双击 Control 打开 CFC 编辑器,即可对此图表进行编辑,可以在图表中添加、删除块,创建块类型、图表块等。

1) 操作图表

(1) 打开图表

有以下几种方式打开 S7 程序中的 CFC 图表:

① 使用 SIMATIC 管理器选择工程和工程文件夹,接着打开图表文件夹,双击打开所需的图表,同时打开 CFC 编辑器;

② 在 CFC 编辑器中,Chart 菜单下总是显示最后编辑的 4 个图表。选择其中的一个即可打开相应的图表;

③ 执行 Chart→Open 菜单命令打开最近编辑列表中未显示的图表;接着选择对话框中的工程文件夹,选择程序文件夹和 CFC 对象类型,双击打开选中的图表;

④ 单击 CFC 图表编辑窗口工具栏的“打开”按钮,可以打开 Open 对话框,从中打开相应的图表。

(2) 创建“图表块”

具有图表块 I/O 的图表称为“图表块”,即把某个图表当做块,指定其中希望的块 I/O 作为图表块的 I/O,则生成的图表对象可以作为普通块放置到不同图表中(图表嵌套),与其他图表或块互连,或者编译为新的块类型。在图表块 I/O 编辑窗口指定图表块 I/O,即创建了“图表块”。

(3) 嵌套图表

可以将 CFC 图表插入到其他的 CFC 图表的中心工作区,创建层次结构,称为图表的

嵌套。嵌套到其他图表的图表如同其他简单图表一样,可以在编辑模式下打开和修改,在测试模式下运行和监视。

包含其他图表的图表称为顶层图表,也称为父图表,它包含的图表称为子图表。在 SIMATIC 管理器中,只能在图表文件夹中看到顶层图表。图表最大嵌套深度是 8 层(顶层图表+7 层嵌套的图表)。

从目录窗口拖放子图表到当前图表中可以继续编辑。若执行菜单命令 Insert→New Chart 插入子图标,则系统在父图表中(表单视图中只在当前的表单中)自动搜索空白位置,新建子图表放在空白位置。

图表文件夹中所有图表在图表目录中以树形结构显示,可以像其他简单块一样使用这些图表。选择目录中的图表,如果包括嵌套图表将一同进行复制。按住 Shift 键拖动鼠标,可以移动父图表到图表中,但是父图表没有被复制,只是移动了,此时目录窗口中,父图表不再位于先前的位置,而是显示在当前活动图表的等级中。

在父图表中选择子图表,右击,在弹出的菜单中选择 Edit→Open 命令可以打开嵌套的子图表。双击图表主体(除块头或 I/O 之外)的空白区域也可以打开嵌套的图表;再次双击,可以打开嵌套最深的一级图表。选中嵌套的图表,右击,在弹出的菜单中选择 Char→Open 命令按等级移动图表。

2) 操作块

CFC 软件包包含了基本的块资源,可以从库或其他工程中添加更多的块。

(1) 导入块和新版本

CFC 中只能应用已知的块,通过导入块到块文件夹,使得块对于 CFC 是“已知的”。

从目录窗口式导入或显式地通过菜单命令 options→Block Types 导入块。如果要从用户程序(块文件夹)导入许多块,可通过菜单命令 Options→BlockTypes 打开 Block Types 对话框,如图 3.19 所示,在此可以修改块文件夹中的块(类型修改)、修改图表文件夹中同名的块,以及需要在图表文件夹中更新的块等。

从 Offline block folder 框中选中所需的块,然后通过拖动鼠标或使用向右箭头将其导入到图表文件夹,如果图表文件夹中已经存在该块,则将显示提示信息。

如果导入的块类型对象名称已经存在(块编号,如 FB61),但是具有不同的符号名,则将打开一个对话框,在该对话框中可以对块进行重命名。

修改块类型后,如果要使用修改后的块类型,则必须将新的块类型导入到图表文件夹。CFC 图表中“旧”块类型的实例能够采用修改后的块类型,即块类型的集中修改。导入新版本的块步骤如下:

① 从 S7 程序或库中插入修改后的块类型到图表中;

② 打开 Block Types 对话框,从源文件列表(离线块文件夹)中选择导入的相应块,然后将其拖动至“图表文件夹”列表。

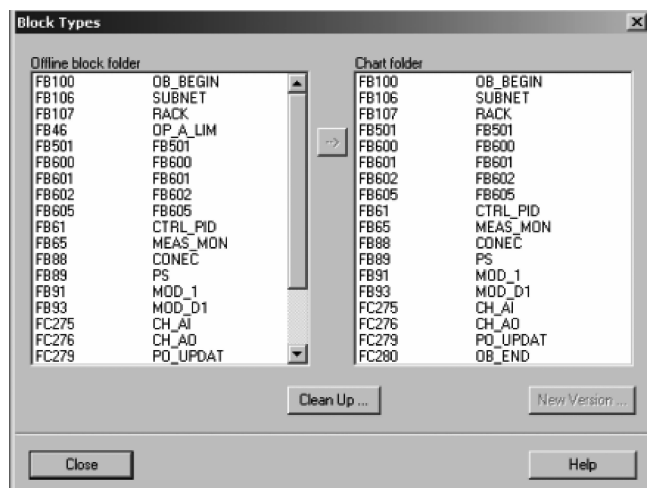


图 3.19 Block Types 对话框

如果图表文件夹中存在名称相同但版本不同的块类型,则将弹出 Import New Version 对话框进行提示,并显示两个块的版本信息。如果对 Do you still want to use the new version of the block types? 回答为 Yes,则所有相应的块实例将会更新,如果回答为 No,则不进行更新。

(2) 修改块类型

如果 CFC 图表中的块 I/O 或块类型的系统属性发生了改变,所有块类型将自动更新,下面是各种改变情况下的结果。

① 块 I/O 修改

- 若添加 I/O,则同时对块实例进行修改,默认设置为系统属性。如果块大小增加后中心工作区没有足够的空间,则块成为“重叠块”。
- 删除 I/O 时则删除各块实例相应的 I/O,若删除的 I/O 与其他对象进行了互连,则互连(或 SFC 访问)也将被删除,删除的互连(或删除的 SFC 访问)记录在日志中。
- 若修改 I/O 的顺序,除考虑 I/O 的顺序外,互连、参数分配和属性都保留下来。
- 修改 I/O 的数据类型,则参数及与该 I/O 的所有互连都将丢失、删除后重新创建同名 I/O 时,有可能发生这种情况。
- 修改 I/O 的名称时,参数及与该 I/O 的所有互连都将丢失,对旧名称的引用不能由系统自动建立。删除后又重新创建 I/O 时,有可能发生这种情况。

② 系统属性修改

如果对属性进行了修改,则在块实例中不能修改的内容将作为类型修改的结果自动在实例中进行修改。

下载时不能只下载修改的部分,而是要下载整个程序;另外,由于现场站上的块与 CFC 中的组态不再匹配,因此不能再使用 Read Back Chart 功能。

CFC 图表可以直接在 WinCC 中监控,对 CFC 图表中的算法块进行修改后,对 WinCC 中的监控会产生如下影响:

- 如果对块类型进行了修改,通过压缩和编译创建有新 DB 编号的 DB。要确保在线访问得以保留,必须将数据再次输到 WinCC(OS 编译)。
- 如果用于操作员监控的块 I/O 受到影响(属性 $S7_m_c=true$),则添加了 I/O 对操作员站进行编译后,可以在 WinCC 中使用;如果删除了 I/O,则不再有 WinCC 变量标签,存在的互连也将在 WinCC 中删除;如果修改了 I/O 的名称,也要修改 WinCC 中对应的变量标签,画面元素、块图表和面板上的互连必须进行更新。

只要修改对 WinCC 产生了影响,都必须对操作员站重新编译。

(3) 操作快

① 插入块

在表单视图和预览时都可以插入块。从其他图表的块目录或库中选择合适的块类型,通过拖放插入到当前图表中。插入的图表自动复制到当前的 S7 程序,并导入到 CFC 中,同时显示在块目录中。如果放置的位置对当前的块空间不足,则插入的块将作为重叠块。重叠块的 I/O 不可见,其“块头”和主体显式为亮灰色,将其移动到空白区域,“重叠块”将正常显示。

从块目录的“S7 程序”中插入块需要注意,S7 程序中的相应块即使已经导入,对 CFC 来说也是未知的,即要从“S7 程序”中插入块到图表中,仍然需要检查该块是否已经导入。已经导入的块存在于某个块系列的树形列表中,或者如果尚未将其分配到某个系列(在块头显示),则位于“其他块”文件夹,同时按字母顺序列于“所有块”文件夹中。

从块系列中插入块到图表中不需要检查,如果块类型已经导入则推荐采用这种方式插入块。

插入块到图表中以后,CFC 分配给该块一个编号作为默认名称。每次通过复制或从目录中拖放插入新块,都将分配一个更高的空闲编号,即进行自动命名。

插入块的同时需要指定运行时间属性,如指定运行序列中块插入的位置。当前插入点显示在状态栏的右侧,状态栏上显示任务名称、图表及块名称。插入块到 CFC 图表中时,将会把块放在运行序列中所显示的块和图表之后。

每插入一个块,安装向导就自动更新,用户可以对安装向导进行自定义:选择中心工作区中的某个块(非状态栏上显示的块),然后执行菜单命令 Edit→Predecessor for Installation,该块则成为下一插入块的“前导”,进而修改运行序列中块的安装位置。

执行菜单命令 Edit→Align→Left→Right→Top→Bottom,则自动将图表中插入的块及选中的块对齐,但是对于重叠块,则不能执行该菜单命令。如果对齐会引起冲突,例如

块重叠,则 CFC 进行信息提示,保持旧的位置,这种情况下,不能完成对齐的操作。

可以在 CFC 图表中插入文本框来编辑系统信息、图表注释等。具体方法是:从块目录中拖动文本框插入图表中的空白位置;单击文本框,激活文本光标,可以输入文本信息,输入的文本可以自动换行;如果文本信息过多,超出文本框的显示范围时,文本框大小不会自动调整。要浏览全部文本信息时,需要手动调整文本框的大小。打开文本框,文本光标定位在单击的位置。

② 复制和移动块

可以将块在图表内部或图表之间进行复制或移动,也可以复制或移动几个或图表中的所有块。还可以复制块到不同的 CPU,其影响与复制图表到 CPU 相同。

复制的块插入到与原图表相同的位置,即块插入到具有相同编号的表单中。如果该位置已经有其他对象,则块将显示为重叠块。移动重叠块以便于块的显示,但重叠块同样可以在现场站中执行,不调整重叠块也不会影响块的执行。

