

实验一 组态软件的应用特点及项目的创建

一、实验目的：

- 1、了解 WinCC的特点。
- 2、熟悉和掌握项目管理器的结构。
- 3、掌握项目项目创建的步骤。

二、组态软件的介绍

WinCC 是进行廉价和快速组态的 HMI 系统，从其他方面看，它是可以无限延伸的系统平台。WinCC 的模块性和灵活性为规划和执行自动化任务提供了全新的可能。

从组态的角度上来看，在 WinCC 中有三种解决方案：使用标准 WinCC 资源的组态；利用 WinCC 通过 DDE、OLE、ODBC 和 ActiveX 使用现有的 Windows 应用程序；开发嵌入 WinCC 中的用户自己的应用程序（用 Visual C++ 或 Visual Basic 语言）。

WinCC 是基于 Microsoft 的 32 位操作系统（Windows NT4.0，Windows 2000 和 Windows 2000 XP）。该操作系统是 PC 平台上的标准操作系统。

WinCC 为过程数据的可视化、报表、采集和归档以及为用户自由定义的应用程序的协调集成提供了系统模块。此外，用户还可以合并自己的模块。

WinCC 的特点如下：

1. WinCC 的开放性 WinCC 对用户所添加的任何形式的扩充是绝对开放的。该绝对开放性是通过 WinCC 的模块结构及其强大的编程接口来获得。

2. 将应用软件集成到 WinCC 中 WinCC 提供了一些方法将其他应用程序和应用程序块统一地集成到用于过程控制的用户界面中。OLE 应用程序窗口和 OLE 自定义控件（32 位 OCX 对象）或 ActiveX 控件可以集成到 WinCC 应用软件中，就好像是真正的 WinCC 对象一样。

3. WinCC 中的数据管理 WinCC 中的默认数据库 Sybase SQL Anywhere 从属于 WinCC，该数据库用于存储（事务处理保护）所有面向列表的组态数据（例如变量列表和消息文本），以及当前过程数据（例如消息、测量值和用户数据记录）。该数据库具有服务器的功能，WinCC 可以通过 ODBC 或作为客户通过开放型编程接口（C-API）来访问数据库，也可以将同样的权限授予其他程序。因此，不管应用程序是在同一台计算机上运行，还是在联网的工作站上运行，Windows 中的应用程序均可访问 WinCC 数据库的数据资源，在数据库查询语言 SQL 和相关连接的工具（例如 ODBC 驱动程序）的帮助下，其他客户端程序（例如 UNIX 数据库，Oracle、Informix、Ingres 等）也可以访问 WinCC 数据库的数据资源。

4. 在项目开始之前规定组态分类 在项目开始之前，组态规定分为：WinCC 项目的名称，变量的名称，WinCC 画面的名称，创建脚本和动作的规则，组态规则（共同标准、库函数、按组工作），归档项目和方法。

运行项目的规定：这些规定很大程度上取决于应用领域（例如冶金、汽车工业、机械制造等）。规定有：用户界面（画面安排、字体和字体大小、运行语言、对象显示等）；控制概念（画面体系、控制原理、用户权限、有效键操作等）；用于消息、限制值、状态、文本等的颜色；通信模式（连接类型、更新的周期和类型等）；数量表（报警、归档值、趋势、客户端程序等的数目）；消息和归档的方法。

三、项目管理器

1. 启动

WinCC 安装成功后，WinCC 将出现在操作系统的开始菜单上。启动 WinCC 可使用 Windows Control 5.0 命令，也可通过其他方式启动 WinCC 项目管理器。

2. WinCC 项目管理器的结构

使用 WinCC 项目管理器，可完成以下工作：

创建和打开项目。

管理项目数据和归档。

打开各种编辑器。

激活或取消激活项目。

WinCC 项目管理器的用户界面由以下元素组成：标题栏、菜单栏、工具栏、状态栏、浏览窗口和数据窗口，如图 1-1 所示。

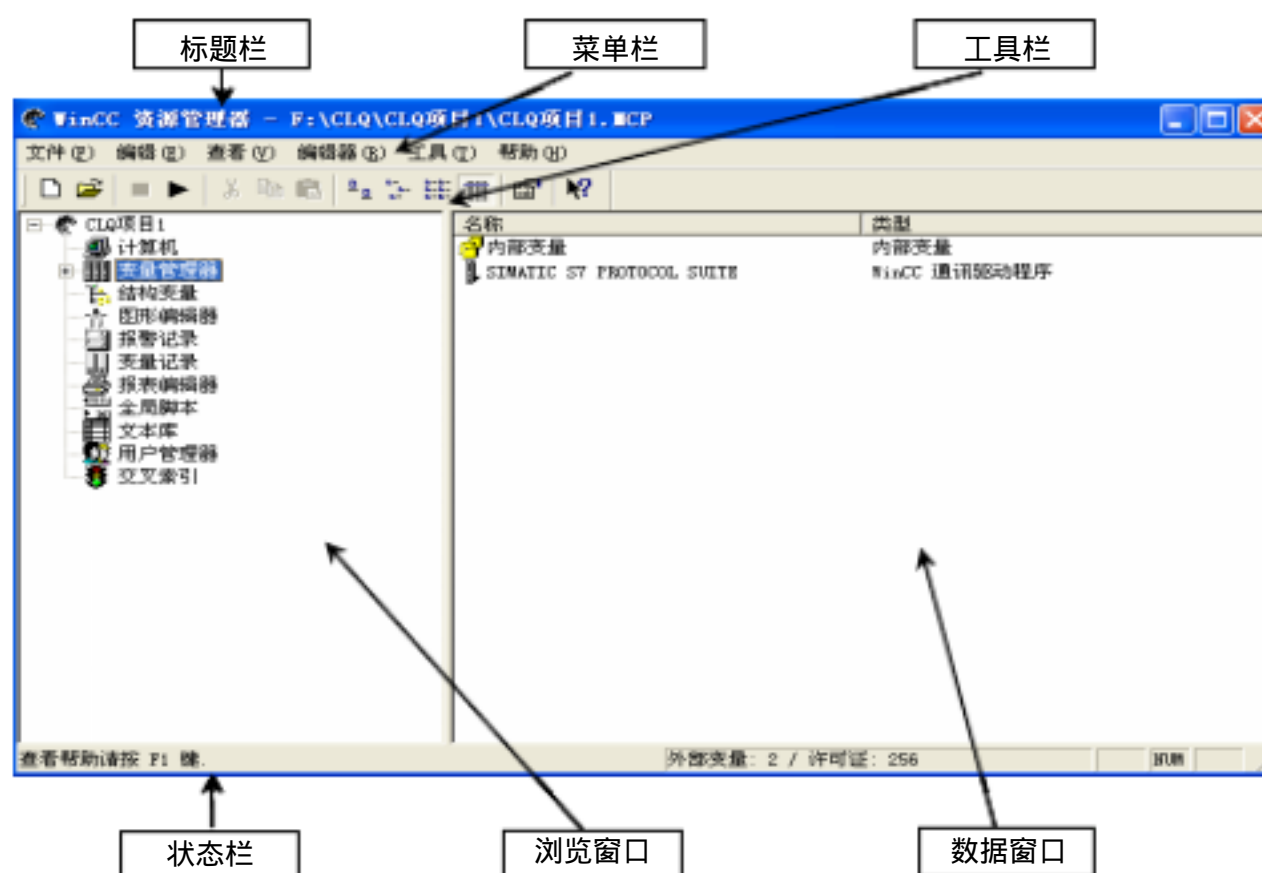











图1-1 WinCC 项目管理器

(1). 标题栏 标题栏显示当前所打开项目的详细路径和项目是否被激活。

(2). 菜单栏 菜单栏包含在 WinCC 项目管理器的组态系统中所有有效命令。这些命令排列成组并分配给不同的菜单。执行命令时可以打开适当的菜单，单击该命令。如有必要，在打开的对话框中设置所需的参数。

(3). 工具栏 工具栏上的图标使动作的实施更快捷。不需要通过菜单实现所需的功能。工具栏上各图标的描述如下：

图 标	描 述
	“文件”“新建”
	“文件”“打开 ...”
	“文件”“取消激活”
	“文件”“激活”
	
	
	
	
	

- “编辑”“剪切”
- “编辑”“复制”
- “编辑”“粘贴”
- “查看”“大图标”
- “查看”“小图标”
- “查看”“列表”
- “查看”“详细资料”
- “编辑”“属性”
- “帮助”“查找...”

(4). 状态栏 状态栏显示在 WinCC 项目管理器的下方空白处。左边显示的是关于当前项目的常规信息，右边显示键盘状态。使用“查看” “状态栏”菜单条目显示 /隐藏状态栏。在状态栏中各区域名的含义如下：

区域名	含义
已组态变量的数目	在此显示已组态的外部变量的数目。
连接变量的数目	在此显示许可证包括的变量数目。
CAPS	Caps Lock 是激活的。
NUM	数字键盘在键盘的右边，它是激活的。
SCRL	Scroll Lock 是激活的。

(5). 浏览窗口 浏览窗口位于 WinCC 项目管理器的左边，浏览窗口列出属于项目的所有组件（例如“计算机”、“变量管理器”等）。在浏览窗口中有一个以项目名称代表的主节点。浏览窗口包含 WinCC 项目管理器中的编辑器和功能的列表。双击“列表”或使用相应的快捷菜单可打开相应的编辑器。

浏览窗口中一个项目的组件含义如下：

项目组件	含义
计算机	分配给项目的所有工作站和服务器均在此区域里进行管理。变量管理器所有的通道、逻辑连接、过程与内部变量以及变量组均在此区域里进行管理。
结构变量	用来组合标准数据类型到一个新的数据结构。这些数据类型形成一个逻辑单元。
编辑器	包括图形系统（图形编辑器）、动作（脚本）处理（全局脚本）、消息系统（报警记录）、测量值归档和编辑（变量记录）、报表系统（报表编辑器）、用户授权（用户管理器）以及文本库。这些模块均为 WinCC 系统的一部分，但并非所有模块均需安装。同样可以安装选项，编辑器，即设备状态监控、基础数据和时间同步。

可以通过打开弹出式菜单，选择“属性”选项，编辑项目组件“计算机”和“变量管理器”的属性。

用户可在浏览窗口中进行如下移动：

- 1) 在浏览窗口中，单击加号和减号来打开或者关闭附加的层。

2) 双击名称打开或关闭附加的层 (例如, 双击“计算机”打开下一层)。

可使用鼠标或键盘在浏览窗口中移动。键盘命令符合 Windows NT 的规则 (例如, 箭头键、数字键盘上的加号和减号)。位置的每次改变都将触发立即更新屏幕右边的数据窗口。

(6). 数据窗口 数据窗口的内容根据浏览窗口中所选组件的不同而改变。数据窗口位于 WinCC 资源管理器的右边。数据窗口中各列的含义如下:

列	含义
名称	此列显示对象的名称, 一个变量或一个文件 (文件名已被分配给指定的编辑器)。
类型	此列显示对象的类型: 例如, 假设是计算机, 此列会包含“服务器”; 若是变量, 则显示每个变量的数据类型。
命令行参数	此列显示外部变量的地址描述: 例如, 数据块号码 (DB) 和数据字地址 (双字)。关于内部变量, “内部变量”输入在此位置。
最新改变	此列显示上次改变的日期和时间。

可以使用“名称”、“类型”“参数”以及“修改时间”按钮以升序或降序对数据窗口的内容进行排序。

可在数据窗口里进行如下的移动:

- 1) 双击名称打开列表或对象。
- 2) 选择一个对象并单击鼠标右键, 会打开弹出式菜单。
- 3) 如果在窗口的空白区域单击鼠标右键, 会打开浏览窗口中相应列表元素的弹出式菜单。

3. 项目类型

WinCC 中的工程项目分为 3 种类型: 单用户项目、多用户项目和客户机项目。项目包括“计算机”、“变量管理器”、“编辑器”等组件。下面对该部分涉及到的几个术语进行描述。

客户机: 在多用户项目中被永久分配到服务器的客户机。客户机能被用于多用户的项目或一个分布式的系统。

多客户机: 在 WinCC V5.0 中, 一个多客户机可以访问多达 6 个服务器的数据。多客户机不是服务器项目的组件。

服务器: 带客户机和多客户机的多用户项目的服务器。冗余服务器组也能代表一个服务器。

功能分区: 各种服务器在指定的过程区域中承担不同的任务。多客户机项目中涉及到的各个服务器执行不同的任务。例如, 一个 WinCC 服务器执行归档, 另一个执行消息处理, 而第三个则建立过程数据连接。

技术分区: 不同的服务器承担全部必需的指定区域的任务。涉及到的各个服务器执行相同的任务, 例如消息归档、测量值归档以及建立过程数据连接。然而每个服务器被认为是不同的逻辑系统区。在被周围子系统结构化的应用程序中, 服务器被连接到不同的彼此独立的 PLC 上。在 PCS7 中, 逻辑分区的形式被首先使用。

(1). 单用户项目 单用户项目是一种只拥有一个操作终端的项目类型。在此计算机上可以完成组态、操作、与过程总线的连接以及项目数据的存储。项目的计算机既用作进行数据处理的服务器, 又用作操作员的输入站。其他计算机不能访问该计算机上的项目 (通过 OPC 等访问的除外)。

单用户项目可与多个控制器建立连接。在单用户项目计算机所在的自动化网络中, 一般

只有一台 PC 机。如果有多台 PC 机，则 PC 机上的数据也是相互独立的，不可通过 WinCC 进行相互访问。

如果只希望在 WinCC 项目中使用一台计算机进行工作，可创建单用户项目，运行 WinCC。

(2). 多用户项目 多用户项目的特点是同一项目使用多台客户机和一台服务器，在此最多可有 16 台客户机访问一台服务器，可以在服务器或任意客户机上组态。项目数据，如画面、变量和归档，最好存储在服务器上，并且使它们能被所有客户机使用。服务器执行与过程总线的连接和过程数据的处理，运行系统通常由客户机控制。任意一台客户机可以访问多台服务器上的数据，任意一台服务器上的数据也可被多台客户机访问。

如果希望在 WinCC 项目中使用多台计算机进行协调工作，则可创建多用户项目。在服务器上创建多用户项目，与 PLC 建立连接的过程通信只在服务器上进行，而客户机没有与 PLC 的连接。

(3). 多客户机项目 多客户机项目是一种能够访问多个服务器的数据的项目类型。每个多客户机和相关的服务器都拥有自己的项目。其功能是：在服务器或客户机上完成服务器项目的组态；在多客户机上完成多客户项目的组态。

最多 16 个客户机或多客户机能够访问服务器。在运行时多客户机能访问至多 6 个服务器。也就是说，6 个不同的服务器的数据可以在多客户机上的同一幅画面中可视化显示。

四. 图形编辑器

图形编辑器可以用来创建过程图。

1. 浏览窗口的快捷菜单 右击 WinCC 项目浏览器的“图形编辑器”，将弹出快捷菜单，如图 1-2 所示，单击“打开”菜单项，打开图形编辑器，并新建一个画面。

2. 画面名称的快捷菜单 选择 WinCC 项目浏览器的图形编辑器，在数据窗口中右击任一画面，将弹出快捷菜单，如图 1-3 所示，单击“打开画面”菜单项，打开图形编辑器。



