

1. 概述 .....	1-1
1.1 可视化编辑器概述.....	1-1
1.2 创建一个新的可视化.....	1-2
2. CoDeSys中的可视化编辑器 .....	2-1
2.1 可视化元件的调用.....	2-1
2.2 布置视图 .....	2-3
2.3 配置一个可视化概述.....	2-6
2.4 配置可视化元素 .....	2-6
2.5 可视化元件，配置.....	2-50
3. 语言转换 .....	3-1
3.1 语言设置 .....	3-1
3.2 静态语言转换 .....	3-2
3.3 动态语言转换 .....	3-3
3.4 对动态语言转换的配置.....	3-3
3.5 用于动态文本的XML文件.....	3-4
4. 在线模式的可视化 .....	4-1
4.1 在线模式的可视化.....	4-1
4.2 在线模式下的键盘操作.....	4-1
4.3 在线模式下的‘文件’ ‘打印’ .....	4-1
5. 库文件的可视化 .....	5-1
5.1 库文件的可视化 .....	5-1
6. 可视化中的系统变量 .....	6-1
6.1 隐含变量 .....	6-1
7. 可以用于特定的可视化进程的组键.....	7-1
7.1 可以用于特定的可视化界面的组键.....	7-1
索引: .....	1



# 1. 概述

## 1.1 可视化编辑器概述

可视化是一种图形化表达法，它允许工程项目变量通过鼠标和键盘的方式输入在线运行的 PLC 程序中。CoDeSys 的可视化编辑器，是 CoDeSys 编程系统的一部分，它提供了一种图形化的元素，这种元素可以和工程项目中的变量链接并能按照需要进行安排。因此在运行模式下，图形化元素会根据变量值的改变而变化。简单的例子：为表现液位值在 PLC 程序中计算出的结果，可以绘制条状图并将其链接到相应的变量上，这样条状图形的长度和颜色将会显示出液位的当前值。增加文本区域，就能以字符串形式显示当前值，利用按钮可以控制程序的开始和停止。

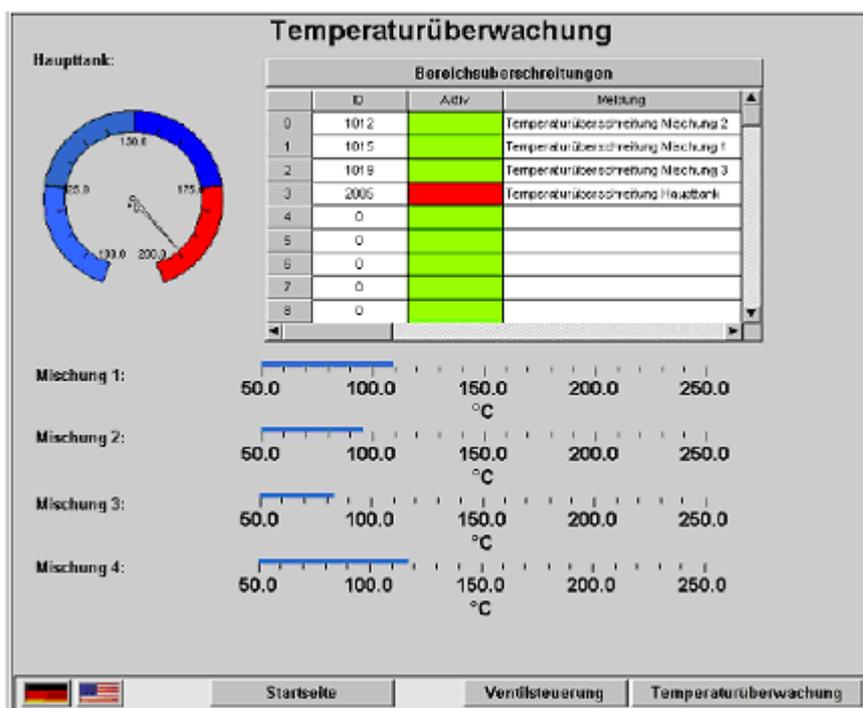
单个可视化元素和整个的可视化对象的属性都可以在组态对话框和属性对话框中进行定义，其中还能通过激活选项的方式设置基本的参数以及通过输入对象变量的方式定义动态的参数。

此外，可以在可编程的元素属性中通过结构变量来给出特殊的功能配置。如果要多次使用同一个可视化对象来显示程序运行中的不同结果，利用组态对话框中的占位符 (placeholders) 功能可以实现这样的功能。在程序联机运行的状态下，编程环境中创建的可视化界面可以作为唯一可用的用户接口，来控制及观察相关的 PLC 程序的运行，基于这种目的，程序中的变量只能通过激活可视化界面中的元素来输入。要实现上述功能，可以在可视化属性配置中选择所需的输入方法，并且对于每个特殊的可视化变量可以选择特殊的热键进行定义。

CoDeSys 中创建的可视化界面可以在下述情况下使用：

- 1、 CoDeSys HMI, 这是在配置了 PLC 的电脑中操作可视化界面的一种全屏显示的运行系统。
- 2、 Web-可视化, 基于网页的可视化，可以通过互联网操作和调用（常用于远程维护）。
- 3、 目标可视化, 目标可视化，可以直接在 PLC 中运行。

可视化界面的实例：



## 1.2 创建一个新的可视化

可视化对象可以在对象管理器中的‘可视化界面’中进行管理，它包含可视化元件的管理并且对不同的对象可以根据个人需要进行管理。一个 CoDeSys 工程文件中可以包含一个或多个可视化对象，并且相互之间可以连接。

要想在对象管理器中创建一个可视化对象，必须先从程序界面左下角点击 进入“可视化界面”区域，在下拉菜单中按以下路径选择‘工程\对象\添加’命令，或者直接在“可视化界面”中点击鼠标右键选择‘添加对象’，都能创建一个新的可视化界面，并且可以在弹出的‘新建可视化界面’对话框中为创建的界面命名。当输入的名称中不含非法字符并且无重名时，即可按‘确认’按钮关闭对话框，此时一个新的界面窗口将被打开，在其中即可编辑新的可视化对象。在创建的可视化对象中可以通过选择下拉菜单上的‘工程’‘对象’‘属性’选项，可以打开‘属性’对话框，在那里可以设置对象的用法以及是否作为主界面。

在定义可视化界面对象的名称时，请注意以下几点：

1. 如果没有明确的可视化界面被配置，自动默认命名为‘PLC\_VISU’的可视化界面作为目标可视化，Web 可视化或 CoDeSys HMI 的启动可视化界面。
2. 在工程中可视化界面的名称不能与其它对象重名，因为当切换可视化界面时可能会出现错误。

**注意：** 如果要使用系统变量 CurrentVisu(字符串类型)来直接选择当前打开的可视化对象，必须始终使用大写字母来命名界面（如 PLC\_VISU）





## 2. CoDeSys中的可视化编辑器

### 2.1 可视化元件的调用

#### 插入可视化元件

可视化元件是一种图形化元素，用来构建可视化对象。所有可用的元素都可以在 CoDeSys 的菜单栏中找到，每种都有独立的属性配置文件。用户可以在新建的可视化对象中随意插入各种几何图形、位图、图元文件、按钮，以及已有的可视化组件。几何图形包括矩形、圆角矩形、椭圆/圆、多边形等。

可以在下拉菜单的'插入'选项中自由选择以下命令：矩形 、圆角矩形 、椭圆/圆 、多边形 ，多义线 ，曲线 ，饼图 ，位图 ，按钮 ，表格 ，ActiveX 元素 ，仪表 ，条状图 ，折线图 ，报警列表 ，趋势图 ，图元文件  等。其功能如上表所示。编辑视图的快捷工具按钮位于视图编辑区域顶端标题栏的工具栏中，如图所示，



在编辑视图时，将鼠标移到视图编辑工具按钮处，会看到相关工具的自动提示。选中命令后，如果将鼠标置于编辑窗口，鼠标指针将会出现相应的图标（如选择 ，将出现 ），在编辑窗口选择所绘制图形的开始点的区域进行点击，并且按住鼠标左键拖动直至图形达到所要的尺寸。

#### 插入长方形

命令图标： ，用此图标在视图中插入长方形。

#### 插入圆角长方形

命令图标： ，用此图标在视图中插入圆角长方形。

#### 插入椭圆形

命令图标： ，用此图标在视图中插入椭圆形。

#### 插入多边形

命令图标： ，用此图标在视图中插入多边形。

#### 插入多线段

命令图标： ，用此图标在视图中插入多线段。

#### 插入曲线

命令图标： ，用此图标在视图中插入贝塞尔曲线。

#### 插入饼图

命令图标:  , 用此命令在视图中插入饼图。

点击选中该命令后, 按下鼠标左键并拖动可以生成所需尺寸的椭圆区域, 区域中的位置包含一个显示半径的线段, 只要按住左键通过拖曳鼠标就可以改变所绘图形的位置。

以下操作可以定义饼图的起始角度: 在绘制好的椭圆区域上点击鼠标左键, 将会在椭圆区域内出现三个黑点, 圆心位置是一个带有白色十字的圆点, 圆周和半径的交汇处是一个矩形黑点, 指明构成角度的两边, 两条构成  $0^\circ$  的半径能够分别选中并绕中心拖动; 在图形边上还有一个矩形黑点, 指示出所绘制椭圆虚拟的外接矩形的一个角。如果想给定具体数值来定义起始角度, 可以通过双击鼠标或者点击右键从菜单中选择“配置对话框”弹出自定义窗口, 在‘角度’目录中输入想定义的变量名。通过拖动图形中心点的位置还能调整图形的尺寸和形状(鼠标左键置于图形中心后将出现两个斜交叉的箭头, 此时按住鼠标左键并拖动, 即可改变), 同样, 选中并移动图形边上的黑点也可以改变图形的尺寸和形状。如果想移动整个图形, 则应将鼠标放在椭圆区域, 当出现两个垂直交叉的箭头时, 按下左键并拖动鼠标即可移动图形, 抬起鼠标左键即完成图形的移动。

### 插入位图文件

命令图标:  , 用此命令在视图中插入位图。

点击选中该命令后, 按鼠标左键并拖动, 可以形成一个带对角线的矩形区域, 同时弹出对话框提示选择要填充的文件, 当选择了要打开的文件后, 图片将被填充在矩形区域内。通过双击鼠标或者点击右键从菜单中选择“配置”将会弹出自定义窗口, 对图片进行多项处理。

### 插入已创建的其他文件

命令图标:  , 用此命令可以将已创建的视图作为一个元素插入当前视图。

点击选中该命令后, 按鼠标左键并拖动, 形成一个带对角线的矩形区域, 同时弹出对话框提示选择打开已存在的视图文件, 选中后视图文件将插入已定义的区域。

### 插入按钮

命令图标:  , 用此命令在视图中插入按钮。

点击选中该命令后, 在界面所要放置的区域按下鼠标左键并拖动即可生成所需尺寸的按钮。如果定义的是自锁的按钮, 则按下按钮和弹起按钮将会以的不同显示表示出所定义变量的状态。如果是不自锁的按钮, 则只有按着按钮不放才能锁定变量。

### 插入图元文件文件

命令图标:  , 用此命令在视图中插入图元文件。

点击选中该命令后, 按鼠标左键并拖动, 形成一个带对角线的矩形区域, 同时弹出对话框提示选择图元文件(扩展名\*.wmf), 选择文件后将插入已定义的区域, 需要注意的是没有可连接的文件插入后和插入位图的效果是一样的。

### 插入表格

命令图标:  , 用此命令在视图中插入表格, 该命令常用于显示一组变量的当前值。

点击选中该命令后, 按鼠标左键并拖动可以形成想要的尺寸区域, 同时弹出对话框“配置表”, 在其中可以定义表格的安全性、提示信息, 以及行、列等属性。

### 插入 ActiveX 元素

命令图标:  , 用此命令在用户定义的视图中插入 ActiveX 控件。

ActiveX 可以使用在 win32 系统的 CoDeSys HMI 以及 CoDeSys 目标可视化功能中，当按下鼠标左键并拖动后，将形成一个选定大小的区域，在区域中将提示选择确定的 ActiveX 控件插入其中，可以分别通过“附加”和“配置”命令在弹出的“配置 ActiveX 控件”窗口中对属性进行定义。

### 插入仪表

命令图标：，用此命令在用户定义的视图中插入表控件。

该命令提供的仪表中带有能定义刻度的圆弧，以及能同变量关联的元素点，当按按下鼠标左键并拖动后，将形成一个选定大小的区域，在显示仪表前会先弹出一个属性对话框“配置仪表”，在其中可以定义表中要显示的变量参数并预览完成定义后的表格。

### 插入棒图

命令图标：，用此命令在用户定义的视图中插入棒图。

该命令能显示指定变量的值，并以柱（条）状图来表明其值的变化。当按按下鼠标左键并拖动后，将形成一个选定大小的区域，图形显示前会先弹出一个属性对话框‘配置棒图显示’，在其中可以定义表中要显示的变量参数，并在确认插入前预览完成定义后的图形。

### 插入柱状图

命令图标：，用此命令在视图中插入柱状图。它利用柱状图的长度表明变量的数值。

当按按下鼠标左键并拖动后，将形成一个选定大小的区域，图形显示前会先弹出一个属性对话框“配置柱状图”，在其中可以定义表中要显示的变量参数，并在确认插入前预览完成定义后的图形。

### 插入报警列表

命令图标：，用此命令在视图中插入报警列表。

使用该命令时，当按下鼠标左键并拖动后，将形成一个选定大小的区域，图形显示前会先弹出一个属性对话框“配置报警表”，其中有包括“报警列表”、“报警种类设置”、“列(栏)”等显示项目在内的多个选项，还有提示信息、安全级别设置等一般选项，利用这些选项可以对报警列表的显示内容、显示格式等进行定义。

**注意：**该处的报警列表仅仅对用于对报警的显示，具体的设置及定义在“工程”菜单的“配置报警”选项中。

### 插入趋势图

命令图标：，用此命令在视图中插入趋势图。

使用该命令时，当按下鼠标左键并拖动后，将形成一个选定大小的区域，带有“定义坐标轴”、“变量”、“历史记录”等项目的属性对话框“趋势”将会自动弹出。趋势图，又称示波器，通常用来观察变量在一个特定时间段的变化情况，它将数据存储为用户定义的文件中并以图形显示出来，当变量的值发生改变时，就会有新的数据输入存储的文件并在数据/时间轴上显示更新，通过设定用户需要的背景（图片，颜色等），可以使要观察的变量更加清晰、分明。

## 2.2 布置视图

### 选择可视对象

“选择方式”是鼠标默认的指令，用鼠标左键点击视图对象即可将其选中。

可以按下<Tab>键，选择视图对象中的第一个视图对象，再次按下<Tab>键则选择下一个视图对象。如果同时按下<Tab>和<Shift>键，可以按照视图对象中相反的顺序选择。在选择一个视图对象后按下<Shift>键，同时点击相应的视图对象，可以选择多个视图对象。也可以按住鼠标左键不放，在要选择的视图对象上拉一个窗口，同样可选择多个对象。

### 选择方式

如果“附加”菜单中的“选择方式”菜单前有“√”，或者快捷工具栏中表示鼠标状态的快捷工具按钮被按下时，表示此时处于“选择”状态，可以选择视图对象。否则处于绘图状态。

全选:使用“附加”菜单中的“全选”菜单可以将当前视图中的所有视图对象全部选中。

如果打开了元件列表(‘附加’元件列表’),你可以选择其中某一行以便选择在可视化界面中的相应元件.

### 改变选择和插入模式

插入视图元件后，鼠标将自动改回选择模式，要想继续进行下一个视图元件的插入，可以从下拉插入菜单中选取命令，或者点选工具条上的“”图标，退出选择模式。在按下“Ctrl”键不放时，通过点击鼠标右键可以在选择模式和插入模式间进行切换。

#### ‘附加’选择’

从下拉插入菜单中选取命令，或者点选工具条上的“”图标，退出选择模式。在按下“Ctrl”键不放时，通过点击鼠标右键可以在选择模式和插入模式间进行切换。

#### ‘附加’全选’

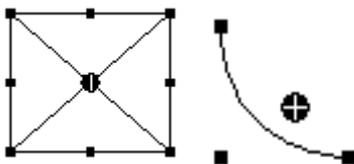
使用这个命令可以在当前的可视化状态条中全部选择可视化元件。

### 复制可视化元件

使用“编辑”/“复制”命令，或者<Ctrl>+<C>组合键，可以复制所选择的一个或多个视图对象。复制视图的另外一种方法是选择要复制的视图对象，按下<Ctrl>键的同时点击该视图对象，则会在原来的视图对象上产生复制的视图对象。

### 修改可视化元件

用鼠标点击对象或者按下<Tab>键，可以选择一个视图对象。在所选择视图对象的周围有一些小的黑色矩形。通过点击这些小的黑色矩形，按住鼠标左键，可以改变视图对象的大小，控制视图对象的轮廓。选择一个视图对象后，同时显示旋转点。旋转点是一个中间带有白色十字的黑色圆圈()，可以按一定的角度使对象绕这点旋转。可以按下鼠标左键拖动旋转点。对于多边形，使用同样的方法拖动每个顶角。鼠标拖动时，按下<Ctrl>键，可以插入一个顶角。按下<Shift>+<Ctrl>键，可以删除一个顶角。



### 拖动可视化元件

点击鼠标左键选择视图对象。在所选的视图对象上按下鼠标左键，或者按下方向键，可以拖动一个或多个视图对象。

### 组合可视化元件

选择多个视图对象，点击“附加”菜单中的“组合”菜单，可以将多个视图对象组合成一个视图对象。组合后视图对象的行为与一个视图对象的行为相同。

取消组合: 选择一个组合的视图对象，点击“附加”菜单中的“取消组合”菜单，可以将该组合的视图对象分解为多个单独的视图对象。

#### ‘附加’置于前面’

用这个命令可将可视化元件放到最前面显示。

#### ‘附加’置于后面’

用这个命令可将可视化元件放到最后面显示。

**'附加'对齐'**

使用这个命令可以排列已选择可视化元件。

以下的排列选择是可用的：

左：最左边对齐方式排列元件

右/上/下，与上面相同

水平中心：按元件水平中心对齐

垂直中心：按元件垂直中心对齐

**'附加'元件列表'**

点击标题栏“其它”菜单中的“对象列表”菜单，打开视图对象列表对话框，如下图所示，其中包括视图对象的号码、类型和位置等信息。点击该对话框右侧的工具按钮，可以对其进行编辑。在绘图区点击鼠标右键，选中“对象列表”，也会弹出此对象列表。

“确认”按钮：当对对象列表内容操作完毕后，点击“确认”按钮，关闭对话框。

“置于最前面”按钮：把选择的视图对象放在最上层，此时元件号最大。

“置于最后面”按钮：把选择的对象放在最底层，此时元件号最小。

“向前一个”按钮：把选择的视图对象向上移动一层，此时元件号增一。

“向后一个”按钮：把选择的视图对象向下移动一层，此时元件号减一。

“删除”按钮：删除选择的对象

“取消”按钮：取消上一次操作

“重做”按钮：恢复上一次操作。

“编辑”按钮：对视图对象进行编辑操作。

**可视化状态条**

如果一个可视化有中心，相对于图像的左上角，鼠标指针在状态条上显示当前的 X 和 Y 坐标。如果鼠标

锁定了一个元件，或者元件被处理，那么显示元件的编号。如果选择插入一个元件，然后这个元件也将出现（例如，长方形）。

## 2.3 配置一个可视化概述

对于可视化元件或是可视化对象的全部配置可以通过“附加”菜单的对话框来配置，可视化的附加设置可以在属性对话框中进行。

在工程选项中,有单独的目录用于保存可视化文件。

## 2.4 配置可视化元素

### 配置可视化元件

通过‘附加’配置’命令打开配置对话框。在该对话框中用激活选项或动态插入工程变量来设置一个元件或对象的属性。此外，可以通过定义各可视化元件结构变量的组件来编写属性。

有关在线模式中遵循的解释程序：

——通过工程变量动态给出的值重写同一属性设定的参数。

——如果一个元件属性通过“标准”工程变量以及通过一个结构变量的组件来定义时，那么在在线模式中，首先关注工程变量的值。

**请注意**，如果在CoDeSys HMI中，要将可视化用作目标系统可视化或web可视化，即将可视化用作PLC程序的用户接口，那么“占位符”和“专用输入”是很有用的。

**注意**：对于含有配置可视化元件颜色和字体的对话框，随着当时选定的目标系统的不同而有不同的外观，它可能不是标准的对话框，而是一个与项目相适应的选项对话框，它用于指定的目标可视化。

### 在可视化中的占位符

在配置对话框中，每个输入变量或文字的地方，都能放入一个占位符来替代相关的变量和文字。如果创建的一个可视化对象不直接用在程序中，而是作为一个“实例”插入到其它可视化对象中，那么占位符是有意义的。在配置这样的一个实例时，可以用变量名或文字来替代此占位符。（参看配置一个插入的可视化界面，在那部分帮助中有占位符使用的例子）。

任何用两个“\$”括起来的字符串都是一个有效的占位符（例\$Variable\$, Variable\$X\$）。在‘占位符列表’对话框（从‘附加’‘占位符列表’调用）中，对每个占位符，都可以把一个“值组”定义为一个输入明细。在配置一个可视化对象的实例时，你可以用这些值中的一个替代这个占位符。在实例中采用一个占位符列表来做这种替代。

占位符应用举例：

在相同可视化界面的实例帮助下，功能块实例很容易被显示。例如，在配置用于显示功能块变量的可视化界面中，每个变量以\$FUB\$开头（如\$FUB\$. a），如果使用可视化界面实例（在另一个可视化界面中插入一个可视化界面或使用‘转到’调用），那么在这个实例的配置中，占位符\$FUB\$可以用功能块的实例名替代，以便显示变量值。

如下所示：

在工程中定义了包含下列声明：

```
FUNCTION_BLOCK fu
```

```
VAR_INPUT
```

```
  changecol : BOOL; (* 在可视化界面中改变颜色 *)
```

```
END_VAR
```

在PLC\_PRG 定义了两个‘fu’实例：

```
inst1_fu : fu;
```

```
inst2_fu : fu;
```

创建可视化对象‘visu’，插入一个元件并打开配置对话框，打开‘变量’选项，在字段‘改变颜色’中输入

“\$FUB\$. changecol”. 打开‘输入’选项, 在字段‘触发但不保持变量值’中输入“\$FUB\$. changecol”. 打开‘文字’选项, 输入“\$FUB\$ - change color”

创建另一个可视化对象‘visu1’. 在‘visu1’中插入两个‘visu’ (两次参考‘visu’)

选择‘visu’的第一个参考, 打开配置对话框中的‘可视化’选项, 点击‘占位符’按钮后将显示占位符列表, 用‘PLC\_PRG.inst\_1’替换‘FUB’.

选择第二个‘visu’实例, 按上面的描述, 用‘PLC\_PRG.inst\_2’替换‘FUB’.

在联机模式下, 用于配置‘fu’实例的变量值将在相应的‘visu’实例中显示结果.

当然, 占位符\$FUB\$可以在‘visu’配置中的所有可输入变量或文本的位置使用.

### ‘附加’占位符列表

占位符列表如图所示.

占位符: 列出用于配置所有视图对象的占位符。

元件号: 显示包含占位符的对象号。

替换: 可以输入一些字符串。例如文本、变量或表达式等。

在 CoDeSys 中这个列表用于管理和配置占位符:

在配置以后用于可视化对象实例的可视化界面时, 使用此列表. 由于这个原因, 可以使用占位符替换配置对话框中的变量或字符串. 可以通过菜单‘附加’占位符列表’或上下文菜单打开占位符对话框, 在列表中有三栏.

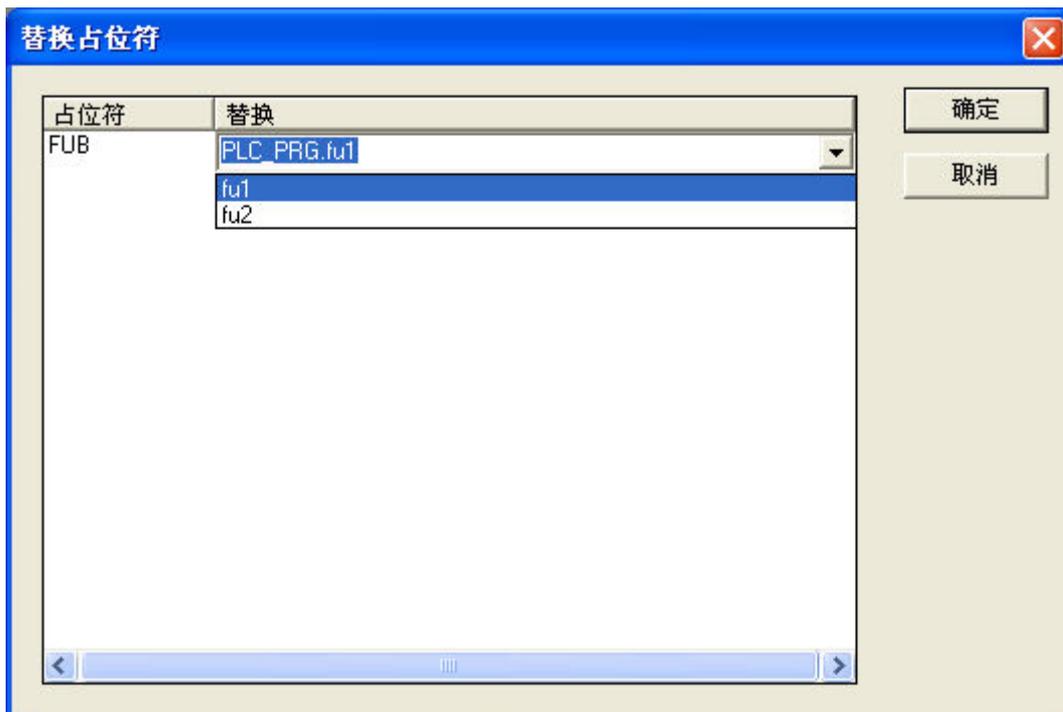
占位符列表例子



占位符列出了所有在当前可视化界面配置中使用的占位符. 元件号包含占位符的元件. 在替换栏中, 可以输入一个或多个字符串(文字, 变量, 表达式), 以便在配置可视化界面实例中替换占位符时选择它们. 输入的可选元件必须用逗号分开. 如果没有设置或设置了一个不可能的替换字符串, 在配置可视化界面实例时可以用期望的文字替换.