按照运行时间属性和运行序列,复制块和从目录中插入块的作用相同,也就是说,插入的位置由“安装向导”确定。如果块安装在运行时间组中,不复制运行时间组,块名称也会尽可能保留。如果存在冲突,则在名称后面会附加一个编号以示区别。

③ 删除块

删除 CFC 图表中块的同时,会丢失与该块互连的对象。如果删除输出互连至其他块输入的块,则将显示警告信息,确定删除后,系统给未删除的块输入赋默认参数值。

(4) 设置块属性

块属性包括块的名称、注释和消息组态等,可以分别进行设置。选中某个块,右击,在弹出的快捷菜单中选择 Object Properties 命令,打开块属性设置对话框。表 3.5 中列出了属性窗口中 General 选项卡下的部分内容。

表 3.5 属性窗口 General 选项卡的内容

选 项	描 述
Name	块名称,在整个图表中是唯一的,最多 16 个字符,显示在“块头”中
Comment	块注释,可以编辑块注释,在块头中显示。最多允许输入 80 个字符,对于较大的块和嵌套图表,块头中只能显示头 14 个字符;而对于较小的块,块头只能显示 7 个字符的注释
OCM possible (Operator Control and Monitoring)	该复选框确定是否将块添加到操作员站的监视列表中用于监控,如果需要块在 OS 中用于操作员监视和控制,则选择该选项,该选项激活 Operator C and M
Operator C and M	单击该按钮可以打开一个对话框,在对话框中显示可以用于监控的 I/O,这些 I/O 的 WinCC 属性可以在此编辑。对于 WinCC 和 PCS7 库中的块,有默认设置,无须调用此对话框

续表

选 项	描 述
Create Block icon	用于监视和控制的块可以在 WinCC 中以块图表的形式显示,可以对相同块类型的不同块设置不同的图标,可以在此输入域中输入最多 16 个字符,指定 WinCC 中该块显示哪个图标
Message	此按钮用来打开消息组态对话框
Inputs	如果块输入变量可变,而且数据类型相同,可通过 Edit-Number of I/Os 菜单命令修改块输入的个数
Internal	FC、FB 等标识符及编号, SIMATIC 管理器中, Blocks 文件夹以“对象名称”显示内部块标识符
Instance	FB 对应背景 DB 块的编号
Name(header)	显示块创建时,在内部“块头”指定的算法块名称
Family	显示块创建时,在内部“块头”指定块系列的名称,用于在块目录窗口中各块分类显示
Author	显示块创建时,在内部“块头”指定作者的名字,如对于基本操作块,作者为 CFC-BOP,技术块则显示为 TECH
To be inserted in	某些块必须安装在特定的任务中,如为确保启动响应,过程控制消息响应及其他属性将会起作用。此处显示 CFC 块的任务列表,通过菜单命令 Edit→Run Sequence

CFC 提供两种方式修改块输入和输出的属性:

① 双击所选块,打开属性对话框中的 Inputs→Outputs 选项卡,I/O 的各属性以表格的形式显示,在此编辑块所有的 I/O。

② 对于单独的 I/O,双击所需的块 I/O,打开 Properties→Input→Output 对话框。根据 I/O 属性和数据类型的不同,对话框中会包含不同的信息,有些信息可以编辑,有些则是只读的。

(5) 修改 I/O 个数

对于某些块类型,其 I/O 个数是可变的,用户可以修改此类块的输入个数。下面以 BIT_LGC 块系列中的 OR 块为例,介绍修改输入个数的方法。

将 OR 块拖放到图表的中心工作区,如图 3.20 所示。



图 3.20 默认的两个输入的 OR 块

选中该块,执行 Edit→Number of I/O…菜单命令,弹出如图 3.21 所示对话框,其中在 Number(个数)可编辑文本框中,输入块输入的个数(2~120),则 OR 块的输入个数随之改变。本例中输入 5,确定后,OR 块外如图 3.22 所示。如果输入的数字不在 2~120 范围内,则会提示输入的数字过大或过小。对比图 3.21 和图 3.22 可以看出由于输入个数增加,算法块也变大,如果该块所处位置没有足够的空间,块大小增加后,则显示为重叠块。

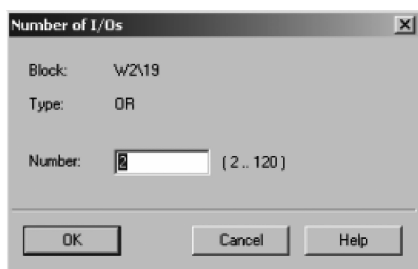


图 3.21 输入/输出个数对话框



图 3.22 5 个输入的 OR 块

3) 互连

CFC 图表中的互连是指块/图表的输入/输出和其他块/图表或者同一个块的输入/输出、运行时间组,以及如共享地址等 CFC 之外的对象之间的连接。输入/输出的数据类型必须一致。互连的块/图表可以位于同一表单,也可以在相同图表的不同表单或同一 CPU 的其他图表中。

(1) 到共享地址的互连

共享地址如 S7 中的共享数据块, I/O 信号, 为存储器, 定时器, 计数器等, 其连接端点位于 CFC 图表之外。选择一个 I/O, 右击, 在弹出的菜单中选择 Interconnection to Address 命令, 如图 3.23 所示, 或通过菜单命令 Inserts→Interconnection to Address 显示地址选择列表, 该列表包含了当前所有的可用于此 I/O 的符号表。

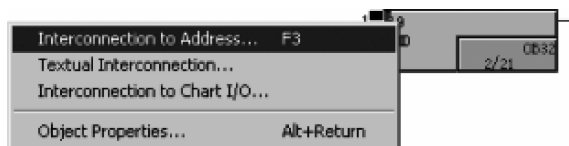


图 3.23 互连到共享地址

(2) 到运行时间组的互连

可以连接一个块的二进制输出到运行时间组, 即一个块的输出值决定了一个特定的运行时间组是否执行。选择一个块的二进制输出, 右击, 在弹出的菜单中选择 Interconnection to Run→Time Group 命令, 或通过菜单命令 Insert→Interconnection to Run→Time Group, 将显示一个带 CPU 所有任务及相应运行时间组列表的对话框, 如图 3.24 所示, 双击选择需要的运行时间组, 则在页栏输入互连。若删除运行时间组, 则互连也自动删除, 运行时间组的使能属性重新设置为 1。

(3) 文本互连

一个块互连特殊的形式是与位于另一个图表中的对象甚至未知对象的文本互连。一

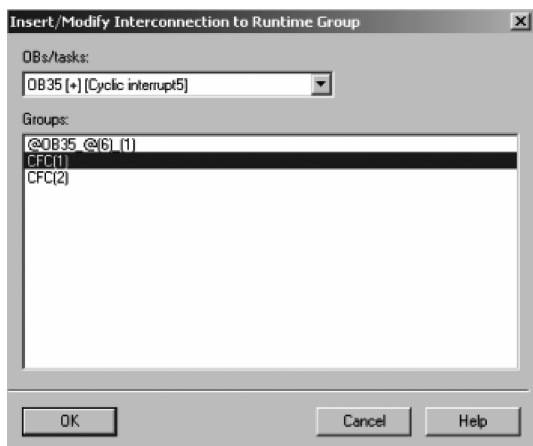


图 3.24 运行时间组对话框

个文本互连存在于一个块/图表的输入端并注明一个块/图表的输出,是一个带字符串用于确定具体互连源(输出)的输入地址,图 3.24 所示是选择 Textual Interconnection 命令后可以实现文本互连的对话框。

(4) 与 SFC 图表互连

CFC 图表中互连的特殊形式是从动作或转换条件到块或嵌套图表的输入或输出的直接 SFC 访问。这种类型的 SFC 访问可以在 CFC 图表中“重接线”,即可以从一个 I/O 移动到一个带兼容数据类型另外的 I/O。如果此 I/O 为互连的,则可以决定是仅仅移动 SFC 访问还是同互连一起移动。按住 Alt 键拖动选择的 I/O 到需要的新 I/O, SFC 访问符号将移动到新的 I/O,但任何存在的互连未移动;拖动选择的 I/O 到需要的新 I/O,显示一个消息询问是否同时移动 SFC 访问,如果选择 Yes,则互连和 SFC 访问都移动,而选择 No,则仅互连移动。

CFC 图表中,有一个块 I/O 标记来指示 SFC 的访问,在 I/O 上的标记意味着“读访问”,反之为“写访问”,标记的颜色与用来指示连接的数据类型的颜色相同。可以在 Objects Properties→I/Os 对话框显示 SFC 参考,如果 I/O 包括 SFC 访问,则对话框包含 SFC Access 按钮用来显示带 SFC 参考列表的对话框。双击一个列表 SFC 图表名称可打开显示选择的参考对象的相关图表。

4) 块的运行时间属性

一个块的运行时间属性决定了块在整个 CPU 运行序列中的位置。这种属性对于目标系统在响应时间、死区时间,以及基于时间的结构稳定性方面非常关键。插入一个块时,在运行序列中的每个块都有一个默认的运行时间属性,块也可以安装到已经在任务中的运行时间组中去。

显示运行时间属性有几种方法,如图 3.25 所示为单个块的运行时间属性,而对整个 CPU,可以查看全部运行序列。

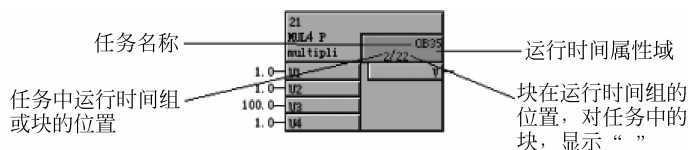


图 3.25 单个块的运行时间属性

5) 编译

用户可以通过菜单命令 Chart/Compile 选择将图表编译为一个程序或者将单个的图表编译为一个块类型。

(1) 编译图表为程序

当编译时,活动的 CPU 图表被转化为机器码,S7 中使用的是 SCL 编辑器。编译时,可以选择完全编译和只对修改部分进行编译,如果只对修改部分进行编译,则只检查修改部分程序;如果希望检查整个程序,启动连续性检查功能。在选择编译为程序后,将出现一个对话框,如图 3.26 所示,实际编译前可以选择一些选项。