图 1-2 浏览窗口的快捷菜单

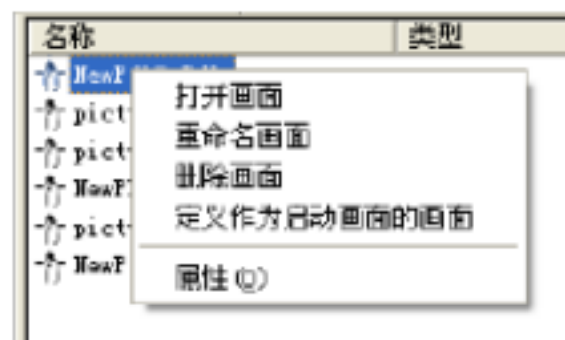


图 1-3 画面名称的快捷菜单

3. 图形编辑器的布局 图形编辑器的布局如图 1-4 所示。

用于操作图形编辑器的选项板和栏的功能如下：

菜单栏：菜单栏包含图形编辑器中所有可用的菜单命令，不能激活的命令以灰色显示。

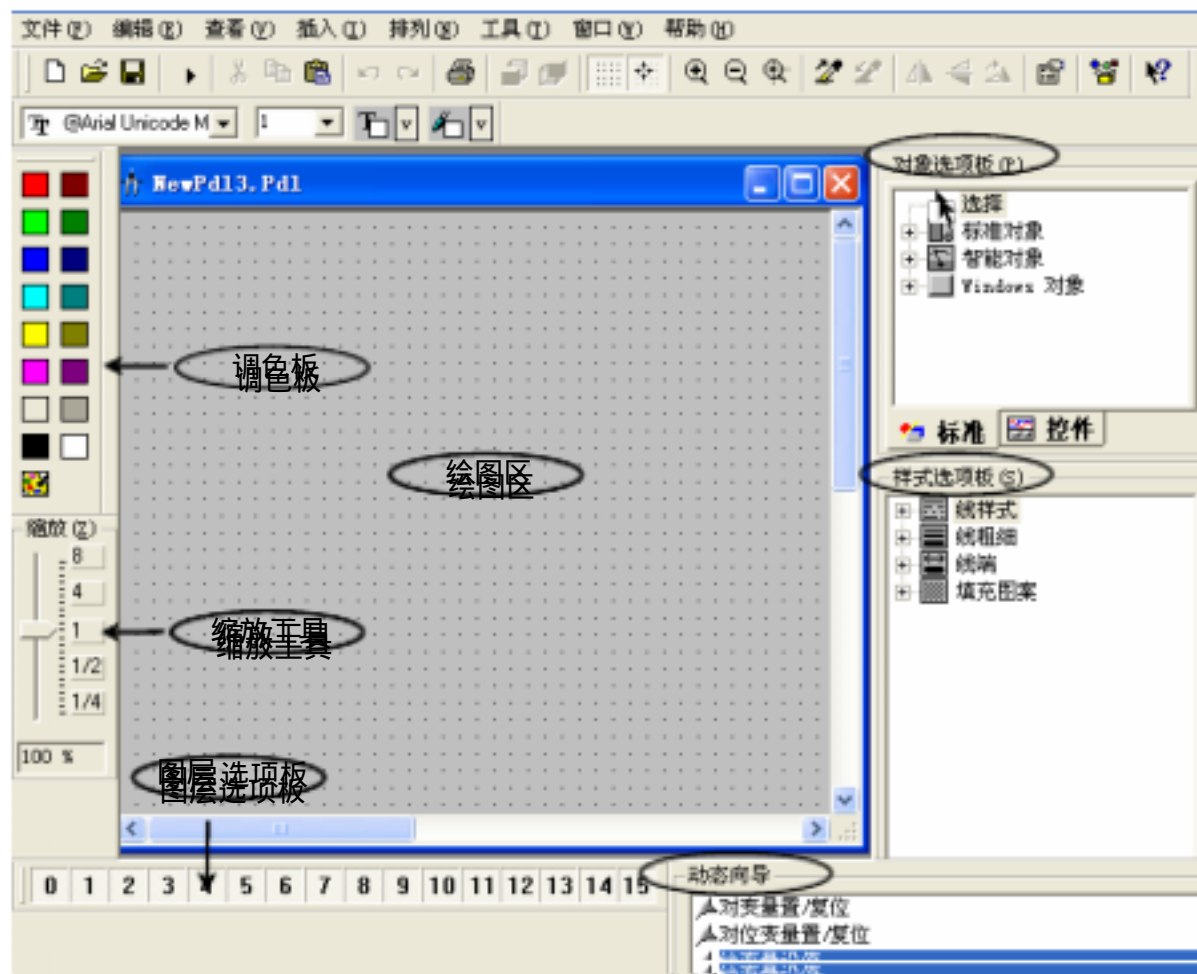


图 1-4 图形编辑器窗口

如果命令后面有三个点（省略号），将打开一个设置附加参数的对话框窗口。

操作方法与标准 Windows 操作类似。

标准工具栏：用鼠标单击标准工具栏包含的图标可以执行一般命令。标准工具栏是可组态的。为了添加或从标准工具栏删除按钮，可按住“Alt”键，并双击任一按钮，“改编工具栏”对话框将打开，可以按需要改编标准工具栏。

状态栏：除了常规程序信息以外，状态栏还显示所选对象的名称和坐标位置。

层面栏：在层面栏中，单击 16 层（层 0 到层 15）中应切换到可见状态的层，对象建立在第一图形可见层中。在图形编辑器中为该目的而保留层 0。系统不允许关闭所有的层。

使用“层...”命令设置层为激活的或非激活的。该操作在被设置的不同层上显示或隐藏对象。对象可通过“对象属性”被分配到某个层面上。

4. 设置和改变对象属性的对话框窗口 “对象属性”窗口显示所选对象、对象组的所有属性或对象类型的默认设置，可以改变或复制这些属性。此外，在这里还可以通过设置相应属性的方法使对象动态化，并且可以与动作相链接。

“对象属性”窗口由下列条目组成：

图钉：用来固定窗口。

吸管：用来复制和分配属性。

对象列表：“属性”标签和“事件”标签。

5. 选项板 用于创建和编辑图形对象，其功能如下：

颜色调色板：使用鼠标分配面向对象的 16 种标准颜色之一，即一种基本的颜色或一种自定义的颜色。

对象选项板：使用对象选项板选择将在图形工作空间的第一可见层面创建的对象。0 层为图形编辑器保留。

对象选项板包含“标准”和“控件”标签。“标准”标签中可用的对象由对象组来组织。

“控件”标签提供 ActiveX 控件。该标签的内容可以由用户更改。

“标准”标签：标准对象，例如多边形、椭圆和矩形；智能对象，例如控件、元素、棒图和 I/O 域；Windows 对象，例如按钮和复选框。

样式选项板：使用样式选项板改变选定对象的外观。

对齐选项板：使用对齐选项板，可改变一个或多个对象的绝对位置。改变所选择的对象相互之间的位置关系，使多个对象的高度和宽度一致。

缩放选项板：用滚动条或按钮可设置进入工作状态的窗口的缩放因子。当前缩放因子显示在滚动条下方。也可以通过标准工具栏功能一步步设置缩放。

文本选项板：使用字体选项板快速改变文本对象的字体和字体颜色以及标准对象的线颜色。

五、创建项目的步骤


第一步：准备工作。

创建项目前应对项目的结构给出一些初步的考虑。可从如下几个方面进行：

1. 项目类型 在开始创建项目前，应清楚创建的是单用户项目，还是多用户项目。
2. 项目路径 可将 WinCC 项目创建在一个单独的分区上，不要将 WinCC 项目放在系统分区上。

3. 项目名称 建议在创建项目前就确定合适的名称。因为一旦完成项目的创建，再对项目的名称进行修改就会涉及许多步骤。

第二步：指定项目的类型。

单击 WinCC 项目管理器工具栏上的  按钮，或单击“文件”菜单上的“新建”，或使用组合键“Ctrl + N”，可打开“WinCC 资源管理器”对话框，如图 1-5 所示。

选择所需要的项目类型，并单击“确定”按钮，即打开“创建新项目”对话框，如图 1-6 所示。



图 1-5 指定项目的类型

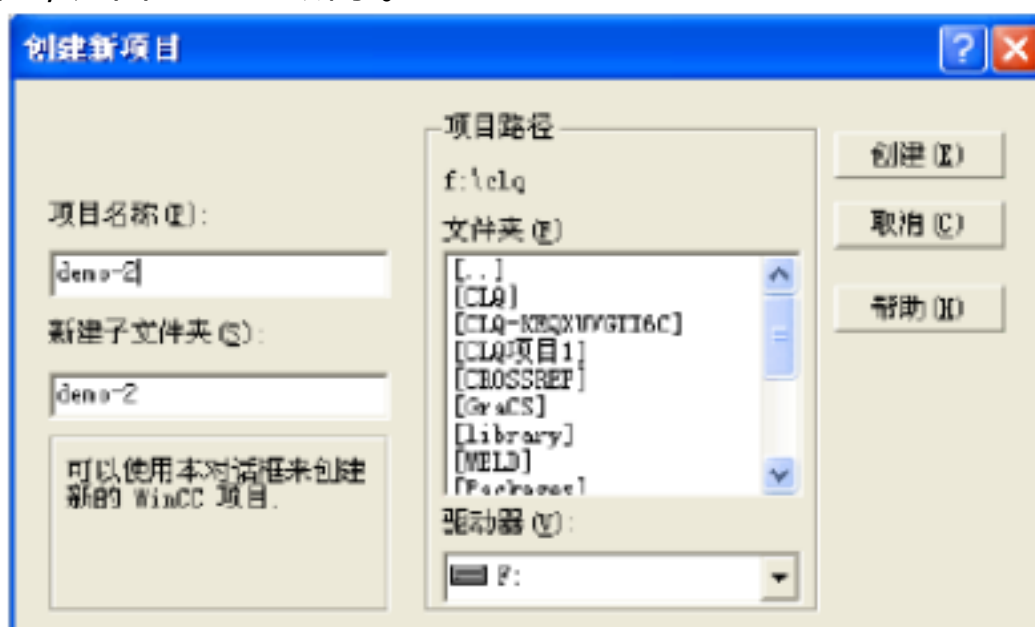


图 1-6 “创建新项目”对话框

第三步：指定项目名称和项目存放的文件夹。

在图 1-6 所示的“创建新项目”对话框中输入项目名称和完整的项目存放路径，然后单击“创建”按钮。

第四步：更改项目的属性。

单击 WinCC 项目管理器浏览窗口中的项目名称，并在快捷菜单中选择“属性”菜单项。打开“项目属性”对话框，如图 1-7 所示。

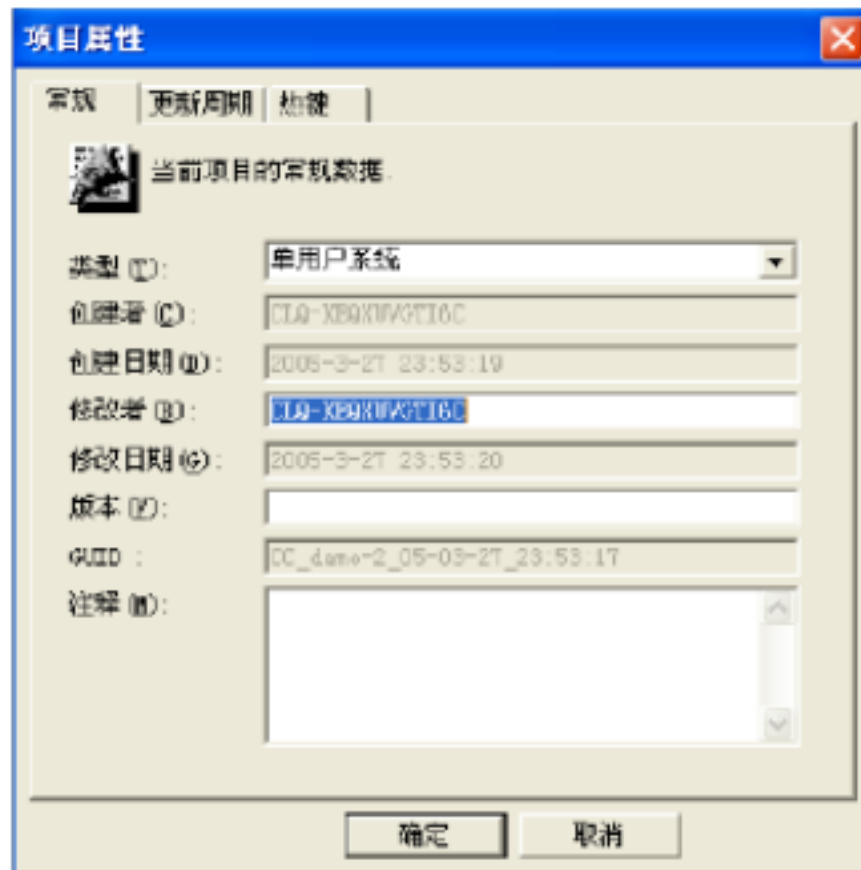


图1-7 “项目属性”对话框

在“项目属性”对话框中，可修改项目的类型、修改者及版本等内容。

在“更新周期”选项卡上，可选择更新周期，并可定义五个用户周期。用户周期的时间为可选择。

在“热键”选项卡上，可为 WinCC 用户登录和退出定义热键。

第五步：更改计算机的属性。

打开“计算机属性”对话框。

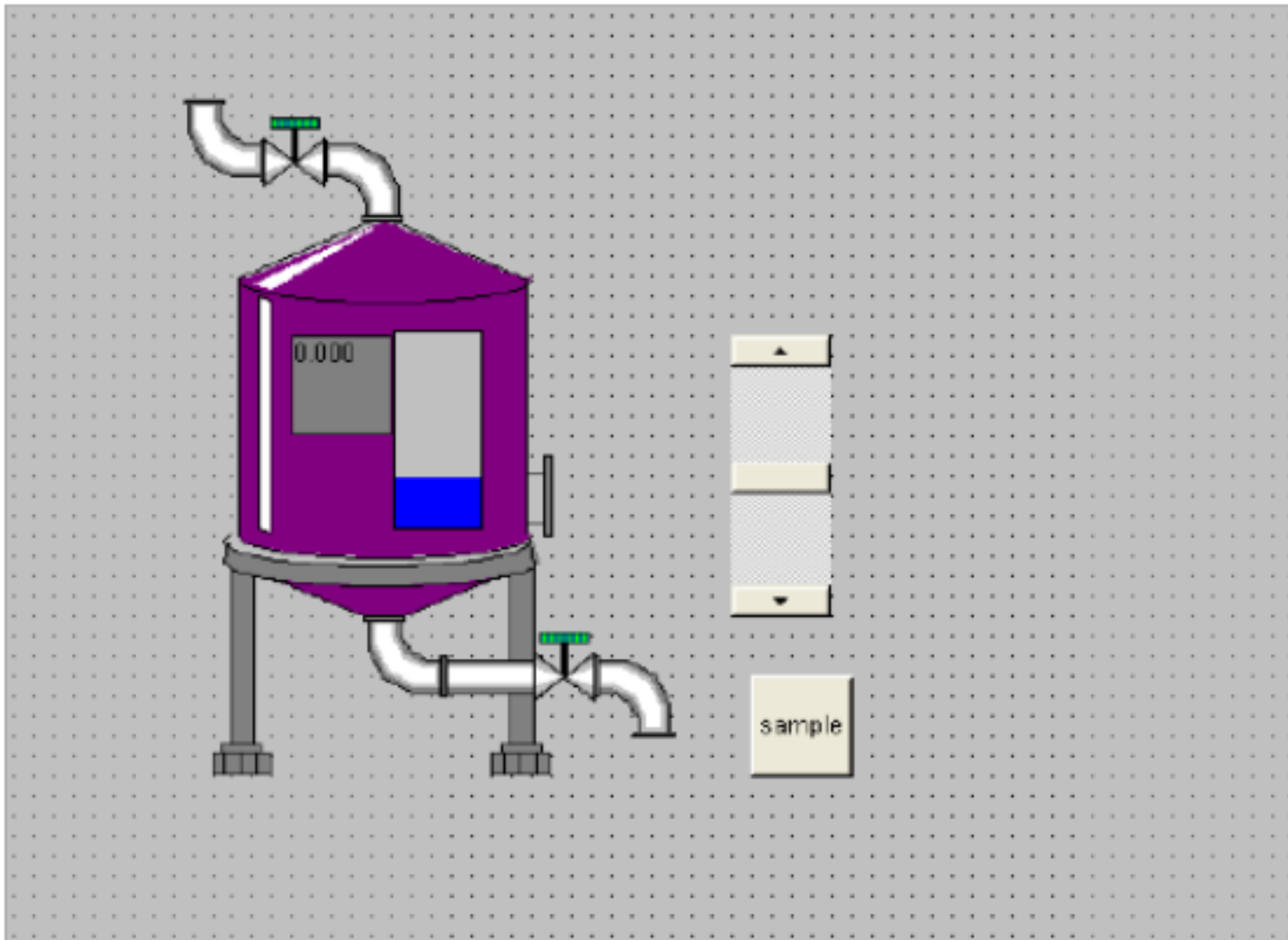
在“常规”选项卡上，检查“计算机名称”输入框中是否输入了正确的计算机名称。此名称应与 Windows 的计算机名称相同。