当配置上面提到的可视化界面实例时, 可以使用占位符列表, 即通过命令‘插入’可视化’在另一个可视化界面中插入这个对象后. 为了实现这个, 作如下操作, 打开对话框, 选择插入的可视化界面, 执行命令‘附加’配置’, 然后按‘可视化’选项中按钮‘占位符’, 在这种情况下, 对话框只包含两栏:

在可视化实例中用于替换占位符的占位符列表



如上图所示, 占位符栏中列出了所有在主可视化对象中定义的占位符. 如果可能的替换元件已定义, 可在替换栏中选择它们., 选择一个替换当前实例中的占位符. 如果没有预先定义, 手动输入一个表达式或变量. 也可以在替换栏的字段处双击打开编辑字段 .

#### ’附加’ ’配置’

用这个命令, 可以打开 ‘配置元件’ 对话框, 用来配置将选择的对话框, (参看 选择可视化元件). 当你在这个元件上双击时, 便弹出此对话框。

在对话框的左边区域内选择一个类目 (可用的类目取决于元素类型), 并在右边区域内填写所需的信息. 它通过激活选项插入定义元素属性值的有效变量名来完成。

**注意:** 还有供元件组使用的配置对话框. 该设置仅对元素组有效. 如果你想配置组内的某元件, 你必须分解这个组。

**注意:** 如果你已通过 “静态” 静态设定且通过一个动态变量来定义一个元件的属性, 那么该变量在在线模式中改写静态值。(例如可以在类别 “颜色” 中静态定义 “内部报警颜色”, 又在类别 “颜色变量” 中用一个变量给一动态定义)。如果用一个 “标准” 项目变量和一个结构变量, 同时进行设定, 那么 “标准” 项目变量将改写结构变量值。

**注意:** 对于仪表图、条形显示和柱形图, 必须事先重组。

在元件配置中操作变量的部位可以作下列输入:

- 变量名, 可采用输入助手来输入它。
- 表达式, 它是由分量存取, 有常数索引的字段存取, 变量和直接地址装配而成的。
- 操作子和常数, 需要时可与上述的表达式组合。
- 用变量名或正文串替换占位符。

允许的表达式举例:

$x + y$

100\*PLC\_PRG. a

TRUE

NOT PLC\_PRG. b

9\*sin(x + 100)+cos(y+100)

不能调用函数. 无效的表达式会在登陆时产生一个错误信息. 无效表达式的例子: fun(88), a:=9, RETURN.

在配置对话框中, 写全局变量有两种可能的方法: “.globvar” 和 “globvar” 是等同的. 然而, 有一个

点的表达式不允许用于装配表达式中。

**注意：**还可以使用占位符。

### 角度

角度 Angle 属性用来定义饼图对象的角度。双击饼图对象，弹出饼图对象属性配置对话框，如下图所示。点击角度 Angle 属性，在“开始角度”和“结束角度”文本框中分别输入饼图对象的开始角度和结束角度，则将会以顺时针方向画出所需要的饼图。选中“只显示弧段”，则饼图对象不显示夹角，只显示弧段。

举例：

声明变量

```
PROGRAM PLC_PRG
```

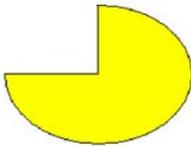
```
VAR
```

```
angle_start: REAL := 90;
```

```
angle_end: REAL := 180;
```

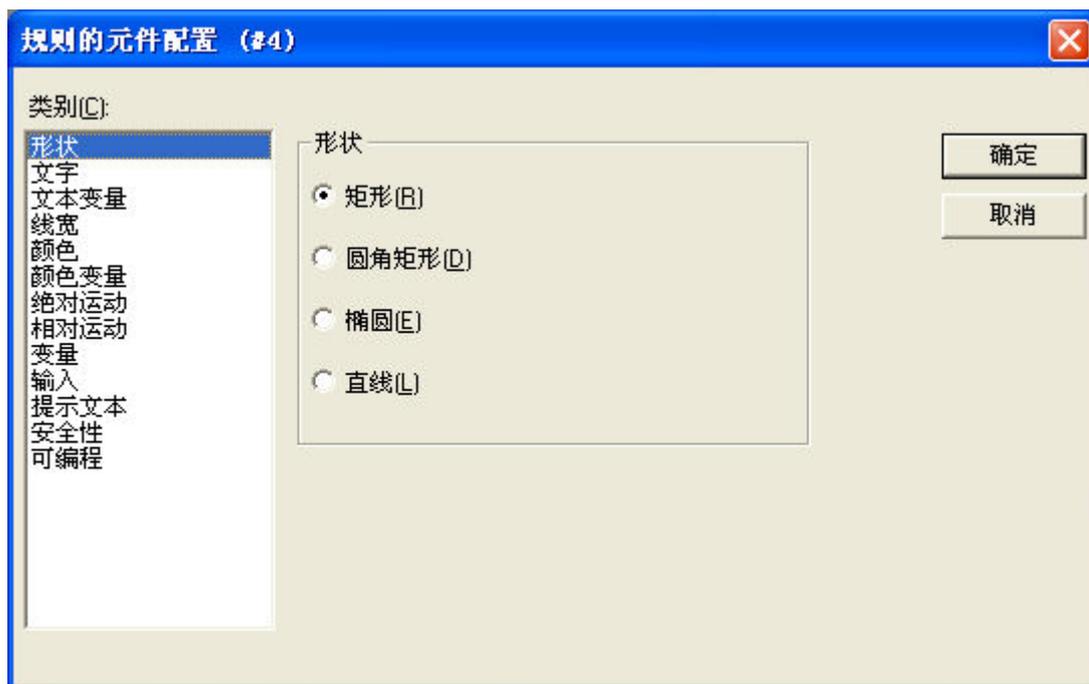
```
END_VAR
```

视图运行结果如图所示：



### 形状

形状 Shape 属性用来定义视图对象的形状。对于规则视图，形状 Shape 属性可以选择矩形、圆角矩形、椭圆或直线等。对于不规则视图，形状 Shape 属性可以选择多边形、折线或曲线等。形状 Shape 属性的改变只在所确定的范围内进行。



## 文字

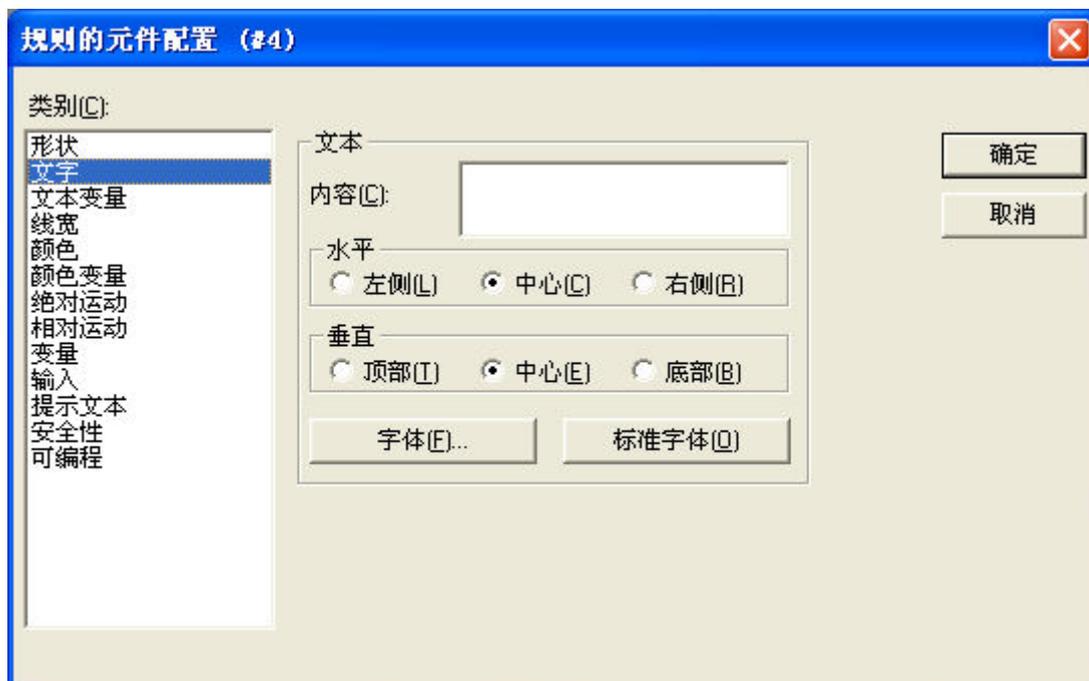
在视图对象中可以添加文本，用文字属性来设置，输入变量时可以使用 F2 功能键弹出快捷选项框。如图所示。

内容：在“内容”文本框中输入文本，按<Ctrl>+<Enter>组合键换行。

水平：在“水平”选项中设置文本在视图对象中的左、中、右位置。

垂直：在“垂直”选项中设置文本在视图对象中的上、中、下位置。

点击“字体”或“标准字体”按钮，可以设置文本的字体。



注：

1) 如果在文字中包含“%s”，那么文字中的这个位置在程序运行后，将被变量的特定形式的值所代替，具体的含义参见下表，

字符	含义
d, i	整型（十进制）
o	无符号的八进制数（0 不能作首位）
x	无符号的十六进制数（0 不能作首位）
u	无符号的十进制数
c	单字符
s	字符串
f	浮点型格式，在“f”前面以 x.y 的形式书写数字，前面的 x 表示整数部分显示的最少位数，后面的 y 表示小数部分的精确度（默认数字为 6，即 6 位小数）。例如 3.4f% 表示最大整数位是百位，精确到小数点后第四位。

2) 还可以在标准的 C 程序库中，使用特定的字符串，其含义如下表所示。

%a	星期缩写
%A	星期全写
%b	月缩写
%B	月全称
%c	日期时间
%d	一月中的天数（01-31）
%H	24 小时格式（00-23）
%I	12 小时格式（01-12）
%j	一年中的天数（001-366）
%m	月（01-12）
%M	分钟（00-59）
%p	12 小时 A.M/P.M 格式
%S	秒（00-59）
%U	一年的第几个星期（00-53），周日为星期第一天
%w	星期（0-6，周日是 0）
%W	一年的第几个星期（00-53），周一为星期第一天
%x	日期
%X	时间
%y	不含世纪的年（00-99）
%Y	含世纪的年
%z, %Z	时区名
%%	百分号

举例

1. 如果在文本内容中填入 %2.5f mm，则在程序运行时显示：32.8889 mm。
2. 若填入“%t%a %b %d.%m.%y %H:%M:%S”，则运行时显示：Wed Aug 28.01.07 16:32:45。
3. %Today is %d.%m.%y，则运行时显示：Today is 28.01.07

文本变量

在配置可视化界面对话框的‘文本变量’选项中可以定义一个变量,用于设置‘文本’选项中的文本的颜色和字体.最好借助输入助手(F2)输入变量名.

也可以使用结构 VisualObjectType中的变量来设置文本属性.参看‘可编程性’选项部分的描述;在那儿可以找到特定结构变量的允许值和它们的作用.

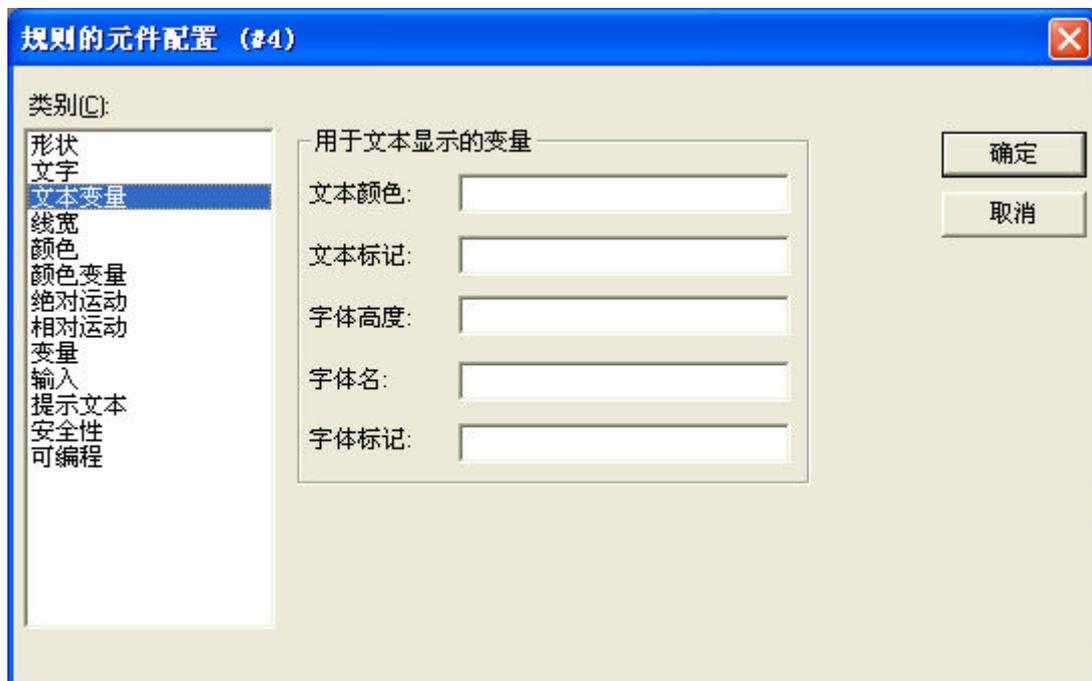
**注意:**如果在‘文本’选项中有相应的静态设置,它们将被动态文本替换。

在一个元件属性有多个定义的情况下,依据联机模式下数值可能被替换,要考虑优先的顺序关系.

对话框参数:

参数:	含义:	工程变量输入:	程序中变量的使用:	对应到结构 VisualObjectType 中的变量:
文字颜色:	文字颜色	"plc_prg. var_textcolor"	var_textcolor=16#FF00FF 颜色	dwTextColor
文本标记:	对齐 (右, 左, 中心...)	"plc_prg. textpos"	textpos:=2 字体右对齐	dwTextFlags
字体高度:	字体高度,以像素为单位	". fonth"	fonth:=16; 字体高度 16pt	ntFontHeight
字体名:	字体名	"vis1. fontn"	fontn:=arial; 使用 arial 字体	stFontName
字体标记:	字体显示 (粗体, 下划线, 斜体...)	"plc_prg. fontchar"	fontchar:=2 文字按粗体显示	dwFontFlags

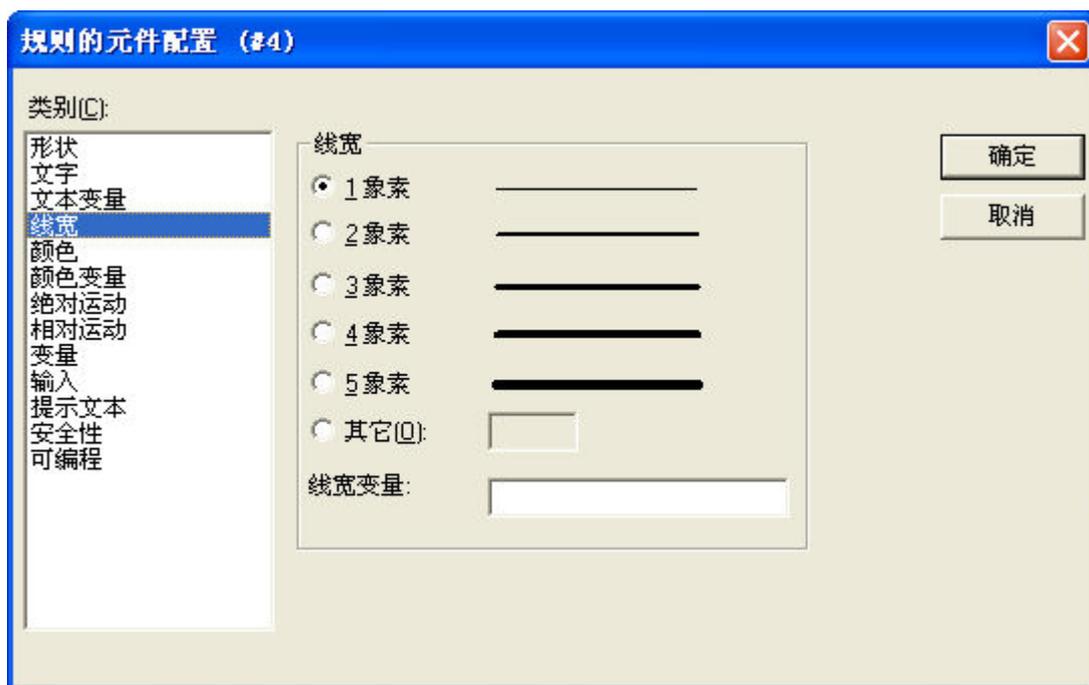
‘文本变量’选项对话框



## 线宽

线宽属性用来定义视图对象线条的宽度.下图所示为线宽设置对话框,共有 5 种线宽可以直接选择.选中“其它”选项可以输入线宽.在“线宽变量”中可以输入控制线宽的工程变量,使线宽产生动画效果,输入工程变量时可以使用 F2 功能键.联机模式下,动态线宽属性会替换静态线宽属性.

线宽设置对话框



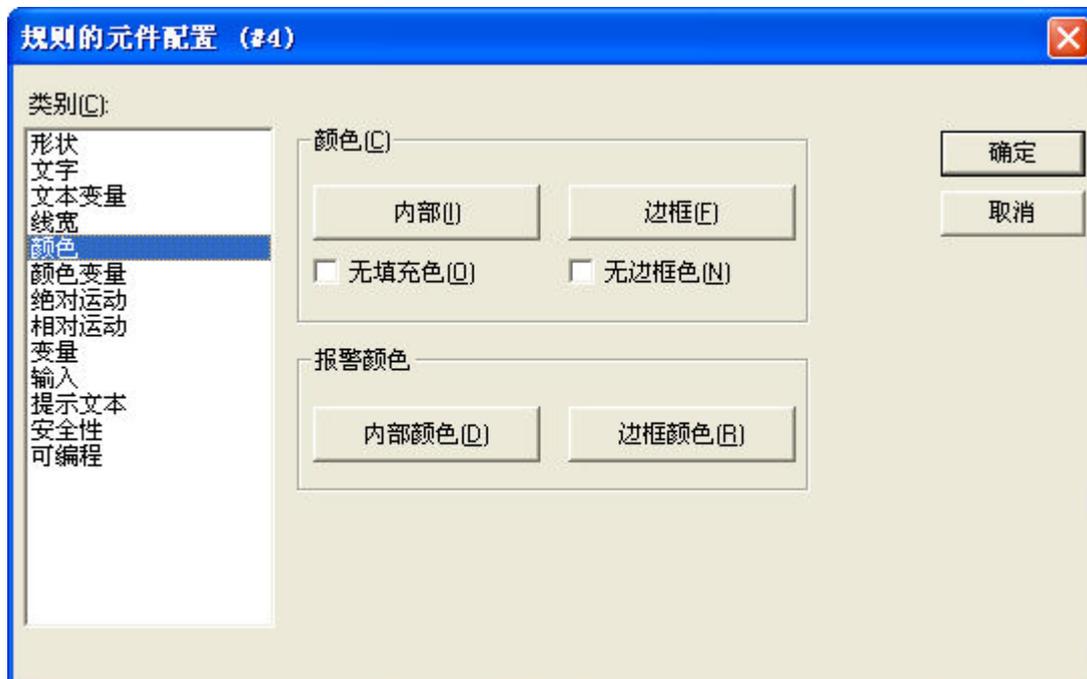
## 颜色

颜色属性用来定义视图元件的颜色和报警的颜色。选择'无填充色'和'无边框色'可以创建透明元件。

**注意：**如果有相应的静态设置,它们将被动态的替换。

在一个元件属性有多个定义的情况下,依据联机模式下数值可能被替换,要考虑优先的顺序关系。

颜色设置对话框



如果在'变量'选项的'改变颜色'字段输入一个布尔型变量,如果变量是'False',元件按'颜色'设置中的颜色显示,如果是'True',按'报警颜色'设置中的颜色显示。

按'边框'或'内部'按钮打开颜色选择对话框来设置颜色。

**注意：**只有 PLC 在联机模式下,元件的颜色才发生改变。

## 颜色变量

在这个对话框中输入工程变量(如:PLC\_PRG.color\_inside),它可以确定在联机模式下元件的颜色属性.也可以使用结构 VisualObjectType 中的变量来设置颜色属性.参看 '可编程性' 选项部分的描述;在那儿可以找到特定结构变量的允许值和它们的作用.

**注意:** 如果有相应的静态设置,它们将被动态的替换。

在一个元件属性有多个定义的情况下,依据联机模式下数值可能被替换,要考虑优先的顺序关系.



对话框参数:

参数:	含义:	工程变量输入:	程序中变量的使用:	对应到结构 VisualObjectType 中的变量:
填充颜色:	填充颜色	"plc_prg.var_fillcol"	var_fillcol=16#FF00FF 填充颜色是粉红色	dwFillColor
报警填充颜色:	当'改变颜色'变量为 True 时填充的颜色	"plc_prg.var_fillcol_a"	var_fillcol_a=16#FF00FF 报警填充颜色是粉红色	dwFillColorAlarm
边框颜色:	边框颜色	"plc_prg.var_framecol"	var_framecol=16#FF00FF 边框颜色是粉红色	dwFrameColor
报警边框颜色:	当'改变颜色'变量为 True 时边框的颜色	"plc_prg.var_framecol_a"	var_framecol_a=16#FF00FF 报警边框颜色是粉红色	dwFrameColorAlarm
填充标志:	可以启用 (False) 或禁用 (True) 填充颜色设	"plc_prg.var_col_off"	var_col_off:=1 不改变填充颜色, 边框保持有效	dwFillFlags

	置			
边框标志	边框的显示 (实线, 点划线等)	"plc_prg.var_linetype"	var_linetype=2; 边框用点划线显示	dwFrameFlags

颜色变量设置对话框



### 绝对运动

在可视化元件配置对话框中的'绝对运动'选项里,可以在'X 偏移量'或'Y 偏移量'输入变量.根据相应的变量值,元件可以在X或Y方向上偏移.在'缩放比例'中的变量将使元件根据当前值进行放大或缩小.它的数值作为比例因子使用,在起作用时要除以 1000,因此要缩小元件不需要使用实数类型变量.缩放涉及到参考点.

'角度'字段中的变量将使元件按照旋转点旋转一个角度(正值为顺时针旋转).单位是度.对于多边形来说每个点都旋转,即整个多边形旋转,对于其它可视化元件,在旋转时总是保持上边在顶部.

在元件上点击将出现一个中间是白十字线的黑圈,它是元件的旋转中心.可以按住鼠标左键进行拖拽.

**注意:** 在联机模式下,在'绝对运动'对话框中设置的变量值可以被用于定义同一属性的结构变量的值替换.参看'可编程序'。

在一个元件属性有多个定义的情况下,依据联机模式下数值可能被替换,要考虑优先的顺序关系.

绝对运动设置对话框



### 相对运动

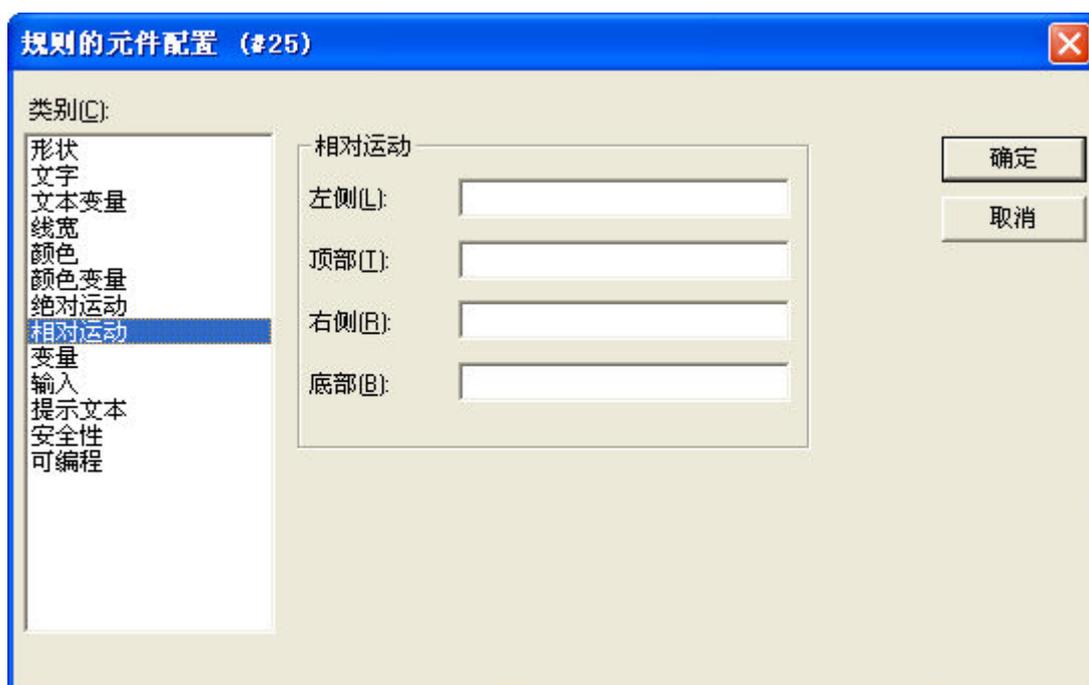
在可视化元件配置对话框中的'相对运动'选项里, 可以给各个元件的边界分配一个变量, 根据变量的数值, 相应的元件边界移动一个距离. 输入变量的最简单的方法是使用输入助手(F2).