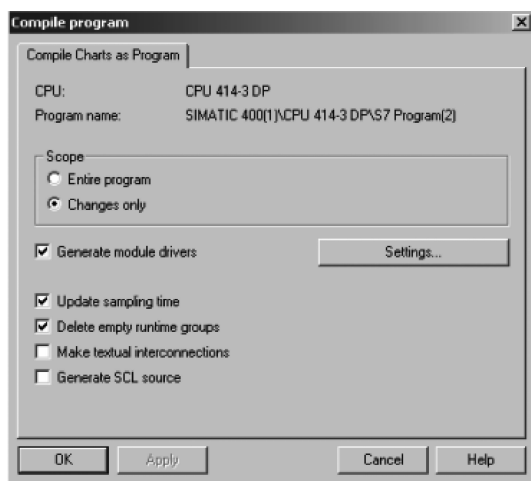


图 3.26 编译图表为程序的对话框

(2) 编译图表为块类型

可以将希望多次使用的 CFC 图表创建块类型,并指定这些块类型的系统属性。通过菜单命令 Charts/Compile/Charts as Block Type 启动编译器,如图 3.27 所示。其中 General 选项卡,指定编译器的块类型属性,如 FB 号、符号名、名称(头)、家族、作者,以及

版本等,还可以指定将使用块的目标系统(S7-300/400)等。可以通过选择本地需求还是在运行时下载修改来优化代码。

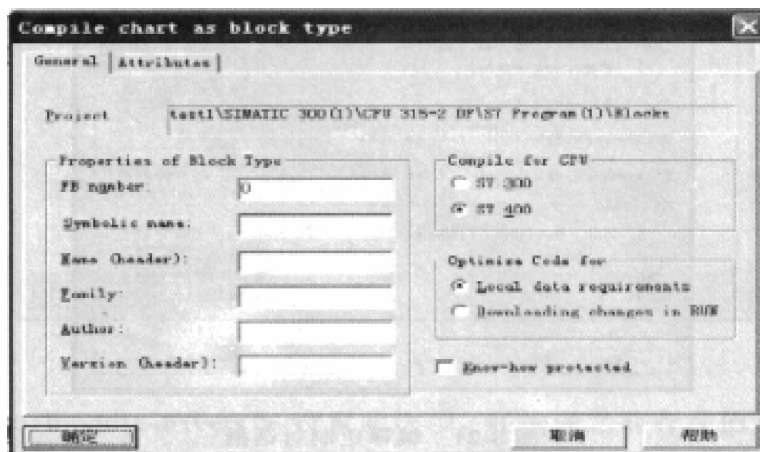


图 3.27 编译图表为块类型的对话框

对于本地数据需求类型的优化代码,图表的改变不会提高本地数据的需求,因为所有临时变量都存储在背景 DB 中,但是会导致背景 DB 结构的改变,以及时间标记接口的改变,在此情况下,不能在线下载修改;而运行下载修改类型优化代码,如果图表中有修改,由于临时变量存储在 VAR_TEMP 区域,不改变背景 DB 的时间标记接口,通常情况下修改可以在线下载,但会使本地数据的需求增加。

还可以激活 Know_How 保护,这样块的算法只能在 SCL 源文件存在的时候查看或修改。在属性选项卡中可输入块类型的系统属性,如系统属性 S7_blockview 的值为 big。

6) 下载和读回图表

要使用 CFC 创建的程序,必须将其从 CFC 下载到目标系统,因为 CFC 的下载功能保证现场站组态数据的连续性。编译整个程序并不一定要完全下载,可以只下载修改部分,编程时要频繁地编译程序。下载全部程序可以在 STOP 和 RUN-P 模式下进行,下载修改部分的程序在 RUN-P 模式下进行,但也有可能导致 CPU 停止。

当现场站的数据修改后,例如操作员站上通过 WinCC 的操作员接口修改限制值或控制参数等,用户可以读取包含在活动图表的图表文件夹中所有 CFC 图表的参数。通过菜单命令 Chart/Read back 启动,选择 CPU 程序作为 Read back 对话框的源,CPU 中的程序和当前参数被读到当前的图表文件夹中;另一种方法是通过 SIMATIC 管理器上传程序的方式从 CPU 读出数据,要在 Read back 对话框中指定 Program Offline 作为源,数据就从离线块文件夹读出并写到图表文件夹中。在对话框中可以选择是否希望读取所有的块输入参数,或者只是指定给系统属性用于监控(S7_m_c := 'true')的参数。

读取程序后,自动对整个程序进行编译,但是不会下载整个程序。可以在 RUN 模式下下载(只下载修改部分)。读取程序后将创建并显示一个记录,可以通过选项卡 Options/Logs/Read back 显示记录;此外,记录还可以显示哪些 I/O 被读取和修改,问题出在哪里等。

需注意的是:不能指定参数值(属性: $S7_param := 'false'$)的 I/O 不能读取;读取时忽略 FC 输入值和数据类型 ANY, ARRAY, POINTER, UDT;如果现场站数据被修改,则图表就被读回,这些修改值进行标注;从读回功能可以排除一些类型属性 $S7_read_back := 'false'$ 的块,如 BATCH 块。

3. 调试

CFC 编辑器集成了测试功能来监视或修改站中的块 I/O。CFC 有两种操作模式:编辑模式和测试模式,编辑模式是一种离线状态,组态一个 CPU 的全部软件结构;测试模式用于监视和指定块 I/O 的参数,即在线监视和修改值,测试时,CPU 中的图形程序表示和物理地址间必须有唯一的分配,因此测试模式下用户无法修改用户程序的结构,如插入删除块,改变输入/输出的互连等。

此外,用户可以将要监视的块 I/O 和 CPU 中的图表放到一个动态显示窗口显示连续更新的值,还有如显示模块信息、设置时间等一些功能辅助调试。

1) 测试模式

CFC 编辑其提供的测试功能用来监视和影响 CPU 中块的执行或根据需要改变设置等,测试模式与属于当前激活的图表 CPU 并联。通过菜单命令 Debug→Test Mode,激活或停止测试模式。多数编辑模式下的功能在测试模式下不可以使用。

测试模式下监视的 I/O 值从 CPU 循环更新,默认的更新循环时间是 CPU 指定的 2 s,对当前图表文件夹的所有图标都有效。可以修改循环周期时间,通过菜单命令 Debug→Test Settings 打开一个对话框,设置监视循环周期时间。

测试分为过程模式和实验模式两种。在过程模式下,块的在线动态显示通信是受限的,只是 CP 处理器和总线增加了有限的额外负担。激活测试模式,则所有块的状态都是 Watch off 状态;在实验模式下,块的在线动态显示通信是不受限的,处于激活测试模式,所有块的状态都是 Watch on 状态,可以用于监视测试 I/O。

2) 监视和指定块 I/O 的参数

如果测试的 I/O 的 Watch 功能是激活的,其值由测试模式下的 CPU 提供,激活测试模式,Watch on 功能也激活,CFC 图表在实验模式下,可以显示块/图表 I/O 的值,也就是说,这些值周期性地从 CPU 中读出并显示。在测试模式下可以修改动态显示项和 I/O 参数。在过程模式下,Watch off 功能是激活的,这就要先选择图中要监视的动态显示的块,接着单击 Watch on 按钮。如果 CPU 有过载,可以选择块/图表后通过菜单命令

Watch off 从监视表中移除单个的块或者图表。

如果没有激活监视,可以通过选择用于测试的 I/O 按钮,这样该 I/O 和原来测试的 I/O 同时激活。

激活测试模式,则在动态显示窗口建立所有 I/O 与 CPU 的连接,可以通过选择单个 I/O 的 Watch 栏激活 Watch 功能。可以通过菜单命令或单击从监视列表中移除 I/O 的按钮。

3) 动态显示

在测试模式下,块和图表 I/O 的值可以在一个分割的窗口进行动态显示,这对基本数据类型(BOOL, W, R 等)和结构的元素是有效的。

动态显示窗口同任何其他窗口一样可以在 CFC 编辑器窗口打开和管理,可以调整窗口的大小。通过菜单命令 Viwe→Dynamic Display 激活或禁止动态显示窗口,它包括以下内容:

- (1) 状态: 该行中显示状态信息(如 CPU 故障,值错误等)。
- (2) 监视: 用户可以选择是否监视 I/O,即是否动态显示值。
- (3) 图: 显示图的名字,若为嵌套的图,则上一级图的名字也显示。
- (4) 块: 显示 CFC 块或嵌套图的名字。
- (5) I/O: 显示 I/O 的名字,如 LMN,IN1。
- (6) 值: 读自 CPU 的块 I/O 的实际值,如果有问题则显示####。
- (7) 单元: 显示选择物理单元(如果存在)的文本。
- (8) 注释: I/O 的注释。

在编辑和测试模式可以插入 I/O 到动态显示,从一个打开的图表插入 I/O 的步骤如下: 选择需要的块 I/O,通过菜单命令 Debug→Inputs 或 Outputs 或右击,在弹出的菜单中选择 Insert in Dynamic Display 命令即可,或者用鼠标拖动相关的图表 I/O 到动态显示窗口。对于结构类型块 I/O,在 I/O 拖到动态显示之前,出现一个对话框可以选择显示的结构元素。如果希望在动态显示中包括一个块或一个嵌套图中所有的 I/O,则可选择块或图将其拖到动态显示窗口,但结构类型 I/O 除外。

4) 测试前和测试时的功能

除了直接测试功能外,还有例如设置日期和时间等一般的功能,以及在系统诊断和排除故障时的支持功能。例如显示操作模式或模块信息等帮助用户调试。

(1) 对比 CPU 程序的时间标记

通过菜单命令 PLC→Compare 显示下载用户程序的时间标记,对话框显示 3 种时间标记: 最后一次下载相关的修改,最后一次离线程序的修改,以及最后一次在线程序的修改。通过时间标记是否匹配,用户可以看出是否对用户程序进行了修改来决定是否需要重新编译或者重新下载。

(2) 开始或停止 CPU 程序

通过菜单命令 PLC→Operating Mode 在对话框中单击“冷启动”、“热启动”、“暖启

动”或“停止”按钮来启动或停止用户程序,与 SIMATIC 管理器的相应功能是一样的。

(3) 清除/复位 CPU

消除或复位意味着所有 CPU 中的用户程序和存在的连接都被删除,CPU 必须在停止状态与 SIMATIC 管理器的相应功能是一样的。如果要下载全部的用户程序,则在确认一个提示后所有 CPU 中的块将被删除,也就是说不必先清除 CPU 存储区。当清除 CPU 存储区时,CPU 中的用户程序和存在的连接都被删除。