如果创建了一个多用户项目，则“计算机类型”可指示此计算机组态是服务器还是客户机。单击“确定”按钮，关闭对话框。

如果对项目中的计算机名称进行了修改，则必须关闭再重新打开项目才能生效。

六、实验内容：

创建一个项目名称为 Qckstart 的项目；在图形编辑器（名称为 START.pdl）中，编辑如下所示图形。



一、实验目的：

1、掌握 WinCC 变量的创建。

2、熟悉掌握按钮的直接连接、库元件的创建、创建和动态话 I/O 域、创建和动态化滚动条等。

二、WinCC变量

要在变量管理器中创建 WinCC 变量，首先必须组态一个与 PLC 的连接。但是，不必安装硬件，安装所期望的通信驱动程序并组态期望的连接就足够了。创建变量组与变量的具体过程如下：

1．安装新的驱动程序。这可以通过鼠标右键单击变量管理器，并从弹出式菜单中选择添加驱动程序来完成，如图 2 - 1 所示。

2．选择所期望的驱动程序。从所显示的对话框中，通过单击“打开”按钮将驱动程序插入 WinCC 项目中，WinCC 资源管理器即可将驱动程序条目显示在变量管理器中，而不是仅显示内部变量。例如建立 S7 PLC 与 WinCC 的连接，如图 2-2 所示。

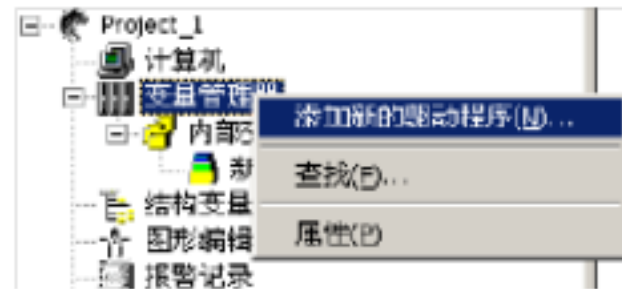


图 2 - 1 创建新的驱动程序

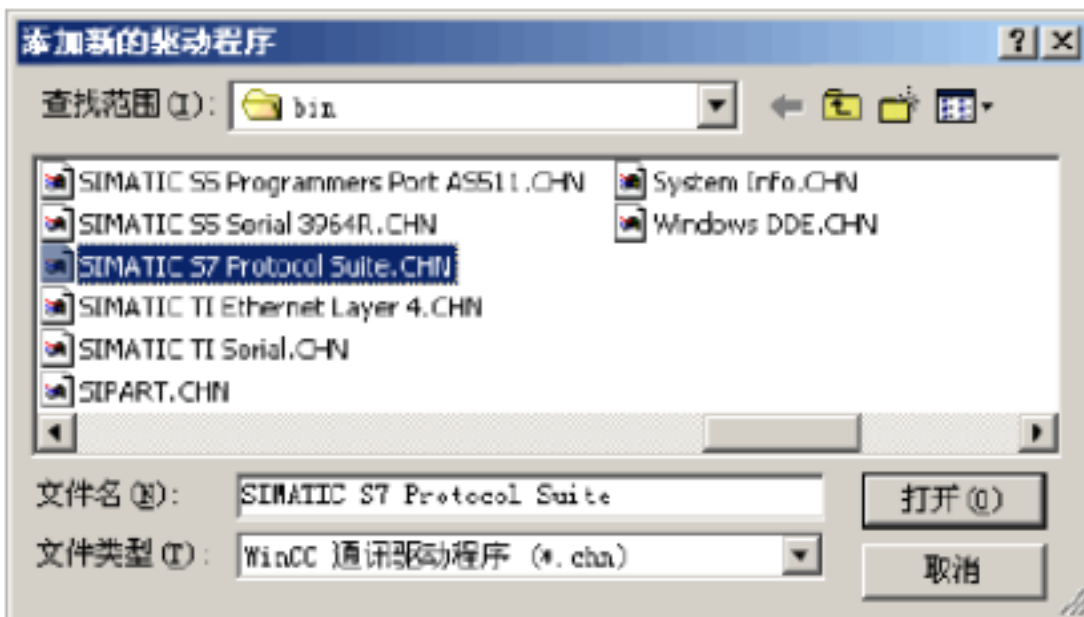


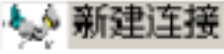
图 2-2 创建 S7 PLC 的驱动程序

3．通过鼠标右键单击“新驱动程序”连接条目，可显示一个或几个子条目（所谓的通道单元）。接下来就可以创建一个连接，这通过鼠标右键单击“通道单元”条目并从弹出式菜单中选择新建驱动程序连接来完成，如图 2-3 所示。



图 2-3 新建驱动程序连接

在图 2-3 所显示的对话框中，在常规信息标签内为连接分配一个名称。连接参数可通过单击“属性”按钮来进行设置。

4. 通过鼠标右键单击“新添加的连接”条目 ，可按上面所描述的方式添加变量组或变量。在创建 WinCC 变量时，除定义内部变量所需的设置外，还必须定义地址和格式调整设置，地址需要参考 PLC 中变量地址。

二、按钮事件的直接连接

在图形编辑器中的实现（直接连接）

在图形编辑器中实现直接连接如图

2-4 所示。

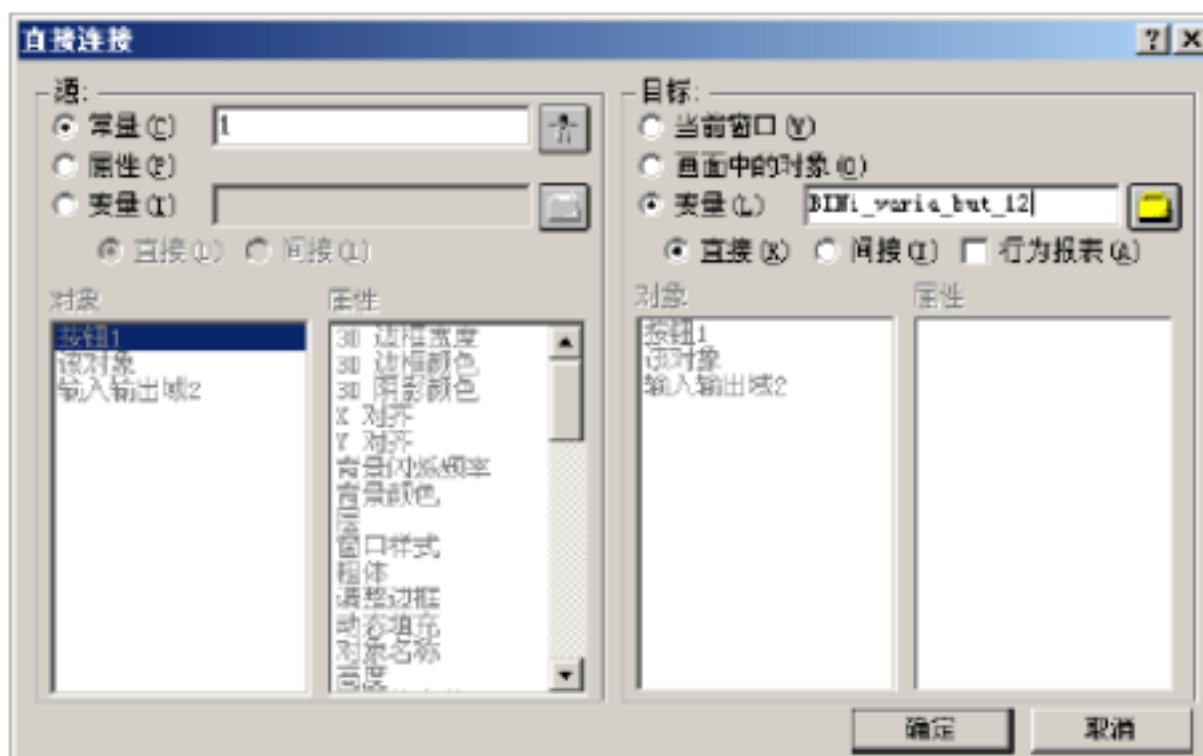


图 2-4 直接连接变量

1. 在变量管理器中创建一个二进制类型的变量。例如使用 BINi_varia_but_12 变量。
2. 在某个画面中，组态 Windows 对象中的“按钮”。
3. 在“对象属性”对话框的“事件”“鼠标”“按左键”处为该按钮组态了一个直接连接。将“源常量”设置为“1”并与“目标变量”“BINi_varia_but_12”连接，单击确定按钮即可应用这些设置。在“对象属性”对话框的“事件”“鼠标”“释放左键”处为该按钮组态另一个直接连接，但这次将“源常量”设置为“0”。
4. 通过 BINi_varia_but_12 变量对动画进行控制。

三、创建和动态话 I/O 域、动态化滚动条

1. 在变量管理器中创建变量，例如使用变量 M。
2. 在画面中对智能对象中的 I/O 域进行组态。在组态对话框中设置变量 M，在更新域中将默认值“2s”修改为“根据变化”，并将域类型设置为“输出”，如图 2-5 所示。



图 2 - 5 I/O 域组态

3. 在同一画面中，组态 Windows 对象中的“滚动条”对象。在“对象属性”对话框的“事件”“其他”“对象改变”处创建一个直接连接，如图 2-6 所示。

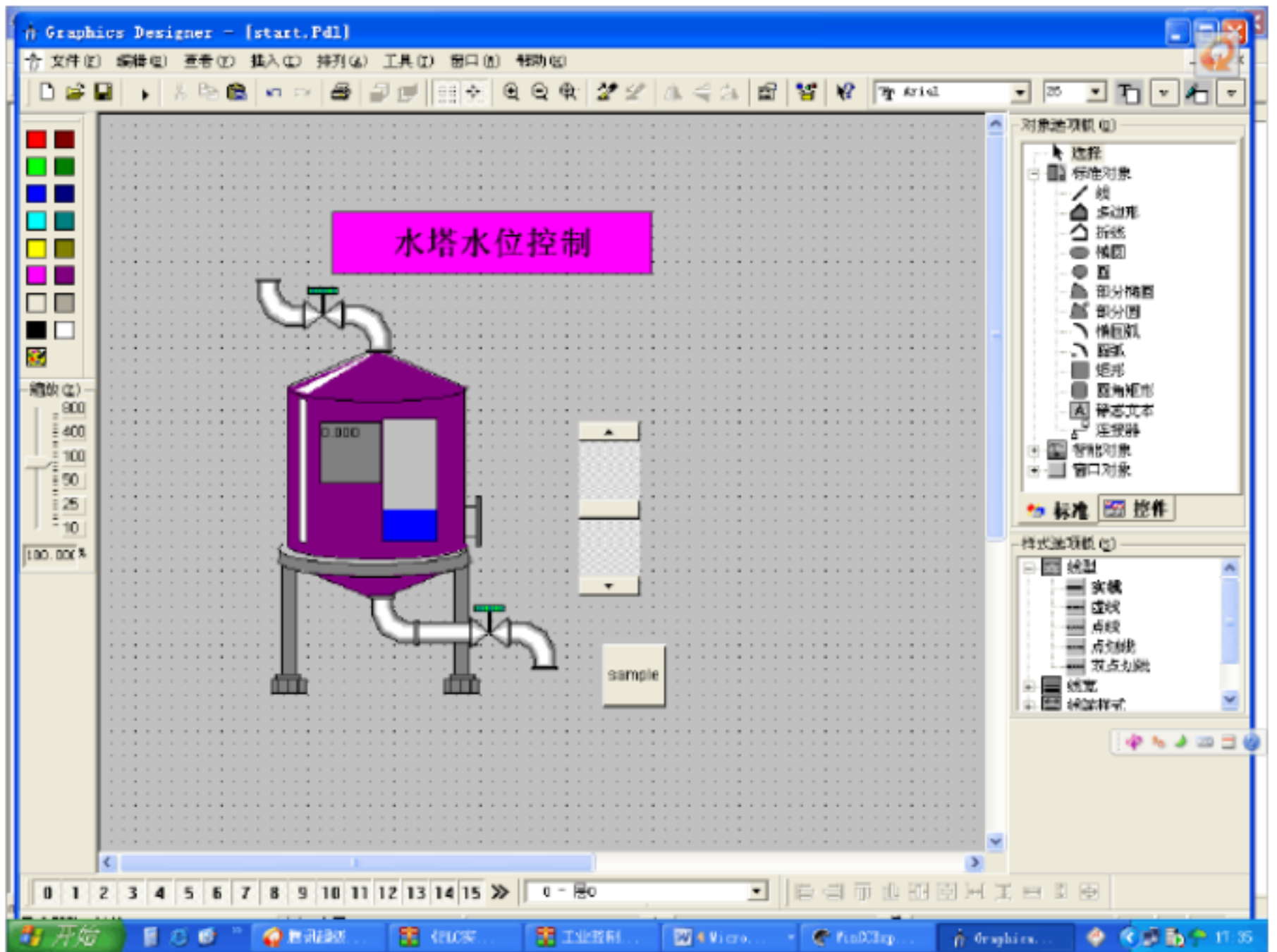


图2-6 滚动条对象属性

4. 在直接连接对话框中，将“源对象”“过程驱动连接程序”与“目标变量”“变量 M”相连接。单击确定按钮即可应用这些设置。

四、实验要求及内容

建立变量名为“TankLevel”的“无符号 16 为数”的内部变量；创建两个名称为“START.pdl”和“SAMPLE.pdl”的画面。在图形编辑器中完成如图所示画面。通过模拟测试时，滚动条由最低滑动到最高时，水塔的水变满，同时塔身上的 I/O 域窗口中的数字由 0 变到 100，当点击 SAMPL按钮时，画面切换到“SAMPLE.pdl”的画面。



实验三 画面组态

一、实验目的：

熟悉和掌握画面切换、显示画面窗口、操作控制权限等方法和步骤

二、画面切换的方法与步骤

通过直接连接打开画面

1. 说明 在画面窗口中，通过按钮及借助直接连接，可完成画面切换。

使用三个“ Windows 对象 ”选项板中的“ 按钮 ”，当将按钮按下时，即可将该按钮对应的画面显示在画面窗口中，通过按不同的按钮，即可进行画面切换。“ 画面窗口 ”从“ 智能对象 ”选项板中选取，在画面中，所用的“ 静态文本 ”从“ 标准对象 ”选项板中选取，静态文本用于说明画面切换后画面窗口中所显示的画面名称。

2. 实现步骤

1) 打开“ 图形编辑器 ”。

2) 通过“ 文件 ”菜单中的“ 新建 ”菜单条目，创建一个新的画面，并通过“ 文件 ”菜单中的“ 另存为 ... ”菜单条目，将其以名称 pictu_5_kzz_00.pdl 进行保存。将鼠标指向新建画面，单击鼠标右键，在弹出的菜单中选择“ 属性 ”菜单条目，打开“ 对象属性 ”对话框（见图 3-1）。如图 3-1 所示将“ 属性 ”标签下的“ 几何 ”部分中的“ 画面宽度 ”设置为 270，“ 画面高度 ”设置为 280。

3) 在 pictu_5_kzz_00.pdl 画面中，组态一个静态文本。鼠标点击该“ 静态文本 ”对象，单击鼠标右键，在弹出的菜单中选择“ 属性 ”菜单条目，打开“ 对象属性 ”对话框，如图 3-2 所示。将“ 属性 ”标签下的“ 字体 ”部分中的“ 粗体 ”设置为“ 是 ”，将“ 文本 ”的默认内容从静态列中删除，这样可以避免建立画面时输出不正确的文本。