四个输入项对应于元件的四个边. 角的基准坐标是0. 变量输入新值后, 对应的边以像素为单位移动这个距离, 因此变量应该使用整型变量.

**注意:** 在联机模式下, 在'相对运动'对话框中设置的变量值可以被用于定义同一属性的结构变量的值替换. 参看'可编程序'.

在一个元件属性有多个定义的情况下, 依据联机模式下数值可能被替换, 要考虑优先的顺序关系.

相对运动设置对话框



### 变量

变量设置对话框



在可视化元件配置对话框中的‘变量’选项里, 可以输入描述可视化元件状态的变量. 输入变量的最简单的方法是使用输入助手(F2).

**注意:** 在联机模式下, 在‘相对运动’对话框中设置的变量值可以被用于定义同一属性的结构变量的值替换. 参看‘可编程性’.

在一个元件属性有多个定义的情况下, 依据联机模式下数值可能被替换, 要考虑优先的顺序关系.

在‘不可见’和‘改变颜色’字段中输入布尔型变量, 由变量值来决定动作. 如果‘不可见’字段中的变量是 False, 则可视化元件可见, 否则不可见.

取消输入: 如果此处变量是 True, 则忽略‘输入’选项中的设置.

改变颜色: 如果此处变量是 False, 可视化元件按默认颜色显示, 否则按报警颜色显示.

文本显示:

如果已经在“文本”的“内容”中输入了“%s”, 则联机模式下, “文本显示”中定义的变量值将会替换“%s”显示在文本区域。

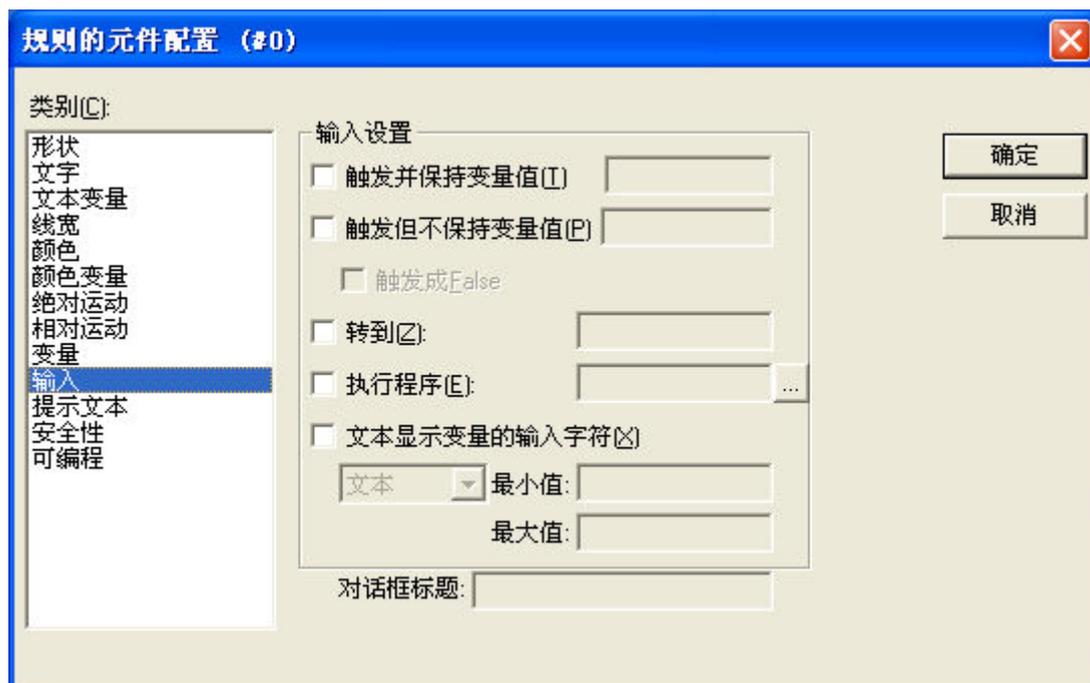
如果已经在“文本”的“内容”中输入了“%<PREFIX>” (PREFIX 必须是字符串), 那么变量所对应的数值应作为 ID 号输入到‘文本显示’中, ID 和 PREFIX 组合在一起对应 XML 文件中描述的文本. 在联机模式下它们对应的文本将会替换“%<PREFIX>”. 因此可以联机修改文本显示. 参看‘设置’‘语言’选项.

如果希望在联机模式下使用键盘编辑变量值, 可以使用‘输入’选项下的‘文本显示变量的输入字符’来设置.

工具提示: 此处使用的变量是字符串型变量, 在联机模式下当鼠标位于此元件区域内时显示变量值(字符串).

## 输入

‘输入’设置对话框



**触发并保持变量值：**如果激活此选项，在联机模式时，每次点击可视化元件都将触发变量。可以通过<F2>输入变量，鼠标每点击一次，逻辑变量的值都会从 TRUE 变成 FALSE，再点击从 FALSE 变成 TRUE。

**触发但不保持变量值：**如果激活此选项，在联机模式时，字段中的布尔变量值将在 TRUE 和 FALSE 之间切换。把鼠标光标放在对象上，按下鼠标键不要松开。如果激活‘触发成 False’选项，当按下鼠标时，变量值被设置成 FALSE，否则设置成 TRUE。当释放鼠标时，变量的值变回初始值。

**转到：**如果激活此选项，在字段中输入同一工程中的可视化界面名，它是在联机模式时用鼠标点击元件后期望打开的界面。

允许的输入是：

当前工程中可视化界面名(参看对象管理器)

如果要转到包含‘占位符’的界面中，在调用时占位符可以直接被变量名或文本替换，但必须遵守下列语法：

<Visuname>(<Placeholder1>:=<Text1>, <Placeholder2>:=<Text2>,..., <Placeholder n>:=<Textn>).

在编译过程中将检查文本是否占位符列表中定义的文本相匹配，如果不匹配将产生警告。

例如：

调用界面 visul，在界面 visul 中使用的占位符\$var\_ref1\$和\$var\_ref2\$将分别被变量 PLC\_PRG.var1 和 PROG.var1 替换：

```
visul(var_ref1:=PLC_PRG.var1, var_ref2:=PROG.var1)
```

如果在此处输入了类型为 STRING 的程序变量(如 PLC\_PRG.xxx)而不是可视化对象，那么这个变量要定义成可视化名(如 ,visul’)则当点击鼠标时，程序变量被赋值(如. xxx:= ,visul).

如果输入命令 „ZOOMTOCALLER”，在联机模式下点击元件将回到上次打开的界面。

**注意：**隐含变量 CurrentVisu (类型为 STRING，作为隐含(系统)变量使用，参看‘隐含变量’)保存当前打开的可视化界面名称。例如在应用中可以控制当前要打开哪个界面。但要注意，当编译器的版本小于 V2.3.7.0 时，如果在工程中没有包含库 SysLibStr.lib，那么可视化界面的名称必须使用大写字母(参看‘创建一个可视化对象’)。例如：CurrentVisu:=‘PLC\_VISU’；

**执行程序：**如果激活此选项，可以在字段中输入‘ASSIGN’或‘INTERN’指令，在联机模式下，当用鼠标点击元件时可以执行这个指令。按‘...’可以打开配置对话框，在对话框中可以选择命令(添加)和设置指令的执行顺序(上移，下移)。

**注意：**当可视化界面是唯一的操作接口时，这个功能非常有用。参看‘用于操作的可能的特殊输入’。

文本显示变量的输入字符：如果激活此选项, 在联机模式下, 可以将文本输入到可视化元件的编辑字段. 按<Enter>键后输入值写入到'变量'文本显示'中定义的变量中.

在滚动条中选择联机模式下的输入方式:

文本: 打开编辑字段, 输入数据.

Numpad 和 Keypad: 它们分别打开数字键盘和字母键盘, 然后输入所需数据. 如果使用的是触摸屏来操作界面, 这个功能非常游泳. 可以在最小值和最大值中定义输入数值的范围.

**注意:** 在使用目标可视化时, 由于可以在鼠标点击后通过特殊接口功能获得用户输入信息, 所以要考虑使用的可能性.

在一个元件属性有多个定义的情况下, 依据联机模式下数值可能被替换, 要考虑优先的顺序关系.

### 工具提示文本

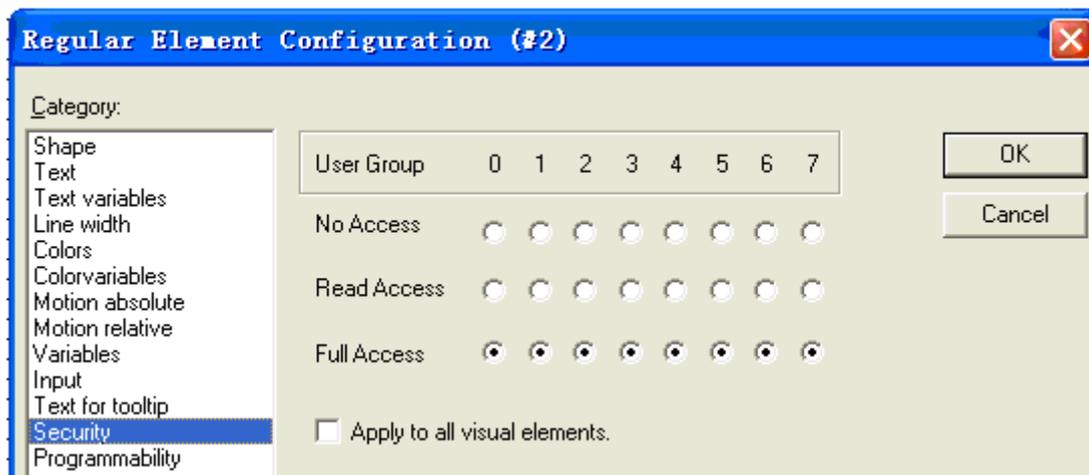
工具提示文本提供了一个文本输入字段, 当在联机模式下光标移动到元件上时显示输入的内容. 在“内容”文本框中输入文本, 按<Ctrl>+<Enter>组合键可换行。



### 安全属性

安全属性 (Security) 用来定义不同的用户组的操作方式和可视化界面的显示。可以通过给不同的可视化元件定义相应的访问权限来实现这个. CoDeSys 有 8 个用户组 (参看'工程'对象'属性'或'Project'工程'用户组密码'). 在访问权限对话框中通过激活相应的选项来分配元件的权限。

访问权限对话框



在联机模式下可视化元件的权限:

无权限	元件不可见
只读权限	元件可见，但不可操作，即无输入权限。
完全访问	元件可见，也可操作

如果给某个元件分配的权限也适合于所有其它元件, 激活选项' 适用于所有可视化元件'.

**注意:** 在'工程'对象'属性'中为可视化对象设置的访问权限不依赖于特定可视化元件的权限!

### 可编程性

一个可视化元素的属性不仅可以用一个静态设定值或用一个“标准”工程变量来定义，而且可以用一个结构变量的分量来定义。结构变量是专门用于编写可视化元件的。

为了达到这个目的, 必须使用库文件 SysLibVisu.lib 中的结构 VisualObjectType. 它的结构分量可以用于定义元件的大部分属性.

**注意:** 在一个元件属性有多重定义的情况下, “标准”工程变量的值将改写结构变量的值, 同时两者都会改写一个静态定义。

为了使用一个结构变量来配置元件属性, 需做下列工作:

打开配置对话框, 选择‘可编程性’选项, 在字段“对象名”内输入一个新的、唯一的名字! 要输入对象名, 必须在复选框中用鼠标激活此选项。此变量被自动声明为 Visual ObjectType 类型, 它是一个包含在库 SysLibVisu.lib 内的结构。该声明是隐含作出的, 对用户来说是不可见的。要确保该库文件已包含在库管理器内。

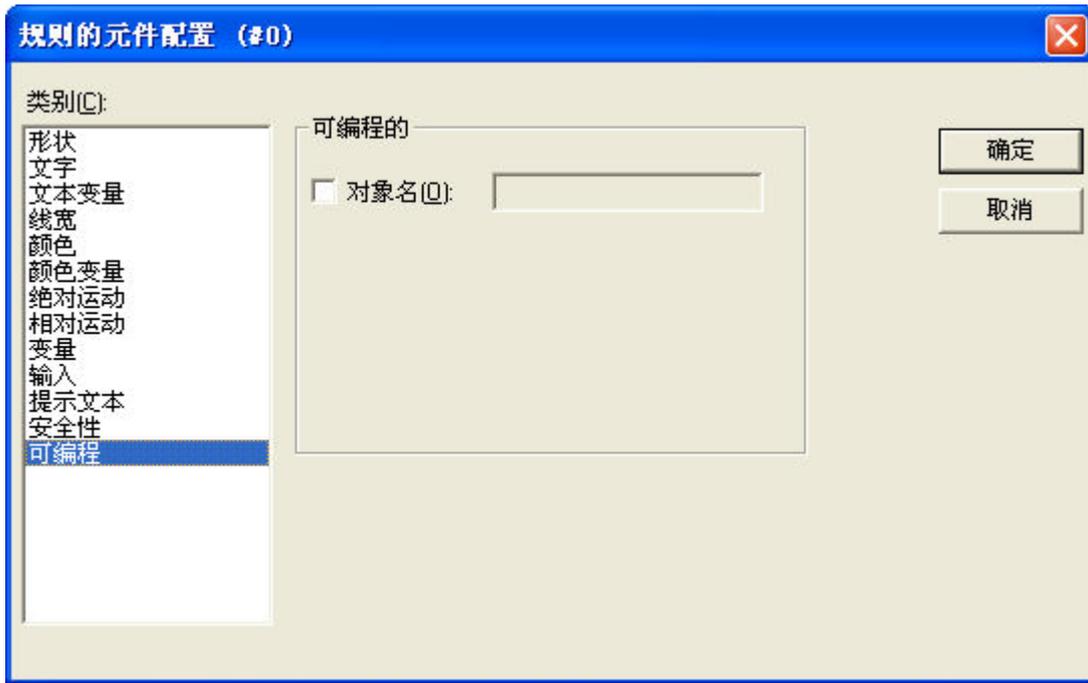
在下次编译后, 在工程内就可采用新分配的结构变量。(提示: 激活'工程'选项'编辑器'中的智能功能‘显示结构变量’, 从而在变量名字后输入一个点后, 就可得到结构分量的选择列表)

例子: 如果你已为一个可视化对象定义了一个对象名‘Visul-line’, 那么就可为这个元件编写线宽, 例如“Visul-line.nLineWidth:=4”。

结构 Visual/ObjectType :

下表表示此结构的现有分量及在配置对话框的不同类目中对相应条目的引用。

可编程性对话框



VisualObjectType 的结构:

下表列出了结构中所有变量, 这些变量对应于不同的配置对话框:

在变量名前表明变量的类型:

n INT  
dw DWORD  
b BOOL  
st STRING

变量 (+类型)	功能	举例 (定义的对象名是"vis1") (在 CodeSys 中的可视化编辑器)	对应的配置对话框中的输入项:
nXOffset : INT;	X-方向上移动元件	vis1.nXOffset:=val2; (元件在 X 方向上移动到 val2)	- 绝对运动: X 偏移量
nYOffset : INT;	Y-方向上移动元件	vis1.nYOffset:=22; (元件在 Y 方向上移动到 val2)	- 绝对运动: Y 偏移量
nScale : INT;	改变大小	vis1.nScale:=plc_prg.scale_var; (元件大小按 plc_prg.scale_var 值发生变化)	- 绝对运动: 缩放比例
nAngle : INT;	绕元件中心旋转	vis1.anglevar:=15; (元件顺时针旋转 15 度)	- 绝对运动: 角度
bInvisible : BOOL;	元件可见/ 不可见	vis1.visible:=TRUE; (元件不可见)	- 颜色: 无填充色 + 无边框色  - 颜色变量: 填充颜色 + 报警填充颜色
stTextDisplay : STRING;	元件中显示文本	vis1.TextDisplay:='ON / OFF'; 显示文本 ON / OFF	- 文本: '内容'
bToggleColor : BOOL;	当变量在 TRUE 和 FALSE 之间变化时, 改变颜色	vis1.bToggleColor:=alarm_var; (只要 alarm_var 变为 TRUE, 元件使用'颜色变量'或'颜色'中设置按 dwFillColorAlarm 和 dwFrameColorAlarm 定义的颜色显示.	- 输入: 触发并保持变量值 + - 变量: 改变颜色
bInputDisabled: BOOL;	如果是 FALSE: 忽略'输入'选项中的设置.	vis1.bInputDisabled:=FALSE; (不允许输入)	- 变量: '取消输入'
stTooltipDisplay: STRING;	工具提示文本	vis1.stTooltipDisplay:='Switch for .....';	- 工具提示: '内容'
dwTextFlags:DWORD;	文本位置: 1 左对齐 2 右对齐: 4 水平中心 8 顶部 10 底部 20 垂直中心 注意: 总设置在水平和垂直中心(两值相加)!	vis1.dwTextFlags:=24; (文本放置在元件的中心(4 + 20))	- 文字: 水平和垂直选项 - 文本变量: 文本标记
dwTextColor : DWORD;	文本颜色(颜色定义参看表下面的说明)	vis1.dwTextColor :=16#00FF0000; (文本是蓝色)	- 文字: 字体   颜色 - 文本变量: 文本颜色
nFontHeight : INT;	字体高度, 以像素为单位. 范围是 10-96	vis1.nFontHeight:=16; (文字高度 16 pt)	- 文字: 字体   大小 - 文本变量: 字高
dwFontFlags :	字体样式:	vis1.dwFontFlags:=10;	- 文字: 字体   样式

定义颜色值:

例如: `e1.dwFillColor := 16#00FF00FF;`

颜色按十六进制输入,它由蓝/绿/红 (RGB) 组成. "16#"后面必须先添加两个 0 以便满足 DWORD 型. 每个颜色都可以得到 256 个数值(0-255).

FF 蓝色

00 绿色

FF 红色

闪烁可视化元件 定义全局变量 'blink1',类型是 `VisualObjectType`,将它设置到一个矩形元件.在程序中修改结构变量的值.

```
PROGRAM PLC_PRG
```

```
VAR
```

```
n:INT:=0;
```

```
bMod:BOOL:=TRUE;
```

```
END_VAR
```

```
(* Blinking element *)
```

```
n:=n+1;
```

```
bMod:=(n MOD 20) > 10;
```

```
IF bMod THEN
```

```
  blinker.nFillColor := 16#00808080; (* Grau *)
```

```
ELSE
```

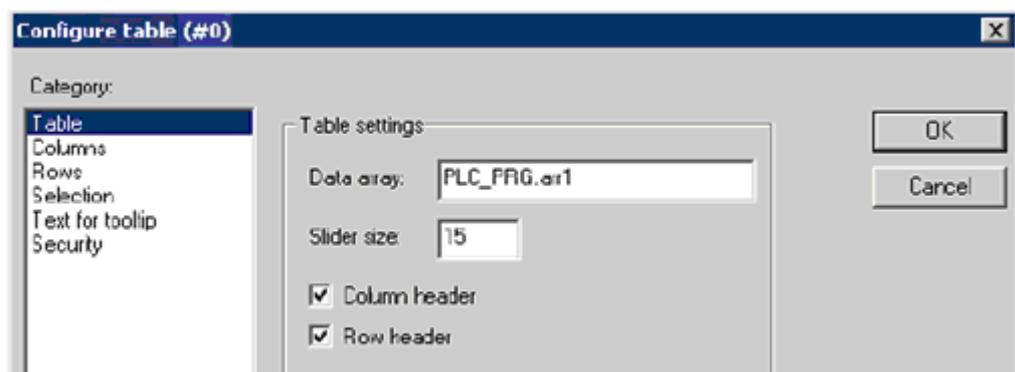
```
  blinker.nFillColor := 16#00FF0000; (* Blau *)
```

```
END_IF
```

## 表格

为了在可视化界面显示一个数组,可以插入一个表格,插入表格后将打开表格配置对话框.除了'工具提示'和'安全属性'可以用于这个可视化元件外,下面的选项也可用于配置表格的内容显示:

表格配置对话框



表格设置如下:

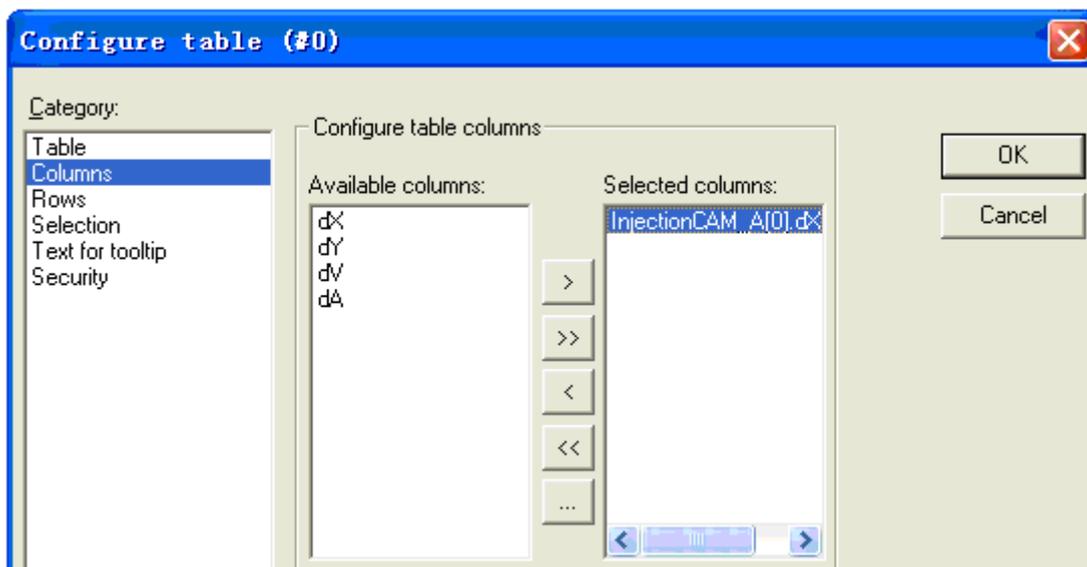
数组(Data array):输入在表格中将要显示的数组名称,建议使用输入助手<F2>.

滑尺尺寸(Slider size):当数组的大小超过表格的高度时,显示滑尺以便显示看不见的部分.

标题栏(Column header)和行标题(Line header):如果期望显示它们,激活相应的选项.行标题是数组的索引号(表格的第一列),在选项'列(Column)''中定义列标题.

'列(Column)''设置

列设置对话框



在列设置对话框中设置表格元素, 在左侧的窗口中列出了可选的所有元素, 它们都是按照数组的索引号来处理. 如果是结构数组, 它们将是一个结构组件.

使用'>'按钮可将选择的元素移动到右侧窗口, 在右侧窗口的元素将在表格中显示. 点击'>>'按钮可将所有元素移动到右侧窗口. 同样的, 点击'<'和'<<'可删除右侧窗口中的元素. 如果要修改表格中每个元素的显示样式, 可以双击元素或按'...'按钮, 这将打开列设置对话框.

配置'表格'列'列属性'对话框



编辑列标题和列宽:

通常, 在列标题 (Column header) 将自动创建一个标题 (如, 如果是结构数组, 标题为 "PLC\_PRG.arr1[INDEX].iNo", 'iNo' 表示的是结构中的变量). 你可以修改这个标题. 此外可以设置列宽 (字符数).

配置列中的元素:

在默认方式下, 表格单元只显示为一个矩形且不可编辑. 但是如果在选择列后激活'编辑模板', 可以修改此列中的字段参数设置, 如线宽, 文本输入等. 模板对此列中所有字段都起作用, 模板也可以借助可视化元件配置对话框进行编辑.

如果期望配置列中的几个特定字段, 可以使用占位符来确定行和列: \$ROWCONST\$, \$COLCONST\$, INDEX. (INDEX 与 \$ROWCONST\$ 具有相同的作用).

占位符使用的例子

例 1:

如果监控布尔型数组 "arr1 [0..2] of BOOL" (表格只有 1 列), 在联机模式下, 当鼠标点击表格单元时, 单元格颜色变为红色, 同时数组元素也被触发, 反之亦然.

为了实现这个, 激活列配置对话框中的'使用模板', 按如下说明定义模板:

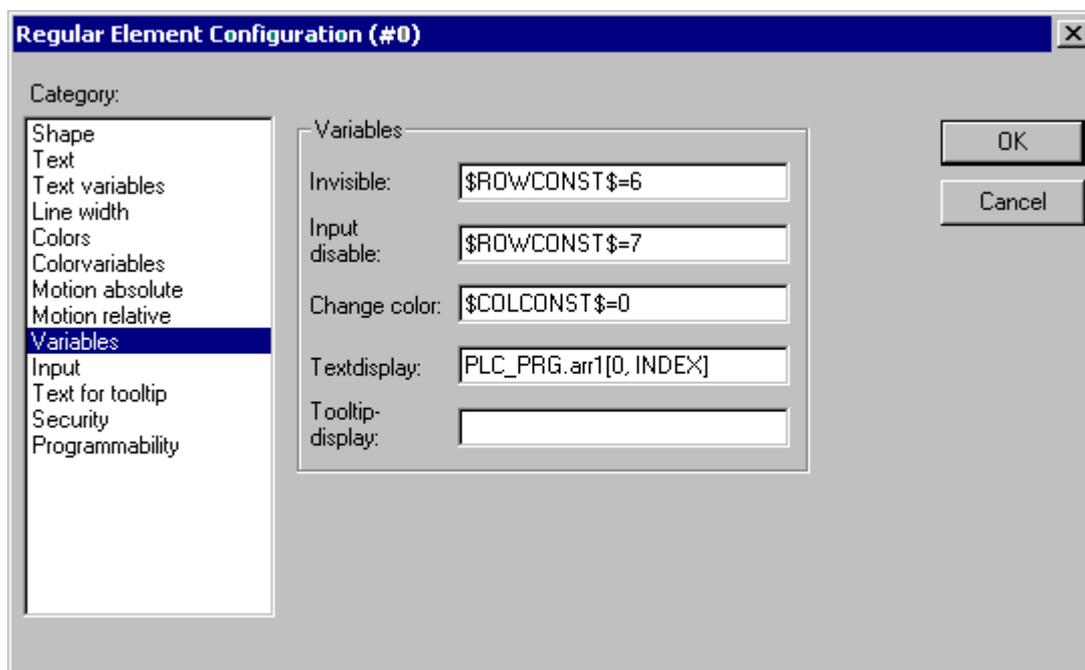
选项'输入': 触发变量且保持变量值: "PLC\_PRG.arr1[INDEX].