(4) 设置时间和日期

通过菜单命令 PLC/Set Time and Date 可以在特定的 CPU 上设置日期和时间,与 SIMATIC 管理器的相应功能是一样的。

(5) 显示模块信息

通过菜单命令 PLC/Module Information 可以在特定的 CPU 上设置日期和时间,与 SIMATIC 管理器的相应功能是一样的。

3.3.3 使用 SFC 编制程序

SFC 是顺序功能图(Sequential Function Chart)。SFC 功能图是 SFC 编辑器中的一个可编辑的功能图,而 SFC 类型则表示在 SIMATIC Manager 中创建的一个作为类型的功能图,并在 SFC 编辑器中对其进行编辑。顺序控制可以在以下不同控制级中实施:

- 设备控制级(打开阀门,启动电机);
- 成组控制级(调配、搅拌、加热、填充);
- 单元级(罐、搅拌器、称、反应器);
- 工厂级(各单元和公共资源同步,进程安排)。

1. SFC 的原理

1) 操作状态

顺序控制系统的操作状态指的是当前状态,例如,系统继续运行是否需要操作员的干预,或者哪个命令可能使系统切换到一个不同的运行状态。

操作状态受到下列情况的影响:

- 在手动模式下,使用命令,如图 3.28 所示;
- 在自动模式下,通过功能图的外部视图。

SFC 功能图的所有指令和动作列表如下: Start, Abort, Restart, Hold, Complete, Resume, Stop 和 Error。动作将使 SFC 进入不同的状态,即 Idle, Starting, Run(Active), Completing, Holding, Error, Held, Held-Error, Resuming, Aborting, Stopping 和 Stopped。

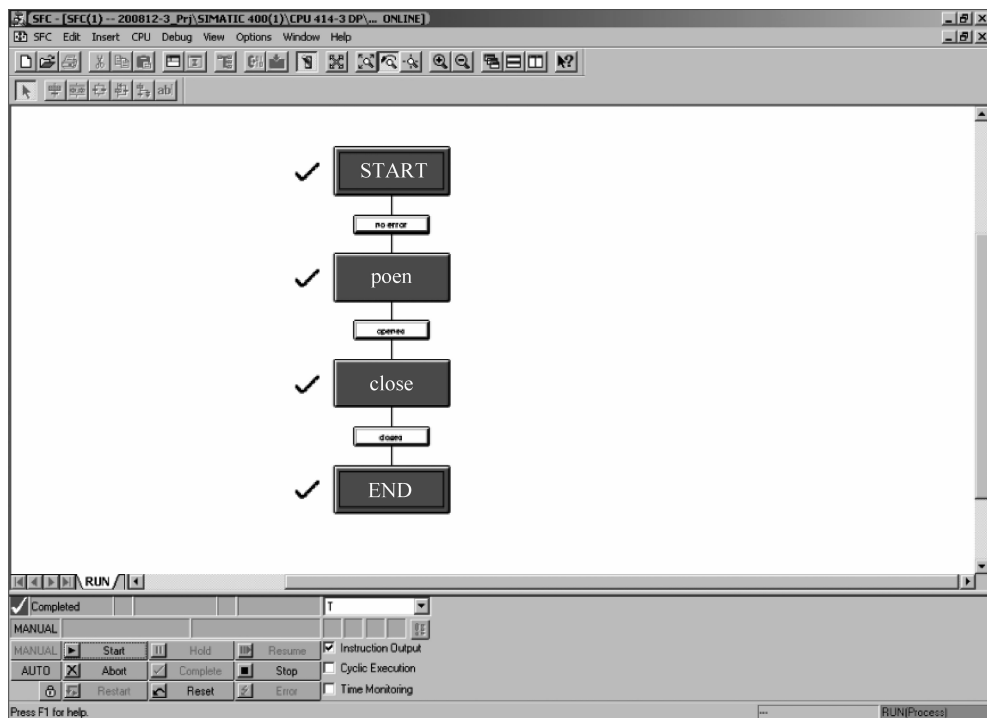


图 3.28 SFC 编辑器以及 SFC 在线测试模式

在 MANUAL(手动)模式下用户手动控制进程。可以使用所有步控制模式。在 AUTO(自动)模式下进程会自动运行。CFC-plan 中 AS 程序的运行可以通过参数设置或连接 SFC 功能图或外部视图。

2) SFC 功能图的顺序路径

一个 SFC 可包含多个用于不同应用中的顺序路径。不同的起始条件可用来确保当某个事件发生时,则执行相应的顺序路径。此可以为每个操作状态(准备、激活、故障等)或者每个操作特性(加热、冷却、等温等)组态各自的顺序路径。在一个 SFC 功能图中最多可以插入 8 个顺序路径。在图 3.29 中有 6 个路径,图中的上下文菜单中包含了处理顺序路径的功能,应用这些功能可以对一个顺序路径进行插入、移动以及删除等。

这个功能图第一个顺序路径的条件是 $SFC.RUN=1$,其中 SFC 是这个功能图的名称,RUN 是这个功能图的一个变量。每个新加入的顺序路径的起始条件都是空的,因此是不满足的,这就意味着这条路径不会执行。必须为一个新加入的顺序路径定义一个起始条件。在顺序属性 Sequence Properties 窗口中定义起始条件,如图 3.29 所示。

如果两个路径具有相同的起始条件,高优先级路径先于低优先级路径执行。可以给顺序路径分配从 1~255 个优先级,其中 255 级为最高优先级,如图 3.30 所示。

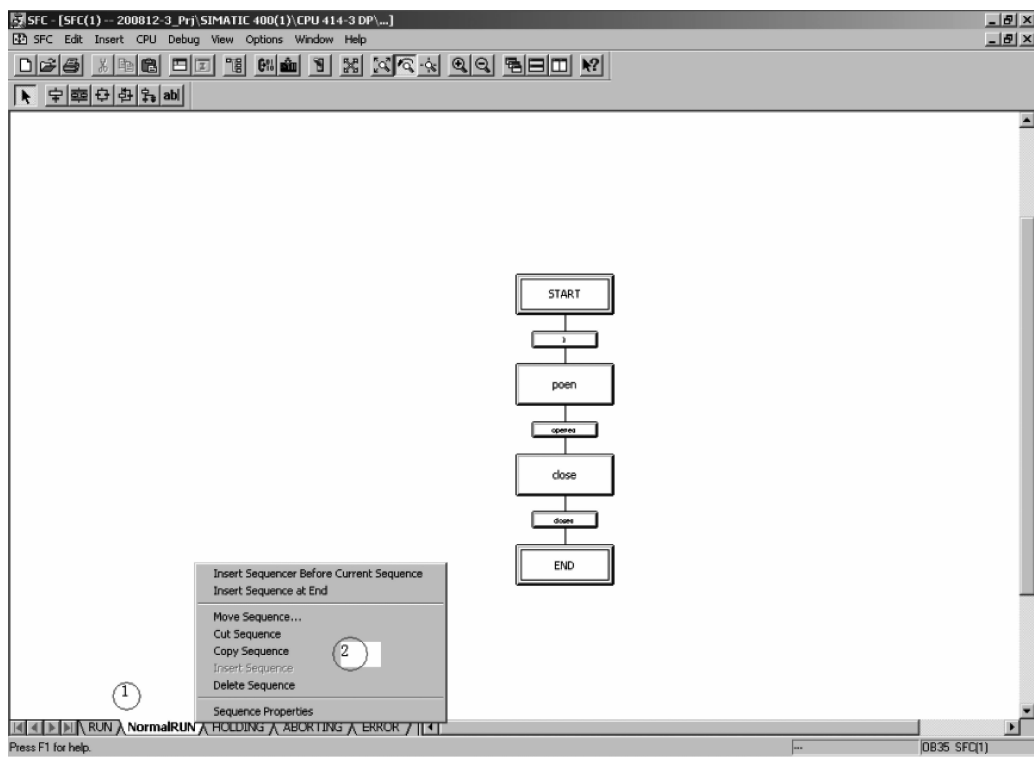


图 3.29 处理 SFC 顺序路径的功能

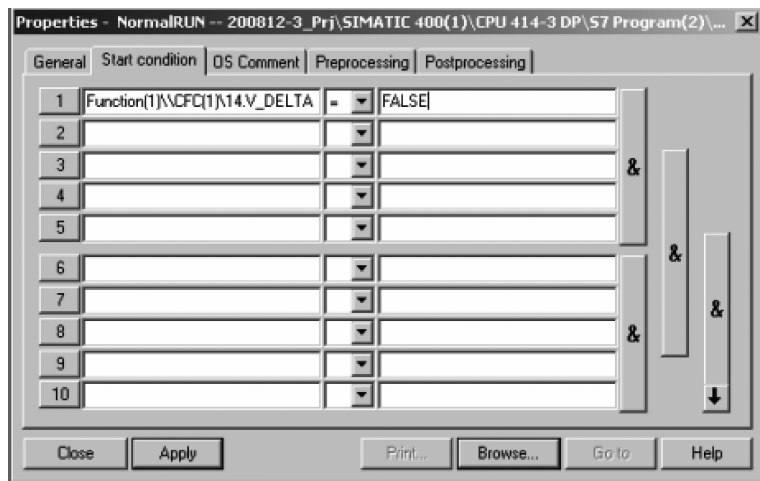


图 3.30 顺序路径的优先级

如果几个动作同时满足,SFC 编辑器中最左栏的顺序路径将会被激活,并且它们只执行一次。每个顺序路径都可组态一个循环动作,循环动作包括两个部分:预处理(Preprocessing)和后处理(Postprocessing)。预处理在循环顺序路径处理之前执行,后处理在循环顺序路径处理之后执行,这两个部分都可在 Sequence Properties 对话框中组态。

3) 步的阶段

每个步分为 3 个动作阶段,如图 3.31 所示:

- Initialization(初始化)是一个步第一个执行的动作;
- Processing(处理)是一个步循环执行的动作;
- Termination(终止)是一个步最后执行的动作。

