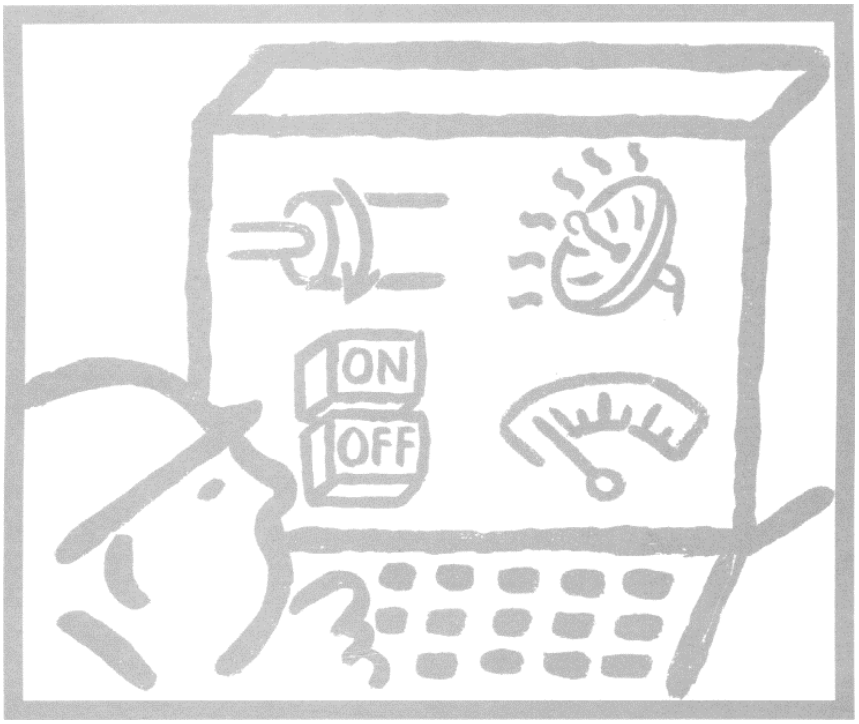
ASIC-200 Version 5.0

integrated industrial control software



HMI Guide

Pro-face
XYCOM

Revision	Description	Date
C	Name change, correct where applicable with document	4/07

HMI Guide: 139168(C)

Published by: Pro-face
750 North Maple Road
Saline, MI 48176

ASCI-200 release 5.0 documents include:

Getting Started	137586
User Guide	139837
Language Reference	139183
HMI Guide	139168

Note: the current revisions of each of these documents should be used.

Note: Features available on your system depend on product version and installed options (toolkits).

Contents

Introduction

HMI Introduction

The Operator Interface has two modes of operation: the edit mode lets you create operator interfaces for an application and the activate mode lets operators use the interface screens to control and monitor application programs. A project can have one or more operator interface files, and each operator interface file can have one or more operator interface screens. Only one operator interface file can be active at a time. One operator interface screen is defined as the start screen (the first screen displayed) for the operator interface file.

Starting the Operator Interface

To start the Operator Interface

Do one of the following:

- Locate the *ASAP Applications* menu from the Windows *Start* menu and choose *Operator Interface*.
- If the Program Editor is open, choose *Operator Interface* from the Program Editor *Tools* menu.

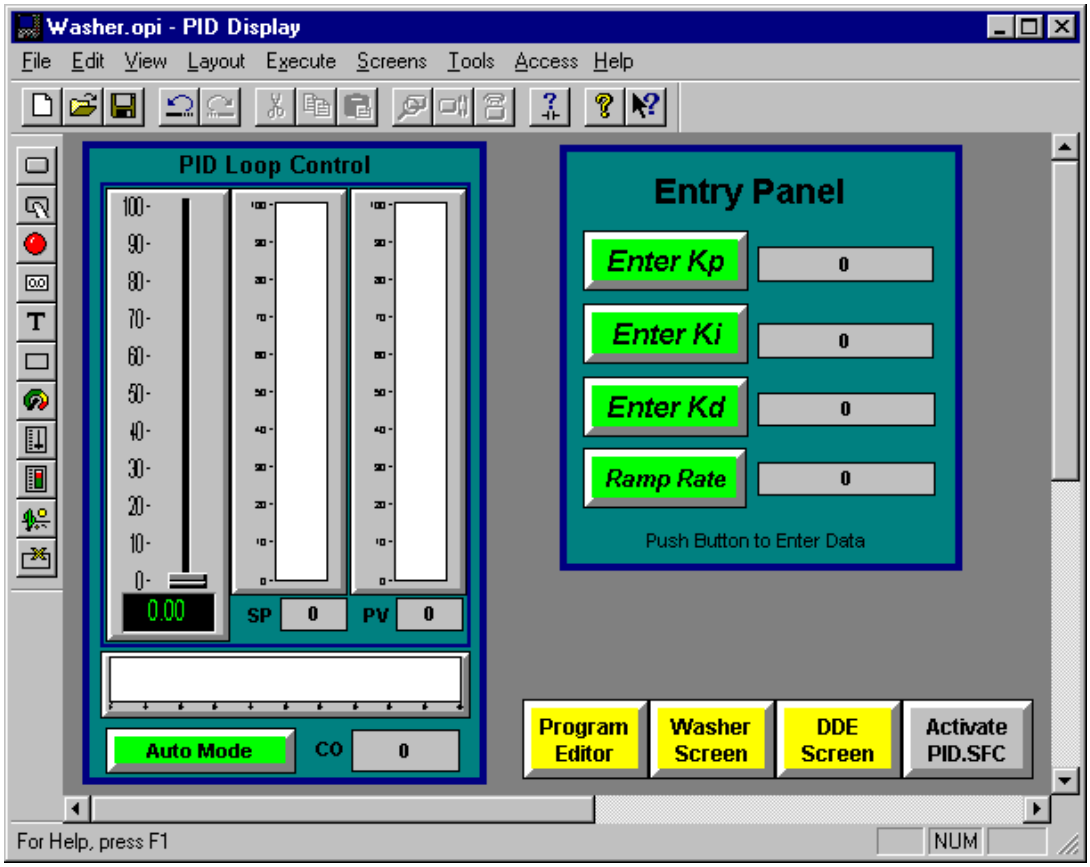
Notes:

1. The Operator Interface starts in activation mode. The last operator interface file opened for the current project is used to define the operator interface. The start screen defined in the operator interface file is the first operator interface screen displayed.
2. If the runtime subsystems are not running when the Operator Interface is started, a prompt appears providing an opportunity to start the runtime subsystems.
3. If any controls were to fall outside the screen boundaries (due to the operator interface screen being larger than the currently available screen area), a prompt appears asking if you want adjust the controls outside of the screen boundaries. This can occur, for example, if you designed at a higher Windows display resolution than the current

display settings (e.g., designed at 800*600 and switched to 640*480), or designed with *Design for MenuBar space* off, and then started the operator interface with *Design for MenuBar space* on. If you do not want the controls adjusted, you may not be able to access these controls.

Editor Window

The following figure shows an example of an operator interface screen. The title bar shows the operator interface file and screen name of the current screen. Example screens (OPI files) can be found in the *Samples* folder.



Access Levels

Access levels and access codes are used to control access to application programs, operator interface screens, and configuration data. There are five access levels, with a different access code for each. The following table describes the privileges of each level. The lowest access level is 0; each successively higher access level has the privileges of the preceding levels.

Access Level	Default Code	Privileges
0	<Cancel>	Use operator interface screens, including selecting and running programs and using jogging and homing axes.
1	1234	Open, view, and monitor programs and symbols from the Program Editor. Perform licensing functions.
2	2345	Same as Access Level 1. (Can be used as an additional access level within the HMI for button operation.)
3	3456	Perform project functions. Edit system configurations. Edit programs. Edit operator interface screens.
4	4567	Change passwords for access levels 1 to 4 (from the Operator Interface).

Note: If you have the Program Editor open, changing the access level within the Operator Interface does not change the access level of the Program Editor.

Entering an Access Code

To enter an access code

1. Choose *Enter Password* from the *Access* menu. A keypad appears.
2. Enter the numbers on the keypad that represent your access code and click *OK*.

To set the access level to 0

- Click *Cancel* on the access keypad.

Assigning a Code to an Access Level

Access codes can be assigned to an access level from the Operator Interface provided that the current access level is level 4. There is one access code for each level.

To assign an access level code

1. Choose *Enter Password* from the *Access* menu. A keypad appears.
2. Click * (the asterisk key) four times. The message *Enter Access Level to Change.* appears.
3. Enter the number of the access level you want to change and click *OK*.
4. Enter the new four digit number. The message: *Enter new password again.* appears.
5. Enter the new four digit password again and click *OK*. If you verify the new password correctly, the password is changed.

Notes:

1. Click *Cancel* on the keypad to set the access level to level 0.
2. When attaching control functions to operator controls (in the operator interface edit mode) an access level can be specified to control the use of the operator control.

Editing Screens

The Operator Interface has two operating modes: activation mode and edit mode. In activation mode, the operator interface runs and controls the application functions. You cannot edit controls when the screens are activated. The Operator Interface starts in activation mode. In the edit mode, screens can be created and edited.

To switch to edit mode from activate mode

You need an access level of level 3 or level 4 to enter the edit mode.

1. Choose *Operator Interface Screen Editor* from the *Tools* menu.
2. Enter the password for the level 3 or 4 access.

Activating Screens

The Operator Interface has two operating modes: activation mode and edit mode. In activation mode, the operator interface runs and controls the application functions. You cannot edit controls when the screens are activated. The Operator Interface starts in activation mode. In the edit mode, screens can be created and edited.

To switch to activate mode from edit mode

- Select *Activate Screens* from the *Execute* menu.

  Notes:

  When the screens are activated:

  1. You can see how the screen will look and interact with the controls.
  2. At the lowest access levels (0 and 1):
     - The menu bar is hidden.
     - The frame around the main window cannot be moved.
     - The following key combinations are intercepted and disabled:

       ALT+ESC
       ALT+TAB
       ALT+F4
       Applications key (Microsoft natural Keyboard, etc.)

     If the HMI is set to be the shell (refer to Securing Access Outside the HMI), this effectively blocks operator access to the operating system (and any programs). The key combination CTRL+ESC and the "Windows" keys (Microsoft natural Keyboard, etc.) are not intercepted, but if the HMI is the shell, these keys are not a problem.

     The key combination CTRL+ALT+DEL continues to work.
  3. When the menu bar is absent, the following key combinations can be used:

     ALT+A displays the Access Level Keypad.

     ALT+H displays help contents.

Creating Screens

Operator Interface Operations

Starting a New Operator Interface File

An operator interface file contains one or more screens. A project can contain more than one operator interface file. The start screen of the last opened operator interface file is the active screen when the Operator Interface is started.