选项'颜色': 报警颜色为红色.

选项'变量' '改变颜色': "PLC\_PRG.arr1[INDEX].

例 2:

下面创建的模板用于一个表格列,它代表数组索引号 0:



上面的对话框用于列模板设置(点击'编辑模板')

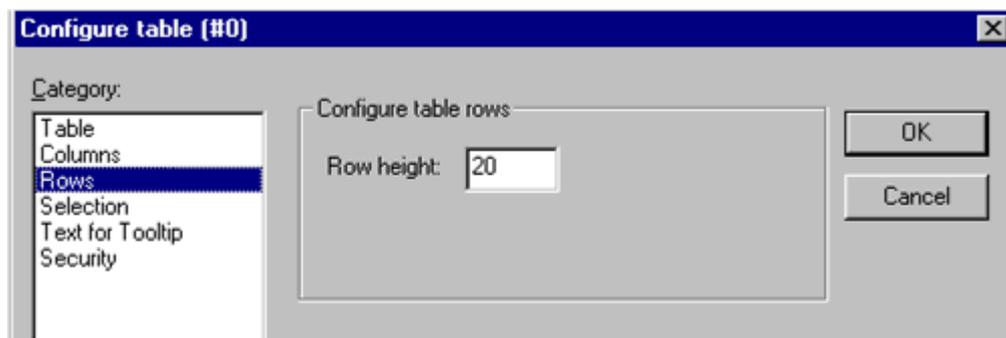
上图的意思是:

- 第 6 行不可见
  - 第 7 行不能输入
  - 在列 0 中的所有字段中,颜色将改变为报警颜色,列的其它单元格保持原来的颜色.
  - 文本显示的内容可以改变,这儿的配置与'文字' '内容'中的's'一起来决定表格单元格中内容的显示.
- 占位符输入可以使用,,AND “ 或 “OR “指令;例如: "\$ROWCONST\$=1 OR \$ROWCONST\$=3" 使得单元格不同的设置.

是否使用配置模板可以激活或取消选项'使用模板'.

行 (Row) 设置

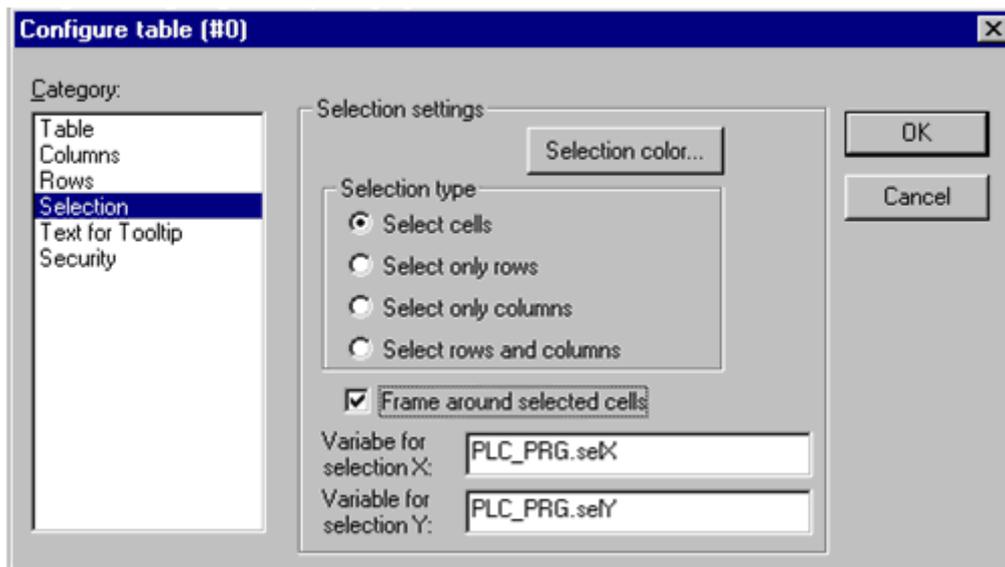
行设置对话框:



行高:输入期望的高度值,单位是像素.

选择(selection)设置

'选择' 设置对话框



设置与选择单元格有关的参数：

Here you can set the following parameters concerning the selection behaviour within the table:

选择颜色： 按此按钮定义单元格选中时的颜色. 它将打开标准的颜色对话框以便选择所需颜色.

选择类型： 在联机模式下, 当用鼠标点击表格中的单元格后, 将选择表格中的哪些部分：

选择单元格： 只有点击的单元格被选中..

只选择行： 单元格所在行被选中.

只选择列： 单元格所在列被选中.

选择行和列： 单元格所在行和列都被选中.

所选单元格周围显示边框： 选中单元格用一个框包围.

选择 X(行)的变量, 选择 Y(列)的变量： 在此处输入一个工程变量, 它们分别表示所选单元格的 X 和 Y 索引号.

创建一个表格元件来显示一个结构数组

声明下面的结构：

```
TYPE strucTab :
```

```
STRUCT
```

```
iNo: INT;
```

```
bDigi : BOOL;
```

```
sText:STRING;
```

```
byDummy: BYTE;
```

```
END_STRUCT
```

```
END_TYPE
```

在 PLC\_PRG 中声明下面的结构数组：

```
arr1:ARRAY [1..5] OF strucTab;
```

and the following variables:

```
selX:INT;
```

```
selY:INT;
```

创建一个可视化界面并插入一个表格元件. 配置如下：

表格： 数据数组：“PLC\_PRG.arr1”

列： 将结构中的变量 iNo, bDigi, sText 移动到右侧窗口中, 在右侧窗口中鼠标双击第一项 (PLC\_PRG.arr1[INDEX].iNo), 则打开对话框, 用“Number”替换默认标题. 按‘确认’, 然后重复定义另外两列的

标题(如, "Value" 和 "Text").

选择: 在 '选择 X(行)的变量' 中输入: "PLC\_PRG.se1X" 在 '选择 Y(列)的变量' 中输入: PLC\_PRG.se1Y". 激活选项' 所选单元格周围显示边框'. 按' 选择颜色' 后选择' 黄色'. 关闭配置对话框. 表格元素显示如下:

	Number	Value	Text
1			
2			
3			
4			
5			

左侧边界显示了数组索引号, 在上面显示了所选结构变量的标题. 当将光标位于两列的分界线, 会出现一个水平双箭头, 移动鼠标可以修改列宽.

在联机模式, 在表格单元格里显示数组变量值. 当用鼠标点击单元格时, 单元格变为黄色且边框加粗显示. 如:

	Number	Value	Text
1	0	TRUE	text1
2	33	TRUE	text2
3	55	FALSE	abc
4	0	FALSE	
5	0	TRUE	

第一个按钮是设置选择后表格颜色的变化, 在选择类型 (selection type) 选项中可以定义点击时选择的区域, 分别是单元格 (cells)、整行 (only rows), 整列 (only column) 和全部 (rows and columns). 还可以定义选择区域的框架.

在表格中选择单元格后显示的表格形式如图例所示。

### ActiveX 控件

ActiveX 控件用于在一个可视化中显示一个被动的 ActiveX 控件. 这个控件用于基于 WIN32 系统的 CoDeSys HMI 和 CoDeSys 目标可视化中。

双击插入到可视化界面中元件打开它的配置对话框. 该配置对话框包括三个子对话框, 分别是选择控件类型、定义调用方法和显示配置:

#### 1. 控制:

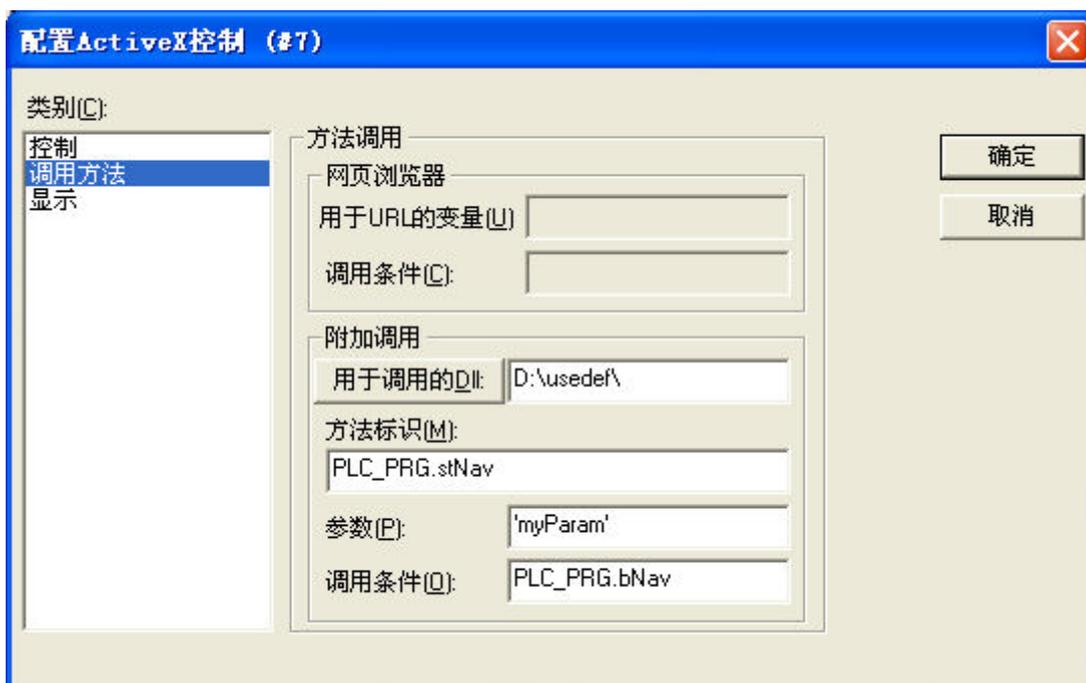
ActiveX 控件中的控制选项



这个对话框列出了所有已经注册到计算机中的 ActiveX 控件，从列表中选择其中一个控件。

## 2. 方法调用:

ActiveX 控件中的调用方法选项



在此，配置所选控件的调用方法:

网页浏览器:

只有配置的控件支持 IWebBrowser 接口（例如：IE 或者 Mozilla 浏览器）才可以编辑这些字段。在这种情况下 CoDeSys 能够直接调用 Navigate 方法。（其它方法必须通过用户自己定义的 DLL 库调用，见下面的“附加调用”）

在“用于 URL 的变量”字段中输入一个 URL 作为参数值,它对应于一个定义 URL 的字符串型工程变量,只要“调用条件”字段中的变量为 TRUE（上升沿）时，这个浏览器将被立即调用。如果“调用条件”一栏中没有指

明变量，目标可视化会在每一个可视化界面任务周期中都调用这个浏览器！

附加调用：

借助用户定义的 WINDOWS-DLL，用户可以定义 ActiveX 控件的方法调用用于控制控件在调用时的执行。为此，必须在“用于调用的 DLL”中输入 DLL，如果点击这个按钮，将打开文件对话框来浏览 dll，如果 DLL 文件在“工程选项”中指明的可视化文件路径中，则可以输入一个相对于这个目录的路径，否则就必须输入完整的路径。

**注意：**如果 DLL 用于一个带有“目标可视化”的运行系统中，这个 DLL 必须复制到运行系统中。如果在“目标可视化”中调用这个控件时，则只需要考虑目录中的文件名。

当“调用条件”定义的变量变为 TRUE（上升沿）时，DLL 被立即调用。如果“调用条件”一栏中没有指明变量，目标可视化会在每一个可视化界面任务周期中都调用这个 dll！

在创建 DLL 时需要注意：

DLL 必须输出一个名为 ExecuteActiveXCall 的函数原型：

```
void ExecuteActiveXCall(IUnknown* pUnk, char* pszId, char* pszParam, char* pszReturnBuffer, int nReturnBufferSize, DWORD* pdwReturnFlag);
```

这个函数和下面的参数将被调用，其中的参数在配置对话框中定义。

pszId :在“方法标识”字段中定义字符串变量

pszParam :在“参数”字段中定义数值

参数 pUnk 允许一个 Com(ActiveX-)接口的请求调用。用这个接口，能够调用任何一个带参数的 ActiveX 控件的方法。其中的参数放在一个字符串中。

参数 pszReturnBuffer, nReturnBufferSize 和 pdwReturnFlag 目前没有使用。

Dll 源文件举例(点击这里)：

如果支持 IWebBrowser 接口,这个 Dll 将调用控制的 GoBack 或 GoForward 方法.

通过参数 pszId 选择方法.

```
#include "stdafx.h"
```

```
#include <unknwn.h>
```

```
#include <exdisp.h>
```

```
BOOL APIENTRY DllMain( HANDLE hModule,
```

```
    DWORD ul_reason_for_call,
```

```
    LPVOID lpReserved
```

```
)
```

```
{
```

```
    return TRUE;
```

```
}
```

```
extern "C" __declspec(dllexport) void ExecuteActiveXCall(IUnknown* pUnk, char* pszId, char* pszParam, char* pszReturnBuffer, int nReturnBufferSize, DWORD* pdwReturnFlag)
```

```
{
```

```
if (strcmp(pszId, "IWebBrowser|GoBack") == 0)
```

```
{
```

```
IUnknown* pNewUnk;
```

```
IWebBrowser* pwb;
```

```
pUnk->QueryInterface(IID_IWebBrowser, (void**) &pNewUnk);
```

```
pwb = (IWebBrowser*) pNewUnk;
```

```
if (pwb)
```

```
{
```

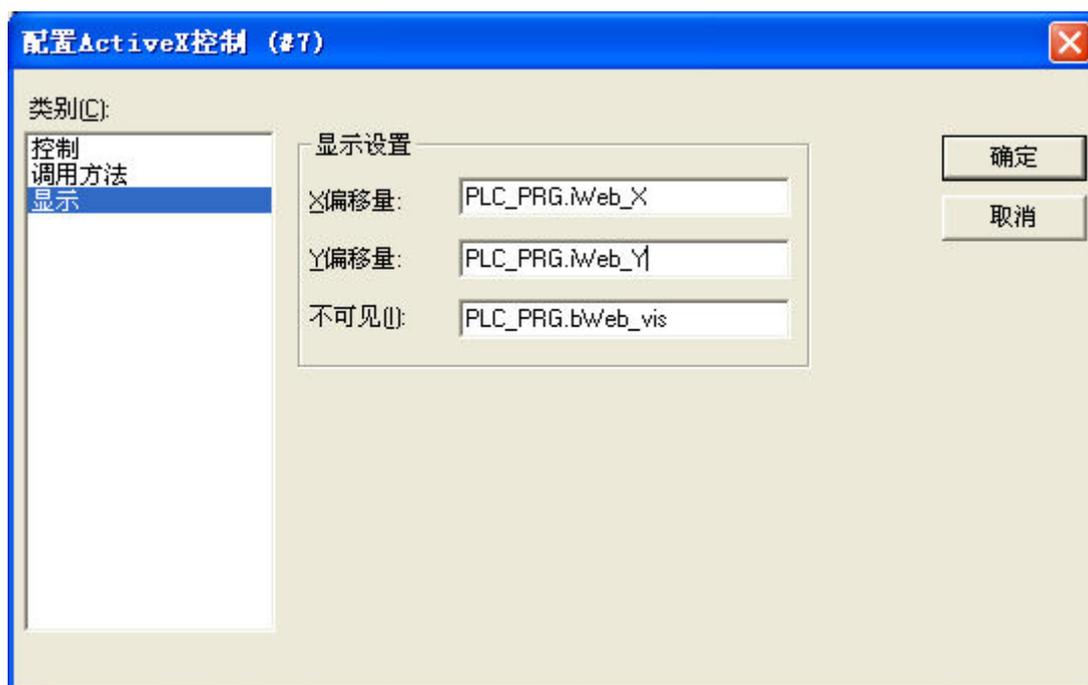
```

pwb->GoBack();
pwb->Release();
}
}
else if (strcmp(pszId, "IWebBrowser|GoForward") == 0)
{
IUnknown* pNewUnk;
IWebBrowser* pwb;
pUnk->QueryInterface(IID_IWebBrowser, (void**) &pNewUnk);
pwb = (IWebBrowser*) pNewUnk;
if (pwb)
{
pwb->GoForward();
pwb->Release();
}
}
}
}

```

### 3. 显示:

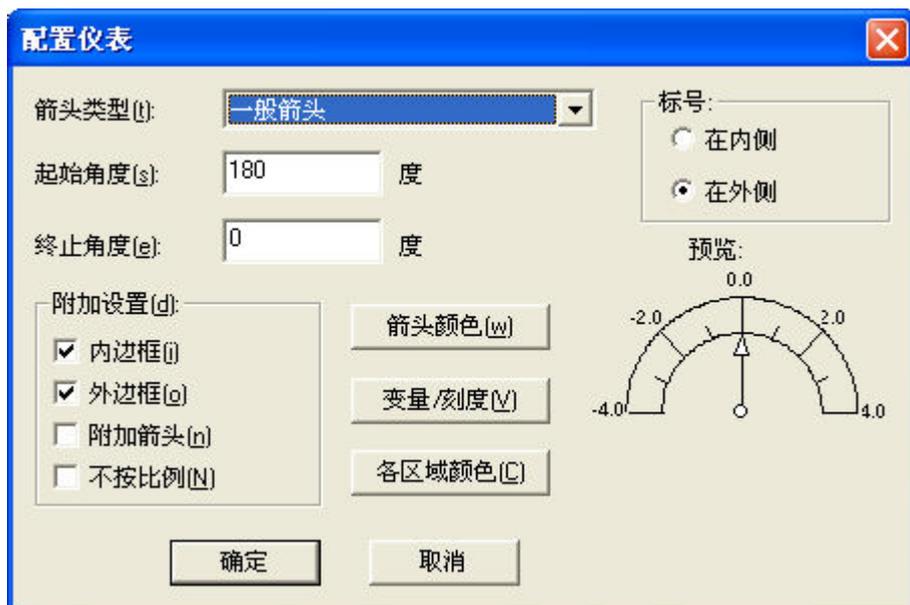
ActiveX 控件中的显示选项



在对话框中指明定义位置的变量(X 偏移量, Y 偏移量, 参看 '绝对运动'选项中的描述)和控件是否可见(不可见, 参看'变量'选项中的描述).

### 仪表

仪表元件设置对话框



在一个可视化对象中插入一个仪表元件将自动打开一个对话框。在对话框的右下角有预览，可以显示设置完下列参数后的结果：

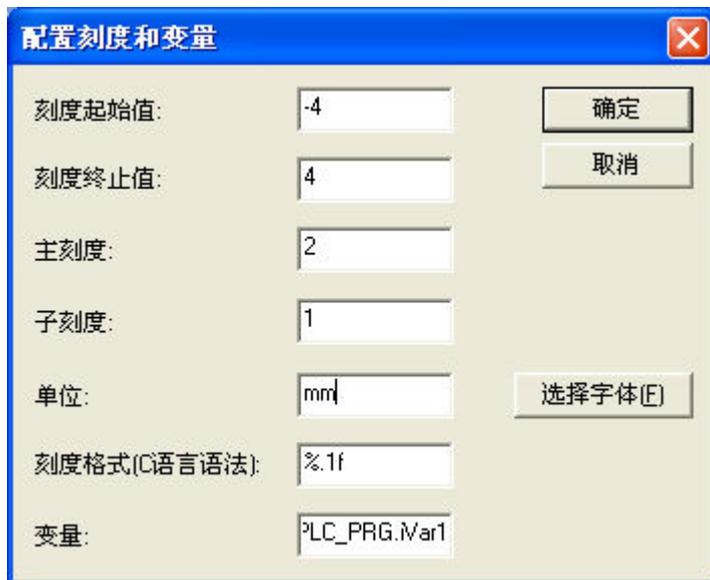
箭头类型：定义仪表指针的箭头类型。可选的类型：一般箭头，窄箭头，宽箭头，窄针箭头。

起始角度，终止角度：定义仪表圆弧起始和终止角度，以度为单位。（如：起始角度是 180 度，终止角度是 0 度将显示上半圆）。

箭头颜色： 按此按钮将打开标准颜色选择对话框或特定目标的颜色选择列表来定义指针的颜色。

变量/刻度： 按此按钮打开配置刻度和变量对话框。

配置刻度和变量对话框



刻度起始值，刻度终止值：刻度的最小值和最大值如“-4”和“4”。

主刻度：定义标记主刻度的间隔，即带标号的刻度线。如果输入“2”，是 2 的整数倍的刻度都将带标号。

子刻度：除了主刻度（标号+长刻度线），可在这儿定义子刻度线，它是不带标号的短刻度线。

单位：定义刻度单位，如“cm”或“sec”。它在指针的原点处显示。

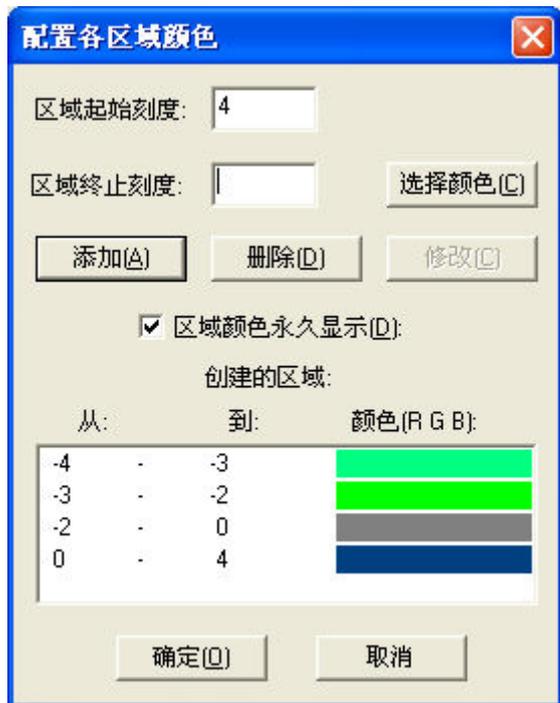
刻度格式（C-语言语法）：依据 C 语言语法定义刻度标号的格式；参看选项‘文字’中的描述。如：如果输入“%.1f”，刻度值用一位小数的浮点数表示，如“12.0”

变量：定义指针位置的变量，如“PLC\_PRG.posvar”。

选择字体：打开标准字体对话框，定义在仪表元件中使用的字体。（选择列表可以是特定目标的）。

各区域颜色：按此按钮打开 配置颜色区域对话框，为刻度的不同区域定义不同的颜色。

域颜色配置对话框



区域起始刻度, 区域终止刻度：输入相应的区域数值并定义此区域的颜色, 方法如下：

选择颜色：按此按钮将打开标准颜色选择对话框或特定目标颜色选择列表来定义区域颜色。按‘确认’按钮确认选择的颜色并关闭对话框，然后按‘添加’按钮，则设置的区域和颜色将添加到‘创建的区域’列表中。如果需要删除定义好的区域，选中这项后按‘删除’按钮。

如果选中区域颜色永久显示，定义的颜色范围将永久显示，否则在联机模式，只有当当前值在此区域内时才显示它的颜色。

标号：依据选中的是‘在内侧’还是‘在外侧’，刻度的标号将显示在圆弧内侧或外侧。

其它设置：

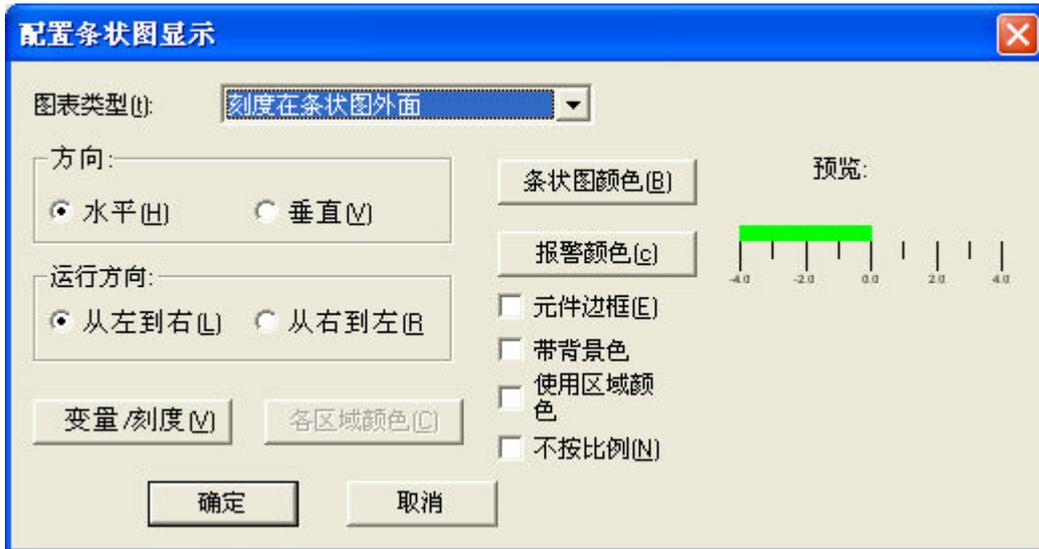
内边框，外边框：是否显示仪表元件圆弧的内外边框。

附加箭头：除了主指针外，一个小箭头指向刻度的当前值。

不按比例：选中此项后，仪表的大小不随元件大小的改变而改变。

## 条状图

条状图设置对话框



在一个可视化对象中插入条状图元件将自动打开一个对话框。在对话框的右下角有预览,可以显示设置完下列参数后的结果:

图表类型: 可选类型: '刻度在条状图外面', '刻度在条状图里面' 和 '条状图在刻度里面'.