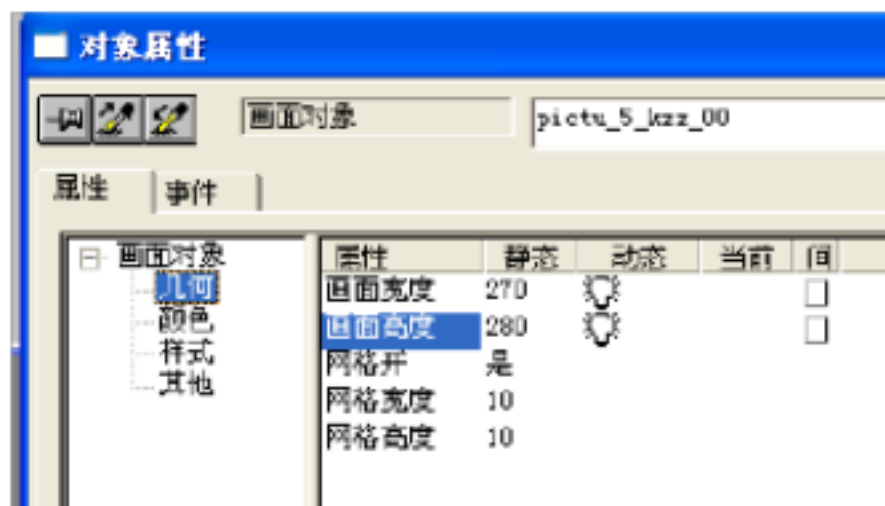


图 3-1 新建画面对象属性对话框

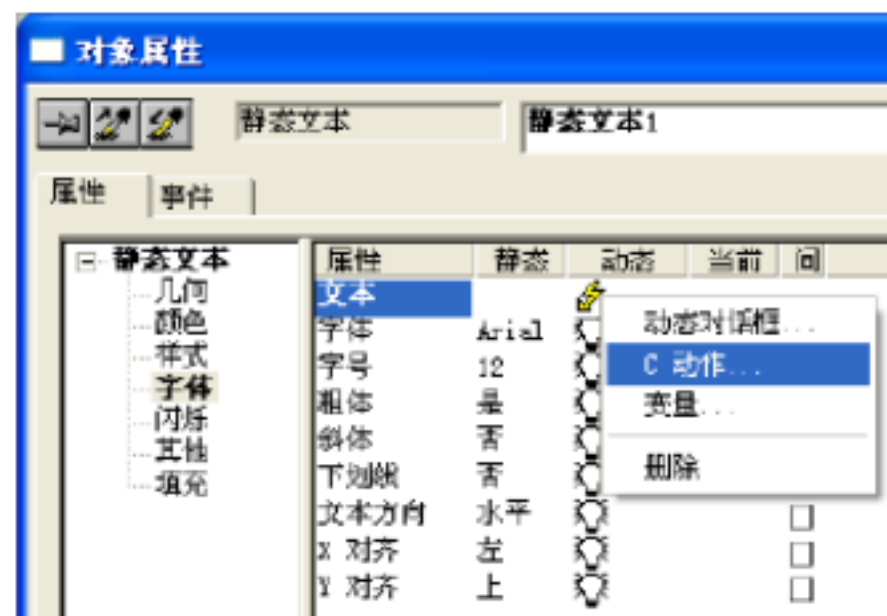


图 3-2 静态文本对象属性对话框

因“ 静态文本 ”中显示的是“ 画面窗口 ”中所显示的画面名称，随着画面的切换，其所显示的名称也将随着改变。欲达到此目的，可在“ 静态文本 ”中使用 C 动作，该 C 动作把当前画面名称作为返回值返回。作用于 C 动作的触发器，使用默认周期 1 小时（低系统负载，不需任何修改）。在本例中，建立 C 动作的过程如图 3-2 所示，选取“ 字体 ”条目下的“ 文本 ”属性，指向该属性的“ 动态 ”列，单击鼠标右键，在弹出的菜单中选择“ C 动作 ... ”菜单条目，然后在弹出的界面中输入、编译 C 动作程序段。静态文本的 C 动作程序段如下：

```
# include " apdefap,h "  
char* _nain(char* lpszpictureName, char* lpszobjectName,char* lpszPropertyName)  
{
```



```

Char *name =lpszPictureName;
Char *pdest;
int ch = " : " ;
//check if picture path contains char
Pdest=strrchr(lpszPictureName,ch);
//read only picture name without path
If (pdest ==NULL) return lpszPictureName;
else {
    name=strcpy(name,strrchr(name,ch)+1);
    return name;
}
}

```


4) 在 pictu_5_kzz_00.pdl 画面中，组态全局库中的“反应器 6”对象作为所要显示的信息。组态的方法为：首先通过“查看”菜单中的“库”菜单条目或通过工具栏上的  按钮打开库浏览，如图 3-3 所示；然后将“反应器 6”对象拖到画面适当位置，即完成组态。



图3-3 库浏览

5) 组态两个以上用于画面切换的画面，简便的方法可通过利用“文件”菜单中的“另存为...”菜单条目完成。一个画面名称为“pictu_5_kzz_01.pdl”，另一个画面名称为“pictu_5_kzz_02.pdl”。在这两个画面中分别组态期望显示的内容。画面名称显示在静态文本中，画面切换时不需要改变静态文本对象。

6) 通过“文件”菜单中的“新建”菜单条目，创建一个新的画面，在该画面中，组态一个画面窗口。修改画面窗口的尺寸，使其大小与先前创建的画面大小一致。为了使窗口在运行系统中带边框显示，可将“画面窗口”的“边框”属性设置为“是”。

7) 在同一画面中，组态一个按钮。打开“按钮”的“对象属性”窗口，在窗口中选取“对象属性”对话框的“事件”条目，如图 3-4 所示。然后选“鼠标”，在“执行条件”中选“按左键”，在“按左键”所对应的“动作”处创建一个直接连接。在“直接连接”对话框中，将画面 pictu_5_kzz_00.pdl 选为源。在“目标”部分，选取“画面中的对象”，在“画面中的对象”列表中选“画面窗口 1”。

8) 复制按钮 1 对象，生成两个按钮：按钮 2 和按钮 3。更改按钮 2 和按钮 3 所组态的“直

接连接”的“源”属性，对按钮 2，将其设置为“ pictu_5_kzz_01.pdl ”；而对按钮 3，则设置为“ pictu_5_kzz_02.pdl ”。

9) 将启动画面设置为“ pictu_5_kzz _00.pdl ”画面。方法为：首先在 WinCC 的资源管理器中选择“计算机”，弹出“计算机属性”窗口，在该窗口中选择“图形运行系统”；然后在“启动画面”窗口内输入或通过“浏览 (B) ... ”按钮选取“ pictu_5_kzz _00.pdl ”，结果如图 3-5 所示。

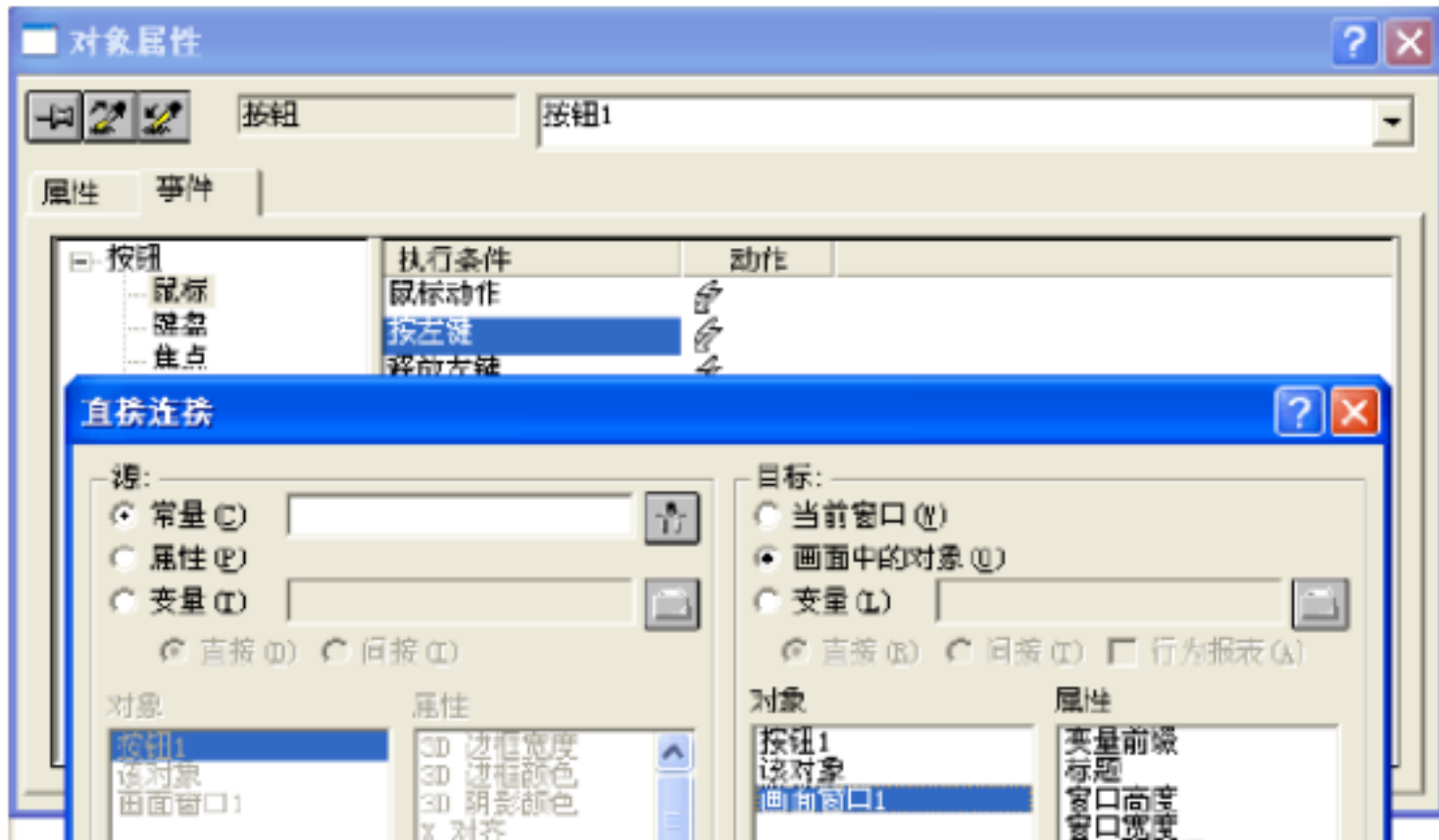


图3-4 设置按钮属性



图3-5 设置启动画面

三、 显示画面窗口的方法与步骤

使用动态向导对信息框进行组态

1. 说明 如果变量超过数值 100，则显示一个信息（指令）框；如果该变量值超过 150，则显示紧急框。在实例中使用一个滚动条对象以输入变量值，使用一个 I/O 域来显示变量值。

2. 实现步骤

1) 如果没有显示动态向导，则从菜单“查看”“工具栏...”里将其激活。

2) 新建一个画面，在画面中，组态一个 I/O 域（选自“智能对象”）。在画面中，用鼠标选中 I/O 域对象，然后选择动态向导中的“画面函数”标签，从“画面函数”中选择“显示错误信息框”条目。用鼠标左键双击该条目，按照动态向导的指令完成设置。在选择触发器时，选择“鼠标左键”列表条目；在选择附加参数时，选择“信息框”并输入显示的文本。

3) 为 I/O 域再次使用动态向导。在选择触发器时，选择“鼠标右键”列表条目；在选择附加参数时，选择“紧急框”并输入显示的文本。

4) 在变量管理器中创建一个“有符号的 32 位数”类型的变量。在本实例中，使用了 S32i_pictu_boxes_00 变量。

5) 在同一画面中组态一个“滚动条”对象（选自“Window 对象”）。在滚动条对象的“对象属性”对话框的“事件”“属性主题”“其他”“过程驱动程序连接”处为滚动条对象创建一个直接连接。在如图 3-6 所示的“直接连接”对话框中将“源”“属性”“滚动条对象 1”“过程驱动程序连接”与“目标”“变量”“S32i_pictu_boxes_00”相连接。单击“确定”按钮完成设置。

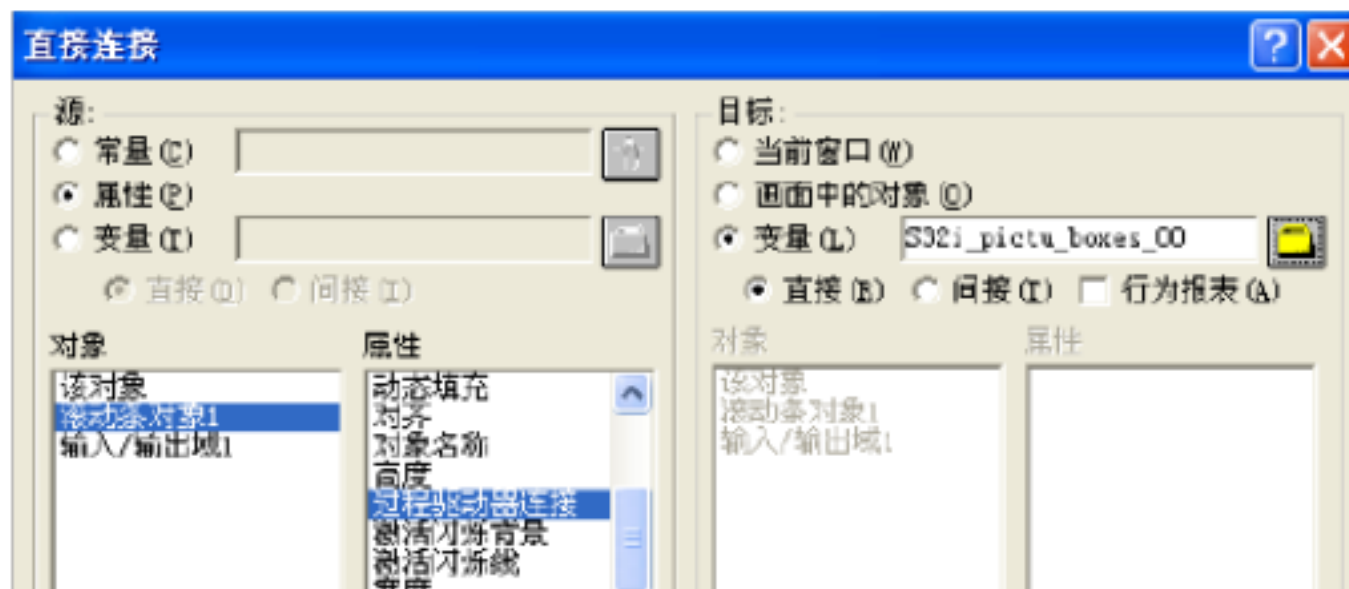


图3-6 “直接连接”对话框

6) 打开 I/O 域对象的“对象属性”对话框，如图 3-7 所示。在“属性”“输入/输出域”“输出/输入”“输出值”上创建一个动态“变量”，并连接给变量 S32i_pictu_boxes_00，并选择“根据变化”进行触发。在“对象属性”对话框的“事件”“属性主题”“输出/输入”“输出值”处为 I/O 域对象创建一个 C 动作，如果 S32i_pictu_boxes_00 变量值超出 100，则显示一个信息框，如果超出 150，则显示一个紧急框。在第 2) 步中由动态向导为 I/O 域对象所产生的按下左键和按下右键的 C 动作可复制和粘贴到该 C 动作中。