To start a new operator interface file

- Select *New* from the *File* menu or use the tool bar button.

The default file name of an operator interface file is *OPI* and the default screen name is *Operator Interface*.

Opening an Operator Interface File

To open an operator interface file

- Select *Open* from the *File* menu or use the tool bar button.

Saving an Operator Interface File

Saving an operator interface file saves all the screens in the file.

To save the file

- Select *Save* from the *File* menu or use the tool bar button.

When you save a file for the first time, the *Save As* dialog box appears so you can name the file.

To save the file with a new name or under a different revision

- Choose *Save As* from the *File* menu. In the dialog box that appears, provide a new path or name for the file, or specify the revision.

Note: When you save a file to a revision less than the current revision, any features in the current version that are not in the previous version you are saving to are permanently removed. Even if you open the file in a version you originally created it in, these features will not be present.

Screen Operations

Creating a New Operator Interface Screen

To create a new operator interface screen

1. Select *New Screen* from the *Edit* menu. The *Enter New Screen Name* dialog box appears.
2. Type in the name for the new screen and click *OK*. A blank operator screen is created with the screen name in the title bar.

Deleting a Screen

The operator interface file must include at least one screen. If you try to delete the last screen of the operator interface, an error message appears.

To delete an operator interface screen

1. Select *Delete Screen* from the *Edit* menu. The *Select Screen to Delete* dialog box appears.
2. Type in the name of the screen you want to delete or select the screen name from the from the drop down list box and click *OK*.

If the screen you deleted was the startup screen, the *Select New Startup Screen* dialog box is displayed. Type in the name of the new startup screen or select the screen name from the drop down list box.

Copying a Screen

To copy an operator interface screen

1. Select *Copy Screen* from the *Edit* menu. The *Select Screen to Copy* dialog box appears.
2. Select the screen from the dialog box and click *OK*. The *Enter New Screen Name* dialog box appears.
3. Type a name to give to the new screen and click *OK*. The new screen is created with the same contents as the original screen.

Renaming a Screen

To rename an operator interface screen

1. Select *Rename Screen* from the *Edit* menu. The *Select Screen to Rename* dialog box appears.
2. Select a screen to rename and click *OK*. The *Enter New Screen Name* dialog box appears.
3. Type a name to give the new screen and click *OK*.

Selecting the Startup Screen

The startup screen is the screen that appears (for the current project's active configuration) when the Operator Interface is started.

To select the startup screen

1. Select *Start Screen* from the *Edit* menu. The *Select New Startup Screen* dialog box appears.
2. Select a startup screen and click *OK*. The selected screen is the new startup screen.

Selecting a Screen to Edit

Before you can edit a screen, you must select it.

To select an operator screen to edit

1. Click on the *Screens* menu item. A list of all operator screens in the current operator interface file appears in the menu.
2. Select the desired screen. The screen is displayed for editing.

Editing Operations

Notes

1. The menu bar is hidden at Access Levels 0 and 1. If you will be using the HMI at these levels and wish to take advantage of this extra screen space, uncheck *Design for MenuBar space* on the *Layout* menu.
2. If the Windows Explorer shell is present (normal Windows operation), the Taskbar can be set to *Auto hide*. Screen design in the operator interface does not take this into account. Screen designers should be consistent in that the Taskbar *Auto hide* feature should be set the same way in edit mode and activation mode.

Adding Controls

To add a control

1. Select *New Standard Control* from the *Edit* menu. A list a standard controls appears.
2. Select the control to add. The cursor shape changes to reflect the control.
3. Position the cursor in the screen and click to add the control.

   Notes:

   1. Most controls can also be selected from the Control Tools tool bar.
   2. To cancel the operation, press Esc.

3. For information on adding ActiveX controls, refer to Inserting ActiveX Controls.

Editing Controls

To edit a standard control

1. Do one of the following
   - Double-click on the control.
   - Select the desired control and press Enter.
   - Select the desired control and select *Edit Standard Control* from the *Edit* or context menu.
2. An appropriate edit dialog box for the control appears. For information on standard controls, refer to the **Standard Controls** section.

   Note: For information on editing ActiveX controls, refer to **Editing ActiveX Controls** in the **ActiveX Controls** section.

Selecting Controls

To select a single control

- Position the mouse cursor over the control and click. The control is highlighted.

To select multiple controls using a selection box

A rectangular *rubber-band* will be drawn to surround the controls to be selected.

1. Position the selection arrow at one corner of the rectangle. Click and drag to the opposite corner.
2. Release the mouse button. All controls entirely within the rectangular area are selected.

To select multiple controls using the keyboard and mouse

- Hold the Ctrl or Shift key as you select individual controls.

To deselect a control from a selected group of controls

- Hold the Ctrl or Shift key as you move the mouse cursor over the control and click.

Moving Controls

To move controls using the mouse

1. Select the controls.

2. Move the cursor over one of the selected controls. Click and drag the selected control to the desired location. A black border shows the new bounding area of the control group.
3. Place the controls at the new location by releasing the mouse button.

   To cancel the operation, press Esc.

To move controls using the keyboard

1. Select the controls.
2. Use the arrow keys to move the control group one pixel in the direction of the arrow key.

Sizing Controls

To size a control using the mouse

1. Select the controls. If the control can be sized, size handles appear on each corner and side of the control.
2. Move the cursor over one of the size handles. The cursor changes to a sizing arrow, indicating the direction in which the control can be sized.
3. Click and drag the selected sizing handle to stretch or shrink the control. A black border shows the new bounding area of the control group.
4. Once the control is sized, release the mouse button.

   To cancel the operation, press Esc.

Copying, Cutting, and Pasting

Copy, cut, and paste operations are often used together. Copy saves selections to the clipboard. Cut deletes the selection and saves it to the clipboard. Paste inserts the clipboard contents into the current screen. You can cut, copy, and paste between screens.

> **Note:** Although the copy, cut, and paste operations work in a similar manner as in other Windows applications, the controls clipboard format is not compatible with other applications. Controls can only be copied and pasted within or between screens.

To copy a control

- Select the control, then choose *Copy* from the *Edit* or context menu or use the keyboard Ctrl+C keys.

To cut a control

- Select the control, then choose *Cut* from the *Edit* or context menu or use the keyboard Ctrl+X keys.

To paste clipboard contents

1. Display the screen into which the control is to be pasted (if not the current screen).
2. Choose *Paste* from the *Edit* or context menu or use the keyboard Ctrl+V keys.

The controls are placed onto the operator screen from the clipboard. They are placed in the same position they occupied when they were cut or copied.

Deleting Controls

To delete controls from the operator screen

1. Select the controls.
2. Press the Del key or select *Delete* from the *Edit* or context menu.

Aligning Controls

You can align controls left, right, top or bottom.

Left	The selected controls are aligned so that their left side is even with the furthermost left control.
Right	The selected controls are aligned so that their left side is even with the furthermost right control.
Top	The selected controls are aligned so that their tops are even with the topmost control.
Bottom	The selected controls are aligned so that their bottoms are even with the bottommost control.

To align controls

1. Select the controls to be aligned (at least two controls must be selected).
2. Select *Align Control* from the *Layout* menu, then choose the desired alignment: *Left*, *Right*, *Top*, or *Bottom*.

Moving Controls Front/Back

You can move a control to the front or back of overlapping controls.

To move a control front or back

1. Select the control.
2. Do one of the following:
   - To move the selected controls in front of any overlapping controls, choose *Move to Front* from the *Layout* menu.
   - To move the selected controls behind all other controls, choose *Move to Back* from the *Layout* menu.

HMI Symbol Operations

The Symbol Manager can be used directly inside the Operator Interface Editor. However, the Symbol Manager cannot be open inside the Operator Interface Editor and the Program Editor at the same time. When the Symbol Manager is opened from the Operator Interface Editor, it operates only on global symbols. It is available from the tool bar, Operator Interface menu, and control dialog boxes that allow the user to select symbols.

> **Note:** Only global symbols can be used within operator interface controls.

Editing Symbols

To edit global symbols

- Select *Symbol Manager* from the *Tools* menu. Refer to **Symbol Manager** in the **User Guide** for information on using the Symbol Manager.

Activating Configurations

To make the symbol edits available within the Program Editor, you must activate the configuration.

To activate the configuration, do one of the following

- Click *Apply* in the Symbol Manager .
- Select *Activate Config* from the *Tools* menu.

To save the configuration

- Select *Save Config* from the *Tools* menu.

Standard Controls

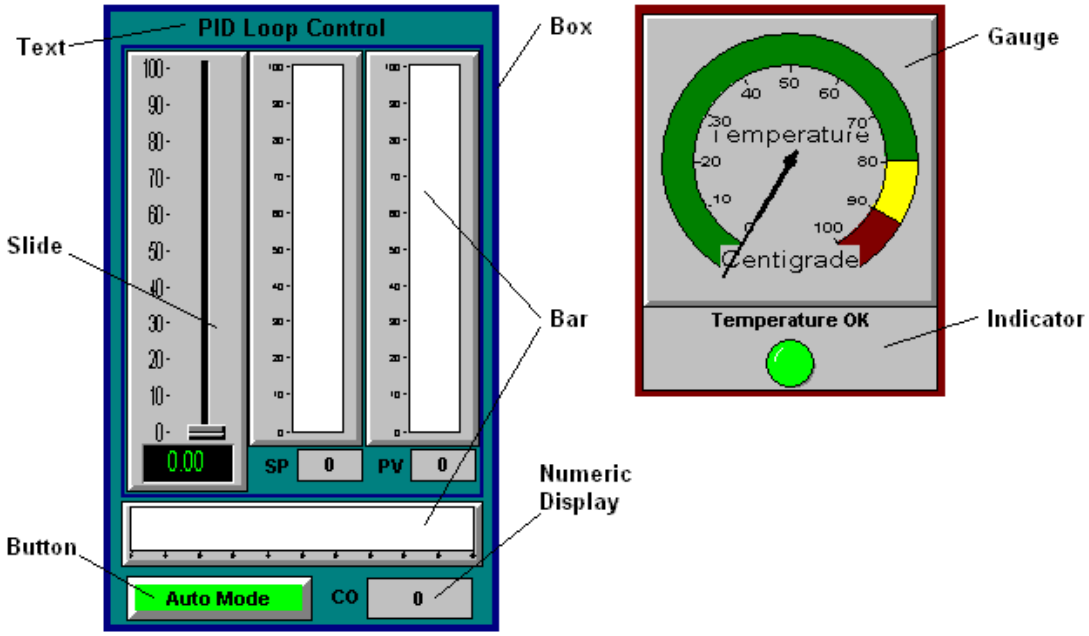
Introduction

The following lists and briefly describes standard controls and visual elements that can be used in an operator interface screen.