方向: 条状图的方向是水平方式还是垂直方向.

运行方向: 条状图的伸长方向是从左到右还是从右到左.

条状图颜色: 按此按钮打开标准对话框来选择颜色. 定义条状图在一般状态下的颜色(非报警颜色). 如果激活'使用区域颜色'(参看下面), 此项将被禁用.

报警颜色: 按此按钮打开配置报警表对话框, 在此处定义显示报警颜色的数值和报警颜色: 在编辑字段输入限制值并选中报警条件是大于设置值还是小于设置值, 以便当值大于或小于限制值后显示报警颜色. 按报警颜色按钮打开标准对话框或或特定目标的颜色选择列表选择报警颜色. 按'确认'按钮确认选择的颜色并返回到主配置对话框. 如果激活'使用区域颜色'(参看下面), 此项将被禁用.

变量/刻度: 按此按钮打开配置刻度和变量对话框, 此对话框的使用与仪表元件中的一样.

元件边框: 选中此项后条状图被一个边框包围.

带背景色: 选中此项后条状图的整个显示范围以黑色作为背景, 否则只按当前值显示条状图的长度.

使用区域颜色: 选中此项后, 在'条状图颜色'和'报警颜色'中的设置将无效. 在这种情况下将使用定义的区域颜色, 在'各区域颜色'中设置区域颜色. 按'各区域颜色'按钮打开区域颜色设置对话框.(参看下面)为刻度的不同区域定义不同的颜色.

域颜色配置对话框



区域起始刻度, 区域终止刻度: 输入相应的区域数值并定义此区域的颜色, 方法如下:

选择颜色: 按此按钮将打开标准颜色选择对话框或特定目标颜色选择列表来定义区域颜色. 按' 确认' 按钮确认选择的颜色并关闭对话框, 然后按' 添加' 按钮, 则设置的区域和颜色将添加到' 创建的区域' 列表中. 如果需要删除定义好的区域, 选中这项后按' 删除' 按钮.

如果选中区域颜色永久显示, 定义的颜色范围将永久显示, 否则在联机模式, 只有当当前值在此区域内时才显示它的颜色.

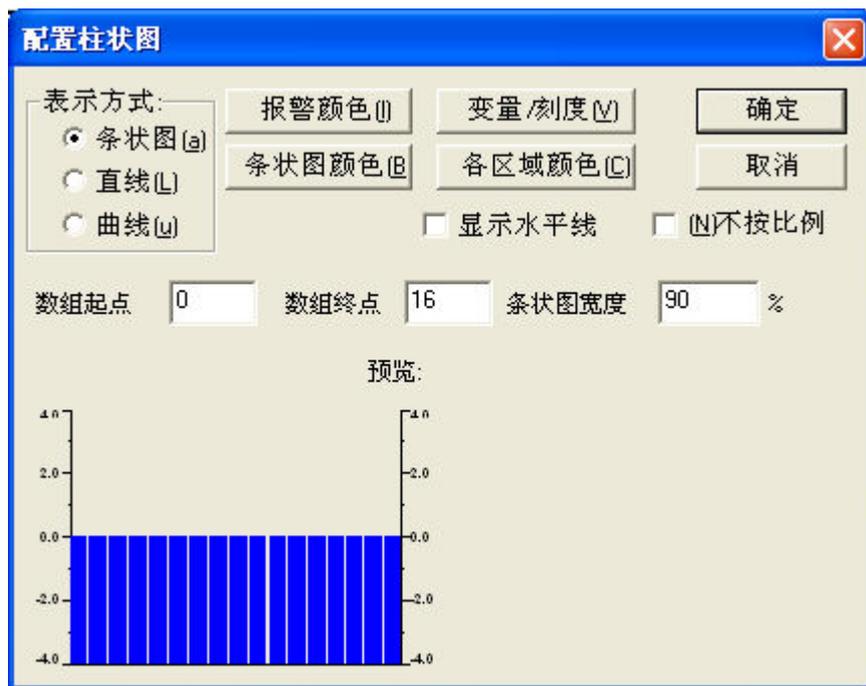
各区域颜色: 按此按钮打开打开' 配置区域颜色' 对话框, 在此处为刻度的不同区域定义不同的颜色. 这些定义只有在选中' 使用区域颜色' 才生效. 此对话框的使用与仪表元件中的一样.

不按比例: 选中此项后, 条状图的大小不随元件大小的改变而改变.

### 柱状图

为了在可视化界面显示一个数组, 数组元素的值通过并排的条状图或线来表示, 它们的高度表明元素的当前值.

柱状图配置对话框:



在一个可视化对象中插入柱状图元件将自动打开一个对话框。在对话框的左下角有预览,可以显示设置完下列参数后的结果:

表示方式: 选择柱状图的显示方式:条状图,直线,和曲线.

显示水平线: 选中此项,每个刻度上将增加一条横贯柱状图的水平线.

不按比例: 选中此项后,柱状图的大小不随元件大小的改变而改变.

报警颜色: 按此按钮打开配置报警表对话框,在此处定义显示报警颜色的数值和报警颜色:在编辑字段输入限制值并选中报警条件是大于设置值还是小于设置值,以便当值大于或小于限制值后显示报警颜色.按报警颜色按钮打开标准对话框或或特定目标的颜色选择列表选择报警颜色.按'确认'按钮确认选择的颜色并返回到主配置对话框.

变量/刻度: 按此按钮打开配置刻度和变量对话框,此对话框的使用与仪表元件中的一样.

各区域颜色: 按此按钮打开打开'配置区域颜色'对话框,在此处为刻度的不同区域定义不同的颜色.此对话框的使用与仪表元件中的一样.

条状图颜色: 按此按钮打开标准对话框来选择颜色.定义条状图在一般状态下的颜色(非报警颜色).

定义数组显示的范围:

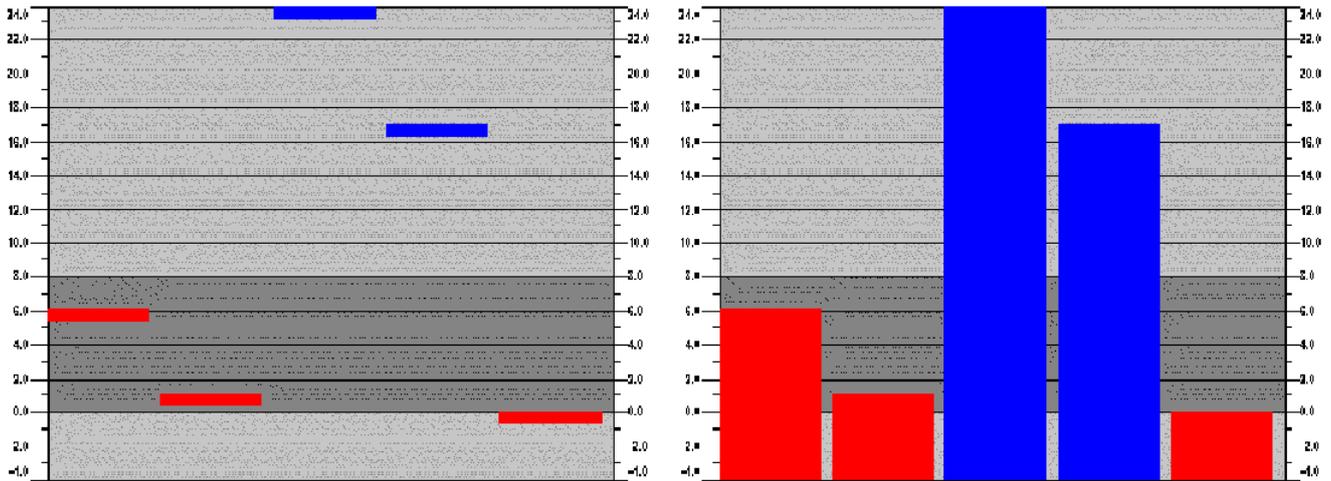
数组起点: 显示的第一个数组元件(索引号).

数组终点: 显示的最后一个数组元件(索引号).

条状图宽度: 定义条状图的显示宽度,以百分比为单位.

例如:

下图是柱状图(条状图和直线)在联机模式下的显示,它们表示整型数组 `arr1 [0..4]`.数组起点是"0",终点是"4",刻度起点是"-4",终点是"4",主刻度是"2",子刻度是"1",刻度范围 0 到 8 的颜色是深灰.当某个数组元素的值超过 8 时,条状图显示蓝色.在图中数组元素 `arr1[2]`和 `arr1[3]` 处于报警状态:



## 报警表

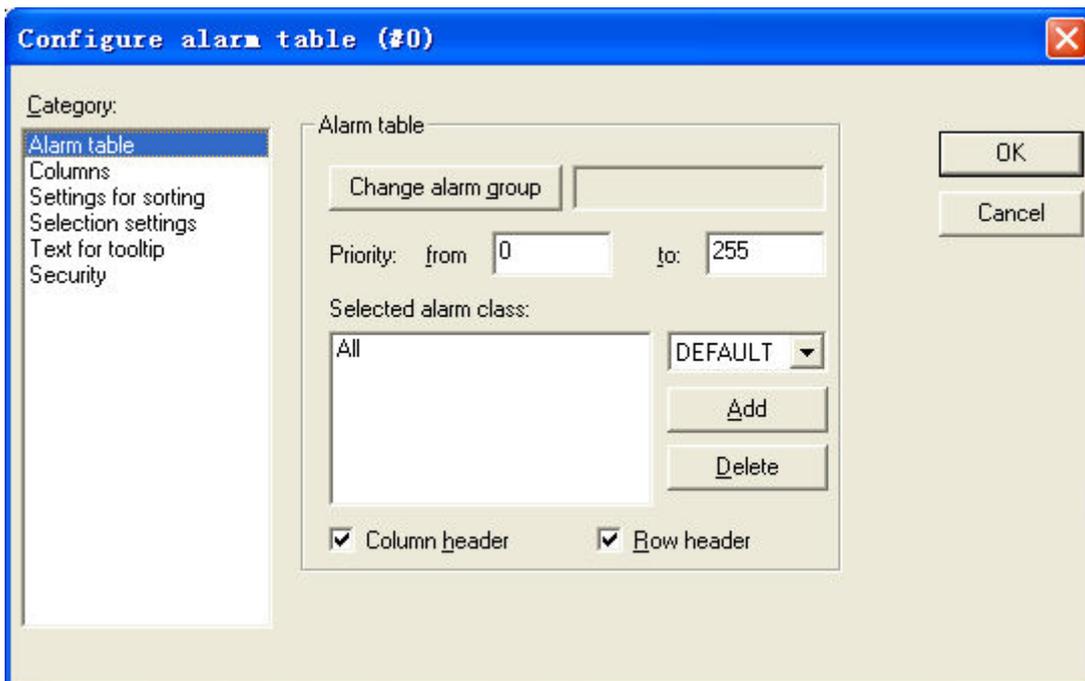
元件'报警表'用于查看联机模式下的报警,这些报警要首先要在 CoDeSys 的报警配置中配置。

**注意:** Depending 依据不同的目标系统,要考虑在 PLC 中处理报警的可能性。

插入报警表后将打开报警表配置对话框。除了'工具提示'和'安全属性'可以用于这个可视化元件外,下面的选项也可用于配置报警表的显示:

报警表选项:

报警表选项对话框



定义在报警表中显示的内容:

改变报警组: 按此按钮将打开已在报警配置中定义的报警组列表,选择期望显示的报警组(即使报警组只包含一个报警)。

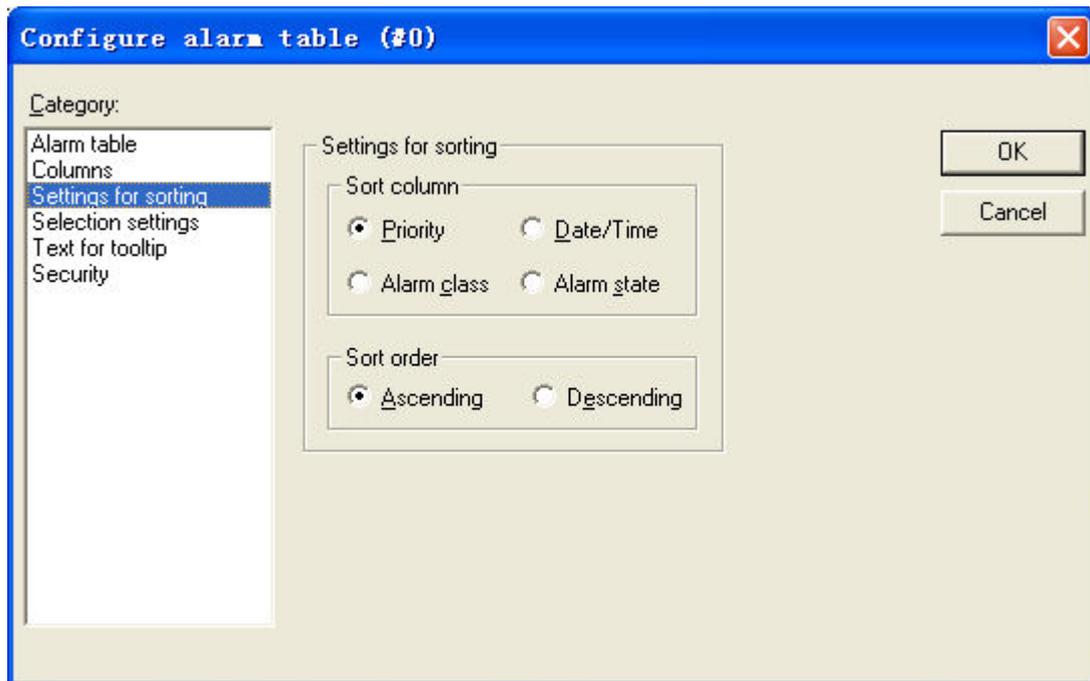
权限: 定义显示所有报警的权限.允许值从 0 到 255.

报警类: 选中期望显示的报警类然后按'添加'按钮将它添加到'已选报警类'窗口中.重复做此步以选择所有期望显示的报警类.如果想删除某个已选报警类,按'删除'按钮。

激活选项列标题和行标题,可以在报警表中显示标题。

设置分类选项:

设置分类选项对话框



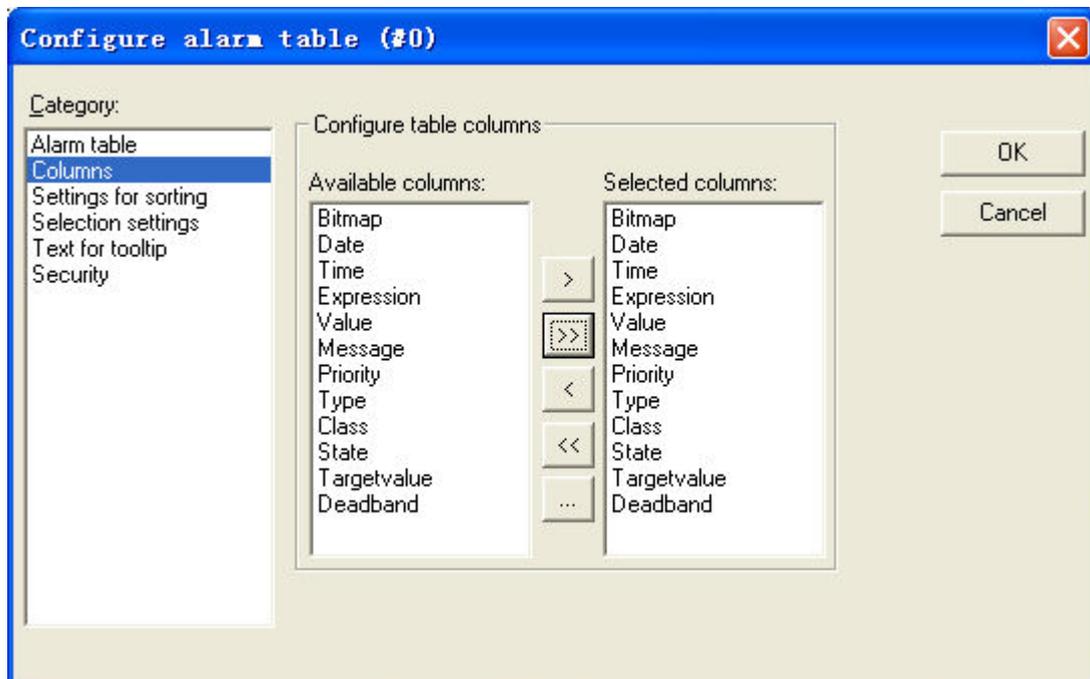
定义报警表分类的方式:

分类列: 根据权限,报警类,日期/时间或报警状态分类

分类排序: 升序或降序; 例如:根据权限升序排序,从 0 开始,后面的数越来越大.

列选项:

列选项对话框



定义在报警表中显示哪些报警参数: 可以定义的参数(除了报警发生的日期和时间以及在报警组中配置的报警状态以外)是: 位图, 日期, 时间, 表达式, 值, 信息, 权限, 类型, 类别, 状态, 目标值(用于报警类型 DEV+和 DEV-), 死区.

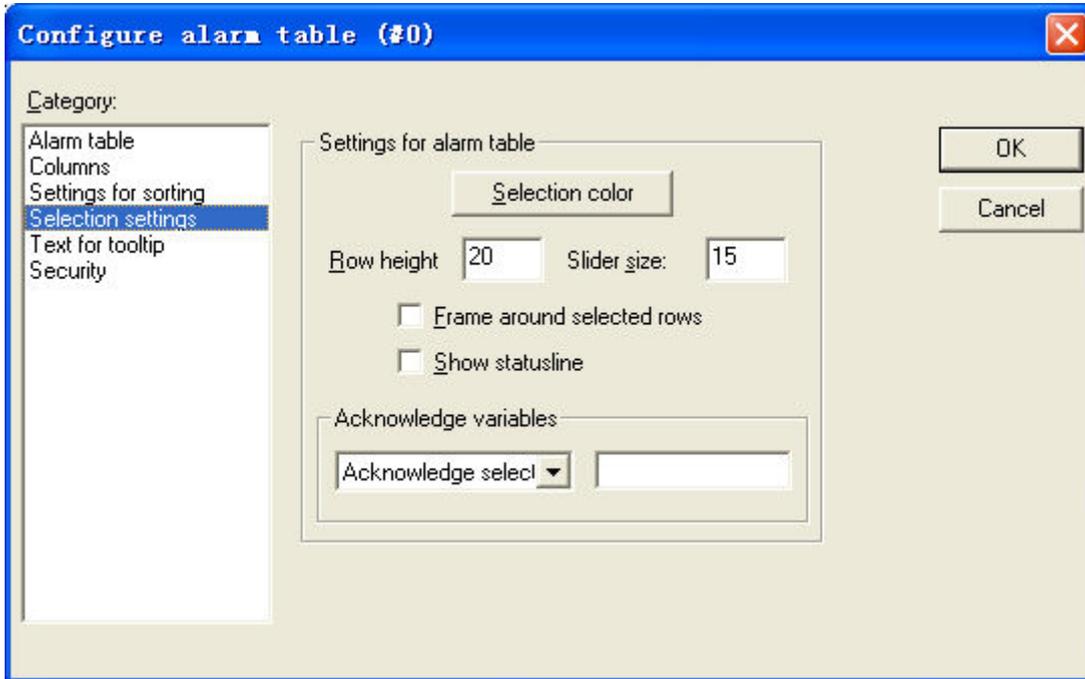
使用按钮 ">", ">>" 可以将左侧窗口中的一个或所有参数添加到右侧窗口.在右侧窗口中定义的选项将在报

报警表中显示,使用按钮"<"和"<<"可以删除已选的选项.

在右侧窗口中的某个选项上双击鼠标可以打开'列配置'对话框.在对话框中可以定义列标题和列宽.

报警表的选择设置选项:

报警表的选择设置选项对话框



设置表格字段的显示方式:

选择颜色: 按此按钮打开标准颜色对话框或特定目标的颜色选择列表来选择颜色.定义当选中字段后显示哪种颜色.

行高: 表格中每行的高度,单位:像素.

滑尺尺寸: 滑尺高度,单位:像素.

选中行带边框: 选中此项后,所选的行带边框显示.

显示状态行: 选中此项后,在报警表下面显示一个状态行,在联机模式下它提供如下的按钮用于操作:

确认: 在报警表中标记的所有报警将被确认.

全部确认: 在报警表中列出的所有报警将被确认.

历史: 按此按钮,在报警表中显示的是发生的所有事件(所有转换)的列表,而不是报警的当前状态.在列表中不能进行报警确认!新事件自动添加.

如果定义了记录文件,在激活'保存'后,在文件中记录发生的所有报警类事件.

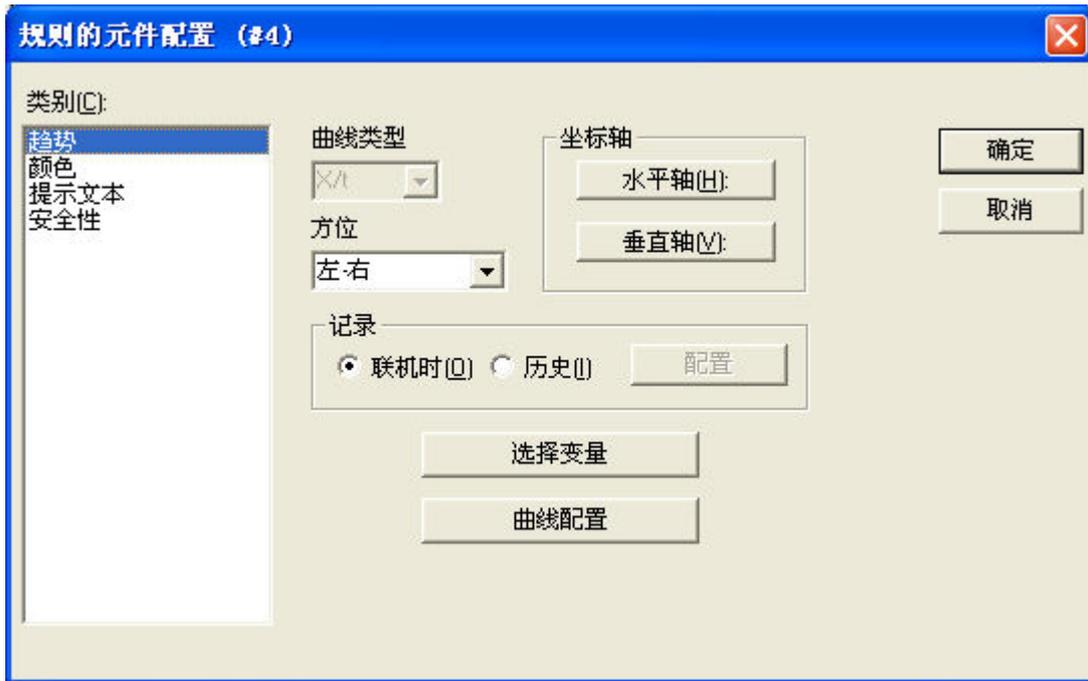
启动: 取消停止 (参看下面)

停止: 在重新按'启动'按钮以前,停止记录发生的事件.

确认变量: 此选项只有在不选择'显示状态行'的情况下才能使用(参看上面描述). 如果选中此项,在状态行中的按钮操作将由变量来控制.要定义变量,首先从列表中选择功能,然后在编辑字段中输入工程变量.因此,例如,在联机模式下,确认所有报警可以通过一个上升沿变量来触发.

## 趋势图

趋势图配置对话框



在联机模式下,趋势图元件用于记录变量值随时间的变化.可以与跟踪功能相比较.在联机下采用图表的方式表示,如果记录到文本文件,每个值独占一行.

**注意:** Depending 依据不同的目标系统,要考虑在 PLC 中保存趋势数据的可能性.

在趋势图配置对话框的'趋势'选项中可作如下配置:

曲线类型: X/t, 水平轴是时间轴,垂直轴是数值

方位: 从左到右或从右到左: 新记录的值是显示在左边还是右边;

轴:

水平轴:

趋势图元件中的水平轴配置对话框



**分割线:** 如果显示拉长的刻度线,激活选项'可见'. 在这种情况下可以定义'刻度线设置': 设置的数值确定在水平轴上分割线的间隔.可以用鼠标点击对应的矩形分别来设置分割线的类型(直线 \_\_\_, 虚线 \_ \_ \_ , 点线 ....., 点划线 \_ . \_ .) 和颜色.

**刻度:** 在'持续时间'中定义刻度的显示范围.如果定义为"T#20s0ms",刻度范围将是 20 秒的周期. 采用相同的语法,可以定义显示的主刻度分割线和子刻度分割线,它们分别用长线段和短线段来标记.

**精度:** 此处定义显示变量当前值的间隔(采用日期的标准格式,如 T#5ms).

**图例设置:** 此处设置图例. 按字体按钮打开标准或特定目标的字体对话框选择字体. 在刻度位置处输入显示刻度的间隔距离(如: T#4ms,那么每隔 4ms 显示一个刻度值).根据激活的选项,刻度中可以包含时间和日期.可以在'格式'字段中定义期望的格式.

**变量:** 这些变量用于水平轴的缩放和偏移.例如偏移量变量的值是 10,那么水平轴显示范围的偏移量设置成 10.

**工具条:** 选中此项将在水平轴下面增加一个工具条,在联机模式下,使用工具条中的按钮可以滚动或缩放趋势图.单箭头按钮将显示范围沿时间轴移动一步,双箭头将显示范围偏移 to 记录的开始或结尾处.缩放按钮实现水平轴刻度的缩放.为了恢复成缩放和偏移量的初始设置,在垂直轴中使用工具条将可得到'恢复' 按钮.