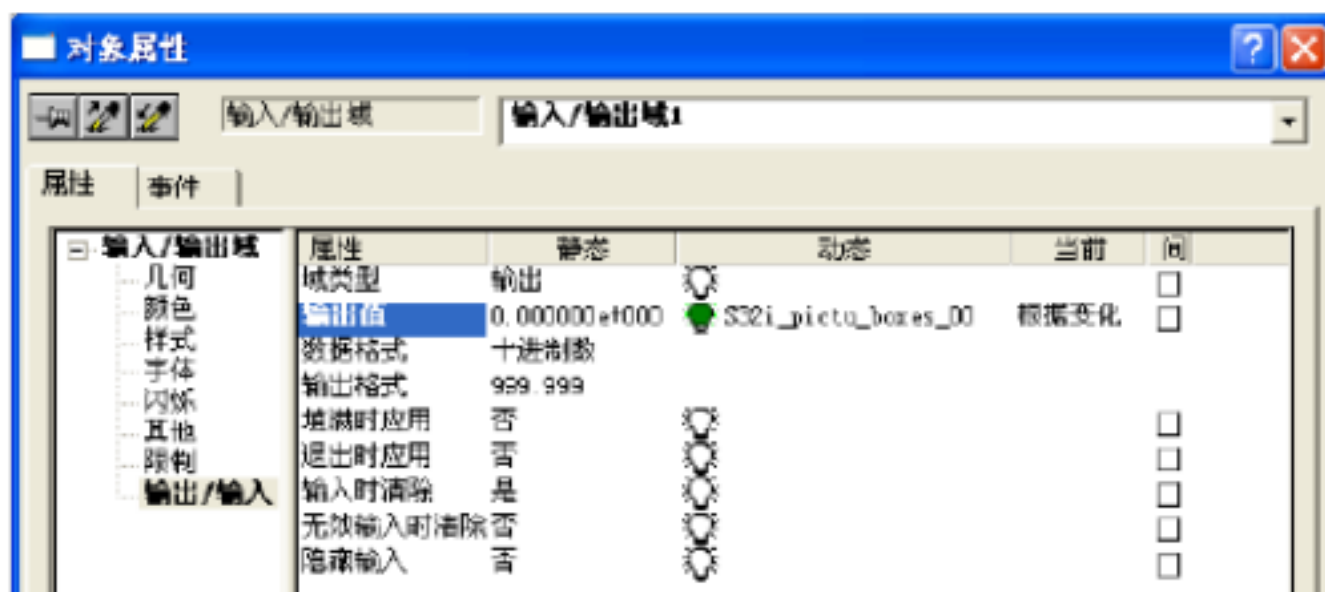


图3-7 I/O 域对象的“对象属性”对话框

I/O 域的 C 动作说明：使用内部函数 GetTagDWord 读入变量值。如果超出 100，则显示信息框；如果低于 100，则静态 C 变量 i 复位为 0；如果变量值超出 150，则显示紧急框；如果低于 150，则静态 C 变量 j 复位为 0。程序段如下：

```
#include "apdefap.h"

Void onpropertyChanged (char* lpszpictureName, char*lpszobjectName,char* lpszPropertyName)
{
    int a;
    Static int i= 0, j =0;
    //get tag value
    a=GetTagDword( " S32i_pictu_boxes_00 " );
    //set visible info box
    if ((a>100)&&(i= =0)){
        i=1;
        MessageBox(NULL, " Der Variablenwert hat\r\n100 überschritten " ,
            " Hinweis " ,MB_OK|MB_ICONEXCLAMATION|MB_SETFOREGROUND);
    }//if
    if (a<=100) (i=0);
    //set visible emergency box
    if ((a>150)&&(j= =0)){
        j=1;
        MessageBox(NULL, " Der Variablenwert hat\r\n150 überschritten " ,
            " Achtung!!! " , MB_OK|MB_ICONSTOP|MB_SETFOREGROUND);
    }//if
    if (a<=150) (j=0);
}
```

7) 在 I/O 域对象的“对象属性”对话框中，选择“对象属性”对话框的“事件” “鼠标” “按左键”和“按右键”处的 C 动作进行删除。如果变量值超出 100，则使用由动态向导所产生的 C 动作来显示信息框。

四、操作控制权限的方法与步骤

根据用户授权执行画面切换

1. 说明 通过两个按钮，只有当用户具有相应的授权时，才执行画面切换。为了实现该任务，将使用两个“Windows 对象”类中的按钮，当使用鼠标将按钮按下时，每个按钮可在“智能对象”类中的画面窗口中显示不同画面。在用户管理器编辑器中，可完成对用户权限进行分配所需的设置。

2. 实现步骤

1) 在 WinCC 资源管理器中，选择“用户管理器”，单击鼠标右键，从弹出的菜单中选择“打开”，将用户管理器编辑器打开，如图 3-8 所示。

2) 通过用户管理器，创建一个新的用户组，并为其分配一个名称：在本实例中，将使用名称 service。

3) 通过“表格”添加新的授权等级，选择授权等级“改变画面”。该授权等级将分配给 service 组。分配给组或用户的授权等级由靠近授权列的红点来识别。

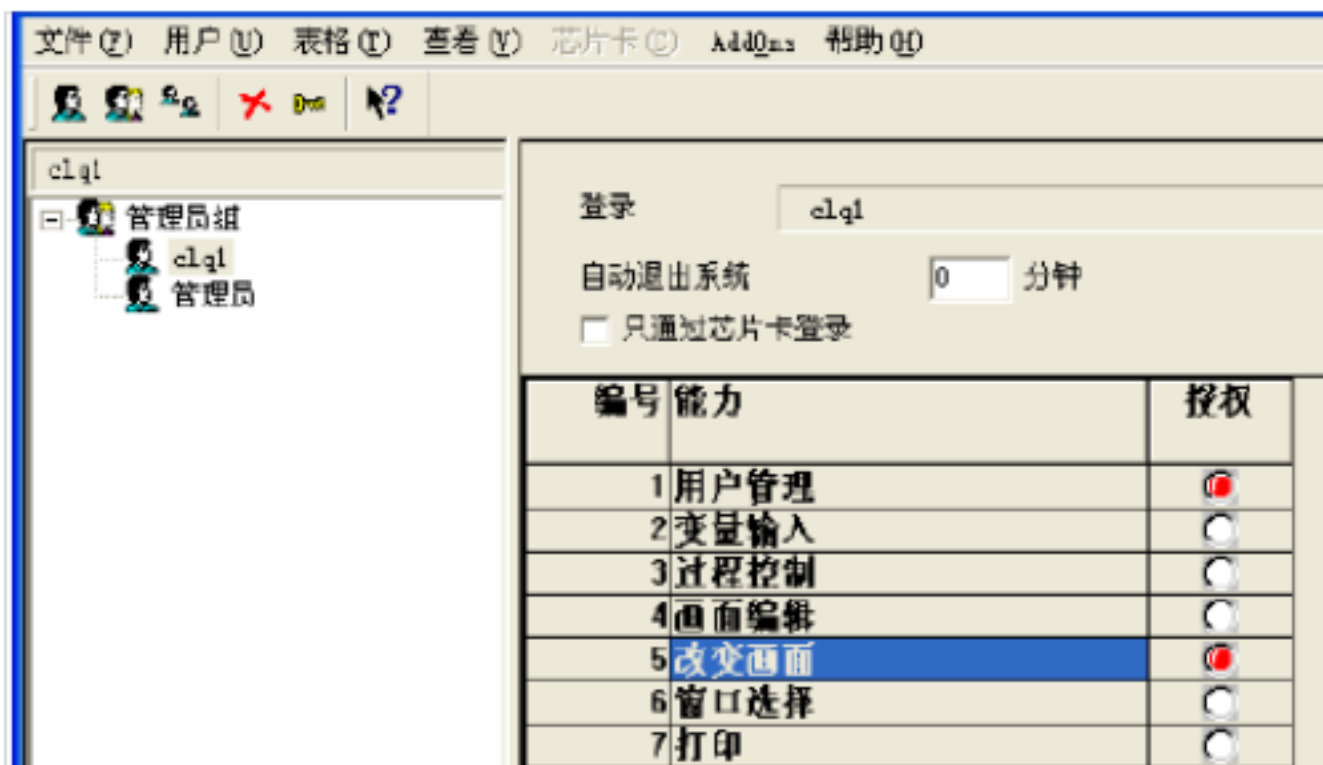


图 3-8 用户管理器编辑器

4) 通过用户管理器，为 service 用户组创建一个新的用户。在同一项目中，已创建了一个名称为 willi 的用户，该用户带有口令 Project_CreatePicture。激活同时复制组设置复选框，把可用于这个组的授权等级传送给用户。通过“文件”“退出”菜单，可关闭用户管理器编辑器。

5) 在 WinCC 资源管理器中，通过在项目名称上单击鼠标右键打开项目属性对话框。选择“热键”标签，并完成用于调用登录和退出对话框的期望设置。

6) 创建一个画面，在该画面中，组态两个“按钮”（Windows 对象）。在本实例中，使用了“按钮 3”对象和“按钮 4”对象。组态一个“画面窗口”（智能对象），通过两个按钮处的“直接连接”可将需切换的画面插入到窗口中。

7) 对于对象按钮 3 与按钮 4，选择“画面切换用户级”（位于“属性”“其他”“用户级”上），并将按钮属性对话框中的“属性”“其他”“操作员控制允许”设置为“否”。

8) 选择按钮 3 对象，从“动态向导”对话框中选择“标准向导”标签，然后选择“经授权方可操作”条目。单击完成按钮，即可完成动态向导。对按钮 4 重复同样过程。

9) 在变量管理器中，创建“文本变量 16 位字符集”类型的 @CurrentUser 系统变量。将

当前注册的用户名自动分配给该变量。

10) 触发按钮 3 与按钮 4 处的 C 动作，该 C 动作是在一旦改变该变量时由动态向导所产生的。这意味着 C 动作将不再每 2s 执行一次，而是只有在用户名发生变化后才执行。

由动态向导生成的 C 动作程序代码如下：

```
#include " apdefap. h "
Boot_ main(char* lpszpictureName, char*lpszobjectName,char* lpszPropertyName)
{
    #pragma code ( " UseAdmin,Dll " )
    #include " Pwrt_opi.h "
    #pragma code ()
    #define ND_MESSAGEBOX    1
    CMN_ERROR err ;
    DWORD    pvlevel = 0 ;
    pvlevel  = (DWORD) GetPasswordlevel (lpszPictureName,lpszobjectName) ;
    if (pvlevel == 0 ;
    return (TRUE) ;
    else
    return (PwrtcheckPermissionOnPicture(pvlevel ,lpszPictureName,M0_MESSAGEBOX));
}
```

五、实验内容：

1. 利用动态向导打开画面。要求：写出方法和步骤。
2. 退出运行系统或整个系统。要求：现在 Wincc 中实现后，再写出方法和步骤。

一、实验目的：

熟悉和掌握过程值归档创建方法和步骤，并完成规定的归档创建。

二、创建过程值归档

创建过程值归档

1. 首先，在变量管理器中创建需要进行归档的变量。例如创建 ProcessValue_1、ProcessValue_2、ProcessValue_3 三个变量，它们可以由 WinCC 变量模拟器来提供数值。

2. 打开“变量记录”编辑器。在 WinCC 资源管理器中通过鼠标右键打开变量记录条目，然后从弹出式菜单中选择“打开”来完成，如图 4-1 所示。

3. 创建一个新归档。通过鼠标右键打开归档条目，从弹出式菜单中选择“归档向导”来启动向导，该向导将指导用户创建一个新归档，如图 4-2 所示。



图 5-1 打开变量记录编辑器



图 4-2 启动归档向导

4. 通过单击“下一步”按钮退出起始页。在下一页中，将归档类型设置为过程值归档选项。输入归档名称，在本例中，归档名称为“ ProcessValueArchive ”。通过单击“下一步”继续到下一页，如图 4-3 所示。

5. 在向导的第三页中，定义要进行归档的变量，并通过选择按钮来完成。在本例中，使用 ProcessValue_1、ProcessValue_2、ProcessValue_3 三个变量。通过单击“完成”按钮关闭此向导页，如图 4-4 所示。

6. 在常规信息标签中，可设置基本的归档参数。归档参数与归档类型已在归档向导中指定，且归档类型不能再更改。系统启动时归档是激活的。在系统启动后将直接启动归档。不需要通过一个单独的功能来激活归档。在授权等级域中，将读访问和写访问设置为无访问保护。该数据可被任何用户使用，而不需要进行特殊的访问保护。在启动归档时，不执行特殊动作，这类动作可用来获取有关归档状态的信息，如图 4-5 所示。



图 4-3 创建归档步骤 1



图 4-4 创建归档步骤 2

7. 在归档参数标签中, 还可设置其他的归档属性。例如可将归档的条目数设置为 1000 条数据记录, 将存储位置选择在硬盘上, 将归档模式选择短期归档。也可为用于导出短期归档的动作设置一个函数, 如果短期归档已满, 将自动执行该函数。本例没有指定任何动作。

使用这些设置, 将有 1000 条数据记录被归档到硬盘上。如果超出了数据记录的最大数, 则最前面的归档条目将被删除并由新的条目取代。单击“确定”按钮关闭归档属性对话框, 如图 4-6 所示。



图 4-5 过程值归档属性的常规信息设置

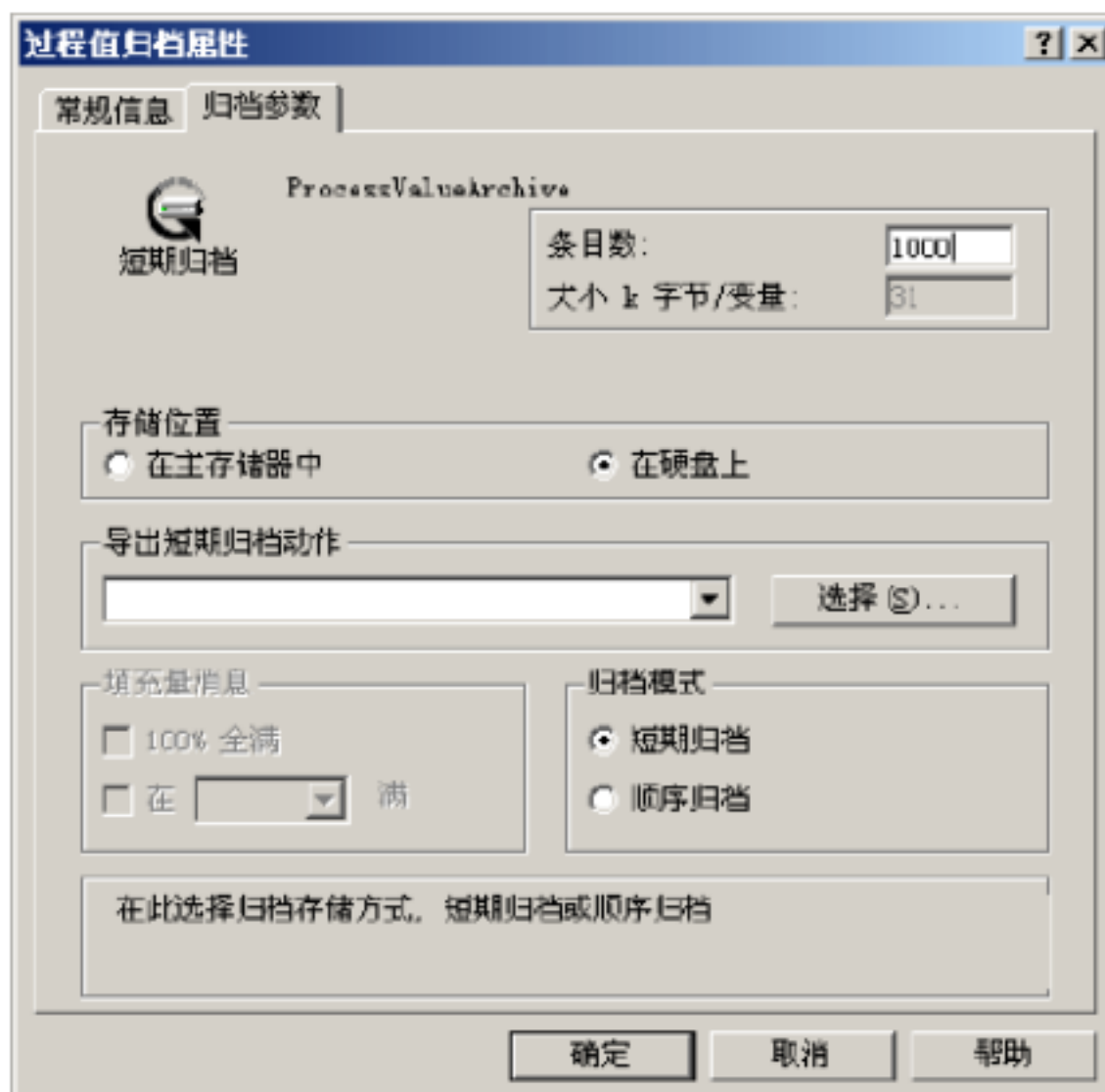


图4-6 过程值归档属性的归档参数设置

8. 指定各归档变量的属性。用鼠标右键打开底部的表格窗口，从弹出式菜单中选择“属性”以打开“归档变量属性”对话框，如图 4-7 所示。

...	变量名称	变量类型	注释
▶	ProcessValue_1	模拟量	
	ProcessValue_2	模拟量	删除(D)
	ProcessValue_3	模拟量	属性(P)

图4-7 选择“属性”

9. 在归档变量标签中, 可对基本变量属性进行设置。相应的过程变量已在归档变量中指定, 可为其分配一个名称以作为归档变量的名称。

在提供变量域中, 选择系统选项钮。在系统启动时自动开始归档域中, 选择允许选项钮。在采集类型域中, 设置周期—连续。在周期域中, 输入 1s 作为采集周期, 输入 1*1s 作为归档周期。这些设置表示数据采集在系统启动时开始, 并在恒定的时间间隔内连续进行直到系统关机, 如图 4-8 所示。