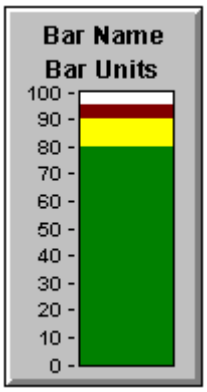
Control	Description
Bar	The bar control displays a scale and a moving bar. The moving bar tracks the current value of the symbol assigned to the bar. The bar orientation can be configured to be horizontal or vertical.
Bitmap	A bitmap is a visual element that can be used to describe the control application screen or add visual interest to a screen.
Box	A box is a visual element that can be used to group controls, give a border to text or bitmap, emphasize parts of a screen, or to otherwise make a screen visually interesting.
Click (multi-state) Button	A click (or multi-state) button executes a function when it is clicked. Functions include setting or clearing a Boolean symbol; displaying a screen; selecting, running, stopping, or aborting a program; getting operator input; and running an executable file. The button can be configured for single-state operation or two-state with override state operation. Functions can be programmed for each state and the button executes functions depending on its current state. It can be configured for automatic state change or state change based on a Boolean symbol.
Continuous Button	A continuous button executes a function when it is clicked. Functions include setting or clearing a Boolean symbol; displaying a screen; selecting, running, stopping, or aborting a program; getting operator input; and running an executable file. Functions can be programmed to execute when the button is pressed and when the button is released.
Gauge	The gauge control displays a scale and a rotating pointer. The rotating pointer tracks the current value of a symbol assigned to the gauge. Four gauge styles are available.

Control	Description
Indicator	An indicator is a state control whose appearance (text, color, etc.) changes depending on the value of a BOOL or BYTE data type symbol.
Numeric	The numeric display control is used to display a numeric value of a symbol.
Selected Program Status Panel	The Selected Program Status Panel displays the name and current status (i.e., running, stopped, etc.) of the selected program. The selected program is that program selected by a click button or continuous button SELECT PROGRAM command.
Slide	The slide control provides the capability of continuously changing a symbol value. When the slide is moved up or down, the symbol assigned to the slide control changes value.
Text	The text element can be used to label controls, groups of controls, document screen functions, or otherwise provide descriptions of the screen operation or functionality.

The following figures show a screen example using standard controls and visual elements.

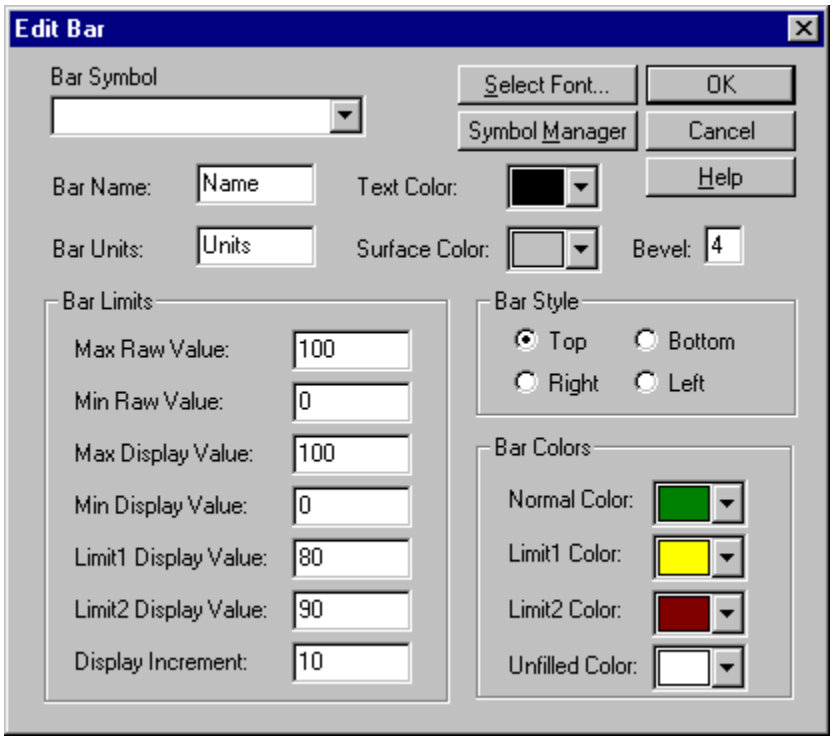


Bar



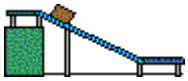
The bar control displays a scale and a moving bar. The moving bar tracks the current value of the symbol assigned to the bar. As the symbol value changes, the bar color changes, filling in the area of the bar corresponding to values less than the current symbol value. Limits can be given so that as the symbol value approaches these limits, the bar color changes to a specified color. The bar orientation can be configured to be horizontal or vertical.

The *Edit Bar* dialog box is shown in the following figure and described in the following table.



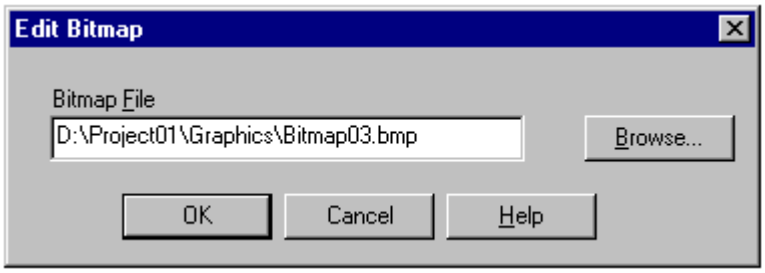
Field	Description
Bar Symbol	Specifies the symbol whose value is tracked by the moving bar. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Bar Name	Specifies a name for the control. For screen documentation only.
Bar Units	Specifies units for the control. For screen documentation only.
Text Color	Specifies a color for the control name, units, and scale numbering.
Surface Color	Specifies a background color for the control.
Bevel	Bevel affects the border of the control. Bevel values are from 0 to 6. A bevel of 0 makes the control appear flat. Increasing the bevel gives the control a 3-D appearance.
Bar Limits	
Max Raw Value Min Raw Value	Specifies the maximum and minimum values to which the moving bar responds. They limit the value of the symbol tracked by the bar. Any symbol value above the maximum raw value fills the moving bar to its upper limit, and any symbol value below the minimum raw value clears the moving bar. By reversing the values of the Max Raw Value and Min Raw Value fields, the inverted value of the bar symbol can be tracked. The maximum and minimum logic is reversed.
Max Display Value Min Display Value	Specifies the maximum and minimum numbering of the scale.
Limit1 Display Value Limit2 Display Value	Specifies limits at which the moving bar changes color (with respect to display values). For example, to show alarm conditions.
Display Increment	Specifies the scale numbering increment.
Bar Style	The bar style specifies the orientation of the bar control and the direction toward which the moving bar fills.
Bar Colors	There are four bar colors: Normal - the color of the bar when it is tracking the symbol value but before reaching a limit. Limit 1 - the color of the bar once it reaches its specified Limit 1. Limit 2 - the color of the bar once it reaches its specified Limit 2. Unfilled - the background color of the moving bar.
Select Font	Clicking this button displays a *Font* dialog box to specify the font, style, size, and color for the control text (name, units, and scale numbering).
Symbol Manager	Clicking this button opens the Symbol Manager.

Bitmap



A bitmap is a visual element that can be used to describe the control application screen or add visual interest to a screen. A bitmap will never obscure the appearance of any control or text.

The *Edit Bitmap* dialog box is shown in the following figure and described in the following table.

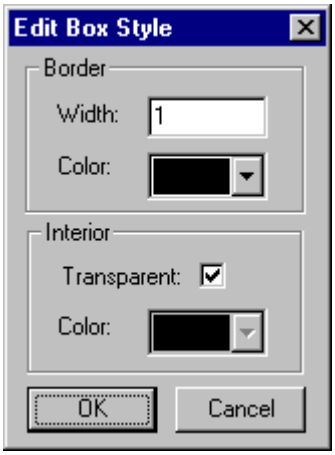


Field	Description
Bitmap File	Type in a path and bitmap file name or click Browse to locate one. Only Windows bitmap files (BMP) can be used.

Box

A box is a visual element that can be used to group controls, give a border to text or bitmap, emphasize parts of a screen, or to otherwise make a screen visually interesting. A box has a border and optional fill color. A box will never obscure the appearance of any control or text.

The *Edit Box* dialog box is shown in the following figure and described in the following table.



Field	Description
Border	This defines the appearance of the border. Type in a numeric border width and select a border color from the color list box.
Interior	If transparent is checked, the box has a transparent interior and the fill color is disabled. Otherwise, a fill color can be assigned to the box.

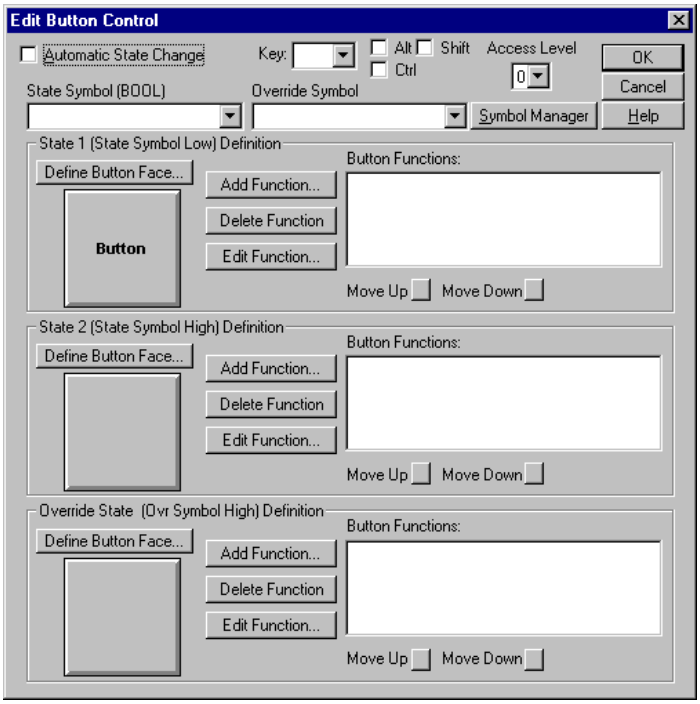
Buttons

You can add click (or multi-state) and continuous buttons to your HMI application.

Click Button

A click (or multi-state) button executes a function when it is clicked. Functions include setting or clearing a Boolean symbol; displaying a screen; selecting, running, stopping, or aborting a program; getting operator input; and running an executable file. The button can be configured for single-state operation or two-state with override state operation. Functions can be programmed for each state, and when the button is clicked, it executes the functions depending on its current state. It can be configured for automatic state change or state change based on a Boolean symbol. An access level and a keyboard key (or combination) can be assigned to the button

The *Edit Button Control* dialog box is shown in the following figure and described in the following table.