**垂直轴:**

趋势图元件中的垂直轴配置对话框



分割线: 参看水平轴描述.

刻度: 设置趋势图的刻度是显示在趋势图的左侧还是右侧. 设置刻度的'起始值'和'终止值'以及'主刻度'和'子刻度'分割线.

图例设置: 字体和分割线:参看水平轴描述

变量: 参看水平轴描述

工具条: 参看水平轴描述, 附加的'恢复' 按钮用于将缩放和便移量恢复到轴的其始设置.

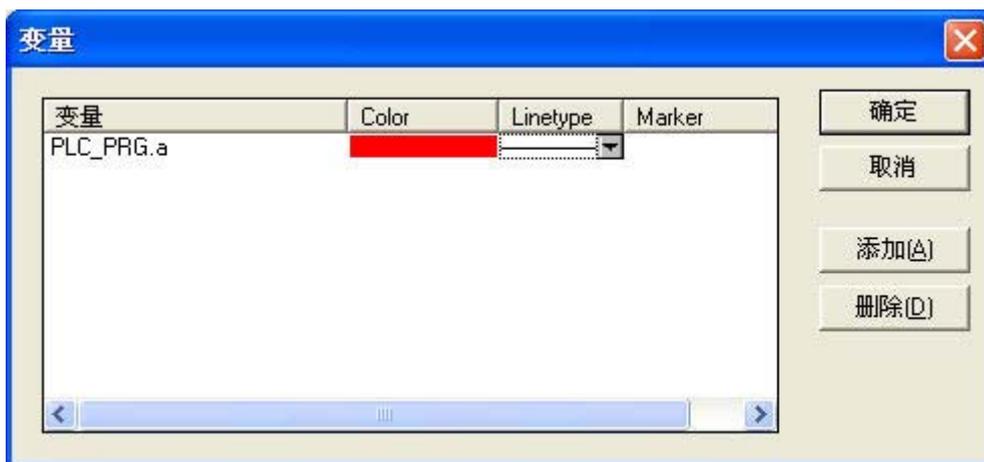
记录: 选择趋势图记录方式:'联机时'或'历史'记录方式.'联机时'就是根据设置的刻度范围按时间将变量值显示在图表上,'历史'就是将记录存储到文件中,按'配置'按钮'打开对话框进行配置,这个对话框和报警日志文件的一样.

注意: 如果在目标系统中激活选项'趋势数据保存到 PLC 中', 将忽略定义的路径,日志文件将存储在 PLC 的下载目录中.

在日志文件中,每次测量结果都单独占一行,每行都包含所有的变量名称和数值.每一行以唯一的标识符按 DWORD 格式开头,它由测量的日期构成.

选择变量: 按此按钮打开变量配置对话框, 在对话框中配置跟踪记录的变量和它们的显示方式:

趋势图元件中的变量配置



在'变量'栏中输入工程变量,可以在输入字段处点击鼠标进行输入.最好使用输入助手 <F2>或智能功能进行输入.

颜色和线型:用于设置变量在记录中的样式.在'颜色'栏中相应的字段处点击鼠标打开标准或特定目标的颜色选择列表,在'线型'栏中的相应字段选择线型(直线 \_\_\_\_, 虚线 \_ \_ \_ , 点线 ....., 点划线 \_ . \_ ).

在'标记'栏中定义一个变量用于在联机模式下使用标记功能显示当前的记录值.标记在图表的左上角显示成一个小灰三角形.如果点中它并保持鼠标按下,可以沿时间轴平移这个标记.作为标记的变量将从记录曲线中读出相应的变量值.

设置所期望的所有记录.使用添加键在列表的末尾处增加一行,用删除键删除当前行.

曲线配置: 按此按钮打开曲线配置对话框设置趋势曲线:

趋势图元件中的曲线配置



曲线类型: 选择直线,台阶和点中的一种. 对于前两项可以使用显示附加点.在显示附加点时可以使用位图,否则附加点显示成一个填充的矩形(颜色与曲线的一致)符号.点击位图旁边的矩形将打开选择位图文件对话框选择文件.使用删除键删除配置的位图.

公差带: 可以定义垂直轴的上下极限值作为公差带来显示.每个公差带可以定义一个颜色(按矩形将打开

标准或特定目标的颜色选择对话框). 如果在联机模式下显示公差带,激活选项公差带作为直线. 当数值超过公差值时期望曲线显示的颜色与公差带的一致,则激活公差带外的曲线颜色与公差带的颜色一致选项.如果想同时使用上述两种方式或都不使用,激活选项两者或无.

举例: 联机模式下的趋势图元件:

在程序 PLC\_PRG 中的声明:

```
VAR
n: INT;
rSinus:REAL;
rValue:REAL;
rSlider1:REAL; (*用于标记功能*)
rSlider2:REAL; (*用于标记功能*)
END_VAR
PLC_PRG 程序:
```

```
n:=n+1;
rValue := rValue + 0.01;
rSinus:=SIN(rValue)*50 + 50;
IF n>100 THEN
n:=0;
END_IF
```

在可视化界面中配置趋势图元件:

方位:右-左; 历史记录方式

水平轴: 分割线: T#2s, 持续时间: T#10s, 主刻度: T#1s, 子刻度: T#500ms, 精度: T#200ms, 图例设置: 时间格式 ('hh':'mm':'ss'), 刻度位置 T#2s. 使用工具条.

垂直轴: 分割线可见, 刻度线设置: 10, 点线, 灰色; 刻度:左, 起始值: 0, 终止值: 100, 主刻度: 10, 子刻度: 5; 图例设置: 10; 使用工具条.

变量:

1. 变量 PLC\_PRG.rsinus, 蓝线, 标记变量: PLC\_PRG\_TRD.rSlider1;
2. 变量 PLC\_PRG.n, 红线, 标记变量: PLC\_PRG\_TRD.rSlider2

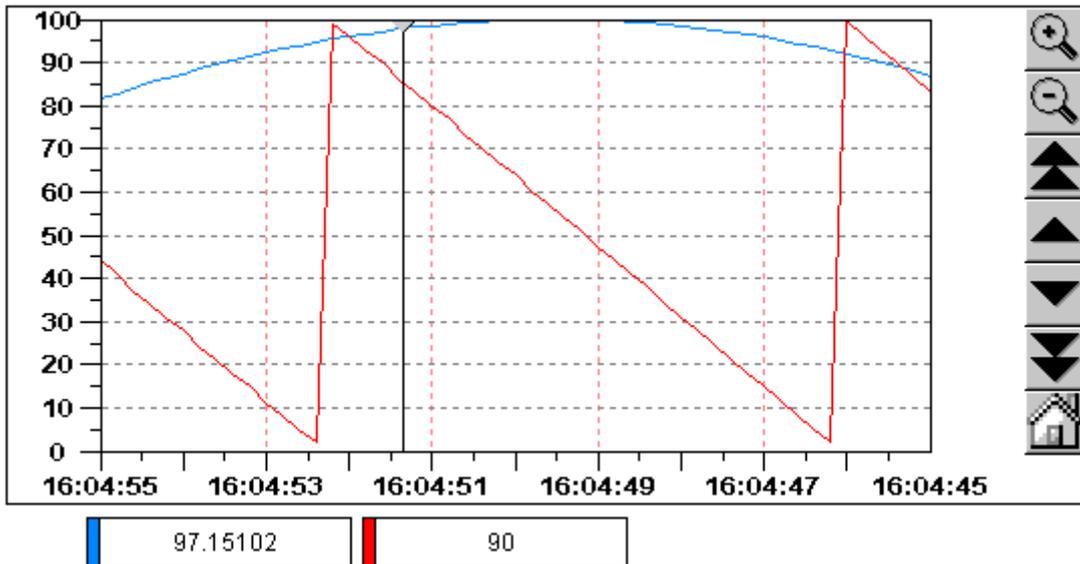
曲线配置: 直线,无公差带

配置由标记变量提供的当前记录值的显示:

矩形元件 1: '文字'选项: 输入 "%s" 到内容字段; '变量'选项: 在文本显示中输入: PLC\_PRG.rSlider1

矩形元件 2: '文字'选项: 输入 "%s" 到内容字段; '变量'选项: 在文本显示中输入: PLC\_PRG.rSlider2  
(在矩形元件 1 和 2 的左边分别放置一个矩形元件,矩形的填充颜色与记录变量的曲线颜色一致.)

程序在联机模式下的运行结果:



记录曲线从左向右移动显示; 最新值显示在最左边; 每隔 200ms 增加一个新值. 如果沿时间轴移动标号(左上角的灰三角), 在矩形元件中显示记录变量在曲线当前位置的数值.

### 位图

在可视化元件配置对话框的‘位图’选项中, 可以作如下的设置:

指定要使用哪个图象文件, 其定义可以是静态或动态的. 如果在‘位图’字段和‘位图变量’字段中都有输入, 那么将忽略‘位图’字段中的输入:

位图:静态定义: 输入本地文件系统的有效图象文件路径. 通过按钮‘!’, 打开浏览文件对话框来选择所需文件.

位图变量:动态定义: 输入 **STRING** 类型的工程变量, 它包含当前要使用的图像文件名. 这可以实现在联机模式下动态改变图像, 但是只能使用工程全局“位图列表”中的“位图文件”(见‘附加’‘位图列表’). 即使在位图列表中指明了全路径, 这个字符串变量也必须指定文件名.

请看下面的例子, ‘stBitmap’为字符串变量, 它从全局位图列表中获得不同的图象文件.

CASE nId OF

0: stBitmap := 'background.bmp';

1: stBitmap := 'deutest.bmp';

2: stBitmap := 'alarm.bmp';

END\_CASE

如果在 PLC\_PRG 对象中声明 stBitmap, 那么, 可以在配置对话框‘位图变量’字段中输入: “PLC\_PRG.stBitmap”。

下列设置项将影响位图的显示。

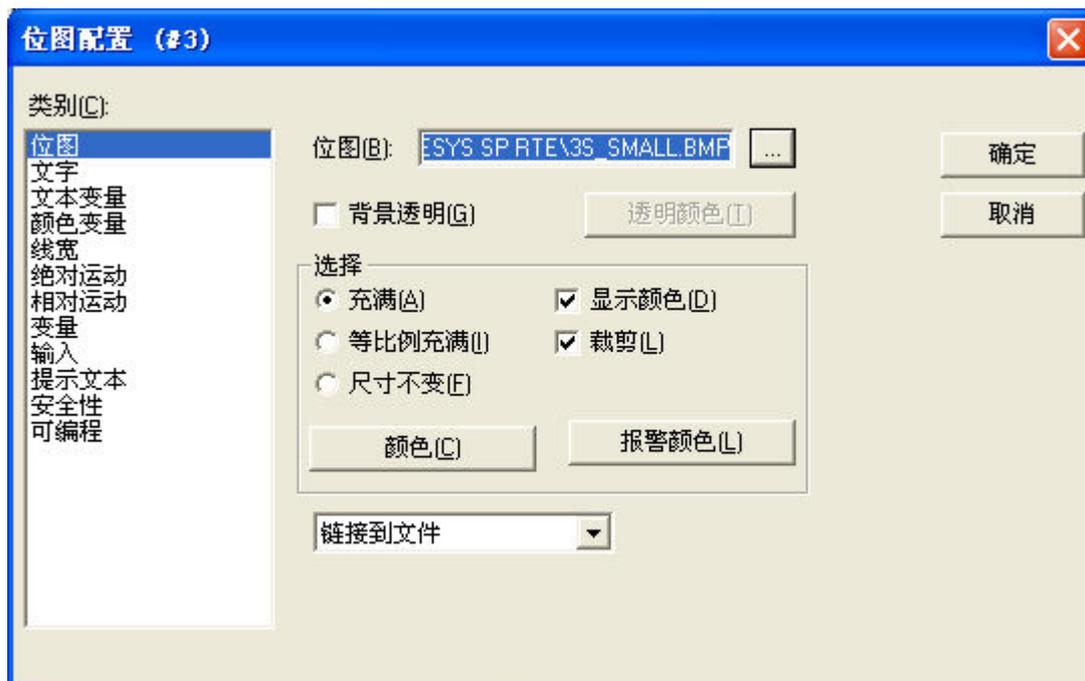
选择“充满”“等比例充满”或“尺寸不变”来指明当边框尺寸发生变化时, 位图的尺寸将如何改变. “充满”就是位图的尺寸与边框的一样大, 位图的高与宽分别调整以便适合于边框. “等比例充满”就是即使位图的整个尺寸发生变化, 但位图依旧保持着高与宽之间的比例. “尺寸不变”就是位图保持其原始尺寸, 不受边框尺寸变化的影响.

如果同时选择了“尺寸不变”和“裁剪”选项, 那么只显示在边框中包含的那部分位图.

如果选择“显示颜色”选项, 那么边框按选择的颜色显示, 颜色是在颜色对话框(标准或特定目标)中的“颜色”和“报警颜色”选项中进行设置. 当“变量”“改变颜色”中的变量为“TRUE”时, 才显示报警颜色.

在对话框下部的选择列表中, 可以定义将位图插入到工程内(嵌入的)还是链接到一个外部位图文件中(其路径在上面的‘位图’字段中输入). 将位图文件保存在工程目录下是合理的, 因为此时可以输入一个相对路径, 否则必须输入一个绝对路径, 特别是当把工程导入到另一个工作环境时会产生一些麻烦.

位图元件配置对话框



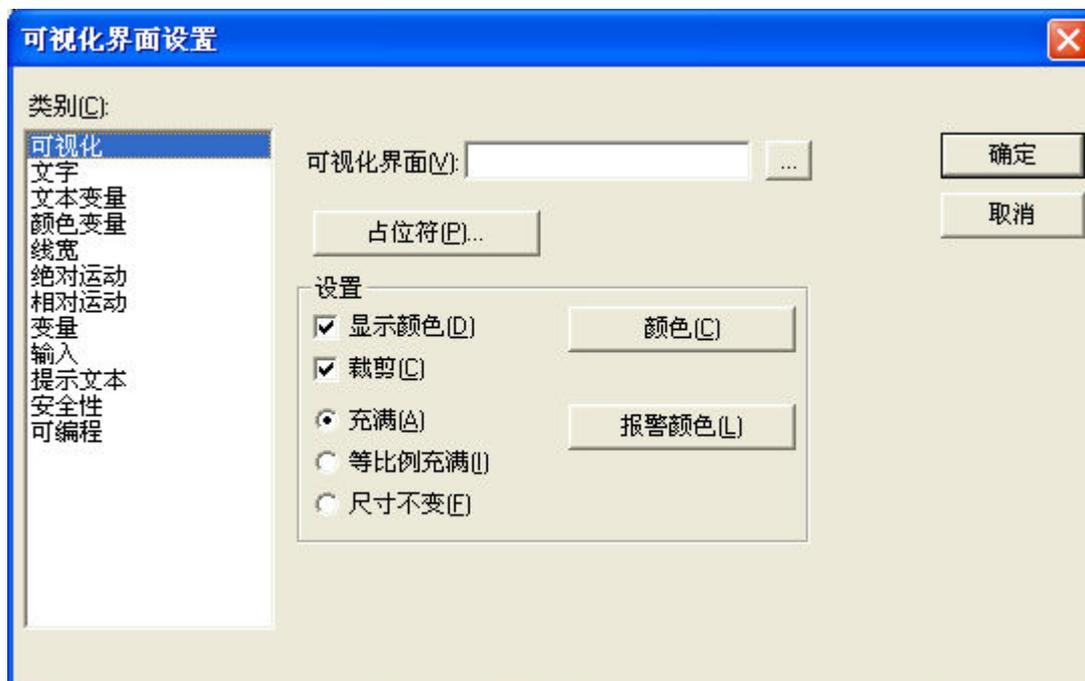
### 可视化界面元件

当将一个可视化界面插入到其他可视化界面中时，意味着正在建立一个可视化的“实例”。

可以在可视化元件配置对话框中的“可视化”选项对这个实例作配置。

在“可视化”字段内输入可视化对象的名字。使用'...'按钮打开一个包含工程中可使用的可视化界面列表对话框。除了本身的可视化界面外，可插入任何一个可视化界面。

可视化界面元件配置对话框



有关可视化界面元件边框的设置，可以参看'位图'中的描述。

点击'占位符'按钮可以打开'占位符列表'对话框，在"占位符"栏中列出了所有可以插入到"主"可视化配置对话框中的占位符。在"替换"栏中可以用一个确定值来替代当前实例的占位符。在给定情况下是否能够替代取

决于在"主"可视化界面中的'附加'占位符列表'对话框中预先定义的数值组。如果是这种情况，就会显示在一个选择组合框。如果未预先定义，则在'替换'栏的相应字段上双击，打开一个编辑区段，填入所需的设置值。

另一种替换实例中的占位符的方法是在'输入'转到'配置对话框中设置可视化界面的调用。

当作用配置对话框'输入'类目中的'Zoom to vis'区段来定义可视化调用时，可以直接替代实力占位符。

一个可视化实例的在线表现：并且它含有的可视化元素作出与原始可视化元素一样的响应。

**注意:**不可能有按时序替代的控制！因此，不应该用含有占位符的正文来替代占位符！

**注意:**当使用占位符时，它已不再有可能在工程编译时立即检查可视化元件配置中的无效输入。因此，相应的错误信息首先在联机模式中显示。(…无效的查看表达式..)

占位符的应用例子:

使用相同的可视化界面可以很容易的显示功能块实例.例如,配置一个可视化界面 visu 用于监控功能块中的变量,每个变量以占位符 \$FUB\$开头 (如 \$FUB\$.a). 如果使用 visu 的实例(在另一个可视化界面中插入 visu 或使用'转到'来调用),那么在这个实例的配置中,占位符\$FUB\$ 必须用功能块的实例名替换,这样才能监控变量.

如下所示:

在工程中定义了包含下列声明:

```
FUNCTION_BLOCK fu
```

```
VAR_INPUT
```

```
  changecol : BOOL; (* 在可视化界面中改变颜色 *)
```

```
END_VAR
```

在 PLC\_PRG 定义了两个'fu'实例:

```
  inst1_fu : fu;
```

```
inst2_fu : fu;
```

创建可视化对象'visu',插入一个元件并打开配置对话框,打开'变量'选项,在字段'改变颜色'中输入"\$FUB\$.changecol".打开'输入'选项,在字段'触发但不保持变量值'中输入"\$FUB\$.changecol".打开'文字'选项,输入"\$FUB\$ - change color "

创建另一个可视化对象'visu1'.在'visu1'中插入两个'visu'(两次参考'visu')

选择'visu'的第一个参考,打开配置对话框中的'可视化'选项,点击'占位符'按钮后将显示占位符列表,用'PLC\_PRG.inst\_1'替换 'FUB'.

选择第二个'visu'实例,按上面的描述,用'PLC\_PRG.inst\_2'替换 'FUB'.

在联机模式下,用于配置'fu'实例的变量值将在相应的'visu'实例中显示结果.

当然,占位符\$FUB\$可以在'visu'配置中的所有可输入变量或文本的位置使用.

**注意:** 联机模式下的可视化界面实例: 如果插入一个可视化元件, 然后选择和配置这个实例, 那么它被视作单个对象, 且在联机模式中根据它的配置, 对输入作出响应, 相反: 如果不配置实例,那么在联机模式下它的显示与原始的一样.

## 组

用于成组的可视化元件对话框提供的设置与位图的'边框'选项中的一样,如充满,等比例充满,尺寸不变,显示颜色,修剪,颜色和报警颜色.

但要考虑在组边框拉伸或缩小时,如何保持组中可视化元件的大小.

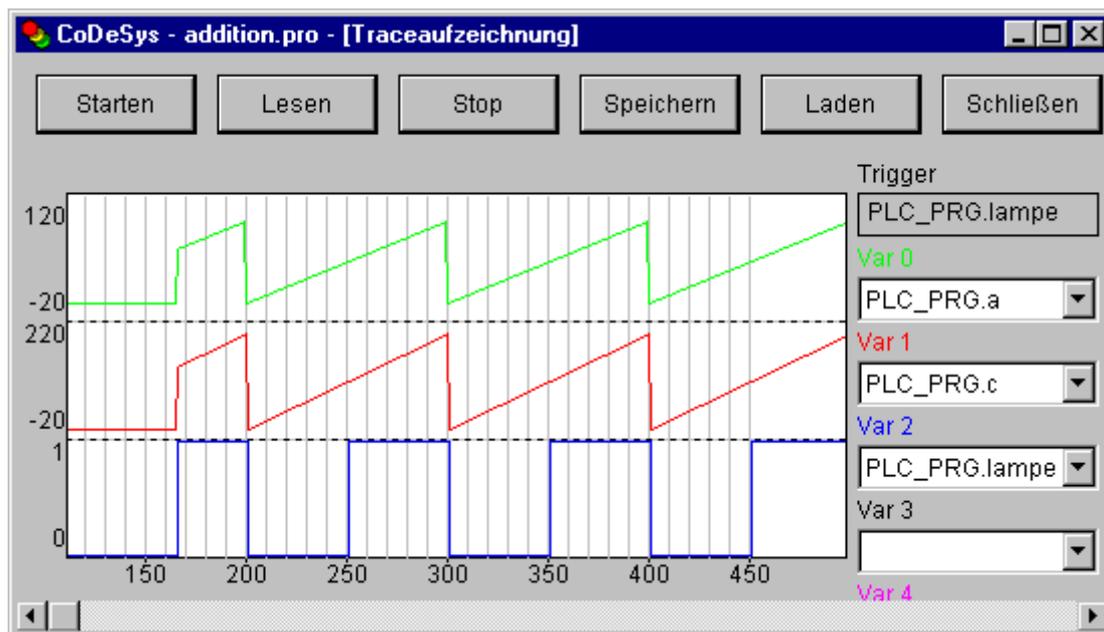
可视化组配置对话框



LANGUAGE <在语言*.vis, *.tlt 或 *.txt 文件中设置的标识符 > 注意: For 对于可视化界面建议使用 *.vis 语言文件!,参看这里	可视化界面设置,语言	在不打开可视化界面设置对话框的情况下,设置期望的语言.
LANGUAGE DEFAULT <在语言文件中设置的标识符>	可视化界面设置,语言	对于动态文本,使用在 xml-文件中定义的语言.
DELAY <延时时间,单位是毫秒>1)2)		在延时结束前下一个命令不执行. 如:在 DEFINERECEIPT 或 READRECEIPT 和 SAVEWATCH 之间需要 500ms 延时.
DEFINERECEIPT <监控列表名>	选择监控列表	命令执行时从配方管理器中选择一个监控列表. 在监控列表中的变量将被监控和显示.
READRECEIPT <监控列表名>	'读取配方'	在定义的监控列表中用当前值替换变量的预先定义值. 注意: 在使用 DEFINERECEIPT 以前必须定义监控列表,并添加 500 ms 延时(参看上面的 DELAY 命令)!
WRITERECEIPT <监控列表名>	'写入配方'	需要输入配方管理器中监控列表的名称.写入监控列表的配方.不需要先执行 DEFINERECEIPT.
SAVEWATCH	'保存监控列表'	保存在文件中的配方将读入到监控列表中. 注意: 为定义当前的配方而调用一个先前的 DEFINERECEIPT,应延时 500ms (参看上面的 DELAY 命令)!
LOADWATCH	'加载监控列表'+ '写入配方'	打开标准的'文件打开'窗口,从选择保存的配方.选择后,这个配方中的数据立刻写入到控制系统中.
CHANGEUSERLEVEL	-	打开设置用户组对话框. CoDeSys 中有 8 个用户级别可供选择.
CHANGEPASSWORD	'工程' '用户组密'	打开修改用户组密码对话框

	码...'	框.
SAVEPROJECT 1)2)	'文件' '保存'	保存工程文件.
EXITPROGRAM 1)2)	'文件' '关闭'	退出程序.
PRINT 1)2)	'文件' '打印'	打印当前可视化界面.
HELP <帮助文件名 >1)2)	帮助文件调用	根据在可视化界面中设置的语言,调用在 CoDeSys ini-文件中设置的帮助文件.(参看 '附加' '设置')
TRACE 1)2)	资源, 采样跟踪	打开采样跟踪窗口. 在全屏显示下,显示命令'启动跟踪','读取跟踪','停止跟踪','保存跟踪值','从文件中加载'.
CNC1)2)	资源, CNC 程序列表	如果工程中有 CNC 列表(资源中),打开 CNC 编辑器. 关闭 CNC 编辑器,将返回到可视化界面.
<p>1) 在目标可视化下不支持 2) 在 Web 可视化下不支持 只在 Web 可视化中使用:</p>		
INTERN LINK <URL>	Web 可视化切换到定义的 URL 浏览器(Unified resource location:在 Internet 的 WWW 服务程序上用于指定信息位置的表示方法); 如 ."INTERN LINK http://www.3s-software.com"	
INTERN LINK <HTTP file path>	打开定义的文件; 如 "INTERN LINK http://localhost:8080/test.pdf"	
INTERN LINK mailto:<EMail-Address>	打开发送邮件对话框,邮件地址要先定义好; 如. "INTERN LINK mailto:s.sdfjksk@companyxy.com"	
INTERN CONNECT_TO <PLC name> <Start-Visu>	<p>改变目标 PLC;先决条件: WebServer 必须配置连接目标系统的参数,要有与 PLC-Handler 相匹配的 ini 文件.</p> <p>PLC 名称: 目标 PLC 名称, 它在 PLC-Handler 的 ini 文件中定义.</p> <p>启动的可视化界面: 首先打开的可视化界面名称.</p> <p>WebServer 将自动实现与各自的 PLC 的连接.</p> <p>如: "INTERN CONNECT_TO PLC1 PLC_VISU"</p>	

操作下的跟踪记录对话框



## 2.5 可视化元件，配置

### 可视化元件，配置...

除了配置各个可视化元件外，也可以配置整个可视化对象。这可能涉及到框架、语言、背景、占位符等以及专门的热键定义分配（键盘使用）。这些对一个可视化对象来说都应该是完全有效的。

这些设置是在可视化编辑器中通过‘附加’菜单来完成。

除了编辑器外，也可以在可视化对象的属性对话框中作某些设置。它涉及到目标可视化，web 可视化的用法及总体布局。

**注意：**在工程选项中，可以用一个独立的目录来保存可视化文件。

### ‘附加’设置

使用“附加”“设置”命令，打开一个对话框，可作出一些影响可视化界面显示和语言以及检查可视化变量的设置。

**注意：**显示、框架和语言也可以在联机模式中编辑。

1. “显示”选项：将一个在 10 到 500%之间的缩放因子输入到“缩放”区段内，以便放大或缩小可视化界面显示的尺寸。

2. “框架”选项：若选择‘自动滚动’，那么在拖拉或移动可视化元件并达到图像边缘时，可视化窗口的可见部分将自动移动。若选择‘在联机模式下适合屏幕’，那么在联机模式下整个可视化界面将全部显示在窗口内，而不考虑窗口的尺寸。当选择‘包含背景位图’，那么背景位图要适合于窗口，否则只考虑元件的显示。

3. “网格”选项：在这里定义网格点在离线模式中是否可见。可视网格点之间的间隔至少为 10，即使输入尺寸小于 10 也取 10。在这种情况下，网格点按输入间隔的倍数显示。若设置为“激活”，那么在元件被拖移和移动时，元件将被放置在捕捉到的网格点上。在“间距”字段中设置网格点的间隔。

4. “编译”选项：在默认情况下，在工程进入联机模式前不检查它们的有效性。如果你想在编译生成工程时检查它们，那么要在编译时激活选项“在编译时检查可视化变量”。在信息窗口中用警告来显示无效的变量。

5. “语言”选项：这里定义用何种语言来显示元件在‘文本’和‘工具提示’选项中的文本。此外，用‘动态文本’选项还可以动态改变显示的文本。

也可参看：

- 可视化界面中的语言转换

- 动态语言转换的配置

**注意:** 仅在“联机模式”中才能改变正文显示!

### ‘附加’选择背景位图

使用这个命令, 打开文件选择对话框, 选择扩展名为‘\*. bmp’的文件, 选择的文件将作为背景显示在可视化界面中.