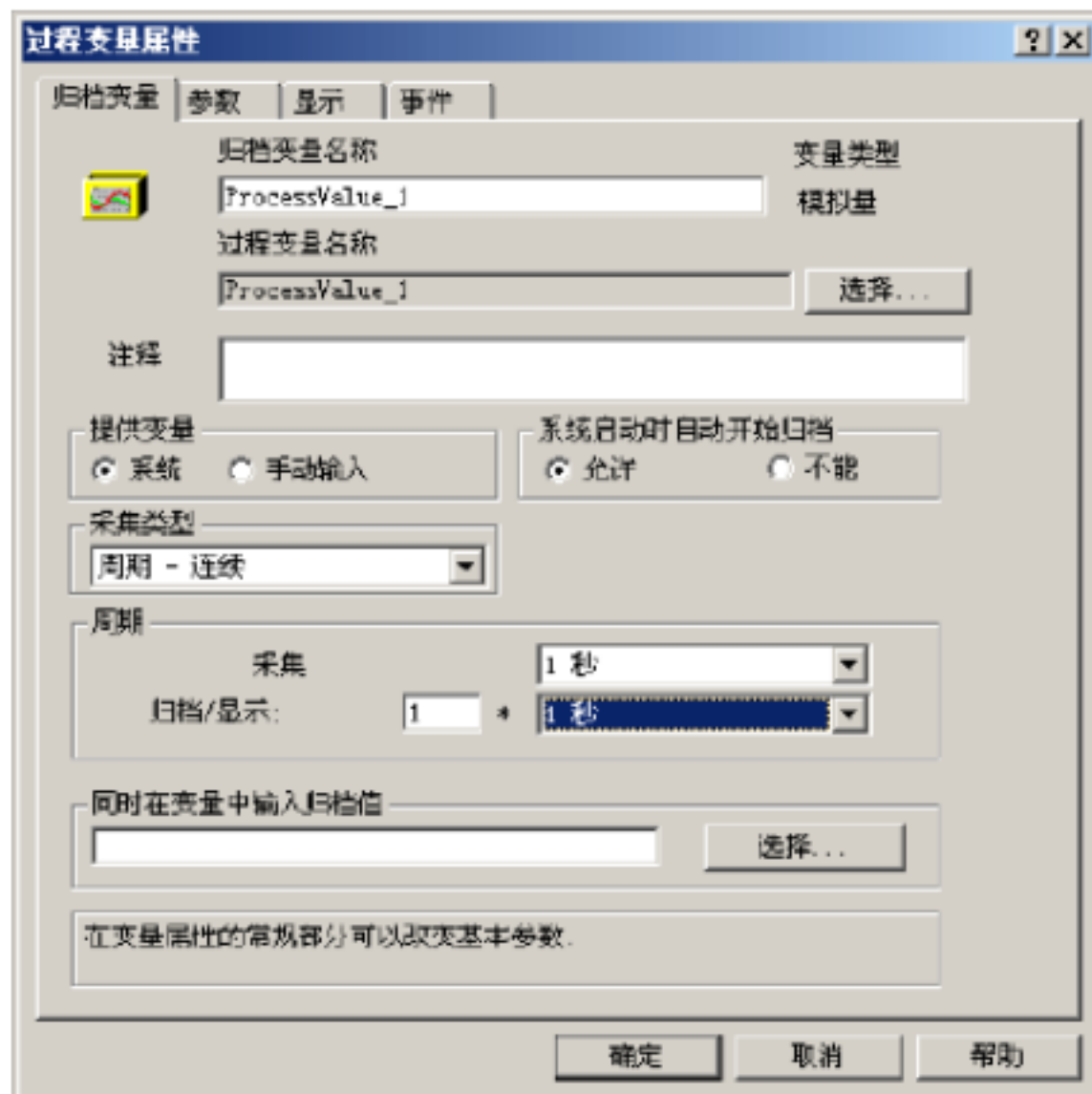


图 4-8 过程变量属性的归档变量设置

10. 在参数标签中进行其他设置。在处理域中, 选择真实值选项钮, 没有指定单元。在出错的情况下, 将保存最近的值, 如图 4-9 所示。

11. 在显示标签中, 指定变量进入归档的可接受范围。在本例中, 选择没有显示限制选项钮, 如图 4-10 所示。

12. 在事件标签内, 本例没有在动态域中输入改变归档周期的动作。单击“确定”按钮关闭过程变量属性对话框, 如图 4-11 所示。

13. 另外, 还必须指定另两个归档变量 (ProcessValue_2 ProcessValue_3) 的属性。因此, 还必须重新执行步骤 8 ~ 步骤 12。

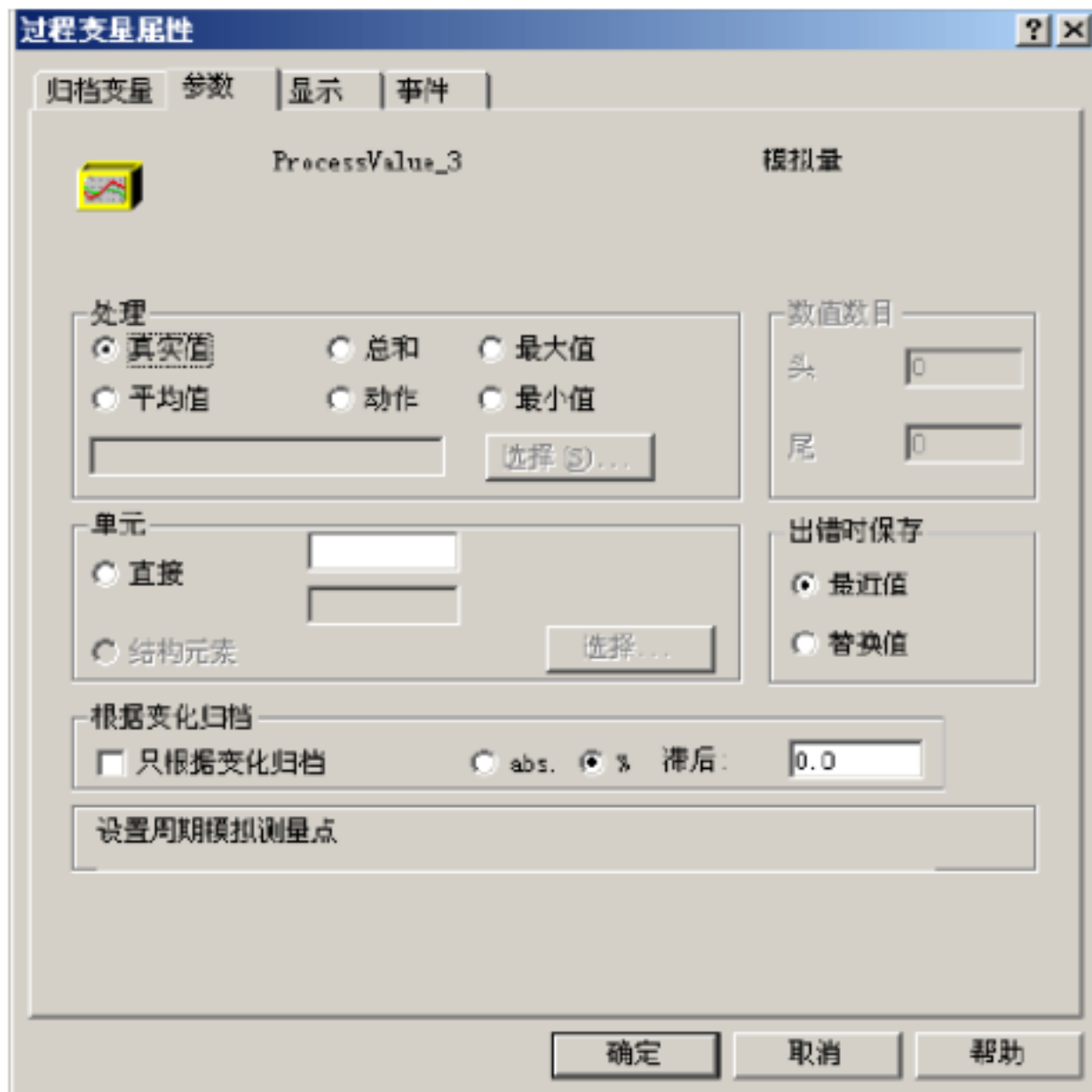


图4-9 过程变量属性的参数设置



图4-10 过程变量属性的显示设置

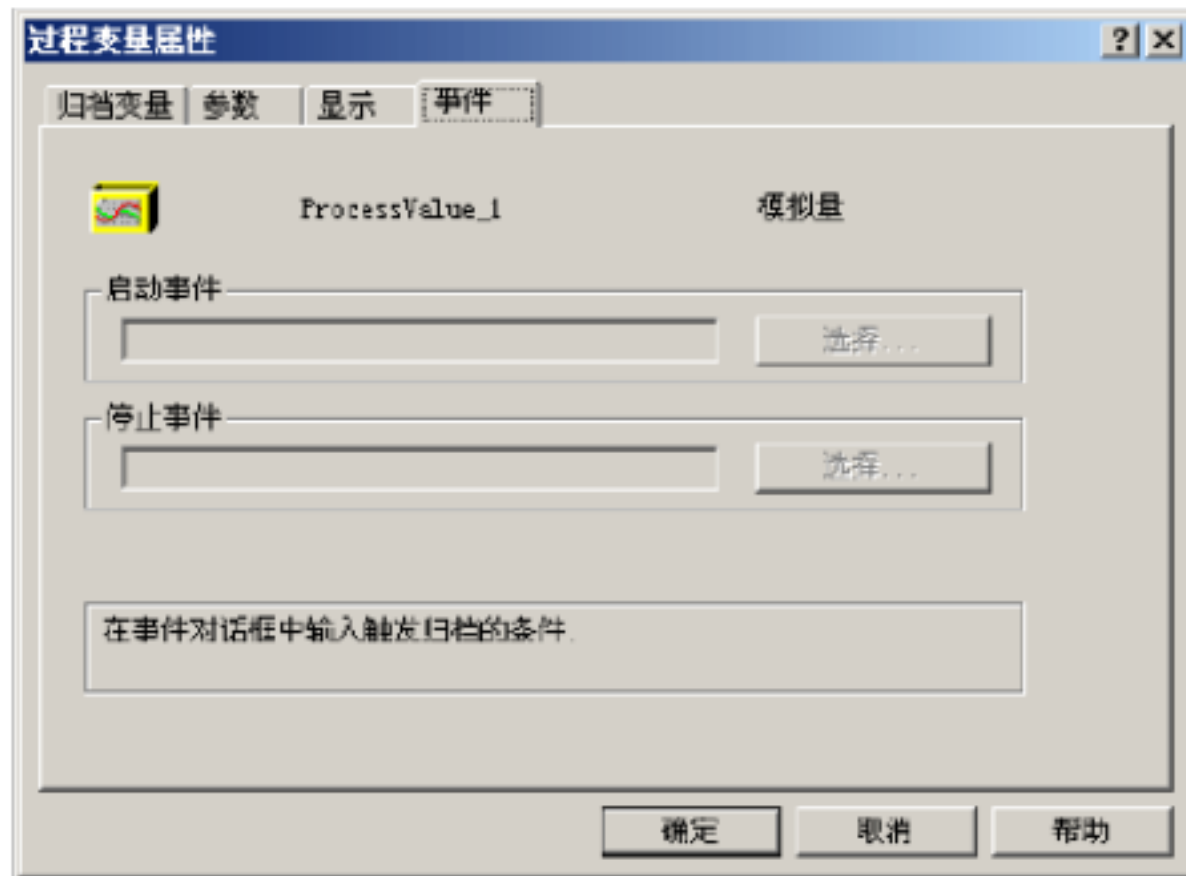


图4-11 过程变量属性的事件设置

三、组态趋势显示

1. 在图形编辑器中创建一个新画面。

2. 组态用于显示趋势图的线趋势控件。从“对象选项板”的控制选择菜单中选择该控件，然后将其置于画面中，如图 4-12 所示。

3. 将控件置于画面之中，将会自动打开其组态对话框。在常规信息标签中，可以指定控件的窗口标题以及它如何进行标记。在本例中选择显示复选框，并输入先前创建的归档名“ProcessValueArchive”作为窗口标题。

在打开画面域中，选择装载归档数据复选框，如果没有选择该复选框，则在画面打开后该控件将只显示已归档的值。

在数据源域中，可选择显示归档变量或在线变量。如果选择在线变量，则也可以显示没有进行归档的变量的趋势图。在本例中，设置为归档变量。

通过颜色按钮，可指定窗口的背景色。

在显示域中，本例规定显示状态栏和工具栏，为趋势的写方向选择从右写入。此外，还使用共享 X 轴和共享 Y 轴，并且窗口大小不可改变，如图 4-13 所示。



图4-12 在对象选项板中选择线趋势控件

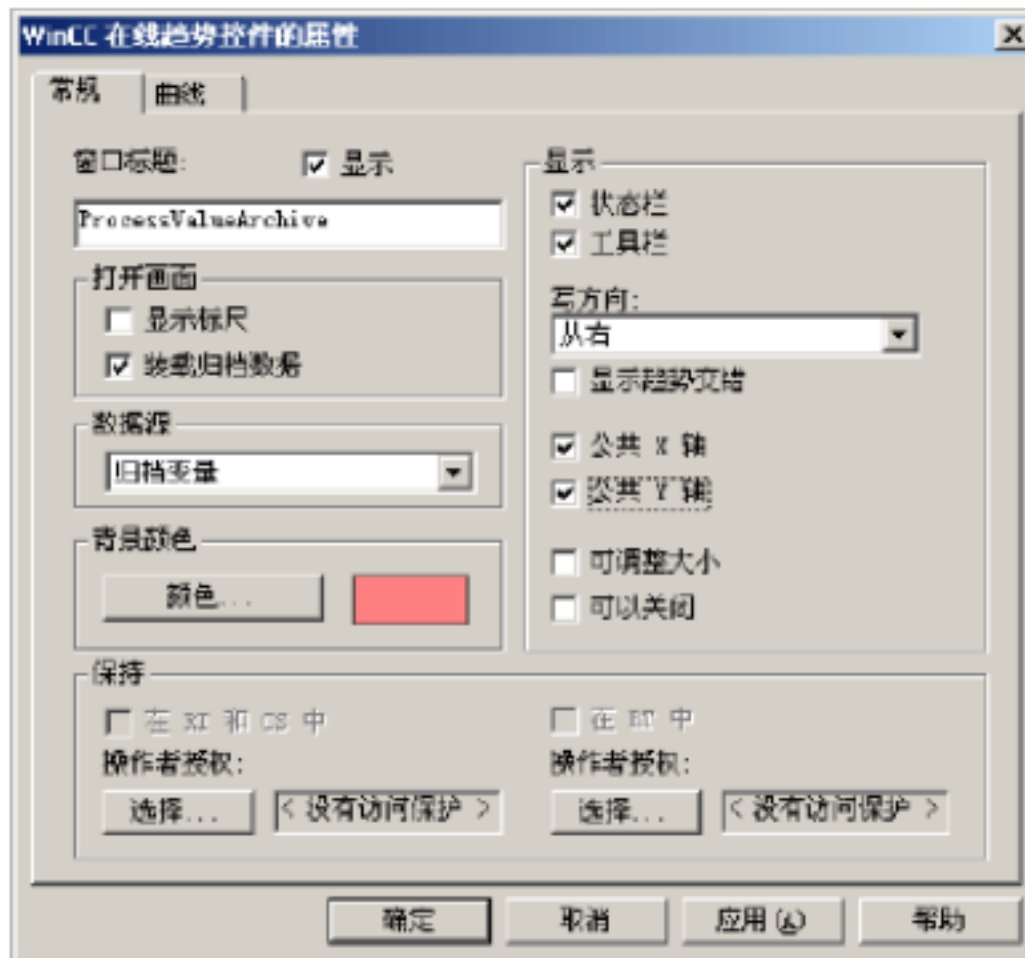


图4-13 线趋势控件的常规属性设置

4. 在“曲线”标签中，可详细地规定要显示的趋势图。打开曲线标签，已经创建了一条趋势。在本例中，将该趋势命名为“趋势 1”。通过“颜色”按钮，可以选择该趋势线显示的颜色。在显示类型中可将趋势线的显示类型设置为连接点。通过“选择”按钮，可以把要显示的归档变量分配给该趋势，如图 4-14 所示。