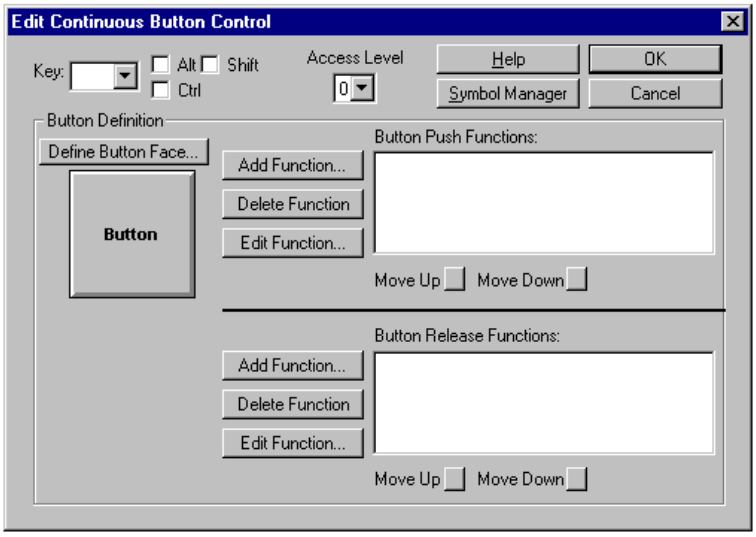
Field	Description
Automatic State Change	If checked, enables automatic state change when the button is clicked. It will alternate between State 1 and State 2 automatically. When enabled, the State Symbol edit box is disabled. If Automatic State Change is disabled and no state symbol is defined for the button, the button is a single state button and remains in State 1.
State Symbol	Defines the Boolean state symbol for the button. If a state symbol is defined for a button, the button changes to State 1 when the state symbol is low and to State 2 when the state symbol is high. Type or select a state symbol for the button from the list. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button. If Automatic State Change is disabled and no state symbol is defined for the button, the button is a single state button and remains in State 1.
Override Symbol	Defines the Boolean override symbol for the button. If an override symbol is defined for a button, the button changes to the override state when the override symbol is high. Type or select an override symbol for the button from the list. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Key	Specifies a keyboard key that can be pressed to click the button. Function keys can be selected from a list, or a single key in the range A..Z or 1..9 can be typed. The Alt, Shift, and Ctrl keys can also be used in combination with another key.
Access Level	Specifies an access level for the button. If an access level is specified, the access level of the operator interface must be set to the specified access level or higher in order to click the button.
Button Functions	Displays the list of control functions for each button state. The control functions for the current button state are executed when the click button is pressed and released. Refer to Button Functions for a description of all button functions.
Add Function	Adds a control function to the Button Functions list. Refer to the Button Functions table for a list and description of functions that can be assigned to a button control. If the function added is a RUN, STOP, or ABORT function, and a program name is given, an Edit Program button appears below the function list. If the Edit Program button is pressed, the Program Editor is activated and the specified file is opened. If the file does not exist a new file is opened.

Field	Description
Delete Function	Deletes the highlighted control function from the Button Functions list.
Edit Function	Edits parameters of the highlighted control function in the Button Functions list.
Move Up	Moves the highlighted control function up one position in the Button Functions list.
Move Down	Moves the highlighted Control Function down one position in the Button Functions list.
Define Button Face	You can configure how the button looks, refer to Define Button Face for more information.

Continuous Button

A continuous button executes a function when it is clicked. Functions include setting or clearing a Boolean symbol; displaying a screen; selecting, running, stopping, or aborting a program; getting operator input; and running an executable file. Functions can be programmed to execute when the button is pressed and when the button is released. An access level and a keyboard key (or combination) can be assigned to the button

The *Edit Continuous Button Control* dialog box is shown in the following figure and described in the following table.



Field	Description
Key	Specifies a keyboard key that can be pressed to click the button. Function keys can be selected from a list, or a single key in the range A..Z or 1..9 can be typed. The Alt, Shift, and Ctrl keys can be used in combination with another key.
Access Level	Specifies an access level for the button. If an access level is specified, the access level of the operator interface must be set to this access level or higher in order to click the button.
Button Push Function	Displays the list of control functions executed when the button is pushed.
Button Release Function	Displays the list of control functions executed when the button is released. Refer to Button Functions for a description of all button functions.
Add Function	Adds a control function to the Button Push Functions list or the Button Release Functions list. Refer to the Button Functions table for a list and description of functions that can be assigned to a button control. If the function added is a RUN, STOP, or ABORT function, and a program name is given, an Edit Program button appears below the function list. If the Edit Program button is pressed, the Program Editor is activated and the specified file is opened. If the file does not exist a new file is opened.
Delete Function	Deletes the highlighted control function from the Button Push Functions list or the Button Release Functions list.
Edit Function	Edits parameters of the highlighted control function in the Button Push Functions list or the Button Release Functions list.
Move Up	Moves the highlighted control function up one position in the Button Push Functions list or the Button Release Functions list.
Move Down	Moves the highlighted Control Function down one position in the Button Push Functions list or the Button Release Functions list.
Define Button Face	You can configure how the button looks, refer to for more information.
Define Button Face	
Line 1 Title Line 2 Title	The button can be given two lines of text to describe its function.
Bevel	Bevel affects the border of the control. Bevel values are from 0 to 6. A bevel of 0 makes the control appear flat. Increasing the bevel gives the control a 3-D appearance.
Surface Color	Specifies a background color for the control.
Select Icon Remove Icon	Clicking Select Icon displays icons that can be assigned to the button.

Field	Description
Select Font	Clicking this button displays a *Font* dialog box to specify the font, style, size, and color for the control text.

Button Functions

The following table lists the control functions that can be assigned to button controls. The function is executed when the button is clicked (or released) with the screen active.

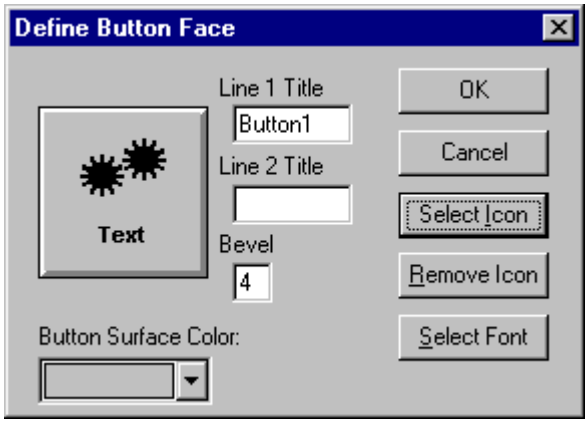
Button Function	Description
SCREEN... (Activate Operator Screen)	Activates a specified operator screen. After selecting SCREEN as the button function, you are prompted for the screen name to activate at runtime.
RUN... (Run Program)	Runs a specified program or the program selected by the operator using the Select Program function (SELECTED PROGRAM). After selecting RUN as the button function, you are prompted for the program name to activate at runtime. Alternatively, you can select SELECTED PROGRAM to allow the operator to select the program at run time.
STOP... (Stop Program)	Stops a specified SFC program or an SFC program selected by the operator using the Select Program function (SELECTED PROGRAM). The program can be resumed from where it was stopped using the Run Program function. Only SFC programs can be stopped. After selecting STOP as the button function, you are prompted for the program name to stop at runtime. Alternatively, you can select SELECTED PROGRAM to allow the operator to select the program at runtime.
ABORT... (Abort Program)	Aborts the specified program or the program selected by the operator using the Select Program function (SELECTED PROGRAM). After selecting ABORT as the button function, you are prompted for the program name to abort at runtime. Alternatively, you can select SELECTED PROGRAM to allow the operator to select the program at runtime.
SET... (Set Symbol)	Sets the value of a specified symbol or sets system ESTOP. After selecting SET as the button function, you are prompted for the symbol to set at runtime. Alternatively, you can select ESTOP. If the symbol is a BOOL data type, then its value is set to '1' at runtime. If the symbol is other than a BOOL data type, you are further prompted for the value you wish to give the symbol.

Button Function	Description
CLEAR... (Clear Symbol)	Clears a specified symbol or resets system ESTOP. After selecting CLEAR as the button function, you are prompted for the symbol to clear at runtime. Alternatively, you can select ESTOP. At runtime, symbol values are set to their default 'empty' values.
OP INPUT... (Operator Input)	At design time, prompts for a symbol and prompt message. At run time, displays an operator entry box with the specified operator prompt. The value entered by the operator is stored in the specified symbol. After selecting OP INPUT as the button function, you are prompted for 1) the symbol that the operator can set at runtime and 2) an operator prompt.
OP KEYPAD INPUT... (Operator Keypad)	At design time, prompts for a symbol and prompt message. At run time, displays an operator keypad entry control with the specified operator prompt. Keypad entry can be from mouse-click or touch screen. The value entered by the operator is stored in the specified symbol. After selecting OP INPUT as the button function, you are prompted for 1) the symbol that the operator can set at runtime and 2) an operator prompt.
EXECUTE... (Run Executable File)	Run the specified *.exe file. Running one of the product utilities requires the appropriate authorized access level. The product utilities include the Program Editor. After selecting EXECUTE as the button function, you are prompted for the program to be executed at runtime. The program editor appears in the drop-down list box, but you can browse to any path in the system. **Note:** A path can be provided. If the path contains spaces, the entire string must be enclosed in double quotes, for example: "C:\my .program folder\program1.exe"
SELECT PROGRAM (Select Program)	When the screen is active, displays a list of Sequential Function Chart programs in the project from which the operator selects the desired program. The Run Program, Stop Program and Abort Program functions can be set up to act on the operator selected program (SELECTED PROGRAM).
MDI Input (Manual Data Input)	Displays a prompt to enter a single line of RS-274 code. When MDI Input is added as a function, you must provide a global string symbol to hold the RS-274 code. After selecting MDI INPUT as the button function, you are prompted for the STRING symbol into which the RS-274 code will be entered.

Button Function	Description
MDI Run (Manual Data Input)	Runs the RS-274 code for the associated MDI Input. When MDI Run is added as a function, you must provide the same global string symbol as the associated MDI Input function. **Note:** You can also insert a Numeric Display and assign it the associated global string symbol to display the current MDI Input line and its status. After selecting MDI RUN as the button function, you are prompted for the STRING symbol corresponding to the MDI INPUT function to be run.