可以使用命令‘附加’取消背景位图’来删除背景位图.

### 取消背景位图

在背景位图设置好后, 如果背景位图不满足需求, 选择“取消背景位图”可对其进行删除.

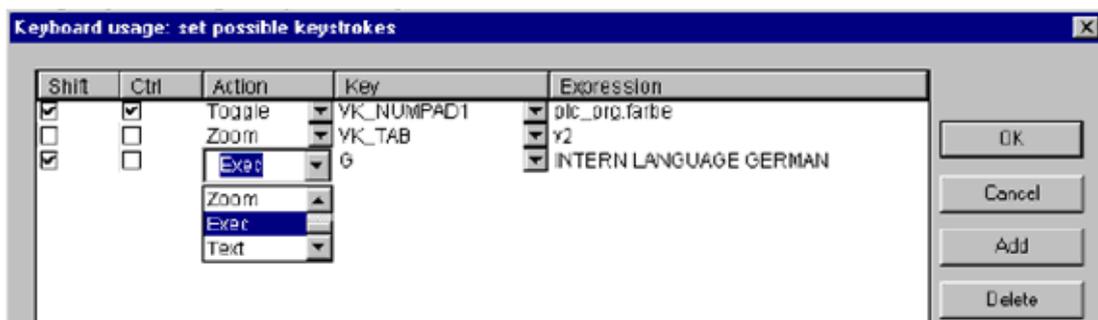
### ‘附加’使用键盘

热键可以优化可视化界面中的纯键盘操作.

在可视化界面的配置中可以定义热键, 它们可以产生与可视化元件一样的动作. 例如, 可以定义在联机模式下当可视化界面‘xy’激活时, 按<ctrl><F2>将停止程序, 此动作也可以通过点击可视化界面‘xy’中的‘z’元件来执行.

总之, 在联机模式下, 可以使用默认键<Tabulator>, <Space>和 <Enter>来选择或激活可视化界面中的每个元件.

执行‘附加’使用键盘’或上下文菜单打开热键设置对话框:



在‘键(Key)’栏中提供了如下可选键列表:

VK_TAB	Tab-键
VK_RETURN	Enter-键
VK_SPACE	Space 键
VK_ESCAPE	Esc 键
VK_INSERT	Insert 键
VK_DELETE	Delete 键
VK_HOME	Home 键
VK_END	End 键
VK_PRIOR	Bild (↑) 键
VK_NEXT	Bild (↓) 键
VK_LEFT	左箭头键 (←)
VK_RIGHT	右箭头键 (→)
VK_UP	上箭头键 (↑)
VK_DOWN	下箭头键 (↓)
VK_F1-VK_F12	功能键 F1 ~ F12
0-9	数字键 0 ~ 9

A-Z	字母键 A ~ Z
VK_NUMPAD0	- 数字操作板上的键 0 ~ 9
VK_NUMPAD9	
VK_MULTIPLY	数字操作板上的键 *
VK_ADD	数字操作板上的键 +
VK_SUBTRACT	数字操作板上的键 -
VK_DIVIDE	数字操作板上的键 /

在栏' Shift' 和' Ctrl', 可以选择<Shift>和/或<Ctrl>键与已选择键组成组合键.

可以参看' 可以用于特定的可视化界面的组合键'.

在栏' 动作(Action)' 中定义当键(组合键)按下时, 产生什么动作. 从列表中选择所要执行的动作和插入适当的表达式. 参看下面的变量动作和有效的表达式, 它们与' 输入' 选项中的设置相对应:

动作	含义	表达式
Toggle	触发变量	变量, 如"plc_prg.tvar"
Tap true	触发变量但不保持(设置为 TRUE)	程序变量, 如 "plc_prg.svar"
Tap false	触发变量但不保持(设置为 FALSE)	程序变量, 如 "plc_prg.xvar"
Zoom	转到.	转到的可视化变量名, 如 "Visul"
Exec	执行程序	执行程序名, 如 "notepad C:\help.txt" (启动 Notepad 并打开文件 help.txt)
Text	变量' 文本显示' 的输入文本	配置用于文本输入的元件号, 如 "#2" (在' 附加' 设置' 中打开显示的元件号; 参看 ' 元件列表...')

在' 表达式(Expression)' 栏中, 根据动作类型必须变量名, INTERN 命令, 可视化界面名称. 这要和' 输入' 选项配置可视化元件严格一致.

使用“添加”键可以在表格的末端添加一个空行. 使用“删除”键可以删除光标所在行.' 确认' 和' 取消' 将保存或不保存设置并关闭对话框.

可以为每一个可视化对象单独设置键盘的使用方法. 所以, 相同的按键(或者组合键)可以在不同的可视化对象中启动不同的动作.

例如:

举例: 下表是两个可视化界面 VIS\_1 和 VIS\_2 的按键设置:

VIS\_1:

Shift	Ctrl	Action	Key	Expression
x		Toggle	A	PLC_PRG.automatic
	x	Zoom	Z	VIS_2

VIS\_2:

Shift	Ctrl	Action	Key	Expression
		Exec	E	INTERN LANGUAGE DEUTSCH
	x	Zoom	Z	PLC_VISU

如果在联机模式下, 且当前可视化界面为 VIS\_1, 按住“Shift+A”复合键, 则 PLC\_PRG 中的 automatic

变量会触发，另外，如果按住“Ctrl+Z”复合键，则会自动跳到可视化界面 VIS\_2。

如果当前可视化界面是 VIS\_2, 按<E>键将把界面中的语言文字转换成德文. 按<Ctrl><Z> 自动跳到可视化界面 PLC\_VISU.

### 用于动态文本的 XML 文件

如何在可视化对象中使用动态文字, 可以参看

- 动态文本转换
- 动态文本语言转换的配置

文件必须使用 XML 文件格式 (即文件名.XML)。在这种文件中, 文本被赋予一个标识符 (这个标识符是一个前缀和一个 ID 号的组合)。为了在联机模式时可以显示各自的文本, 这些组合将输入到一个可视化元素的配置内。即前缀输入到'文本'选项的'内容'字段, ID 输入到'变量'选项的'文本显示'字段。

在文件的开头部分, 定义了默认的语言和字体。可参看本页末尾说明文件中的举例。

在 XML 文件中描述部分必须放在<dynamic-text>和</dynamic-text>之间, 即以<dynamic-text>开头, 以</dynamic-text>结束。

动态文本的语言文件 (从 CoDeSys V2.3.4.0 开始) 可以由 Unicode (UTF-16) 或 ANSI (ISO-8859-1) 代码构成。可以通过位于 xml 文件开头部分的编码进行定义。

#### 请注意:

xml 文件主格式不能用于<dynamic-text></dynamic-text>和文件开头, 但以后会支持的。

目标可视化提供用于扫描动态文本输入的接口。因而可以直接用于程序。

文件结构:

文件的开头部分以<header>开始, 以</header>结束。如果你希望定义一个默认的语言文字, 可以使用输入项<default-language>, 可以使用输入项<default-font>为语言文字定义默认的字体。这些输入项都是可选的。如果某个输入项丢失, 可以根据可视化配置文件进行显示。

<header>	
<default-language><language></default-language>	<p>默认语言文字; 如果没有用于当前语言文字的文本输入项, 那么使用此输入项的默认语言文字。如果默认语言文字也没找到, 将显示"&lt;PREFIX&gt; &lt; ID&gt;".</p> <p>如果使用多个 XML 文件, 从而提供了多个开头部分定义, 那么只使用最后一个。所以建议使用一个开头部分定义! 语言符号必须符合文本输入项中使用的 (参看下面)。</p> <p>注意: 在联机模式下, 默认语言文字可以借助可视化元件设置, 此可视化元件</p>

	使用 '输入', 执行程序配置了命令 INTERN LANGUAGE DEFAULT.
<default-font>	用于<language>的默认字体: 设置的字体(如 "Arial") 将自动用于所有元件, 用<language>定义的语言文字显示动态文本. 语言符号必须符合文本输入项中使用的(参看下面).
<language><language></language>	
<font-name><font></font-name>	
</default-font>	
<default-font>	用于其它语言文字的默认字体.
<language>.....	
....	
</default-font>	
</header>	

PREFIX-ID 组合输入的文本列表必须以<text list>开始, 以</text list>结束. 特殊的文本输入可以单独以<text prefix>开始以<\text>结束.

分配给 PREFIX-ID 组合的文本输入项应包括以下行:

<text prefix>= "<PREFIX> id="<ID>"	"PREFIX" 要与可视化元件配置(选项 文本)中使用的<PREFIX>一致; "ID" 要与选项 '变量', '文本显示' 中的保持一致.
<language> <!CDATA[<TEXT>]] </language>	可以使用任意字符串作为 'language' 的标识符(如 "english"). 这个标识符将显示在 '设置' 对话框中语言选项的语言列表中. 如果要替换 "TEXT", 插入显示的文本以替代在可视化元件中定义的 ID-prefix 组合.
</text>	

对于每一个 Prefix-ID, 必须对应一个语言文字的输入项. 如可以参考以下的举例:

<deutsch>表示德文文字的开始; </deutsch>则表示结束.

动态文本一方面可以用于显示不同的语言文字, 但是在另一方面, 它也用于动态修改文本的内容.

例如: 有两个可视化元件, 一个用于当前机器的识别, 另一个根据当前给出的错误代码, 进行错误信息的显示:

(1) 在 PLC\_PRG 中定义以下的变量: ivar 为 INT 类型, 定义当前机器的识别码, errnum 为 INT 类型, 定义当前的错误号。

(2) 配置一个可视化元件用于显示当前机器的识别码:

- 选项'文本', 字段内容: "%<Maschine>"
- 选项'变量' '文本显示' 中输入: ".PLC\_PRG. ivar"

(3) 配置另一个可视化元件用于显示当产生的错误代码时显示错误信息:

- 选项'文本', 字段内容: "%<Error>"
- 选项'变量' '文本显示' 中输入: "PLC\_PRG. errnum "

(4) 根据以上的说明, 生成一个名称为 dynamictextsample .xml 的文件, 以下是这个文件名的举例。用于动态文本的 XML 文件举例

```
<dynamic-text>
  <header>
    <default-language>deutsch</default-language>
    <default-font>
      <language>deutsch</language>
      <font-name>Arial</font-name>
      <font-color>0,0,0</font-color>
      <font-height>-13</font-height>
      <font-weight>700</font-weight>
      <font-italic>>false</font-italic>
      <font-underline>>false</font-underline>
      <font-strike-out>>false</font-strike-out>
      <font-char-set>0</font-char-set>
    </default-font>
    <default-font>
      <language>english</language>
      <font-name>Arial</font-name>
      <font-color>0,0,0</font-color>
      <font-height>-13</font-height>
      <font-weight>700</font-weight>
      <font-italic>>false</font-italic>
      <font-underline>>false</font-underline>
      <font-strike-out>>false</font-strike-out>
      <font-char-set>0</font-char-set>
    </default-font>
  </header>
  <text-list>
    <text prefix="ERROR" id="4711">
      <deutsch>Fehler an Position 4711</deutsch>
      <english>Error at position 4711</english>
    </text>
    <text prefix="ERROR" id="815">
      <deutsch>Fehler an Position 815</deutsch>
      <english>Error at position 815</english>
    </text>
  </text-list>
</dynamic-text>
```

```

</text>
<text prefix="ERROR" id="2000">
<deutsch>
<![CDATA[
Das ist ein Fehlertext über
                                mehrere Zeilen
]]>
</deutsch>
<english>
<![CDATA[
This is a error text over more than
one line
]]>
</english>
</text>
<text prefix="MASCHINE" id="1">
<deutsch>
<![CDATA[ Vorschub  ]]>
</deutsch>
<english>
<![CDATA[ Feed rate  ]]>
</english>
</text>
<text prefix="MASCHINE" id="2">
<deutsch>
<![CDATA[ Beschleunigung
]]>
</deutsch>
<english>
<![CDATA[ Acceleration
]]>
</english>
</text>
</text-list>
</dynamic-text>

```

(5) 在可视化界面中，打开“设置”对话框中的“语言”选项，激活“动态文本”选项，此时就可以在你的计算机上使用文件列表来添加 dynamictexts.sample.xml 文件。

(6) 联机运行工程

(7) 在可视化界面中，将语言设置为“deutsch”，同时将变量 PLC\_PRG.ivar 设置为“1”，PLC\_PRG.errnum 设置为“4711”。此时可视化元件中将显示以下字符：“Vorschub”、“Fehler an Position 4711”。文本按 Arial 13 号字体进行显示。

(8) 将变量 PLC\_PRG.ivar 设置为“2”；将 PLC\_PRG.errnum 设置为“2000”。随后文本将改变为“Beschleunigung”、“Das ist ein Fehlertext über mehrere Zeilen”。

(9) 在可视化设置中，将语言更改为“English”。现在，将显示出以下的字符串“Acceleration”和

“This is a error text over more than one line”。

### 主界面

在可视化中“主界面”的用处就是提供一个可用于不同可视化界面的对话框，而不需要为每一个可视化界面都插入这个对话框。通过在主界面配置中定义变量来控制在联机模式下对话框何时显示或是否显示。

#### 作为主界面

如果一个可视化界面被定义为“主界面”，这个界面就被自动地插入到其他的所有可视化界面(那些没有被排除在外的)中, 并且在联机模式下，可以使用其所有的功能。这个界面总是处于前面。如果你希望其处于后面，就要激活可视化元素属性对话框中的“背景”选项(参看以下的说明)。主界面不能在所插入的可视化界面中进行编辑。只有在主界面上对主界面进行修改。

通过属性对话框中的‘工程’对象’属性’选项可以定义一个可视化界面作为主界面使用，即激活‘主界面’选项。如果事先已经有另一个可视化界面被定义为主界面，那么那个可视化界面就自动重新定义为“标准”的可视化界面(在属性对话框中的‘可视化界面’选项)。

#### 不带主界面的可视化界面

在属性对话框中，也可以将一个可视化界面定义为一个不带主界面的可视化界面。

### 作为 web 可视化或目标可视化

如果一个工程将用于 web 或目标可视化，那么每一个可视化都可以定义是否作为 web 或目标可视化使用。

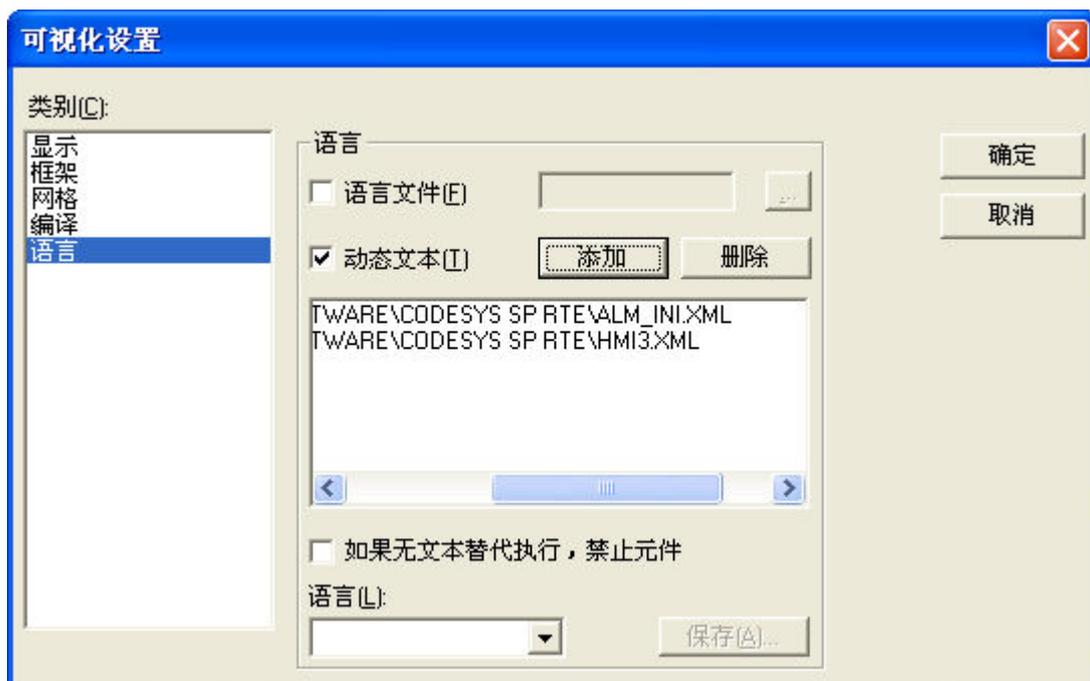
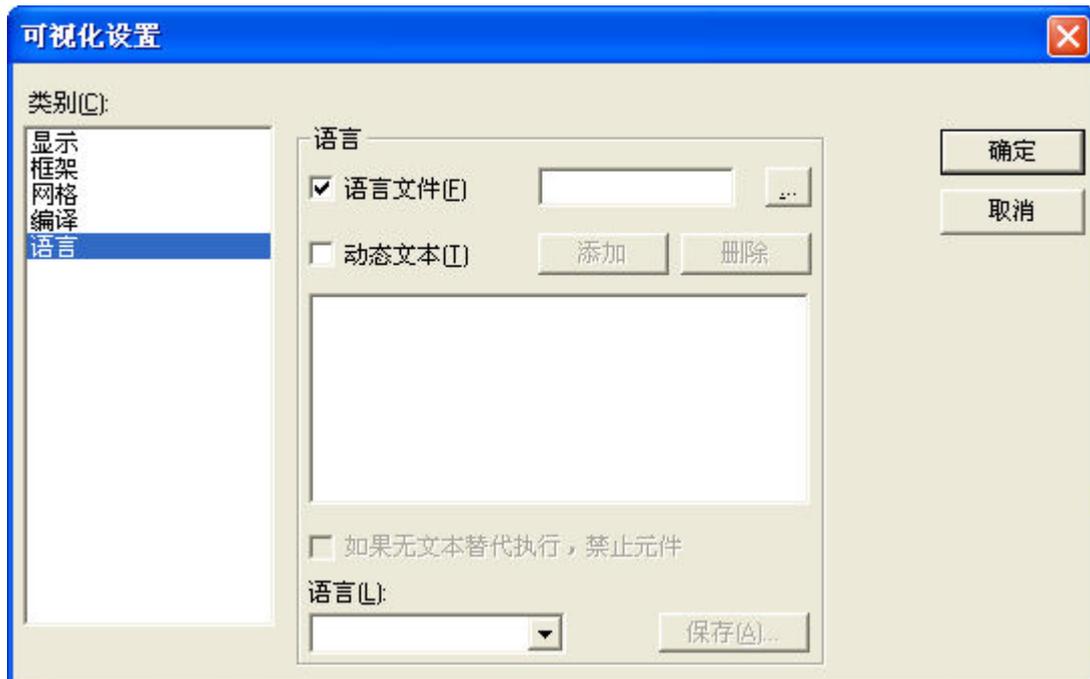
在项目管理中可以选择可视化项目，也可以打开属性对话框(即‘工程’对象’属性’)。如果在目标设置中选中 web 或目标可视化，那么在属性对话框中与 web 可视化或目标可视化相应的选项也被激活。



## 3. 语言转换

### 3.1 语言设置

当开启选择功能时（选中‘语言文件’，即小窗口打√），通过选择右边按钮，可以从程序的安装目录中添加相应的显示语言。然后在下方的下拉菜单中进行选择。其中的动态文本选项，用于在显示语言中插入用户定义的 XML 格式的动态文本。



可以借助提供的静态或者动态文本文件实现可视化界面中文本语言的转换。但 Unicode 格式只能进行动

态转换。

如何进行语言转换：

在可视化对话框中下部的‘语言’选项中选择在语言文件中定义的一种语言作为开始的界面语言。例如为 german（德语）或 english（英语）。

借助可视化元件可以在线模式下进行语言转换。可以使用内部命令“INTERN LANGUAGE <language>” and “INTERN LANGUEDIALOG”进行（可参看“用于操作的特殊输入方法”中的内容）。输入方式可以使用配置对话框中的‘输入’选项。

例如：

你可以在可视化界面中插入一个按钮元件，用于转换到德语。为了实现这个目的，用‘German’标记元件，在配置的‘Input’选项中激活‘执行程序’选项，并定义指令“INTERN LANGUAGE <language>”指令。‘language’将被在语言文件中定义的语言替换。如果在联机模式下按这个按钮，可视化字符串将根据语言文件中应用的德文进行显示。

## 3.2 静态语言转换

静态语言转换需要使用使用一个语言文件（格式为\*.vis, \*.tlt, \*.txt），如何生成一个语言文件可参看以下内容。与动态语言转换不同的是在实时过程中不能使用项目变量进行语言定义。

请注意：对于可视化进程，建议使用\*.vis 格式的语言文件，因为\*.tlt 文件和\*.txt 格式的转换文件只能用于 CoDeSys 平台和 CoDeSys HMI，而不能在‘仪表’，‘条状图’和‘柱状图’元件中使用。

在“可视化设置”对话框中，你可以对项目所使用的语言文件进行配置。为了选择转换文件（\*.tlt 或 \*.txt 文件），或者一个纯的包含了各种远字符的可视化语言文件（\*.vis），需要激活语言文件选项，并且在输入域内输入相应的文件路径。使用按钮就可以得到标准的对话框并打开文件。

用于一个可视化界面语言选择的对话框

有关生成一个\*.tlt 格式的文件，请参看‘工程’‘翻译成其它语言’的内容。

生成一个可视化进程专用的语言文件（\*.vis 格式文件），应执行以下步骤：

为了可视化进程而建立一个新的\*.vis 语言文件，要：

打开可视化设置对话框中的语言选项；

选择语言文件。在相关的输入域内输入你需要存储的文件名，扩展名应为.vis。也可以使用打开文件对话框；

在输入域中填写可视化界面中需要的语言名（例如 german 或者 D），然后按动存储按钮。这样一个以.vis 为后缀的文件就产生了。这个文件可以使用标准的文件编辑器进行编辑。例如你可以使用 Windows 中的记事本工具打开一个语言文件。请看这里的例子。

```

language.vis - Editor
Datei Bearbeiten Suchen ?
[language]
1=german
2=english
[german]
room.2.tip='Mauszeigertext 1'
room.2='Schalter 1'
room.1.tip='Mauszeigertext 2'
room.1='Schalter 2'
[english]
room.2.tip='tooltip 1'
room.2='switch 1'
room.1.tip='tooltip 2'
room.1='switch 2'

```

可以为当前正在使用的可视化界面建立字符变量表。这个字符变量表中包括了表头的参考数字，例如“1=german”可以作为表头[german]的参考数字。可以采用拷贝整行的方法来输入这个变量表，然后再用英文表头取代德文表头，并生成一个新的表头 [english]。于是，除了 1=german 这一行内容外，又添加了 2=english 的新行。为了以一个新的语言中观察可视化界面，需要再次打开语言对话框。现在在选择域中就可以选择德文或者英文了。

### 3.3 动态语言转换

动态语言可以借助可视化元件在不同语言文本版本之间进行转换（包括文字、工具条中的文字和报警表中的警告信息）。与静态转换不同的是可以借助应用程序中的变量进行文本选择。

在可视化元件的配置中，输入一个带前缀的 ID 组合码，这个组合码与 XML 文件中的文本相对应（下面命名为“textlist”）。这个 ID 码可以由工程变量进行定义。