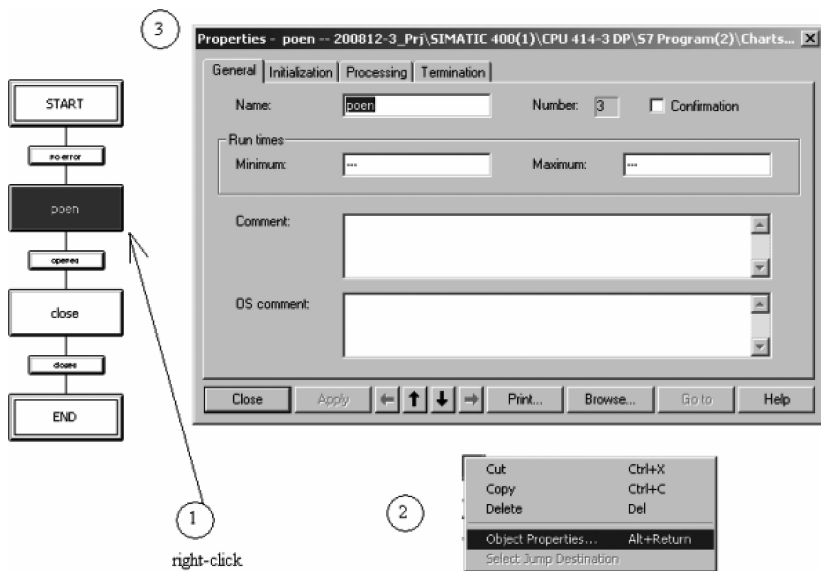


图 3.31 一个步的 3 个阶段

4) 执行一个顺序的步和转移条件

当开始一个没有查询条件的功能图时,激活 Start 步执行它的动作。一个步 Step 可以具有两种状态,即 active(激活)和 inactive(非激活)。当先前的转移条件将控制信号传送给这个步时,这个 Step 就会变为激活状态,随后触发并执行这个动作。查询激活步的后继转移条件,如果后继转移条件是真(该条件是真),先前的步被取消激活,并激活下一个步。如果组态了最小运行时间,那么,只有经过这段运行时间后才会对后继转移条件进行查询(取决于步控制模式)。END 步的动作只执行一次。在功能图从一步进行到下一步的过程中,前一步的终止动作是和下一步的第一个动作(初始化或者执行)在同一个循环周期内完成的。

5) 并行(同时的)顺序的执行

并行顺序在一个循环周期内同时执行,并且执行过程相互独立,如图 3.32 所示。并行顺序只有当并行循序路径结尾处的所有步都激活,并且满足转移条件时,该并行顺序后的转移条件才被执行。

6) 选择顺序的执行

选择顺序的路径是执行其转移条件先得到满足的顺序,如图 3.33 所示。选择顺序如果同时满足多个转移条件,那么,在功能图拓扑结构中最左边的转移条件被激活。

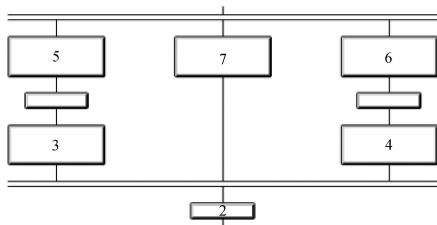


图 3.32 并行顺序

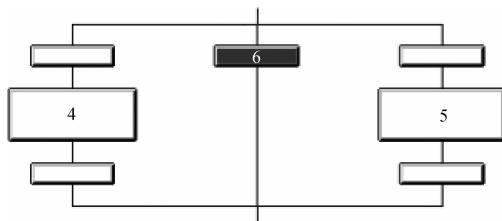


图 3.33 选择顺序

2. SFC 功能图

当新插入一个 SFC 功能图时,默认安装在组织块 OB35 中,因此就确定了该功能图的扫描速率。通常,在考虑到时间响应的情况下,应当检查 SFC 功能图是否安装在一个适当的 OB 块中。

操作模式决定了如何执行功能图,一个功能图可能具有下列操作模式:

- Auto(程序模式): 由控制功能块和其他 SFC 功能图控制,可选择步控制模式 T、T/T 和 C。CFC(连接/参数设置)实现了对 AS 程序的控制。
- Manual(操作员模式): 由操作员控制(处于测试模式或者在 OS 的面板上控制),由操作员手动控制功能图的执行(例如在调试期间),可以使用所有步控制模式。

每个 SFC 功能图都有一个外部视图,这个视图是 CFC 中的一个功能块。图 3.34 给出了功能图 SFC_val 的外部视图,通过下列菜单路径调用功能图的外部视图:(在 SFC 编辑器中)选择菜单命令 View→External view 或者(在 SIMATIC Manager 中)右击功能图,打开外部视图。

在 ES 的 SFC 编译器中,可以编译 SFC 功能图并将其下载到 AS 站。因为编译或者下载的功能是与整个 S7 程序相关的,因此在 CFC 编译器或者 SFC 编译器中执行编译或下载这个程序是一样的。

3. SFC 类型

在过程自动化中,一些顺序控制几乎是相同的,其中只有微小的变化。例如一个罐子

有两种方法可以找到一个 SFC 类型的输入/输出：

- 将 SFC 类型放至一个 CFC 功能图中,这个功能块的 I/O 就是这个 SFC 类型的 I/O;
- 在 SFC 编辑器中,单击按钮打开 Chart I/O。如图 3.36 所示,在窗口右边列出了这个 SFC 类型的 I/O。

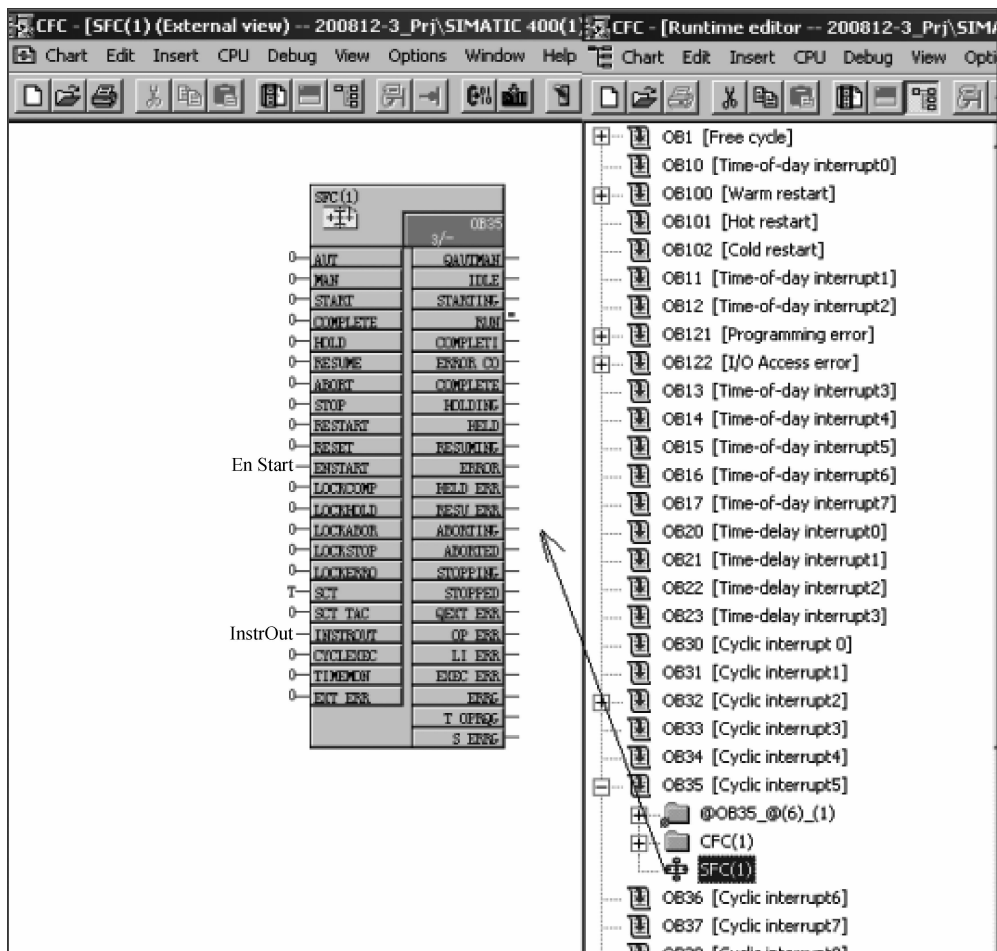


图 3.36 使用 SFC 类型

通过拖放,可以把在主数据库中创建的 SFC 类型方便地从库移到单个项目的 CFC 功能图中。

若更改主数据库中的 SFC 类型,则必须对单个项目中 SFC 类型的复制件进行更新。为此,要把 SFC 类型再一次从库复制到单项目的 CFC 功能图中。

3.4 OS 和图形编辑器

3.4.1 WinCC 系统概述

WinCC 是在生产和过程自动化中解决可视化和控制任务的工业技术中性系统。它提供了适用于工业的图形显示、消息、归档以及报表的功能模板。高性能的过程耦合、快速的画面更新以及可靠的数据使其具有高度的实用性。

除了这些系统功能外,WinCC 还提供了开放的界面用于用户解决方案。这使得将 WinCC 集成入复杂、广泛的自动控制解决方案成为可能。可以集成通过 ODBC 和 SQL 方式的归档数据访问,以及通过 OLE 2.0 和 ActiveX 控件的对象和文档的链接。这些机制使 WinCC 成为 Windows 世界中性能卓越、善于沟通的伙伴。

WinCC 是基于 Windows NT 32 位操作系统。Windows NT 具有的抢先多重任务的特性确保了对过程事件的快速反应并提供了多种防止数据丢失的保护。Windows NT 同样提供了安全方面的功能。WinCC 软件本身是 32 位的应用程序,开发使用调制解调器、面向对象的软件编程技术。

如果通过开始菜单启动 WinCC,将首先打开 WinCC 资源管理器。在此可以访问各种编辑器,从中执行操作和监控系统的指定任务。

1. WinCC 编辑器

(1) 图形编辑器

图形编辑器是一种用于创建过程画面的面向矢量的作图程序。也可以用包含在对象和样式选项板中的众多的图形对象来创建复杂的过程画面。可以通过动作编程将动态添加到单个图形对象上。向导提供了自动生成的动态支持并将它们链接到对象。也可以在库中存储自己的图形对象。

(2) 报警记录

报警记录提供了显示和操作选项来获取和归档结果。可以任意地选择消息块、消息级别、消息类型、消息显示以及报表。系统向导和组态对话框在组态期间提供相应的支持。为了在运行中显示消息,可以使用包含在图形编辑器中对象选项板中的报警控件。

(3) 变量记录

变量记录被用来从运行过程中采集数据并准备将它们显示和归档。可以自由地选择归档、采集和归档定时器的数据格式。可以通过 WinCC 在线趋势和表格控件显示过程值,并分别在趋势和表格形式下显示。

(4) 报表编辑器

报表编辑器是为消息、操作、归档内容和当前或已归档的数据的定时器或事件控制文档的集成的报表系统,可以自由选择用户报表或项目文档的形式。提供了舒适的带工具和图形选项板的用户界面,同时支持各种报表类型。具有多种标准的系统布局和打印作业。