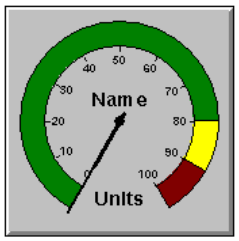
Define Button Face

Refer to the following table for information on the Define Button Face fields.



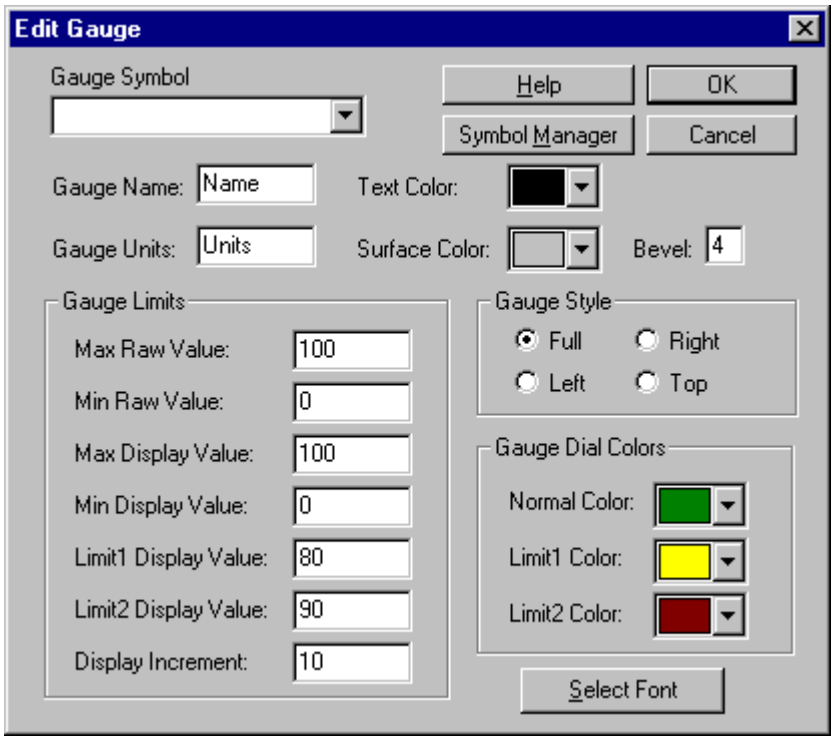
Field	Description
Line 1 Title Line 2 Title	The button can be given two lines of text to describe its function.
Bevel	Bevel affects the border of the control. Bevel values are from 0 to 6. A bevel of 0 makes the control appear flat. Increasing the bevel gives the control a 3-D appearance.
Surface Color	Specifies a background color for the control.
Select Icon Remove Icon	Clicking Select Icon displays icons that can be assigned to the button.
Select Font	Clicking this button displays a *Font* dialog box to specify the font, style, size, and color for the control text.

Gauge



The gauge control displays a scale and a rotating pointer. The rotating pointer tracks the current value of the symbol assigned to the gauge. As the symbol value changes, the pointer moves clockwise or counter-clockwise to indicate the current symbol value. Limits can be given that change the scale color to the specified limit color. Four gauge styles are available.

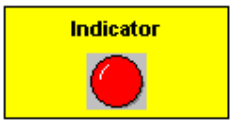
The *Edit Gauge* dialog box is shown in the following figure and described in the following table.



Field	Description
Gauge Symbol	Specifies the symbol whose value is tracked by the rotating pointer. If needed, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Gauge Name	Specifies a name for the control. For screen documentation only.

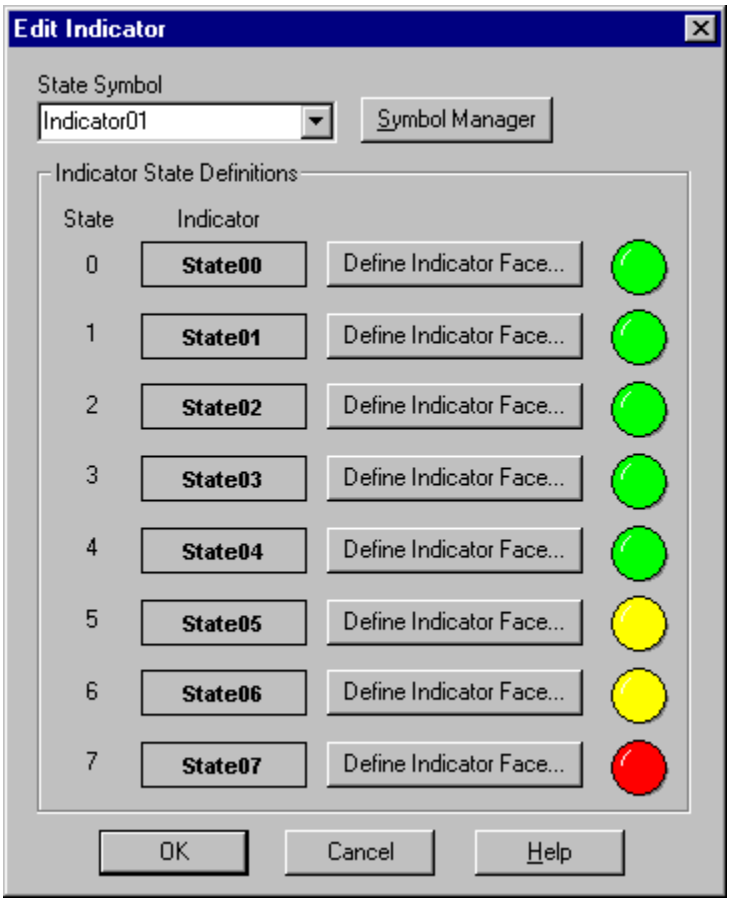
Field	Description
Gauge Units	Specifies units for the control. For screen documentation only.
Text Color	Specifies a color for the control name, units, and scale numbering.
Surface Color	Specifies a background color for the control.
Bevel	Bevel affects the border of the control. Bevel values are from 0 to 6. A bevel of 0 makes the control appear flat. Increasing the bevel gives the control a 3-D appearance.
Gauge Limits	
Max Raw Value Min Raw Value	Specifies the maximum and minimum values to which the gauge pointer responds. They limit the value of the symbol tracked by the pointer. Any symbol value above the maximum raw value positions the moving pointer to its upper limit, and any symbol value below the minimum raw value positions the moving pointer at its lower limit. By default, the moving pointer moves in a clockwise direction as the control symbol value increases. To make the pointer move in a counter-clockwise direction, swap the Max Raw Value and Min Raw Value values.
Max Display Value Min Display Value	Specifies the maximum and minimum numbering of the scale.
Limit1 Display Value Limit2 Display Value	Specifies limits at which the gauge scale changes color. For example to show alarm conditions.
Display Increment	Specifies the scale numbering increment.
Gauge Style	There are four gauge styles: Full circle. Semi-circle with the rounded side up. Semi-circle with the rounded side to the left. Semi-circle with the rounded side to the right
Gauge Dial Colors	There are three gauge colors: Normal - the color of the gauge scale below a limit. Limit 1 - the color of the gauge scale once it reaches its specified Limit 1. Limit 2 - the color of the gauge scale once it reaches its specified Limit 2.
Symbol Manager	Clicking this button opens the Symbol Manager.
Select Font	Clicking this button displays a *Font* dialog box to specify the font, style, size, and color for the control text (name, units, and scale numbering).

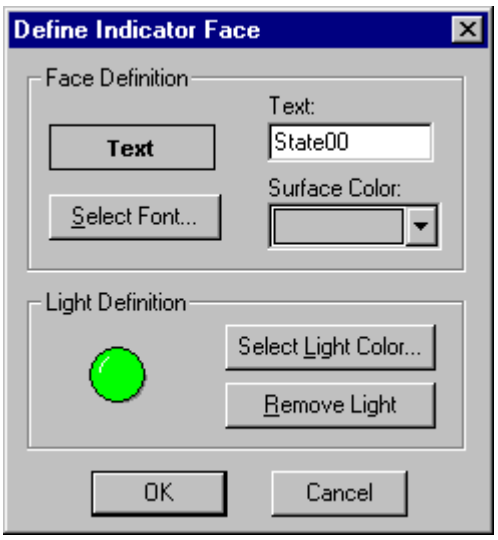
Indicator



An indicator is a state control whose appearance (text, color, etc.) changes depending on the value of a BOOL or BYTE data type symbol. A BOOL symbol can select between two states and a BYTE symbol can select from up to eight states.

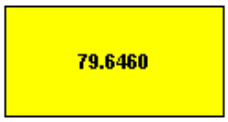
The *Edit Indicator* and *Define Indicator Face* dialog boxes are shown in the following figure and described in the following table.





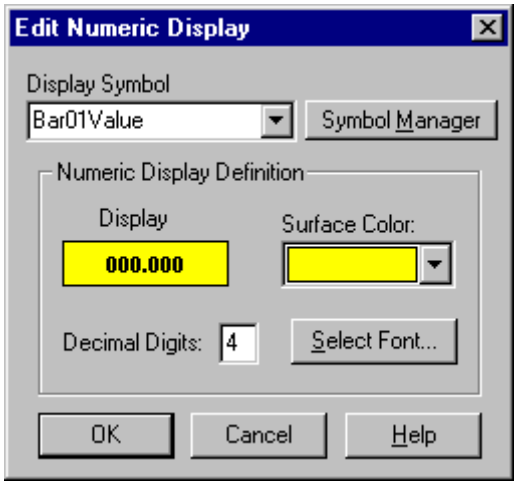
Field	Description
State Symbol	Specifies the symbol to which the indicator responds. State symbol data types can be BOOL or BYTE. If the state symbol is a BOOL, the indicator can have two states; if a BYTE, the indicator can have up to eight states. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Symbol Manager	Clicking this button opens the Symbol Manager.
Indicator State Definitions	For each possible binary state (0 through 7), the corresponding indicator text, background color, and optional light are shown.
Define Indicator Face	Clicking this button displays the *Define Indicator Face* dialog box. This dialog box is used to define the text, text color, and background color for a state and the font type and size for the entire indicator. An optional colored light can also be defined for each indicator state.
Face Definition	
Text	The string to be displayed for the state.
Select Font	Defines the font and size for the indicator control and the text color for the state.
Surface Color	Defines the background color for the state.
Light Definition	
Select Light	Displays a group of light colors that can be selected for the state.
Remove Light	Removes a light defined for the state.

Numeric Display



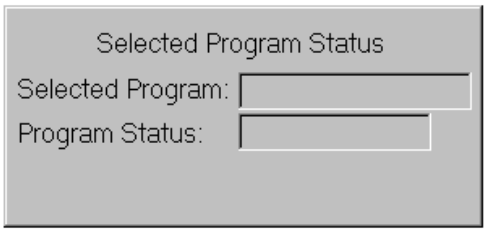
The numeric display control is used to display the value of a symbol (numeric or string).