图4-14 线趋势控件的曲线属性设置

5. 打开“选择归档/变量”对话框。在对话框左边的窗口中，选择所期望的 ProcessValueArchive 归档。在右边的窗口中，选择所期望在该归档中可用的归档变量“ProcessValue_1”，如图 5-15 所示。

6. 在“曲线”标签中创建两条趋势来显示其余归档变量。打开“曲线”标签，单击“+”按钮在趋势标签中添加一条新趋势。通过相同的方法可创建另外两条趋势，并按步骤 4~5 中所描述的过程来设置其属性。

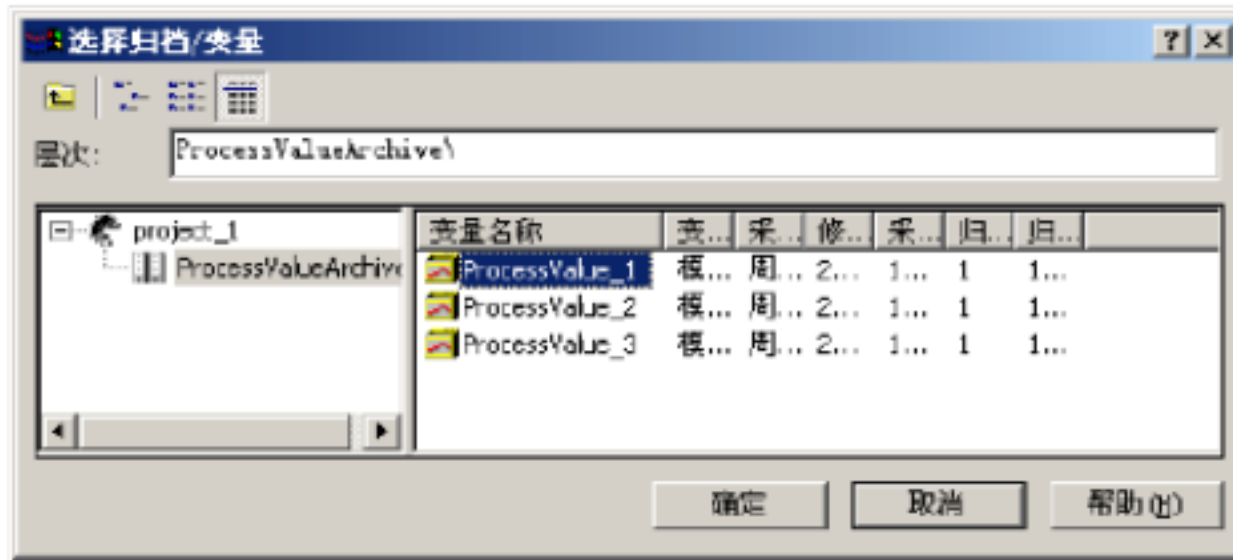


图 4-15 选择归档 /变量

两条新趋势使用的归档变量分别是 ProcessValue_2和 ProcessValue_3,且命名为“趋势 2”和“趋势 3”。单击“确定”按钮，关闭控件的属性对话框，如图 4-16 所示。



图4-16 创建另外两条线趋势

7. 趋势控件属性的其他设置。用鼠标左键双击“图形编辑器”中的趋势控件界面，将会出现更加详细的趋势控件属性设置对话框，除上面用到的“常规”、“曲线”标签外，还有“字体”、“工具栏”、“时间轴”、“数值轴”和“限制值”等标签，通过这些标签中的对话框可对趋势控件进行更加详细的设置。

例如在“数值轴”标签的标签对话框中，可将上面建立的 3 条趋势进行不同的命名，如将趋势 1 命名为“压力 1(Pa)”，趋势 2 命名为“压力 2(Pa)”，趋势 3 命名为“频率 (Hz)”，如图 4-17 所示。

8. 激活变量记录运行系统。在 WinCC 资源管理器中用鼠标右键单击“计算机”条目，然后从弹出式菜单中选择属性来打开“计算机列表属性”对话框。单击“属性”按钮来打开“本地计算机的属性”对话框。在“启动”标签中，选择要激活的运行系统应用程序。注意

必须选择变量记录运行系统复选框。单击“确定”关闭“计算机属性”和“计算机列表属性”对话框，如图 4-18 所示。



图 4-17 趋势控件属性的其他设置

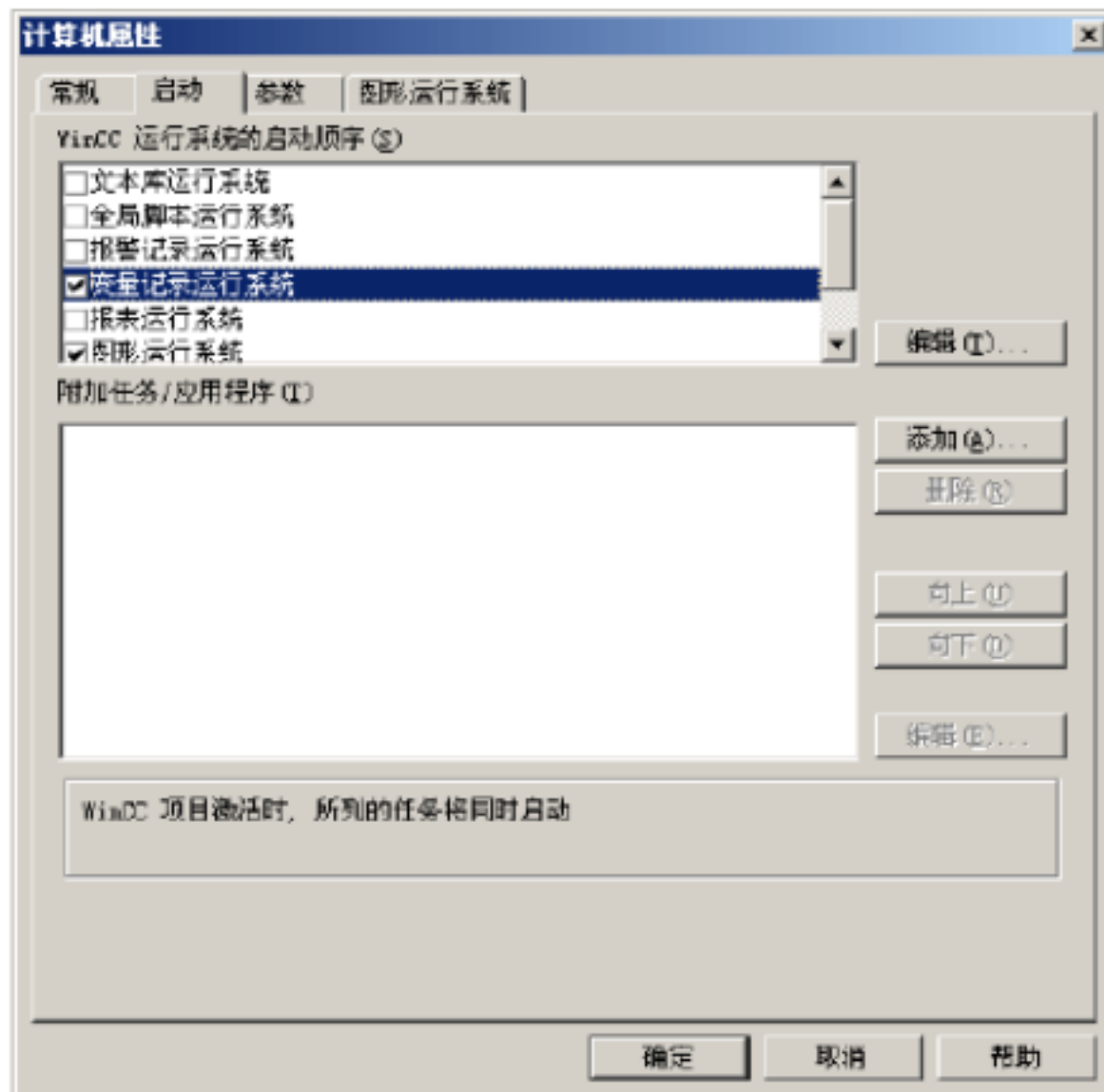


图 4-18 在计算机启动属性中激活变量记录运行系统

9. 连接具体的过程值变量，即可进行各个过程值变量的线趋势显示，如图 4-19 所示。

4-19 所

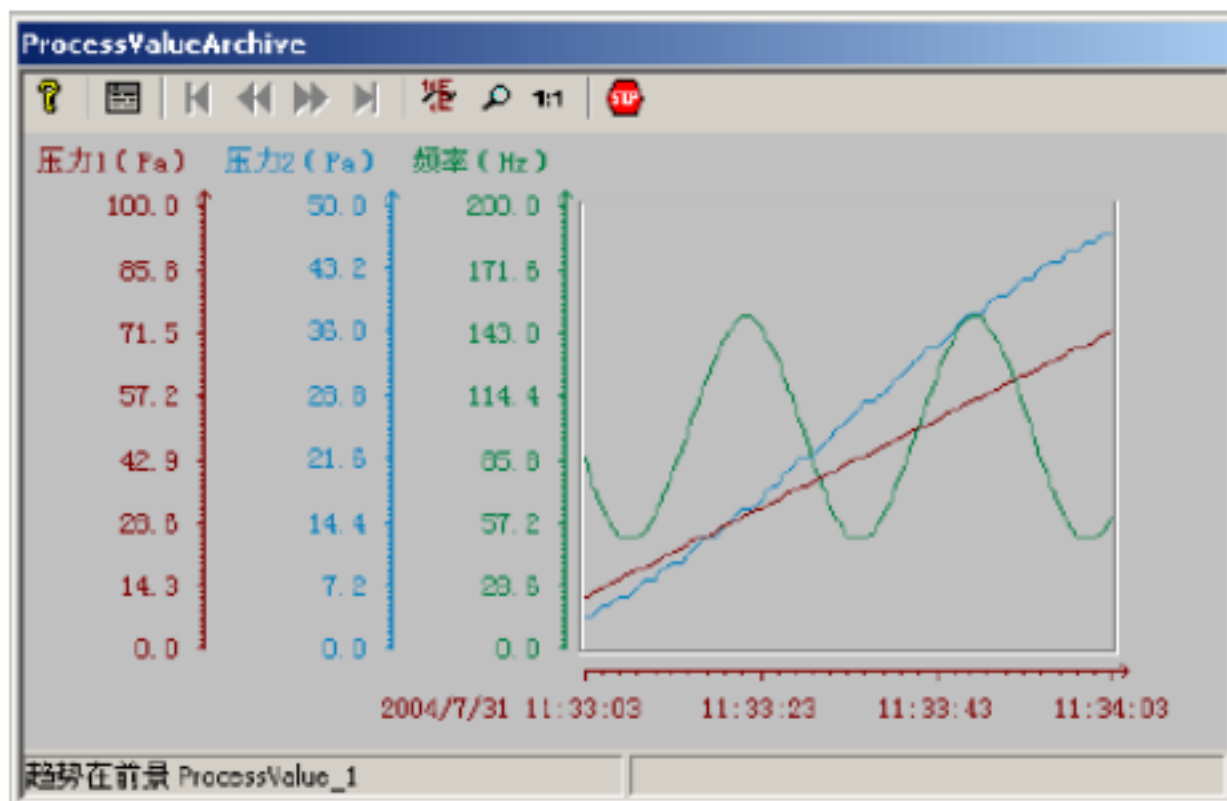
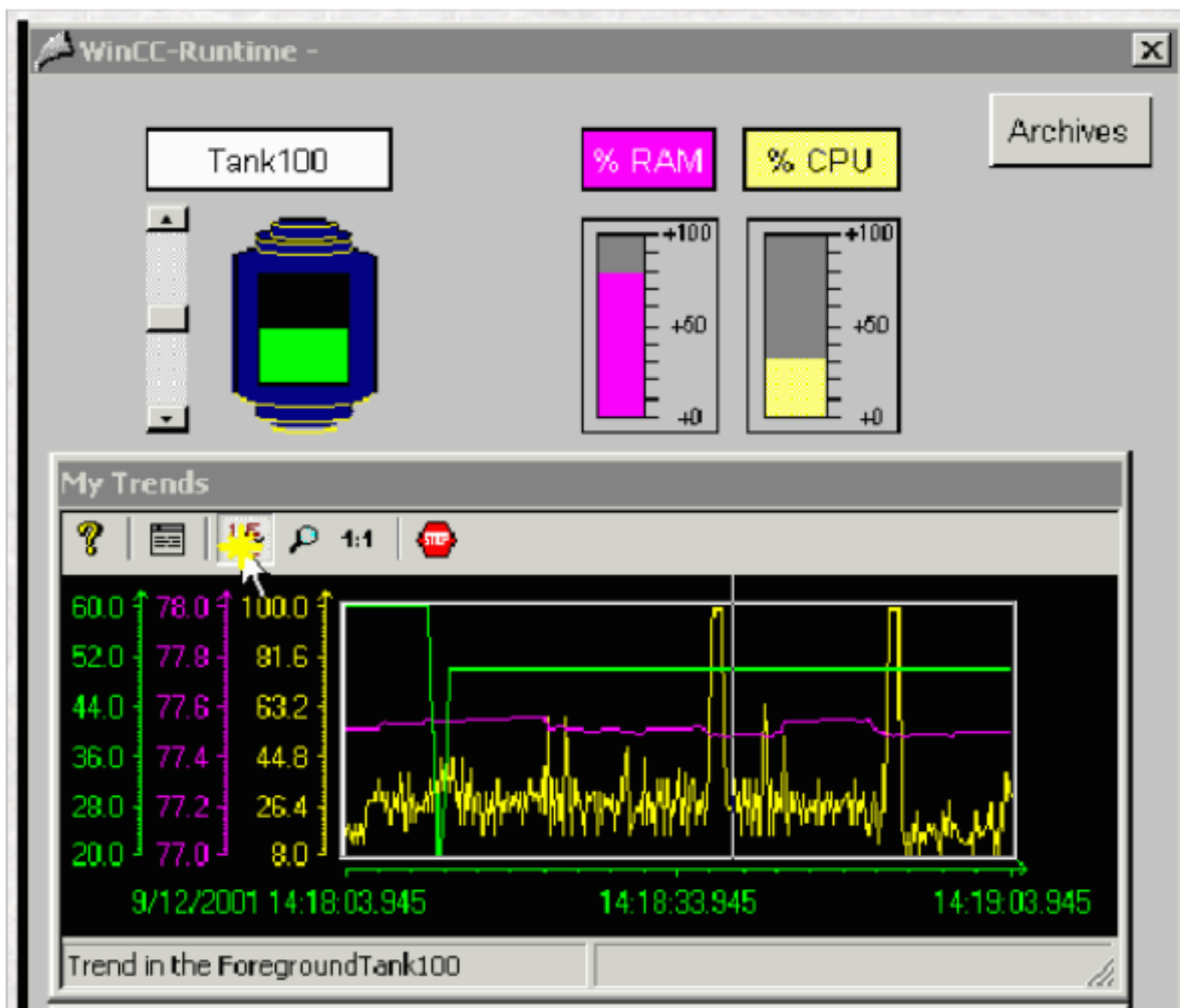


图4-19 激活后的线趋势显示示例

四、 实验内容及要求

在图形编辑器中完成如下图所示图形的创建；然后按步骤完成趋势图的创建；并运行得到如下所示图形。写出创建过程。



实验五 走马灯的实现

一、实验目的：

- 1、熟悉和掌握 Step7 软件的使用。
- 2、掌握 WinCC的使用。
- 3、掌握循环彩灯系统的 PLC控制及 WinCC监控界面的实现。

二、走马灯的实现要求

要求：运用循环移位指令实现 8个彩灯的循环左移和右移。其中 I0.0 位启停开关， MD20 为设定的初始值， MW12 为移位位数，输出 Q0.0 ~ Q0.7。