(5) 全局脚本

全局脚本是 C 语言函数和动作的通称,根据其不同的类型,可用于一个给定的项目或众多项目中。脚本被用于给对象组态动作并通过系统内部的 C 语言编译器来处理。全局脚本动作用于过程执行的运行中。一个触发可以开始这些动作的执行。

(6) 文本库

可以在文本库中编辑多种模块在运行中使用的文本。在文本库中为组态的文本定义了外语输出文本。随后输出在运行语言中。

(7) 用户管理器

用户管理器用于分配和控制用户的单个组态和运行系统编辑器的访问权限。当建立了一个用户,就设置 WinCC 功能的访问权力并独立地分配给此用户。最多可分配 999 个不同的授权。用户授权可以在系统运行时分配。

(8) 交叉索引

交叉索引用于为对象寻找和显示所有使用处,如变量、画面和函数等。使用“链接”功能可以改变变量名称而不会导致组态不一致。

2. WinCC 基本选项

(1) 客户机服务器

使用客户机服务器功能,WinCC 与联网的自动控制系统的互连中,操作几个并列的监控站,理论上,至多 64 个客户机可以集成在单一项目中。

(2) 冗余

WinCC 冗余提供了并行操作一对服务器的可能,因此两台机器间可以相互监控。如果一台失败,另一台接管整个系统的控制。在服务器恢复继续服务后,全部消息和过程归档就复制到先前不能服务的服务器上。

(3) 用户归档

WinCC 用户归档是一个数据库系统,用户自己可以对其组态。这样,来自于技术处理的数据可持续存储在服务器上并在运行时在线显示,而且被连接控制的配置和设定值的赋值也可以存储在用户归档中并且在需要时传递到控件。

3. WinCC 过程控制选项

(1) 存储

存储功能支持硬盘与长期数据介质自动地进行数据交换,同样也可以在硬盘上将数据删除。

(2) 画面树管理器

画面树管理器用来管理系统、子系统、函数名称和图形编辑器画面体系。

(3) 时间同步

时间同步是用于带有 SINEC L2/L2R 总线系统的一种系统功能。为此一台 WinCC 操作员站承担作为主时钟的任务,并控制所有其他操作员站以及与当前时间的系统总线连接的自动系统。这使整个系统可以按时间顺序来排列消息。

(4) 设备状态监控

设备状态监控用来不断地监控单个系统(操作员站和自动控制系统)并且将运行系统的结果可视化作为画面显示,自动触发蜂鸣器组件,以及自动生成系统消息。

(5) 基本数据

基本数据用来通过向导组态基本的 WinCC 数据。

(6) 拆分画面向导

拆分画面向导是拆分画面管理器的组件。用来组态、初始化当前 WinCC 项目的监控器和画面设置。在创建一个项目后由于其他的应用程序(运行、组显示等)要访问这些数据,应立刻执行此初始化。

(7) 报警记录向导

报警记录向导用来组态和初始化消息窗口、消息、消息级别、自动系统消息的系统和当前 WinCC 项目的蜂鸣器信号设备。

4. WinCC 和控制单元之间的通信

1) 与 SIMATIC S5 结合的通信类型

- 通过可编程接口的串行连接(AS511 协议);
- 通过 3964R 的串行连接(RK512 协议);
- 以太网第 4 层(通过数据处理块);
- 以太网 TF(科技功能);
- S5-PMC 以太网(PMC 通信);
- S5-PMC Profibus(PMC 通信);
- S5-Profibus-FDL。

2) 与 SIMATIC S7 结合的通信类型

- M PI 接口(S7 协议);
- PROFIBUS(S7 协议);
- 工业以太网(S7 协议);
- TCP/IP;
- SLOT PLC;
- S7-PMC PROFIBUS(PMC 通信)。

3) 混合操作 SIMATIC S5/S7

SIMATIC S5 和 SIMATIC S7 控制能在相同的网络并行连接并与 WinCC 交换数据。SIMATIC S7 仅使用 S7 协议。

4) 与其他控制通信

- | | | |
|-----------------|-----------------|----|
| • 独立厂商 | Profibus FMS/DP | 选项 |
| • AEG Modicon | Modbus 串行 | 选项 |
| • Allen Bradley | DF1 串行 | 选项 |
| | Data Highway | 选项 |
| | Data Highway+ | 选项 |
| | Highway 485 | 选项 |
| • GE Fanuc | SNP/SNPX 协议 | 选项 |

3.4.2 OS(操作员站)

1. 操作员站

PCS7 中有 3 种类型的操作员站,它们分别是:

- OS 服务器;
- OS 冗余(或备用)服务器;
- OS 客户机。

一台 OS 服务器和它的冗余服务器组成一个服务器对,在很多情况下它们被看作一个统一体。一台 OS 服务器(和/或者它的冗余服务器)连接在工厂总线上与自动化系统通信。服务器并不显示画面以供操作员监控。OS 客户机用于操作员监视和控制。它们通过终端总线与服务器通信。也可使用多台服务器,每台服务器对应工厂的一个分支或工艺车间。来自于不同服务器(多于一台)的数据可以在一台客户机上显示。每台服务器可以被 32 台 OS 客户机访问一台客户机可以访问 12 台 OS 服务器(或者 12 对服务器)。如果安装了多个 VGA 卡,则一台客户 PC 可以连接多达 4 台显示器,这种情况当作 4 台

客户机。

在 Component view(组件视图)中将操作员站插入 PCS7 项目中。首先插入 PC 工作站,然后插入 OS(服务器)、OS(备用服务器)或者 OS 客户机,如图 3.37 所示。OS 站的组态使用 HW-Config、NetPro、Configuration Console(Commissioning wizard,调试向导)和 Station Configuration Editor 等。

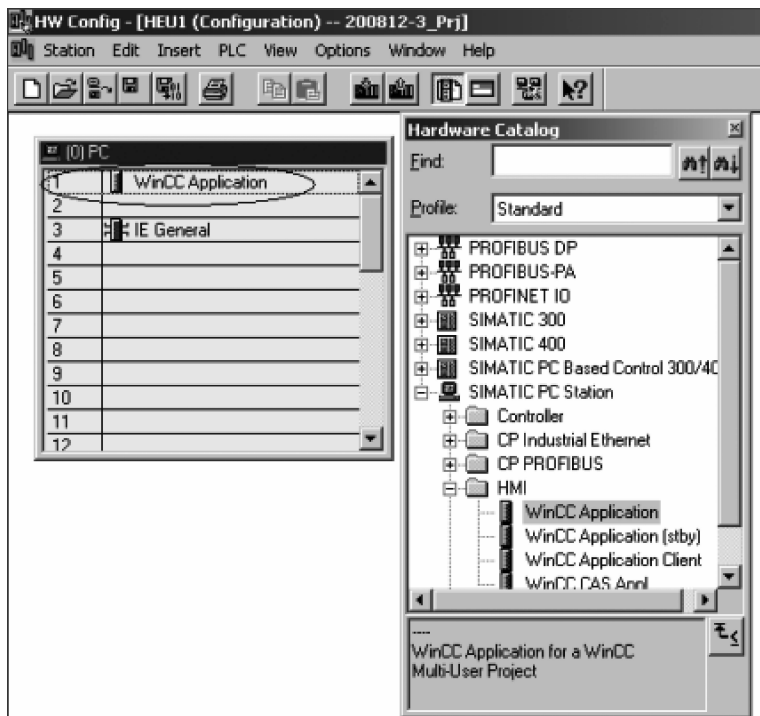


图 3.37 操作员站

2. OS 项目类型

PCS7 系统中有 3 种 OS 项目类型：

- 单用户项目。此种项目的数据库不能被其他的 OS 访问,项目中只有一个 OS 并且具有操作和显示功能。
- 多用户项目。此种项目的数据库可以被其他的 OS 访问,项目中至少有一个 OS 服务器和一个 OS 客户机在运行期间,OS 服务器不具有画面和其他的图形功能。
- 客户机项目。OS 客户机项目通过服务器上产生的数据包来访问 OS 服务器的数据。通过数据包,OS 客户机可以共享 OS 服务器的数据。

将一个 OS 服务器插入到 SIMATIC Manager 中时,默认的项目类型是多用户项目。

项目类型也可以在 PCS7 OS 中更改,如图 3.38 所示为如何指定一个 OS 项目类型。

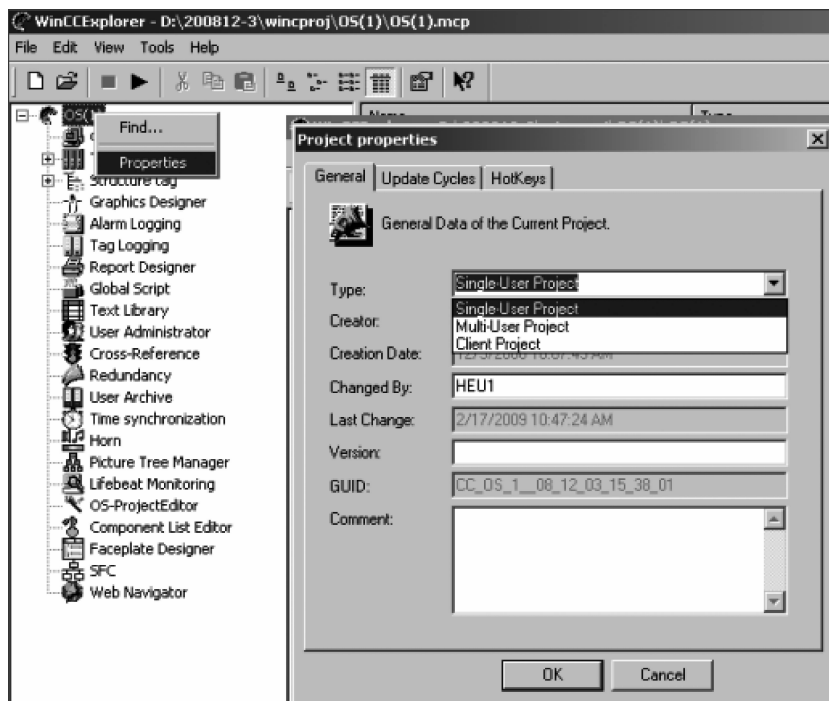


图 3.38 OS 项目类型

3. 在 SIMATIC Manager 中生成 OS 项目数据

相对于 WinCC,在 SCADA 应用程序中使用 PCS7 OS 具有很大优势。其优点在于,全集成的 PCS7 使得 SIMATIC Manager(AS 的组态)中产生的数据可以在 PCS7 OS 中访问。图 3.39 中示意了 PCS7 项目组态的不同阶段。