The *Edit Numeric Display* dialog box is shown in the following figure and described in the following table.



Field	Description
Display Symbol	Specifies the symbol whose value is displayed by the numeric display control. Select a symbol name from the drop down list box or type a symbol name into the edit box. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Symbol Manager	Clicking this button opens the Symbol Manager.
Display	The selected text color, background color, and font type is displayed in the box titled *Display*.
Surface Color	Defines the background color for the control.
Decimal Digits	Defines the number of digits to the right of the decimal point that are displayed.
Select Font	Defines the text font, size, and color for the control.

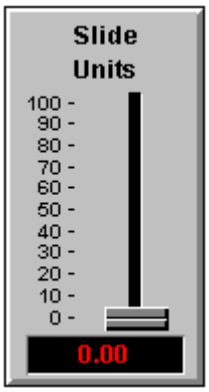
Selected Program Status Panel



The Selected Program Status Panel displays the name and current status (i.e., running, stopped, etc.) of the selected program. The selected program is that program selected by a click button or continuous button SELECT PROGRAM command.

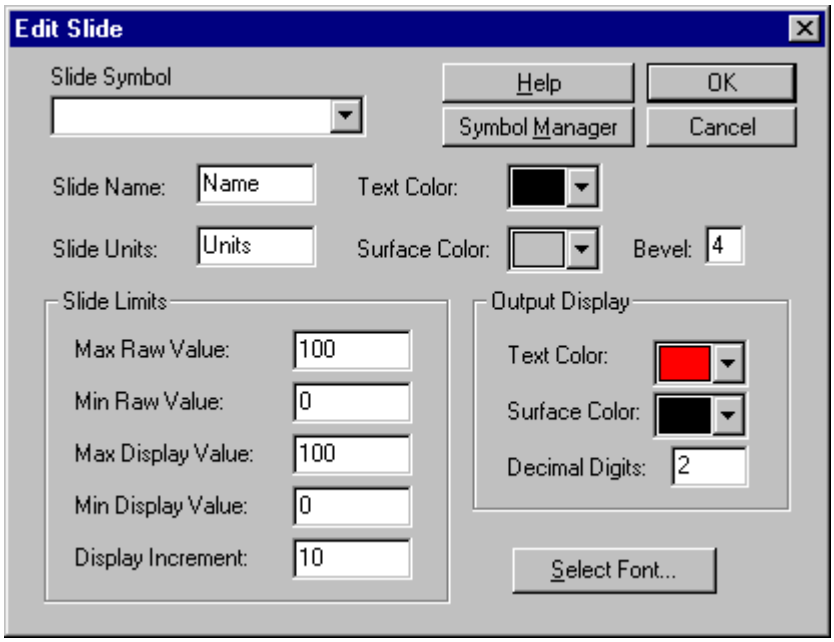
The Selected Program Status Panel has no configuration options.

Slide



The slide control provides the capability of continuously changing a symbol value. When the slide is moved up or down, the symbol assigned to the slide control changes value. The current setting of the slide is displayed at the bottom of the control. A scale running along the slide path gives an approximate value to which the slide is set.

The *Edit Slide* dialog box is shown in the following figure and described in the following table.

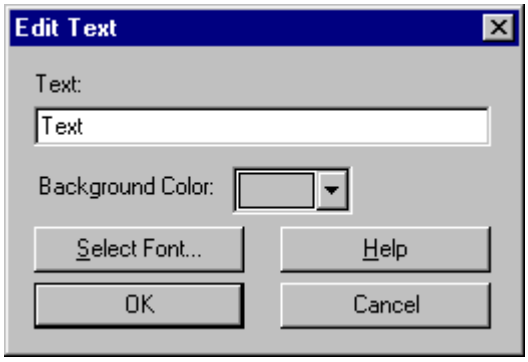


Field	Description
Slide Symbol	Specifies the symbol whose value is changed by the slide control. If necessary, a new symbol can be defined by opening the Symbol Manager with the *Symbol Manager* button.
Slide Name	Specifies a name for the control. For screen documentation only.
Slide Units	Specifies units for the control. For screen documentation only.
Text Color	Specifies a color for the control name, units, and scale numbering.
Surface Color	Specifies a background color for the control.
Bevel	Bevel affects the border of the control. Bevel values are from 0 to 6. A bevel of 0 makes the control appear flat. Increasing the bevel gives the control a 3-D appearance.
Slide Limits	The raw values determine the interaction between the control and its symbol; the display values determine the interaction between the control and its scale.
Max Raw Value Min Raw Value	Specifies the maximum and minimum values that the slide can assign to the symbol. The slide will not set values above the maximum or below the minimum values.
Max Display Value Min Display Value	Specifies the maximum and minimum numbering of the scale.
Display Increment	Specifies the scale numbering increment.
Output Display	The Output Display displays the control display value (not the symbol value). That is, the position of the slide with respect to the scale.
Text Color	Defines the Output Display text color.
Surface Color	Defines the background color for the output display.
Decimal Digits	Defines the number of digits to the right of the decimal point that are displayed.
Symbol Manager	Clicking this button opens the Symbol Manager.
Select Font	Clicking this button displays a *Font* dialog box to specify the font, style, size, and color for the control text (name, units, and scale numbering).

Text

The text element can be used to label controls, groups of controls, document screen functions, or otherwise provide descriptions of the screen operation or functionality.

The *Edit Text* dialog box is shown in the following figure and described in the following table.



Field	Description
Text	Type the text string to be displayed in the Text box.
Background Color	Defines the background color of the text element.
Select Font	Defines the text color, font, and font size for the text string.

ActiveX Controls

Introduction

ActiveX controls are DLLs that have an OLE2 interface, using an OCX file extension by default (instead of DLL). ActiveX controls include simple controls such as buttons and edit boxes, more sophisticated controls such as gauges, and DLL components that have no user interface. ActiveX controls require a container and this is provided by the operator interface.

ActiveX controls have three types of functions:

- **Properties** – these are function calls inside the control that the container can call to configure the appearance of the control (e.g., colors, fonts, captions, bitmaps, etc.). For instance, if a control implementer is writing code for a pushbutton that has a hard-coded blue face it will not look like it belongs in an environment where all other buttons have a gray face.
- **Methods** – these are function calls inside the control that allow the container to send or receive information about the control. For instance, getting the value of a cell in a grid control or telling the control to re-paint itself.
- **Events** – these are functions that get called when a Windows event occurs within the control. For instance, a mouse click or a key press. The container is notified when these events occur.

Properties, methods, and events are configured within the Operator Interface Screen Editor after the control is added to a screen.

ActiveX Limitations

The following is a list of limitations when using ActiveX controls in the Operator Interface:

1. No support for compound controls (controls within other controls).
2. No type coercion (from DWORD to INT for example).
3. No support for controls that do not have their own property pages.
4. No support for changing fonts at runtime. The Operator Interface does not support a "font" type.

5. No support for method/event parameters of type VT_VARIANT.
6. No support for complex data types (structures) for method/event parameters.

ActiveX and Standard Controls

Standard controls provided with the product provide basic functionality for a programmable control system operator interface. These controls also have an internal knowledge of the programmable control system. This knowledge allows them to hide many complex details from the end-user.

To extend the functionality or to accommodate special requirements, the operator interface serves as a container for ActiveX controls. Any available ActiveX control can be dropped into an operator interface screen and communicate with the run-time engine (provided the data types are compatible with the control system software).

Note the following:

- Knowledgeable users can write their own ActiveX controls, or they can be obtained from third-party sources. It is the user's responsibility to ascertain the suitability of any third-party ActiveX control for an application and to obtain its documentation. Poorly behaved controls may cause problems - contact the control implementer regarding any problems you are having as a result of using an ActiveX control.
- Third-party ActiveX controls will **not** have any internal knowledge of the programmable control system. Examples of the knowledge they will **not** have include: how to command an ESTOP and how to start, stop and abort the programmable control system programs.
- ActiveX controls require a registration process that allows them to be seen by Windows NT or Windows 95. Control registration may or may not be provided by the control implementers. To accommodate control registration, a registration process is implemented within the Operator Interface Screen Editor.

ActiveX Control Sources

Two third-party ActiveX control packages have been tested and are recommended sources for ActiveX controls:

VICOMponents™ Version 4.0 by ComputerBoards, Inc.

ComputerBoards Inc.

125 High Street

Mansfield, MA 02048

www.computerboards.com

Global Majic Software ActiveX controls.

Global Majic Software, Inc.
P.O. Box 322
Madison, Alabama 35758
gms@globalmajic.com
www.globalmajic.com

Registering ActiveX Controls

An ActiveX control must be registered before it can be used in an operator interface screen. If the ActiveX control has not been previously registered, it can be registered within the Operator Interface Screen Editor. Registered controls appear in the list displayed when *Insert ActiveX Control* is selected from the *Edit* menu.

To register an ActiveX control

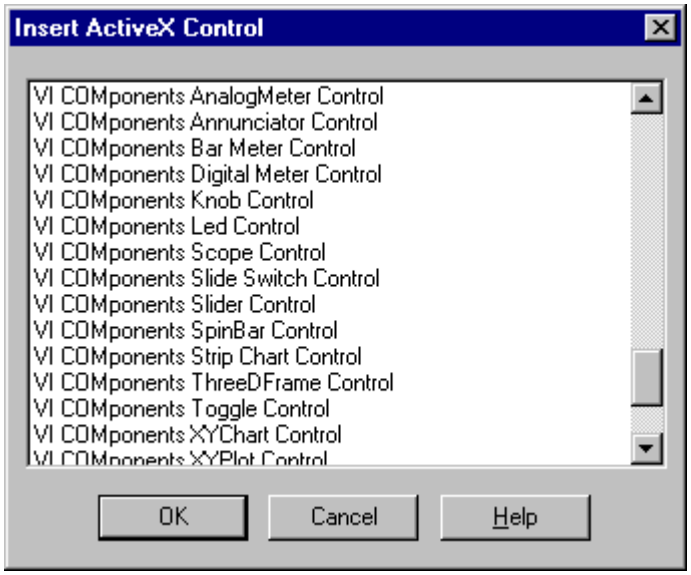
1. Choose *Register ActiveX Control* from the *Edit* menu. The *OCX Files to Register* dialog box appears.
2. Select the control to register, then click *Open* to register it. If successful, a message appears indicating that the registration was successful.

   Note: This process does not guarantee that the control can be properly inserted into a screen. For example, if the control is not licensed or is not implemented properly.