三、PLC 程序及原理说明

首先建立循环彩灯变量表如图 5 - 1所示。

PLC 地址	WinCC 地址	符号表	数据类型	变量名
I0.0	M0.0	Start	BOOL	启动开关
I0.1	M0.1	L_rotate	BOOL	左转
I0.2	M0.2	R_rotate	BOOL	右转
MW12	MW12	Interval	WORD	移位位数
MW14	MW14	I_value	DWORD	启停开关

表 5 - 1循环彩灯符号及变量表

在 Step7 中建立循环彩灯符号表如图 5 - 2所示。

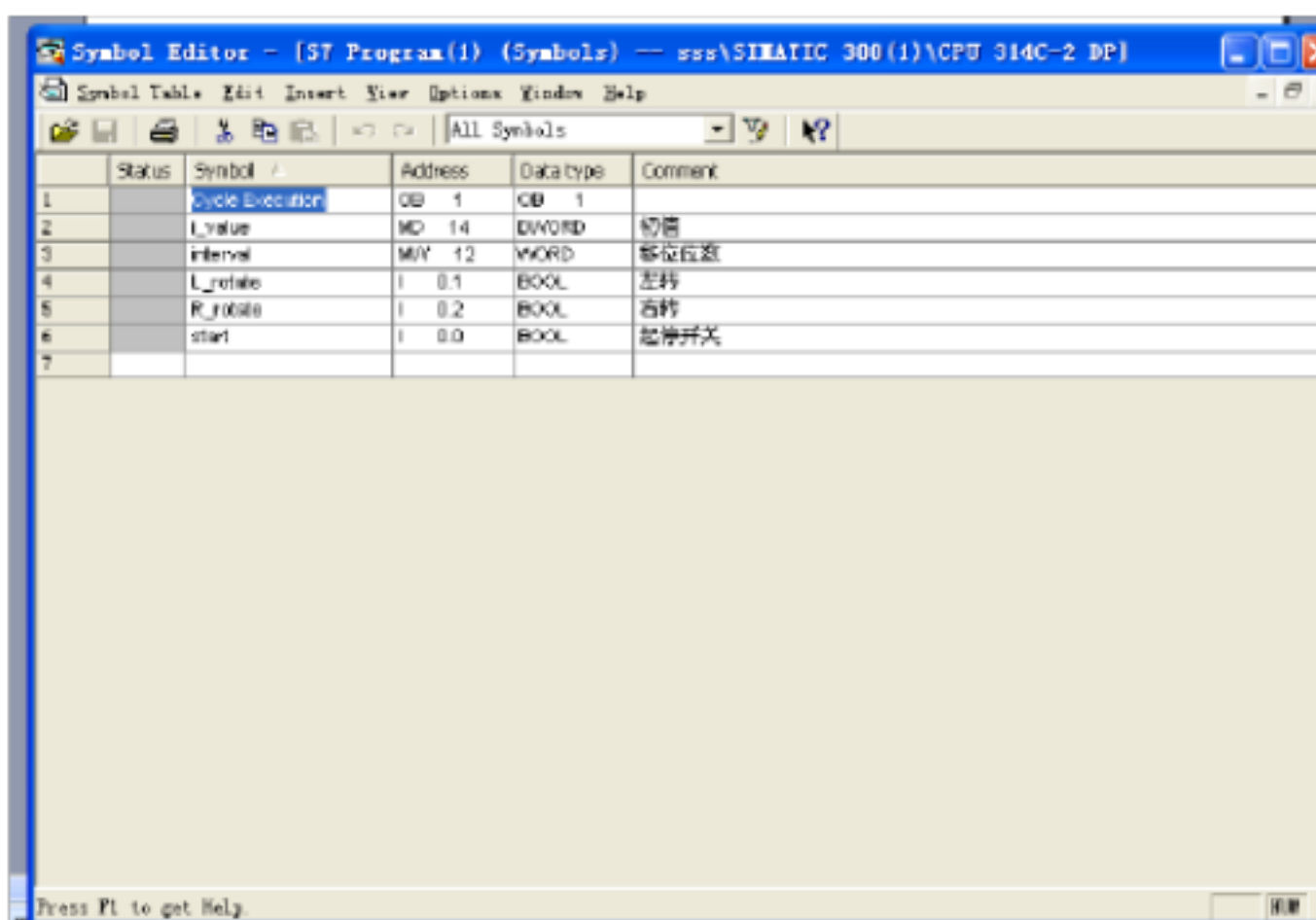
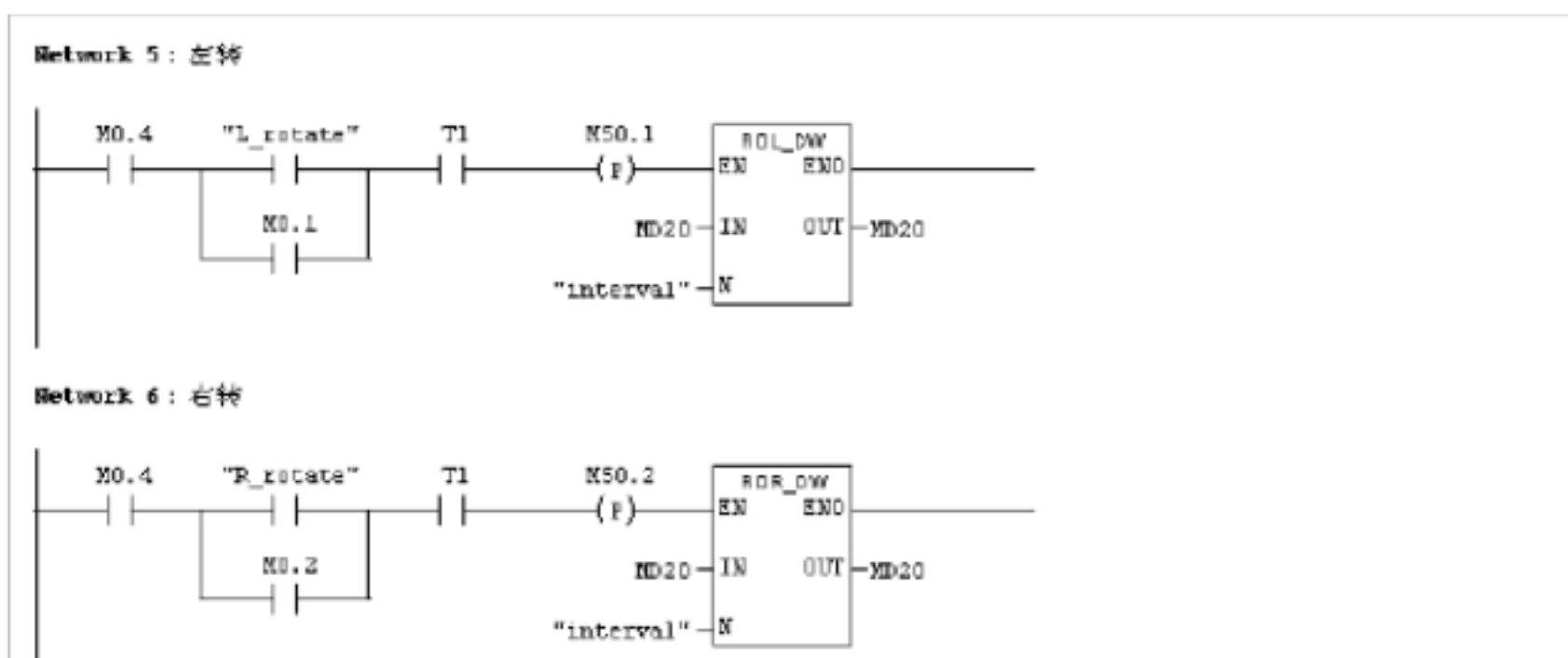
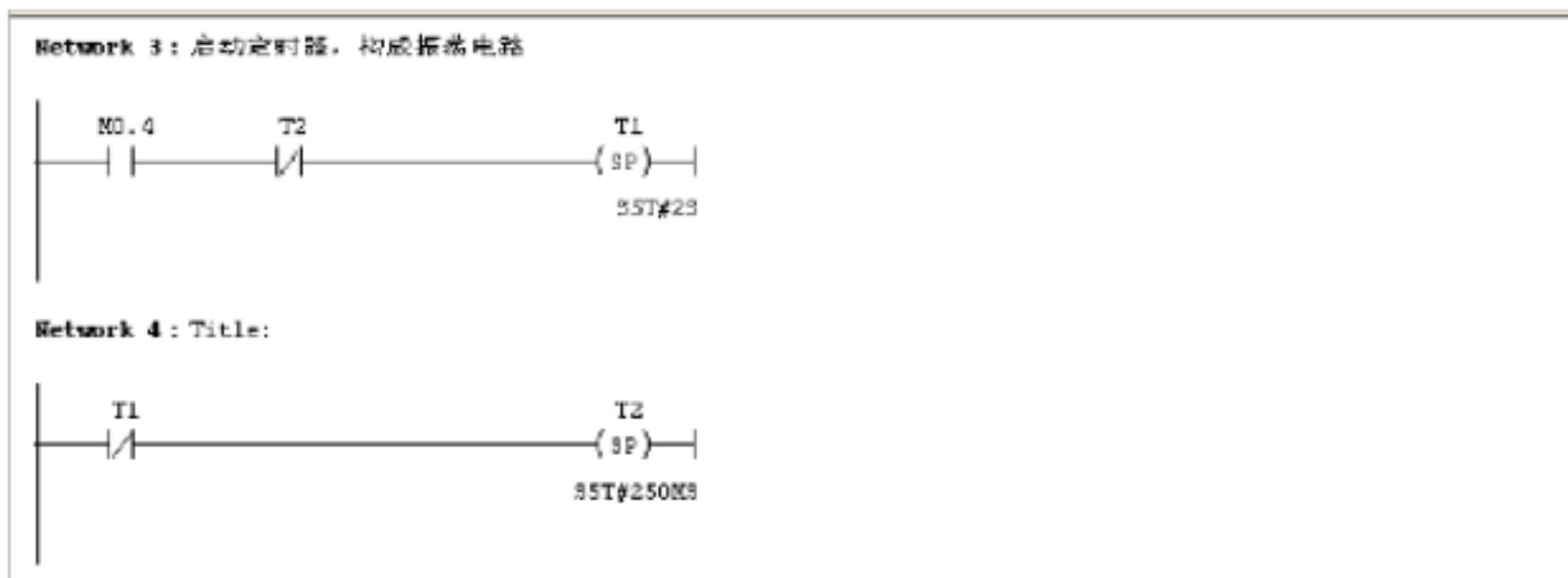
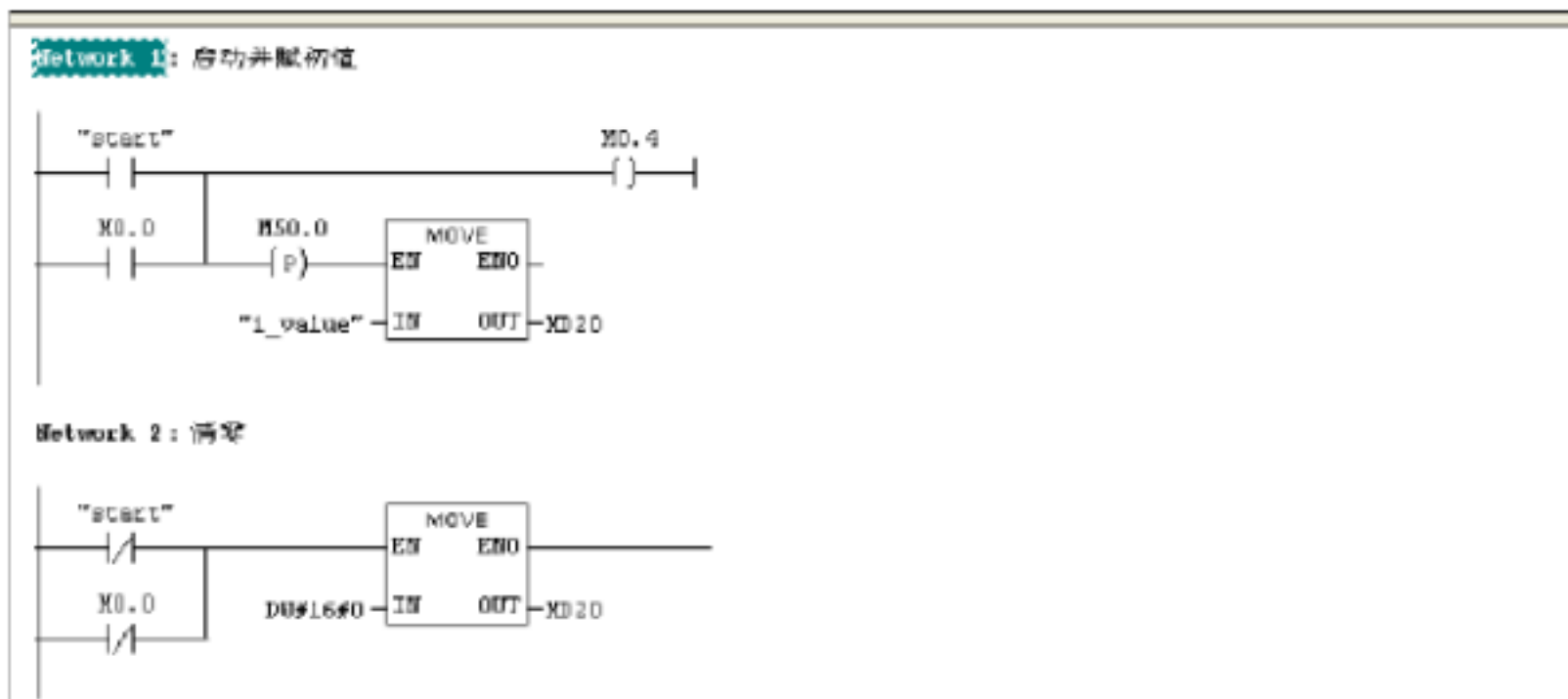


表 5 - 2 循环彩灯符号表

分析：首先建立定时振荡电路，振荡周期为 2S，使得每次定时时间到后，循环移位指令开始移位。在循环移位指令中应用了边缘触发指令，使循环移位在每个定时时间内只移位一次。在程序开始时，必须给循环存储器 MD20 赋初值，比如开始时，只有最低位的彩灯亮（为 1），则初值设定必须为 DW#16#01010101。

参考程序如图 5 - 3 所示。



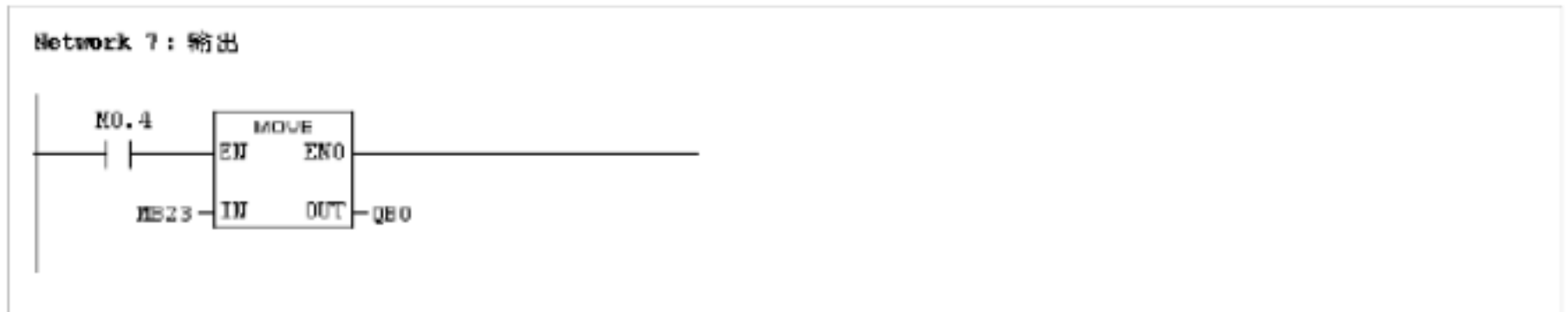


图5 - 3循环彩灯的梯形图程序

五、 WinCC 监控画面

WinCC 监控画面如图 5 - 4所示。

六、 实验要求

- 1、在 Step7 中编制程序并下载到 PLC 中运行程序。
- 2、在 WinCC 图形编辑器中绘制画面，建立变量连接，并在 WinCC 中运行画面。
- 3、写出实验步骤