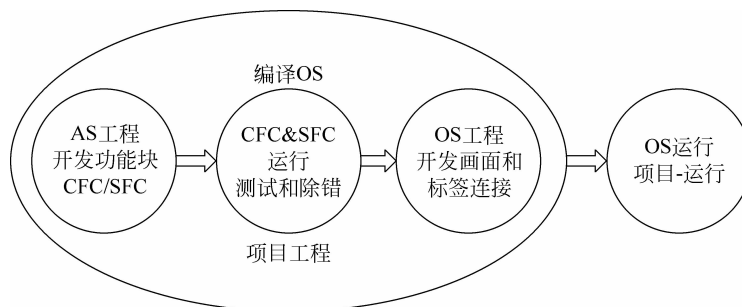


图 3.39 PCS7 OS 项目初始部分从 AS 转移

在编译 OS 时,属性 $S7_m_c=true$ 的功能块和变量被传送到 OS。如果功能块具有图表,它将被自动插入到一个已定义的画面中。通过两个系统对话框中,可以设置如何自动为 OS 插入功能块标签。

第一个要做的设置是设置画面的 Derive the block symbols from the plant hierarchy (从工厂层级中获得功能块标签)选项。

第二个设置是将功能块的符号插入到指定的画面中,如图 3.40 和图 3.41 所示。



图 3.40 调用创建/更新块标签图

在图 3.42 中 Create/Update Block Icons(创建/更新功能块标签)对话框的上半部分,列出的是所有设置为 Derive block symbols from plant hierarchy 的画面。

如果 Lower hierarchy levels included(包括低层级)选项设置为 0,意思是该画面只获取同一层级文件夹中的功能块标签。0 意味着画面中不包含更低级的功能块标签。

如果 Lower hierarchy levels included 选项非 0,低层级中的块标签也将被包括在高层级的所有画面中。例如,Lower hierarchy levels included=1 是指该块标签被嵌入它们所在的层级和更高层级的画面中。

4. OS 项目编辑器

OS Project Editor 将预先设计的画面、脚本动作和变量插入 PCS7 项目。该编辑器

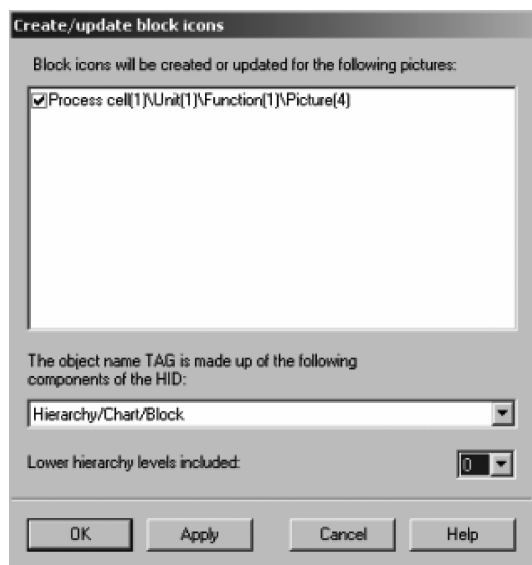


图 3.41 创建/更新块标签对话框

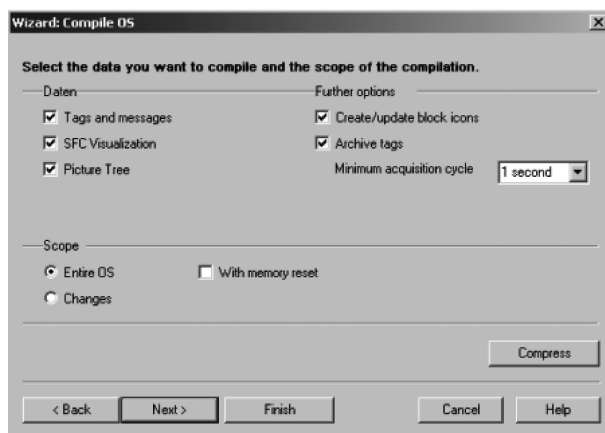


图 3.42 在编译 OS 期间创建/更新块标签

为 PCS7 项目指定典型设置。当在 PCS7 ES 中创建一个 OS 项目时,后台调用 OS Project Editor 并提供默认设置。因此,很多情况下,根本无须改变默认设置。然而,了解后台对了解 OS 项目是如何工作的很有帮助。如果想要改变某些默认设置,可打开编辑器进行修改,如图 3.43 所示。

在 OS Project Editor 的 Layout 选项卡上,可以设定 OS 监视器的分辨率、使用的监

视器的数目和监视器的布局,例如,水平或垂直分布。OS区域的默认数目是16。但是,用户可以更改(按Detail按钮)这个数目。根据监视器的分辨率最多可以组态64个区域。单击Detail按钮可以显示或更改布局。单击Suggestion按钮,调用系统共建的布局。该布局依据SIMATIC Manager的工厂层级中已组态的工厂区域的数目。

3.4.3 图形编辑器

Graphics Designer是PCS7 OS中的一个编辑器,它为创建过程画面提供了多种工具和对象。Graphics Designer的库包含大量现成的图形元素,如管道、阀门、罐等。可以将这些库对象加以修改并保存在应用程序项目库中。图3.44显示了Graphics Designer的组态环境和一些基本设计功能。

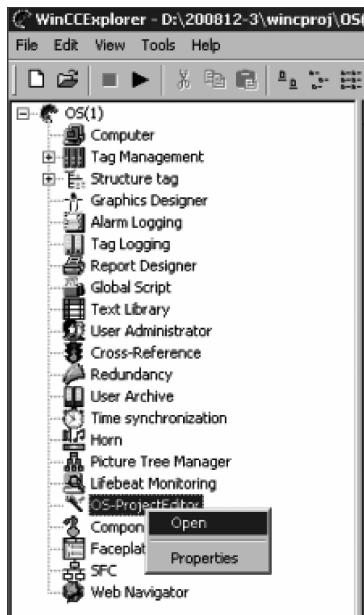


图 3.43 打开 OS 项目编辑器

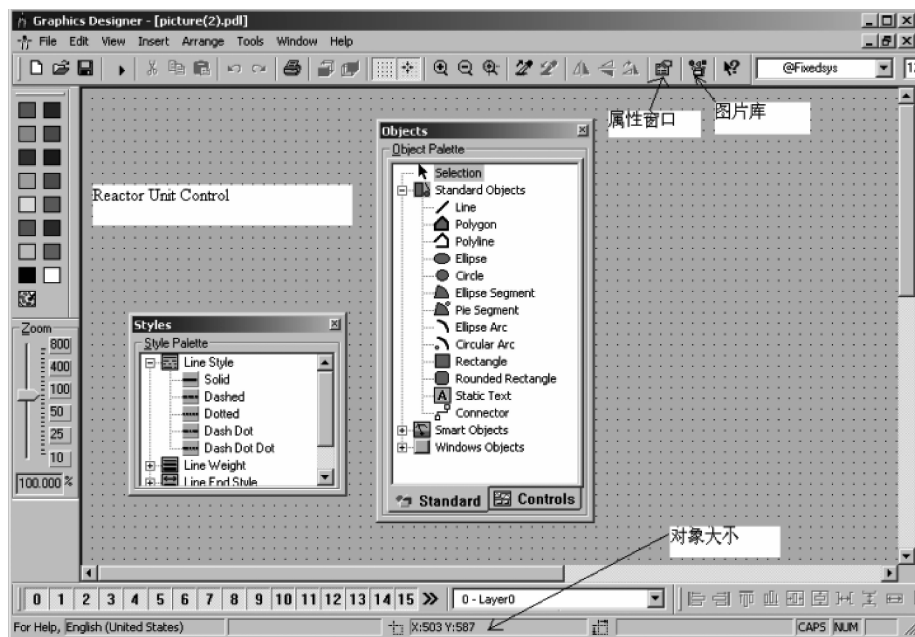


图 3.44 图形编辑器

1. 图形库

可把一个画面或者图形对象拖放到库里来建立自己的图形库。然后,可以通过副本粘贴的方式,用图形库为项目创建画面。西门子 HMI 符号库包括各种图形和过程图表,如图 3.45 所示。

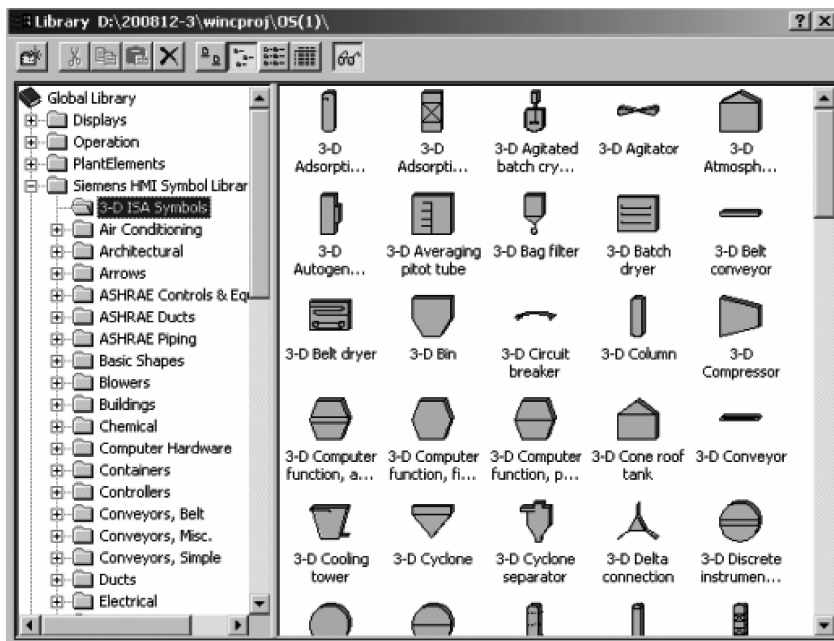


图 3.45 西门子 HMI 符号库

2. 基本操作以及属性对话框

图 3.46 显示了使用 Object Palette(对象选项板)设计图形的方法。例如在 Object Palette 中对 Bar(棒图)、I/O Field(I/O 域)、Static Text(静态文本)、Line(直线)和 Circle(圆)等对象进行拖放操作。

想要修改对象字体、颜色和动态连接等,则需使用 Object Properties(对象属性)对话框。在图 3.47 中,对如何使用属性对话框组态 I/O Field 的详细说明进行了图解。

3. 智能对象

Smart Objects 位于 Object Palette 中,通常它们具有到过程变量的动态连接。所有 PCS7 OS 提供的 Smart Objects,如图 3.48 所示。