Inserting ActiveX Controls

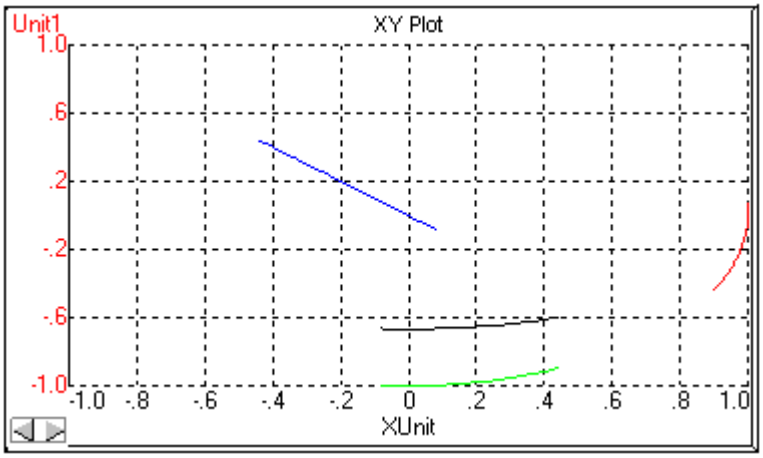
To insert an ActiveX control

1. Choose *Insert ActiveX* control from the *Edit* menu. The *Insert ActiveX Control* dialog box appears.
2. Select an ActiveX control and click *OK* to continue.



Alternatively, a list of the last four recently inserted ActiveX controls appears at the bottom of the *Edit* menu, from which the control to be inserted can be selected.

An example of the VI components XY Plot control is shown in the following figure.



Notes:

1. The list contains all ActiveX controls registered on the PC. Some may not make sense to use in a control environment.
2. If you distribute an operator interface screen configuration using an ActiveX control, the ActiveX control must be registered on the target system.
3. Refer to the control implementer for control usage and distribution rights and agreements.

Editing ActiveX Controls

ActiveX controls can be moved and re-sized as with the standard controls. The effect of re-sizing a control is dependent on the control and does not necessarily produce the expected result. Other editing operations such as delete, cut, copy, and paste work the same as for standard controls.

Editing Properties

To edit properties, do one of the following

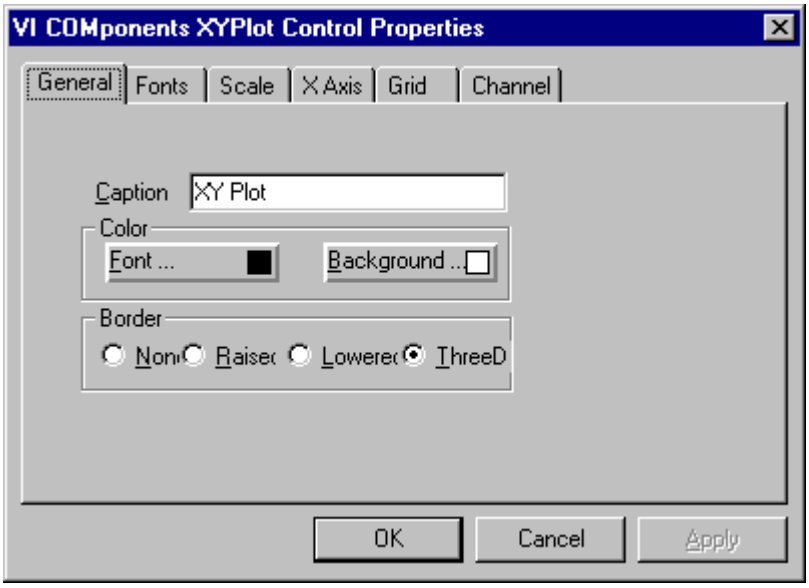
- Select the ActiveX control and choose *Edit ActiveX Properties* from the *Edit* or context menu, or use the tool bar.
- Double-click on the ActiveX control and choose *Edit Property Pages* from the *Select ActiveX Editing Mode* dialog box that appears.

Property page dialogs come from inside the control itself. This means that all controls have a different property page dialog box. For controls that do not have a property page dialog box a message indicating this appears instead.

Note: You must refer to the ActiveX control provider for information on the control properties. This information is not provided within

the Operator Interface Screen Editor itself. However, if a help file is provided with the control, it may be accessed from the control dialog box.

A typical property pages dialog box is shown in the following figure.



Editing Methods

There are three general steps needed to configure an ActiveX control's methods:

- Select which of the control's methods will be called by the operator interface.
- Assign symbols to the method's arguments and return value (if needed).
- Define when the method will be called.

Controls have a variable number of internal methods, where each method is a function call to the control. A function's arguments and return value can be one of many possible data types (currently 38). Each of these data types may or may not be mappable to a system supported data type. If it is not possible to map a supported data type to a type that is expected in the method, it will be impossible to call this method. If the user attempts to set up this method, an appropriate error message appears.

Argument types for ActiveX controls start with *VT_*. The following table shows how the types are mapped to the supported data types.

ActiveX Control Data Type	System Supported Data Type
VT_BOOL	BOOL
VT_UI1, VT_I1	BYTE
VT_UI2	WORD
VT_UI4, VT_UINT	DWORD
VT_I2, VT_INT	INT
VT_I4	DINT
VT_R4, VT_R8	REAL
VT_BSTR	STRING
VT_VOID, VT_EMPTY, VT_NULL	Not mapped
VT_PTR	Pointer to all above VT_ types will be supported

One part of configuring an ActiveX control is to define when the control's methods should be called. This is done on a timer basis by a 200 millisecond timer running in the operator interface. Methods are eligible to be called every 200 milliseconds or a multiple of 200 milliseconds. The multiple will be user defined up to a value of 100 (=20 seconds).

A method is eligible to be called in two different ways:

Every timer interval – or a user-defined multiple of the timer interval.

A BOOL symbol goes TRUE – when a user-defined BOOL symbol goes true, the method is called within the next timer interval. After the method returns, the BOOL symbol is optionally set FALSE.

To edit methods

1. Do one of the following
   - Select the ActiveX control and choose *Edit ActiveX Methods* from the *Edit* or context menu, or tool bar.
   - Double-click on the ActiveX control and choose *Edit Methods* from the *Select ActiveX Editing Mode* dialog box that appears.
2. A *Method Setup* dialog box appears. Refer to the following table and configure methods as needed. When done, click *OK* to save the configuration and return to the Operator Interface Screen Editor.

Field	Description
Name	Select the method to configure from the Name list box. They will be in alphabetical order.
Enable Method	Check this to enable the method. The method must be enabled to be called.

Field	Description
Return	If the method has a return type of VOID, VT_VOID appears in the Return box and it cannot be changed; otherwise the return type is indicated and a symbol of the appropriate type must be assigned in the Return list box. If necessary click *Symbol Manager* to create a new symbol.
Arguments	If the method accepts arguments, they are listed with their number, name, and data type. Methods are limited to 16 arguments. An error message appears if the method has greater than 16 arguments. For each argument to which a value is to be assigned, assign a symbol from its Symbol Assignment list. If necessary, click *Symbol Manager* to create new symbols. Pointers to arrays (defined in the Symbol Manager) can be passed to the method. Specify the array name without a subscript to pass the entire array. The array size must match the method's required size.
When method should be called	If the method is to be called strictly by time interval, click *Every Timer Interval* and select a multiplier. If the method is to be called when a Boolean symbol goes TRUE, select a time interval multiplier, then click on *BOOL value goes TRUE* and select a Boolean symbol from the list box. If *Automatic Reset* is checked, the Boolean symbol is reset FALSE after the method is called.
Symbol Manager	Displays the Symbol Manager.
Dialog Help	Displays help for the Method Setup dialog box.
Control Help	Displays help for the ActiveX control (provided by the control implementer).
Method Help	Displays help for the ActiveX control's method that is currently displayed (provided by the control implementer).

Editing Events

Most ActiveX controls have a list of events that they report to the operator interface container. For example, mouse clicks or key presses. Events may or may not have parameters associated with them. For example, an *on focus* event does not have any parameters, but a mouse-click event might have two integer parameters for x and y coordinates. When an event occurs, the operator interface will be able to do two separate things:

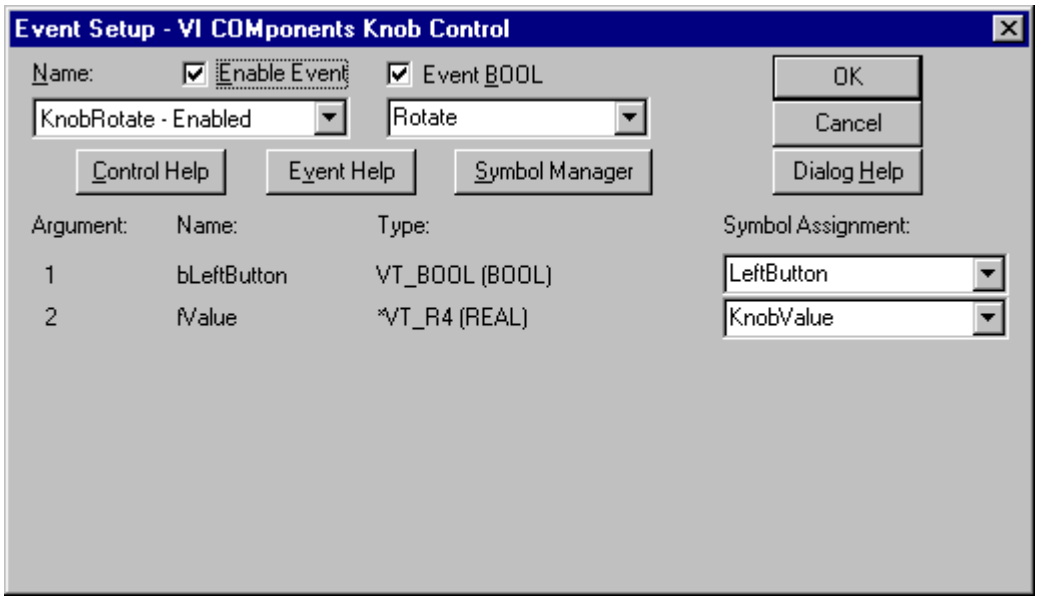
- **Set a Boolean symbol TRUE** – this is an option if the event has at least one parameter (argument). If the event has no parameters, this is mandatory. (The BOOL is **not** set to FALSE by the operator interface after the event.)

- **Assign the value of an event argument to a symbol** – if a symbol has been assigned to an event argument in the configuration, the value of the argument is assigned to the symbol in operation.