应用实例：ID 表示一个错误号，例如使用“Error”作为前缀。借助前缀 ID 组合码，语言文件提供相应的错误信息，可以依据当前所选择的语言进行显示。

#### 请注意：

- 用于动态字符的语言文件可以有 Unicode (UTF-16) 或者 ANSI (ISO-8859-1) 两种格式。即 `<?xml version='1.0' encoding='UTF-16' ?>`。对于原先选定的用途，请参看这里。

- 对于目标可视化，开始时使用的语言，xml 文件的路径以及 xml 文件表等都可以在目标系统中进行定义。这说明不需要生成一个新的引导工程就可以修改这些参数。所以，可以有一个容易的方法去修改已经存在的工具条，添加新的语言。如果目标系统可以提供这样的配置，在 CoDeSys 的可视化进程中定义的工具条就可以不论是否在线模式都可以使用。如果没有为语言转换专门使用的目标配置，对 CoDeSys 中定义的工具条的修改之后，就必须下载一个项目文件。

### 3.4 对动态语言转换的配置

在联机模式下，用户可以利用前缀 ID 组合码动态控制在可视化元件中显示的文本，因为前缀 ID 组合码指向 XML 文件中定义的文本。

为了实现这个目的，在可视化配置中，应将用于描述文本赋值的 XML 文件与工程连接在一起。这个 XML 文件具有确定的格式。语言代码添加到特定的字符串内，因而随后用户不仅可以在不同的文本内容之间进行转换，而且也支持语言的转换。

在可视化元件的配置中，对于哪些文本显示可以实现动态转换，可以输入前缀和 ID 号（参看以下的内容），ID 号可以由工程变量提供。通过 INTERN 指令 (LANGUAGE DEFAULT) 就可以定义默认的语言。

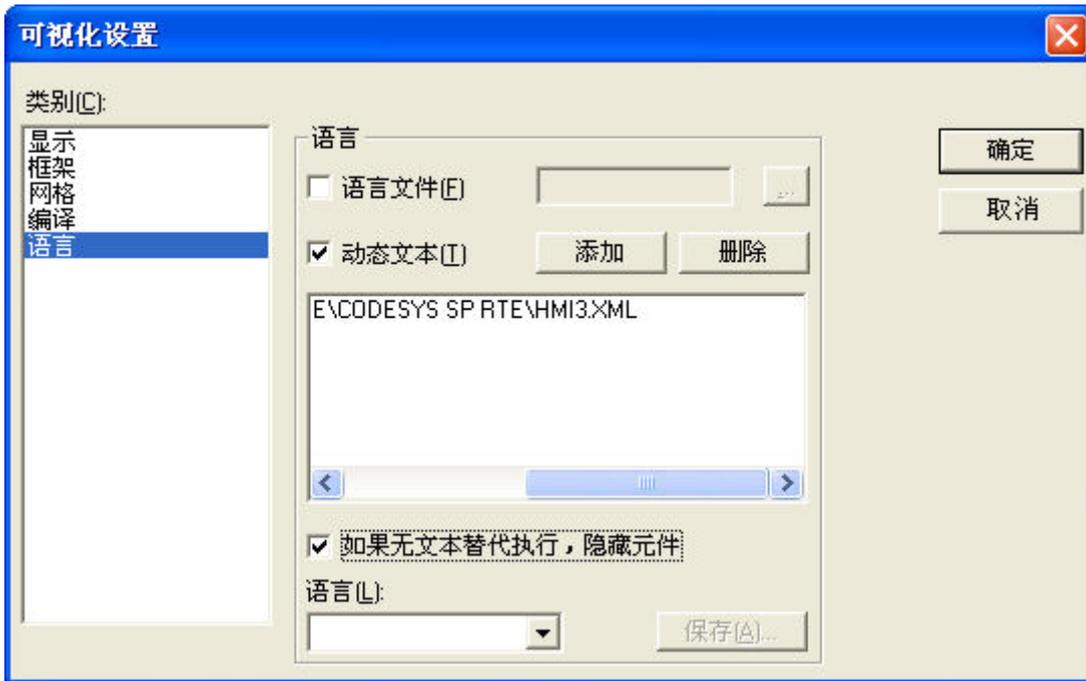
因此，应在一个可视化界面的不同配置对话框内做如下设置：

1. 连接 XML 文件，选择开始时的语言：“设置”对话框中的“语言”选项：

为了将一个或一些可以应用于你的系统的 XML 文件连接到工程上，需要激活“动态文本”选项，并按“添加”按钮。可选文件将列表显示在按钮下方的窗口内。如果要从列表中删除一个选定的文件，可以按“删除”按钮。如果你只希望显示经过动态变换的可视化元件，那么可以激活“若无文本替代执行，隐藏元件”选项。

选择一个在“语言”字段中列出的语言识别符，就可以显示这些文本的版本（即相应的前缀 ID 组合），这个版本是由 XML 文件中的语言标识符标记的。

用于配置动态文本的“设置”对话框，“语言”选项



2. 在选项“变量”的字段“文本显示”中定义 ID 码（在 XML 文件中使用的）：  
输入一个数字，它对应定义了文本 ID 的工程变量（在 XML 文件中使用的）。  
一个报警表中的信息字符，其 ID 码应与表中相应的行数相一致（参看此处的图形）。
3. 在选项“文本”中定义文本格式：

在内容字段处，插入一个占位符“%<PREFIX>”，用于在联机模式下显示动态文本。为了替换这个“PREFIX”，你可以插入任何一个与 XML 文件中 PREFIX 定义相匹配的字符串。请参看有关配置对话框“文本”的说明。

对于每一个可以在 XML 文件找到的前缀 ID 组合码，所赋予的文本都在联机模式下，被显示在可视化元件中。如果没有找到相应的输入项，就不做替换。

### 3.5 用于动态文本的 XML 文件

如何在可视化对象中使用动态文字，可以参看

- 动态文本转换
- 动态文本语言转换的配置

文件必须使用 XML 文件格式（即文件名.XML）。在这种文件中，文本被赋予一个标识符（这个标识符是一个前缀和一个 ID 号的组合）。为了在联机模式时可以显示各自的文本，这些组合将输入到一个可视化元件的配置内。即前缀输入到‘文本’选项的‘内容’字段，ID 输入到‘变量’选项的‘文本显示’字段。

在文件的开头部分，定义了默认的语言和字体。可参看本页末尾说明文件中的举例。

在 XML 文件中描述部分必须放在<dynamic-text>和<\dynamic-text>之间，即以<dynamic-text>开头，以<\dynamic-text>结束。

动态文本的语言文件（从 CoDeSys V2.3.4.0 开始）可以由 Unicode (UTF-16) 或 ANSI (ISO-8859-1) 代码构成。可以通过位于 xml 文件开头部分的编码进行定义。

请注意：

xml 文件主格式不能用于<dynamic-text><\dynamic-text>和文件开头，但以后会支持的。

目标可视化提供用于扫描动态文本输入的接口。因而可以直接用于程序。

文件结构：

文件的开头部分以<header>开始，以<\header>结束。如果你希望定义一个默认的语言文字，可以使用输

入项<default-language>, 可以使用输入项<default-font>为语言文字定义默认的字体。这些输入项都是可选的。如果某个输入项丢失, 可以根据可视化配置文件进行显示。

<header>	
<default-language><language></default-language>	<p>默认语言文字; 如果没有用于当前语言文字的文本输入项, 那么使用此输入项的默认语言文字. 如果默认语言文字也没找到, 将显示"<code>&lt;PREFIX&gt; &lt; ID&gt;</code>".</p> <p>如果使用多个 XML 文件, 从而提供了多个开头部分定义, 那么只使用最后一个. 所以建议使用一个开头部分定义! 语言符号必须符合文本输入项中使用的(参看下面).</p> <p>注意: 在联机模式下, 默认语言文字可以借助可视化元件设置, 此可视化元件使用 '输入', 执行程序配置了命令 <code>INTERN LANGUAGE DEFAULT</code>.</p>
<default-font>	<p>用于&lt;language&gt;的默认字体: 设置的字体(如 "Arial") 将自动用于所有元件, 用&lt;language&gt;定义的语言文字显示动态文本. 语言符号必须符合文本输入项中使用的(参看下面).</p>
<language><language></language>	
<font-name><font></font-name>	
</default-font>	
<default-font>	<p>用于其它语言文字的默认字体.</p>
<language>.....	
....	
</default-font>	

```
</header>
```

PREFIX-ID 组合输入的文本列表必须以<text list>开始, 以</text list>结束. 特殊的文本输入可以单独以<text prefix>开始以</text>结束.

分配给 PREFIX-ID 组合的文本输入项应包括以下行:

<code>&lt;text prefix&gt;= "&lt;PREFIX&gt; id="&lt;ID&gt;"</code>	"PREFIX" 要与可视化元件配置(选项 文本)中使用的<PREFIX>一致;"ID" 要与选项 '变量', '文本显示' 中的保持一致.
<code>&lt;language&gt; &lt;!CDATA[&lt;TEXT&gt;]] &lt;/language&gt;</code>	可以使用任意字符串作为 'language' 的标识符(如 "english"). 这个标识符将显示在 '设置' 对话框中语言选项的语言列表中. 如果要替换 "TEXT", 插入显示的文本以替代在可视化元件中定义的 ID-prefix 组合.
<code>&lt;/text&gt;</code>	

对于每一个 Prefix-ID, 必须对应一个语言文字的输入项. 如可以参考以下的举例:

<deutsch>表示德文文字的开始; </deutsch>则表示结束.

动态文本一方面可以用于显示不同的语言文字, 但是在另一方面, 它也用于动态修改文本的内容.

例如: 有两个可视化元件, 一个用于当前机器的识别, 另一个根据当前给出的错误代码, 进行错误信息的显示:

(1) 在 PLC\_PRG 中定义以下的变量: ivar 为 INT 类型, 定义当前机器的识别码, errnum 为 INT 类型, 定义当前的错误号.

(2) 配置一个可视化元件用于显示当前机器的识别码:

- 选项 '文本', 字段内容: "%<Maschine>"
- 选项 '变量' '文本显示' 中输入: ".PLC\_PRG.ivar"

(3) 配置另一个可视化元件用于显示当产生的错误代码时显示错误信息:

- 选项 '文本', 字段内容: "%<Error>"
- 选项 '变量' '文本显示' 中输入: "PLC\_PRG.errnum "

(4) 根据以上的说明, 生成一个名称为 dynamictexts sample.xml 的文件, 以下是这个文件名的举例. 用于动态文本的 XML 文件举例

```
<dynamic-text>
<header>
<default-language>deutsch</default-language>
<default-font>
<language>deutsch</language>
<font-name>Arial</font-name>
<font-color>0,0,0</font-color>
<font-height>-13</font-height>
<font-weight>700</font-weight>
<font-italic>>false</font-italic>
<font-underline>>false</font-underline>
<font-strike-out>>false</font-strike-out>
<font-char-set>0</font-char-set>
</default-font>
<default-font>
```

```

<language>english</language>
<font-name>Arial</font-name>
<font-color>0, 0, 0</font-color>
<font-height>-13</font-height>
<font-weight>700</font-weight>
<font-italic>>false</font-italic>
<font-underline>>false</font-underline>
<font-strike-out>>false</font-strike-out>
<font-char-set>0</font-char-set>
</default-font>
</header>
<text-list>
<text prefix="ERROR" id="4711">
<deutsch>Fehler an Position 4711</deutsch>
<english>Error at position 4711</english>
</text>
<text prefix="ERROR" id="815">
<deutsch>Fehler an Position 815</deutsch>
<english>Error at position 815</english>
</text>
<text prefix="ERROR" id="2000">
<deutsch>
<![CDATA[
Das ist ein Fehlertext 黚er
                mehrere Zeilen
]]>
</deutsch>
<english>
<![CDATA[
This is a error text over more than
one line
]]>
</english>
</text>
<text prefix="MASCHINE" id="1">
<deutsch>
<![CDATA[ Vorschub  ]]>
</deutsch>
<english>
<![CDATA[ Feed rate  ]]>
</english>
</text>
<text prefix="MASCHINE" id="2">
<deutsch>

```

```
<![CDATA[ Beschleunigung  
]]>  
</deutsch>  
<english>  
<![CDATA[ Acceleration  
]]>  
</english>  
</text>  
</text-list>  
</dynamic-text>
```

(5) 在可视化界面中，打开“设置”对话框中的“语言”选项，激活“动态文本”选项，此时就可以在你的计算机上使用文件列表来添加 dynamictextsample.xml 文件。

(6) 联机运行工程

(7) 在可视化界面中，将语言设置为“deutsch”，同时将变量 PLC\_PRG.ivar 设置为“1”，PLC\_PRG.errnum 设置为“4711”。此时可视化元件中将显示以下字符：“Vorschub”、“Fehler an Position 4711”。文本按 Arial 13 号字体进行显示。

(8) 将变量 PLC\_PRG.ivar 设置为“2”；将 PLC\_PRG.errnum 设置为“2000”。随后文本将改变为“Beschleunigung”、“Das ist ein Fehlertext über mehrere Zeilen”。

(9) 在可视化设置中，将语言更改为“English”。现在，将显示出以下的字符串“Acceleration”和“This is a error text over more than one line”。





## 4. 在线模式的可视化

### 4.1 在线模式的可视化

应注意与在线模式可视化有关的以下内容：

- 评估的顺序：

动态定义的元件属性通过变量将对在配置对话框中由“选项”定义的基本设置(即静态设置)进行改写。

如果一个元件属性由一个“标准的”工程变量进行定义，同时又由一个结构变量组进行定义，那么，工程变量的数值将首先采用。

可视化在线配置的途径可以是单独通过键盘进行输入。对于使用 CoDeSys 的 HMI 作为目标。或者作为 web 可视化的时候，键盘输入是一个特别重要的功能。

- 在下载工程之后，应关注当前目标系统设置中‘禁止当前可视化界面文件下载’选项的当前设置情况。这个设置与当前可视化进程中使用的所有文件都有关系。所有用于目标或 web 可视化的可视化文件都要下载，包括位图文件、语言文件和用于网站可视化的 XML 说明文件。
- “显示，框架和语言”也可以在在线模式下进行编辑。
- 只要一个可视化界面的“实例(引用)”没有进行详细的配置，在在线模式中这个“实例(引用)”的特定元件将会起作用，就象它们在原有的可视化界面中一样。
- 当进行语言转换时（即选用‘附加’设置’时），这些选择只对在线模式的显示有效。
- 可以在在线模式下，进行可视化界面的打印。
- 如果一个可视化界面被作为目标可视化使用，用户通过鼠标点击输入的信息就可以作为专用接口的帮助程序，在工程中使用。

**注意：**用户可以在一些在线使用的可视化元素中找到各种相关说明，可以作为相应元素配置的帮助页面。

### 4.2 在线模式下的键盘操作

为了分别从鼠标或者触摸屏获取信息，单纯使用键盘进行可视化的配置是很有用的：

在线模式下，每次按动以下按键（或者组合键）就可以进行以下的操作：

- 按动<Tabulator>键，可以选择经过配置的输入元件表中的第一个元件。每按动一次这个按键，都可以下移到元件表中的下一个元件上。在元件表中可以跳转到下一个表字段。在按住<Shift>键的同时按动这个按键可以选择前边的元件。根据目标的不同，也可以选择简单的输入处理。
- 箭头键可以用于改变所选定的元件，即根据箭头方向移动到当前选定元件旁边的元件上。
- <Space bar>键可以用于激活所选定的可视化元件。如果这个元件带有输出变量，或者是一个表字段，就可以打开一个文字输入框，在框中显示当前变量的文字内容。按<Enter>键确认写入的数值。

在配置对话框“使用键盘”中可以定义用于联机操作的附加键（即组合键）。除了上边说过的标准按键，也可以将<Tab>、<Space>和<Enter>等按键赋予其他功能。

在线模式中引用的各个元件的动作都可以与被引用的可视化元件相对应。所以这些元件对于鼠标或键盘的输入或操作具有相同的反应。在引用中提示工具条的显示内容依赖于元件。在处理元件表时，例如在使用<tab>键，从一个输入元件移动到另一个输入元件时，在移动到列表中的下一个元件前，参考实例中所有元件的处理从列表中的参考位置继续进行。

**注意：**如果在 web 可视化或者目标可视化中使用 CoDeSys HMI，那么在线模式下使用键盘具有重要的意义。在 web 可视化进程中，如果当前打开了一个输入框，在 web 中进行设置（参看 8.3 节），使用键盘可以非常有效。

### 4.3 在线模式下的‘文件’打印’

在线模式下，'文件' '打印' 指令可以用于打印可视化窗口的内容。但在可视化窗口中有移动元件时，可视化窗口边界的变化可能产生前后不一致的情况。





## 5. 库文件的可视化

### 5.1 库文件的可视化

库文件也可以存储可视化进程，因而也可以使用库 POU 中的工程。库文件可以如同引用那样进行插入，使用“转到”指令进行调用。

**注意：**在一个工程中使用的可视化进程应具有单独的名称。如果出现从库文件中调用的可视化进程在一个工程中具有相同的名称，就可能出问题。因为在处理程序的引入或者调用的可视化进程时，只处理第一个可视化进程。



## 6. 可视化中的系统变量

### 6.1 隐含变量

以下由系统生成的隐含变量可以在程序可视化中使用：

隐含变量名称	数据类型	功能	可用的位置			
			HMI	仿真	目标可视化	Web可视化
CurrentVisu	String[40]	当前打开的可视化界面。如果名称被改变, 对应的另一个可视化界面也将改变 I. 请注意名称字符串必须用大写字母。根据目标系统设置, 这个变量可在目标系统设置中的可视化选项激活或不激活。	x	x	x	x
CurrentCaller	String[40]	先前打开的可视化界面的名称。被用于 ZOOMTOCALLER 功能。只能在目标可视化中设置和修改。	-	-	x	-

CurrentLanguage	String[40]	当前设置的可在语言文件中获得的语言. 语言必须用大写字母. 只能在目标可视化中设置和修改.	-	-	x	-
CurrentUserLevel *	INT	当前设置的用户级别 0..7.	x	x	x	x
CurrentPasswords[0 .. 7] *	ARRAY [0..7] of String[20]	在用户组密码中定义的所有密码	x	x	x	x

作为保持变量使用的隐含变量:

一个目标可视化进程中的隐含变量可以声明为保持变量:

为了这个目的, 变量应先声明为全局变量。此项声明必须资源选项中最上面的'全局变量'文件夹中进行(这个变量表是字母排序的)。如果声明在其他的全局变量表中进行, 就会出现编译错误。

例如:

```
VAR_GLOBAL RETAIN
```

```
VisuDoExecuteUserlevelInit : BOOL := TRUE;
```

```
CurrentUserLevel : INT := 0;
```

```
CurrentPasswords : ARRAY[0..7] OF STRING[20] := 'a','b','c','d','e','f','g','h';
```

```
END_VAR
```

\* 请注意变量 CurrentUserLevel 和 CurrentPasswords。这些变量必须具有相同的类型(如 normal、RETAIN、PERSISTENT 等)! 如果他们被定义为剩余变量, 另一个类型为 BOOL 的变量 VisuDoExecuteUserlevelInit (其初始值应为 TRUE) 就必须在全局变量表中公告为 RETAIN 型的变量。





## 7. 可以用于特定的可视化进程的组合键

### 7.1 可以用于特定的可视化界面的组合键

下边的表格中说明了所有可以用于特定的可视化进程中的组合键  
在表格中使用以下的缩写：

- C 用于 CoDeSys 和 CoDeSysHMI
- TV 用于目标可视化
- WV 用于网站可视化

如果在某一列中使用了一个缩写，表示所涉及的可视化进程支持相应的组合键。  
每一个特定行中内容的说明都可以在下面的表格中找到。

	no modifier	Shift	Ctrl	Shift+Ctrl	comment
VK_TAB	C	C	C	C	K4
VK_RETURN	C / TV	C / TV	C / TV	C / TV	
VK_SPACE	C / WV	C / WV	C / WV	C / WV	K4;K5
VK_ESCAPE	C / TV / WV	C / TV / WV			K3
VK_INSERT	C / TV	C / TV	C / TV	C / TV	
VK_DELETE	C / TV	C / TV	C / TV	C / TV	
VK_HOME	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_END	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_PRIOR	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_NEXT	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_LEFT	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_RIGHT	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_UP	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
VK_DOWN	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
F1	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	K1
F2	C / TV / WV	C / TV / WV	C / TV / WV	C / TV / WV	
F3	C / TV / WV	C / TV / WV	C / TV /	C / TV / WV	

			WV		
F4	C / TV / WV				
F5	C / TV / WV				
F6	C / TV / WV				
F7	C / TV / WV				
F8	C / TV / WV				
F9	C / TV / WV				
F10	C / TV	C / TV	C / TV	C / TV	K2
F11	C / TV / WV				
F12	C / WV	C / WV	C / WV	C / WV	
0	C / TV / WV				
1	C / TV / WV				
2	C / TV / WV				
3	C / TV / WV				
4	C / TV / WV				
5	C / TV / WV				
6	C / TV / WV				
7	C / TV / WV				
8	C / TV / WV				
9	C / TV / WV				
A	C / TV / WV				
B	C / TV / WV				
C	C / TV / WV				

			WV		
D	C / TV / WV				
E	C / TV / WV				
F	C / TV / WV				
G	C / TV / WV				
H	C / TV / WV				
I	C / TV / WV				
J	C / TV / WV				
K	C / TV / WV				
L	C / TV / WV				
M	C / TV / WV				
N	C / TV / WV				
O	C / TV / WV				
P	C / TV / WV				
Q	C / TV / WV				
R	C / TV / WV				
S	C / TV / WV				
T	C / TV / WV				
U	C / TV / WV				
V	C / TV / WV				

W	C / TV / WV				
X	C / TV / WV				
Y	C / TV / WV				
Z	C / TV / WV				
VK_NUMPAD0	C / TV / WV	C / TV / WV			
VK_NUMPAD1	C / TV / WV	C / TV / WV			

- K1 在 CoDeSys 软件中，可以附加显示在线帮助
- K2 在网站可视化进程和浏览器中，可以使用文件菜单。
- K3 Ctrl/Esc 组合键可以打开开始菜单，Shift/Ctrl/Esc 组合键可以打开“任务管理”菜单
- K4 Tab 键和 Space 键在目标可视化中具有不同的功能
- K5 Shift/Space 可以打开相应的配置功能





## 索引:

- A**
- ActiveX 控件 ..... 4-23
  - 安全属性 ..... 4-14
- B**
- 报警表 ..... 4-32
  - 变量 ..... 4-11
  - 表格 ..... 4-19
- CH**
- 插入 ActiveX 元素 ..... 1-4
  - 插入按钮 ..... 1-3
  - 插入棒图 ..... 1-4
  - 插入报警列表 ..... 1-4
  - 插入表格 ..... 1-4
  - 插入饼图 ..... 1-3
  - 插入长方形 ..... 1-2
  - 插入多边形 ..... 1-3
  - 插入多线段 ..... 1-3
  - 插入可视化元件 ..... 1-2
  - 插入曲线 ..... 1-3
  - 插入趋势图 ..... 1-4
  - 插入图元文件文件 ..... 1-3
  - 插入椭圆形 ..... 1-2
  - 插入位图文件 ..... 1-3
  - 插入仪表 ..... 1-4
  - 插入已创建的其他文件 ..... 1-3
  - 插入圆角长方形 ..... 1-2
  - 插入柱状图 ..... 1-4
  - 创建一个新的可视化 ..... 1-2
- D**
- 动态语言转换 ..... 6-3
  - 对动态语言转换的配置 ..... 6-3
- 符号**
- '附加' '配置' ..... 4-3
  - '附加' '设置' ..... 5-1
  - '附加' '使用键盘' ..... 5-1
  - '附加' '对齐' ..... 2-2
  - '附加' '全选' ..... 2-1
  - '附加' '选择' ..... 2-1
  - '附加' '选择背景位图' ..... 5-1
  - '附加' '元件列表' ..... 2-2
  - '附加' '占位符列表' ..... 4-2
  - '附加' '置于后面' ..... 2-2
  - '附加' '置于前面' ..... 2-2
- F**
- 复制可视化元件 ..... 2-1
- G**
- 改变选择和插入模式 ..... 2-1
  - 工具提示文本 ..... 4-14
- J**
- 角度 ..... 4-4
  - 静态语言转换 ..... 6-2
  - 绝对运动 ..... 4-10
- K**
- 可编程性 ..... 4-15
  - 可视化编辑器概述 ..... 1-1
  - 可视化界面元件 ..... 4-41
  - 可视化元件, 配置 ..... 5-1
  - 可视化状态条 ..... 2-3
  - 可以用于特定的可视化界面的组合键 ..... 10-1
  - 库文件的可视化 ..... 8-1
- P**
- 配置可视化元件 ..... 4-1
- Q**
- 趋势图 ..... 4-34
  - 取消背景位图 ..... 5-1
- SH**
- 输入 ..... 4-12
- T**
- 条状图 ..... 4-28
  - 拖动可视化元件 ..... 2-1

<i>W</i>		用于操作的可能的特殊输入 .....	4-43
位图 .....	4-40	用于动态文本的 XML 文件 .....	5-3, 6-4
文本变量 .....	4-6	语言设置 .....	6-1
文字 .....	4-5		
<i>X</i>		<i>Z</i>	
线宽 .....	4-7	在可视化中的占位符 .....	4-1
相对运动 .....	4-11	在线模式的可视化 .....	7-1
形状 .....	4-4	在线模式下的键盘操作 .....	7-1
修改可视化元件 .....	2-1	在线模式下的'文件' '打印' .....	7-1
选择可视对象 .....	2-1		
<i>Y</i>		<i>ZH</i>	
颜色 .....	4-8	主界面 .....	5-7
颜色变量 .....	4-9	柱状图 .....	4-30
仪表 .....	4-26		
隐含变量 .....	9-1	<i>Z</i>	
		组 4-42	
		组合可视化元件 .....	2-2
		作为 web 可视化或目标可视化 .....	5-7