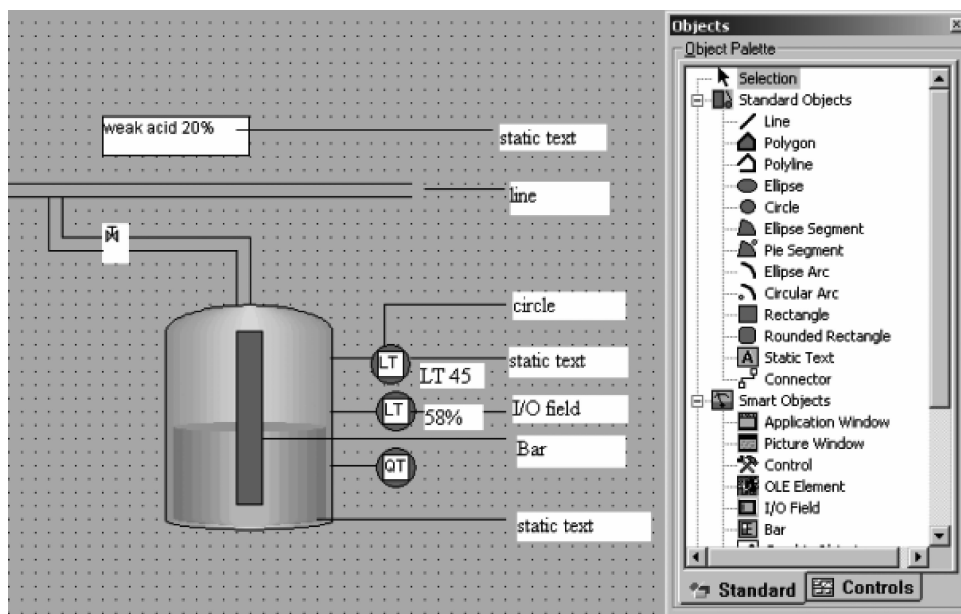


图 3.46 图形编辑器的基本操作

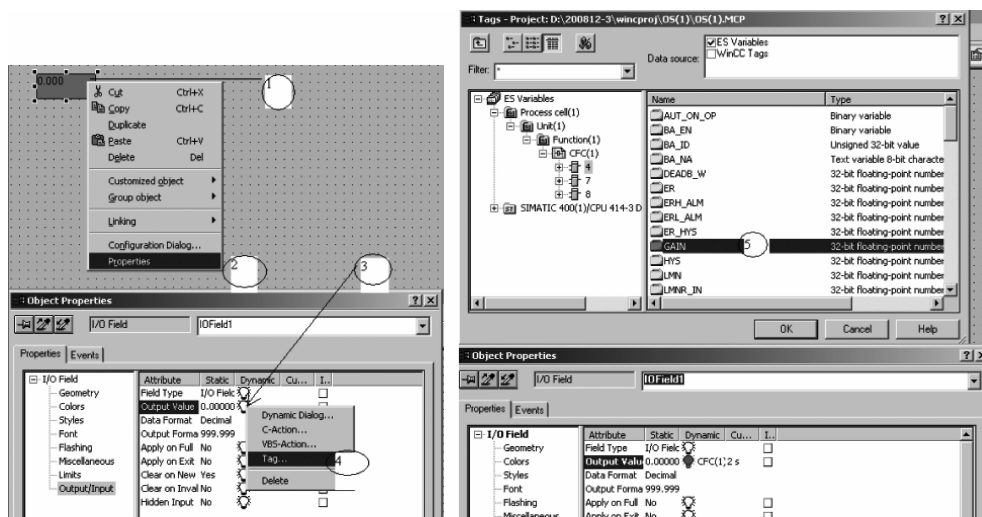


图 3.47 I/O 区块属性

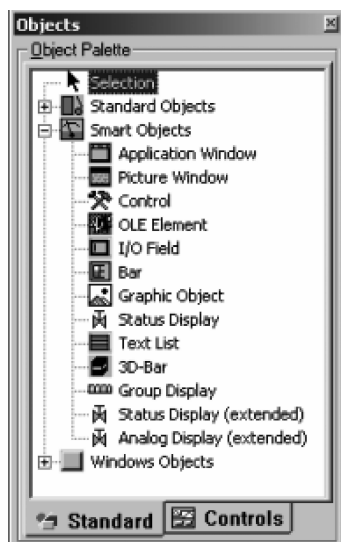


图 3.48 Smart Objects

4. 使图像动态化

有多种方法可以让图像动态化。前面讨论了变量连接(I/O 域)和鼠标动作(如打开一个面板)。其他的动态连接包括 Dynamic Dialog(动态对话框)、Direct Connection(直接连接)和 C-Action(C 动作)/VBS-Action(VBS 动作),如图 3.49 所示。

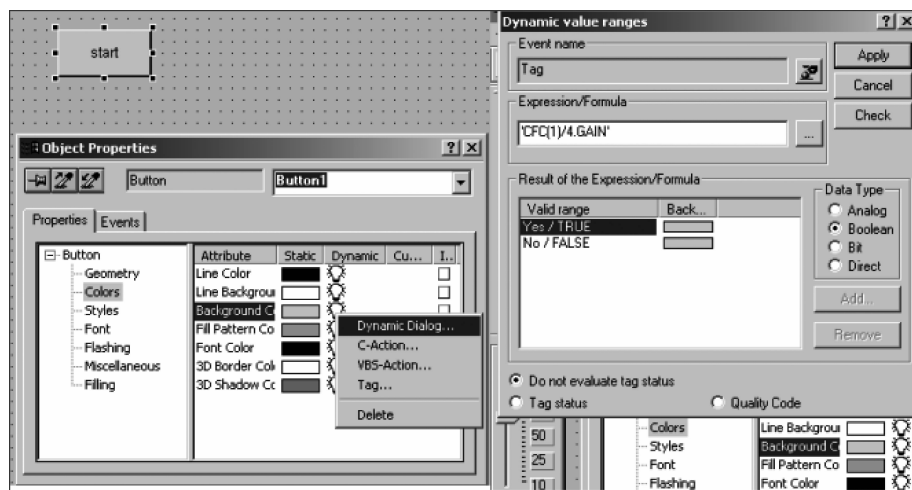


图 3.49 可用的动态连接

WinCC 在线帮助中很好地介绍了如何使图像动态化,如图 3.50 所示。

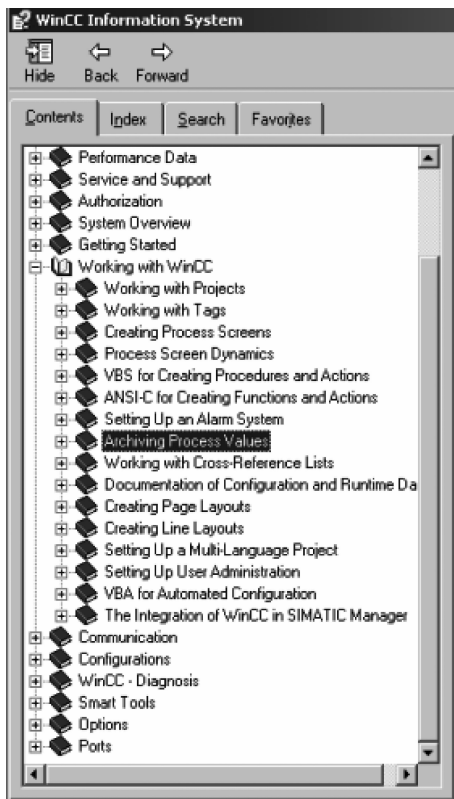


图 3.50 关于组态动作的在线帮助

3.5 Profibus PA 简介

Profibus PA 规范传送技术可谓是根据过程工业量身定制的。标准化的通信服务可保证不同制造商的现场设备之间的互操作性以及运行期间现场设备的远程参数化。

Profibus PA 网络基于电气传输部件。信息和能源管理通过一根屏蔽的双绞线即可完成。使用直线形、树形和星形拓扑网络,传输距离可达 1.9 km。可以使用无源 Profibus PA 终端元件(SpliTConnect 端接器)连接总线段。主要使用 DP/PA 链接器用于 Profibus DP 的网络传输。

对于 SIMATIC PCS 7,自动化系统和现场设备之间通过 CPU 中的代理块进行通信,覆盖所有现场设备。可采用运行过程中装入 DP/PA 链接器(DP/PA 链接器连接图如

图 3.51 所示)和现场设备中的组态数据使用工程师站进行组态。通过集成在工程师站中的过程设备管理器 SIMATIC PDM,实现参数化、调试和诊断。

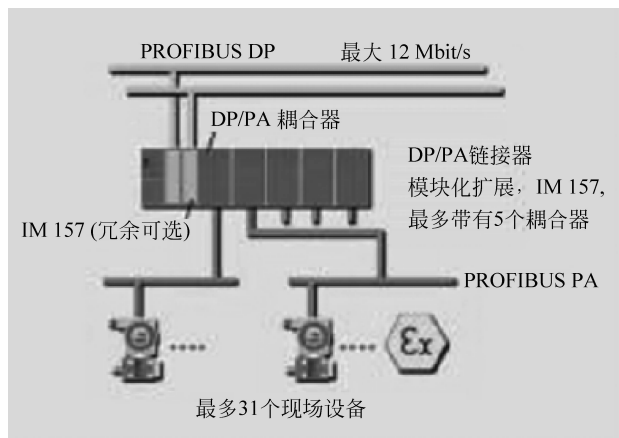


图 3.51 DP/PA 链接器连接图

采用 Profibus PA 规范下的分布式现场自动化系统的主要优点是,降低硬件成本和工程造价,提高运行安全性,实现免维护运行。这些优点表征如下:

- 从传感器到控制级的模块化和通用性,可以实现一种全新的设备理念;
- 通过在防爆场合中使用现场总线,可以实现安全应用;
- 通过现场设备的简单、集中工程组态(Profibus PA 和 HART 以及 SIMATIC PDM,与制造商无关),降低组态成本;
- 通过双绞线即可实现电源供应和数据传输,安装极为简单;
- 通过简化回路检查(Loop-Check),可以降低调试成本;
- 借助于简单的布线以及丰富的诊断功能,保养和维护成本很低。

DP/PA 链接器是一种非常简单的网关,用于连接 PROFIBUS DP 和 PROFIBUS PA 两个总线系统,但是降低了传输速率。因此,Profibus DP 和 Profibus PA 可以相互组合,对于 Profibus DP 的时间性能没有影响。