To edit events

1. Do one of the following:
   - Select the ActiveX control and choose *Edit ActiveX Events* from the *Edit* or context menu, or tool bar.
   - Double-click on the ActiveX control and choose *Edit Events* from the *Select ActiveX Editing Mode* dialog box that appears.
2. An *Event Setup* dialog box appears. Refer to the following table and configure events as needed. When done, click *OK* to save the configuration and return to the Operator Interface Screen Editor.

For controls that do not have an *Event Setup* dialog box a message indicating this appears instead.



Field	Description
Name	Select the event to configure from the Name list box.
Enable Event	Check this to enable the event. The event must be enabled to be reported to the operator interface.

Field	Description
Event BOOL	If this is enabled and a Boolean symbol has been assigned, then in operation, the event sets the Boolean symbol TRUE. If necessary click Symbol Manager to create a new symbol.
Arguments	If the event accepts arguments, they are listed with their number, name, and data type. Events are limited to 16 arguments. An error message appears if the event has greater than 16 arguments. For each argument that a value is needed, assign a symbol from its Symbol Assignment list. If necessary click Symbol Manager to create new symbols.
Symbol Manager	Displays the Symbol Manager.
Dialog Help	Displays help for the Method Setup dialog box.
Control Help	Displays help for the ActiveX control (provided by the control implementer).
Method Help	Displays help for the ActiveX control's methods (provided by the control implementer).

Motion Controls

Motion Controls Overview

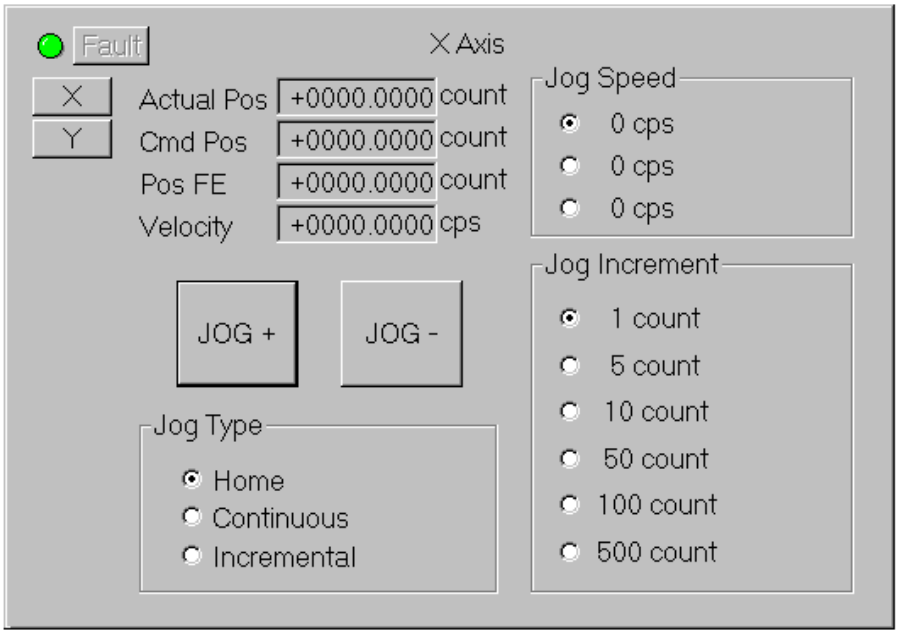
The following motion controls are included with the product:

Standard HMI

- Jog Panel
- Multi Axis Status Panel
- RS274 Block Display
- Single Axis Panel

Jog Panel

The Jog Panel provides the functions needed to home and jog configured axes. The jog axis panel contains buttons to select the jog axis, jog type (home, continuous, or incremental), jog speed, and jog increment. The selected axis is displayed at the top of the Jog Panel. The Jog Panel also displays absolute position, commanded position, following error and velocity status for the selected axis. An axis fault indicator is used to indicate that a fault has occurred on the selected axis. If no fault has occurred the indicator is green. When an axis fault occurs, the indicator turns red and the fault button is enabled. To view detailed information on the axis fault, press the fault button. When the axis fault is cleared the indicator turns green. The selected axis is jogged or homed in the desired direction by pressing and holding the Jog+ or Jog- button. If the Jog+ or Jog- button is released, the jog or home is aborted. For incremental jogs, the axis stops when the increment is completed. To jog another increment, release and press the Jog+ or Jog- button.

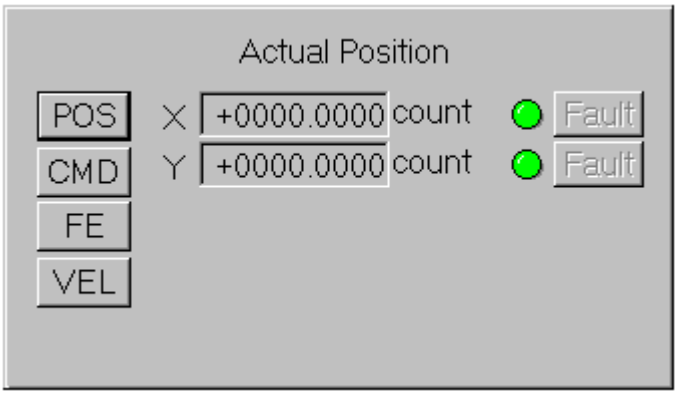


The Jog Panel is automatically set up for the number of configured axes. There are no configuration options for the Jog Panel control.

Multi-Axis Status Panel

The Multi Axis Status Panel displays the specified status of all configured axes at the same time. The specific status which is displayed is selected from any of the following: absolute position (POS), commanded position (CMD), following error (FE) and velocity (VEL). The desired status is selected by pressing the specific status button on the panel. The title of the selected status is displayed at the top of the panel. For each axis, a fault indicator is used to indicate that a fault has occurred on that axis. If no fault has occurred the indicator is green. When an axis fault occurs, the indicator for that axis turns red and the fault button for that axis is enabled. To view detailed information on the axis fault, press the fault button. When the axis fault is cleared the indicator turns green.

The Multi Axis Status Panel is automatically set up for the number of configured axes. There are no configuration options for the Multi Axis Status Panel control.



Using the Multi-Axis Motion Status Panel

The selected status information is displayed at the top of the status box. To change the selected status information, press the button with the desired status information. Multiple copies of the multi-axis motion status can be activated at the same time.

RS274 Block Display

The RS274 Block Display allows you to display the active RS274 block commands on the operator interface screen. All the block commands in the active SFC step will be displayed. The active block will be highlighted green and a pointer (>) will be at the front of the block.

Editing an RS274 Block Display

To edit an RS274 Block Display, double click on the desired RS274 Block Display or select the desired RS274 Block Display and press Enter. The Edit RS274 Block Display dialog box is displayed allowing the user to change the RS274 Block Display definition data.

The RS274 Program Symbol combo box is used to define the STRING symbol that contains the file path of the SFC program from which active RS274 blocks will be displayed. To select a display symbol for the RS274 Program Symbol, either select a symbol name from the drop down list box or type a symbol name into the edit box.

The RS274 Block Display uses OLE technology to link to the Program Editor. The Program Editor displays the RS274 blocks in the RS274 Block Display on the operator screen. Whenever the program symbol changes, the RS274 Block Display links to the new SFC program.

Only RS274 blocks located in the top level SFC program are displayed. If RS274 blocks are located in a macro step SFC they are not displayed.

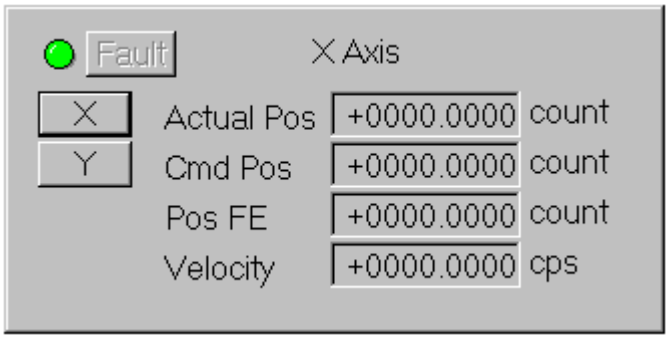
Performance Considerations

To link the RS274 Block Display to the Program Editor, the OLE subsystem must run the Program Editor application. The time required to startup the Program Editor may cause an undesirable delay in the operator interface on some systems. To eliminate this delay in the operator interface, the following two recommendations are presented: 1) Start the Program Editor minimized in the background (with read only access if desired) before the operator interface; 2) Initialize all RS274 Block Display symbols with the desired SFC file paths. Since all OLE links are connected when the operator interface is activated, the link delay occurs at system startup.

Single Axis Panel

The Single Axis Status Panel displays the status for a single axis. The axis for which status is displayed is selected by pressing the desired axis button on the panel. The title of the selected axis is displayed at the top of the panel. The status which is displayed for the selected axis consists of absolute position, commanded position, following error, and velocity. An axis fault indicator is used to indicate that a fault has occurred on the selected axis. If no fault has occurred the indicator is green. When an axis fault occurs, the indicator turns red and the fault button is enabled. To view detailed information on the axis fault, press the fault button. When the axis fault is cleared the indicator turns green.

The Single Axis Status Panel is automatically set up for the number of configured axes. There are no configuration options for the Single Axis Status Panel control.



Using the Single Axis Motion Status Panel

The selected axis is displayed at the top of the status box. To change the selected axis, press the button with the desired axis name. Multiple copies of the single axis motion status can be activated to view complete axis status on more than one axis at the same time.

Addendum

HMI Background Color

The background color of the HMI is set to the Windows application background color.

To change the HMI background color

1. Open the *Display* item in the Control Panel.
2. When the *Display Properties* dialog box appears, select the *Appearance* tab.
3. In the *Item* list box select *Application Background* and then select your desired color. The next time you open the HMI, the background color will be changed.

Securing Access Outside the HMI

This procedure describes how to block operator access to the operating system (from the HMI).

1. Add a password protected button to the MMI to execute the explorer (**EXECUTE c:\winnt\explorer.exe**) to perform administrative functions.
2. Change the following registry entry

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell

from **explorer.exe** to **c:\asic\bin\oicfg.exe**. This will cause NT to boot the MMI with no explorer or desktop.

3. Rename **c:\winnt\system32\taskmgr.exe** to another name such as **taskmgrSave.exe**. This will prevent the operator from accessing the run command through the task manager from the Ctrl-Alt-Del sequence. This procedure will block all operator access to the operating system except the Ctrl-Alt-Del sequence. In this scenario, the Ctrl-Alt-Del sequence can only be used to log out, shut down the system, change the password, or lock the workstation (the task manager has been renamed).

Index

139168(C)

Xycom Automation, LLC.
750 North Maple Road
Saline, MI 48176–129
734-429-4971
Fax: 734-429-1010
http://www.profaceamerica.com